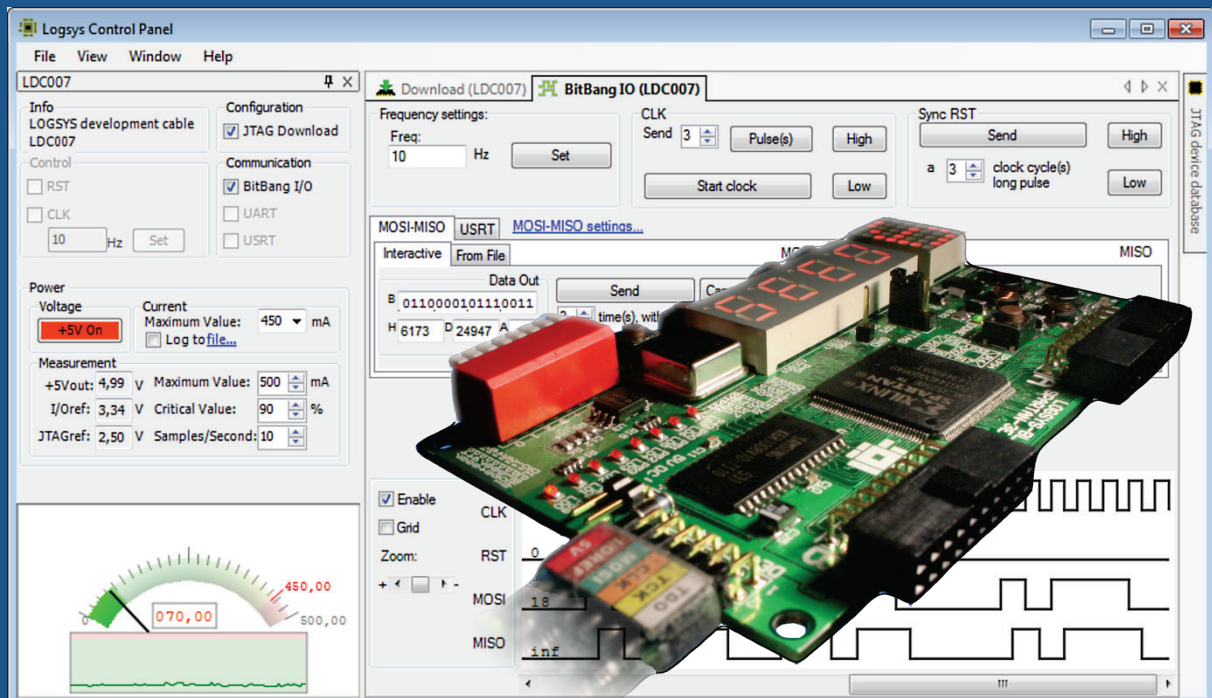




BME VILLAMOSMÉRNÖKI ÉS INFORMATIKAI KAR
MÉRÉSTECHNIKA ÉS INFORMÁCIÓS RENDSZEREK TANSZÉK

FEHÉR BÉLA
RAIKOVICH TAMÁS
LACZKÓ PÉTER

LOGSYS FPGA FEJLESZTŐI KÖRNYEZET



© Dr. Fehér Béla, Raikovich Tamás, Laczkó Péter, 2012

© Hungarian edition TERC Kft., 2012

ISBN 123 456 7890

Lektorálta: Lazányi János, Szántó Péter

A munka szakmai tartalma kapcsolódik a „Minőségorientált, összehangolt oktatási és K+F+I stratégia, valamint működési modell kidolgozása a Műegyetemen” c. projekt szakmai célkitűzéseinek megvalósításához.

A projekt megvalósítását az Új Széchenyi Terv TÁMOP-4.2.1/B-09/1/KMR-2010-0002 programja támogatja.

Kiadja a TERC Kereskedelmi és Szolgáltató Kft.

Szakkönyvkiadó Üzletága, az 1795-ben alapított Magyar Könyvkiadók és Könyvterjesztők Egyesülésének a tagja

Felelős kiadó: a kft. ügyvezető igazgatója

Felelős szerkesztő: Lévai-Kanyó Judit kiadóvezető

Kereskedelmi és marketingvezető: Augusztini-Szekeres Judit

A borítót tervezte: Dr. Fehér Béla és Laczkó Péter terve alapján Bagi Miklós

Tördelte: Sárközi Boglárka

Azonossági szám: JT-1204

Nyomtatás és kötés: Mesterprint Kft.

Felelős vezető: Szita Lajos ügyvezető igazgató

Minden jog fenntartva.

Tilos ezen kiadvány bármely részét sokszorosítani, információs rendszerben tárolni vagy sugározni bármely

formában vagy módon a kiadóval történt előzetes megállapodás nélkül; tilos továbbá terjeszteni másféle kötésben, borítóval és tördeléssel, mint amilyen formában eredetileg kiadásra került.

TERC Kereskedelmi és Szolgáltató Kft.

1149 Budapest, Pillangó park 9.

Tel.: 222-2402, Fax: 222-2405

E-mail: terc@terc.hu

www.terc.hu, www.terc.eu

Tartalomjegyzék

1	Előszó	1
2	LOGSYS FPGA alapkártyák	3
2.1	<i>Bevezetés</i>	5
2.2	<i>LOGSYS Spartan-3E FPGA kártya</i>	6
2.2.1	Memóriák	8
2.2.2	Megjelenítő eszközök	9
2.2.3	Beviteli eszközök	11
2.2.4	Órajel források.....	11
2.2.5	FPGA konfigurációs módok.....	11
2.2.6	LOGSYS fejlesztői port	12
2.2.7	Tápellátás	13
2.2.8	Bővítőcsatlakozók.....	13
2.2.9	Kapcsolási rajz	14
2.3	<i>LOGSYS Cyclone III FPGA kártya</i>	19
2.3.1	Memóriák	21
2.3.2	Megjelenítő eszközök	21
2.3.3	Beviteli eszközök	22
2.3.4	Órajel források.....	23
2.3.5	FPGA konfigurációs módok.....	23
2.3.6	LOGSYS fejlesztői port	24
2.3.7	Tápellátás	24
2.3.8	Bővítőcsatlakozók.....	24
2.3.9	Kapcsolási rajz	26
2.4	<i>LOGSYS ECP2 FPGA kártya</i>	31
2.4.1	Memóriák	33
2.4.2	Megjelenítő eszközök	33
2.4.3	Beviteli eszközök	34
2.4.4	Órajel források.....	34
2.4.5	FPGA konfigurációs módok.....	35
2.4.6	LOGSYS fejlesztői port	35
2.4.7	Tápellátás	36
2.4.8	Bővítőcsatlakozók.....	36
2.4.9	Kapcsolási rajz	37
3	LOGSYS fejlesztő kábel	43
3.1	<i>Bevezetés</i>	45
3.1.1	A LOGSYS fejlesztő kábel fontosabb jellemzői	45
3.2	<i>A LOGSYS fejlesztő kábel részletes ismertetése</i>	46
3.2.1	A LOGSYS fejlesztő kábel paraméterei	46
3.2.2	A LOGSYS fejlesztő kábel felépítése	47
3.2.3	A LOGSYS fejlesztő kábel blokkvázlata	51
3.3	<i>A LOGSYS fejlesztő kábel funkciói</i>	52
3.3.1	A tápellátási funkciók	52
3.3.2	A mérési funkciók	53
3.3.3	Konfigurációs és programozási funkciók	54
3.3.4	Gyártóspecifikus konfigurációs fájlok támogatása	55
3.4	<i>Kommunikációs funkciók</i>	55
3.4.1	Szoftveres (BitBang) adatátviteli mód	56

3.4.2	UART mód	58
3.4.3	USRT mód.....	59
3.4.4	SPI mód	59
3.4.5	I2C/TWI/SMBUS mód.....	59
3.4.6	Vezérlő jelek kiadása.....	59
4	LOGSYS grafikus felhasználói interfész	61
4.1	Bevezetés.....	63
4.2	A LOGSYS rendszer.....	63
4.3	A LOGSYS grafikus felhasználói felület általános ismertetése.....	64
4.4	A kábel panel.....	65
4.5	Konfiguráció letöltés, eszközök programozása	66
4.6	BitBang I/O adatkommunikáció	68
4.6.1	A BitBang I/O funkció modellje	69
4.6.2	Az órajel és a reset jel vezérlése	69
4.6.3	A BitBang I/O hullámforma grafikonja	70
4.6.4	BitBang I/O soros adatátvitel	71
4.6.5	BitBang I/O adatátviteli beállítási lehetőségek	75
4.7	Soros kommunikációs lehetőségek.....	76
4.7.1	A közvetlen UART és USRT adatátvitel.....	76
4.7.2	A regiszter szintű tesztinterfész	78
4.7.3	Memória szintű tesztinterfész	79
5	LOGSYS modulkészlet	81
5.1	LOGSYS VGA és PS/2 modul.....	83
5.1.1	VGA interfész	84
5.1.2	PS/2 interfész	85
5.1.3	Audio interfész	92
5.1.4	Kapcsolási rajz	93
5.2	LOGSYS karakteres LCD kijelző modul	94
5.2.1	Kommunikációs interfész	95
5.2.2	Memóriák az LCD vezérlőben.....	96
5.2.3	Parancskészlet.....	97
5.2.4	Az LCD kijelző inicializálása	99
5.2.5	Kapcsolási rajz	100
5.3	LOGSYS 10/100 Ethernet modul	101
5.3.1	Kommunikációs interfész	102
5.3.2	A modul használata	104
5.3.3	Kapcsolási rajz	106
5.4	LOGSYS sztereó CODEC modul.....	107
5.4.1	A modul működése	108
5.4.2	A CODEC konfigurációja	108
5.4.3	Időzítési paraméterek	109
5.4.4	Az MCLK és az LRCLK kapcsolata	110
5.4.5	Soros interfész.....	111
5.4.6	Kapcsolási rajz	112
5.5	LOGSYS hőmérő és EEPROM modul.....	113
5.5.1	LM75 – I ² C buszos hőmérő.....	114
5.5.2	TMP121 – SPI buszos hőmérő.....	119
5.5.3	M95080 – SPI buszos EEPROM	121
5.5.4	Kapcsolási rajz	126

1 Előszó

A digitális elektronikai rendszerek fejlesztésében a felhasználók által programozható eszközök (Filed Programmable Gate Array, FPGA) megjelenése jelentős változásokat hozott. A hagyományos diszkrét áramkörökkel, közepes- vagy nagybonyolultságú áramkörökkel történő építkezés helyébe a valódi rendszertervezés lépett, ami megtartva a funkcionális tervezés előnyeit, felhasználva a rendelkezésre álló korábbi saját vagy mások által tervezett egységeket (Intellectual Property, IP), megengedi az egyedi megoldásokat is a rendszer architektúra kialakításában. Miközben a közepes bonyolultságú eszközök is jelentős számú erőforrást biztosítanak a felhasználók számára, a gyártók legnagyobb eszközei ma már a millió logikai elemet jelentő határt is átlépték.

Ezzel az eszközökkel a tervezési technológia is jelentősen átalakult. A korábbi kapcsolási rajz alapú módszert leváltotta a szintézis alapú megvalósítás a hardver leíró nyelvek (Verilog, VHDL) teljes körű alkalmazásával. A HDL nyelvek jellemzően a regiszter transzfer szintű tervezést támogatják, ami a megfelelő specifikáció esetén a szintézis után megbízható működésű és jó paraméterekkel rendelkező technológiai realizációt eredményez.

Az FPGA-k a 80-as évek közepén jelentek meg és hamar széles körben elterjedtek. A BME VIK hamar felismerte a technológia jelentőségét és a nyolcvanas évek végén már egy úttörő jellegű oktatás és kutatásfejlesztési program eredményeképpen több tanszéken is biztosította az eszközök és fejlesztési módszertan megismerhetőségét. Az első években diplomatervek, önálló laboratóriumi projektek és különböző tudományos kutatások jelentették a technológia adoptálását. A legelső vizsgálatok az általános alkalmazásokon túlmenően az FPGA-k digitális jelfeldolgozásban történő alkalmazhatóságát vizsgálták. Az első publikációk különböző lineáris transzformációk és speciális digitális szűrők FPGA alapú megvalósíthatóságát ismertették.

A technológia széleskörű bevezetése az oktatásba a különböző szakokon eltérő ütemben, de egyre fokozódó mértékben történik. A Méréstechnika és Információs Rendszerek Tanszék által oktatott Beágyazott rendszerek szakirányon a területet bemutató Logikai tervezés már a hagyományos képzésben is súlyponti tárgy volt. A kétlépcsős képzés bevezetésekor a témakör részletes bemutatására az addig egyedi tárgyakban megjelenő ismereteket a Programozható logikai eszközök alkalmazástechnikája mellékszakirány tárgycsoportban egyesítettük. A tárgycsoport célja, hogy egységes keretben mutassa be a programozható logikai eszközökre épülő rendszermegvalósítási lehetőségeket az alkalmazási területhez tartozó összes tervezői ismeretet átfogva. A hallgatók megismerik az FPGA eszközök legfontosabb tulajdonságait, a hardver tervezés korszerű módszereit, a komplex egyetlen áramkörön realizált processzoros rendszerek (SoPC) kialakításának kérdéseit. Elsajátítják az integrált tervező rendszerek és fejlesztési technológiák használatát konkrét mérnöki tervezési problémák megoldásának lépésein keresztül.

A terület oktatói közössége egyetért a „gyakorlat teszi a mestert” szállóige tartalmával. Az elméleti ismeretek gyakorlati megalapozása azonban csak korszerű oktatástechnikai segédeszközökkel lehetséges. Ennek megfelelően a tanszéki FPGA laborban mindig törekedtünk a legkorszerűbb számítógépes tervező eszközök és fejlesztőkártyák biztosítására, de a hallgatók érdeklődését tapasztalva, úgy döntöttünk, hogy egy szélesebb körű elérhetőséget is biztosítunk. Az első tervek nagyjából egy évtizede alakultak ki a LOGSYS rendszerről, a prototípus tervezésében aktív hallgatói részvétellel (Molnár Gábor, Fülöp Tamás, Kiss Gergely, Zigó Attila: Digitális áramkörök oktatási és demonstrációs eszköze, TDK 2003). Az elképzelés azóta tovább finomodott, újabb TDK és diplomadolgozatok során sok más hallgató áldozatos munkájának

eredményeképpen érte el a mai verzióját. Érdemes hangsúlyozni, hogy a szerzők közül is többen hallgatóként kapcsolódtak be a LOGSYS rendszer fejlesztésébe.

A LOGSYS fejlesztői környezet a gyári CAD eszközök hasznos kiegészítőjeként a tervek megvalósításában nyújt jelentős segítséget. Az LOGSYS FPGA kártyák egységes formában biztosítják eltérő gyártók hasonló képességű eszközeinek alkalmazhatóságát, a LOGSYS programozó és kommunikációs kábelén keresztül a rendszer energiaellátását, konfigurálását és a kész terv ellenőrzését. Mindehhez egy olyan felhasználóbarát grafikus kezelői felület áll rendelkezésre, ami a teljesen kezdők elemi áramköreinek tesztelésétől a tapasztalt felhasználók akár többprocesszoros rendszereinek hardver és szoftver hibakeresését (debug) is magas szinten támogatja.

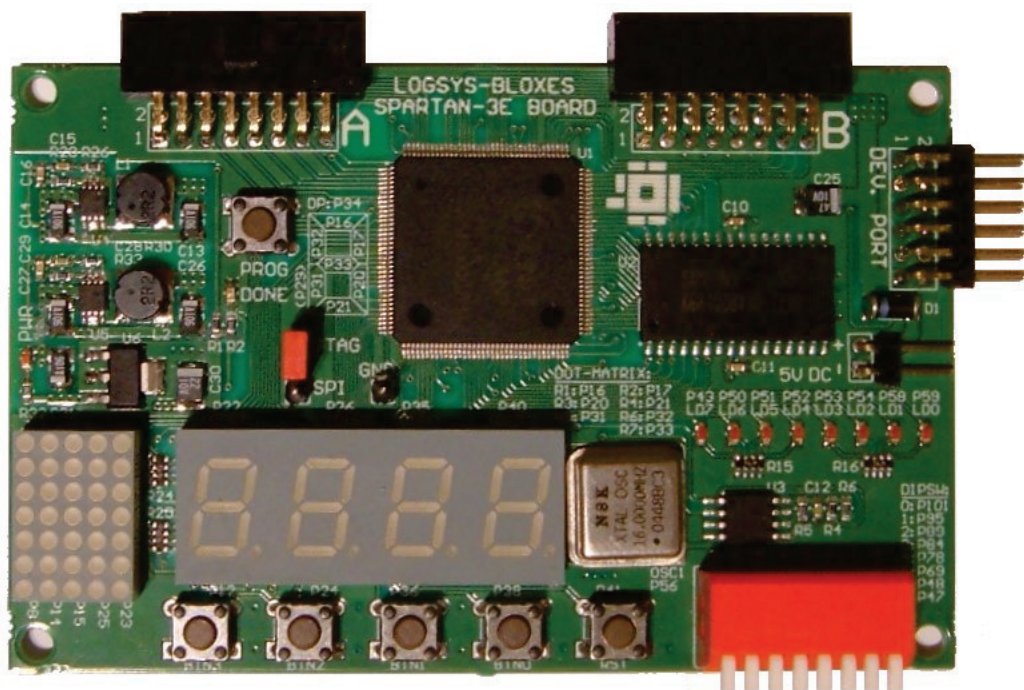
A könyvben ennek a fejlesztői környezetnek az elemeit ismertetjük. Az első három fejezet a rendszer főbb alkotóelemeit, az FPGA kártyákat, a programozó és kommunikációs kábelt és a használatot támogató grafikus felhasználói felületet ismerteti. Az FPGA kártyák a gyártási költségek és az egyszerű felhasználhatóság érdekében a legszükségesebb alkatrészeket és interfészeket tartalmazzák, de a bővítőcsatlakozókon keresztül számos kiegészítő modul képesek kezelni. Az így kialakított rendszer egyrészt lehetővé teszi, hogy a beágyazott rendszerek tervezése, alkalmazása során a gyakorlatban előforduló tipikus feladatok elvégezhetőek legyenek (interfészek kezelése, grafikus és szöveges megjelenítés, szenzorok használata), ugyanakkor a tetszőleges bővíthetőség is biztosított. A könyvben ismertetett alapmodulok (VGA és PS/2, LCD kijelző, Ethernet interfész, hőmérő, audio CODEC) a legáltalánosabb eseteket fedik le, de a léteznek WiFi, LED-mátrix panel, Bluetooth, CAN és egyéb modulok is. Ezek a modulok jellemzően hallgatói projektek keretében készülnek, a kapcsolódó hardvertervezési kérdések (analóg áramkörök, alkatrészválasztás, tápellátás, NYÁK tervezés, szerelés) egy konkrét feladat végrehajtása közbeni elsajátítása érdekében.

A LOGSYS FPGA fejlesztő környezet az eddigiek során igazolta az eredeti tervek és szándékok helyességét. Sok hallgató ismerte meg ezen keresztül az FPGA eszközökkel történő digitális rendszertervezés előnyeit, buktatóit, és tapasztalta meg a gondos tervezés fontosságát az esetleg nem működő első verziók tesztelése során. Reméljük az eszközök a továbbiakban is sok érdeklődő igényeit fogják kiszolgálni és csábítják érdekes feladatok megvalósítására.

Budapest, 2012. július 15.

A szerzők

2 LOGSYS FPGA alapkártyák



2.1 Bevezetés

A LOGSYS FPGA fejlesztői környezet központi elemei az FPGA alapkártyák. Az elsődleges cél egy jól használható rugalmas eszközkészlet kialakítása volt, amely elfogadható költségszint mellett biztosítja mindazokat a lehetőségeket, amik kihasználásával a modern eszközkészleten alapuló tervezési módszertanok bemutatathatók. A gyártók és más piacorientált fejlesztők által kínált eszközbázis igen széles spektrumot ölel át. Ezek között találhatók nagyon egyszerű, belépő szintű megoldásokat és gazdag szolgáltatásokat nyújtó, általános célú fejlesztő, demonstrációs eszközöket is. Természetesen minden FPGA gyártó igyekszik a saját eszközkészletének előnyeit hangsúlyozni, ennek megfelelően a kínált fejlesztő eszközök a gyártó specifikusak és hasonló, de nehezen összehasonlítható szolgáltatásokat nyújtanak.

A LOGSYS FPGA alapkártyák prototípusa a Xilinx Spartan-3E SRAM technológiájú FPGA család egy közepes bonyolultságú, hagyományos felületszerelt TQFP tokozású alkatrészére épül. Ezzel a választással sikerült egy olyan jellemző paraméter együttest kiválasztani, amihez más gyártók (Altera, Lattice) hasonló képességű elemei is illeszthetők, tehát létrejöhett egy olyan platform, ami lehetővé teszi azonos forrásszinten megadott specifikációk közvetlen realizálását eltérő gyártók egymással versengő típuscsaládjai között.

A LOGSYS FPGA alapkártyák mindegyik gyártótól nagyjából a harmadik generációs 90 – 65 nm technológiájú egyszerűsített felépítésű FPGA termékvonalát alkalmazzák. Ezek a termékek már jelentősen túlhaladják azt a korai a FPGA architektúrát, amelyben a logikai erőforrások egy szigorúan egységes, homogén alapstruktúra elemeit kínálják a fejlesztők számára. Ezekben az egyébként általános célú, költséghatékony alkalmazásokra optimalizált eszközökben is megjelennek különböző funkcionális blokkok (aritmetikai áramkörök, belső memória, szorzó, órajelszintézer), amik a rendszerfejlesztésnél jelentős erőforrás kihasználtság javulást biztosítanak.

A gyártók által biztosított tervelőkészítő és feldolgozó CAD rendszerek természetesen eltérnek egymástól, de a HDL szintű specifikációt mindhárom környezet hasonlóan értelmezi, fogadja. Az általuk generált konfigurációs fájlok is gyártófüggőek, de az egységes ipari szabvány SVF formátum mellett egységesen kezelhetők. A LOGSYS FPGA fejlesztői környezet egyik igen fontos tulajdonsága, hogy a mindezeket a specialitásokat kezelni tudja, megfelelő interfészeket biztosít és az eszközök használatához és egységes környezetben támogatja a technológia független fejlesztést.

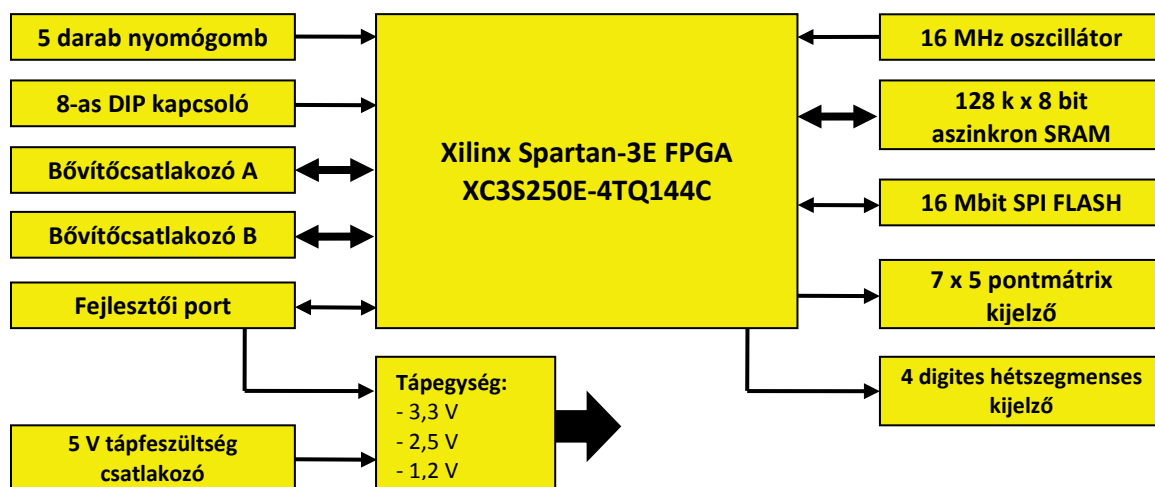
A LOGSYS FPGA kártyák a kiegészítő modulokkal együtt igen széles alkalmazói területen használhatók előzetes megvalósíthatósági vizsgálatokhoz, tesztkonfigurációk kialakításához és elsősorban oktatási feladatok esetén kezdő és középfeladatok különböző feladatainak megoldásához. A kártyák biztosítják a legfontosabb FPGA gyártók (Xilinx, Altera, Lattice) kiválasztott eszközeinek azonos feltételek melletti használatát, az egyedi LOGSYS modulokkal való bővíthetőség pedig a hardver rendszertervezés minden fázisának bemutatathatóságát. Az apróbb eltérések a kártyák felépítése között kizárólag az eszközök korlátainak kezeléséből fakadnak.

2.2 LOGSYS Spartan-3E FPGA kártya

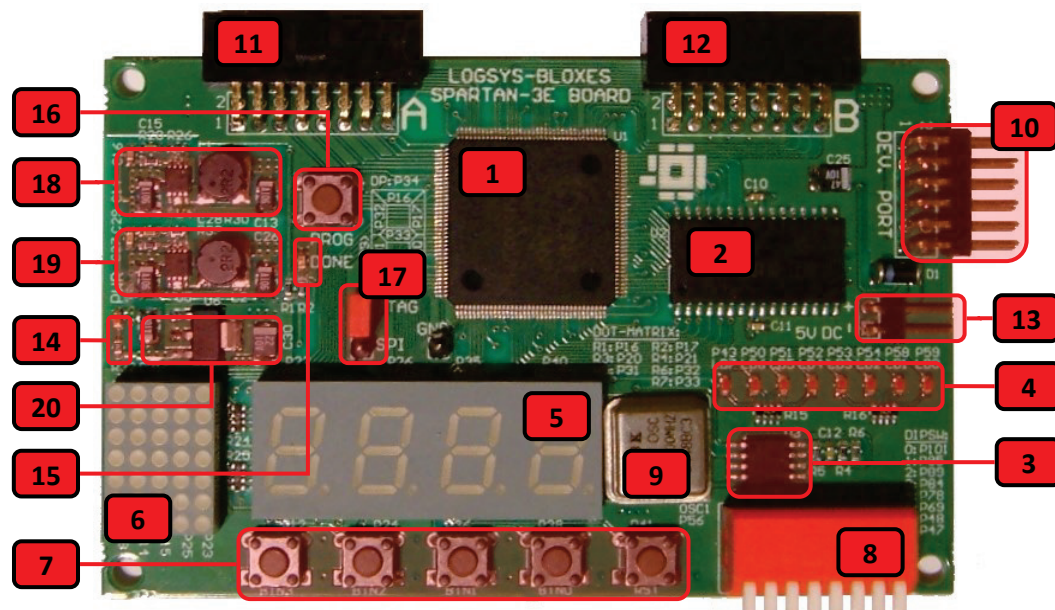
A LOGSYS Spartan-3E FPGA kártya egy egyszerű felépítésű, elsősorban kezdő felhasználók számára készült FPGA kártya. A felhasznált FPGA-nak köszönhetően azonban alkalmas összetettebb tervek megvalósítására is. A kártya blokkvázlata a 2-1. ábrán látható, a kártya felépítését a 2-2. ábra szemlélteti.

A kártyán az alábbi komponensek találhatók:

- Xilinx XC3S250E-4TQ144C típusú FPGA, amely lehetővé teszi összetettebb logikák és kisebb mikroproceszoros rendszerek megvalósítását. Az eszköz főbb jellemzői:
 - 250 ezer kapu (4896 4 bemenetű LUT és flip-flop)
 - 12 darab 18 kbytes blokk-RAM
 - 12 darab 18 x 18 bites előjeles szorzó
 - 4 darab DCM (Digital Clock Manager) modul
- Memóriák a program és az adatok tárolására:
 - Egy 128 k x 8 bites, 10 ns-os aszinkron SRAM (Samsung K6R1008V1D-TI10)
 - Egy 16 Mbytes SPI buszos soros FLASH memória (Winbond W25P16 vagy W25X16)
 - A soros FLASH memória konfigurációs memóriaként is szolgál az FPGA számára
- Megjelenítő eszközök:
 - 8 darab LED
 - 4 digitális hétszegmenses kijelző
 - 7 x 5 pontmátrix kijelző
- Beviteli eszközök:
 - 5 darab nyomógomb
 - 8-as DIP kapcsoló
- Egy 16 MHz-es oszcillátor
- Csatlakozó a LOGSYS fejlesztői kábel számára
- 2 darab csatlakozó a kiegészítő modulok számára:
 - 13 FPGA I/O láb (11 kétirányú, 2 csak bemenet)
 - 5 V és 3,3 V tápfeszültség kimenet



2-1. ábra: A LOGSYS Spartan-3E FPGA kártya blokkvázlata.



2-2. ábra: A LOGSYS Spartan-3E FPGA kártya.

A LOGSYS Spartan-3E FPGA kártya felépítése:

1. Xilinx XC3S250E-4TQ144C típusú FPGA
2. 128 k x 8 bites, 10 ns hozzáférési idejű aszinkron SRAM
3. Winbond W25P16 vagy W25X16 típusú 16 Mbytes SPI buszos soros FLASH
4. 8 darab LED
5. 4 digit-es hétszegmenses kijelző
6. 7 x 5 pontmátrix kijelző
7. 5 darab nyomógomb
8. 8-as DIP kapcsoló
9. 16 MHz-es oszcillátor
10. Csatlakozó a LOGSYS fejlesztői kábel számára (fejlesztői port)
11. Csatlakozó a kiegészítő modulok számára (A)
12. Csatlakozó a kiegészítő modulok számára (B)
13. 5 V tápfeszültség csatlakozó
14. A bekapcsolt tápfeszültséget jelző piros LED (PWR)
15. Az FPGA sikeres felkonfigurálását jelző zöld LED (DONE)
16. Az FPGA újrakonfigurálását elindító nyomógomb (PROG)
17. Az FPGA konfigurációs módját kiválasztó jumper
18. 3,3 V feszültséget előállító tápegység
19. 1,2 V feszültséget előállító tápegység
20. 2,5 V feszültséget előállító tápegység

2.2.1 Memóriák

2.2.1.1 Aszinkron SRAM

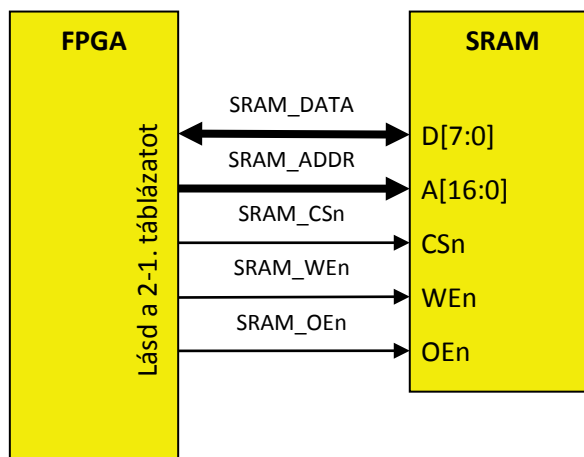
Az FPGA kártyán található Samsung K6R1008V1D-TI10 típusú 128 k x 8 bites, 10 ns elérési idejű aszinkron SRAM bekötését a 2-1. táblázat és a 2-3. ábra mutatja. A vezérlő jelek (CSn, WEn és OEn) aktív alacsony szintűek.

2-1. táblázat: Az SRAM bekötése.

Adatbusz	
Jel	FPGA láb
D0	P60
D1	P74
D2	P75
D3	P76
D4	P93
D5	P94
D6	P106
D7	P105

Vezérlő jelek	
Jel	FPGA láb
CSn	P62
WEn	P77
OEn	P104

Címbusz	
Jel	FPGA láb
A0	P70
A1	P68
A2	P67
A3	P66
A4	P81
A5	P82
A	P83
A7	P85
A8	P86
A9	P87
A10	P88
A11	P91
A12	P92
A13	P103
A14	P98
A15	P97
A16	P96



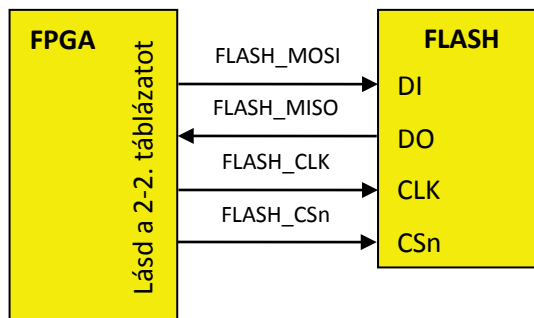
2-3. ábra: Az SRAM bekötése.

2.2.1.2 SPI buszos soros FLASH memória

A LOGSYS Spartan-3E FPGA kártyán található Winbond W25P16 vagy W25X16 típusú 16 Mbytes SPI buszos soros FLASH memória bekötését a 2-2. táblázat és a 2-4. ábra mutatja. A soros FLASH memória konfigurációs memóriaként is szolgál az FPGA számára. Az eszköz működéséről és használatáról részletesen annak adatlapjában olvashatunk, amely letölthető a gyártó honlapjáról: <http://www.winbond.com>. A CSn chip select jel aktív alacsony szintű.

2-2. táblázat: A FLASH memória bekötése.

FLASH	FPGA láb
DI (MOSI)	P44
DO (MISO)	P63
CLK	P71
CSn	P39



2-4. ábra: A FLASH memória bekötése.

2.2.2 Megjelenítő eszközök

2.2.2.1 LED-ek

A LOGSYS Spartan-3E FPGA kártyán található 8 darab LED bekötését a 2-3. táblázat mutatja. A LED-ek LD0-tól LD7-ig vannak számozva, a bal szélső LED az LD7, a jobb szélső LED az LD0. A LED-ek vezérlő jelei aktív magas szintűek.

2-3. táblázat: A LED-ek bekötése.

LED	LD7	LD6	LD5	LD4	LD3	LD2	LD1	LD0
FPGA láb	P43	P50	P51	P52	P53	P54	P58	P59

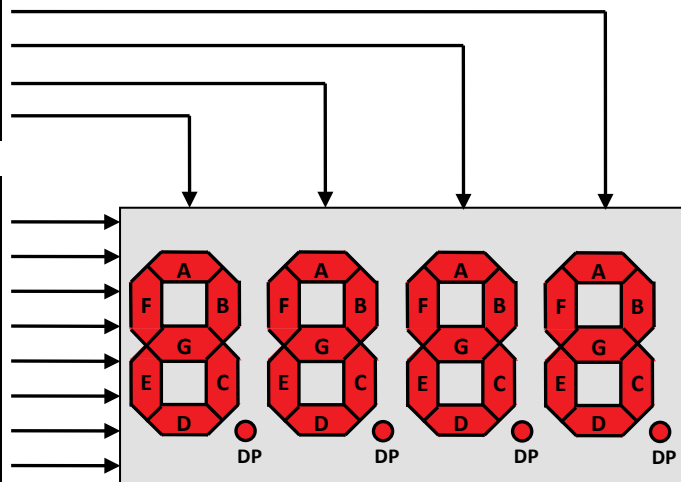
2.2.2.2 Hétszegmenses kijelző

A LOGSYS Spartan-3E FPGA kártyán található 4 digitis hétszegmenses kijelző bekötését a 2-4. táblázat és a 2-5. ábra mutatja. A karakterek DIG0-tól DIG3-ig vannak számozva, a bal szélső karakter a DIG3, a jobb szélső karakter a DIG0. A hétszegmenses kijelző minden vezérlő jele aktív alacsony szintű. A szegmens jelek és a pontmátrix kijelző sor jelei közösek.

2-4. táblázat: A hétszegmenses kijelző bekötése.

Kiválasztó jel	FPGA láb
DIGn0	P40
DIGn1	P35
DIGn2	P26
DIGn3	P22

Szegmens	FPGA láb
SEGn0 (A)	P16
SEGn1 (B)	P17
SEGn2 (C)	P20
SEGn3 (D)	P21
SEGn4 (E)	P31
SEGn5 (F)	P32
SEGn6 (G)	P33
DP (tizedespont)	P34



2-5. ábra: A hétszegmenses kijelző bekötése.

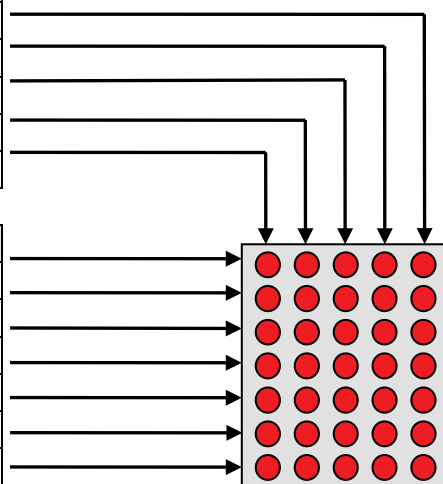
2.2.2.3 Pontmátrix kijelző

A LOGSYS Spartan-3E FPGA kártyán található 7 x 5-ös pontmátrix kijelző bekötését a 2-5. táblázat és a 2-6. ábra mutatja. A sorok jelölése fentről lefelé rendre ROW1-ROW7. Az oszlopok COL4-tól COL0-ig vannak számozva, a bal szélső oszlop a COL4, a jobb szélső oszlop a COL0. A pontmátrix kijelző minden vezérlő jele aktív alacsony szintű. A sor jelek és a hétszegmenses kijelző szegmens jelei közösek.

2-5. táblázat: A pontmátrix kijelző bekötése.

Kiválasztó jel	FPGA láb
COLn0	P23
COLn1	P25
COLn2	P15
COLn3	P14
COLn4	P8

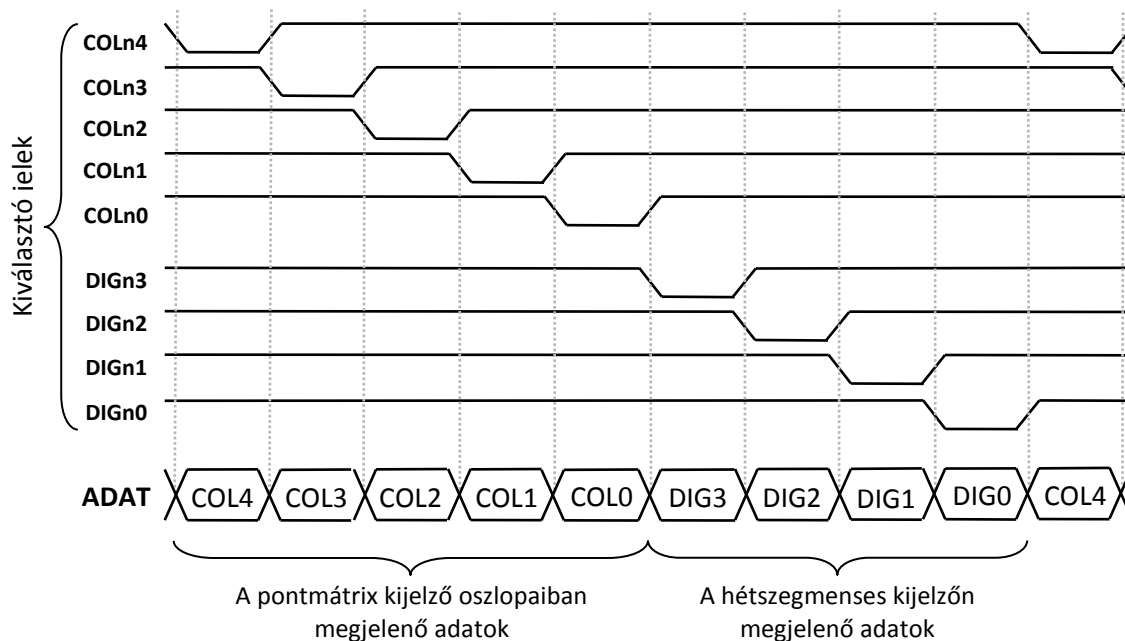
Sor	FPGA láb
SEGn0 (ROW1)	P16
SEGn1 (ROW2)	P17
SEGn2 (ROW3)	P20
SEGn3 (ROW4)	P21
SEGn4 (ROW5)	P31
SEGn5 (ROW6)	P32
En6 (ROW7)	P33



2-6. ábra: A pontmátrix kijelző bekötése.

2.2.2.4 Időmultiplexelt vezérlés

A kijelzők vezérlése időmultiplexelt módon lehetséges a 2-7. ábra szerinti módon. A két kijelző esetén hét vezérlőjel közös, ezekkel lehet a hétszegmenses kijelző egyes szegmenseit, illetve a pontmátrix kijelző oszlopaiban található LED-eket bekapcsolni. Minden egyes karakter, illetve oszlop külön kiválasztó (anódvezérlő) jellel rendelkezik. A két kijelző természetesen önállóan is használható, ebben az esetben a hétszegmenses kijelzőhöz csak 4 ütemű, a pontmátrix kijelzőhöz csak 5 ütemű időmultiplexelt vezérlést kell használni.



2-7. ábra: A kijelzők időmultiplexelt vezérlése.

2.2.3 Beviteli eszközök

2.2.3.1 DIP kapcsoló

A LOGSYS Spartan-3E FPGA kártyán található 8-as DIP kapcsoló bekötését a 2-6. táblázat mutatja. A kapcsolók 0-tól 7-ig vannak számozva, a bal szélső kapcsoló sorszáma a 7, a jobb szélső kapcsoló sorszáma a 0. Az adott FPGA láb a kapcsoló alsó állásában logikai alacsony szintű (0 V), a kapcsoló felső állásában pedig logikai magas szintű (3,3 V) lesz.

2-6. táblázat: A DIP kapcsoló bekötése.

Kapcsoló	7	6	5	4	3	2	1	0
FPGA láb	P47	P48	P69	P78	P84	P89	P95	P101

2.2.3.2 Nyomógombok

A LOGSYS Spartan-3E FPGA kártyán található 5 darab nyomógomb bekötését a 2-7. táblázat mutatja. A nyomógombok jelölése balról jobbra rendre BTN3-BTN0 és RST. Az adott FPGA lábra logikai magas szint (3,3 V) kerül a nyomógomb megnyomása esetén. Az RST gomb elsősorban az alaphelyzetbe állításra szolgál, de tetszőlegesen is felhasználható.

2-7. táblázat: A nyomógombok bekötése.

Nyomógomb	BTN3	BTN2	BTN1	BTN0	RST
FPGA láb	P12	P24	P36	P38	P41

2.2.4 Órajel források

Az FPGA a kártyán lévő 16 MHz-es oszcillátortól és a fejlesztői port CLK vonaláról kaphat órajelet. Mindkettő az FPGA egy-egy órajel bemeneti lábára (GCLK) csatlakozik. Az oszcillátor 16 MHz-es órajeléből az FPGA-ban található DCM (Digital Clock Manager) modulok segítségével egyéb frekvenciák is előállíthatók. Az órajel források bekötését a 2-8. táblázat mutatja.

2-8. táblázat: Az órajel források bekötése.

Órajel forrás	FPGA láb
16 MHz-es oszcillátor	P56
Fejlesztői port CLK vonala	P129

2.2.5 FPGA konfigurációs módok

A LOGSYS Spartan-3E FPGA kártya esetén kétféle konfigurációs mód lehetséges. Az FPGA felkonfigurálható a fejlesztői port JTAG interfészén keresztül, illetve az eszköz képes magát felkonfigurálni a kártyán lévő SPI buszos soros FLASH memóriából is. A konfigurációs mód egy jumperrel választható ki (lásd a 2-9. táblázatot). A JTAG interfész a kiválasztott módtól függetlenül mindig rendelkezésre áll.

2-9. táblázat: Az FPGA lehetséges konfigurációs módjai

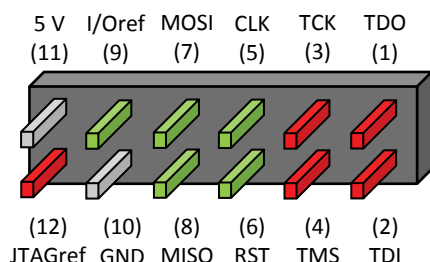
Jumper állása	Konfigurációs mód	Leírás
	JTAG	Az FPGA-t a JTAG interfészen keresztül kell felkonfigurálni.
	SPI	Az FPGA az SPI buszos soros FLASH memóriából konfigurálja fel magát a tápfeszültség bekapcsolása vagy a PROG gomb megnyomása után.

2.2.6 LOGSYS fejlesztői port

A LOGSYS fejlesztői kábel a fejlesztői porton keresztül illeszthető a kártyához. A fejlesztői port az alábbi interfészekkel rendelkezik:

- JTAG interfész: TDI, TDO, TCK és TMS vonalak
- Vezérlési interfész:
 - CLK órajel bemenet a fejlesztői kábeltől az FPGA felé
 - RST reset bemenet a fejlesztői kábeltől az FPGA felé
- Soros kommunikációs interfész:
 - MOSI soros adat bemenet a fejlesztői kábeltől az FPGA felé
 - MISO soros adat kimenet az FPGA-tól a fejlesztői kábel felé
- Tápellátás:
 - 5 V tápfeszültség bemenet
 - Referenciafeszültség kimenetek a fejlesztői kábel számára: I/Oref, JTAGref

A fejlesztői port tűsorosának lábkiosztása a 2-8. ábrán látható. A vezérlési és a soros kommunikációs vonalak bekötését a 2-10. táblázat mutatja. Mivel a MOSI, a CLK és az RST bemeneteken nincs külső lehúzó ellenállás, ezért használatuk esetén a hozzájuk tartozó FPGA I/O lábakon engedélyezni kell a belső lehúzó ellenállást (lásd az UCF fájlt). Ellenkező esetben ezek a bemenetek lebegni fognak és a letöltött terv nem fog megfelelően működni. A MISO kimenetre egy külső 4,7 k Ω -os felhúzó ellenállás van kötve.



2-8. ábra: A fejlesztői port tűsorosának lábkiosztása.

2-10. táblázat: A fejlesztői port bekötése.

Jel	Írány	FPGA láb
MOSI	bemenet	P120
MISO	kimenet	P143
CLK	bemenet	P129
RST	bemenet	P119

2.2.7 Tápellátás

Az FPGA kártya 5 V-os tápfeszültséget igényel. A tápellátás alapvetően a fejlesztői kábelről történik, de lehetőség van egyéb külső 5 V-os egyenfeszültség forrás csatlakoztatására is. Az 5 V DC tápcsatlakozó védett a fordított polaritású bekötés ellen.

Az FPGA a működéséhez 3,3 V-os (I/O vonalak), 2,5 V-os (JTAG, DCM) és 1,2 V-os (belső mag) tápfeszültséget igényel. Ezeket a feszültségeket az 5 V-os tápfeszültségből külön tápegységek állítják elő. A kártyán található összes periféria és az I/O vonalak a JTAG interfész kivételével 3,3 V-ról működnek, a fejlesztői kábel 3,3 V-os I/O referenciafeszültséget (I/Oref) kap a fejlesztői porton keresztül. A JTAG interfész 2,5 V-os feszültségről működik, a fejlesztői kábel 2,5 V-os JTAG referenciafeszültséget (JTAgref) kap a fejlesztői porton keresztül.

2.2.8 Bővítőcsatlakozók

A LOGSYS Spartan-3E FPGA kártyához a kiegészítő modulok illesztését két 16 pólusú csatlakozó teszi lehetővé. Mindkét csatlakozó lábkiosztása azonos, ez a kártya szerinti nézetből a 2-9. ábrán látható. A csatlakozókra ki van vezetve a 3,3 V-os és az 5 V-os tápfeszültség is, azonban az adatvonalak 3,3 V-ról működnek és nem 5 V toleránsak. A 13 adatvonal közül 11 ténylegesen kétirányú, 2 viszont csak bemenet. A bővítőcsatlakozók bekötését a 2-11. táblázat és a 2-12. táblázat mutatja.



(15) Input	(13) I/O	(11) I/O	(9) I/O	(7) I/O	(5) I/O	(3) +3,3V	(1) GND
(16) Input	(14) I/O	(12) I/O	(10) I/O	(8) I/O	(6) I/O	(4) I/O	(2) +5V

2-9. ábra: A bővítőcsatlakozók lábkiosztása.

2-11. táblázat: Az A bővítőcsatlakozó bekötése.

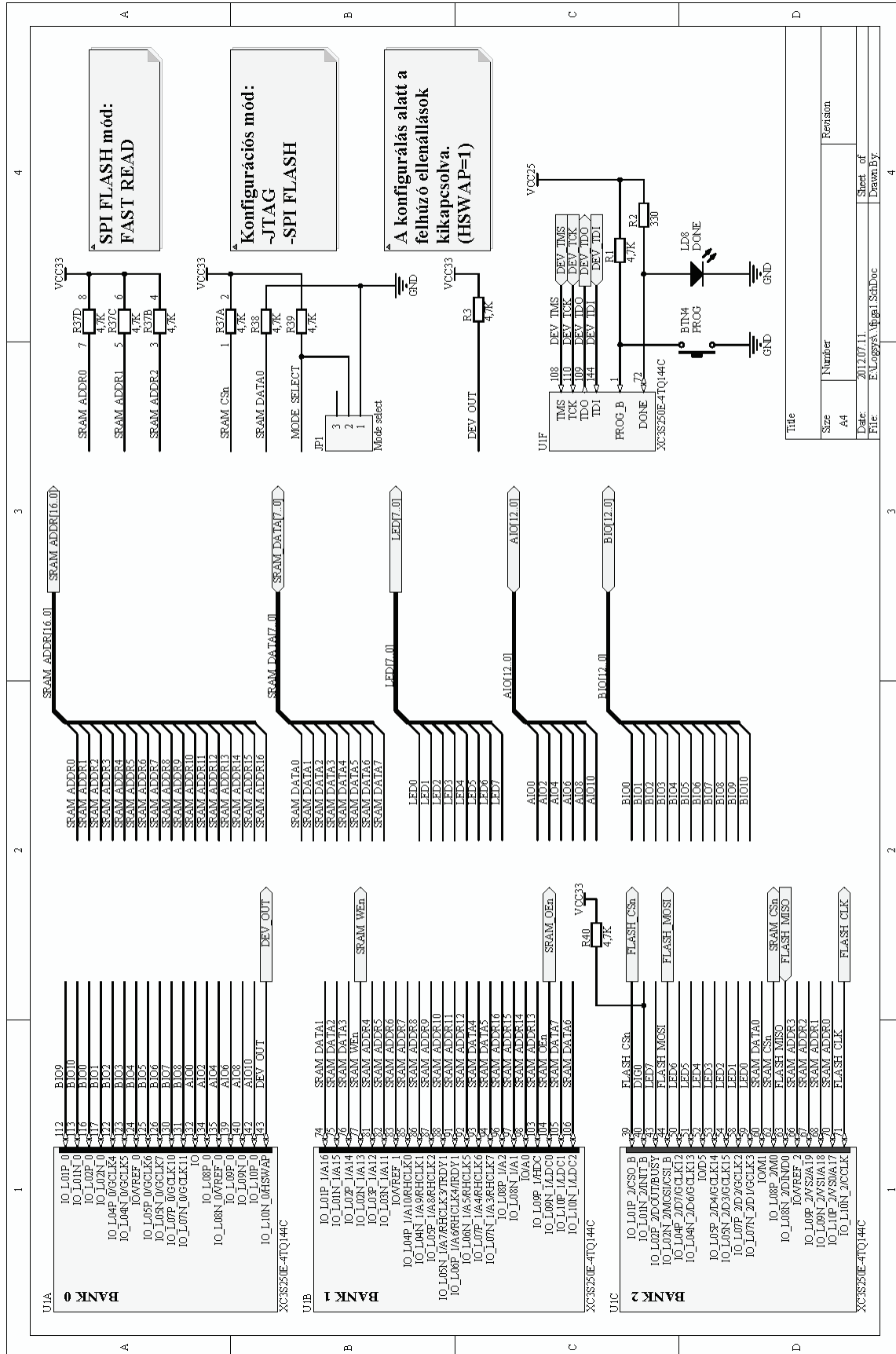
A bővítőcsatlakozó		
Csatlakozó láb	Írány	FPGA láb
4	kétirányú	P132
5	kétirányú	P7
6	kétirányú	P134
7	kétirányú	P5
8	kétirányú	P135
9	kétirányú	P4
10	kétirányú	P139
11	kétirányú	P3
12	kétirányú	P140
13	kétirányú	P2
14	kétirányú	P142
15	bemenet	P6
16	bemenet	P141

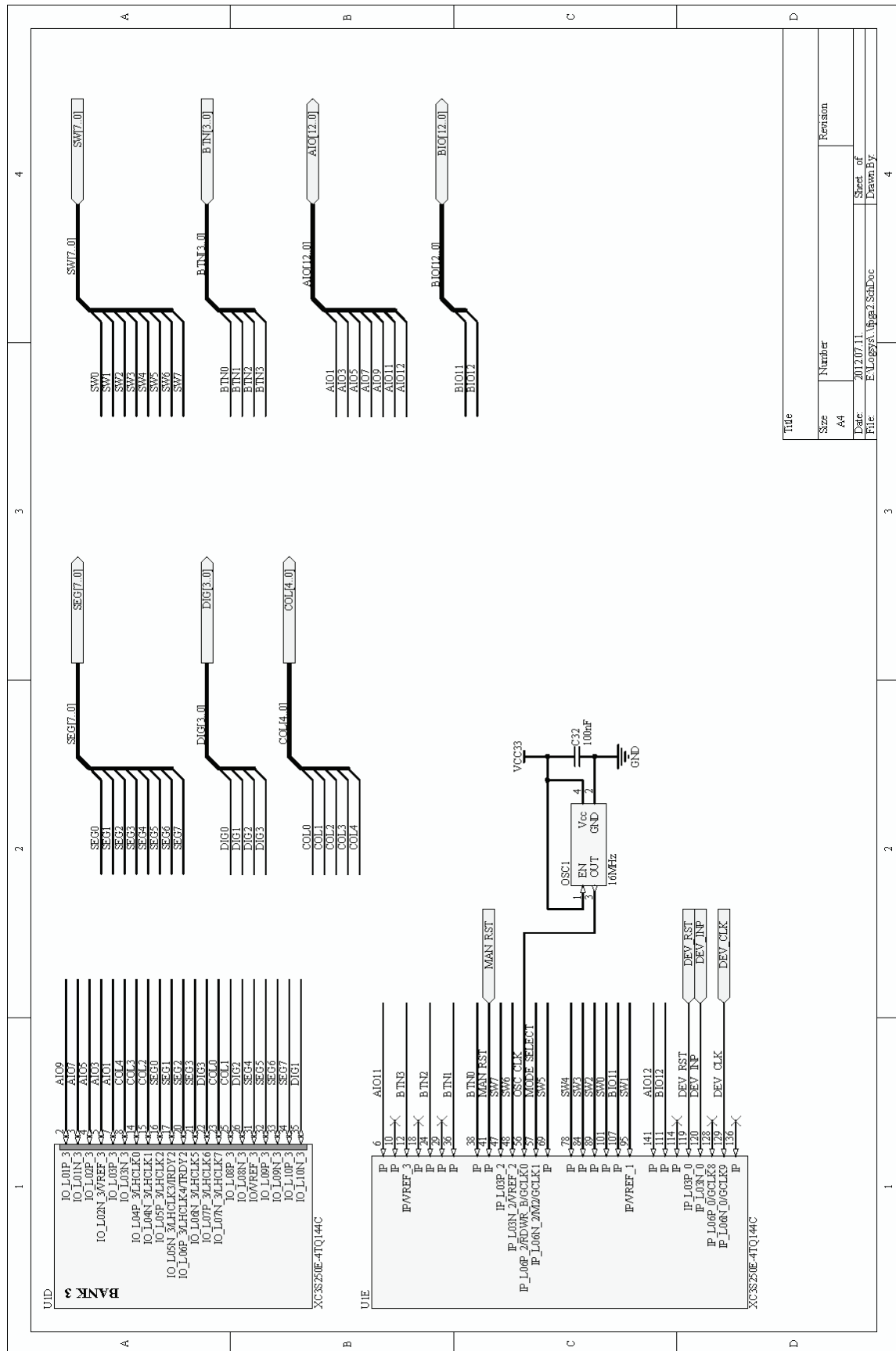
2-12. táblázat: A B bővítőcsatlakozó bekötése.

B bővítőcsatlakozó		
Csatlakozó láb	Írány	FPGA láb
4	kétirányú	P116
5	kétirányú	P117
6	kétirányú	P122
7	kétirányú	P123
8	kétirányú	P24
9	kétirányú	P125
10	kétirányú	P126
11	kétirányú	P130
12	kétirányú	P131
13	kétirányú	P112
14	kétirányú	P113
15	bemenet	P107
16	bemenet	P111

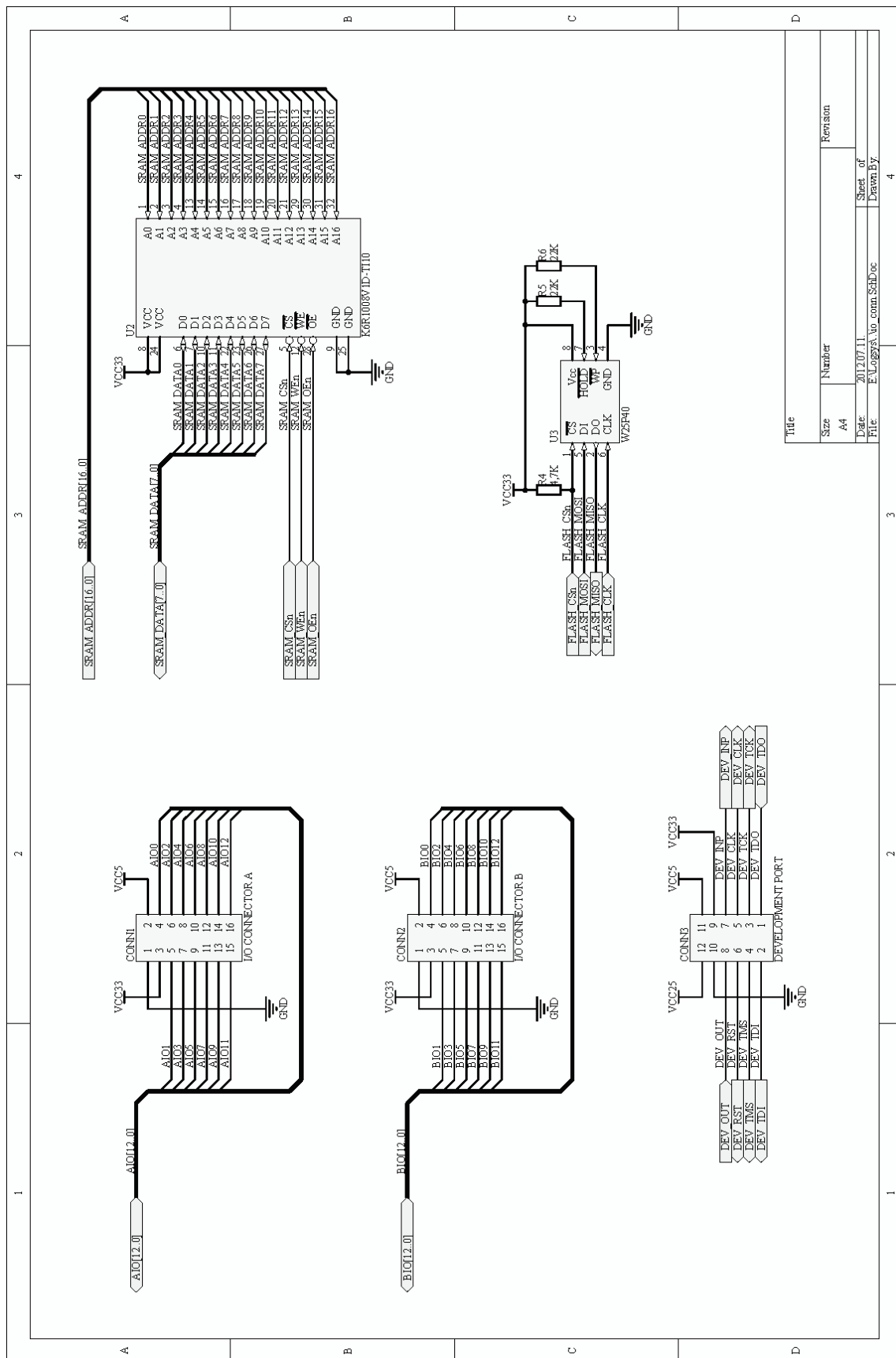
2.2.9 Kapcsolási rajz

2.2.9.1 FPGA

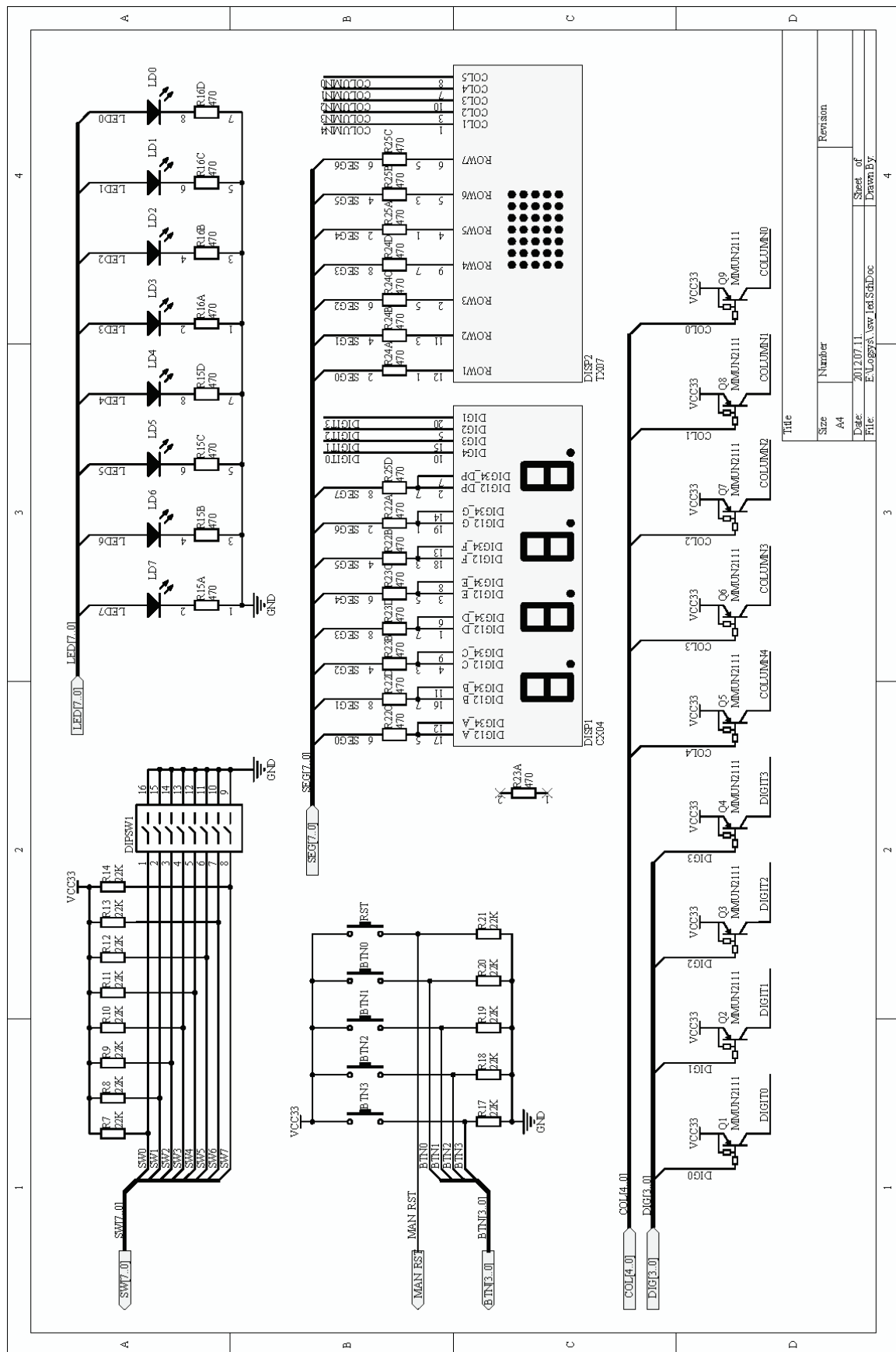




2.2.9.2 Csatlakozók, memóriák



2.2.9.3 LED-ek, kijelzők, DIP kapcsoló, nyomógombok



2.2.9.4 Tápegység

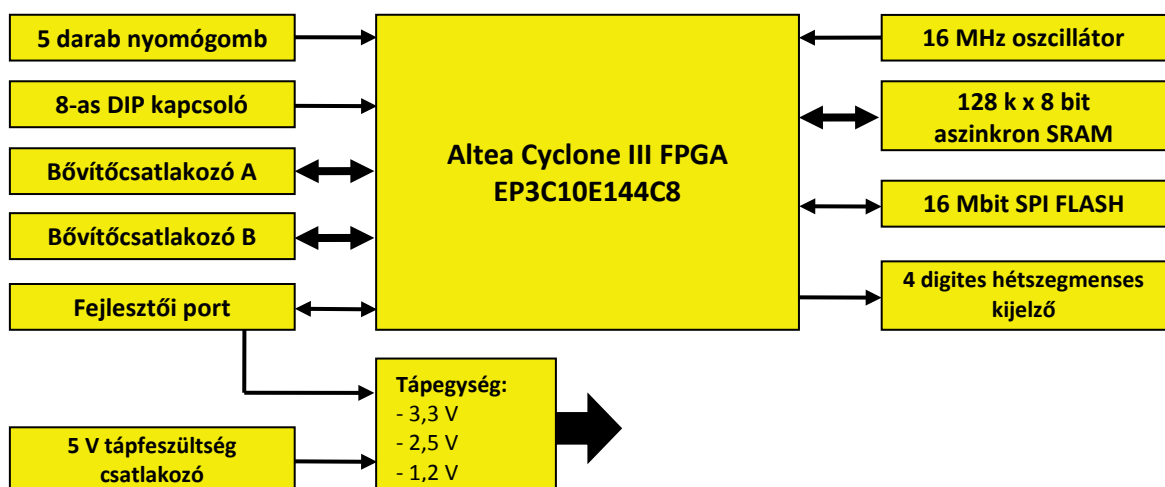


2.3 LOGSYS Cyclone III FPGA kártya

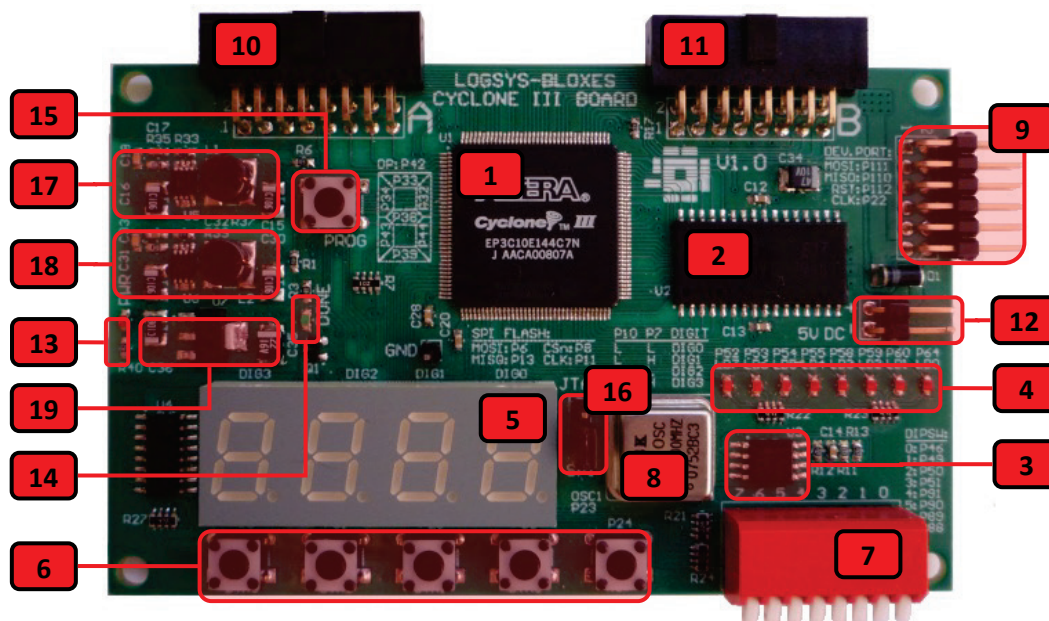
A LOGSYS Cyclone III FPGA kártya a LOGSYS Spartan-3E FPGA kártyához hasonlóan egy egyszerű felépítésű, elsősorban kezdő felhasználók számára készült FPGA kártya, amely alkalmas összetettebb tervek megvalósítására is. A kártya blokkvázlata a 2-10. ábrán látható, a kártya felépítését a 2-11. ábra szemlélteti.

A kártyán az alábbi komponensek találhatók:

- Altera EP3C10E144C8 típusú FPGA, amely lehetővé teszi összetettebb logikák és kisebb mikroproceszoros rendszerek megvalósítását. Az eszköz főbb jellemzői:
 - 10320 darab 4 bemenetű LUT és flip-flop
 - 46 darab 9 kbytes blokk-RAM
 - 23 darab 18 x 18 bites előjeles szorzó
 - 2 darab PLL
- Memóriák a program és az adatok tárolására:
 - Egy 128 k x 8 bites, 10 ns-os aszinkron SRAM (Samsung K6R1008V1D-TI10)
 - Egy 16 Mbytes SPI buszos soros FLASH memória (Winbond W25P16 vagy W25X16)
 - A soros FLASH memória konfigurációs memóriaként is szolgál az FPGA számára
- Megjelenítő eszközök:
 - 8 darab LED
 - 4 digitális hétszegmenses kijelző
- Beviteli eszközök:
 - 5 darab nyomógomb
 - 8-as DIP kapcsoló
- Egy 16 MHz-es oszcillátor
- Csatlakozó a LOGSYS fejlesztői kábel számára
- 2 darab csatlakozó a kiegészítő modulok számára:
 - 13 FPGA I/O láb (11 kétirányú, 2 csak bemenet)
 - 5 V és 3,3 V tápfeszültség kimenet



2-10. ábra: A LOGSYS Cyclone III FPGA kártya blokkvázlata.



2-11. ábra: A LOGSYS Cyclone III FPGA kártya.

A LOGSYS Cyclone III FPGA kártya felépítése:

1. Altera EP3C10E144C8 típusú FPGA
2. 128 k x 8 bites, 10 ns hozzáférési idejű aszinkron SRAM
3. Winbond W25P16 vagy W25X16 típusú 16 Mbytes SPI buszos soros FLASH
4. 8 darab LED
5. 4 digitális hétszegmenses kijelző
6. 5 darab nyomógomb
7. 8-as DIP csatlakozó
8. 16 MHz-es oszcillátor
9. Csatlakozó a LOGSYS fejlesztői kábel számára (fejlesztői port)
10. Csatlakozó a kiegészítő modulok számára (A)
11. Csatlakozó a kiegészítő modulok számára (B)
12. 5 V tápfeszültség csatlakozó
13. A bekapcsolt tápfeszültséget jelző piros LED (PWR)
14. Az FPGA sikeres felkonfigurálását jelző zöld LED (DONE)
15. Az FPGA újrakonfigurálását elindító nyomógomb (PROG)
16. Az FPGA konfigurációs módját kiválasztó jumper
17. 3,3 V feszültséget előállító tápegység
18. 1,2 V feszültséget előállító tápegység
19. 2,5 V feszültséget előállító tápegység

2.3.1 Memóriák

A LOGSYS Spartan-3E FPGA kártyához hasonlóan a LOGSYS Cyclone III FPGA kártyán is a Samsung K6R1008V1D-TI10 típusú 128 k x 8 bites, 10 ns elérési idejű aszinkron SRAM és a Winbond W25P16 (vagy W25X16) típusú 16 Mbites SPI buszos soros FLASH memória található meg. A memóriák bekötését a 2-13. táblázat és a 2-14. táblázat mutatja. A soros FLASH memória konfigurációs memóriaként is szolgál az FPGA számára.

2-13. táblázat: Az SRAM bekötése.

Adatbusz		Címbusz	
Jel	FPGA láb	Jel	FPGA láb
D0	P66	A0	P71
D1	P65	A1	P70
D2	P72	A2	P69
D3	P73	A3	P68
D4	P87	A4	P75
D5	P98	A5	P76
D6	P99	A6	P77
D7	P100	A7	P79
		A8	P80
		A9	P83
		A10	P84
		A11	P85
		A12	P86
		A13	P105
		A14	P104
		A15	P103
		A16	P101

Vezérlő jelek	
Jel	FPGA láb
CSn	P67
WEn	P74
OEn	P106

2-14. táblázat: A FLASH memória bekötése.

FLASH	FPGA láb
DI (MOSI)	P6
DO (MISO)	P13
CLK	P11
CS	P8

2.3.2 Megjelenítő eszközök

2.3.2.1 LED-ek

A LOGSYS Cyclone III FPGA kártyán található 8 darab LED bekötését a 2-15. táblázat mutatja. A LED-ek LD0-tól LD7-ig vannak számozva, a bal szélső LED az LD7, a jobb szélső LED az LD0. A LED-ek vezérlő jelei aktív magas szintűek.

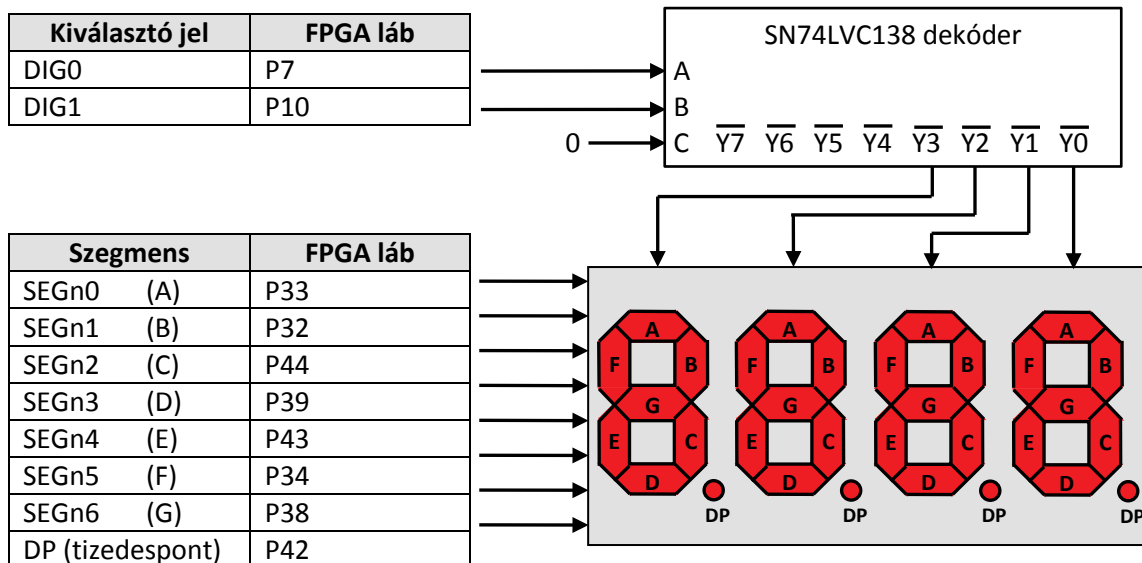
2-15. táblázat: A LED-ek bekötése.

LED	LD7	LD6	LD5	LD4	LD3	LD2	LD1	LD0
FPGA láb	P52	P53	P54	P55	P58	P59	P60	P64

2.3.2.2 Hétszegmenses kijelző

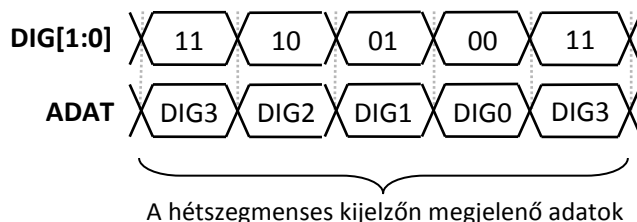
A LOGSYS Cyclone III FPGA kártyán található 4 digités hétszegmenses kijelző bekötését a 2-16. táblázat és a 2-12. ábra mutatja. A karakterek DIG0-tól DIG3-ig vannak számozva, a bal szélső karakter a DIG3, a jobb szélső karakter a DIG0. A hétszegmenses kijelző szegmensvezérlő jelei aktív alacsony szintűek. A karakter kiválasztó jeleket egy külső SN74LVC138 típusú dekóder állítja elő.

2-16. táblázat: A hétszegmenses kijelző bekötése.



2-12. ábra: A hétszegmenses kijelző bekötése.

Az alkalmazott külső dekóder miatt a kijelző időmultiplexelt vezérlése (2-13. ábra) eltér a 2.2.2.4. pontban ismertetett módtól. A kijelző esetén hét vezérlőjel közös, ezekkel lehet az egyes szegmensekhez tartozó LED-eket bekapcsolni. A kétbites DIG[1:0] kiválasztó jel segítségével lehet az adott sorszámú karaktert engedélyezni a kijelzőn (egyszerre csak egy karakter lehet aktív). A DIG[1:0] jel az SN74LVC138 dekóder bemenetére kapcsolódik, amely előállítja a karakterek számára az aktív-alacsony kiválasztó (anódvezérlő) jelet.



2-13. ábra: A hétszegmenses kijelző időmultiplexelt vezérlése.

2.3.3 Beviteli eszközök

2.3.3.1 DIP kapcsoló

A LOGSYS Cyclone III FPGA kártyán található 8-as DIP kapcsoló bekötését a 2-17. táblázat mutatja. A kapcsolók 0-tól 7-ig vannak számozva, a bal szélső kapcsoló sorszáma a 7, a jobb szélső kapcsoló sorszáma a 0. Az adott FPGA láb a kapcsoló alsó állásában logikai alacsony szintű (0 V), a kapcsoló felső állásában pedig logikai magas szintű (3,3 V) lesz.

2-17. táblázat: A DIP kapcsoló bekötése.

Kapcsoló	7	6	5	4	3	2	1	0
FPGA láb	P88	P89	P90	P91	P51	P50	P49	P46

2.3.3.2 Nyomógombok

A LOGSYS Cyclone III FPGA kártyán található 5 darab nyomógomb bekötését a 2-18. táblázat mutatja. A nyomógombok jelölése balról jobbra rendre BTN3-BTN0 és RST. Az adott FPGA lábra logikai magas szint (3,3 V) kerül a nyomógomb megnyomása esetén. Az RST gomb elsősorban az alaphelyzetbe állításra szolgál, de tetszőlegesen is felhasználható.

2-18. táblázat: A nyomógombok bekötése.

Nyomógomb	BTN3	BTN2	BTN1	BTN0	RST
FPGA láb	P30	P31	P28	P25	P24

2.3.4 Órajel források

Az FPGA a kártyán lévő 16 MHz-es oszcillátortól és a fejlesztői port CLK vonaláról kaphat órajelet. Mindkettő az FPGA egy-egy órajel bemeneti lábára (CLK) csatlakozik. Az oszcillátor 16 MHz-es órajeléből az FPGA-ban található PLL blokkok segítségével egyéb frekvenciák is előállíthatók. Az órajel források bekötését a 2-19. táblázat mutatja.

2-19. táblázat: Az órajel források bekötése.

Órajel forrás	FPGA láb
16 MHz-es oszcillátor	P23
Fejlesztői port CLK vonala	P22

2.3.5 FPGA konfigurációs módok

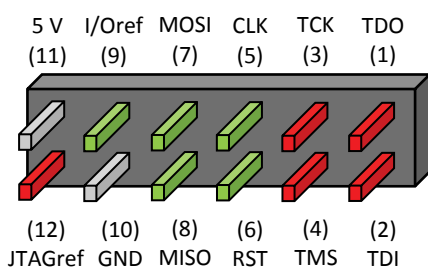
A LOGSYS Cyclone III FPGA kártya esetén kétféle konfigurációs mód lehetséges. Az FPGA felkonfigurálható a fejlesztői port JTAG interfészén keresztül, illetve az eszköz képes magát felkonfigurálni a kártyán lévő SPI buszos soros FLASH memóriából is. A konfigurációs mód egy jumperrel választható ki (lásd a 2-20. táblázatot). A JTAG interfész a kiválasztott módtól függetlenül mindig rendelkezésre áll.

2-20. táblázat: Az FPGA lehetséges konfigurációs módjai

Jumper állása	Konfigurációs mód	Leírás
	JTAG	Az FPGA-t a JTAG interfészén keresztül kell felkonfigurálni.
	SPI	Az FPGA az SPI buszos soros FLASH memóriából konfigurálja fel magát a tápfeszültség bekapcsolása vagy a PROG gomb megnyomása után.

2.3.6 LOGSYS fejlesztői port

A LOGSYS fejlesztői kábel a fejlesztői porton (2-14. ábra) keresztül illeszthető a kártyához, melynek részletesebb leírása megtalálható a 2.2.6. pontban. A csatlakozó vezérlési és soros kommunikációs vonalainak bekötését a 2-21. táblázat mutatja. A CLK és RST vonalakra 22 k Ω -os lehúzó ellenállások csatlakoznak, a MOSI vonalra pedig egy 22 k Ω -os felhúzó ellenállás csatlakozik. Így ezeken a bemeneteken akkor is valós logikai szintek jelennek meg, ha nincsenek meghajtva. A MISO kimenetre egy 22 k Ω -os felhúzó ellenállás van kötve.



2-14. ábra: A fejlesztői port tűskesorának lábkiosztása.

2-21. táblázat: A fejlesztői port bekötése.

Jel	Irány	FPGA láb
MOSI	bemenet	P111
MISO	kimenet	P110
CLK	bemenet	P22
RST	bemenet	P112

2.3.7 Tápellátás

Az FPGA kártya 5 V-os tápfeszültséget igényel. A tápellátás alapvetően a fejlesztői kábelről történik, de lehetőség van egyéb külső 5 V-os egyenfeszültség forrás csatlakoztatására is. Az 5 V DC tápcsatlakozó védett a fordított polaritású bekötés ellen.

Az FPGA a működéséhez 3,3 V-os (I/O vonalak), 2,5 V-os (JTAG, PLL) és 1,2 V-os (belső mag, PLL) tápfeszültséget igényel. Ezeket a feszültségeket az 5 V-os tápfeszültségből külön tápegységek állítják elő. A kártyán található összes periféria és az I/O vonalak a JTAG interfész kivételével 3,3 V-ról működnek, a fejlesztői kábel 3,3 V-os I/O referenciafeszültséget (I/Oref) kap a fejlesztői porton keresztül. A JTAG interfész 2,5 V-os feszültségről működik, a fejlesztői kábel 2,5 V-os JTAG referenciafeszültséget (JTAGref) kap a fejlesztői porton keresztül.

2.3.8 Bővítőcsatlakozók

A LOGSYS Cyclone III FPGA kártyához a kiegészítő modulok illesztését két 16 pólusú csatlakozó teszi lehetővé. Mindkét csatlakozó lábkiosztása azonos, ez a kártya szerinti nézetből a 2-15. ábrán látható. A csatlakozókra ki van vezetve mind a 3,3 V-os, mind pedig az 5 V-os tápfeszültség is, azonban az adatvonalak 3,3 V-ról működnek és nem 5 V toleránsak. A 13 adatvonal mindegyike kétirányú. A bővítőcsatlakozók bekötését a 2-22. táblázat és a 2-23. táblázat mutatja.

(15)	(13)	(11)	(9)	(7)	(5)	(3)	(1)
I/O	I/O	I/O	I/O	I/O	I/O	+3,3V	GND
(16)	(14)	(12)	(10)	(8)	(6)	(4)	(2)
I/O	I/O	I/O	I/O	I/O	I/O	I/O	+5V

2-15. ábra: A bővítőcsatlakozók lábkiosztása.

2-22. táblázat: Az A bővítőcsatlakozó bekötése.

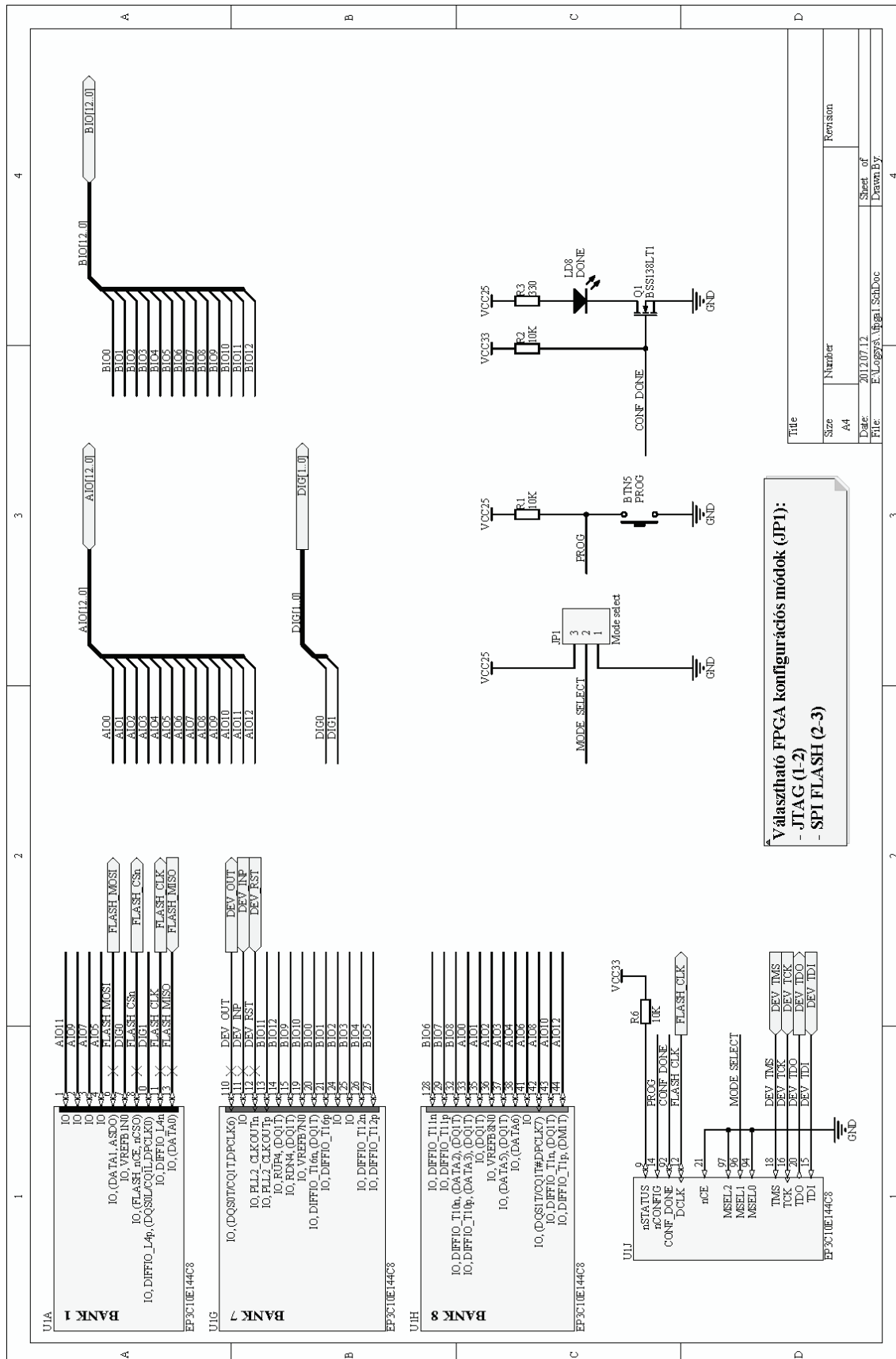
A bővítőcsatlakozó		
Csatlakozó láb	Irány	FPGA láb
4	kétirányú	P133
5	kétirányú	P135
6	kétirányú	P136
7	kétirányú	P137
8	kétirányú	P138
9	kétirányú	P4
10	kétirányú	P141
11	kétirányú	P3
12	kétirányú	P142
13	kétirányú	P2
14	kétirányú	P143
15	kétirányú	P1
16	kétirányú	P144

2-23. táblázat: A B bővítőcsatlakozó bekötése.

B bővítőcsatlakozó		
Csatlakozó láb	lány	FPGA láb
4	kétirányú	P120
5	kétirányú	P121
6	kétirányú	P124
7	kétirányú	P125
8	kétirányú	P126
9	kétirányú	P127
10	kétirányú	P128
11	kétirányú	P129
12	kétirányú	P132
13	kétirányú	P115
14	kétirányú	P119
15	kétirányú	P113
16	kétirányú	P114

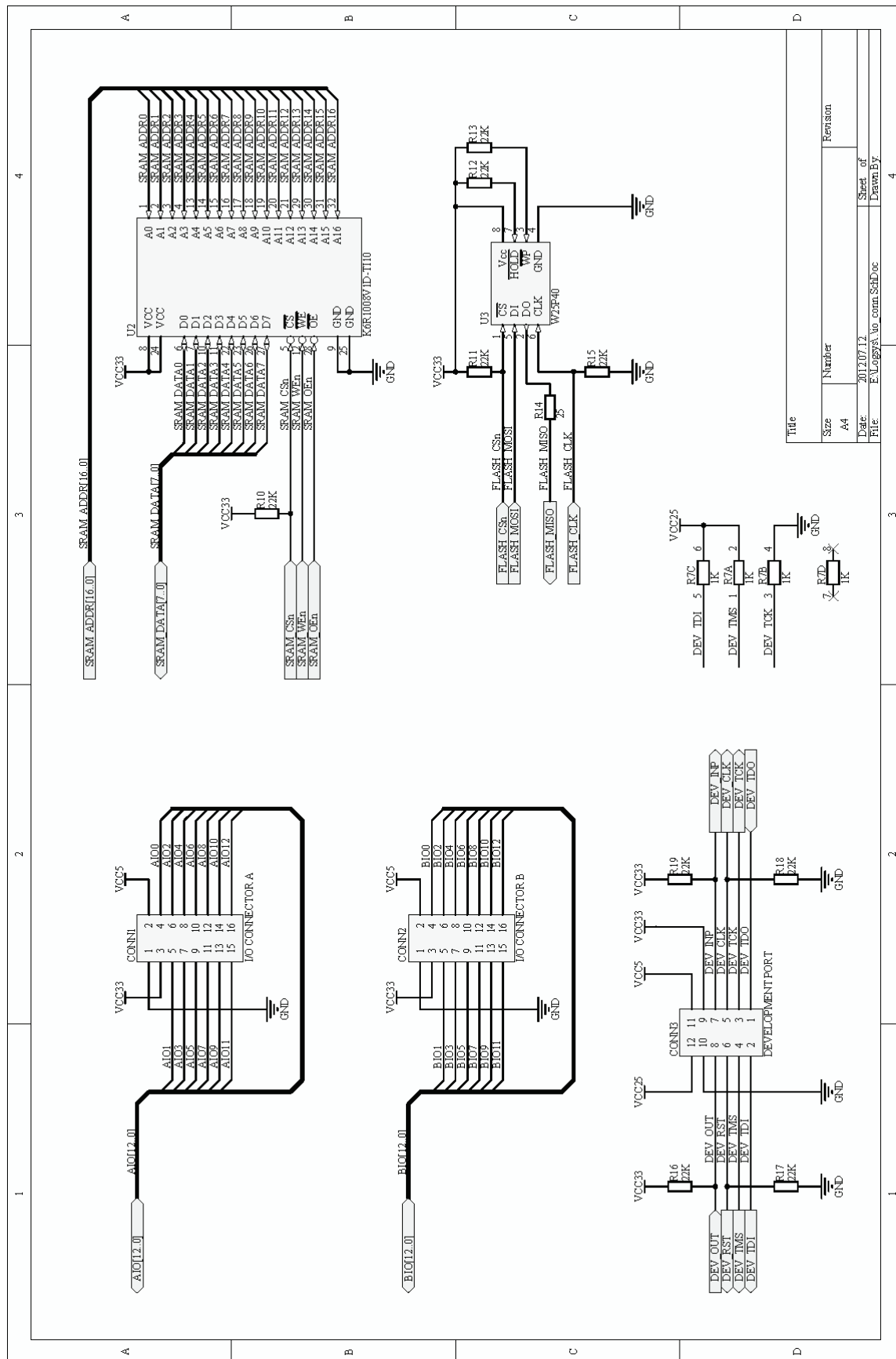
2.3.9 Kapcsolási rajz

2.3.9.1 FPGA

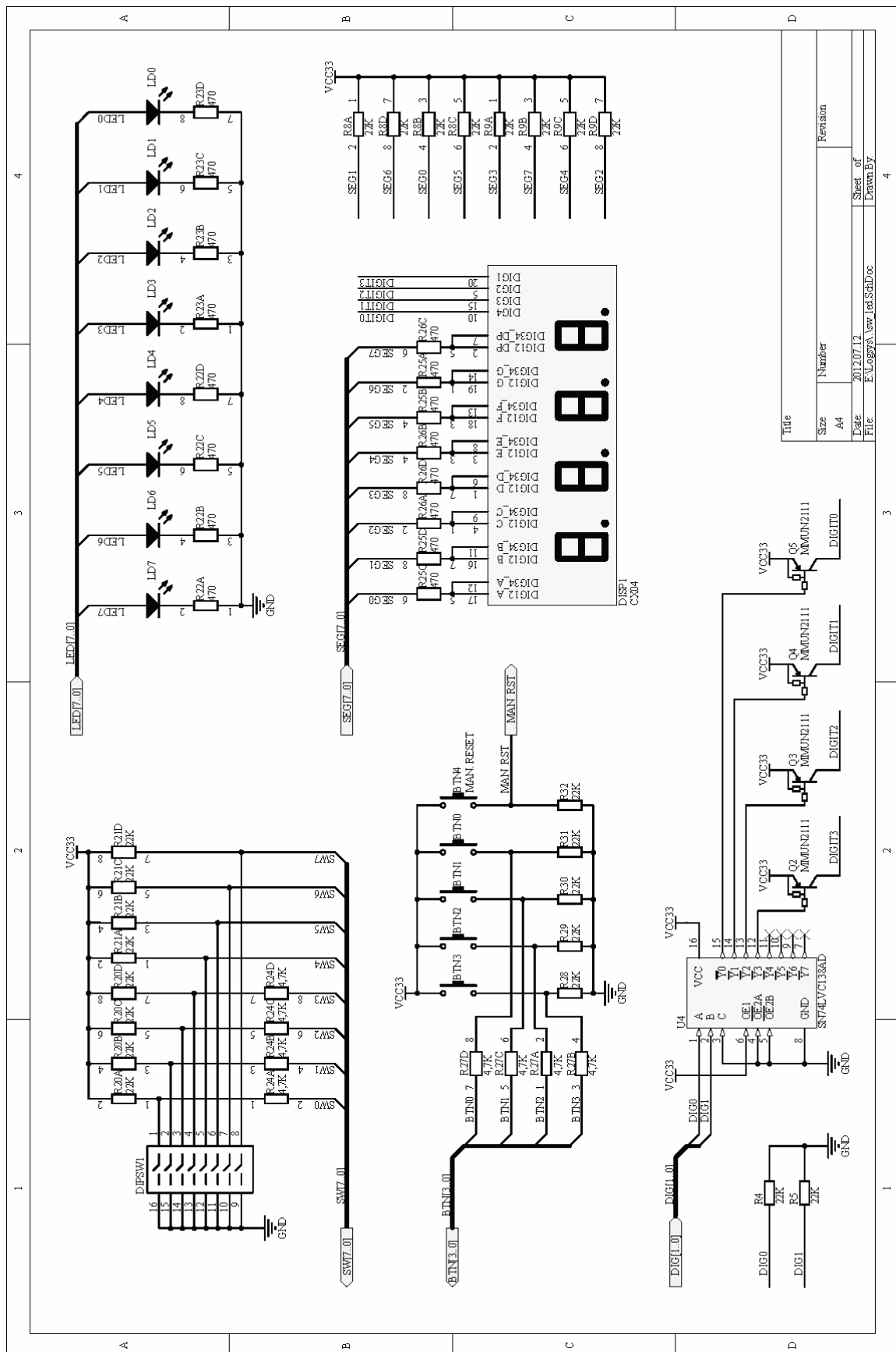




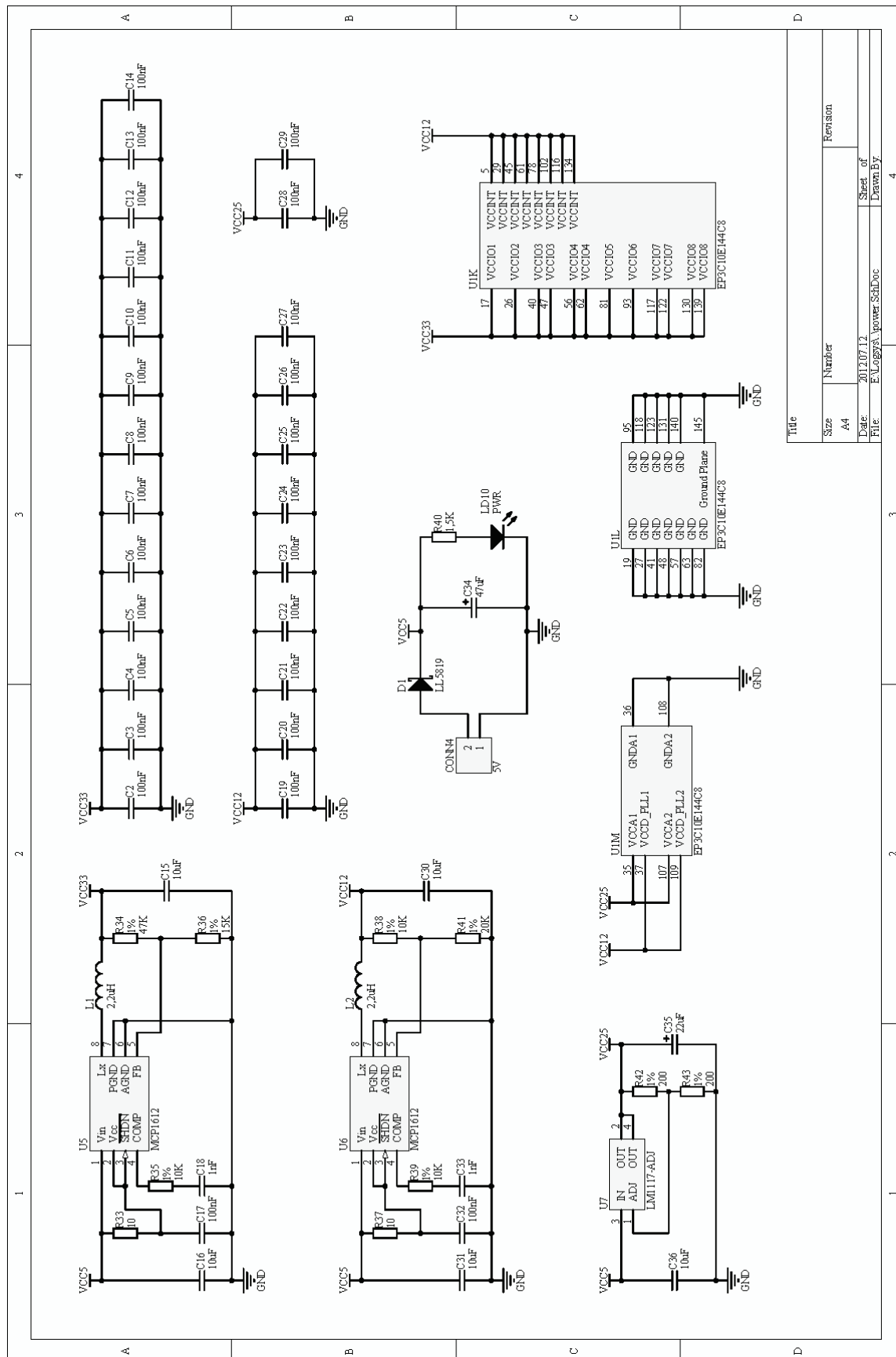
2.3.9.2 Csatlakozók, memóriák



2.3.9.3 LED-ek, kijelzők, DIP kapcsoló, nyomógombok



2.3.9.4 Tápegység

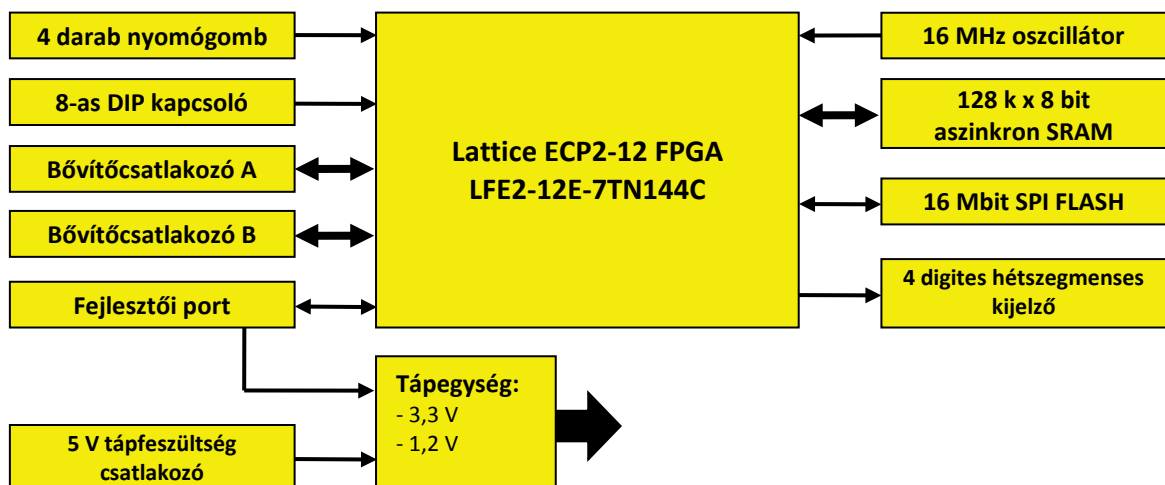


2.4 LOGSYS ECP2 FPGA kártya

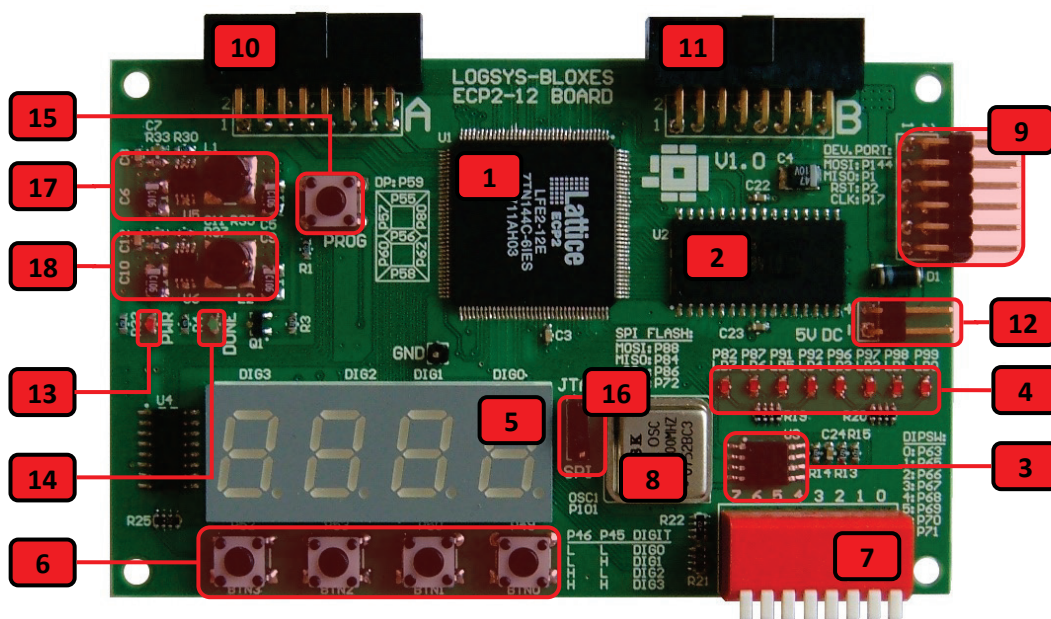
A LOGSYS ECP2 FPGA kártya a többi LOGSYS FPGA alaplártyához hasonlóan egy egyszerű felépítésű, elsősorban kezdő felhasználók számára készült FPGA kártya, amely alkalmas összetettebb tervek megvalósítására is. A kártya blokkvázlata a 2-16. ábrán látható, a kártya felépítését a 2-17. ábra szemlélteti.

A kártyán az alábbi komponensek találhatók:

- Lattice LFE2-12E-7TN144C típusú FPGA, amely lehetővé teszi összetettebb logikák és kisebb mikroproceszoros rendszerek megvalósítását. Az eszköz főbb jellemzői:
 - 12000 darab 4 bemenetű LUT és flip-flop
 - 12 darab 18 kbytes blokk-RAM
 - 6 darab sysDSP blokk (egy 36 x 36 bites, négy db 18 x 18 bites vagy nyolc 9 x 9 bites MAC egység sysDSP blokkonként)
 - 2 darab PLL és 2 darab DLL (Delay Locked Loop)
- Memóriák a program és az adatok tárolására:
 - Egy 128 k x 8 bites, 10 ns-os aszinkron SRAM (Samsung K6R1008V1D-TI10)
 - Egy 16 Mbytes SPI buszos soros FLASH memória (Winbond W25P16 vagy W25X16)
 - A soros FLASH memória konfigurációs memóriaként is szolgál az FPGA számára
- Megjelenítő eszközök:
 - 8 darab LED
 - 4 digitális hétszempennyes kijelző
- Beviteli eszközök:
 - 4 darab nyomógomb
 - 8-as DIP kapcsoló
- Egy 16 MHz-es oszcillátor
- Csatlakozó a LOGSYS fejlesztői kábel számára
- 2 darab csatlakozó a kiegészítő modulok számára:
 - 13 FPGA I/O láb (11 kétirányú, 2 csak bemenet)
 - 5 V és 3,3 V tápfeszültség kimenet



2-16. ábra: A LOGSYS ECP2 FPGA kártya blokkvázlata.



2-17. ábra: A LOGSYS ECP2 FPGA kártya.

A LOGSYS ECP2 FPGA kártya felépítése:

1. Lattice LFE2-12E-7TN144C típusú FPGA
2. 128 k x 8 bites, 10 ns hozzáférési idejű aszinkron SRAM
3. Winbond W25P16 vagy W25X16 típusú 16 Mbytes SPI buszos soros FLASH
4. 8 darab LED
5. 4 digités hétszegmenses kijelző
6. 4 darab nyomógomb
7. 8-as DIP csatlakozó
8. 16 MHz-es oszcillátor
9. Csatlakozó a LOGSYS fejlesztői kábel számára (fejlesztői port)
10. Csatlakozó a kiegészítő modulok számára (A)
11. Csatlakozó a kiegészítő modulok számára (B)
12. 5 V tápfeszültség csatlakozó
13. A bekapcsolt tápfeszültséget jelző piros LED (PWR)
14. Az FPGA sikeres felkonfigurálását jelző zöld LED (DONE)
15. Az FPGA újrakonfigurálását elindító nyomógomb (PROG)
16. Az FPGA konfigurációs módját kiválasztó jumper
17. 3,3 V feszültséget előállító tápegység
18. 1,2 V feszültséget előállító tápegység

2.4.1 Memóriák

A LOGSYS Spartan-3E FPGA kártyához hasonlóan a LOGSYS ECP2 FPGA kártyán is a Samsung K6R1008V1D-TI10 típusú 128 k x 8 bites, 10 ns elérési idejű aszinkron SRAM és a Winbond W25P16 (vagy W25X16) típusú 16 Mbites SPI buszos soros FLASH memória található meg. A memóriák bekötését a 2-24. táblázat és a 2-25. táblázat mutatja. A soros FLASH memória konfigurációs memóriaként is szolgál az FPGA számára.

2-24. táblázat: Az SRAM bekötése.

Adatbusz		Címbusz	
Jel	FPGA láb	Jel	FPGA láb
D0	P100	A0	P109
D1	P110	A1	P108
D2	P111	A2	P107
D3	P112	A3	P104
D4	P125	A4	P114
D5	P126	A5	P115
D6	P140	A6	P116
D7	P137	A7	P118
		A8	P119
		A9	P121
		A10	P122
		A11	P123
		A12	P124
		A13	P134
		A14	P129
		A15	P130
		A16	P131

Vezérlő jelek	
Jel	FPGA láb
CSn	P103
WEn	P113
OEn	P136

2-25. táblázat: A FLASH memória bekötése.

FLASH	FPGA láb
DI (MOSI)	P88
DO (MISO)	P84
CLK	P72
CS	P86

2.4.2 Megjelenítő eszközök

2.4.2.1 LED-ek

A LOGSYS ECP2 FPGA kártyán található 8 darab LED bekötését a 2-26. táblázat mutatja. A LED-ek LD0-tól LD7-ig vannak számozva, a bal szélső LED az LD7, a jobb szélső LED az LD0. A LED-ek vezérlő jelei aktív magas szintűek.

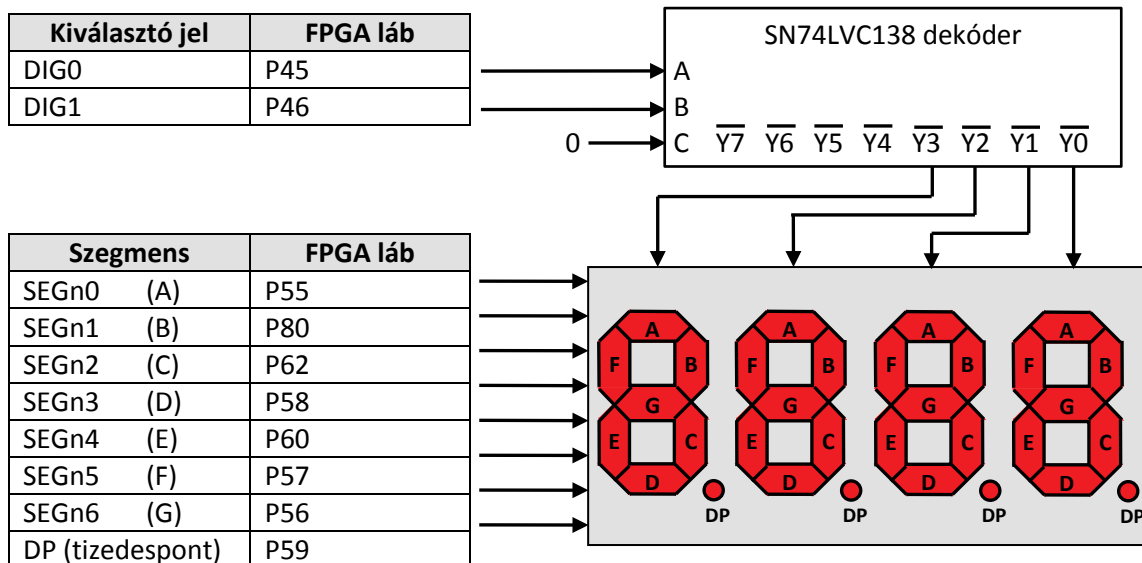
2-26. táblázat: A LED-ek bekötése.

LED	LD7	LD6	LD5	LD4	LD3	LD2	LD1	LD0
FPGA láb	P87	P87	P91	P92	P96	P97	P98	P99

2.4.2.2 Hétszegmenses kijelző

A LOGSYS ECP2 FPGA kártyán található 4 digitális hétszegmenses kijelző bekötését a 2-27. táblázat és a 2-18. ábra mutatja. A karakterek DIG0-tól DIG3-ig vannak számozva, a bal szélső karakter a DIG3, a jobb szélső karakter a DIG0. A szegmensvezérlő jelek aktív alacsony szintűek. A kijelző vezérlésének megvalósítása ugyanolyan, mint a LOGSYS Cyclone III FPGA kártya esetén, erről bővebb leírás a 2.3.2.2. pontban olvasható.

2-27. táblázat: A hétszegmenses kijelző bekötése.



2-18. ábra: A hétszegmenses kijelző bekötése.

2.4.3 Beviteli eszközök

2.4.3.1 DIP kapcsoló

A LOGSYS ECP2 FPGA kártyán található 8-as DIP kapcsoló bekötését a 2-28. táblázat mutatja. A kapcsolók 0-tól 7-ig vannak számozva, a bal szélső kapcsoló sorszáma a 7, a jobb szélső kapcsoló sorszáma a 0. Az adott FPGA láb a kapcsoló alsó állásában logikai alacsony szintű (0 V), a kapcsoló felső állásában pedig logikai magas szintű (3,3 V) lesz.

2-28. táblázat: A DIP kapcsoló bekötése.

Kapcsoló	7	6	5	4	3	2	1	0
FPGA láb	P71	P70	P69	P68	P67	P66	P65	P63

2.4.3.2 Nyomógombok

A LOGSYS ECP2 FPGA kártyán található 4 darab nyomógomb bekötését a 2-29. táblázat mutatja. A nyomógombok jelölése balról jobbra rendre BTN3-BTN0. Az adott FPGA láb a gomb megnyomása esetén logikai magas szint (3,3 V) kerül.

2-29. táblázat: A nyomógombok bekötése.

Nyomógomb	BTN3	BTN2	BTN1	BTN0
FPGA láb	P52	P53	P50	P49

2.4.4 Órajel források

Az FPGA a kártyán lévő 16 MHz-es oszcillátortól és a fejlesztői port CLK vonaláról kaphat órajelet. Mindkettő az FPGA egy-egy órajel bemeneti lábára csatlakozik. Az oszcillátor 16 MHz-es órajeléből az FPGA-ban található PLL és DLL blokkok segítségével egyéb frekvenciák is előállíthatók. Az órajel források bekötését a 2-30. táblázat mutatja.

2-30. táblázat: Az órajel források bekötése.

Órajel forrás	FPGA láb
16 MHz-es oszcillátor	P101
Fejlesztői port CLK vonala	P17

2.4.5 FPGA konfigurációs módok

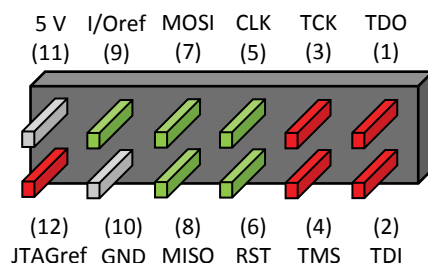
A LOGSYS ECP2 FPGA kártya esetén kétféle konfigurációs mód lehetséges. Az FPGA felkonfigurálható a fejlesztői port JTAG interfészén keresztül, illetve az eszköz képes magát felkonfigurálni a kártyán lévő SPI buszos soros FLASH memóriából is. A konfigurációs mód egy jumperrel választható ki (lásd a 2-31. táblázatot). A JTAG interfész a kiválasztott módtól függetlenül mindig rendelkezésre áll.

2-31. táblázat: Az FPGA lehetséges konfigurációs módjai

Jumper állása	Konfigurációs mód	Leírás
	JTAG	Az FPGA-t a JTAG interfészen keresztül kell felkonfigurálni.
	SPI	Az FPGA az SPI buszos soros FLASH memóriából konfigurálja fel magát a tápfeszültség bekapcsolása vagy a PROG gomb megnyomása után.

2.4.6 LOGSYS fejlesztői port

A LOGSYS fejlesztői kábel a fejlesztői porton (2-19. ábra) keresztül illeszthető a kártyához, melynek részletesebb leírása megtalálható a 2.2.6. pontban. A csatlakozó vezérlési és soros kommunikációs vonalainak bekötését a 2-32. táblázat mutatja. Mivel a MOSI, a CLK és az RST bemeneteken nincs külső lehúzó/felhúzó ellenállás, ezért használatuk esetén a hozzájuk tartozó FPGA I/O lábakon engedélyezni kell a belső lehúzó/felhúzó ellenállást. Ellenkező esetben ezek a bemenetek lebegni fognak és a letöltött terv nem fog megfelelően működni. A MISO kimenetre egy külső 22 k Ω -os felhúzó ellenállás van kötve.



2-19. ábra: A fejlesztői port tűsorsorának lábkiosztása.

2-32. táblázat: A fejlesztői port bekötése.

Jel	Írány	FPGA láb
MOSI	bemenet	P144
MISO	kimenet	P1
CLK	bemenet	P17
RST	bemenet	P2

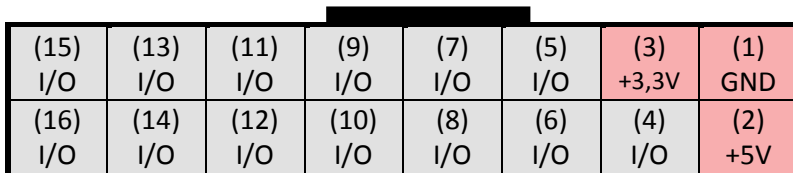
2.4.7 Tápellátás

Az FPGA kártya 5 V-os tápfeszültséget igényel. A tápellátás alapvetően a fejlesztői kábelről történik, de lehetőség van egyéb külső 5 V-os egyenfeszültség forrás csatlakoztatására is. Az 5 V DC tápcsatlakozó védett a fordított polaritású bekötés ellen.

Az FPGA a működéséhez 3,3 V-os (I/O vonalak, konfigurációs interfész) és 1,2 V-os (belső mag, PLL, DLL) tápfeszültséget igényel. Ezeket a feszültségeket az 5 V-os tápfeszültségből külön tápegységek állítják elő. A kártyán található összes periféria, valamint az I/O és a konfigurációs vonalak 3,3 V-ról működnek. A fejlesztői kábel 3,3 V-os I/O referenciafeszültséget (I/Oref) és 3,3 V-os JTAG referenciafeszültséget (JTAGref) kap a fejlesztői porton keresztül.

2.4.8 Bővítőcsatlakozók

A LOGSYS ECP2 FPGA kártyához a kiegészítő modulok illesztését két 16 pólusú csatlakozó teszi lehetővé. Mindkét csatlakozó lábkiosztása azonos, ez a kártya szerinti nézetből a 2-20. ábrán látható. A csatlakozókra ki van vezetve mind a 3,3 V-os, mind pedig az 5 V-os tápfeszültség is, azonban az adatvonalak 3,3 V-ról működnek és nem 5 V toleránsak. A 13 adatvonal mindegyike kétirányú. A bővítőcsatlakozók bekötését a 2-33. táblázat és a 2-34. táblázat mutatja.



(15) I/O	(13) I/O	(11) I/O	(9) I/O	(7) I/O	(5) I/O	(3) +3,3V	(1) GND
(16) I/O	(14) I/O	(12) I/O	(10) I/O	(8) I/O	(6) I/O	(4) I/O	(2) +5V

2-20. ábra: A bővítőcsatlakozók lábkiosztása.

2-33. táblázat: Az A bővítőcsatlakozó bekötése.

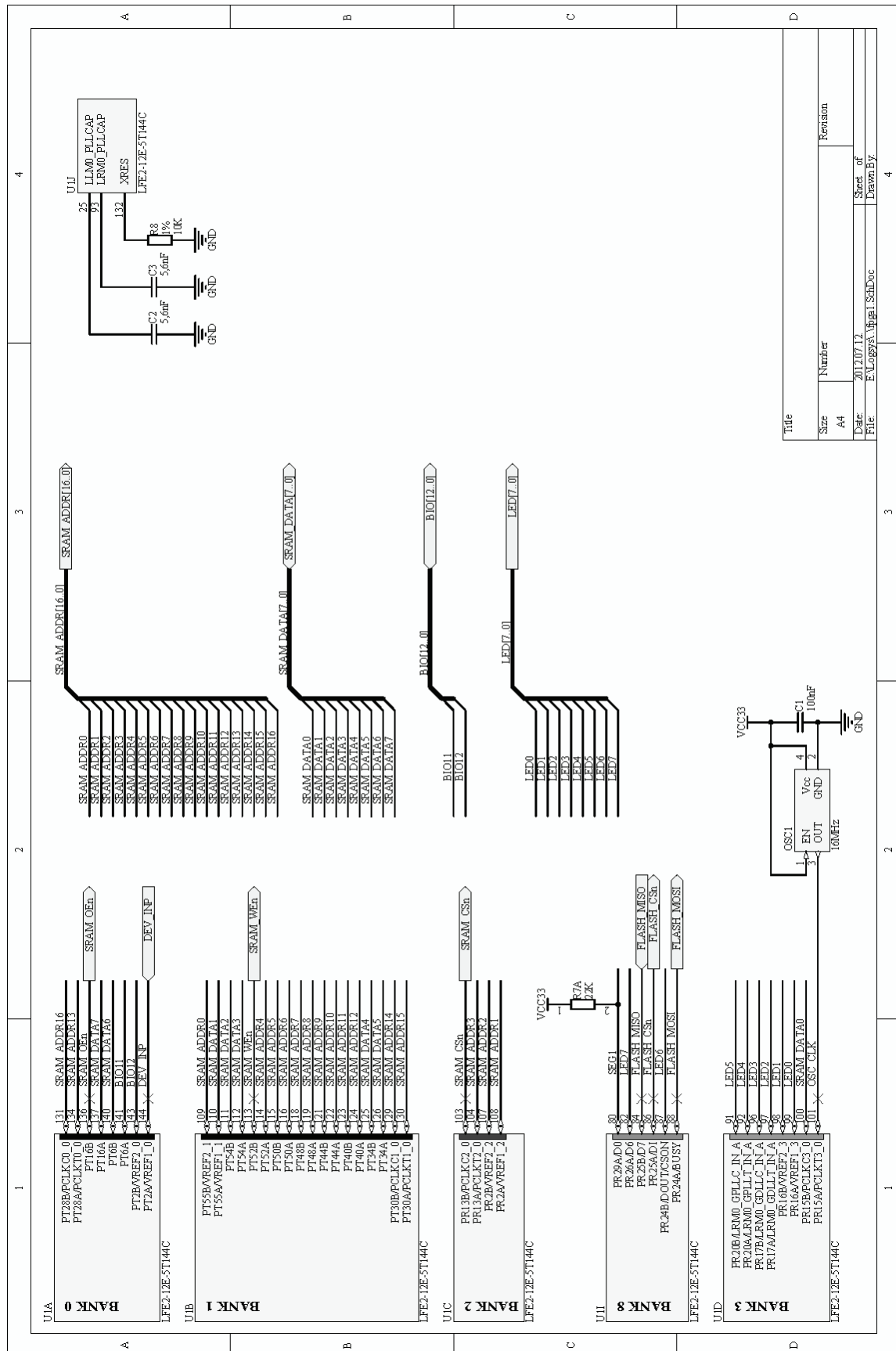
A bővítőcsatlakozó		
Csatlakozó láb	Írány	FPGA láb
4	kétirányú	P19
5	kétirányú	P44
6	kétirányú	P20
7	kétirányú	P43
8	kétirányú	P23
9	kétirányú	P41
10	kétirányú	P24
11	kétirányú	P40
12	kétirányú	P26
13	kétirányú	P38
14	kétirányú	P27
15	kétirányú	P37
16	kétirányú	P28

2-34. táblázat: A B bővítőcsatlakozó bekötése.

B bővítőcsatlakozó		
Csatlakozó láb	Írány	FPGA láb
4	kétirányú	P5
5	kétirányú	P7
6	kétirányú	P8
7	kétirányú	P10
8	kétirányú	P12
9	kétirányú	P13
10	kétirányú	P14
11	kétirányú	P15
12	kétirányú	P18
13	kétirányú	P3
14	kétirányú	P4
15	kétirányú	P141
16	kétirányú	P143

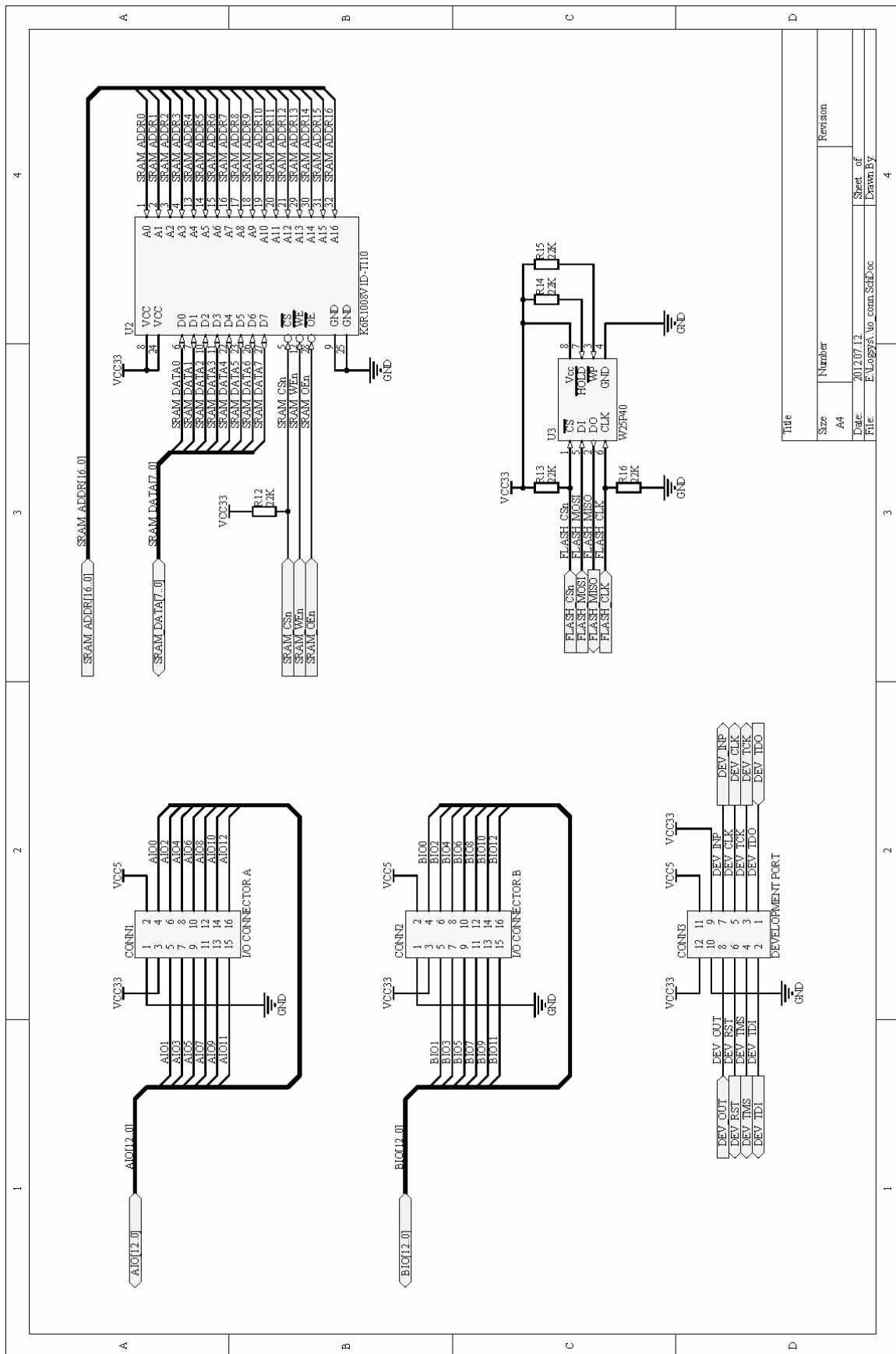
2.4.9 Kapcsolási rajz

2.4.9.1 FPGA

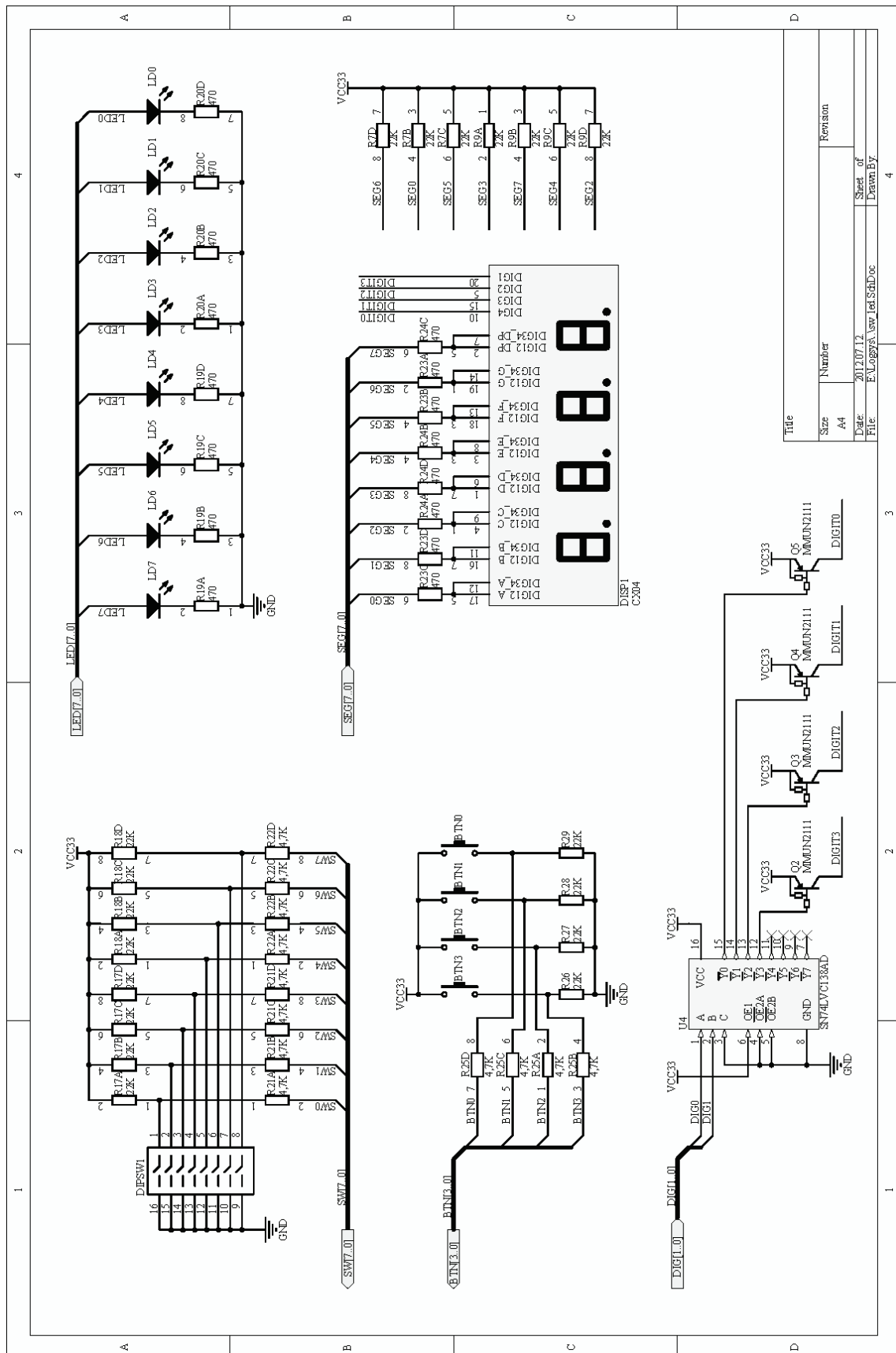




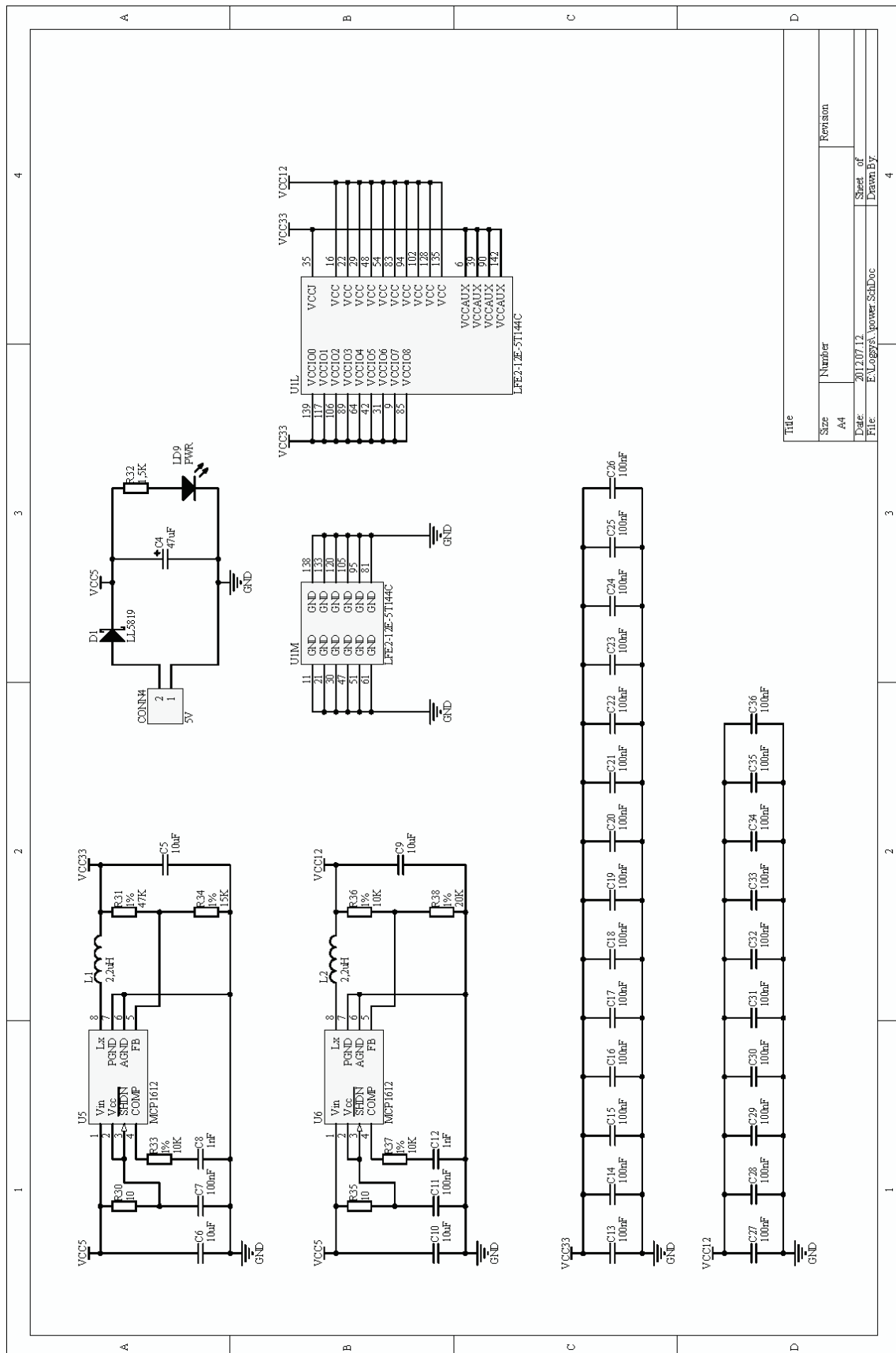
2.4.9.2 Csatlakozók, memóriák



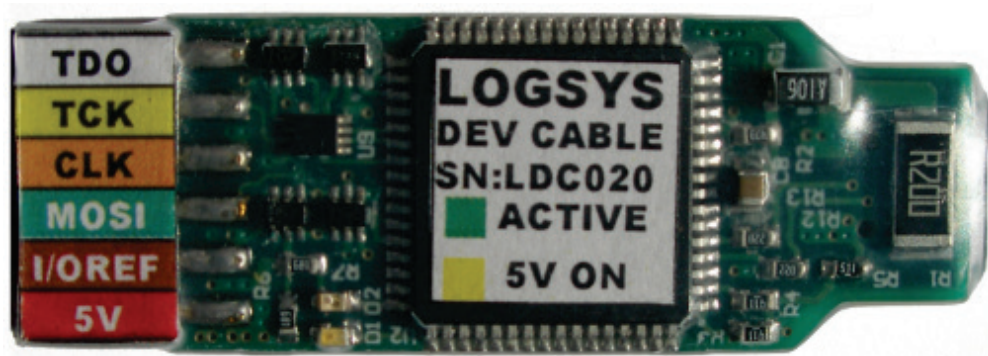
2.4.9.3 LED-ek, kijelzők, DIP kapcsoló, nyomógombok



2.4.9.4 Tápegység



3 LOGSYS fejlesztő kábel



3.1 Bevezetés

A LOGSYS fejlesztő kábel a programozható eszközöket tartalmazó beágyazott rendszerek általános célú, flexibilis fejlesztőeszköze. Használatával mind a fejlesztés korai szakaszában, mind a későbbi alkalmazásfejlesztés során egy hatékony, könnyen kezelhető eszköz áll a fejlesztő rendelkezésére. Az eszköz elsősorban oktatási célokra készült, de flexibilitása, széleskörű szolgáltatásai alkalmassá teszik egyéb környezetben is a programozási, hibakeresési feladatok hatékony végrehajtására. Szolgáltatásai, könnyű kezelhetősége, a különböző gyártók eszközeinek támogatási lehetősége a legtöbb hasonló célú eszközzel összehasonlítva is jelentős előnyöket mutat.

3.1.1 A LOGSYS fejlesztő kábel fontosabb jellemzői

A LOGSYS fejlesztő kábel tervezésekor a lehető legszélesebb körű szolgáltatások beépítése volt a cél, a felhasználó számára egyszerűen, könnyen kezelhető módon. A LOGSYS fejlesztő kábel szolgáltatásai elsősorban a 4. fejezetben ismertetett LOGSYS grafikus felhasználói interfészen keresztül használhatók ki, de egyedi alkalmazásokhoz is testre szabhatók. A legfontosabb jellemzők a következők:

- Teljes integráltság a felhasználói környezethez, minden üzemi paraméter az USB interfészen keresztül konfigurálható a PC alkalmazásból.
- Energiaellátás az USB kimenetről, a működési állapottól függő teljesítmény felvétel folyamatosan mérhető.
- Konfiguráció és alkalmazás letöltés a JTAG interfészen keresztül az ipari szabvány SVF fájlformátum használatával.
- Az alkatrész adatbázis rugalmas bővíthetősége a gyártók által megadott BSDL adatok alapján.
- UART kommunikációs terminál, szokásos beállítási lehetőségekkel.
- Kiegészítő soros kommunikációs módok különböző elektronikai alkatrészek felprogramozásához, konfigurációjához (SPI, I²C, TWI, USART).
- Speciális, nem szabványos konfigurációs üzemmódok kialakíthatósága (pl. PIC mikrovezérlők programozásához).
- Demonstrációs célú szoftveres időzítésű vezérlőjelek egyszerű tesztelésekhez.

A LOGSYS fejlesztő kábel használata megengedi a gyártófüggetlen használatot, de természetesen lehetővé teszi a különböző programozható eszközök gyártói által biztosított fejlesztői környezetekhez történő optimális illeszkedést is. Ennek megfelelően a LOGSYS fejlesztő kábel a LOGSYS GUI-n keresztül biztosítja a Xilinx gyártmányú CPLD-k és FPGA-k programozása során a gyártóspecifikus programozási formátumok (pl. BIT fájl, JED fájl) felismerését és használatát. Mivel a rendszer natív formátuma az SVF szabvány, a letöltő tetszőleges gyártó termékéhez használható, ha a gyártó támogatja az SVF fájl előállítását. Ez általában teljesül, a gyártóspecifikus fejlesztőkörnyezetbe épített valamilyen fájlkonverter, SVF generátor segítségével. A standard JTAG interfészen programozható mikrovezérlőkhöz is gyakran rendelkezésre áll az SVF kimenetet generáló funkció.

3.2 A LOGSYS fejlesztő kábel részletes ismertetése

A LOGSYS fejlesztő kábel egy általános célú programozó és kommunikációs interfész, amely a PC USB portján keresztül megvalósítja mindazt a kapcsolatot a fejlesztett beágyazott rendszer és a PC-s alkalmazás között, amely szükséges a teljes fejlesztési folyamat során a beágyazott rendszer működtetéséhez, konfigurálásához, vezérléséhez, és megfigyeléséhez. A LOGSYS fejlesztő kábel szolgáltatásai biztosítják a fejlesztett rendszer energiaellátását, a konfiguráció letöltését, a rendszer működtetését az elsődleges vezérlőjelek kiadásával, illetve a kommunikáció lehetőségét az elterjedt soros kommunikációs üzemmódok valamelyikének használatával.

3.2.1 A LOGSYS fejlesztő kábel paraméterei

Általános tulajdonságok:

- USB 2.0 kompatibilis, 12 Mb/s (full-speed) interfész
- Támogatott operációs rendszerek:
 - 32 bites és 64 bites Windows XP
 - 32 bites és 64 bites Windows Vista
 - 32 bites és 64 bites Windows 7
- A grafikus felhasználói felület a .NET 2.0 (vagy újabb) keretrendszert igényli
- Speciális szoftveres meghajtók alkalmazás specifikus használathoz

Energiaellátás:

- Kapcsolható 5 V/450 mA energia ellátás az USB portról
- Beállítható 700 mA vagy 950 mA áramlimit Y-kábel használata esetén
- Rövidzárvédett tápellátás, programozható áram méréshatár
- Üzemi feszültségek monitorozása (5Vout, JTAGref, I/Oref)
- Tetszőleges JTAG és I/O referenciafeszültségek szimultán használata (1,65 V – 5,5 V)
- LED-es visszajelzés a feszültség vonalak aktív állapotáról

Konfiguráció:

- Szabványos IEE1149.1 JTAG interfész SVF fájl letöltéssel
- A jelszinteket a vizsgált rendszer által alkalmazott JTAGref feszültség értéke határozza meg, értékük a szokásos tartományban tetszőleges lehet (1,65 V – 5,5 V)
- Speciális letöltési módok egyedi protokoll alapján (2/3/4 vezetékes interfészek)
- LED-es visszajelzés az aktív kommunikációról

Kommunikáció:

- Soros kommunikációs üzemmódok az USB interfészen keresztül
- Hagyományos UART kommunikáció RX/TX jelvezetékekkel
 - PC soros port (virtuális COM port) típusú eszközként is használható
- Egyszerű USRT szinkron kommunikáció, CLK/RX/TX jelvezetékekkel
- SPI kommunikációs interfész SCK/CSn/MOSI/MISO jelvezetékekkel
- I²C kommunikációs interfész SCL/SDA jelvezetékekkel
- Szoftveres adatinterfész (BitBang) interfész CLK/RST/MOSI/MISO jelvezetékekkel
- A jelszinteket a vizsgált rendszer által alkalmazott I/Oref feszültség értéke határozza meg, értékük a szokásos tartományban tetszőleges lehet (1,65 V – 5,5 V)

3.2.2 A LOGSYS fejlesztő kábel felépítése

A LOGSYS fejlesztő kábel kialakítása a használati módok figyelembevételével történt. A kisméretű elektronika a tervezett funkciók realizálásához szükséges komponenseket tartalmazza, kétoldalas felületszerelt négyrétegű hordozón megvalósítva. A LOGSYS fejlesztő kábel tipikus használati módja a 3-1. ábrán látható.



3-1. ábra: A LOGSYS fejlesztő kábel használata.

A LOGSYS fejlesztő kábel két fizikai interfésszel rendelkezik, melyekkel a fejlesztett rendszerhez és a PC-hez csatlakozik. Az egyik interfész egy szabványos USB mini-B anya csatlakozó, amelyhez egy szabványos USB 4 pólusú normál A csatlakozóval és 5 pólusú USB mini-B csatlakozóval szerelt kábel csatlakoztatható (3-2. ábra). Ezen keresztül lehet hozzákapcsolni a PC-n, laptopon vagy USB HUB-on található normál USB interfészhez. Az eszköz elektromosan és protokoll szinten kompatibilis a szabványos USB 2.0 interfésszel, annak 12 Mb/s sebességű (full-speed) üzemmódját használja. A LOGSYS fejlesztő kábel nagyteljesítményű (500 mA) kompozit USB eszközként regisztrálódik a PC operációs rendszerében. Az eszközkészítés részleteit a meghajtóprogram és a felhasználói szoftver leírása tartalmazza.



3-2. ábra: A LOGSYS fejlesztő kábel használatához szükséges USB kábel és a két végén található apa csatlakozók nézete (szabványos USB 4 pólusú A és 5 pólusú mini-B csatlakozók)

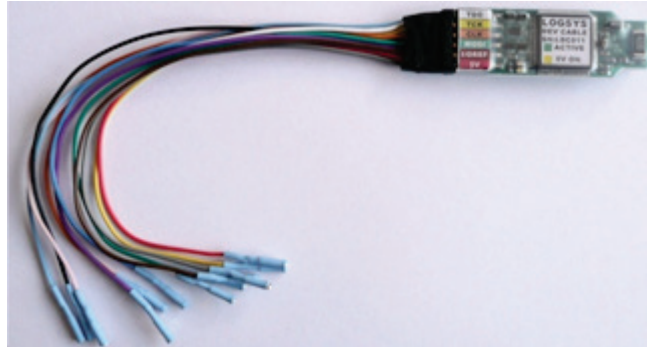
Az eszköz másik fizikai csatlakozója az egyedi kialakítású LOGSYS fejlesztői interfész (3-3. ábra). Ez az interfész egy 12 pólusú 2,54 mm-es (0,1") dupla soros anya csatlakozó. A csatlakozó pontok a kábel funkcióinak megfelelően 3 jól elkülöníthető jelcsoportot tartalmaznak: konfiguráció, kommunikáció és tápellátás. Az egyes jeleket a fejlesztői csatlakozón a megfelelő színek és a jelnevek rövidítése azonosítja. A jelnevek a jelek elsődleges funkcióját jelölik, a leggyakrabban használt üzemmódokra utalva. Ennek megfelelően a szokásos soros adatkommunikációs (adat ki/adat be) egységesen MOSI ill. MISO jelöléssel jelöltük, egyértelműen utalva a jelek funkciójára (Master Output, Slave Input, ill. Master Input, Slave Output). Így az UART-ok szokásos RX/TX jelvezetékeinek párosítása a tesztelt eszköznél a következő: FPGA RX = PC MOSI, FPGA TX = PC MISO. Bizonyos jelek a használat során speciális funkcióval rendelkeznek, ami egyedi jelneveket igényelne (pl. I²C interfész SCL és SDA jelnevek), ezeket a kábel jelölés természetesen nem tartalmazza.



JTAG TDO	JTAG TCK	CLK	MOSI	I/O ref.	5V
JTAG TDI	JTAG TMS	RESET	MISO	GND	JTAG ref.

3-3. ábra: A LOGSYS fejlesztői interfész.

A LOGSYS fejlesztő kábel interfészének használata megkönnyíti a kábel csatlakoztatását a rendszer FPGA kártyáihoz. Gyári készítésű fejlesztőkártyák (FPGA vagy μ C) esetén a közvetlen csatlakoztatás rendszerint nem lehetséges, hiszen szinte minden gyártó egyedi, az általa preferált méretű és kiosztású csatlakozót kínál a fejlesztéshez. Ebben az esetben a LOGSYS fejlesztő kábel és használt eszköz között a kapcsolat a 3-4. ábrán bemutatott lengő csatlakozós kiegészítővel is megoldható. A kiegészítő lengő csatlakozó színekódolása követi a LOGSYS fejlesztő kábel alkalmazott színekódokat, így a jelek csatlakoztatása egyértelmű a célrendszer megfelelő pontjaihoz. A lengő csatlakozós kiegészítő biztosítja a csatlakoztathatóságot a szokásos 2,54 mm (0,1") távolságú 0,64 mm átmérőjű tűkesorokhoz és használható a 2 mm távolságú csatlakozókon is. Más típusú csatlakozókhoz, finomabb kiosztású tűkesorokhoz egyedi megoldások készíthetők. Fontos megjegyezni, hogy a gyártók által készített, sokszor nagyteljesítményű alkatrészeket tartalmazó fejlesztőkártyák tápellátása már ritkán biztosítható az USB kimenetről. Ebben az esetben kártyához tartozó eredeti külső tápegységet kell használni, és a LOGSYS kábel +5V kimenetét nem kell csatlakoztatni, vagy bekapcsolni. Természetesen, ha szükséges, az inaktív +5V-os kimenettel monitorozás céljából bármilyen 0 V – +5 V feszültséget mérhetünk a használat közben.

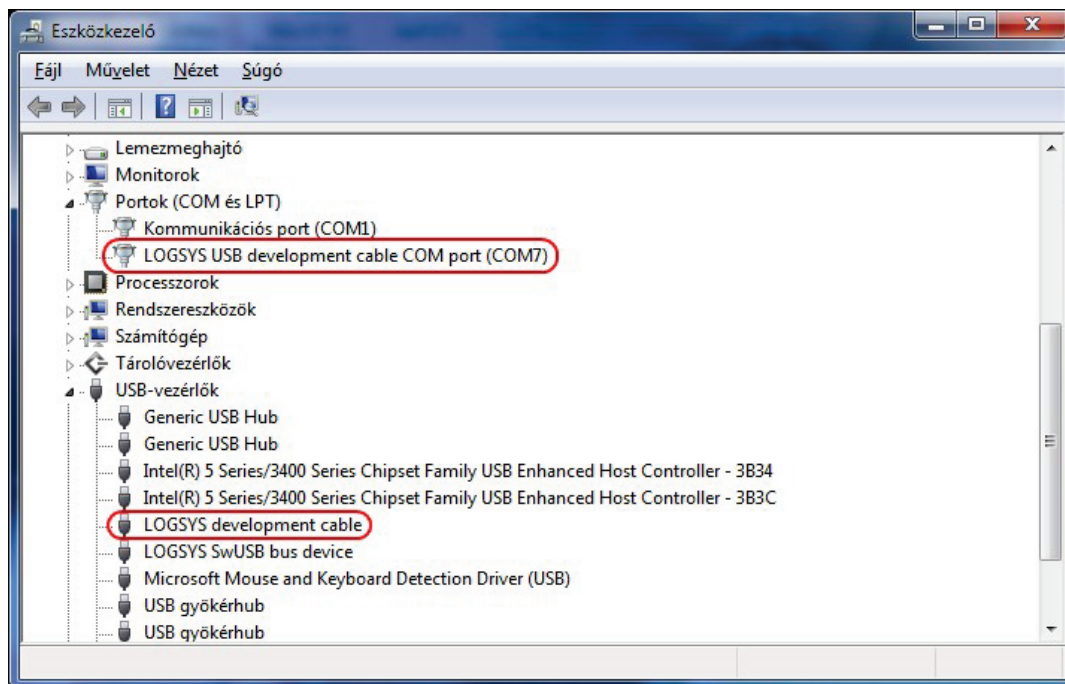


3-4. ábra: A LOGSYS fejlesztő kábel lengő csatlakozós kiegészítése.

A lengő csatlakozó használatával különböző egyszerűsített szolgáltatásokra is használható a fejlesztői kábel. Ezekben az esetekben a jelek egy részét kell csak használni. (Az ismertetésben a jelnevek rövidítéseit a LOGSYS fejlesztő kábel csatlakozójának színkódjával is kiemeltük.)

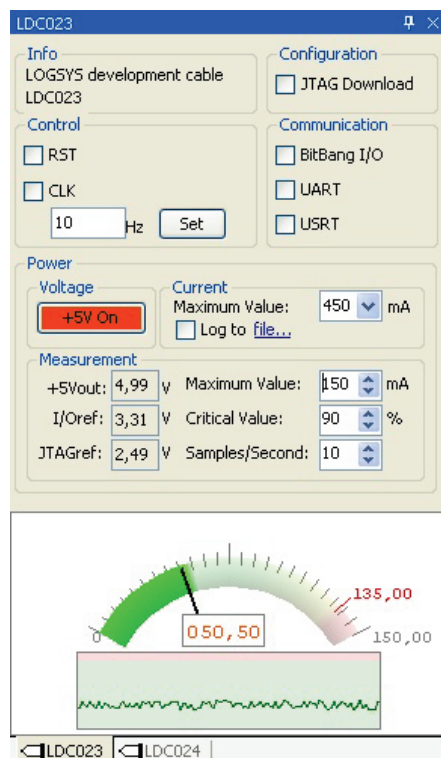
1. Tápfeszültség ellátás az **+5V** és **GND** jelekkel (Ez az USB specifikációnak megfelelően $5\text{ V} \pm 5\%$ lehet, azaz $4,75\text{ V} - 5,25\text{ V}$, tipikusan $4,85\text{ V}$). Az USB port kimenetén a rendelkezésre álló 450 mA (Y kábel 450/700/950 mA) maximális fogyasztás folyamatosan monitorozható.
2. További két csatornán feszültségmérés a $0\text{ V} - +5\text{ V}$ tartományban az **I/Oref**, **JTAGRef** jelekkel (A **GND** jel a negatív pont).
3. JTAG konfiguráció a **TMS**, **TCK**, **TDI**, **TDO** és **JTAGRef**, **GND** jelekkel. A **JTAGRef** által meghatározott jelszinteken, az IEEE 1149.1 JTAG protokoll szerint programozás, konfigurálás, peremfigyeléses tesztelés végezhető.
4. SPI kommunikáció vagy konfiguráció a **MOSI**, **MISO**, **CLK**, **RESET**, **I/Oref** és **GND** jelekkel. A **CLK**=SCK, **RESET**=CSn megfeleltetéssel a szokásos SPI kommunikációs protokoll használható.
5. UART soros kommunikáció a **MOSI**, **MISO**, **I/Oref** és **GND** jelekkel. Az eszköz Windows XP/Windows 7 operációs rendszerben virtuális soros port eszközként (VCP) jelenik meg. A kábel a PC host funkciójának megfelelően a **MOSI** = TX és **MISO** = RX vonalakon az eszközközlelőn keresztül beállítható módokban használható. Hardveres átvitelvezérlési lehetőség (RTS/CTS) nincs.
6. I²C/TWI/SMBUS kommunikáció a **MOSI**, **MISO**, **I/Oref** és **GND** jelekkel. A jelek megfeleltetése a következő: **MOSI** = SDA és **MISO** = SCL.
7. Szoftveres (Bit Bang) adat I/O és vezérlés a **MOSI**, **MISO**, **CLK**, **RESET**, **I/Oref** és **GND** jelekkel. Tetszőleges vezérlés és bitsoros kommunikáció kialakítható.

A fenti funkciók és természetesen a normál használat is megkívánja az eszközhöz tartozó szoftverkomponensek helyes telepítését. A szoftver komponenseket tartalmazó csomagokat az eszközök dokumentumainak legfrissebb verzióit tartalmazó <http://logsys.mit.bme.hu> oldalról érhetjük el a Letöltések menüpont alatt. A telepítőkészlet tartalmazza a különböző operációsrendszer verziókhoz szükséges (32 és 64 bites) csomagokat. A meghajtóprogram a telepítés során az USB kompozit eszközhöz kapcsolódóan 2 db eszközt telepít, ezek a LOGSYS USB virtual serial port (COMx) és a LOGSYS development cable. Sikeres telepítés után ezek az eszközközlelő Portok és USB vezérlők csoportjaiban megtekinthetők (3-5. ábra).



3-5. ábra: A telepített LOGSYS eszközök az Eszközkezelőben.

A LOGSYS grafikus felhasználói interfész kábelkezelő funkcióit a 3-6. ábra szemlélteti. A kezelői felület több kábel egyidejű használatát is támogatja, az aktív kábel azonosítóját az Info mezőben és a kiválasztó fülön (alul, a mérőműszerablak alatt) jelzi. A fejlesztő kábel kívánt funkcióit a Configuration, Control, Communication és Power mezőkben lehet kiválasztani. Egyidőben csak egy kábel kezelői felülete látható, de a beállított paramétereknek megfelelő működést az eszközválasztás nem befolyásolja.



3-6. ábra: A LOGSYS grafikus felhasználói interfész kábelkezelő funkciói.

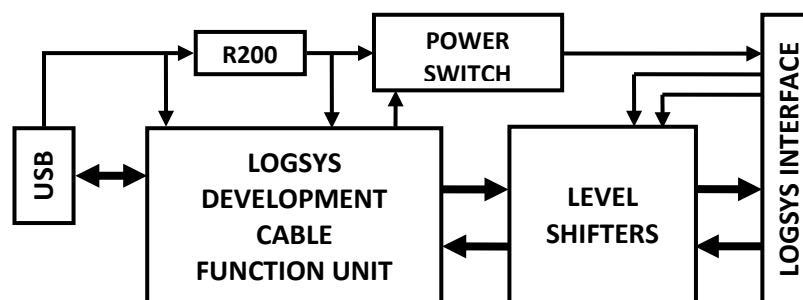
A tápfeszültség be- és kikapcsolása a Power mező Voltage területén a +5V On / +5V Off gomb működtetésével végezhető el. A Current mezőben beállítható a maximális kimeneti áramlimit, ami 450/700/950 mA lehet.

A Measurement mező numerikusan jelzi a mért feszültség értékeket (USB +5Vout kimenet, JTAGref, I/Oref bemenetek), és az árammérés eredményét analóg kijelzővel, numerikusan és folyamatában, az idő függvényében. A mérőműszer skálázása rugalmasan állítható, a mérések gyakorisága maximum 10 mérés/s. Az analóg kijelző a beállított mindenkor maximális érték 75 % felett sárga, ill. piros figyelmeztető jelzéssel jelenik meg (tehát pl. 300 mA kijelzési felbontás esetén 225 mA felett), de ez természetesen független a Current mezőben kiválasztott kiadható maximális áramlimit értékétől.

A Configuration, Control és Communication mezők egyedi funkcióinak kiválasztásakor egy-egy új kezelői panel nyílik meg, az aktuális üzemmód funkcióit felkínálva. A kommunikációs üzemmódok használata (az azonos interfész lábak miatt) kizárásos jellegű. Egy adott funkció kiválasztása más üzemmódok, vagy funkciók elérhetőségét blokkolhatja (pl. BitBang mód), ezt a kezelői ablak a megfelelő kiválasztó négyzetek elhalványításával jelzi. Ezeknek a funkcióknak a leírását a LOGSYS grafikus felhasználói felület ismerteti.

3.2.3 A LOGSYS fejlesztő kábel blokkvázlata

A LOGSYS fejlesztő kábel egyszerűsített blokkvázlata a 3-7. ábrán látható. Az eszköz az USB interfészre kapcsolva működőképes, a tápellátást az USB buszról igényelve. Az eszköz a regisztráció során nagyteljesítményű USB eszközként engedélyeződik, ami lehetővé teszi a teljes elérhető USB teljesítmény használatát a célrendszer tápellátására is. A teljesítménykapcsoló a bekapcsolási tranziens áramokat korlátozva engedélyezi a +5 V feszültség bekapcsolását, az áramfogyasztást a soros 0,2 Ω -os ellenálláson mérve. Ez természetesen okoz egy kis feszültségesést (max. 0,2 V, Y-kábel és 900 mA áramkorlát esetén), de ez általában a fejlesztés és mérések során megengedhető.



3-7. ábra: A LOGSYS fejlesztő kábel blokkvázlata.

A LOGSYS fejlesztő kábel interfésze tartalmazza a tápfeszültség, a konfiguráció és a kommunikációs vonalakat. A vezérlő egység biztosítja az USB funkciók kezelését, a parancsok értelmezését, a kommunikációs funkciók végrehajtását. A vezérlő és az interfész vonalak között jelszint illesztő meghajtók biztosítják, hogy a célrendszer által megkívánt jelszinten vezéreljük a konfigurációs és kommunikációs vonalakat. Mindkét interfész egység vonalai az 1,65 V – 5,5 V feszültségtartományba eső jelszinteket tudnak fogadni és meghajtani. A meghajtók célrendszer oldali áramellátását a célrendszer biztosítja a JTAGref és az I/Oref vonalakon keresztül. Az itt fellépő áramigény minimális, gyakorlatilag a meghajtandó CMOS bemenetek áramát igényli, tehát a célrendszer energiaellátását semmilyen módon nem befolyásolja.

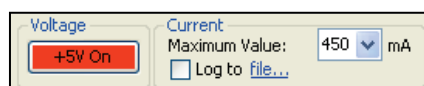
3.3 A LOGSYS fejlesztő kábel funkciói

3.3.1 A tápellátási funkciók

A LOGSYS fejlesztő kábel tápellátási funkciói az USB interfész lehetőségeinek és korlátainak megfelelőek. Az USB 2.0 interfészek alapértelmezésben 100 mA áramfogyasztást biztosítanak, de az eszközök paramétereik alapján igényelhetnek nagyobb tápáramot, maximum 500 mA-t. Ezt az értéket a rövid idejű tranziensek sem léphetik túl. Ennek megfelelően, nem csak a bekapcsolási folyamat áramkorlátozott, de a működés teljes folyamata alatt is érvényes az áramlimit figyelése.

A kábel a +5 V névleges tápfeszültséget az USB port kimeneti feszültsége alapján biztosítja. Ez a szabványban megadott $+5\text{ V} \pm 5\%$ tűrésnek és a gyakorlati tapasztalatoknak megfelelően mindig kisebb, mint 5 V, jellemzően 4,8 V és 4,9 V közötti érték (de pl. a 3-6. ábrán 4,99 V!).

A működés során az árammérés egy soros $0,2\ \Omega$ -os ellenálláson keresztül történik. Ez további feszültségesést okozhat, tehát a kimeneti feszültség várható értéke általában 4,6 V és 4,8 V közötti érték.



3-8. ábra: A tápfeszültség bekapcsolás és áramlimit választás.

A megengedett 450 mA-nél nagyobb áramfelvételű célrendszerek fejlesztése az ún. Y-típusú USB kábel használatával lehetséges. A 3-9. ábrán látható Y-kábel két USB A típusú csatlakozóval rendelkezik.



3-9. ábra: Az Y-típusú USB kábel kettőzött USB A csatlakozóval.

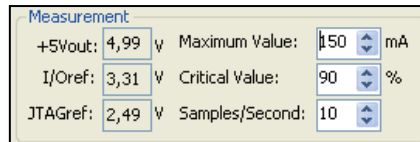
Az Y-kábel elsődleges A csatlakozója (a kép közepén) biztosítja a valódi USB kapcsolat kialakítását, a kiegészítő A csatlakozó (a kép bal szélén) csak a +5V és GND jeleket tartalmazza. Figyeljünk oda, hogy csak azonos USB HUB-hoz tartozó USB csatlakozókba dugjuk be a két csatlakozót, egyébként a két teljesítmény kimenet esetlegesen eltérő névleges feszültsége kiegyenlítő áramokat okozhat az USB kimeneti források között.

Az Y-kábel használatával a két USB kimeneti portról együttesen a névleges áram kétszerese nyerhető. A LOGSYS fejlesztő kábellel a megnövelt áramterhelhetőség esetén két áramkorlát állítható be, a névleges 700 mA és a 950 mA, a kábel szoftver funkcióin keresztül.

3.3.2 A mérési funkciók

A LOGSYS fejlesztő kábel a tápellátáson felül biztosítja a célrendszer fontosabb tápellátási paramétereinek mérhetőségét is. Beállítható gyakorisággal, maximum 10 mérés/másodperc, méri a célrendszer áramfelvételét, a kiadott 5 V tápfeszültség értékét és a JTAGref ill. I/Oref pontokon a konfiguráció és a kommunikációs interfészek referencia feszültségeit (3-10. ábra).

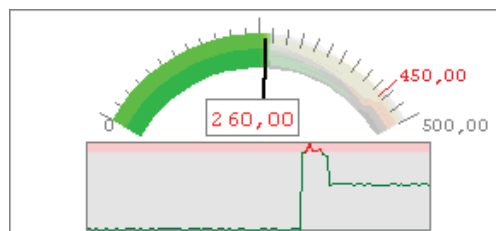
A feszültségmérések névleges méréshatára +5 V, a relatív pontosság 2 %. Alapvetően tájékoztatási szerepük van, amihez ez a pontosság elegendő.



3-10. ábra: A feszültségmérési eredmények és árammérési paraméterek.

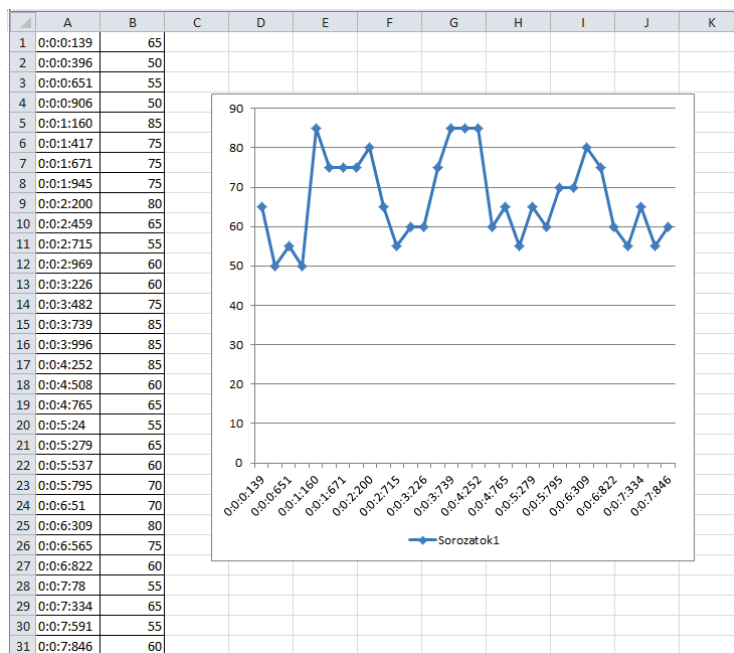
Az árammérés két méréshatárban történik, a szükséges nagy mérési tartomány átfogása érdekében. A méréshatár váltása az eszközben automatikus, a felhasználó ezt nem befolyásolhatja. A megvalósított árammérési határok 64 mA, illetve 1280 mA. A relatív pontosság az árammérések esetében is 2 %.

A kezelői felületen az árammérés kijelzésének (3-11. ábra) végértéke az aktuálisan használt méréshatártól függetlenül tetszés szerint skálázható. A végérték és kritikus érték megfelelő megválasztása jó vizuális visszacsatolást jelent.



3-11. ábra: Az árammérés kijelző panel.

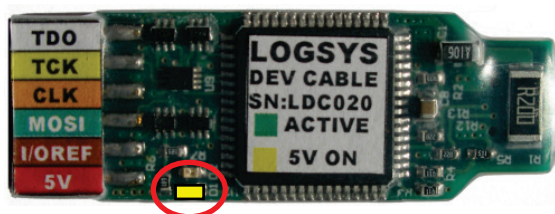
A LOGSYS grafikus felhasználói felület használatakor lehetséges az áramfelvétel hosszúidejű monitorozása és a mért értékek fájlba mentése a későbbi analízis céljából. A tápellátási funkciók használatát a szoftver leírás ismerteti részletesen. A mindenkori pillanatnyi áramfogyasztást a mutatós műszer jelzi, a rövid idejű korábbi értékekről egy kis grafikon ad tájékoztatást. A mért értékek fájlba menthetők, későbbi analízis, ill. archiválás céljából. A log fájl formátuma egyszerű időskála, amely egy megfelelő alkalmazásba (pl. Excel) importálva könnyen ábrázolható (3-12. ábra). Az időskála formátuma h:m:s:ms, az áramérték mA egységben értendő.



3-12. ábra Az áramfelvétel mérés elmentett adatai és ábrázolásuk grafikonban

A LOGSYS fejlesztőkábel tápfeszültség kivezetéseinek állapotáról a beépített sárga színű LED tájékoztat (3-13. ábra):

-  • A LED nem ég, ha nincs feszültség a tápellátást biztosító kivezetéseken.
-  • A LED folyamatosan ég, ha a fejlesztői kábel +5 V tápfeszültséget ad a célrendszernek.
-  • A LED villog, ha a fejlesztői kábel nem ad ki +5 V tápfeszültséget, de valamelyik tápellátással kapcsolatos jelen (+5V, JTAGref, IOREf) 0,2 V-nál nagyobb feszültség van.



3-13. ábra: A tápfeszültség állapotát jelző sárga LED.

3.3.3 Konfigurációs és programozási funkciók

A LOGSYS fejlesztő kábel elsődleges funkciója a programozható logikai áramkörök és mikrovezérlők konfigurációja, programozása. A feladat jellegéből következően különböző követelményeknek és sokszor egyedi megoldásoknak kellene megfelelni. Szerencsére az áramkörök döntő többsége az egyedi speciális megoldásokon túl biztosítja az ipari szabvány JTAG interfészen keresztüli programozhatóságot. A JTAG interfészt eredetileg tesztelési célokra fejlesztették ki, de általános jellege és használhatósága hamar népszerűvé tette a programozás funkciókra is. Hasonló módon sok mikrovezérlő és mikroprocesszor alkalmazza hibakeresési, illetve debug interfészként is.

A LOGSYS fejlesztő kábel egy teljes értékű JTAG interfészt valósít meg. A JTAG interfész vonalak TMS, TCK, TDI, TDO meghajtása a JTAGref ponton érzékelt feszültségről történik, ami +1,65 V és +5,5 V közötti érték lehet. Jelenleg tipikus a +3,3 V vagy a +2,5 V használata, de terjed a +1,8 V-os

jelszint is. Az egyes áramkörökön megvalósított opcionális TRST jelet a fejlesztő kábel nem biztosítja, de ez a használhatóságot nem veszélyezteti, mivel minden IEEE149.1 szabványnak megfelelő működésű TAP vezérlő a JTAG protokoll alapján a TMS és TCK jelekkel mindig átvezethető a TEST LOGIC RESET állapotba.

A LOGSYS fejlesztő kábel a konfigurálható eszközöket a JTAG protokoll szerinti konfigurációs módot használva programozza és a konfigurációs adatokat az az SVF (Serial Vector Format) formátum szerint olvassa be. Minden TAP vezérlő állapot átmeneti parancsot, mintavételt és adatátvitelt az SVF parancsokkal hajtunk végre, ami az elfogadott szabványos értelmezésnek megfelelően egyértelmű állapotvezérlést és adatkommunikációt biztosít. Az SVF parancsok egyszerű ASCII karaktersorozatokként adhatók meg egy szövegfájlban. Ez a szövegfájl az FPGA tervező rendszerekkel, vagy mikrovezérlő fejlesztői környezetekkel generálható.

A JTAG Download funkció a konfigurációhoz a kiválasztott forrásfájl alapján dönti el a szükséges lépéseket. Amennyiben a forrásfájl SVF fájl, a konfiguráció közvetlenül megtörténik. Ez egyúttal a leggyorsabb konfiguráció letöltési mód.

3.3.4 Gyártóspecifikus konfigurációs fájlok támogatása

Ha pl. a Xilinx FPGA eszközöknél BIT, vagy a Xilinx CPLD eszközöknél JED típusú forrásfájl jelölünk ki és a Xilinx WebPack vagy ISE környezetet megtalálható az adott gépen, akkor a Xilinx iMPACT alkalmazás háttérben történő meghívásával az SVF átalakítás automatikusan megtörténik. Más gyártók egyedi adatformátumainak kezelésére hasonló megoldások kidolgozhatók, de a LOGSYS grafikus felhasználói felület jelenleg csak a Xilinx formátumokat támogatja.

3.4 Kommunikációs funkciók

A LOGSYS fejlesztő kábel széleskörű kommunikációs funkciókkal rendelkezik, melyek natív módon elérhetőek a PC Windows operációs rendszeréből. A fizikailag rendelkezésre álló 4 vezetéken különböző üzemmódok szerint valósíthatunk meg adatátvitelt. A LOGSYS fejlesztő kábel az USB porton keresztül nyújtja a kommunikációs szolgáltatásokat. A szolgáltatások egy része konkurens módon, azonos időben is elérhető, azonban bizonyos szolgáltatások egyidejű használata kizárja egymást. Alapvetően elmondható, hogy a konfigurációs és adatátviteli szolgáltatások nagyrészt függetlenek egymástól.

A LOGSYS fejlesztő kábel jeleinek funkcióit a 3-1. táblázat tartalmazza. A kommunikációs aktivitásról a beépített zöld színű LED tájékoztat (3-14. ábra):

-  • A LED nem ég, ha egyetlen kommunikációs funkciót sem kapcsoltunk be.
-  • A LED folyamatosan ég, ha legalább egy funkció aktív, de nem történik átvitel.
-  • A LED villog, ha legalább egy funkció aktív és valamilyen adatátvitel van folyamatban.



3-14. ábra: Az aktív kommunikációt jelző zöld LED.

3-1. táblázat: A LOGSYS fejlesztő kábel jeleinek funkciói.

Funkciók	A fejlesztőkábel I/O kivezetései							
	TDI	TDO	TMS	TCK	CLK	RST	MOSI	MISO
JTAG	TDI	TDO	TMS	TCK				
PIC ISP		PGD ¹	PGM	PGC		RST ²		
UART					CLK ³	RST ⁴	TXD ⁵	RXD ⁵
USRT					CLK ⁶	RST ⁴	TXD ⁶	RXD ⁶
SPI					SCK ⁷	CSn ⁷	MOSI ⁷	MISO ⁷
I ² C					CLK ³	RST ⁴	SDA ⁸	SCL ⁸
BitBang I/O					CLK ⁹	RST ⁹	MOSI ⁹	MISO ⁹

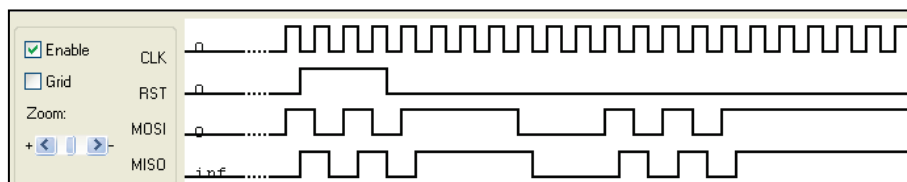
Megjegyzések:

1. A PIC mikrovezérlők LVP programozásakor kétirányú adatvonal.
2. A jel a PIC MCLRn jelét vezérli a programozás alatt, az alkalmazás tesztelésekor normál RST vezérlőjel.
3. A LOGSYS grafikus felhasználói felületen a kommunikációtól független, tetszőlegesen beállítható órajel, az UART RXD/TXD adatsebességet nem befolyásolja.
4. A LOGSYS grafikus felhasználói felületen az RST jel szabadon vezérelhető.
5. A TXD és RXD vonalak adatsebessége a szabványos soros adatátviteli értékek szerinti.
6. A TXD és RXD vonalak adatsebessége a CLK frekvenciával azonos.
7. Szabványos SPI kommunikációs vonalak, MASTER üzemmódban.
8. Szabványos I²C/TWI/SMBUS interfész.
9. A LOGSYS grafikus felhasználói felület szoftveres bitszintű adatátvittele.

A LOGSYS fejlesztő kábel kommunikációs szolgáltatásai a beágyazott rendszerek alkalmazásainak fejlesztésénél használhatók ki. A rendelkezésre álló opciók a leggyakrabban használt soros kommunikációs interfészekhez illeszkednek, kiegészítve a speciális BitBang szoftveres protokollal.

3.4.1 Szoftveres (BitBang) adatátviteli mód

A szoftveres BitBang mód célja az egyszerű, demonstrációs vagy oktatási célú FPGA mintaalkalmazások vezérlése. A BitBang módban a rendelkezésre álló jelek, mint CLK, RST, MOSI és MISO (3-15. ábra) tetszőlegesen vezérelhetők, de a RST és MOSI jelek vezérlése, kimeneti értékük megváltoztatása csak aktív CLK órajel mellett lehetséges, azaz az előírt változások órajel szinkron módon jutnak érvényre.



3-15. ábra: A BitBang mód vezérlőjelei és időzítésük.

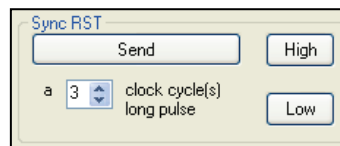
A BitBang funkció szoftveres CLK órajel frekvenciája az 1Hz és 1000Hz frekvenciatartományban tetszőlegesen állítható. Lehetséges a kiválasztott frekvencián adott számú órajel pulzus kiadása, továbbá az órajel fel- vagy lefutó élenkénti léptetése is a Clock mező High és Low felíratú gombjaival (3-16. ábra).



3-16. ábra: A BitBang mód órajel vezérlési lehetőségei.

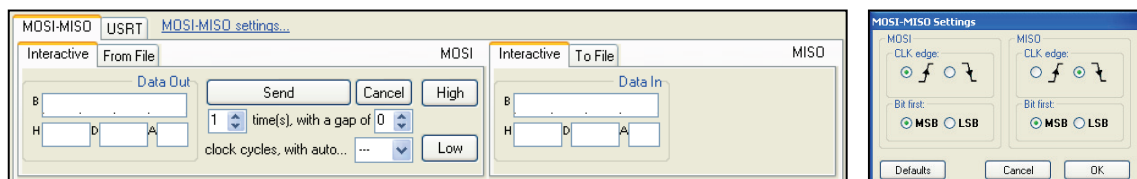
A maximális frekvencia 1000 Hz, ez az operációs rendszer ütemezési tulajdonságai és a szoftveres adatfeldolgozás korlátai miatt nem lehet magasabb. A beállított új érték a Frequency settings mező Set gombjával érvényesíthető. A szoftveres BitBang módban, különösen a „magasabb” frekvenciákon a Windows operációs rendszerek nem megbízható időzítése miatt a rövid idejű periódusidő stabilitás nem biztosítható, ezért a folytonos órajelen rövidebb ideig kisebb szünetek jelenhetnek meg, illetve a kitöltési tényező ciklusról-ciklusra változva nem lesz pontosan 50 %-os. Ez a frekvenciaingadozás a tesztelt logikai áramkörök működését nem befolyásolja, hiszen a szinkron áramkörök szempontjából az idő csak mint az órajel váltása jelenik meg, a konkrét intervallum hossza két váltás között ezen a sebességen indifferens. Az alacsonyfrekvenciás működés problémamentes, az operációs rendszer ütemezési és kommunikációs problémái elhanyagolható relatív periódusidő ingadozást okoznak a 10 Hz-nél kisebb frekvenciákon.

A BitBang üzemmód RST jele az órajellel szinkron módon állítható, statikusan magas vagy alacsony szintre (3-17 ábra). Lehetséges adott hosszúságú impulzus kiadása is, az aktuális (inaktív RST) szinttel ellentétes értékkel.



3-17. ábra: A BitBang mód RST jel vezérlési lehetőségei.

A MOSI és MISO adatátviteli jelek az órajellel szinkronban változtatható (MOSI), illetve mintavételezhető (MISO) soros adatbitek (3-18. ábra). A vezérléstől függően használhatók mint szabadon vezérelhető jelek, melyeken az adat sorosan lép ki, illetve be. Beállítható a kiléptetés iránya (MSB/LSB) illetve ütemezése is (felfutó/lefutó él).



3-18. ábra: A BitBang mód adatátviteli vezérlő panelje.

Az adat megadható bináris, hexadecimális, decimális vagy ASCII formátumban és kiküldhető ismételten, adott számú órajelperiódus hosszúságú szünettel a minták között. Az ismétlések során egyszerű műveletek, mint inkrementálás, shiftelés, rotálás beállíthatók, de választható véletlenszerű minta generálása is a kezdőérték alapján. A lehetőségeket a LOGSYS grafikus felhasználói felület leírása ismerteti.

Az adatvonalakra tesztadat fájlból is kiadható mintasorozat, illetve a vett adatfolyam fájlba menthető.

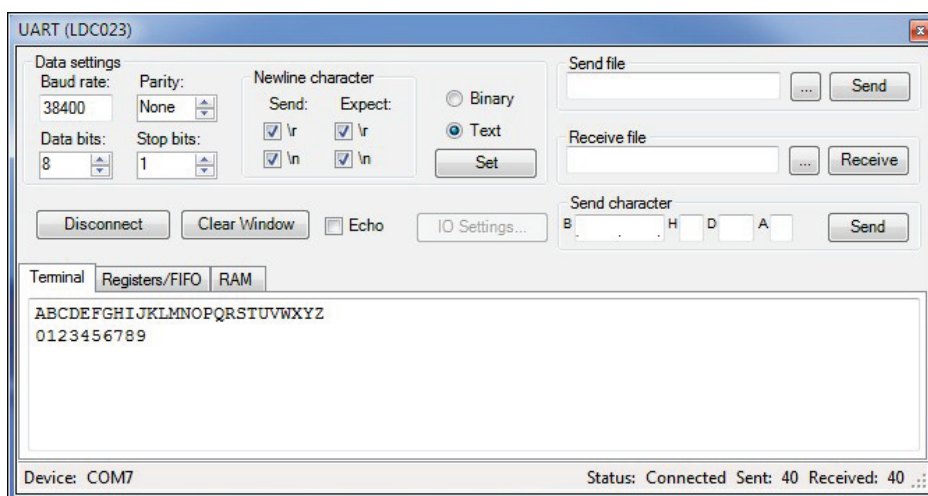
A BitBang mód kiegészítése az adatvonalak soros USRT protokollnak megfelelő használata. Ebben az esetben az átvitel keretezett karakterekből áll, az inaktív alapérték magas szintű, az átvitt az 1 órajel hosszú alacsony START bit indítja és választható számú adatbit (és az opcionális paritás bit) után az egy órajel hosszú STOP bit zárja. A kimeneti adatokat egy a fentihez hasonló kezelői panelen adhatjuk meg vagy fájlból olvashatjuk be. A karakterek keretezését a LOGSYS grafikus felhasználói felület automatikusan elvégzi. A bemeneti MISO jelen a beállított adatformátumnak megfelelően a keretformátum ellenőrzésével adatszűrést végezhető. A MOSI (TX) vonal vezérlése és MISO (RX) vonal mintavételezése ellentétes órajel éleknél történik.

3.4.2 UART mód

Az UART mód a hagyományosan leggyakrabban használt aszinkron soros adatátviteli mód. Legnagyobb előnye, hogy a két adatátviteli vonalon MOSI (TX), MISO (RX) kívül más jelet a kapcsolat felépítéséhez hardveres átvitelvezérlés nélküli módban nem igényel. Az UART protokoll szerinti kommunikációt az iparban különböző fizikai jelszinteken és meghajtási módokkal szokás alkalmazni, mint pl. RS232, RS485 vagy RS422, amelyek a zavarvédeltséget javítják. Mivel a beágyazott rendszereken belül ezen jelszintek használatának nincs jelentősége, az adatátviteli jelszintek az 1,65 V és 5 V közötti TTL/CMOS jelszinteknek megfelelő előírások szerint működnek, aszimmetrikus (egyvezetékes) meghajtást alkalmazva.

A LOGSYS fejlesztő kábel, mint virtuális soros port típusú eszköz paraméterei az adott Windows alkalmazásban állíthatók be. Az adatátviteli sebesség 4800 bit/s és 115200 bit/s között választható meg, az adatbitek száma, a paritás típusa és a STOP bitek száma a szokásos opciók szerint állítható be. A kábel a hardver átvitelvezérlést nem támogatja. Az eszköz a szokásos terminál programokkal (pl. HyperTerminal) vagy más COM port kezelői funkcióval rendelkező alkalmazásokkal egyszerűen használható.

A LOGSYS grafikus felhasználói felület a használat egyszerűsítésére egy mini terminál ablakot kínál (3-19. ábra), amely könnyű beállíthatóságot és rugalmas kezelést biztosít. A kezelői ablak az UART mód kiválasztásával nyitható meg. Részletes ismertetése a LOGSYS grafikus felhasználói felület leírásánál található.



3-19. ábra: A terminál alkalmazás ablaka UART módban.

3.4.3 USRT mód

Ebben az üzemmódban a CLK kommunikációs órajellel szinkronban, az UART átvitelhez hasonló, szabványos sebességű keretezett adatátvitel zajlik. A szinkron átviteli mód miatt nem szükséges az adatvonalak megbízható felismerése érdekében túlmintavételezés. Ez egyrészt javítja a relatív adatátviteli sebességet, másrészt egyszerűsíti a szükséges interfészt. A fejlesztő kábel a MOSI (TX) vonalra a következő bitet a CLK órajel felfutó élére adja ki, a MISO (RX) vonal értékét pedig a CLK órajel lefutó élére mintavételezi.

A LOGSYS fejlesztő kábel USRT módja saját fejlesztésű Windows alkalmazásokból érhető el, a működési paraméterek az adott alkalmazásban adhatók meg. Az operációs rendszer szempontjából az UART/USRT üzemmód lényegében nem különböztethető meg egymástól. Az átviteli sebesség a 4800 bit/s és 115200 bit/s között választható, az adatátviteli bitek, a paritás és a STOP bitek száma a szokásos opciók szerint állítható be. A kábel a hardveres átvitelvezérlést nem támogatja.

A LOGSYS grafikus felhasználói felületen rendelkezésre álló terminálablakban az USRT mód az UART móddal azonos felületen érhető el.

3.4.4 SPI mód

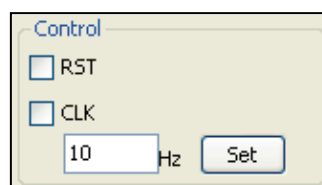
A LOGSYS fejlesztő kábel támogatja az SPI protokoll szerinti adatátvitelt. Ennek használata elsősorban az FPGA kártyákon található soros konfigurációs SPI FLASH memóriák programozása szempontjából előnyös, erről külön alkalmazás gondoskodik. Ez a mód a LOGSYS grafikus felhasználói felületen közvetlenül nem érhető el, későbbi beépítése lehetséges. Az SPI interfészen a szokásos adatátviteli módok paramétereizhetők.

3.4.5 I²C/TWI/SMBUS mód

A LOGSYS fejlesztő kábel támogatja az I²C/TWI/SMBUS protokollok szerinti adatátvitelt. Ennek használata elsősorban a különböző speciális szenzorok, integrált áramkörök közvetlen vezérlése, tesztelése szempontjából előnyös, ehhez külön alkalmazás készült. Ez a mód a LOGSYS grafikus felhasználói felületen közvetlenül nem érhető el, későbbi beépítése lehetséges. Az I²C/TWI/SMBUS interfész képes a 100kHz és a 400kHz sebességű adatátvitelre is.

3.4.6 Vezérlő jelek kiadása

Amennyiben a LOGSYS fejlesztő kábel aktuális használati módja nem igényli a CLK és RST jelek adott adatátviteli protokoll szerinti használatát, ezek a jelek szabadon vezérelhető jelekként rendelkezésre állnak.

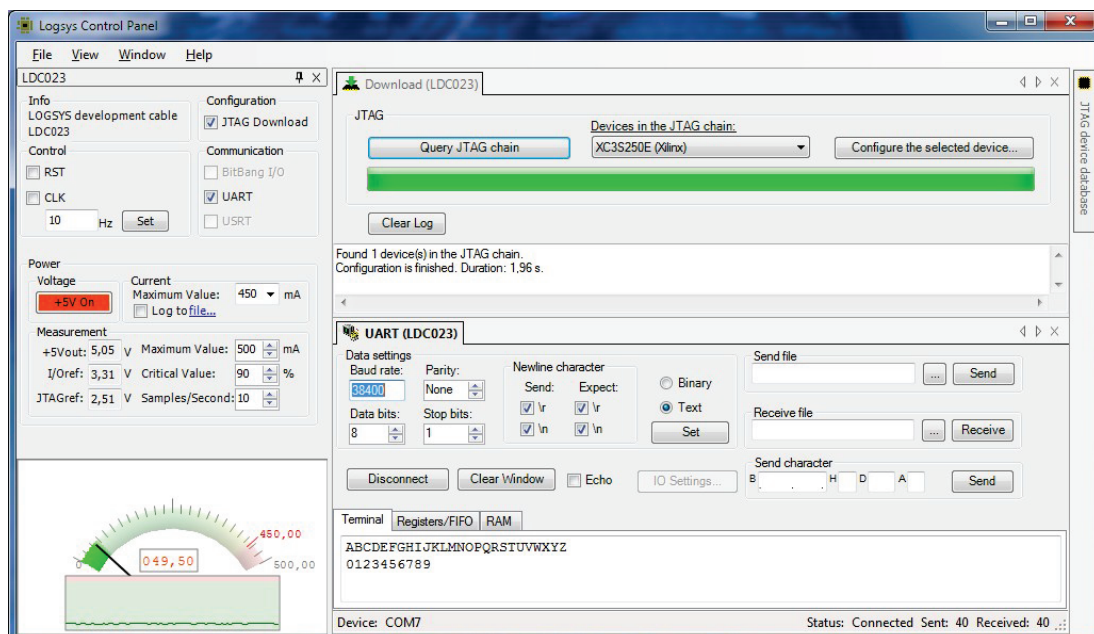


3-20. ábra: A szabadon maradó CLK és RST jelek kezelő ablaka.

A RST jel manuálisan vezérelhető, az órajelhez képest aszinkron módban (akár egy nyomógomb) változtatható pozitív logika szerint értelmezett beállító jel. Alaphelyzete az alacsony szint, a jelölőnégyzetet kiválasztva magas értéket vesz fel.

A CLK órajel egy maximum 8 MHz frekvenciájú, kvarcpontosságú, szimmetrikus négyszögjel, melynek beállítható frekvenciáit a bináris osztó értékei szabják meg. Az órajel frekvenciája a következőképpen számolható ki: $8 \text{ MHz} / (P * N)$, ahol P értéke 1, 8, 64, 256 és 1024 lehet, illetve N értéke 1 és 2^{16} közé eshet. A vezérlő ablakban a kívánt frekvencia értékét valós számként adhatjuk meg Hz-ben, ami 0,2 és 8000000 közötti érték lehet. A LOGSYS grafikus felhasználói felület a beírt értékhez legközelebbi beállítható értéket fogja választani.

4 LOGSYS grafikus felhasználói interfész



4.1 Bevezetés

Az elektronika gyors ütemű fejlődésének köszönhetően a programozható logikai eszközökkel (FPGA-k és CPLD-k) valamint mikrokontrollerekkel (μC) felépített rendszerek egyre szélesebb területen kerülnek alkalmazásra. Nagy előnyük az alkalmazás-specifikus integrált áramkörök (ASIC) tartalmazó rendszerekhez képest a programozhatóságukból adódó rugalmasság, amely nagymértékben csökkenti a rendszer prototípusának kifejlesztése során jelentkező költségeket és az első mintapéldányok piacra kerülési idejét. Az újszerű eszközökkel való munka újszerű segédeszközöket igényel a programozható alkatrészek előnyeinek minél jobb kihasználása végett.

Egy ilyen rendszerrel szemben az alapvető használati követelmények a programozható eszközök beágyazott rendszerekben történő alkalmazásából adódnak. A fejlesztő környezetnek képesnek kell lennie kiaknázni az eszközök programozható természetéből fakadó rugalmasságát, vagyis támogatnia kell ezek konfigurációját szabványos interfészekon keresztül, gyorsan, egyszerűen és megbízhatóan, az általánosan elterjedt eszközök minél szélesebb skáláját lefedve. A beágyazott alkalmazások miatt pedig az általában korlátozott kommunikációs lehetőségeket kell hatékonyan és kényelmesen elérhetővé tenni a fejlesztő számára a produktivitást elősegítő kommunikációs funkciók formájában. Végül nem szabad megfeledezni a célrendszerek tápfeszültség-ellátásáról sem, amit szintén célszerű támogatnia a fejlesztő környezetnek, mind a prototípus külső hardverigényének csökkentése, mind az energiafogyasztás ellenőrzése végett. Mindezek mellett a beágyazott rendszerek széleskörű elterjedése miatt könnyen kezelhetőnek és olcsónak kell lennie.

A LOGSYS grafikus felhasználói interfész (GUI) ezen követelmények szellemében készült. Ezen fejezet célja a LOGSYS FPGA kártyákhoz és a LOGSYS fejlesztő kábelhez kapcsolódó rendszer kezelői felületét megtestesítő alkalmazói program ismertetése. A továbbiakban először áttekintjük az alkalmazói program helyét a rendszerben és az általa megvalósított funkcionalitást, majd a felület általános jellemzése után a konkrét funkciók részleteit ismertetjük.

4.2 A LOGSYS rendszer

A rendszer felépítése a 4-1. ábrán látható. A rendszer alapvető komponensei a LOGSYS FPGA fejlesztői kártya (vagy más célhardver), a LOGSYS fejlesztő kábel, továbbá a PC host gép a LOGSYS grafikus felhasználói felülettel. A számítógép USB interfészen keresztül valósítja meg a fejlesztett rendszer vezérlését. A célhardverhez a LOGSYS fejlesztő kábelen kialakított speciális 12 pólusú LOGSYS porton keresztül csatlakozik, amely az eszközök tápellátásához, konfigurálásához, valamint a velük való kommunikációhoz nyújt fizikai felületet. A LOGSYS fejlesztő kábelt eszközmeghajtó illeszti a Windows operációs rendszerhez, ahol a LOGSYS grafikus felhasználói interfész valósítja meg a beágyazott rendszerek fejlesztését támogató funkciókat. Ez a program jelenti a LOGSYS rendszer grafikus felhasználói felületét a felhasználó felé, ennek ismertetése a fejezet célja.

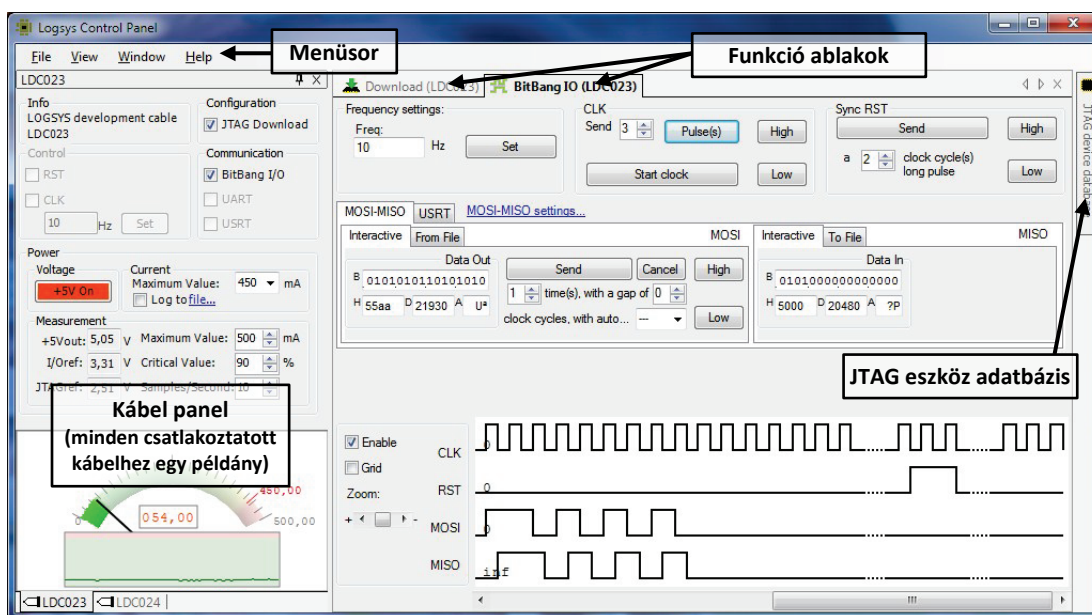


4-1. ábra: A LOGSYS rendszer felépítése.

4.3 A LOGSYS grafikus felhasználói felület általános ismertetése

A program kezelői felülete (4-2. ábra) szabadon áthelyezhető (dokkolható) ablakokból épül fel. A LOGSYS GUI fejlesztőrendszer a céleszközökkel egy vagy több kábel egy vagy több funkcióját használva érintkezik, így a kezelői felület is ezt a felépítést tükrözi: minden kábel funkcióját egy külön kábel panelen keresztül érhetjük el, vagy az onnan elindított ablakon keresztül használhatjuk. A rendszer több kábel együttes használatát is támogatja, sőt, ezek bármikor szabadon csatlakoztathatók és eltávolíthatók az USB portról, a program állapota automatikusan követi a változásokat (csatlakoztatás esetén megjelenik egy új kábel panel, eltávolításnál pedig eltűnik és a kábel funkció ablakai is bezáródnak).

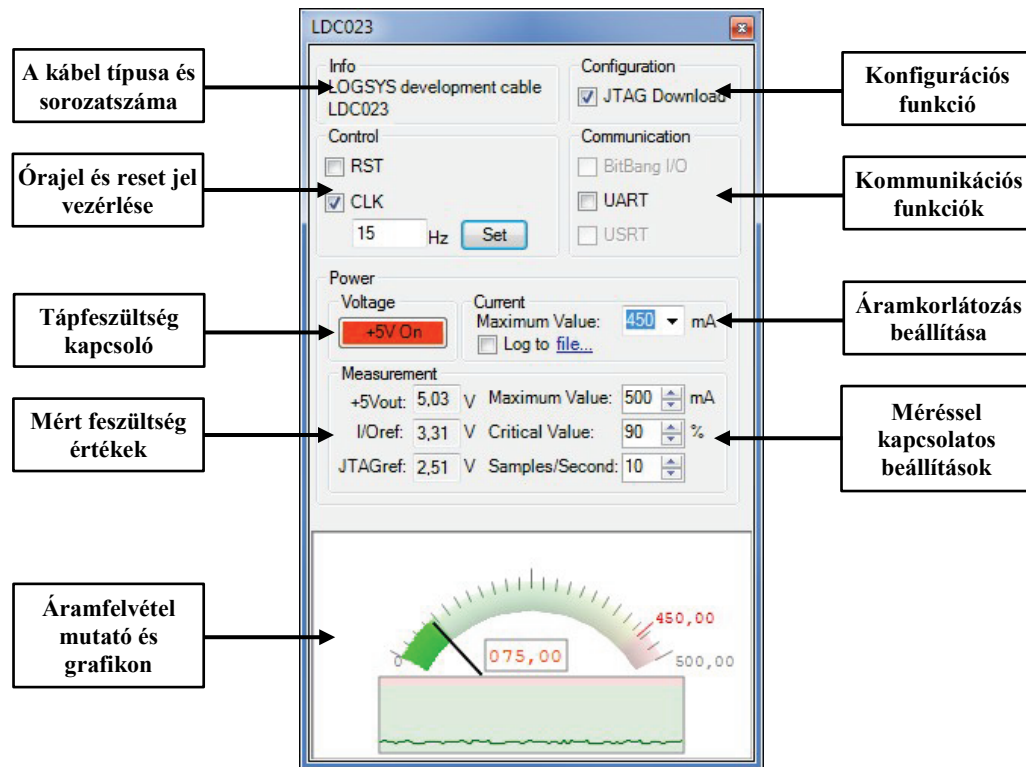
A kábelek egyszerűbben használható funkciói a kábel panelről, míg az összetettebb vezérlést igénylő funkciók külön ablakokból érhetők el. A továbbiakban ezeket ismertetjük részletesen.



4-2. ábra: A program kezelői felülete.

4.4 A kábel panel

Ez a panel szolgál az egyes kábelek funkcióinak elérésére (tápfeszültség és vezérlő jelek) vagy a hozzájuk tartozó ablak megjelenítésére (letöltés és kommunikáció). A panel képe a 4-3. ábrán látható.



4-3. ábra: A kábel panel.

A panel bal felső sarkában az Info mezőben az adott kábel típusa és sorozatszáma látható (pl. LDC007). Utóbbi a kábel egyedi azonosítójaként szolgál, és megjelenik minden ablak fejlécében, ami ehhez a kábelhez tartozik.

A letöltés a JTAG interfészen keresztül támogatott. Az ehhez tartozó ablakot a jobb felső sarokban a Configuration mezőben található jelölőnégyzet bekapcsolásával indíthatjuk el. Ez általánosan is igaz a külön ablakban induló funkciókra: a hozzájuk tartozó jelölőnégyzet bekapcsolásával elindíthatjuk, kikapcsolásával bezárhatjuk a funkcióhoz tartozó ablakot. Ha az ablakot egyéb módon zárjuk be, a jelölőnégyzet állapota ennek megfelelően frissül.

A konfiguráció alatt a Communication mezőben találhatók a kommunikációs funkciókat indító jelölőnégyzetek. Ahogy az ábrán is megfigyelhető, néhány közülük nem engedélyezett. Ennek oka, hogy a LOGSYS port egyes vonalait nem lehet egyszerre több funkció kiszolgálására felhasználni, így az azonos vonalakat használó funkciók (tipikusan a kommunikációs funkciók) egyszerre nem futhatnak ugyanazon a kábelén. Emiatt az ütközést eredményező funkciók jelölőnégyzetei automatikusan letiltódnak, de az adott ablak bezárása után újra elérhetővé válnak.

A kommunikációs funkcióktól balra helyezkedik el a vezérlő jelek (órajel és reset) kezelésére szolgáló Control mező. Itt három dolgot tehetünk: beállíthatjuk az órajel frekvenciáját, az órajel jelölőnégyzetét be- és kikapcsolva engedélyezhetjük, illetve letilthatjuk az órajelet, valamint a

reset jel jelölőnégyzetével magas vagy alacsony állapotba hozhatjuk a reset vonalat. A funkciók összeférhetetlensége itt is eredményezheti az egyes jelölőnégyzetek letiltását: a szinkron kommunikációs funkciók átveszik az órajel vonal feletti irányítást, így az innen ezekben az esetekben nem vezérelhető.

A panel alsó részén a Power mezőben kapott helyet az energiaellátás vezérlésére és mérésére szolgáló mező. Itt lehetőségünk van mindenekelőtt a Voltage mezőben az +5V-os tápfeszültség kapcsolgatására (+5V On / +5V Off), úgy, hogy a gomb aktív állapotát a kapcsoló gomb piros színe jelzi. A gomb melletti Current mezőben áramkorlátot állíthatunk be a célrendszer számára: ennél nagyobb áramfelvétel esetén a fejlesztő kábel lekapcsolja a tápellátást.

A panel maradék területén a Measurement mezőben a méréssel kapcsolatos vezérlők és kijelzők foglalnak helyet. A LOGSYS port három tápfeszültség csatlakoztatásra szolgáltató vonalának pontos értékeit mutatják a bal oldali szövegdobozok. A célrendszer áramfelvételét egy mutatós műszer jelzi, korábbi értékek egy grafikonon láthatók. A grafikon és a műszer tulajdonságai a jobb oldali vezérlőkkel állíthatók be. A legfelső mező a műszer maximális kitéréséhez tartozó áramértéket jelenti. A középső mezőben megadható, hogy ennek hány százaléka legyen az a kritikus áramérték, ami felett a mutató és a grafikon a piros tartományban ábrázolja a fogyasztást. A legalsó mező a grafikon másodpercenként vett mintáinak számát tartalmazza.

A grafikonon a piros terület és a maximális érték is a mutatós műszerrel összhangban változik. Vegyük figyelembe, hogy méréshatár-váltás esetén a grafikon korábbi adatai még a régi méréshatárhoz tartoznak, és ezért az új pontokkal nem hasonlíthatók össze. A grafikon működése során fix számú, kb. 10 mérést végez az áramfelvételtől, és tárolja ezek átlagát, valamint minimális és maximális értékét. Ha a fent beállított „Samples per second” érték alapján ennél ritkább mintavételt ír elő a felhasználó, akkor a hosszabb időintervallum alatt összegyűlt mérések átlagát, minimumát és maximumát rajzolja ki. Például, ha 1 másodpercenkénti mintavételezést írunk elő, akkor a grafikon tényleg másodpercenként egyszer fog frissülni, mégpedig kb. 10 mérés eredményével (átlag, minimum, maximum).

4.5 Konfiguráció letöltés, eszközök programozása

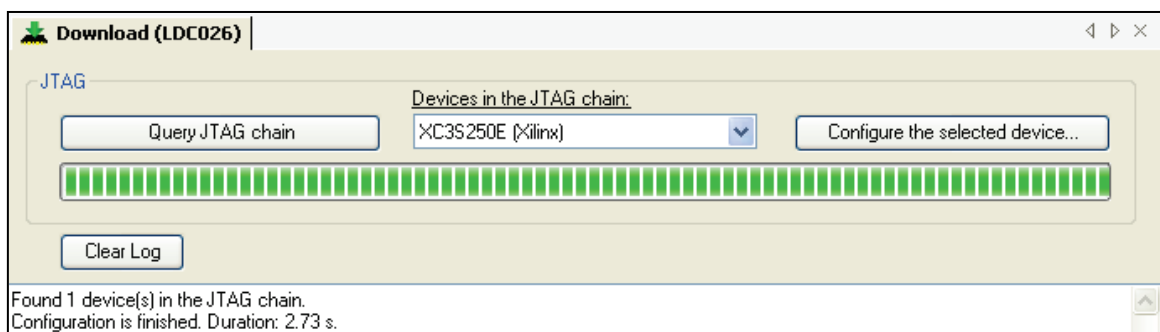
A LOGSYS grafikus felhasználói interfész konfigurációs alrendszer tervezésénél elsődleges célunk volt a lehető legtöbb gyártó legkülönbözőbb eszközeinek támogatása. A különféle gyártóspecifikus letöltési protokollok támogatása nem volt vállalható, ez a rendszer komplexitását jelentősen növelte volna. Azonban létezik egy általánosan használt, egyszerű és nyílt protokoll az eszközök konfigurálására, ami az IEEE 1149.1 JTAG Boundary Scan szabványán alapul, annak fizikai interfészét használja. Ez a JTAG néven emlegetett protokoll minden a gyakorlatban elterjedten használt FPGA eszköznél rendelkezésre áll, és a nagyobb lábszámú mikrovezérlők is (esetleg korlátozottan, letilthatóan) de biztosítják a használatát. A LOGSYS grafikus felhasználói interfész konfigurációs alrendszer ezt támogatja, felkészülve ezen belül a különböző eszközök változatosságára.

A rendszer külső FPGA tervezői környezetek (pl. Xilinx ISE) által generált programozási fájlok lejátszására képes. Ezen fájlokat a szabványos SVF formátumban várja, amely a JTAG interfész primitív műveleteinek sorozatát írja le. Ezen fájlok „lejátszására” alkalmas az alapértelmezett működése esetén a program. Nagy előnye a többi letöltést támogató környezethez képest a több eszközből álló JTAG láncok rugalmas kezelése. A program képes felderíteni a láncot, meghatározni a láncban lévő eszközök számát, és azonosítani a benne lévő eszközöket (amennyiben azok azonosító paraméterei szerepelnek az eszközlíró adatbázisban), majd ezek

alapján automatikusan beállítani a megfelelő regiszterek hosszát úgy, bevezető és kivezető bitsorozatok méretét, hogy az eredetileg egyetlen eszközből álló láncra generált SVF fájlok minden módosítás nélkül letölthetők legyenek tetszőleges összetételű lánc tetszőleges eszközére. Ez a funkció olyan előny, ami nagyon kevés gyártó programozói alkalmazásában található meg. Az idegen eszközök támogatása, kezelése, ha lehetséges is, többnyire manuális beállításokat igényel.

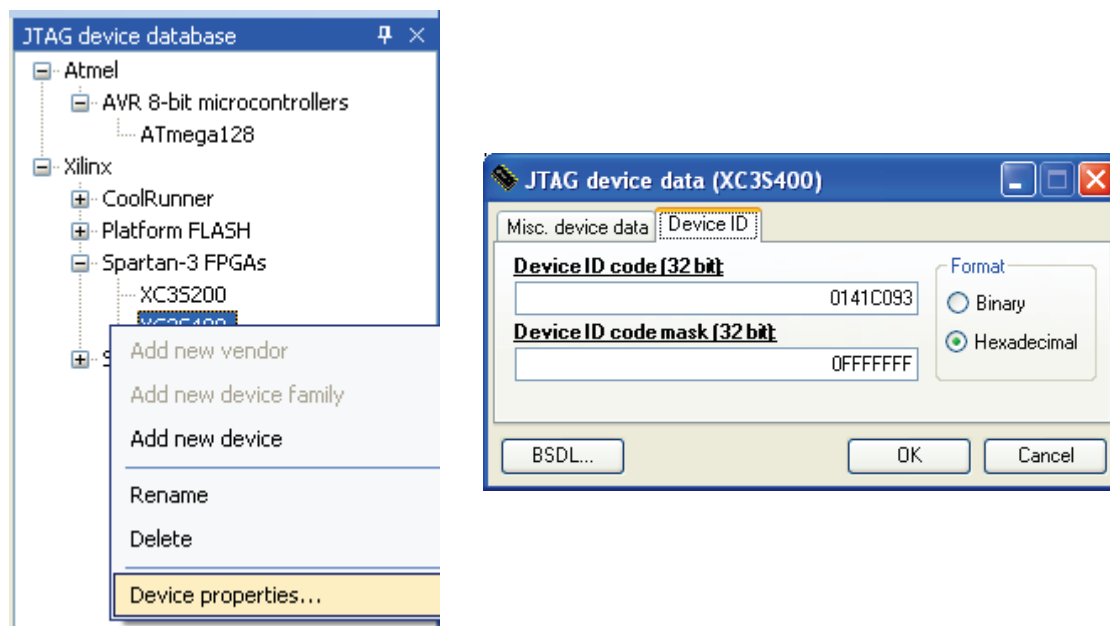
A LOGSYS grafikus felhasználói interfész konfigurációs alrendszer ezen kívül képes a Xilinx ISE iMPACT nevű segédprogramja segítségével SVF fájlok előállítására a letölthető bináris konfigurációs fájlok (FPGA esetén BIT vagy CPLD esetén JED) alapján. Ha a Xilinx ISE megfelelően telepítve van a PC-n, a program a BIT vagy JED fájlok kiválasztása esetén automatikusan meghívja a segédprogramot, majd a háttérben generált SVF fájlt tölti le. Ebben az esetben is ugyanolyan rugalmasan tudja kezelni a különböző JTAG láncokat, mint az SVF fájlok esetén. A letöltés időigénye ilyenkor nagyobb, de közepes méretű eszközök esetén teljes mértékben elfogadható.

A letöltés funkció ablaka a 4-4. ábrán látható. A kezelői felület egyszerű, egy eszköz 3 lépésben felkonfigurálható. A három fő vezérlő elem balról jobbra használva gyakorlatilag megoldottuk a feladatot. Elsőként fel kell deríteni a JTAG láncot, a program innentől kezdve tisztában van annak összetételével, így bármely eszközt képes kezelni. Ezután ki kell választani a felkonfigurálandó/programozandó eszközt a legördülő listából. Utolsó lépésként ki kell választani a letöltendő fájlt. A program ezután elvégzi a konfigurálást, és az esetleges hibaüzeneteket a képernyőre naplózza. A folyamat bármelyik lépésétől tetszőlegesen ismételhető.



4-4. ábra: A letöltés funkció ablaka.

Ahhoz, hogy a program a több eszközt tartalmazó láncokat is sikeresen kezelni tudja, ismernie kell az azt alkotó eszközök JTAG paramétereit. E célrt szolgálja az eszköz adatbázis, amely - mivel nem tartozik egyetlen kábel példányhoz sem - a fejlesztő környezet jobb felső sarkában, egy külön (elrejtett) ablakban kapott helyet. Itt gyártó és eszközcsalád alapján csoportosítva találhatók az ismert eszközök. Egy eszközre jobb gombbal kattintva megadhatók annak paramétereit a 4-5. ábrán látható ablakban. Az FPGA gyártó cégek elérhetővé teszik az úgynevezett BSDL leírófájlt, ami a szerelt nyomtatott áramköri lapok automatikus teszteléséhez szükséges. A program képes ezekből a BSDL fájlokból JTAG TAP vezérlő adatait importálni. Amennyiben ez nem áll rendelkezésre, akkor a parancsregiszter szélessége és az eszközazonosító közvetlenül, manuálisan is megadható.

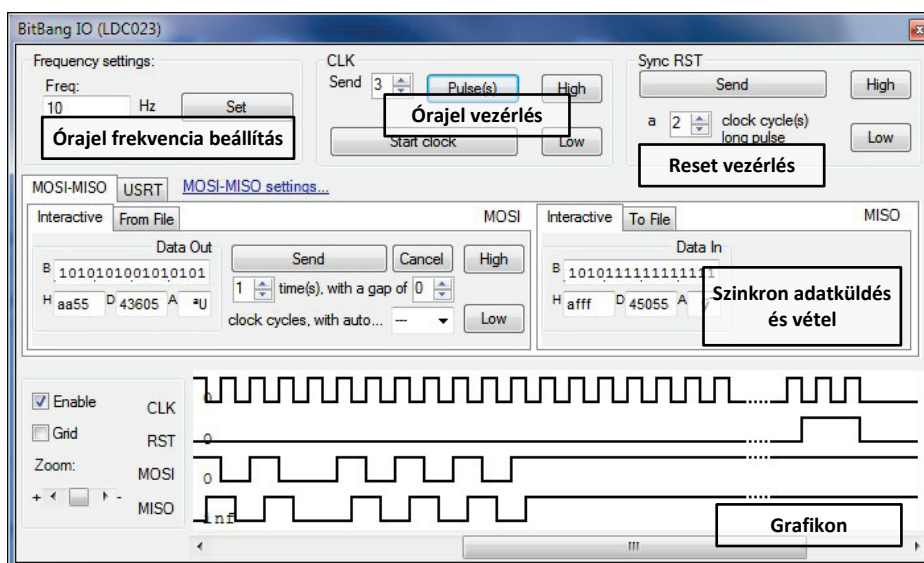


4-5. ábra: Az eszköz adatbázis módosítása.

4.6 BitBang I/O adatkommunikáció

A rendszer BitBang I/O funkciója a LOGSYS port kommunikációs vonalainak rugalmas vezérlését teszi lehetővé, melynek segítségével tetszőleges szinkron kommunikációs protokoll megvalósítható „kézi” módszerrel. Ennek nagy jelentősége a soros kommunikációhoz minimális támogatást nyújtó egyszerű célrendszerek esetén van, beleértve azt az esetet, amikor maga a kommunikációs funkció áll fejlesztés alatt. A tetszőlegesen alacsony frekvenciával, interaktívan vezérelhető vonalak és a látványos kezelői felület ideálissá teszik ezt a szolgáltatást a célrendszerek működésének demonstrálására.

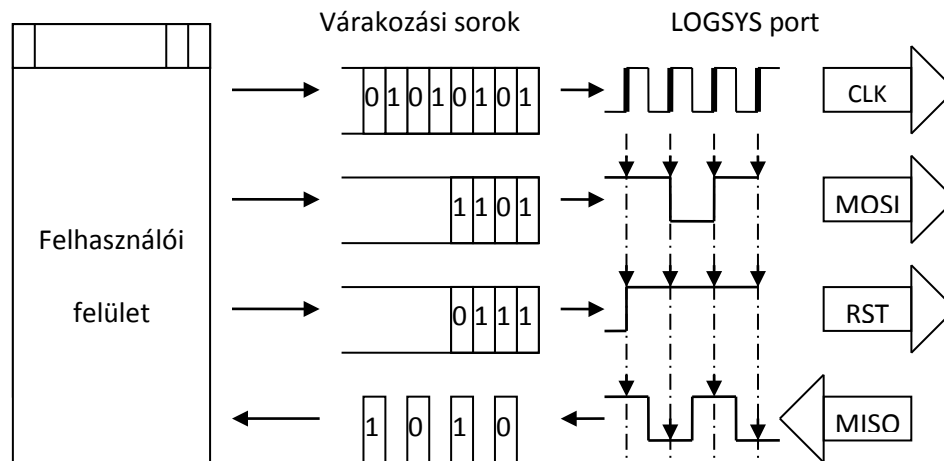
Az ablak képe a 4-6. ábrán látható. Az első ránézésre összetett panel három részre osztható. A felső harmadában a vezérlő jelek (órjel és reset) kezelőszervei találhatók. Középen a szinkron soros be- és kivetel vezérlői kaptak helyet, míg az ablak alsó harmadában a kommunikáció időbeli lefolyását követhetjük nyomon a LOGSYS port adatkommunikációs vonalait ábrázoló grafikonon.



4-6. ábra: A BitBang I/O ablak.

4.6.1 A BitBang I/O funkció modellje

A kezelői felület megismerése előtt szükséges a funkció működésének modelljét megismerni. Ebben a módban a LOGSYS port vonalain szinkron soros adatátvitel valósítható meg egy-egy dedikált be- és kimeneti vonalon, továbbá lehetőség van a reset vonal önálló vezérlésére is. A kommunikáció szinkron, mivel az órajel vonal adott (választhatóan fel- vagy lefutó) élére történik a vonalak vezérlése. A 4-7. ábra szemlélteti a funkció működését.



4-7. ábra: A BitBang I/O funkció modellje.

A koncepció egyszerű: a felhasználói felület kezelőszervei valamilyen séma szerint kiírandó biteket állítanak elő (például egy órajel impulzus egy 1-est és egy 0-át jelent a CLK vonalon, amennyiben annak korábbi állapota 0 volt). Ezek a bitek egy FIFO várakozási sorba kerülnek. A rendszer a beállított órajel frekvenciájának duplájával mintavételezi az órajel sorát, és ha az nem üres, akkor a soron következő bitet kiveszi belőle és az órajel vonalra írja. Ezáltal él keletkezhet az órajel vonalon. Amennyiben így történik, a rendszer megvizsgálja minden olyan kimeneti vonal várakozási sorát, amely az órajel adott (tehát lefutó vagy felfutó) éléhez van szinkronizálva. Ha egy ilyen vonal sora nem üres, akkor a soron következő bitet kiveszi belőle és az adott vonalra írja. Amennyiben a bemeneti vonal mintavételezése ugyanezen élhez van előírva, akkor az ezután történik meg. Ha az ellentétes élhez írjuk elő a mintavételt, akkor a MISO vonal esetlegesen megváltozott állapotát csak a következő élnél látjuk. Fontos különbség, hogy a bejövő bitekhez nem tartozik várakozási sor, azokat egyből megkapja a felhasználó. A fenti ütemezési sémát alkalmazva a LOGSYS port jeleinek változása jól követi egy szinkron rendszer működését. Minden változást az órajel kijelölt típusú változása indít, ezután történik meg a gerjesztő jel új, következő értékének beállítása, majd a válaszjelek mintavételezése.

4.6.2 Az órajel és a reset jel vezérlése

A program tervezése során a várakozási sorok teljeskörű vezérelhetősége helyett az egyszerű kezelhetőséget részesítettük előnyben. Ennek eredményeképpen az órajel vonal általános felhasználását a következő esetekre korlátoztuk: folyamatos órajel, impulzusok kiadása, valamint direkt vezérlés. A reset vonalnál pedig ezek közül az utóbbi kettőre van lehetőség.

Az órajel frekvenciája a Frequency Settings mezőben állítható be 1 Hz és 1000 Hz közötti értékre. Ugyanitt a Set gombra kattintva a beírt érték azonnal érvényre jut, futó órajel vagy akár adatküldés közben is. Az órajel vonal háromféle üzemmódban használható, ezt a Clock mezőben

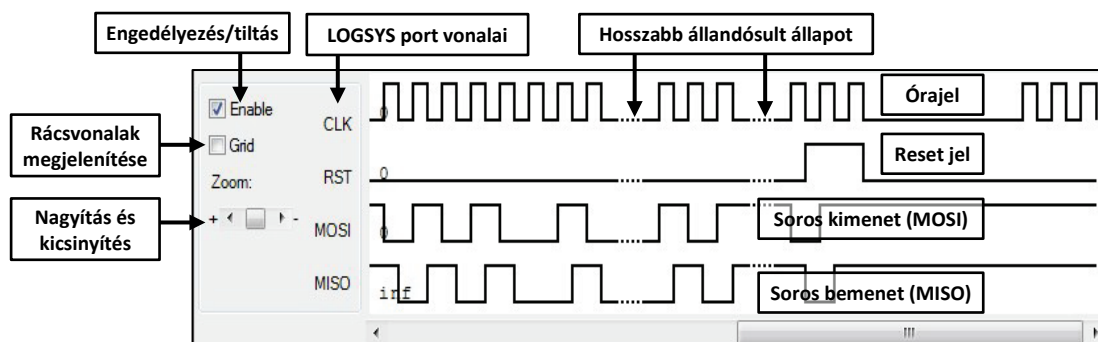
adhatjuk meg. Elindíthatjuk vagy megállíthatjuk az órajelet a Start clock/Stop clock gombbal, kiküldhetünk adott számú impulzust (szintén a beállított frekvenciával) a Send x Pulse(s) gombbal, vagy kézzel is beállíthatjuk az órajel vonal magas (High) vagy alacsony (Low) szintjét. A BitBang funkció modelljének értelmében ezek rendre a következő műveletekkel ekvivalensek: az órajel várakozási sorába végtelen sok váltakozó értékű (0/1 vagy 1/0) bitpár beírásával elindul, míg a sor törlésével megáll a futó órajel, adott számú bitpár beírásával ennek megfelelő számú impulzus keletkezik, míg a sor törlésével majd egyetlen bit beírásával az órajel vonal értéke direkt módon beállítható a kívánt új értékre.

A Sync RST mezőben a reset vonal vezérlését adhatjuk meg, statikus kézi beállítással, vagy impulzusszerű vezérléssel. A reset jelnél megengedett vezérlés némileg eltér az órajelétől: itt mindig adott számú teljes órajel hosszúságú órajel ciklusig tartó impulzus küldésére van lehetőség. Fontos, hogy a reset vonal mindig az órajel felfutó élére változik és teljesíti a szinkron reset kiadásra vonatkozó időzíítési feltételeket, azaz a következő órajel felfutó él előtt elegendő hosszú előkészítési időt és az aktuális órajel felfutó él után elegendő hosszú tartási időt biztosít. Ennek megfelelően a BitBang üzemmódban garantált a rendszer megbízható reset vezérlése, tehát minden állapotváltozó azonos órajel ciklusban veszi fel a reset állapotát és lép ki belőle a reset vonal kikapcsolásakor.

A működési modellből következik, hogy mindkét vezérlő jel „programozott” állapotát (futás vagy impulzus) a „kézi” vezérlés (magas vagy alacsony) azonnal megszünteti, felülírja.

4.6.3 A BitBang I/O hullámforma grafikonja

A BitBang kezelői ablak alsó harmadában lévő hullámforma grafikon a kommunikáció időbeli lefolyásának megjelenítésére szolgál, felülete a 4-8. ábrán látható. A bal oldalán lévő jelölőnégyzetekkel engedélyezhetjük vagy tilthatjuk a működést, valamint a függőleges rácsvonalakat kapcsolhatjuk be és ki. Lehetőség van az ábra nagyítására és kicsinyítésére is (Zoom), továbbá a korábbi hullámformákhoz való visszagörgetésre, bizonyos időtartományban. A grafikon a LOGSYS port négy kommunikációs vonalának állapotát ábrázolja (CLK, RST, MOSI és MISO). Minden vonalhoz tartozó hullámforma elején egy szám jelzi az adott vonal várakozási sorában lévő bitek számát, ahol az „inf” végtelen számú bitet jelöl. Ha hosszabb ideig változatlan a vonalak állapota, az ábra fölöslegesen megtöltése helyett pontozott vonallal jelöli a program a kihagyást. Fontos hangsúlyozni, hogy ennek következtében a grafikon ábrája nem jelent valós időtengelyt, azon aktivitást mindig csak az általunk vezérelt CLK működtető órajel változása okoz. A grafikon megjelenítése 50 Hz órajel felett automatikusan letiltódik, és az órajelet alacsonyabb frekvenciára kapcsolva az aktiváláshoz a grafikont újra engedélyezni kell.



4-8. ábra: A BitBang I/O funkció időtartománybeli grafikonja.

4.6.4 BitBang I/O soros adatátvitel

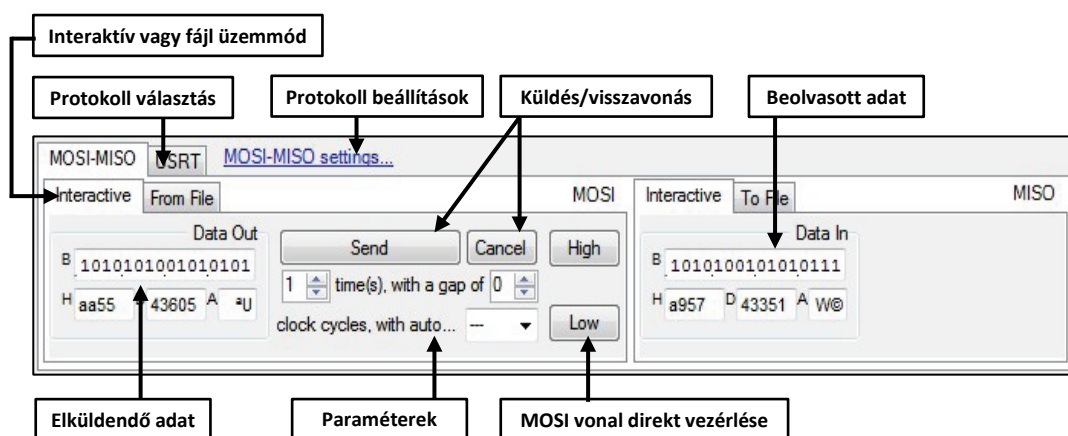
A BitBang I/O panel középső részén található az adatok soros kiírásával és beolvasásával kapcsolatos vezérlők, baloldalon a kiírással, jobb oldalon a beolvasással kapcsolatosak. Ezeknek számos eltérő üzemmódja van, amelyek között fülek segítségével váltogathatunk. Ami azonban minden módban közös elv: a kiírás lényege, hogy a MOSI vonal várakozási sorát valamilyen bitsorozattal megtöltse, míg a beolvasás vezérlője a MISO vonalról származó bitek sorsát határozzák meg. A két alapvető üzemmód a MISO-MOSI és az USRT.

4.6.4.1 MOSI-MISO üzemmód

Ez az üzemmód a „nyers”, valódi BitBang I/O üzemmód: a felhasználó által beküldött (vagy abból előállított) bitsorozat egy az egyben beírja a MOSI vonal várakozási sorába, míg a MISO vonalról beolvasott biteket utófeldolgozás nélkül visszaadja a felhasználónak. Mind a kiírandó adatok forrása, mind pedig a beolvasott adatok célja lehet az interaktív kezelői felület vagy a host PC-n található fájl is, egymástól függetlenül. Vegyük sorra, hogy melyik esetben milyen lehetőségei vannak a felhasználónak.

4.6.4.1.1 Interaktív MOSI-MISO üzemmód

Az interaktív kiírás és beolvasás felületét mutatja a 4-9. ábra. A Data Out mezőbe írhatjuk a kiküldendő adatot, választhatóan bináris, decimális, hexadecimális vagy karakteres (ASCII) formátumban, a bármelyik ablakba írva, a többi megfelelő formátumba konvertálva tükrözi az értéket. Az így beírt értékből képzett bitsorozatot a program a Send gomb megnyomására beírja a kimeneti MOSI vonal sorába. Azt, hogy a megadott adatból pontosan milyen bitsorozatot állít elő a rendszer, a Send gomb alatti vezérlőkkel állíthatjuk be. Az ismétlésszám és az ismétlések közötti szünet egyértelmű, ezen kívül lehetőség van a megadott biteken két kiírás között automatikus transzformációt végezni. Ezeket mindig a Data Out mezőbe beírt adatokon értelmezzük, akárhány bitet adtunk is meg (kivétel: *rnd*). A bitsorrend és a változást ütemező órajel él a MOSI-MISO Settings... előugró ablakban adható meg (lásd később). A transzformációkat a 4-1. táblázat ismerteti.

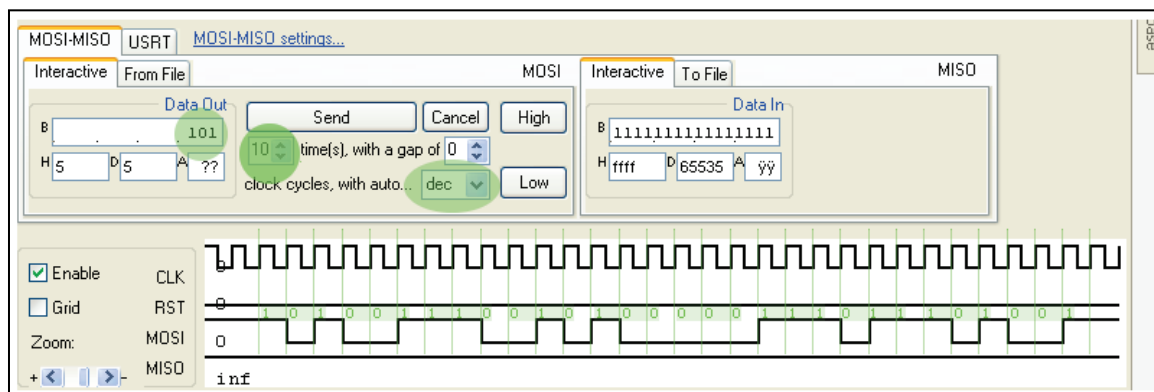


4-9. ábra: A MOSI-MISO üzemmód interaktív felülete.

4-1. táblázat: A MOSI-MISO üzemmód automatikus transzformációi.

Transzformáció	A Data Out mezőbe beírt értéket két ismétlés között...
---	Változatlanul hagyja
inc	Eggyel növeli
dec	Eggyel csökkenti (kettes komplement számábrázolással)
shl	Balra shifteli (nullás bejövő bittel)
shr	Jobbra shifteli (nullás bejövő bittel)
rol	Balra forgatja
ror	Jobbra forgatja
rnd	Balra shifteli, a bemenő bitet egy maximális ciklusú LFSR visszacsatolásai alapján határozza meg. A megadott értéket 16 bitesen értelmezi, nullákkal kitöltve. A 15. (MSB) bitet kezdetben 1-re állítja.

A 4-10. ábra egy példát mutat a transzformációkra, melyben a decimális 5-ös kezdőértéket 10-szer írjuk ki, automatikus csökkenéssel. Az adatokat LSB bitjűkkel kezdve írjuk ki, az órajel felfutó élére. Látható, hogy a program 3 bites számként értelmezi a beírt adatokat, és a 0-t elérve túlcsordul a -1 három bites kettes komplement ábrázolásával. Tehát a kiküldött adatbit sorozat minta az 101,001,110,010,100,000,111,011...



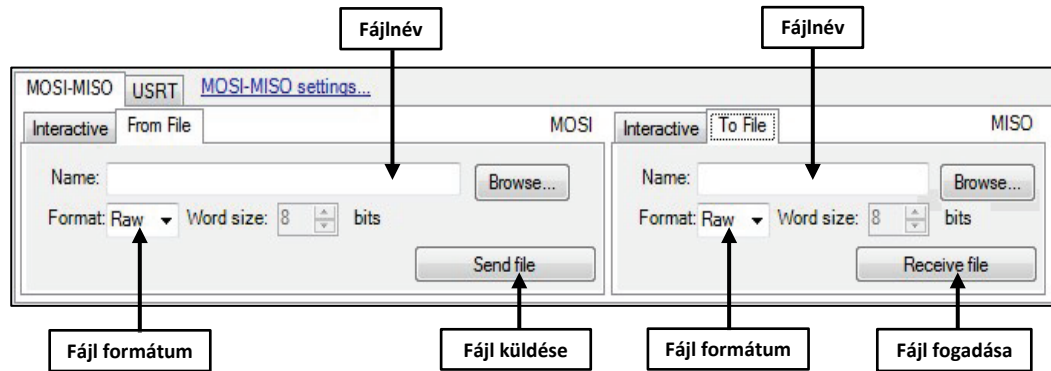
4-10. ábra: Példa a MOSI-MISO automatikus transzformációra.

Az Interactive vezérlési panelen lévő Cancel gomb törli a MOSI vonal várakozási sorát, míg a High és a Low gombok ugyanezt teszik egyetlen 1 vagy 0 bit beírásával kiegészítve.

A beolvasás interaktív esetben egyszerű: a beállított bitsorrend (MSB vagy LSB elől) alapján a beolvasott bitek egy 16 bites FIFO sorba kerülnek, ami a képernyőn is látszik a Data In mezőben. Minden bejövő bitnél a legrégebben bent lévő bit eldobásra kerül. Az éppen aktuálisan bent lévő 16 bitet decimálisan, hexadecimálisan és ASCII karakterekkel is megjeleníti a program.

4.6.4.1.2 BitBang I/O fájl üzemmód

A program képes arra, hogy a kiírandó és a beolvasott adatokat fájlból beolvassa, illetve fájlba kiírja. Ebben a módban a kezelői felület a 4-11. ábrán látható. A lehetőségek egyszerűek: megadható a fájl neve, formátuma (Bin, Hex, Dec, Raw), valamint elindítható és leállítható a küldés, illetve a fogadás.



4-11. ábra: A MOSI-MISO üzemmód fájl küldés/fogadás felülete.

A fájlformátumok azonban távolról sem magától értetődőek. A program tervezése során igyekeztünk a lehető legnagyobb rugalmassággal, a gyakorlati igényeket kielégítve meghatározni őket. Most ezeket ismertetjük.

Speciális fájlformátum a Raw fájl, ezt a program bájtanként olvassa be, és minden beolvasott bájtot külön szóként kezelve kiír. Ebből következik, hogy itt a szóhossz nem állítható. Fájlba olvasáskor a bejövő adatbitek nyolcasával kiírja a fájlba, mindenféle utófeldolgozás nélkül.

A többi fájlformátum meghatározásánál abból indultunk ki, hogy ezeket a fájlokat nagyrészt úgy használjuk, mintha az interaktív szövegmezőbe egymás után beírnánk és kiküldenénk több értéket. Ekkor a felhasználó számára az input fájl elkészítésénél kényelmesebb az interaktív szövegdobozhoz hasonló, 4 – 16 bites egységek (szavak) sorozatában gondolkodni, mint egy hosszú bitfolyamban.

Minden fájl ezért szavakból áll, ezek az adott számrendszerben legális számjegyekből álló számok, szóközzel, tabulátorral vagy enterrel (tetszőleges kombinációban) elválasztva. A program beolvas egy szót, és ennek annyi bitjét veszi figyelembe (a legkisebb helyiértéktől számítva), ahányat a Word size szóhossz mezőben megadtunk. Ha rövidebb a beolvasott szó, akkor kiegészíti MSB felé nullákkal. Az így kapott x darab bitet küldi ki azután, MSB vagy LSB bitjével elől, a beállítástól függően.

Fájlba való írás esetén a program folyamatosan gyűjti a biteket, majd ha az adott szóhossz (Word Size) számú bit összegyűlt, a beállított számrendszerben kiírja őket a fájlba, szóközzel elválasztva az előző szótól, és kb. 80 karakterenként új sort kezd. Amint a Stop receiving file gombot megnyomjuk, a program kiírja a maradék biteket is, még ha azok száma kevesebb is, mint a szóhossz.

A fentiekből adódik, hogy a beállított szóhossz nagymértékben meghatározza a beolvasott fájl értelmezését. A 4-2. táblázat néhány szélsőséges esetet mutat be a program működésének jobb megértése érdekében. Az érvényes beállítások: hexadecimális formátum, 11 bites szóhossz.

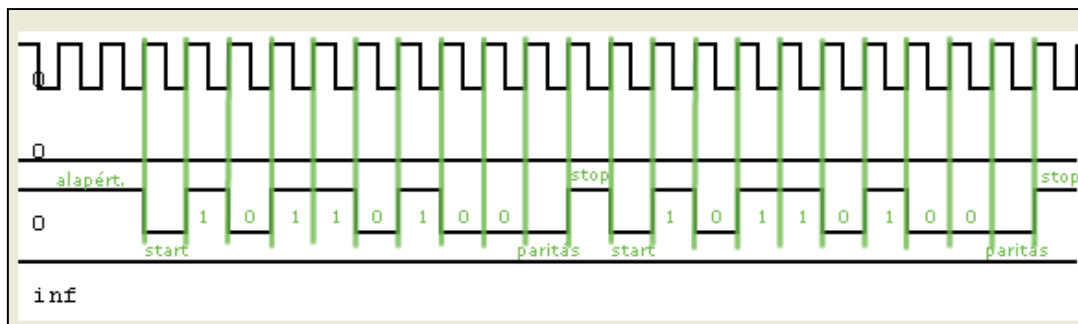
4-2. táblázat: Példák a fájlból olvasott adatok értelmezésére.

Fájlból olvasott szó (hexa, 11 bit)	Kírandó adat
„fff”	11111111111
„800”	00000000000
„7”	00000000111
„aaaabbbbccccddd0000”	00000000000

4.6.4.2 BitBang I/O USRT üzemmód

Ebben az üzemmódban a rendszer oly módon alakítja át a felhasználó által beírt kiküldendő adatot, hogy az az USRT protokoll szabályai szerinti kódolással jelenjen meg a LOGSYS porton. Ez a protokoll (pontosabban a rendszer által megvalósított változata) alapvetően a hagyományos keretezett karakteres adatátviteli protokoll, csak az adatátvitel az órajellel szinkron működésű. A protokoll részletes ismertetése a következő:

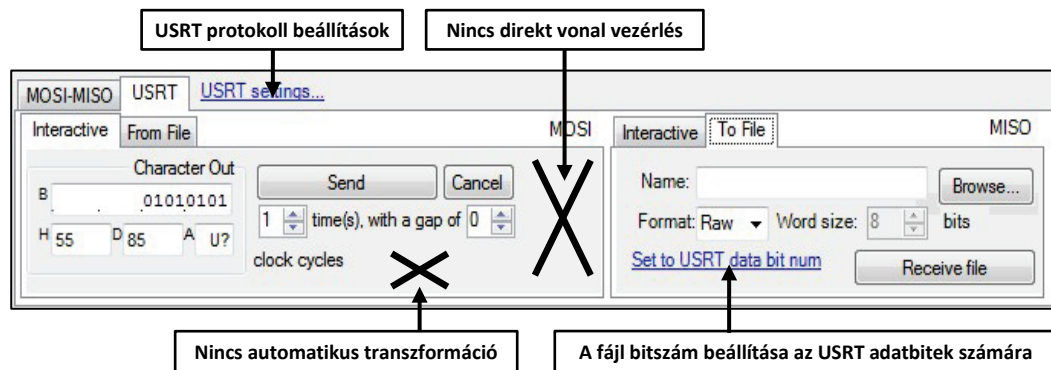
Az átvitel alapegysége a karakter, melynek hossza konfigurációtól függően 4 és 16 bit közé eshet. A dedikált kimeneti vonal alapértelmezés szerint magas szinten áll (ez az üzemmód kiválasztásakor automatikusan beállítódik). Minden egyes karakter átvitelének kezdetét egy-egy órajel ciklus hosszú alacsony értékű START bit jelzi. Ezután következnek a karakter bitjei, majd egy opcionális PTY paritás bit, amely beállítástól függően páros vagy páratlan paritására egészíti ki a karaktert. Az átvitelt egy magas értékű STOP bit zárja. Az adatátvitelhez az adatmezőben természetesen csak az adat értékét kell megadni, a START, PTY és STOP biteket a rendszer automatikusan fűzi a kimeneti bitsorozatba. Természetesen egy-egy karakter átviteléhez ebben az üzemmódban az adatméretnél 3-mal több órajel szükséges (ennek lépésenkénti üzemmódnál lehet jelentősége). Az üzemmód jellemző hullámformáját a 4-12. ábrán láthatjuk. Beállítások: 8 bites karakterhossz, páros paritás. A kiküldött érték: 10110100 kétszer egymás után, LSB-vel kezdődő átvitel, az adatvonal az órajel felfutó élére változik.



4-12. ábra: Az USRT üzemmód jellemző hullámformája.

Az USRT mód interaktív és fájl-orientált kezelői felülete a 4-13. ábrán látható. A kötöttebb protokollból adódóan itt korlátozottabbak a felhasználó lehetőségei. Nem lehet automatikus transzformációkat végezni a beírt adaton, illetve nincs lehetőség a kimeneti vonal direkt vezérlésére sem. Ezekről eltekintve a felület megegyezik a MOSI-MISO esettel.

Interaktív beolvasásnál a program a protokoll jellegzetességéhez illeszkedve karakterenként frissíti a szövegmezőt. Ha a beolvasott karakter végén a stopbit nem magas értékű, az kerethibát jelent. Ha beállítottuk a keretek ellenőrzését (lásd később), akkor a program a beolvasott adatot érvényteleníti, és a kerethibát jelzi. Ha beállítottunk paritás bitet, és a karakter paritásbitje nem ennek megfelelő értékű, annak szintén ez a következménye.

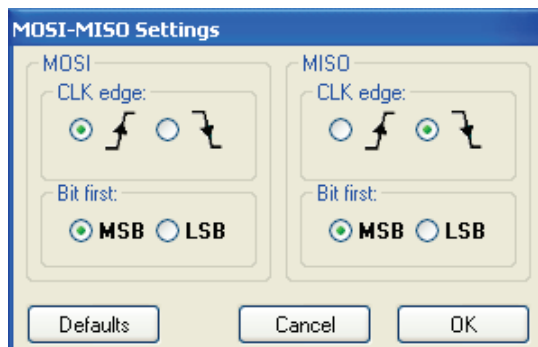


4-13. ábra: Az USRT üzemmód felülete.

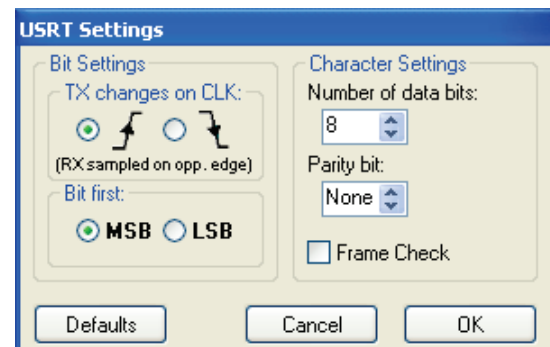
Fájlból olvasásnál a működés megegyezik a MOSI-MISO üzemmóddal. Fájlba íráskor annyiban tér el tőle, hogy a keret- vagy paritáshiba figyelése esetén az ilyen karakterekből keletkezett szavak után egy felkiáltójelet ír a fájlba. Fontos, hogy ez a funkció csak akkor ad helyes eredményt, ha a fájl előírt szóhossza megegyezik az USRT protokoll karakterhosszával, egyéb esetben a felkiáltójelek helye megjósolhatatlan lesz (de adatvesztés nem történik). A két beállítás szinkronba hozását a kezelői felületen egy link segíti: erre kattintva a fájl szóhossza a protokoll karakterhosszához igazodik. A Raw formátum esetén a program az átviteli hibákat figyelmen kívül hagyja.

4.6.5 BitBang I/O adatátviteli beállítási lehetőségek

Mind a MOSI-MISO, mind az USRT protokollok rugalmas beállítási lehetőségekkel rendelkeznek. Ezeket közvetlenül a módválasztás melletti linkekre kattintva felugró ablakban változtathatjuk meg. A MOSI-MISO beállítások ablaka a 4-14. ábrán, míg az USRT beállításoké a 4-15. ábrán látható.



4-14. ábra: MOSI-MISO beállítások.



4-15. ábra: USRT beállítások.

A MOSI-MISO üzemmód egyszerű: itt mindössze az aktív órajel él és a kiírás/beolvasás bitsorrendje állítható, a bemeneti és a kimeneti vonalon egymástól függetlenül.

USRT módban a két vonal tulajdonságai csak egyszerre állíthatók, mivel a protokoll mindkét résztvevő szemszögéből nézve azonosan működik. A kimeneti vonal változása és a bemeneti vonal mintavételezése ellenkező órajel élnél történik. Ezen kívül beállítható a karakterhossz, a paritásbit küldése és ellenőrzése, valamint a kerethiba figyelése MISO vonalon, az olvasott karakterekre.

4.7 Soros kommunikációs lehetőségek

A soros kommunikációs lehetőségek különböző kommunikációs képességű célhardverek, beágyazott rendszerek tesztelését támogatják. Amennyiben a tesztelt rendszer támogatja valamilyen formában a karakteres kommunikációt, úgy közvetlenül használható az UART/USRT konzol alkalmazása a Terminál földre kattintva.

Sok esetben azonban a tesztelendő hardver rendszer legközvetlenebb interfészei a statikus be- és kimeneti pontjai, vagy esetleg egy adatfolyam jellegű be- és kimenet. Ebben az esetben jól használható a Registers/FIFO fül aktiválásával megjeleníthető regiszter szintű adatátvitelt biztosító tesztfelület, ahol 8 bites regisztereken és FIFO csatornákon keresztül tudunk adatokat és paramétereket küldeni a célhardvernek, illetve fogadni a célhardvertől. Hasonlóan használható a RAM füllel aktiválható 2x64 bájtos 2-portos kommunikációs memória elérését biztosító felület is. Ebben az esetben a tesztelt rendszer bemenő adatait egy előzőleg feltöltött 64 bájtos memóriából olvashatja, az eredményeket pedig egy másik 64 bájtos memóriaterületre írhatja.

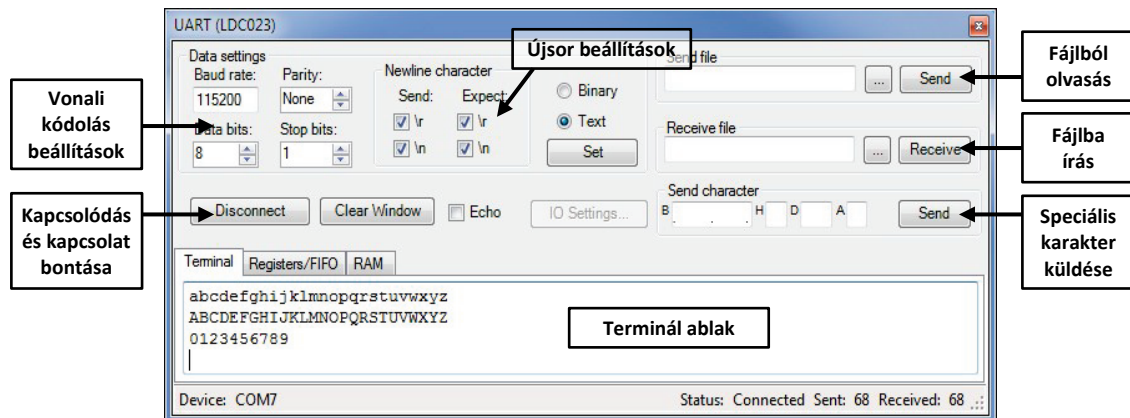
A bemutatott tesztkörnyezet interfészek kifejezetten FPGA alapú célhardverek teszteléséhez készültek, és a fenti szolgáltatások használatához a fejlesztendő hardver rendszerbe a megfelelő kommunikációs interfész blokkot is be kell építeni. Mintaalkalmazások a LOGSYS alkalmazói honlapon találhatók.

4.7.1 A közvetlen UART és USRT adatátvitel

A LOGSYS grafikus felhasználói interfész a LOGSYS port kommunikációs vonalain keresztül képes a hagyományos soros adatátvitel megvalósítására. Ez a jól ismert UART (Universal Asynchronous Receiver/Transmitter) interfészen, illetve annak szinkron változatán (Universal Synchronous Receiver/Transmitter) keresztül történhet. Ez utóbbi megoldásnál a dedikált adó (TX) és vevő (RX) adatvonalak mellett egy órajel vonal is szerepet kap, ennek élével szinkronban történik az adatátvitel. Az órajelet mindig a master eszköz biztosítja, jelen esetben a LOGSYS fejlesztő kábel CLK kimenete. Az órajel az USRT soros kommunikációs üzemmódban folyamatosan jár, frekvenciája a beállított adatátviteli sebességgel egyezik meg (4800 – 115200 Hz).

A LOGSYS fejlesztő kábel által megvalósított virtuális soros port (COM port) eszközt a host PC-n a Windows operációs rendszer tetszőleges COM port kezelő alkalmazása használhatja (pl. Hyperterminal), de a kényelmes fejlesztéshez a LOGSYS grafikus felhasználói interfész egy teljes értékű konzol felületet is ad a beágyazott rendszer vezérléséhez.

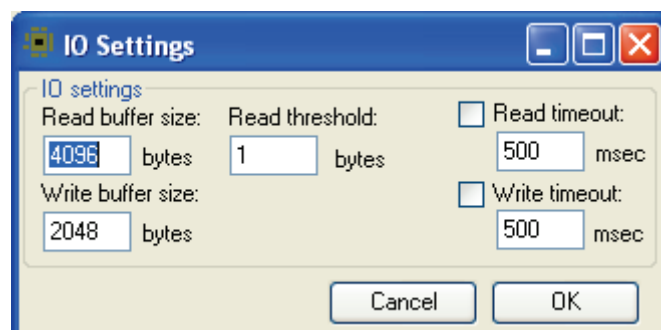
A soros UART/USRT kommunikáció funkciót a 4-16. ábrán látható ablakon keresztül használhatjuk. A felület az interaktív használat céljára egy terminálablakot tartalmaz, továbbá lehetőség van fájlok elküldésére, illetve a vett adatok fájlba írására is. Amennyiben a billentyűzetről nem bevihető karaktert szeretnénk küldeni, decimális, bináris vagy hexadecimális számrendszerben is megadhatjuk a kódját.



4-16. ábra: Az UART és az USRT funkciók ablaka.

A soros adatátviteli paraméterek beállíthatósága a szokásos opciókat tartalmazza. Az átvitel alapvető tulajdonságai, tehát az átviteli sebesség, az adat- és stopbitek száma, valamint a paritásellenőrzés módszere menet közben is állítható. A Data Settings mezőben kiválasztható a sebesség (4800 – 115200 bit/s), az adatbitek száma (5 – 8, illetve speciális vezérlési feladatokra 9 bit), az esetleges paritásbit (None, Odd, Even), és a STOP bitek száma (1 vagy 2). Ezen kívül a terminál program rugalmas beállítási lehetőségeket biztosít az újsorok rendszerenként eltérő kezelésével kapcsolatban. Megadhatjuk, hogy melyik karakter(ek) vétele esetén szűrjön be a program (DOS/Windows szabály szerinti) sortörést a terminál ablakba vagy a vett adatokat tartalmazó fájlba, illetve azt is, hogy a kiküldendő adatokban lévő sortörés esetén milyen karakter(ek)et küldjön. Mindennek természetesen csak szöveges módban van jelentősége, a módot binárisra állítva a program nem alakít a kiküldendő adatokon, illetve a vett bájtokat is változtatás nélkül továbbítja a felhasználó felé.

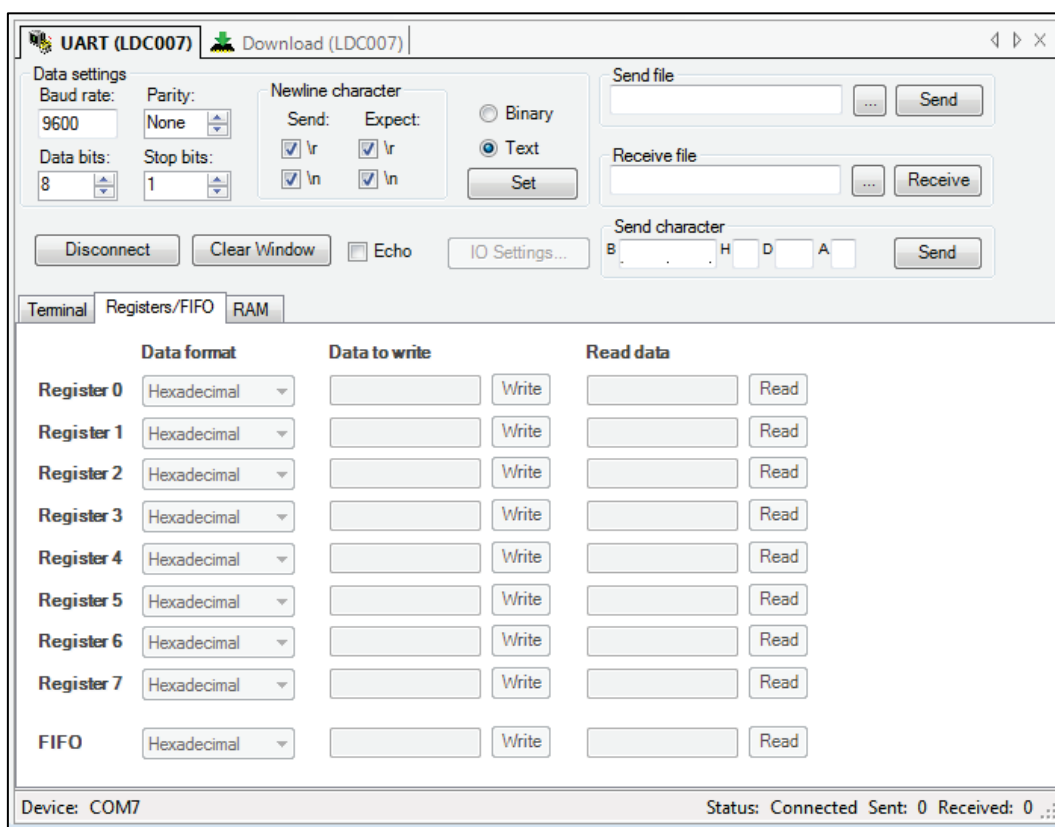
A kommunikáció megkezdése előtt kapcsolódni kell a fejlesztő kábel által megvalósított virtuális soros port (COM port) eszközhöz, a Connect gombra kattintva. Ezzel a gomb felirata Disconnect-re változik, a használat végeztével erre kattintva zárhatjuk le az eszközt. Az átvitel alacsony szintű beállításai csak szétkapcsolt módban változtathatók meg az IO Settings gombra kattintva. Az ekkor megjelenő ablak a 4-17. ábrán látható. Itt beállítható az írási és az olvasási puffer mérete, valamint az, hogy hány bájtos telítettségnél adja vissza a kábel meghajtója a beolvasott adatokat. Ezt interaktív alkalmazásoknál célszerű 1-re állítani. A *timeout* értékek a program és a fejlesztői kábel közötti kapcsolatra vonatkoznak, és általában nincs jelentőségük.



4-17. ábra: UART/USRT I/O beállítások.

4.7.2 A regiszter szintű tesztinterfész

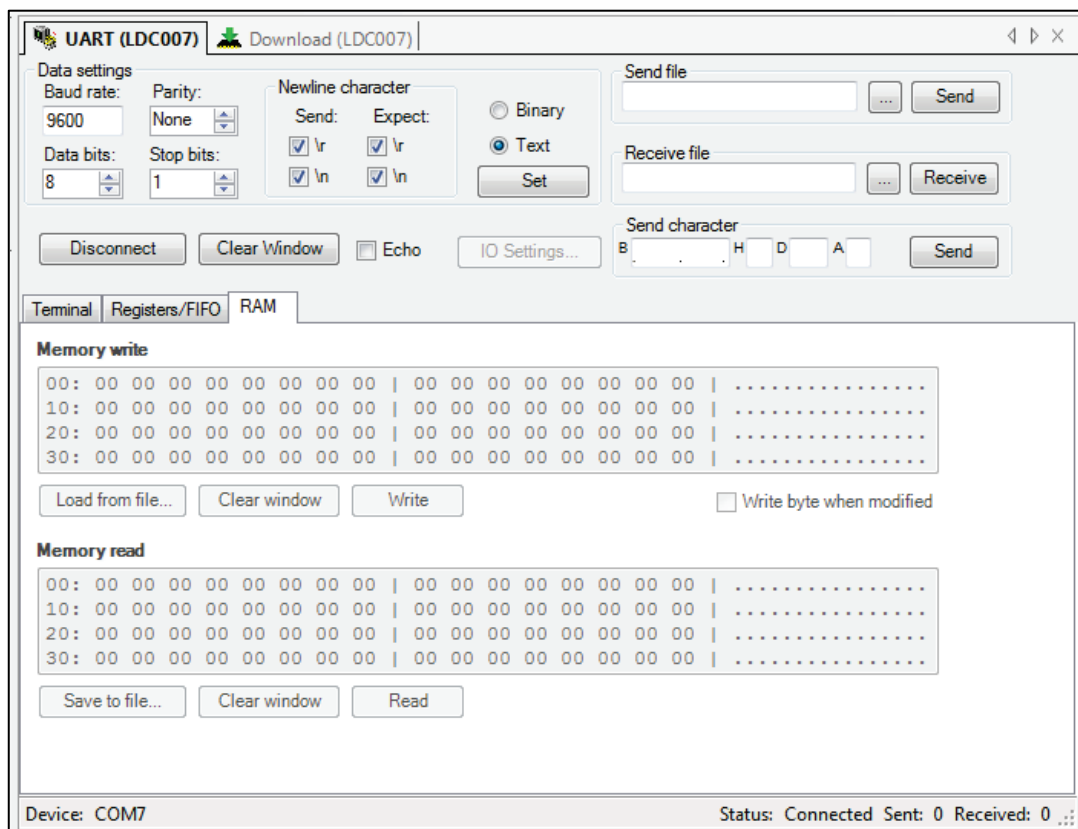
Hardver rendszerek tesztelése jellemzően a bemenetek különböző értékű gerjesztéseivel és a kimeneti válaszok megfigyelésével történhet. A legegyszerűbb esetekben erre a célra jól használhatók a különböző LOGSYS FPGA kártyákon található kapcsolók, nyomógombok, LED-ek és numerikus kijelzők, de ezek korlátozott száma esetleg összetettebb áramkörök tesztelését nehezéssé teszi. A Registers/FIFO fülrel aktiválható ablakban egy olyan tesztinterfész használata válik lehetővé, amely maximum 64 bit bemeneti jel vezérlését, maximum 64 bit kimeneti jel megfigyelését és ezen felül egy 8 bites FIFO jellegű kimeneti és bemeneti csatorna használatát biztosítja. A vezérlőjelek 8 bites regiszterekbe csoportosítva állnak rendelkezésre, az egyedi bitek használata a felhasználó által tetszőlegesen megválasztható. Alapvetően ezek a jelek a fejlesztett rendszer szempontjából, mint aszinkron, kvázi statikus bemeneti jelek használhatók, melyeket az egyedi regiszter írás (Write) parancsok kiadása érvényesít, és a kimenetek megfigyelése is csak a felhasználó által kiadott olvasás (Read) parancs hatására frissül az ablakban. Az adatbeállítás és működtetés szinkronizációja egy regiszter, pl. az Register0 parancsregiszter ill. státuszregiszter funkcióban történő használatával lehetséges, egyes bitekhez a RUN, STEP, ill. READY/BUSY jelentést hozzárendelve. Ez a tesztelési mód ennek megfelelően egy magasabb szintű vezérlő/működtető rendszer kialakítását igényli. A tesztkörnyezet használatát egy külön alkalmazói dokumentum ismerteti, letölthető a <http://logsys.mit.bme.hu> oldalról.



4-18. ábra A Regiszters/FIFO tesztinterfész kezelői felülete

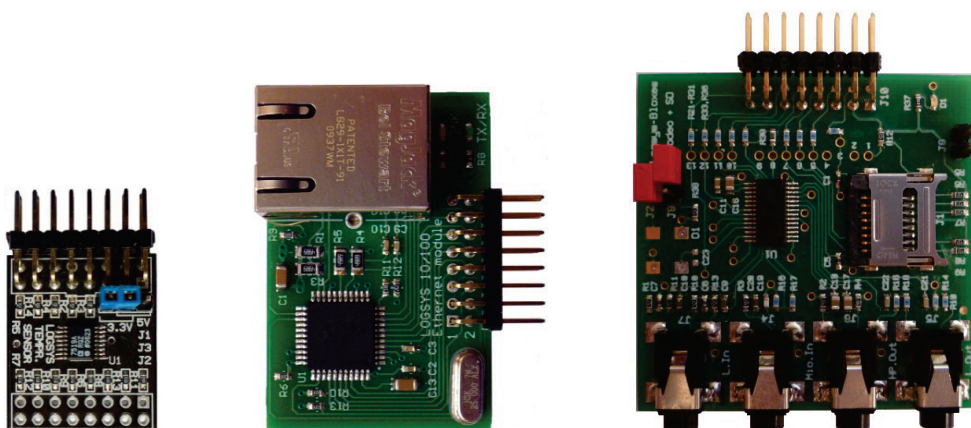
4.7.3 Memória szintű tesztinterfész

Bizonyos hardver egységek adatinterfésze jellemzően valamilyen memória egység, amelybe betölthető a feldolgozandó adattömb és az eredmény is ugyanilyen módon, a memóriába visszatöltve áll rendelkezésre. A RAM füllel megnyitható memória interfész egy ilyen, memória referenciás adatkommunikációt biztosító felületet ad, elsősorban keresési, rendezési feladatok megvalósításának támogatására, algoritmusok tesztelésére. A 2 db független 64 bájtos RAM memória a fejlesztőkörnyezet felől és a tesztelt hardver felől is véletlen hozzáféréssel írható olvasható, ellentétes kiosztással, azaz a kimeneti memóriaterület az ablakban írható, a teszthardver olvashatja, míg a bemeneti memóriaterületet a célhardver írja, a grafikus felület olvassa és megjeleníti. A használat során lehetséges a kimeneti bufferben okozott változások azonnali kiírása (Write byte, when modified), vagy csak a teljes tartomány beállítását követően kiírás. Ez elsősorban fájlból történő műveletvégzés esetén lehet szükséges. A bemeneti buffer frissítése a Read beolvasás parancs hatására történik, ez tetszőleges számban kiadható. A memóriaterület tartalma fájlba elmenthető. A memória tartalom a szokásos hexadecimális formátumban jelenítődik meg, bájtos méretben, továbbá látható a bájtok által reprezentált érték ASCII kódja is.



4-19. ábra A RAM kommunikációs felület kezelői ablaka

5 LOGSYS modulkészlet

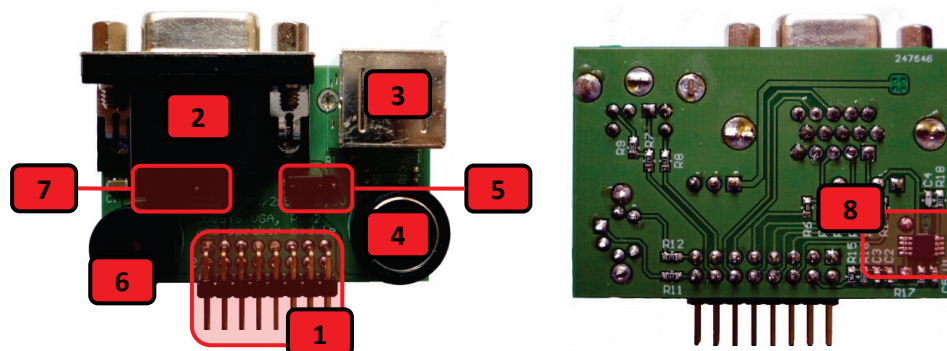


5.1 LOGSYS VGA és PS/2 modul

A modul lehetővé teszi VGA monitor, PS/2 perifériák valamint hangszóró illesztését a LOGSYS FPGA kártyákhoz. A modul felépítését az 5-1. ábra szemlélteti. A bővíthető csatlakozó lábkiosztását az 5-1. táblázat mutatja.

A modul főbb jellemzői:

- 64 szín megjelenítést lehetővé tevő VGA interfész
- 2 darab PS/2 csatlakozó 5V vagy 3,3V választható tápfeszültséggel
- PWM audio bemenet, TS4890IST típusú 1 wattos hangerősítő áramkör
- Beépített hangszóró
- Külső hangszóró csatlakoztatási lehetőség



5-1. ábra: A LOGSYS VGA, PS/2 és hangszóró modul felépítése.

A modul felépítése:

1. LOGSYS bővíthető csatlakozó
2. VGA csatlakozó
3. Első PS/2 csatlakozó
4. Második PS/2 csatlakozó
5. A PS/2 csatlakozók tápfeszültségét kiválasztó jumper (5V vagy 3,3V)
6. Beépített hangszóró
7. Külső hangszóró csatlakozó
8. TS4890IST típusú 1 wattos hangerősítő IC

5-1. táblázat: A bővíthető csatlakozó lábkiosztása.

Láb	Típus ¹	Funkció
1	PWR	GND
2	PWR	5V tápfeszültség
3	PWR	3,3V tápfeszültség
4	I	PWM audio jel
5	I	VGA piros jel (1. bit)
6	I	VGA piros jel (0. bit)
7	I	VGA zöld jel (1. bit)
8	I	VGA zöld jel (0. bit)

Láb	Típus ¹	Funkció
9	I	VGA kék jel (1. bit)
10	I	VGA kék jel (0. bit)
11	I	VGA vertikális szinkronjel
12	I	VGA horizontális szinkronjel
13	OC	PS/2 órajel (1. csatlakozó)
14	OC	PS/2 adat (1. csatlakozó)
15	OC	PS/2 órajel (2. csatlakozó)
16	OC	PS/2 adat (2. csatlakozó)

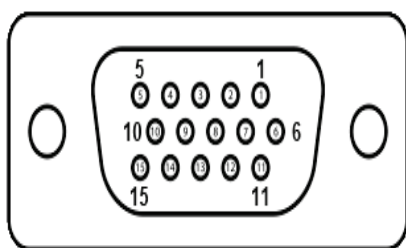
¹ PWR: tápellátás, I: bemenet, O: kimenet, OC: nyitott kollektoros meghajtás

Például: I (bemenet) esetén az FPGA számára az adott láb kimenet, a modul számára bemenet.

5.1.1 VGA interfész

A modulon található 15 pólusú HD-DSUB anya csatlakozó lehetőséget biztosít CRT vagy LCD monitorok csatlakoztatására standard VGA monitor kábelt használva. A VGA csatlakozó láb kiosztását a 5-2. táblázat mutatja, a bővítőcsatlakozóra csak a félkövérrel kiemelt jelek vannak kivezetve. A piros, zöld és kék színjelekhez két-két bit tartozik, így összesen 64 szín jeleníthető meg.

5-2. táblázat: A VGA csatlakozó láb kiosztása.

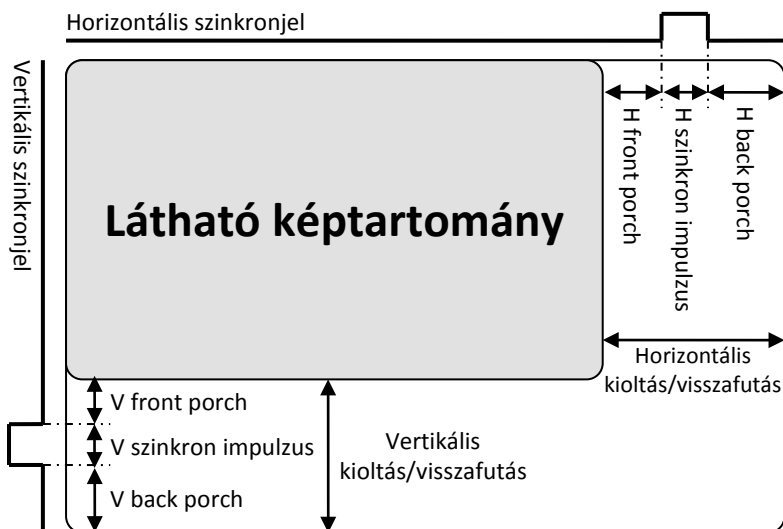


aljzat (anya)

Láb	Funkció
1	Piros videojel
2	Zöld videojel
3	Kék videojel
4	Fenntartott
5	GND
6	Piros GND
7	Zöld GND
8	Kék GND

Láb	Funkció
9	5V DDC tápfeszültség
10	Szinkron GND
11	Monitor ID bit 0 (opcionális)
12	DDC SDA (I ² C adat)
13	Horizontális szinkronjel
14	Vertikális szinkronjel
15	DDC SCL (I ² C órajel)

A VGA kép két fő részre osztható: a látható tartományra és a kioltási/visszafutási (nem látható) tartományra. Ezt szemlélteti az 5-2. ábra. A látható képtartományban minden pixel órajel ütemre ki kell adni a piros, a zöld és a kék színjelekre az aktuális pixel színét meghatározó értéket. A látható képtartományon kívül a színjelek értéke 0 V kell, hogy legyen (kioltás). A horizontális, illetve a vertikális kioltási tartományok további három részre oszthatók fel: a szinkron impulzus előtti front porch szakaszra, a szinkronjel aktív szakaszára (szinkron impulzus), valamint a következő sor/kép kezdetéig tartó back porch szakaszra. A horizontális és a vertikális szinkronjel az adott felbontástól függően lehet aktív alacsony, illetve aktív magas. A vertikális visszafutás alatt is ki kell adni a horizontális szinkron impulzusokat.



5-2. ábra: A VGA kép felépítése.

A 640 x 480 @ 60 Hz, valamint a 800 x 600 @ 72 Hz VGA felbontásokhoz tartozó időzítési paramétereket az 5-3. táblázat tartalmazza. A többi szabványos felbontás időzítési paraméterei megtalálhatók a <http://tinyvga.com/vga-timing> weboldalon. A megadott pixel órajel frekvenciától el lehet térni, ha például kisebb sor felbontást szeretnénk, azonban a szinkronjelek időzítési paramétereit mindig be kell tartani.

5-3. táblázat: VGA időzítési paraméterek.

Felbontás	640 x 480 @ 60 Hz				800 x 600 @ 72 Hz			
Pixel órajel	25 MHz				50 MHz			
H szinkron	aktív alacsony				aktív magas			
V szinkron	aktív alacsony				aktív magas			
	Horizontális időzítés		Vertikális időzítés		Horizontális időzítés		Vertikális időzítés	
	pixel	µs	sor	ms	pixel	µs	sor	ms
Látható rész	640	25,6	480	15,36	800	16	600	12,48
Front porch	16	0,64	10	0,32	56	1,12	37	0,77
Szinkron impulzus	96	3,84	2	0,64	120	2,4	6	0,12
Back porch	48	1,92	29	0,928	64	1,28	23	0,48
Összesen	800	32	521	16,7	1040	20,8	666	13,85

A VGA jelekhez tartozó FPGA I/O lábak beállítása a következő legyen:

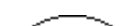
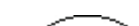
- **I/O szabvány (I/O standard):** LVCMOS33
- **A kimenet meghajtási erőssége (output drive strength):** 8 mA
- **A kimenet jelváltási sebessége (output slew rate):** FAST

5.1.2 PS/2 interfész

5.1.2.1 Általános ismertető

A modulon található két darab 6 pólusú mini-DIN anya csatlakozó lehetőséget biztosít PS/2 perifériák (billentyűzet, egér) illesztésére. A PS/2 csatlakozók lábkiosztását az 5-4. táblázat mutatja. A csatlakozókra kiadott tápfeszültség egy jumper segítségével beállítható 5 V-ra vagy 3,3 V-ra, azonban több PS/2 periféria csak 5 V-ról működik megbízhatóan.

5-4. táblázat: A PS/2 csatlakozó lábkiosztása.

dugó (apa)

aljzat (anya)

Láb	Funkció
1	Soros adat
2	Nincs bekötve
3	GND

Láb	Funkció
4	Tápfeszültség
5	Órajel
6	Nincs bekötve

A PS/2 interfész egyszerű, kétirányú szinkron soros kommunikációs protokollt használ. Az órajel és a soros adatvonal nyitott kollektoros meghajtású. Alapállapotban a busz mindkét vonala magas szintű, csak ebben az állapotban küldhet a periféria adatot. Mindig a host rendelkezik a busz felett és bármikor megszakíthatja a kommunikációt az órajel vonal alacsony szintre húzásával. Az órajelet mindig a periféria generálja. Ha a host kommunikálni akar a perifériával, akkor először az órajel vonal alacsony szintre húzásával megakadályozza, hogy a periféria adatot tudjon küldeni. Ezután a host lehúzza az adatvonalat, majd pedig elengedi az órajelet. Ezzel jelzi a perifériának, hogy generálhatja az adatátvitelhez szükséges órajel pulzusokat. A PS/2 busz időzítési paramétereit az 5-5. táblázat tartalmazza.

5-5. táblázat: A PS/2 busz időzítési paramétereit.

Szimbólum	Paraméter	Min.	Max.	Egység
T_{CLK}	Az órajel alacsony/magas szintjének időtartama	30	50	μs
T_{SU}	Adat előkészítési idő	5	25	μs
T_{HLD}	Adat tartási idő	5	-	μs
T_{HI}	Az órajel magas szintjének időtartama, mielőtt a periféria elkezdhet adatot küldeni	50	-	μs
T_{LO}	Az órajel alacsony szintjének időtartama az adatátvitel megszakításához	100	-	μs
T_{CHK}	A hoszt adatátviteli kérés ellenőrzésének periódusideje a periféria részéről	-	10	ms
T_{RTS}	Az órajel alacsony szintre húzásától a periféria által generált első órajel pulzusig eltelt idő	-	15	ms
T_{RESP}	A periféria válaszüzeje a hoszt kérésére	-	20	ms

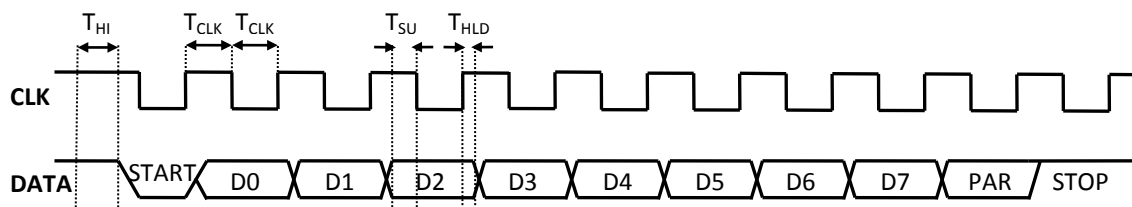
A PS/2 busznak három állapota lehetséges:

- **CLK=H, DATA=H:** alapállapot
- **CLK=L, DATA=H:** a hoszt tiltja a kommunikációt
- **CLK=H, DATA=L:** a hoszt jelzi, hogy adatot akar küldeni a perifériának

Az adatátvitel 11 vagy 12 bites keretekben történik. Egy keret felépítése a következő:

- 1 start bit (mindig 0).
- 8 adatbit (az LSb kerül elküldésre először).
- 1 paritásbit (páratlan számúra egészíti ki az adatbitekben található egyeseket).
- 1 stop bit (mindig 1).
- 1 nyugtázó bit (csak a hoszt-tól az eszköz felé történő adatküldés esetén).

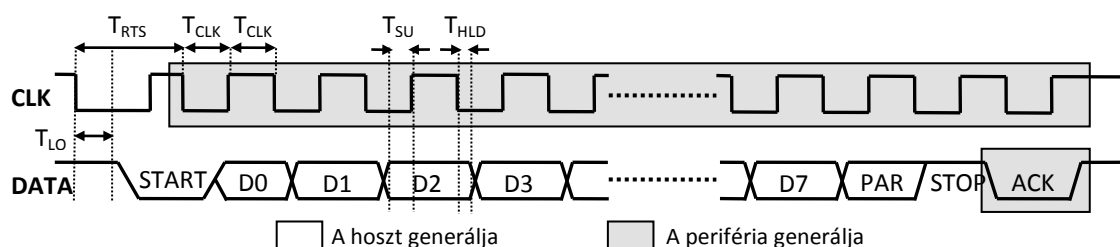
A perifériától a hoszt felé történő adatátvitelt az 5-3. ábra szemlélteti. Ha a periféria adatot szeretne küldeni, akkor először ellenőrzi az órajel vonal állapotát, hogy magas szintű-e. Ha az órajel vonal alacsony szintű, akkor a hoszt tiltja az adatátvitelt. Ez esetben a perifériának tárolnia kell az elküldendő adatot, amíg a hoszt el nem engedi az órajelet. Az órajel vonalnak legalább T_{HI} ideig magas szintűnek kell lennie, mielőtt a periféria elkezdheti az adatküldést. A periféria az adatvonal értékét akkor változtatja meg, mikor az órajel magas szintű. Ha az órajel alacsony szintű, akkor az adatvonalon érvényes adat van, melyet a hoszt beolvas. Ha a hoszt megszakítja az adatátvitelt a STOP bit előtt az órajel vonal legalább T_{LO} ideig történő alacsony szintre húzásával, akkor a perifériának az első bájtól kezdve a teljes adatsomagot újból el kell küldenie (tehát nem csak azt a bájt, amelynek elküldése meg lett szakítva).



5-3. ábra: Adatátvitel a perifériától a hoszt felé.

A hoszt-tól a periféria felé történő adatátvitelt az 5-4. ábra szemlélteti. Mivel az órajelet mindig a periféria generálja, ezért a hoszt-nak jeleznie kell az adatküldést a következő módon: először legalább T_{LO} ideig alacsony szintre húzza az órajelet, majd alacsony szintre húzza az adatvonalat, végül pedig elengedi az órajelet. Ezt a busz állapotot a perifériának legalább T_{CHK} időnként ellenőriznie kell. Amint a periféria észleli a hoszt adatküldési szándékát, elkezdi kiadni 10 órajel

pulzust a 8 adatbit, a paritásbit és a stop bit átviteléhez. Az órajel alacsony szintre húzásától a periféria által generált első órajel pulzusig legfeljebb T_{RTS} idő telhet el. A host akkor változtatja meg az adatvonal értékét, ha az órajel alacsony szintű. Ha az órajel magas szintű, akkor az adatvonalon érvényes adat van, amelyet a periféria beolvas. A STOP bit vétele után a periféria nyugtázza a vett bájtot az adatvonal alacsony szintre húzásával és egy órajel pulzus kiadásával (ACK bit). Az ACK bit küldésekor a periféria az órajel magas szintje alatt húzza le az adatvonalat alacsony szintre. Ha a host nem engedi el magas szintre az adatvonalat a nyugtázás után, akkor a periféria mindaddig generálni fog órajelet, amíg az adatvonal magas szintű nem lesz. A host bármikor megszakíthatja az adatátvitelt az ACK bit előtt az órajel vonal legalább T_{LO} ideig történő alacsony szintre húzásával. Ha a parancs, melyet a host küldött, választ igényel, akkor azt a perifériának az órajel elengedésétől számított T_{RESP} időn belül el kell küldenie.



5-4. ábra: Adatátvitel a host-tól a periféria felé.

A PS/2 jelekhez tartozó FPGA I/O lábak beállítása a következő legyen:

- **I/O szabvány (I/O standard):** LVCMOS33
- **A kimenet meghajtási erőssége (output drive strength):** 8 mA
- **A kimenet jelváltási sebessége (output slew rate):** SLOW

5.1.2.2 Billentyűzet kezelése

A gombok megnyomásakor vagy felengedésekor a billentyűzet úgynevezett scan kódokat küld a host-nak. A scan kódoknak két típusa van, az egyik a gombok lenyomásához, a másik a gombok felengedéséhez tartozik. Mindegyik billentyűhöz egy egyedi kód van hozzárendelve. Ezen kódok összessége a scan kódkészlet, melyből háromféle van:

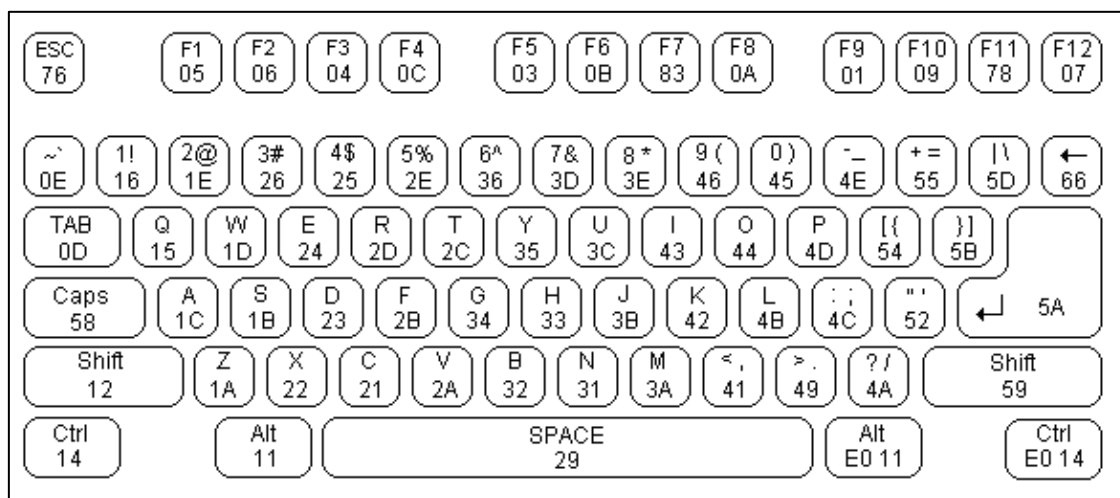
- **Scan kódkészlet 1:** ez az eredeti XT scan kódkészlet, nem mindegyik billentyűzet támogatja.
- **Scan kódkészlet 2:** ez az alapértelmezett kódkészlet minden modern billentyűzet esetén.
- **Scan kódkészlet 3:** opcionális PS/2 scan kódkészlet, nem mindegyik billentyűzet támogatja.

Az 5-5. és az 5-6. ábrákon láthatók az egyes billentyűkhöz tartozó scan kódok a scan kódkészlet 2 esetén. A normál billentyűk lenyomását az ábrákon látható scan kódok jelzik, a kibővített billentyűk lenyomásakor a scan kódot megelőzi egy 0xE0 prefix. A normál billentyűk felengedésekor egy 0xF0 prefixet, a kibővített billentyűk felengedésekor pedig egy 0xE0 0xF0 prefixet küld a billentyűzet a scan kód előtt. Az 5-6. táblázatban látható néhány példa a billentyűk megnyomásakor és felengedésekor elküldött adatokra. A PRINT SCREEN és a PAUSE/BREAK gombokhoz speciális scan kódok tartoznak.

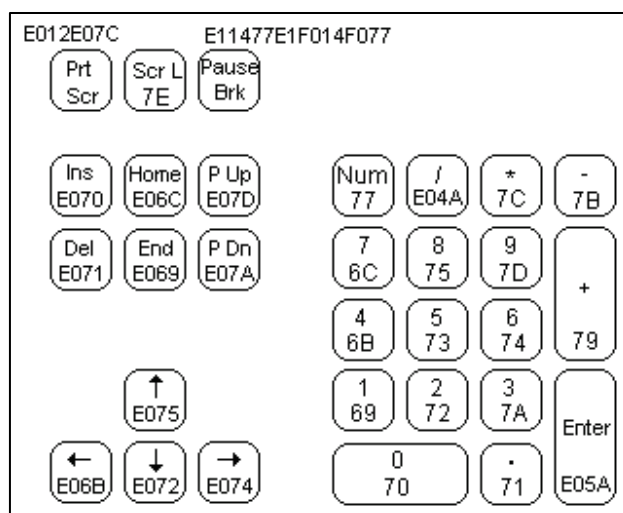
5-6. táblázat: Példák a scan kódokra.

Billentyű	Megnyomás	Felengedés
A	0x1C	0xF0 0x1C
DEL	0xE0 0x71	0xE0 0xF0 0x71
PRINT SCREEN	0xE0 0x12 0xE0 0x7C	0xE0 0xF0 0x12 0xE0 0xF0 0x7C
PAUSE/BREAK	0xE1 0x14 0x77 0xE1 0xF0 0x14 0xF0 0x77	nincs

Ha egy gombot lenyomva tartunk, akkor adott idő eltelte után (typematic delay) a billentyűzet adott sebességgel (typematic rate) ismételni fogja a megnyomáshoz tartozó scan kódot. Ha több gomb is le van nyomva, akkor csak a legutoljára lenyomott gombhoz tartozó scan kód kerül ismétlésre.



5-5. ábra: A normál billentyűk scan kódjai (scan kódkészlet 2).



5-6. ábra: A kibővített és a numerikus billentyűk scan kódjai (scan kódkészlet 2).

A billentyűzet a tápfeszültség bekapcsolásakor vagy a reset parancs hatására öntesztet hajt végre, melyet a LED-ek felvillantásával jelez és az eredményt elküldi a host-nak: 0xAA-t sikeres teszt esetén, illetve 0xFC-t hiba esetén. A reset utáni alapértelmezett állapot a következő:

- **Ismétlés késleltetése (typematic delay):** 500 ms
- **Ismétlési sebesség (typematic rate):** 10,9 karakter/s
- **Kódkészlet:** scan kódkészlet 2
- **LED-ek állapota:** mindegyik kikapcsolva

A billentyűzetnek elküldhető fontosabb parancsokat az 5-7. táblázat tartalmazza. A parancsokkal kapcsolatos tudnivalók:

- A billentyűzet törli a kimeneti puffert parancs vétele esetén.
- A billentyűzet minden érvényes parancsot és paramétert nyugtáz a 0xFA elküldésével.
- Érvénytelen parancs vagy paraméter esetén a billentyűzet 0xFE bájtot küld a host-nak.
- A billentyűzet nem küldhet scan kódot a host-nak a vett parancs feldolgozása alatt.
- Ha a billentyűzet a vett parancshoz tartozó paraméterre vár, de helyette egy újabb parancsot kap, akkor a korábbi parancsot eldobja és az újat dolgozza fel.

5-7. táblázat: A billentyűzetnek elküldhető fontosabb parancsok.

Parancs	Funkció																
0xFF	Reset és az önteszt végrehajtása.																
0xFE	A legutolsó nem 0xFE értékű bájtt ismételt elküldése a host-nak. Nincs nyugtázás (0xFA).																
0xF6	A reset utáni alapértelmezett állapot beállítása.																
0xF5	A reset utáni alapértelmezett állapot beállítása és a billentyűk figyelésének tiltása.																
0xF4	A billentyűk figyelésének engedélyezése.																
0xF3	A lenyomva tartott billentyűk ismétlési késleltetésének és ismétlési sebességének beállítása. A nyugta (0xFA) vételét követően a hostnak az alábbi bájtot kell elküldenie: <table><tr><th>bit 7</th><th>bit 6</th><th>bit 5</th><th>bit 4</th><th>bit 3</th><th>bit 2</th><th>bit 1</th><th>bit 0</th></tr><tr><td>0</td><td colspan="2">ismétlés késleltetése (typematic delay) 00: 0,25 s 01: 0,50 s 10: 0,75 s 11: 1,00 s</td><td colspan="5">ismétlési sebesség (typematic rate) az értékeket lásd az 5-8. táblázatban</td></tr></table>	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	0	ismétlés késleltetése (typematic delay) 00: 0,25 s 01: 0,50 s 10: 0,75 s 11: 1,00 s		ismétlési sebesség (typematic rate) az értékeket lásd az 5-8. táblázatban				
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0										
0	ismétlés késleltetése (typematic delay) 00: 0,25 s 01: 0,50 s 10: 0,75 s 11: 1,00 s		ismétlési sebesség (typematic rate) az értékeket lásd az 5-8. táblázatban														
0xF0	Az aktuális scan kódkészlet kiválasztása vagy lekérdezése. A nyugta (0xFA) vételét követően a host a 0x01, a 0x02 vagy a 0x03 bájtt elküldésével választhatja ki a megfelelő kódkészletet. Ha a host a 0x00 bájtot küldi el, akkor a billentyűzet válasza az aktuális scan kódkészlet sorszáma.																
0xEE	A 0xEE bájtt elküldése a host-nak (echo). Nincs nyugtázás (0xFA).																
0xED	A LED-ek állapotának beállítása. A nyugta (0xFA) vételét követően a host-nak az alábbi bájtot kell elküldenie: <table><tr><th>bit 7</th><th>bit 6</th><th>bit 5</th><th>bit 4</th><th>bit 3</th><th>bit 2</th><th>bit 1</th><th>bit 0</th></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>CAPS LOCK 0: ki 1: be</td><td>NUM LOCK 0: ki 1: be</td><td>SCROLL LOCK 0: ki 1: be</td></tr></table>	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	0	0	0	0	0	CAPS LOCK 0: ki 1: be	NUM LOCK 0: ki 1: be	SCROLL LOCK 0: ki 1: be
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0										
0	0	0	0	0	CAPS LOCK 0: ki 1: be	NUM LOCK 0: ki 1: be	SCROLL LOCK 0: ki 1: be										

5-8. táblázat: Karakter ismétlési sebesség beállítások (0xF3 parancs).

bit 4-0	karakter/s	bit 4-0	karakter/s	bit 4-0	karakter/s	bit 4-0	karakter/s
0x00	30,0	0x08	15,0	0x10	7,5	0x18	3,7
0x01	26,7	0x09	13,3	0x11	6,7	0x19	3,3
0x02	24,0	0x0A	12,0	0x12	6,0	0x1A	3,0
0x03	21,8	0x0B	10,9	0x13	5,5	0x1B	2,7
0x04	20,7	0x0C	10,0	0x14	5,0	0x1C	2,5
0x05	18,5	0x0D	9,2	0x15	4,6	0x1D	2,3
0x06	17,1	0x0E	8,6	0x16	4,3	0x1E	2,1
0x07	16,0	0x0F	8,0	0x17	4,0	0x1F	2,0

5.1.2.3 Egér kezelése

A szabványos PS/2 egér a következő adatokat szolgáltatja a host-nak: X irányú elmozdulást (balra/jobbra), Y irányú elmozdulást (fel/le), bal gomb állapotát, középső gomb állapotát, valamint a jobb gomb állapotát. A további bemenetekkel is rendelkező egerek a következőkben leírtaktól eltérő formátumban küldik el ezeket az adatokat, ha a kiegészítések engedélyezve vannak. Egy ilyen kiegészítés pl. a Microsoft IntelliMouse, ezzel azonban itt nem foglalkozunk.

A szabványos egér két számlálót tartalmaz, melyek nyomon követik az X és az Y irányú elmozdulást. A számlálók értéke 9 bites kettes komplementum formátumú és mindegyik számlálóhoz tartozik egy-egy túlsordulás bit is. A számlálók értéke a gombok állapotával együtt egy 3 bájtos adatcsomagban kerül elküldésre a host felé, melynek tartalmát az 5-9. táblázat szemlélteti. Egy ilyen adatcsomag elküldése után a számlálók törlődnek. Az elmozdulás értékek az előző adatcsomag elküldésekor aktuális helyzethez viszonyított relatív értékek. Negatív érték a balra, illetve a lefele történő elmozduláshoz tartozik.

5-9. táblázat: Az egér elmozdulását és a gombok állapotát tartalmazó adatcsomag.

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1. bájt	Y túlsordulás	X túlsordulás	Y előjel bit	X előjel bit	1	középső gomb	jobb gomb	bal gomb
2. bájt	X irányú elmozdulás							
3. bájt	Y irányú elmozdulás							

A számlálók növelésének és csökkentésének mértékét a felbontás határozza meg. Ennek alapértelmezett értéke 4 számlálás/mm, amely egy parancs elküldésével megváltoztatható. A skálázás nincs hatással a számlálókra, de hatással van az elküldött elmozdulás adatokra. Az alapértelmezett skálázás 1:1, ekkor az elküldött elmozdulás értékek megegyeznek a számlálók értékével. A 2:1 skálázás beállítása esetén az elküldött elmozdulás értékek és a számlálók értéke közötti kapcsolatot az 5-10. táblázat mutatja.

5-10. táblázat: A számlálók értéke és az elküldött adat közötti kapcsolat 2:1 skálázás esetén.

Számláló értéke	0	1	2	3	4	5	N > 5
Elküldött adat	0	1	1	3	6	9	2N

Az egér háromféle üzemmódban képes működni:

- **Stream módban** az egér automatikusan küld adatcsomagot elmozdulás vagy a gombok állapotának megváltozása esetén. Az adatcsomagok küldésének maximális gyakorisága (mintavételezési frekvencia) beállítható. **Alapértelmezésben az adatküldés tiltva van, ezt a host-nak külön engedélyezni kell a 0xF4 parancssal.**
- **Remote módban** az egér a beállított mintavételezési frekvenciával folyamatosan frissíti a számlálókat és beolvassa a gombok állapotát, adatküldés azonban csak a host utasítására történik.
- **Wrap módban** az egér minden vett bájtot visszaküld a host-nak. Két kivétel van: a „reset” (0xFF) és a „wrap üzemmód tiltása” (0xEC) parancs.

A tápfeszültség bekapcsolásakor vagy a reset parancs hatására az egér öntesztet hajt végre és az eredményt (0xAA-t sikeres teszt esetén, illetve 0xFC-t hiba esetén), valamint a 0x00 eszközazonosító bájtot (szabványos PS/2 egér) elküldi a host-nak. A reset utáni alapértelmezett állapot a következő:

- **Felbontás:** 4 számlálás/mm
- **Skálázás:** 1:1
- **Üzem mód:** stream mód
- **Mintavételezési frekvencia:** 100 Hz
- **Az állapot automatikus elküldése:** tiltva

Az egérnek elküldhető parancsokat az 5-11. táblázat tartalmazza. A parancsokkal kapcsolatos tudnivalók:

- Az egér törli a kimeneti pufferét parancs vétele esetén.
- Az egér minden érvényes parancsot és paramétert nyugtáz a 0xFA bájt elküldésével.
- Érvénytelen parancs vagy paraméter esetén az egér a 0xFE bájtot (újraküldés kérése) vagy a 0xFC bájtot (hiba) küldi el a host-nak.
- **Stream mód esetén ki kell adni a 0xF5 parancsot minden egyéb parancs előtt.**
- Az egér nem küldhet adatot a host-nak a vett parancs feldolgozása alatt.

5-11. táblázat: Az egérnek elküldhető parancsok.

Parancs	Funkció																																				
0xFF	Reset és az önteszt végrehajtása.																																				
0xFE	A legutoljára elküldött adatcsomag ismételt elküldése a host-nak. Nincs nyugtázás (0xFA).																																				
0xF6	A reset utáni alapértelmezett állapot beállítása.																																				
0xF5	Az adatcsomagok küldésének tiltása. A nyugta (0xFA) elküldése után az egér törli az elmozdulás számlálót. A parancs csak a stream módra van hatással.																																				
0xF4	Az adatcsomagok küldésének engedélyezése. A nyugta (0xFA) elküldése után az egér törli az elmozdulás számlálót. A parancs csak a stream módra van hatással.																																				
0xF3	A mintavételi frekvencia beállítása. A nyugta (0xFA) elküldése után az egér egy bájtot vár a host-tól, amely a beállítandó mintavételi frekvencia (érvényes értékek: 10, 20, 40, 60, 80, 100 vagy 200 Hz). A paraméter nyugtázása után (0xFA) az egér törli az elmozdulás számlálót.																																				
0xF2	Az eszközazonosító lekérdezése. A nyugtázás (0xFA) után az egér elküldi az 1 bájtos eszközazonosítót (0x00 a szabványos PS/2 egér esetén) és törli az elmozdulás számlálót.																																				
0xF0	A remote üzemmód beállítása. A nyugta (0xFA) elküldése után az egér törli az elmozdulás számlálót és remote üzemmódba lép.																																				
0xEE	A wrap üzemmód beállítása. A nyugta (0xFA) elküldése után az egér törli az elmozdulás számlálót és wrap üzemmódba lép.																																				
0xEC	A wrap üzemmód tiltása. A nyugta (0xFA) elküldése után az egér törli az elmozdulás számlálót és visszaállítja az előző üzemmódot (stream vagy remote).																																				
0xEB	Az egér elmozdulását és a gombok állapotát tartalmazó adatcsomag elküldése a host-nak.																																				
0xEA	A stream üzemmód beállítása. A nyugta (0xFA) elküldése után az egér törli az elmozdulás számlálót és stream üzemmódba lép.																																				
0xE9	A státusz lekérdezése. A nyugtázás (0xFA) után az egér elküldi az alábbi 3 bájtos adatcsomagot és törli az elmozdulás számlálót. <table><tr><td></td><td>bit 7</td><td>bit 6</td><td>bit 5</td><td>bit 4</td><td>bit 3</td><td>bit 2</td><td>bit 1</td><td>bit 0</td></tr><tr><td>1. bájt</td><td>0</td><td>üzemmód 0: stream 1: remote</td><td>adatküldés 0: tiltva 1: engedélyezve</td><td>skálázás 0: 1:1 1: 2:1</td><td>0</td><td>bal gomb</td><td>középső gomb</td><td>jobb gomb</td></tr><tr><td>2. bájt</td><td colspan="8">Felbontás</td></tr><tr><td>3. bájt</td><td colspan="8">Mintavételi frekvencia</td></tr></table>		bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	1. bájt	0	üzemmód 0: stream 1: remote	adatküldés 0: tiltva 1: engedélyezve	skálázás 0: 1:1 1: 2:1	0	bal gomb	középső gomb	jobb gomb	2. bájt	Felbontás								3. bájt	Mintavételi frekvencia							
	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0																													
1. bájt	0	üzemmód 0: stream 1: remote	adatküldés 0: tiltva 1: engedélyezve	skálázás 0: 1:1 1: 2:1	0	bal gomb	középső gomb	jobb gomb																													
2. bájt	Felbontás																																				
3. bájt	Mintavételi frekvencia																																				

0xE8	A felbontás beállítása. A nyugta (0xFA) elküldése után az egér egy bájtot vár a host-tól. A paraméter nyugtázása után (0xFA) az elmozdulás számlálók törlődnek.				
	Vett bájt	0x00	0x01	0x02	0x03
	Felbontás (számlálás/mm)	1	2	4	8
0xE7	A 2:1 skálázás beállítása. A nyugta (0xFA) elküldése után az egér törli az elmozdulás számlálókat.				
0xE6	Az 1:1 skálázás beállítása. A nyugta (0xFA) elküldése után az egér törli az elmozdulás számlálókat.				

5.1.2.4 További információk

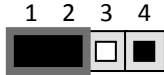
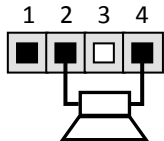
A PS/2 interfészről, valamint a billentyűzet és az egér kezeléséről további hasznos információk találhatóak az alábbi weboldalakon:

- <http://www.computer-engineering.org/>
- <http://www.win.tue.nl/~aeb/linux/kbd/scancodes.html>

5.1.3 Audio interfész

Az audio interfész lehetővé teszi, hogy a modulon található beépített hangszórót vagy a modulhoz csatlakoztatott külső hangszórót impulzusszélesség modulált vagy impulzussűrűség modulált digitális négyszögjellel vezéreljük. A hangszóró meghajtását a TS4890IST típusú 1 wattos végerősítő IC végzi, amely a hozzákapcsolt passzív alkatrészekkel együtt egy kb. 12 kHz határfrekvenciájú másodfokú aluláteresztő szűrőt alkot. A külső hangszóró csatlakozó használatát az 5-12. táblázat mutatja.

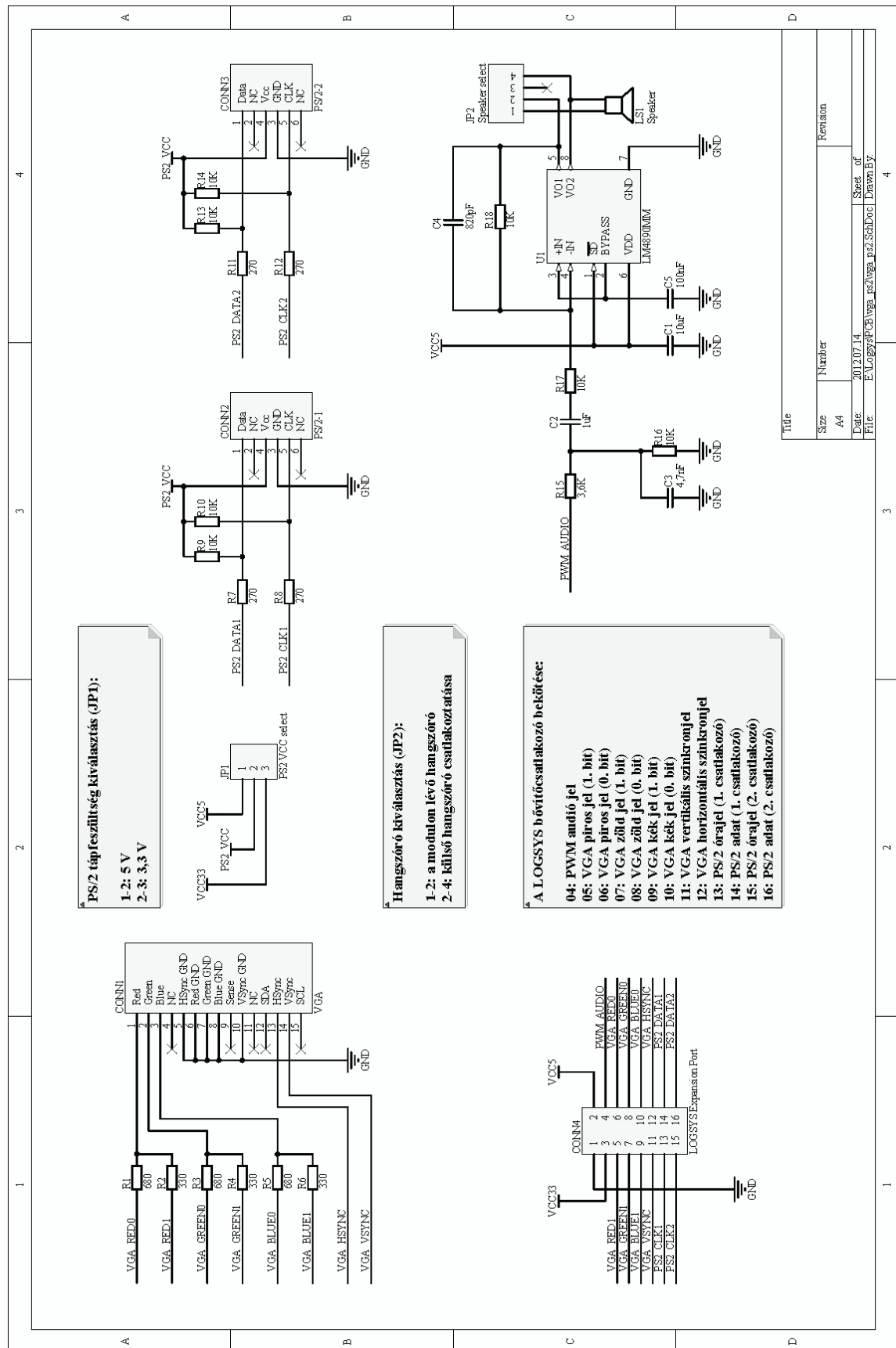
5-12. táblázat: A külső hangszóró csatlakozó.

Csatlakozó	Funkció
	A beépített hangszóró használatához a csatlakozó 1. és 2. lábát zárjuk rövidre egy jumperrel.
	A külső hangszórót a csatlakozó 2. és 4. lábára kell kötni, ez esetben a beépített hangszóró nem működik.

Az audio interfészhez tartozó FPGA I/O láb beállítása a következő legyen:

- **I/O szabvány (I/O standard):** LVCMOS33
- **A kimenet meghajtási erőssége (output drive strength):** 4 mA
- **A kimenet jelváltási sebessége (output slew rate):** HIGH

5.1.4 Kapcsolási rajz

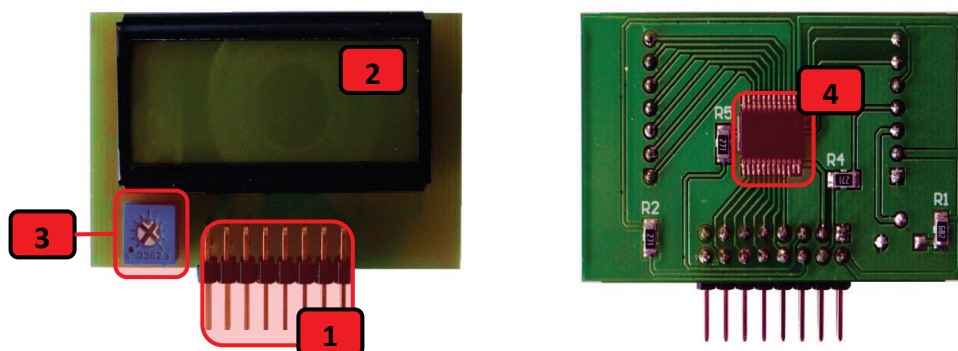


5.2 LOGSYS karakteres LCD kijelző modul

A modul lehetővé teszi karakteres LCD kijelző illesztését a LOGSYS FPGA kártyákhoz. A modul felépítését az 5-7. ábra szemlélteti. A bővíítőcsatlakozó lábkiosztását az 5-13. táblázat mutatja.

A modul főbb jellemzői:

- Samsung KS0066 vezérlőt tartalmazó 2 x 8 karakteres LCD kijelző (adatlap: <http://www.bolymin.com.tw/Doc/BC0802B series VER02.zip>)
- Potenciométeres kontraszt beállítás
- 8 bites kétirányú kommunikációs interfész



5-7. ábra: A LOGSYS karakteres LCD kijelző modul felépítése.

A modul felépítése:

1. LOGSYS bővíítőcsatlakozó
2. 2 x 8 karakteres LCD kijelző
3. Kontraszt beállító potenciométer
4. 74LVC4245A típusú buszmeghajtó és szintillesztő IC

5-13. táblázat: A bővíítőcsatlakozó lábkiosztása.

Láb	Típus ²	Funkció
1	PWR	GND
2	PWR	5V tápfeszültség
3	PWR	3,3V tápfeszültség
4	I	Parancs/adat kiválasztó jel (RS)
5	I	Chip select jel (E)
6	I	Írás/olvasás kiválasztó jel (R/W)
7	I/O	LCD adatbusz (0. bit)
8	I/O	LCD adatbusz (1. bit)

Láb	Típus ²	Funkció
9	I/O	LCD adatbusz (2. bit)
10	I/O	LCD adatbusz (3. bit)
11	I/O	LCD adatbusz (4. bit)
12	I/O	LCD adatbusz (5. bit)
13	I/O	LCD adatbusz (6. bit)
14	I/O	LCD adatbusz (7. bit)
15		nincs bekötve
16		nincs bekötve

² PWR: tápellátás, I: bemenet, I/O: kétirányú

Például: I (bemenet) esetén az FPGA számára az adott láb kimenet, a modul számára bemenet.

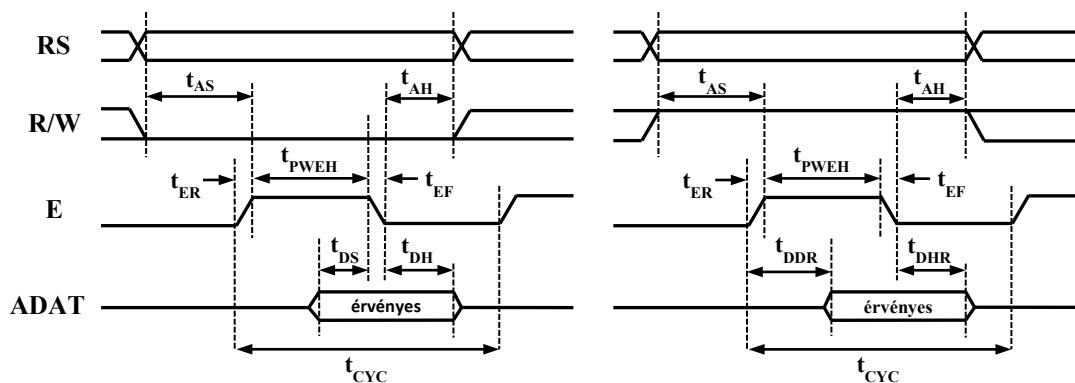
5.2.1 Kommunikációs interfész

Az LCD kijelzőben található vezérlő áramkör két darab 8 bites regisztert tartalmaz, melyeket a 8 bites kétirányú kommunikációs interfészen keresztül érhetünk el. Az utasításregiszter (IR) tárolja a végrehajtandó művelet kódját, valamint a cím információt az adatmemória (DDRAM), illetve a karakter generátor RAM (CGRAM) számára. Az adatregiszter (DR) tárolja a memóriába írandó, vagy a memóriából kiolvasott adatot. A kommunikációs interfész parancs/adat (RS) és írás/olvasás (R/W) kiválasztó jeleivel választhatjuk ki az elérni kívánt regisztert (5-14. táblázat), a chip select jellel (E) pedig engedélyezhetjük az adatátvitelt.

5-14. táblázat: A regiszterek kiválasztása.

RS	R/W	Művelet
0	0	Parancs küldése az LCD kijelzőnek (utasításregiszter írás).
0	1	A foglaltság jelzés (7. bit) és a címszámláló (0. bit – 6. bit) kiolvasása.
1	0	Adat írása a DDRAM-ba vagy a CGRAM-ba (adatregiszter írás).
1	1	Adat olvasása a DDRAM-ból vagy a CGRAM-ból (adatregiszter olvasás).

A kommunikációs interfész idődiagramja az 5-8. ábrán látható, az időzíteni paramétereket az 5-15. táblázat tartalmazza.



5-8. ábra: Az írási és az olvasási művelet idődiagramja.

5-15. táblázat: A kommunikációs interfész időzíteni paraméterei ($T_a=25^\circ\text{C}$, $V_{dd}=5,0 \pm 0,5\text{ V}$).

Paraméter		Minimum	Tipikus	Maximum	Egység
Ciklusidő	t_{CYC}	500	-	-	ns
Chip select jel magas szintje	t_{PWEH}	230	-	-	ns
Chip select jel felfutási/lefutási idő	t_{ER} , t_{EF}	-	-	20	ns
Regiszter kiválasztás előkészítési idő	t_{AS}	40	-	-	ns
Regiszter kiválasztás tartási idő	t_{AH}	10	-	-	ns
Adat előkészítési idő (írás)	t_{DS}	80	-	-	ns
Adat tartási idő (írás)	t_{DH}	10	-	-	ns
Adat késleltetési idő (olvasás)	t_{DDR}	-	-	100	ns
Adat tartási idő (olvasás)	t_{DHR}	5	-	-	ns

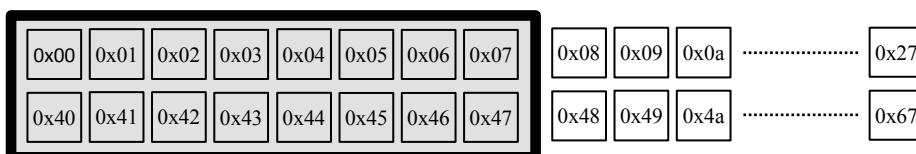
Az LCD interfészhez tartozó FPGA I/O lábak beállítása a következő legyen:

- **I/O szabvány (I/O standard):** LVCMOS33
- **A kimenet meghajtási erőssége (output drive strength):** 4 mA
- **A kimenet jelváltási sebessége (output slew rate):** SLOW

5.2.2 Memóriák az LCD vezérlőben

5.2.2.1 Adatmemória (DDRAM)

Az LCD vezérlőben található 80 x 8 bites adatmemória (DDRAM) lehetőséget biztosít 80 darab karakter tárolására, a kijelzőn megjelenítendő karakterek kódját ide kell beírni. A DDRAM írásra vagy olvasásra a DDRAM címet beállító utasítással választható ki. Egysoros üzemmódban a rendelkezésre álló címtartomány 0x00-0x4f, kétsoros üzemmódban az első sorhoz a 0x00-0x27, a második sorhoz pedig a 0x40-0x67 címtartomány tartozik (5-9. ábra). A kijelző 2 x 8-as ablaka a DDRAM-ban karakterenként eltolható balra vagy jobbra a megfelelő shiftelési parancs kiadásával.



5-9. ábra: A karakterek címe a DDRAM-ban kétsoros üzemmód esetén.

5.2.2.2 Karaktergenerátor ROM (CGROM)

Az LCD vezérlőben található karaktergenerátor ROM (CGROM) tartalmazza a beépített karakterkészletet (angol/európai karakterkészlet), amely az 5-10. ábrán látható.

b7-b4 b3-b0	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
CG RAM (1)	0000	±	!	0	P	°	F	G	E	A	°	°	°	°	°	°
0001	(2)	≡	!	1	A	Q	a	q	Q	z	i	°	°	°	°	°
0010	(3)	7	"	2	B	R	b	r	E	6	°	°	°	°	°	°
0011	(4)	Δ	#	3	C	S	c	s	A	8	°	°	°	°	°	°
0100	(5)	{	\$	4	D	T	d	t	A	8	°	°	°	°	°	°
0101	(6)	{	%	5	E	U	e	u	A	8	°	°	°	°	°	°
0110	(7)	}	&	6	F	V	f	v	A	8	°	°	°	°	°	°
0111	(8)	}	'	7	G	W	w	G	U	R	°	°	°	°	°	°
1000	(1)	{	(	8	H	X	h	x	E	9	°	°	°	°	°	°
1001	(2)	{	)	9	I	Y	i	y	E	0	°	°	°	°	°	°
1010	(3)	{	*	:	J	Z	j	z	E	0	°	°	°	°	°	°
1011	(4)	{	+	:	K	L	k	l	E	0	°	°	°	°	°	°
1100	(5)	{	=	,	<	L	\	/	E	0	°	°	°	°	°	°
1101	(6)	{	-	=	M	I	m	>	E	0	°	°	°	°	°	°
1110	(7)	{	.	>	N	^	n	^	E	0	°	°	°	°	°	°
1111	(8)	{	/	?	0	_	o	Δ	E	0	°	°	°	°	°	°

5-10. ábra: Az LCD kijelző karakterkészlete.

5.2.2.3 Karaktergenerátor RAM (CGRAM)

Az LCD vezérlőben található 64 x 8 bites karaktergenerátor RAM (CGRAM) lehetőséget biztosít 8 darab 5 x 8 pixel méretű egyéni karakter definiálására. A CGRAM írásra vagy olvasásra a CGRAM címet beállító utasítással választható ki. A karaktertáblában az egyéni karakterekhez a 0x00-0x0f címek tartoznak, a felső nyolc karakter megegyezik az alsó nyolc karakterrel. A karakter mintázat értelmezése az 5-11. ábrán látható, az 1 bitekhez tartoznak a fekete pixelek.

CGRAM cím						Karakter mintázat (CGRAM adat)							
Bit						Bit							
5	4	3	2	1	0	7	6	5	4	3	2	1	0
karakter index (0-7)	0	0	0	x	x	x	1	1	1	1	0		
	0	0	1	x	x	x	1	0	0	0	1		
	0	1	0	x	x	x	1	0	0	0	1		
	0	1	1	x	x	x	1	1	1	1	0		
	1	0	0	x	x	x	1	0	1	0	0		
	1	0	1	x	x	x	1	0	0	1	0		
	1	1	0	x	x	x	1	0	0	0	1		
	1	1	1	x	x	x	0	0	0	0	0		

5-11. ábra: Egyéni karakterek definiálása.

5.2.3 Parancskészlet

5.2.3.1 Clear Display

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	0	0	0	0	1	1,53 ms

A parancs kiadásával törölhetjük a kijelzőt. Az LCD vezérlő feltölti a DDRAM-ot szóközökkel (0x20) és a címszámlálót 0x00 értékre állítja. A kurzor az első sor első karakteréhez kerül, a DDRAM és a CGRAM elérésekor a címszámláló értéke egyel növekedni fog.

5.2.3.2 Return Home

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	0	0	0	1	x	1,53 ms

A parancs kiadásának hatására a DDRAM címszámláló értéke 0x00 lesz, a kurzor helye az első sor első karaktere lesz, valamint a kijelző 2 x 8-as ablaka visszakerül az eredeti pozíciójába, ha az el volt shiftelve. A parancs nem módosítja a DDRAM tartalmát.

5.2.3.3 Entry Mode Set

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	0	0	1	I/D	SH	39 μs

A paranccsal beállítható a kurzor és a kijelző ablakának mozgási iránya. Az I/D bittel állítható be a kurzor mozgása és a címszámlálók működése. Ha I/D=1, akkor a kurzor jobbra fog elmozdulni és a címszámláló értéke egyel növekszik a memóriák (DDRAM, CGRAM) elérésekor. Ha I/D=0,

akkor a kurzor balra fog elmozdulni és a címszámláló értéke eggyel csökken a memóriák elérésekor. Ha SH=1, akkor a DDRAM írásakor a kijelző ablaka egyel jobbra vagy balra tolódik az I/D bit értékének megfelelően.

5.2.3.4 Display On/Off Control

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	0	1	D	C	B	39 μ s

A parancs kiadásával bekapcsolhatjuk (D=1), illetve kikapcsolhatjuk (D=0) a kijelzőt, valamint beállíthatjuk a kurzor működését. A kurzor megjelenítését a C bit vezérli (ha C=1, akkor látható a kurzor), a B bittel pedig a kurzor villogása állítható be (ha B=1, akkor a kurzor villogni fog).

5.2.3.5 Cursor or Display Shift

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	1	S/C	R/L	x	x	39 μ s

A parancs kiadásával léptethető a kurzor vagy a kijelző ablaka a memóriák írása vagy olvasása nélkül. Kétsoros üzemmódban a kurzor átlép a második sor első karakterére az első sor negyvenedik karaktere után. A kijelző ablakának léptetése esetén a címszámláló értéke nem változik. Az S/C és az R/L biteknek megfelelő műveleteket az 5-16. táblázat tartalmazza.

5-16. táblázat: Léptetési műveletek.

S/C	R/L	Művelet
0	0	A kurzor léptetése balra, a címszámláló értéke eggyel csökken.
0	1	A kurzor léptetése jobbra, a címszámláló értéke eggyel nő.
1	0	A kijelző ablakának shiftelése balra, a kurzor az ablakkal együtt mozog.
1	1	A kijelző ablakának shiftelése jobbra, a kurzor az ablakkal együtt mozog.

5.2.3.6 Function Set

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	1	DL	N	F	x	x	39 μ s

A paranccsal beállítható az adatbusz szélessége (DL=0: 4 bit, DL=1: 8 bit), a sorok száma (N=0: egy sor, N=1: két sor), valamint a karakterkészlet típusa (F=0: 5 x 8 pixel méretű karakterek, F=1: 5 x 11 pixel méretű karakterek). Az adatbusz szélességének alapértelmezett értéke 8 bit.

5.2.3.7 Set CGRAM Address

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	1	CGRAM cím						39 μ s

A parancs beírja a megadott CGRAM címet a címszámlálóba és kiválasztja a CGRAM elérését.

5.2.3.8 Set DDRAM Address

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	0	1	DDRAM cím							39 μ s

A parancs beírja a megadott DDRAM címet a címszámlálóba és kiválasztja a DDRAM elérését.

5.2.3.9 Read Busy Flag and Address

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
0	1	BF	Címszámláló értéke							0 μ s

A parancs segítségével kiolvashatjuk a foglaltság jelzést (BF) és a címszámláló aktuális értékét. Ha BF=1, akkor a korábban kiadott parancs végrehajtása folyamatban van. A foglaltság jelzés lekérdezésén kívül újabb parancs addig nem adható ki, amíg BF értéke ismét 0 nem lesz.

5.2.3.10 Write Data to RAM

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
1	0	A memóriába írandó adat								43 μ s

A parancs segítségével a megadott bájt beírhatjuk a kiválasztott memóriába a címszámláló által meghatározott címre. Az írási művelet végrehajtása után a címszámláló értéke a korábban kiadott *Entry Mode Set* parancs I/D bitjének megfelelően módosul.

5.2.3.11 Read Data from RAM

Parancs kódja										Végrehajtási idő
RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
1	1	A memóriából beolvasott adat								43 μ s

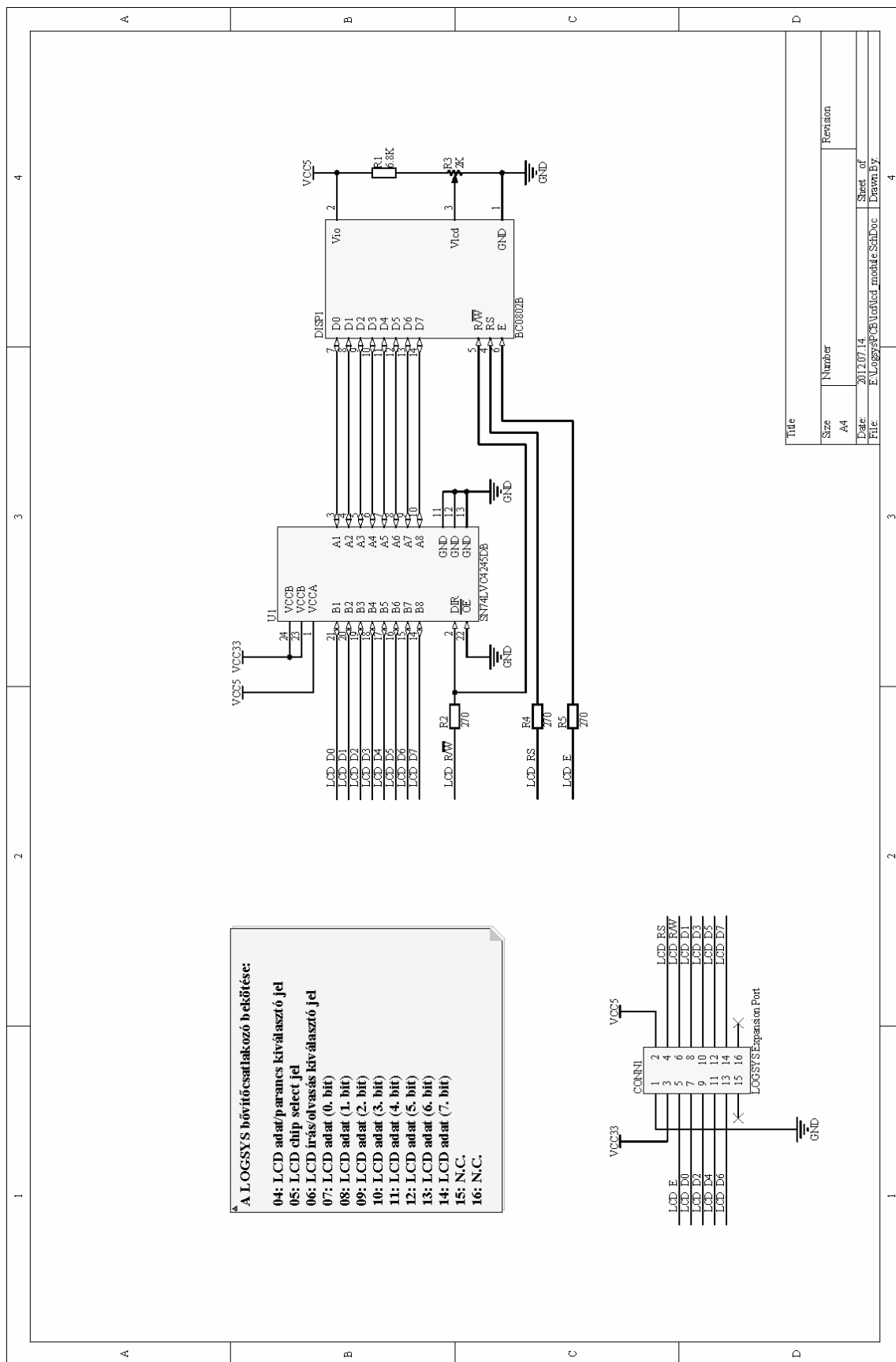
A parancs segítségével olvashatunk a kiválasztott memóriából a címszámláló által meghatározott címről. Az olvasási művelet végrehajtása után a címszámláló értéke a korábban kiadott *Entry Mode Set* parancs I/D bitjének megfelelően módosul.

5.2.4 Az LCD kijelző inicializálása

Mivel az alapértelmezett adatbusz szélesség 8 bit és a LOGSYS bővítőcsatlakozón az adatbusz összes bitje rendelkezésre áll, ezért a foglaltság jelzés (BF) kezdettől fogva lekérdezhető. A kétsoros üzemmód inicializálásához a következő műveleteket kell végrehajtani a bekapcsolás után:

- Várakozzunk az LCD kijelző inicializálásának befejeződéséig, az inicializálás alatt a BF foglaltság jelző bit értéke 1 (15 ms).
- Adjuk ki a *Function Set* parancsot: DL=1, N=1, F=0 (4,1 ms).
- Adjuk ki a *Clear Display* parancsot (1,53 ms).
- Adjuk ki az *Entry Mode Set* parancsot: I/D=1, SH=0 (39 μ s).

5.2.5 Kapcsolási rajz

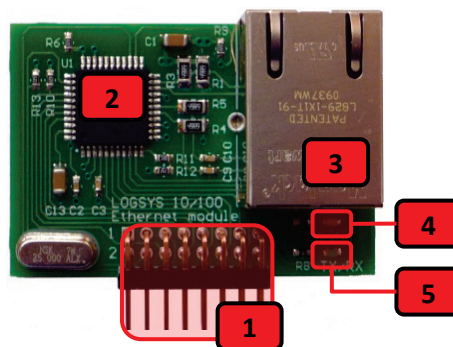


5.3 LOGSYS 10/100 Ethernet modul

A modul lehetővé teszi, hogy a LOGSYS FPGA kártyákat UTP kábellel Ethernet hálózathoz csatlakoztassuk. A modul felépítését az 5-12. ábra szemlélteti. A bővítőcsatlakozó lábkiosztását az 5-17. táblázat mutatja.

A modul főbb jellemzői:

- Microchip ENC424J600 Ethernet vezérlő áramkör
(az eszköz adatlapja letölthető a gyártó honlapjáról: <http://www.microchip.com>)
 - Integrált MAC és 10/100 Base-T PHY
 - Automatikus polaritás detektálás és korrekció
 - Beépített 24 KB adási/vételi csomag puffer
 - Adatok másolása a csomag pufferen belül DMA-val
 - Hardveres IP és TCP ellenőrző összeg számítás
 - Hardveres titkosító motor (RSA, AES, MD5, SHA-1)
- 8 bites multiplexelt párhuzamos kommunikációs interfész
- RJ45 (8P8C) aljzat az Ethernet kábel csatlakoztatásához



5-12. ábra: A LOGSYS 10/100 Ethernet modul felépítése.

A modul felépítése:

1. LOGSYS bővítőcsatlakozó
2. Microchip ENC424J600 Ethernet vezérlő áramkör
3. RJ45 (8P8C) aljzat
4. A kapcsolatot jelző sárga LED
5. Az adást/vételt jelző zöld LED

5-17. táblázat: A bővítőcsatlakozó lábkiosztása.

Láb	Típus ³	Funkció
1	PWR	GND
2		nincs bekötve
3	PWR	3,3V tápfeszültség
4	I	Írás/olvasás engedélyező jel
5	I	Írás/olvasás kiválasztás, AD8
6	I	Cím latch engedélyező jel
7	I/O	Címbusz/adatbusz (0. bit)
8	I/O	Címbusz/adatbusz (1. bit)

Láb	Típus ³	Funkció
9	I/O	Címbusz/adatbusz (2. bit)
10	I/O	Címbusz/adatbusz (3. bit)
11	I/O	Címbusz/adatbusz (4. bit)
12	I/O	Címbusz/adatbusz (5. bit)
13	I/O	Címbusz/adatbusz (6. bit)
14	I/O	Címbusz/adatbusz (7. bit)
15	O	Megszakításkérés (IRQ)
16		nincs bekötve

³ PWR: tápellátás, I: bemenet, O: kimenet, I/O: kétirányú

Például: I (bemenet) esetén az FPGA számára az adott láb kimenet, a modul számára bemenet.

5.3.1 Kommunikációs interfész

A modulon lévő ENC424J600 típusú áramkör többféle kommunikációs interfésszel is rendelkezik, ezek közül a modul a párhuzamos interfész MODE 6 nevű üzemmódját használja. Ebben az üzemmódban a címbusz és az adatbusz részlegesen multiplexelt és az adatszélesség 8 bites. A párhuzamos interfész a MODE 6 üzemmód esetén az alábbi jelekből tevődik össze:

- **CS:** magas-aktív kiválasztó jel (chip-select)
- **AL:** magas-aktív cím latch engedélyező jel
- **R/W:** írás (0) / olvasás (1) kiválasztó jel
- **EN:** magas-aktív engedélyező jel az írási/olvasási műveletekhez
- **AD[7:0]:** 8 bites multiplexelt adat- és címbusz (a címbusz alsó 8 bitje)
- **AD[14:8]:** a címbusz felső 7 bitje

A fenti listában összesen 19 jel található, azonban a LOGSYS bővítőcsatlakozó a Spartan-3E FPGA kártya esetén csak 11, a többi kártyák esetén pedig csak 13 kimenetként is használható kivezetéssel rendelkezik. Az AD[14:9] címbitek (a felső 6 címbit) tápfeszültségre kötésével csökkenthető a kivezetésre kerülő címbitek száma, de ekkor közvetlenül csak a regiszterek érhetőek el az ENC424J600 címtartományának felső 256 bájtnál (5-18. táblázat), az SRAM puffer elérése az indirekt cím- és adatregiszterek használatával lehetséges. A CS kiválasztó jel használata sem kötelező, ez is fixen a tápfeszültségre köthető. Az AD[8] címbit csak a cím latch írásakor van felhasználva, az R/W jel pedig csak az adatátvitelkor, ezért ez a két jel is összevonható. A modul által használt módosított interfész tehát a következő jelekből áll:

- **AL:** magas-aktív cím latch engedélyező jel
- **RNW/AD[8]:** közösített írás/olvasás kiválasztó jel és AD[8] címbit
- **EN:** magas-aktív engedélyező jel az írási/olvasási műveletekhez
- **AD[7:0]:** 8 bites multiplexelt adat- és címbusz (a címbusz alsó 8 bitje)

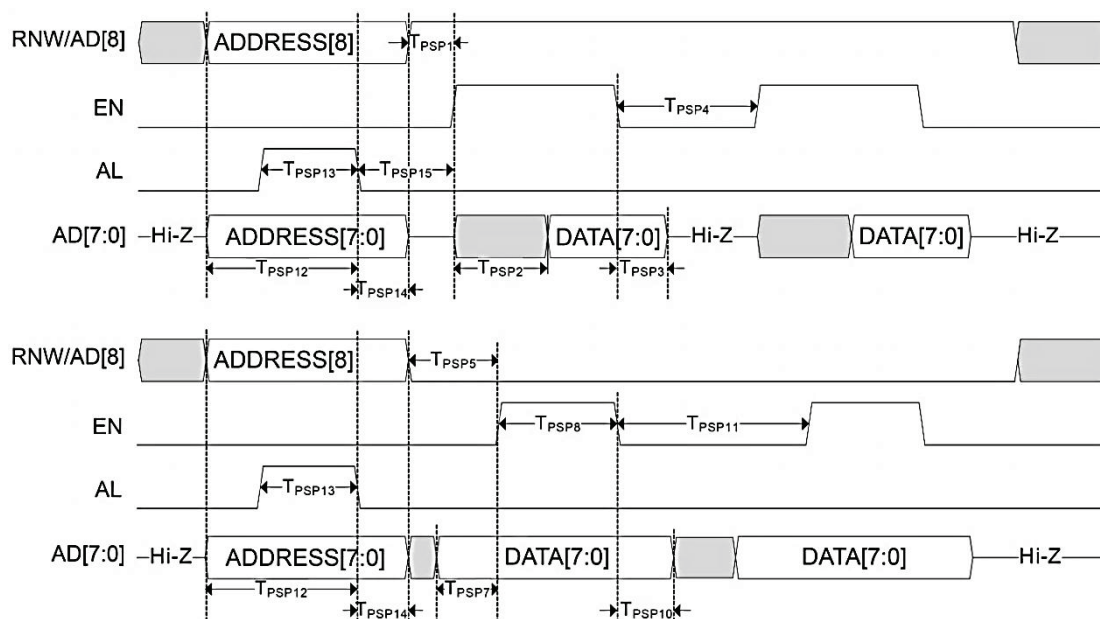
5-18. táblázat: Az ENC424J600 Ethernet vezérlő belső címiosztása.

24 kByte SRAM puffer	0x0000 0x5FFF
Nem használt terület	0x6000 0x77FF
Adatok a hardveres titkosító motor számára (ez a terület csak a belső DMA-val érhető el)	0x7800 0x7C4F
Nem használt terület	0x7C50 0x7DFF
Speciális funkció regiszterek (SFR)	0x7E00 0x7EFF
SFR bit beállító regiszterek	0x7F00 0x7F7F
SFR bit törlő regiszterek	0x7F80 0x7FFF

A modul kommunikációs interfészének idődiagramja az 5-10. ábrán látható, az időzítési paramétereket az 5-19. táblázat tartalmazza. Ha egymás után ugyanazt a regisztert szeretnénk elérni, akkor a cím latch értékét nem kell frissíteni, tehát ismételt cím kiadásra nincs szükség. Ugyanez vonatkozik a 24 kByte méretű belső SRAM puffer indirekt elérésére is, mivel az adott

indirekt adatregiszter írásakor és olvasásakor a megfelelő indirekt címregiszter értéke automatikusan növekszik.

Az ENC424J600 áramkör SFR regiszteri 16 bitesek. A 8 bites kommunikációs interfész miatt a regisztereknek egy írási/olvasási művelettel csak az alsó vagy felső bájtja érhető el, a teljes tartalom eléréséhez két írási/olvasási műveletre van szükség. Írás esetén a 8 bites adat azonnal beíródik az adott regiszter alsó vagy felső bájtjába, tehát nincs szükség két írási műveletre, ha egy regiszternek csak az alsó vagy csak a felső bájtját szeretnénk írni. A regiszterek elrendezése little-endian formátum szerinti, azaz az alsó bájt mindig az alacsonyabb címről érhető el. A 16 bites regiszterek ellenére a belső SRAM puffer elérése mindig bájtosan történik, az indirekt adatregiszterek alsó bájtját kell csak írni vagy olvasni, a felső bájt olvasása mindig 0x00 értéket ad vissza.



5-13. ábra: Az olvasási (felső) és az írási (alsó) művelet idődiagramja.

5-19. táblázat: A kommunikációs interfész időzítési paramétereit.

Szimbólum	Paraméter	Min.	Tipikus	Max.	Egység
T_{PSP1}	RNW jel előkészítési ideje olvasáskor	1	-	-	ns
T_{PSP2}	EN felfutó élétől az érvényes adatig eltelt idő	-	-	75	ns
T_{PSP3}	Adat tartási idő olvasáskor	0	-	3	ns
T_{PSP4}	EN alacsony szintjének időtartama olvasáskor	4,5	-	-	ns
T_{PSP5}	RNW jel előkészítési ideje íráskor	3,5	-	-	ns
T_{PSP7}	Adat előkészítési idő	3,5	-	-	ns
T_{PSP8}	EN magas szintjének időtartama íráskor	6,5	-	-	ns
T_{PSP10}	Adat tartási idő íráskor	1	-	-	ns
T_{PSP11}	EN alacsony szintjének időtartama SFR íráskor	4,5	-	-	ns
	EN alacsony szintjének időtartama SRAM íráskor	40	-	-	ns
T_{PSP12}	Cím előkészítési idő	6,5	-	-	ns
T_{PSP13}	AL magas szintjének időtartama	6,5	-	-	ns
T_{PSP14}	Cím tartási idő	1	-	-	ns
T_{PSP15}	AL alacsony szintjének időtartama	4	-	-	ns

5.3.2 A modul használata

Az alábbiakban röviden ismertetésre kerül az ENC424J600 Ethernet vezérlő áramkör inicializálása, valamint az Ethernet csomagok küldése és fogadása. A leírás nem tér ki olyan ritkábban használt dolgokra, mint például a csomagszűrés vagy az átvitelvezérlés konfigurálása. Az itt említett regiszterek részletes leírása az eszköz adatlapjában megtalálható.

Az áramkör inicializálása előtt szükséges annak alaphelyzetbe állítása, melynek végrehajtásához az alábbi lépések szükségesek:

- Várakozzunk, amíg a párhuzamos kommunikációs interfész nem használható. Írjunk az EUDAST regiszterbe 0x1234 értéket és ezt addig ismételjük, amíg a visszaolvasott adat 0x1234 nem lesz.
- Várakozzunk, amíg az ESTAT regiszter CLKRDY bitje 1 nem lesz.
- Adjuk ki a reset parancsot az ECON2 regiszter ETHRST bitjének beállításával.
- Várakozzunk legalább 25 µs ideig, amíg a reset művelet be nem fejeződik.
- Ellenőrizzük az EUDAST regiszter értékét, amely a reset után 0x0000 kell, hogy legyen.
- Várakozzunk legalább 256 µs ideig, ezután a PHY regiszterek elérhetők.

Az áramkör alaphelyzetbe állítása után a következő lépés a vételi puffer konfigurálása, melyet a csomagok vételének engedélyezése előtt mindenképpen meg kell tenni. Ehhez írjuk be a vételi puffer kezdőcímét az ERXST regiszterbe. A kezdőcím csak páros értékű lehet. A vételi puffer utolsó szavának címét az ERXTAIL regiszter tárolja, mely alapértelmezésben az SRAM puffer végére mutat (0x5FFE). Az Ethernet csomagok vételéhez szükség van egy változóra (NextPacketPointer), amely a következő csomag kezdőcímét tárolja a vételi pufferben. Inicializáljuk ezt a változót a vételi puffer kezdőcímével. Adási puffer külön nincs definiálva, mert az alkalmazás bármely nem használt SRAM területre beírhatja a küldendő csomagot.

A vételi puffer konfigurálása után inicializálni kell néhány MAC regisztert:

- Minden ENC424J600 áramkör egyedi MAC címmel rendelkezik, amely kiolvasható az MAADDR1, az MAADDR2 és az MAADDR3 regiszterekből. Ha más MAC címet szeretnénk használni, akkor ezekbe a regiszterekbe beírható az új MAC cím.
- Állítsuk be a maximális Ethernet kerethosszat az MAMXFL regiszterben. Ez tipikusan egy 64 bájt és 1518 bájt közötti érték.
- Az auto-negotiation használatához írjunk az Ethernet PHY PHANA regiszterébe 0x05E1 értéket. A PHY regiszterek közvetlenül nem érhetőek el, írásuk az alábbi módon lehetséges:
 - Írjuk be a MIREGADR regiszter alsó bájtjába a PHY regiszter címét, a felső bájt értéke 0x01 kell, hogy legyen.
 - Írjuk be az MIWR regiszterbe a PHY regiszterbe írandó adatot, először az alsó bájtot, majd a felső bájtot (ez utóbbi indítja el a PHY regiszter írást).
 - Várakozzunk, amíg a MISTAT regiszter BUSY bitje nem törlődik.
- Engedélyezzük az Ethernet csomagok vételét az ECON1 regiszter RXEN bitjének beállításával.

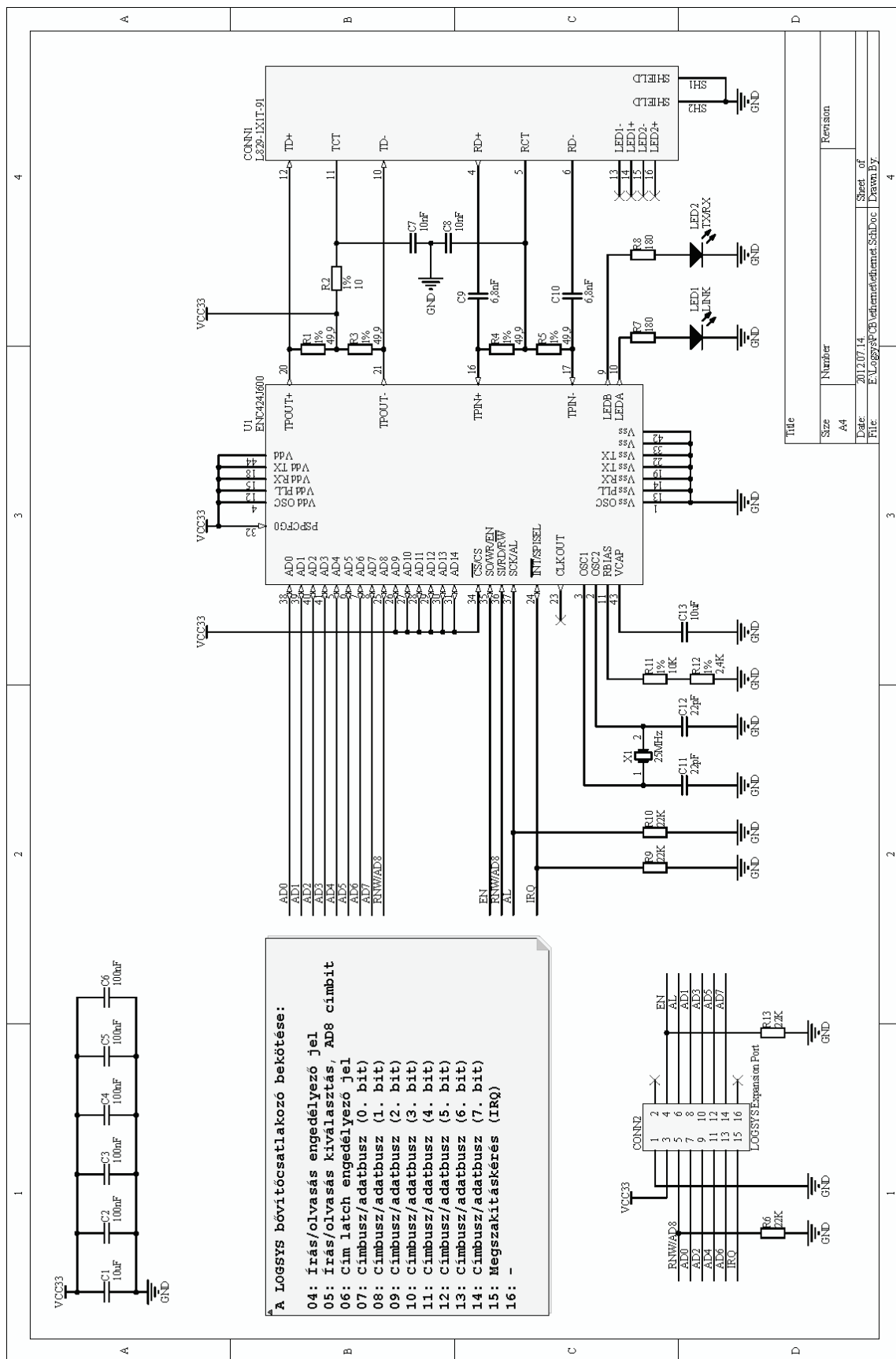
Az Ethernet csomagot annak elküldése előtt be kell másolni az SRAM puffer nem használt részére. Ehhez állítsuk be az SRAM kezdőcímet az EGPWRPT regiszterbe, majd az EGPDATA regiszterbe írjuk be az adatokat. Az EGPWRPT mutató az EGPDATA minden írásakor eggyel növekszik. A csomag elküldéséhez állítsuk be az ETXST regisztert a csomag kezdőcímére, az

ETXLEN regisztert a csomag méretére, majd az ECON1 regiszter TXRTS bitjének beállításával elindítható a küldés. Ha az Ethernet csomag el lett küldve, akkor az ECON1 regiszter TXRTS bitje törlődik. Ezen bit értékének vizsgálatával várakozhatunk, amíg be nem fejeződik az adatküldés.

Az Ethernet csomagok vételéhez az alábbi lépéseket kell végrehajtani:

- Ellenőrizzük az EIR regiszterben a PKTIF bit értékét. Ha ez a bit 0, akkor a vételi pufferben nincs adat.
- Írjuk a következő csomag kezdőcímét tároló változó (NextPacketPointer) értékét az ERXRDPT regiszterbe. Az adatokat az ERXDATA regiszter olvasásával tudjuk kiolvasni az SRAM-ból. Az ERXRDPT mutató az ERXDATA minden olvasásakor eggyel növekszik.
- A csomag első két bájtja tartalmazza a következő csomag kezdőcímét a vételi pufferben, a kiolvasott értékkel frissítsük a következő csomag kezdőcímét tároló változót (NextPacketPointer). Az új kezdőcím little-endian formátumú, tehát annak alsó bájtja található a kisebb címen.
- A csomag következő 6 bájtja tartalmazza a vételi státusz vektort (RSV). A vételi státusz vektor első két bájtja tartalmazza az Ethernet keret méretét bájtokban. Ez little-endian formátumú, tehát annak alsó bájtja található a kisebb címen. A vételi státusz vektor harmadik bájtjának MSb-je jelzi a hibamentes vételt.
- Olvassuk be az Ethernet keretet, melynek méretét a vételi státusz vektor tartalmazza.
- Írjuk be a vételi puffer végének új címét az ERXTAIL regiszterbe. Az új cím a következő csomag címét tároló változó értéke kettővel csökkentve (NextPacketPointer-2).
- Nyugtázzuk a csomag vételét az ECON1 regiszter PKTDEC bitjének beállításával.

5.3.3 Kapcsolási rajz

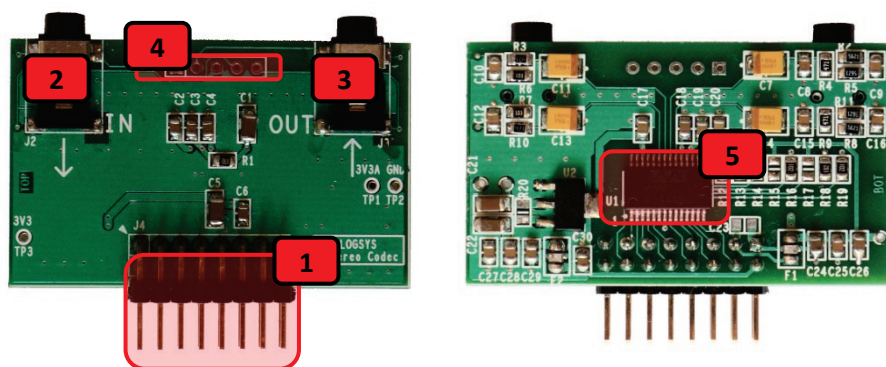


5.4 LOGSYS sztereó CODEC modul

A modul egy könnyen kezelhető két csatornás CODEC, melynek feladata, hogy minél egyszerűbb formában tudjunk hangfrekvenciás tartományban lévő analóg jeleket digitalizálni, valamint analóg formába visszaalakítani. A modul felépítését az 5-14. ábra szemlélteti, a bővítőcsatlakozó lábkiosztása az 5-20. táblázatban látható.

A modul főbb jellemzői:

- Wolfson WM8569 stereo CODEC chip
(adatlap: <http://www.wolfsonmicro.com/products/codecs/WM8569>)
 - 24 bites felbontás
 - 2 csatornás, 32-192 kHz tartományban működő, digitális-analóg konverter (DAC)
 - 2 csatornás, 32-96 kHz tartományban működő, analóg-digitális konverter (ADC)
 - Előre beállított interfész paraméterek (opcionális 3 vezetékes vezérlő busz)
- Sztéreo-jack analóg be- és kimenet.



5-14. ábra: A LOGSYS sztereó CODEC modul felépítése.

A modul felépítése:

1. LOGSYS bővítőcsatlakozó
2. Analóg (hang) bemenet, szabványos 3.5 mm jack csatlakozó
3. Analóg (hang) kimenet, szabványos 3.5 mm jack csatlakozó
4. J1 csatlakozó (tesztpontok)
5. Wolfson WM8569 sztereó CODEC chip

5-20. táblázat: A bővítőcsatlakozó lábkiosztása.

Láb	Típus ⁴	Funkció
1	PWR	GND
2	PWR	5V tápfeszültség
3	PWR	3,3V tápfeszültség
4	I/NA ⁵	SDIN – Config. adat bemenet
5	I/NA ⁵	SCLK – Config. órajel bemenet
6	I/NA ⁵	CSB – Config. kiválasztójel
7	O	ADC_DOUT – ADC adatkimenet
8	I	DAC_DIN – DAC adatbemenet

Láb	Típus ⁴	Funkció
9	I	DAC_LRCLK - DAC mintavételi f.
10	I	ADC_LRCLK - ADC mintavételi f.
11	I	DAC_BCLK – DAC soros if. órajel
12	I	ADC_BCLK – ADC soros if. órajel
13	I	DAC_MCLK – DAC belső órajel
14	I	ADC_MCLK – ADC belső órajel
15		Nincs bekötve
16		Nincs bekötve

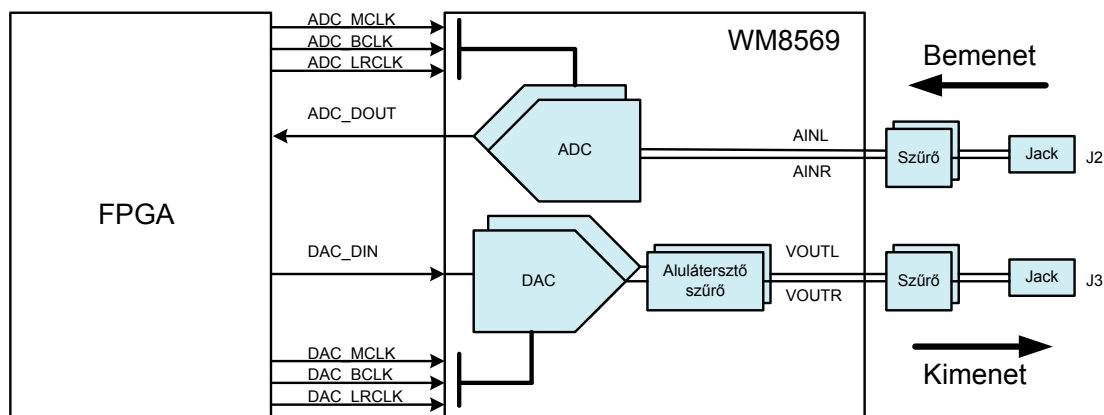
⁴ PWR: tápellátás, I: bemenet, O: kimenet

Például: I (bemenet) esetén az FPGA számára az adott láb kimenet, a modul számára bemenet.

⁵ Az SDIN/SCLK/CSB lábak a modulon jelenleg nem használtak, az opcionális konfigurációs interfész kialakítását teszik lehetővé.

5.4.1 A modul működése

A modul gyakorlatilag egyetlen WM8569 CODEC chip köré épül, a blokkvázlata az 5-15. ábrán látható. A tervezés során fő szempont volt a minél egyszerűbb kezelés az esetleges komplex konfigurációs interfész megtartása mellett.



5-15. ábra: A modul blokkvázlata.

A CODEC adatlapja alapján a be- és kimeneti jelszintek effektív értéke az analóg tápfeszültség ($AVDD - 3,3\text{ V}$) értékével megegyező referenciafeszültség ($DACVREFP$) $1/5$ -ével egyezik meg. Ez 660 mV (RMS) , azaz szinuszos vezérlés esetén $1866\text{ mV-os } V_{pp}$ feszültségszintnek felel meg. A be- és kimenetek AC csatoltak!

Az analóg be- és kimenetek a jack csatlakozón túlmenően a J1 jelű, $2,54\text{ mm}$ raszter osztású csatlakozón is elérhetőek. A J1 csatlakozó kiosztása az 5-21. táblázatban található.

5-21. táblázat: A J1 csatlakozó (mérőpontok) kiosztása

Láb	Név	Funkció
1	VIN1	Bemenő jel bal csatorna
2	VOUT1	Kimenő jel bal csatorna
3	GND	Föld
4	VIN2	Bemenő jel jobb csatorna
5	VOUT2	Kimenő jel jobb csatorna

5.4.2 A CODEC konfigurációja

A CODEC nagy előnye, hogy használata előtt nincs szükség számtalan belső regiszterének beállítására, hanem bekapcsolás után, a megfelelő órajel bemenetek biztosítása mellett azonnal működőképes (a többi hasonló chiphez képest ez jelentős könnyebbség). Ezt a továbbiakban Hardware Control Mode-nak nevezzük.

Abban az esetben, ha valamely komplexebb funkciót szeretnénk elérni vagy egyedi vezérlést szeretnénk kipróbálni, az SDI, SCLK és CSB jelek egy soros konfigurációs interfészhez csatlakozhatnak, amelyen keresztül minden paraméter beállítható. Ezt a továbbiakban Software Control Mode-nak nevezzük. A kártya alap konfigurációja mellett azonban ezeket a jeleket nem kell/lehet használni. A fenti lábakat nem kell az FPGA tervben definiálni.

A Hardware/Software Control Mode között egy beültetési variációval válthatunk, amely a chip MODE lábának állapotát módosítja:

- Amennyiben az R13 ellenállás van beültetve (MODE=1, alapértelmezett), a Hardware Control Mode van kiválasztva. Ebben az esetben az SDI, SCLK és CSB lábakat nem szabad vezérelni.
- Amennyiben az R12 ellenállás van beültetve (MODE=0), a Software Control Mode van kiválasztva. Ebben az esetben az SDI, SCLK és CSB vonalak a soros konfigurációs interfészt vezérik.

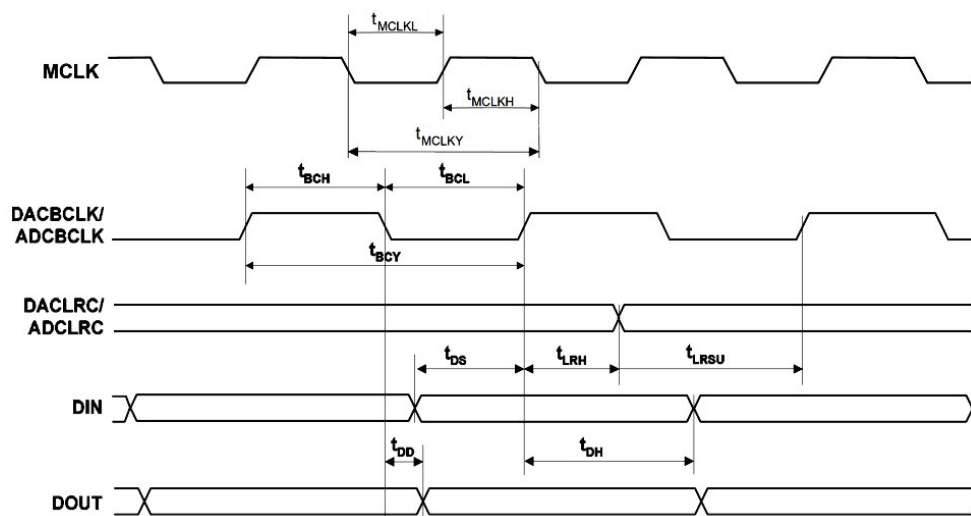
Hardware Control Mode esetén a chip néhány paramétere konfigurálható csak. Ezeket az IC bizonyos lábainak alacsony vagy magas állapotba történő állításával érhetjük el. A NYÁK úgy lett kialakítva, hogy bizonyos változtatásokat ellenállások cseréjével is megtehesünk. Az 5-22. táblázat az alapbeállításokat tartalmazza.

5-22. táblázat: A modul beállításai

Funkció	Állapot	Beállító ellenállás	Leírás
Interfész típusa	-	-	Hardware Control Mode esetén a chip mindig slave módban működik, azaz az ADC_BCLK, DAC_BCLK, ADC_LRCLK és DAC_LRCLK jelek a CODEC számára bemenetek.
Interfész formátum IDF/IWL láb	logikai 00	R14: beültetve R15: üres R16: beültetve	00: 24 bites, jobbra igazított (alapértelmezett) 01: 20 bites, jobbra igazított 10: 16 bites, I ² S 11: 24 bites, I ² S
De-emphasis DM láb	logikai 0	R18: beültetve R17: üres	0: De-emphasis kikapcsolva (alapértelmezett) 1: De-emphasis bekapcsolva
Kimenet tiltás MUTE láb	logikai 0	R19: beültetve	0: kimeneti DAC engedélyezve van (alapértelmezett) 1: kimeneti DAC tiltva van

5.4.3 Időzítési paraméterek

A WM8569 áramkör slave módú digitális interfészének idődiagramja az 5-16. ábrán látható, az időzítési paramétereket az 5-23. táblázat tartalmazza.



5-16. ábra: A WM8569 digitális interfészének idődiagramja (slave mód)

5-23. táblázat: A WM8569 digitális interfészének időzítési paraméterei.

Szimbólum	Paraméter	Min.	Tipikus	Max.	Egység
t_{MCLK}/t_{MCLKH}	MCLK órajel alacsony/magas szintje	11	-	-	ns
t_{MCLKY}	MCLK órajel periódusideje	28	-	-	ns
	MCLK órajel kitöltési tényezője	40	-	60	%
t_{BCL}/t_{BCH}	BCLK órajel alacsony/magas szintje	20	-	-	ns
t_{BCY}	BCLK órajel periódusideje	50	-	-	ns
t_{LRSU}	LRCLK előkészítési idő (BCLK-hoz képest)	10	-	-	ns
t_{LRH}	LRCLK tartási idő (BCLK-hoz képest)	10	-	-	ns
t_{DS}	DIN előkészítési idő (BCLK-hoz képest)	10	-	-	ns
t_{DH}	DIN tartási idő (BCLK-hoz képest)	10	-	-	ns
t_{DD}	DOUT késleltetési idő (BCLK-hoz képest)	0	-	10	ns

5.4.4 Az MCLK és az LRCLK kapcsolata

A digitális hangátviteli rendszerek esetén az ADC és DAC egységeket egyetlen központi órajelhez szokták szinkronizálni. Ezen közös órajel használata biztosítja, hogy az egyes alegységek szinkron járnak. A rendszer központi órajele az MCLK, amely frekvenciájából származtatható az összes többi órajel frekvencia is (az MCLK minden esetben bemenete a CODEC-nak). A CODEC az MCLK segítségével vezérli a belső digitális szűrőit, valamint a szigma-delta modulátort.

A másik fontos órajel az LRCLK, amelynek frekvenciája megegyezik a mintavételi frekvenciával (f_s). Az alkalmazott CODEC esetén a mintavételi frekvencia 32 kHz, 44.1 kHz, 48 kHz és 96 kHz lehet, illetve a DAC járhat 192 kHz-es frekvencián is. Az LRCLK és az MCLK viszonyáról elmondható, hogy csak az alábbi jól meghatározott arányok engedhetők meg, amelyet az 5-24. táblázat foglal össze.

5-24. táblázat: Az MCLK és a mintavételi frekvencia kapcsolata

LRCLK frekvencia / Mintavételi frekvencia (f_s)	MCLK órajel frekvenciája (MHz)					
	128 f_s^6 (DAC)	192 f_s^6 (DAC)	256 f_s	384 f_s	512 f_s	768 f_s
32 kHz	4.096	6.144	8.192	12.288	16.384	24.576
44.1 kHz	5.6448	8.467	11.2896	16.9340	22.5792	33.8688
48 kHz	6.144	9.216	12.288	18.432	24.576	36.864
96 kHz	12.288	18.432	24.576	36.864	N/A ⁷	N/A ⁷
192 kHz (DAC)	24.576	36.864	N/A ⁷	N/A ⁷	N/A ⁷	N/A ⁷

A CODEC az MCLK és az LRCLK arányát automatikusan detektálja, MCLK órajelekben mérve megsámolja az LRCLK periódusidejét. Abban az esetben, ha a mért periódus idő ± 32 órajelnél többel eltér a táblázatban rögzítettől, a rendszer automatikusan a 768 f_s üzemmódot választja ki.

Az MCLK és LRCLK órajeleknek egymással szinkron kell működniük, azonban az MCLK és LRCLK órajelek egymáshoz viszonyított fázisára (és a jitter-re) a CODEC nem érzékeny.

Fontos megemlíteni, hogy mivel az ADC és a DAC rész teljesen különválasztott, külön órajel bemenetekkel rendelkezik, elképzelhető olyan alkalmazás, ahol az ADC és DAC eltérő mintavételi frekvencián dolgoznak.

⁶ A 128 f_s és a 192 f_s üzemmód csak a DAC esetén állítható be.

⁷ N/A: az üzemmód nem elérhető

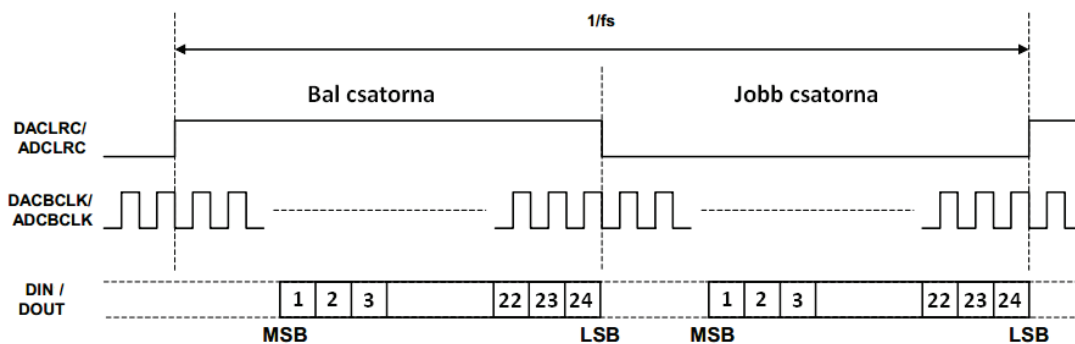
5.4.5 Soros interfész

Mint korábban láttuk, a modul (a Hardware Control Mode-nak megfelelő alapkonfiguráció esetén) 24 bites, jobbra igazított adatformátummal dolgozik (5-17. ábra). Az adatátvitelt a BCLK órajel időzíti, amely a soros interfész órajele.

Az ADC_DOUT adatkimenet mindig a BCLK lefutó éle után vált, ezért mintavételezni a BCLK felfutó élénél érdemes. A legkisebb helyiértékű bit (LSB) mindig az LRCLK váltása előtt jelenik meg. A legmagasabb helyiértékű bit (MSB) pedig 24 órajellel korábban.

A DAC_DIN adatbemenetet a chip a BCLK felfutó élénél mintavételezi. Az LSB ugyancsak az LRCLK váltása előtti utolsó bit, az MSB pedig 24 órajellel korábbi.

Az LRCLK értéke a bal csatornára magas, a jobbra alacsony.



5-17. ábra: A 24 bites, jobbra igazított üzemmód.

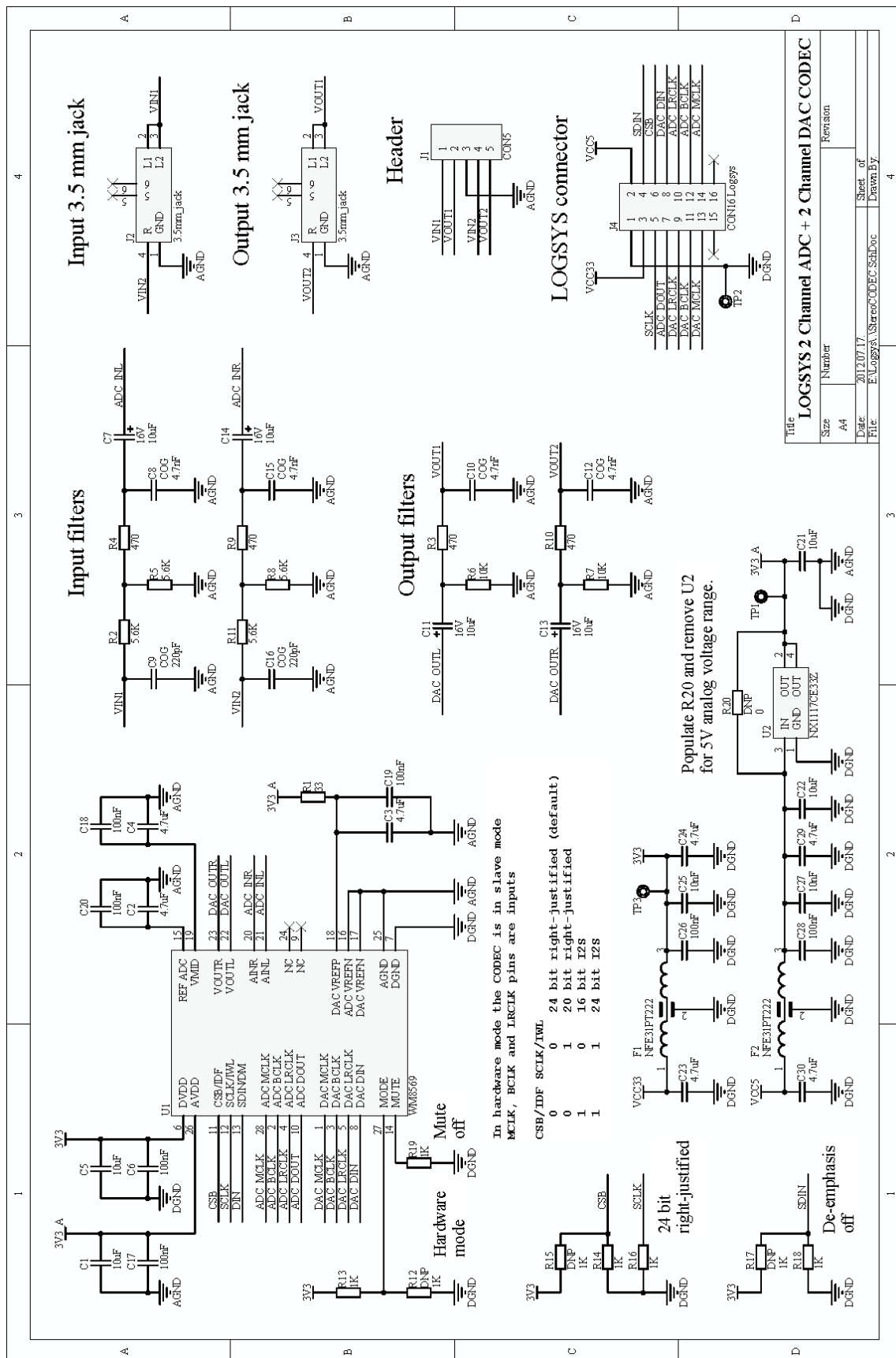
Jól látható, hogy a specifikáció nem tartalmazza a BCLK konkrét frekvenciáját. Ahhoz, hogy a 2x24 bit átvihető legyen, legalább $48 \times \text{LRCLK}$ (f_s) frekvenciára kell választani. A gyakorlatban inkább $64 \times \text{LRCLK}$ (f_s) frekvenciát szoktunk alkalmazni. Ebben az esetben 8 órajel/csatorna bitidő alatt a ADC_DIN/DAC_DOUT adatok értéke határozatlan (bemenet esetén), illetve közömbös (kimenet esetén) az MSB bitet megelőzően.

Mint azt korábban említettük, az MCLK és LRCLK órajeleknek egymással szinkron kell működniük, az LRCLK viszont a BCLK jelhez is szinkron, következésképp mindhárom órajelet (MCLK, BCLK és LRCLK) szinkronra kell terveznünk!

Egy konkrét javasolt beállítás:

- LRCLK, mintavételi frekvencia (f_s): 48 kHz
- BCLK: $64 \times \text{LRCLK} = 3.072 \text{ MHz}$
- MCLK: $256 \times \text{LRCLK} = 12.288 \text{ MHz}$

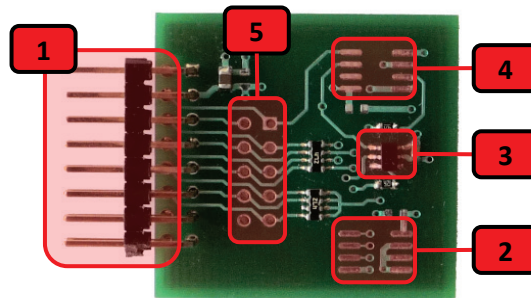
5.4.6 Kapcsolási rajz



5.5 LOGSYS hőmérő és EEPROM modul

A bővítő modul kifejlesztésének fő célja az volt, hogy a hallgatókat megismertessük az alapvető, beágyazott rendszerekben napi szinten használt I²C és SPI soros busz szabványokkal. A modul felépítését az 5-18. ábra szemlélteti, a bővítőcsatlakozó lábkiosztása az 5-25. táblázatban látható. A modulon három integrált áramkör található:

- LM75 I²C buszos hőmérő chip
- TMP121 SPI buszos hőmérő chip
- M95080 SPI buszos EEPROM chip



5-18. ábra: A LOGSYS hőmérő és EEPROM modul.

A modul felépítése:

1. LOGSYS bővítőcsatlakozó
2. LM75 I²C buszos hőmérő chip (nem mindig van beültetve)
3. TMP121 SPI buszos hőmérő chip
4. M95080 SPI buszos EEPROM chip (nem mindig van beültetve)
5. CON2 csatlakozó (tesztpontok, lásd az 5-26. táblázatot)

5-25. táblázat: A bővítőcsatlakozó lábkiosztása.

Láb	Típus ⁸	Funkció
1	PWR	GND (föld)
2	PWR	Nem használt (5V)
3	PWR	3,3V tápfeszültség
4	I	SPI MOSI
5	I	SPI SCK
6	I	SPI nCS-0 (M95080 EEPROM)
7	O	SPI MISO-0 (M95080 EEPROM)
8	I	SPI nCS-1 (TMP121 hőmérő)

Láb	Típus ⁸	Funkció
9	O	SPI MISO-1 (TMP121 hőmérő)
10	OC	I ² C SDA
11	OC	I ² C SCL
12	OC	Hőmérséklet hiba kimenet (OS)
13		Nincs bekötve
14		Nincs bekötve
15		Nincs bekötve
16		Nincs bekötve

5-26. táblázat: A CON2 csatlakozó lábkiosztása (tesztpontok)

Láb	Funkció
1	SPI MOSI
2	SPI SCK
3	SPI nCS-0 (M95080 EEPROM)
4	SPI MISO-0 (M95080 EEPROM)
5	SPI nCS-1 (TMP121 hőmérő)

Láb	Funkció
6	SPI MISO-1 (TMP121 hőmérő)
7	I ² C SDA
8	I ² C SCL
9	Hőmérséklet hiba kimenet (OS)
10	GND (föld)

⁸ PWR: tápellátás, I: bemenet, O: kimenet, OC: nyitott kollektoros meghajtás

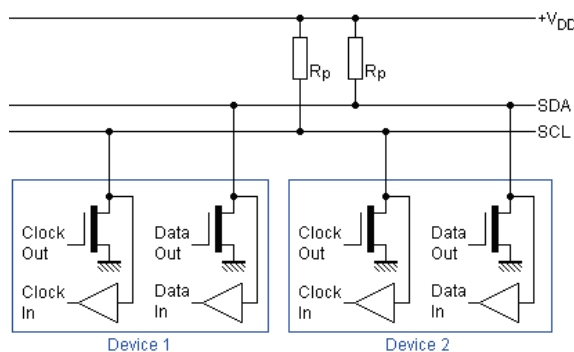
Például: I (bemenet) esetén az FPGA számára az adott láb kimenet, a modul számára bemenet.

5.5.1 LM75 – I²C buszos hőmérő

5.5.1.1 Az I²C busz általános ismertetése

Az I²C (vagy másképpen I2C, IIC) egy igen gyakran használt áramkörök közötti kommunikációs interfész. Alkalmazásával akár több master és több slave egységet is összekapcsolhatunk. A master egység vezérli a buszt és biztosítja az órajelet. A slave eszköz a master egység kérésére válaszol és adatot fogad a master egység felől vagy küld a master egység felé. Az I²C busz nagy előnye, hogy a kommunikáció megvalósításához összesen két vezetékre van szükség, hátránya a limitált átviteli sebesség. A beágyazott rendszerekben leggyakrabban alkalmazott verzió az 1.0, amely 400 kHz-es maximális órajel frekvenciát engedélyez.

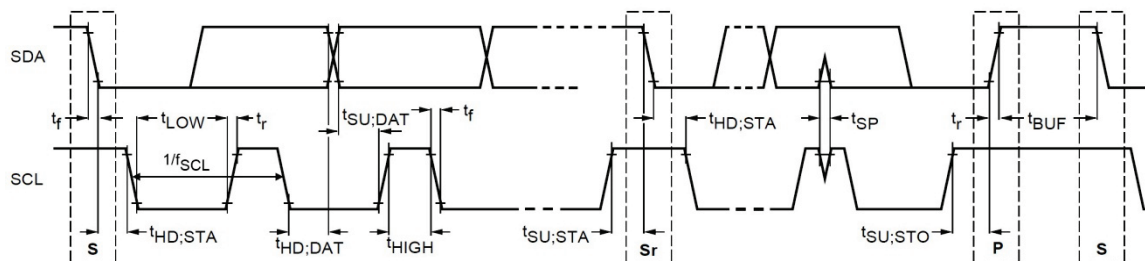
Az I²C busz fizikai interfésze két vezetékből áll, ezt szemlélteti az 5-19. ábra. Az SCL jel a szinkron működésű busz órajel vezetéke, az SDA jel pedig a kétirányú adatvonal. A szabvány szerint mindkét vezetéket nyitott kollektoros meghajtással kell vezérelni. Ez azt jelenti, hogy amennyiben a vezetékre logikai 0-át szeretnénk küldeni, a chipen belüli tranzisztort úgy vezéreljük, hogy az a vezetéket erősen nullába húzza le. Abban az esetben, ha a vezetékre logikai 1-et szeretnénk küldeni, a tranzisztort nem vezéreljük, így az R_p felhúzó ellenállás logikai magas szintet biztosít (feltéve, hogy a buszra kapcsolódó más eszközök közül semelyik nem húzza le a vezetéket logikai alacsony szintre).



5-19. ábra: Az I²C busz fizikai interfésze.

Mivel a logikai magas szintet a felhúzó ellenállás (általában 1 – 10 kΩ közötti érték) biztosítja, a csatlakoztatott eszközök és a busz kapacitása nagymértékben lassíthatja az átvitelt. A szabvány szerint a maximálisan 400 kHz frekvenciájú órajel használata esetén legfeljebb 400 pF kapacitás engedhető meg a buszon.

Az I²C buszon történő kommunikáció általános idődiagramja az 5-20. ábrán látható, az időzítési paramétereket az 5-27. táblázat tartalmazza.



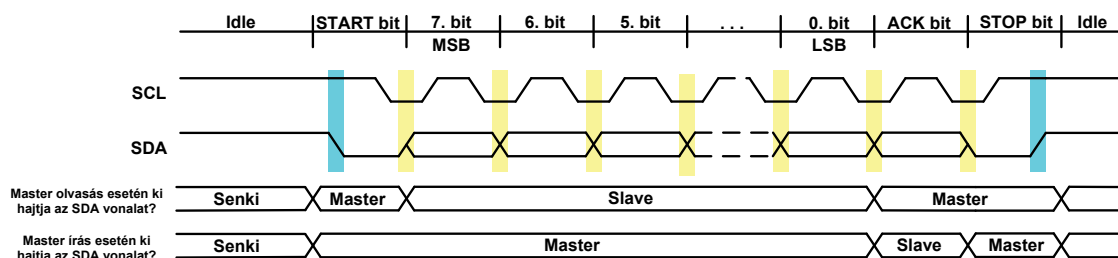
5-20. ábra: Az I²C busz időzítése.

5-27. táblázat: Az I²C busz időzítési paraméterei.

Paraméter	Szimbólum	Normál mód		Fast mód		Egység
		Min.	Max.	Min.	Max.	
SCL órajel frekvencia	f _{SCL}	0	100	0	400	kHz
(Ismételt) START feltétel tartási ideje	t _{HD;STA}	4,0	-	0,6	-	μs
SCL órajel alacsony szintjének ideje	t _{LOW}	4,7	-	1,3	-	μs
SCL órajel magas szintjének ideje	t _{HIGH}	4,0	-	0,6	-	μs
Ismételt START feltétel előkészítési ideje	t _{SU;STA}	4,7	-	0,6	-	μs
Adat tartási idő	t _{HD;DAT}	0	3,45	0	0,9	μs
Adat előkészítési idő	t _{SU;DAT}	250	-	100	-	ns
SDA/SCL felfutási idő	t _r	-	1000	20	300	ns
SDA/SCL lefutási idő	t _f	-	300	20	300	ns
STOP feltétel előkészítési ideje	t _{SU;STO}	4,0	-	0,6	-	μs
Szabad busz idő STOP és START feltétel között	t _{BUF}	4,7	-	1,3	-	μs
A tüskék megengedett pulzusszélessége	t _{SP}	-	-	0	50	ns

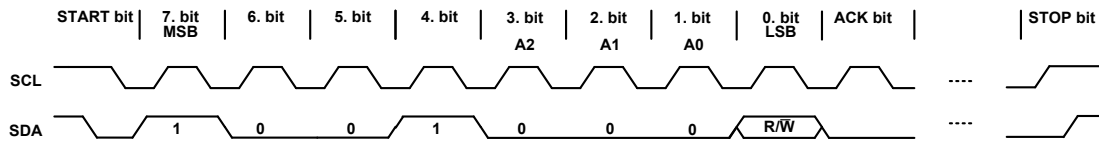
Az I²C adatátvitel az alábbi elemi átviteli blokkból épül fel, melyet az 5-21. ábra is szemléltet:

1. Nem használt állapotban (Idle) a busz egyik vezetéke sincs meghajtva, tehát logikai magas szinten vannak az R_p felhúzó ellenállások hatására.
2. Ha a master adatot szeretne küldeni, akkor egy speciális START feltételt (START bitet) ad ki: az SCL magas állapota alatt az SDA vonalon lefutó élet generál.
3. Ezek után következnek az adatbitek. Az SCL vonalon a master egy órajelet generál, az SDA vonalra pedig a megfelelő bit kerül, legelőször a legmagasabb helyiértékű bit (MSB). Fontos megjegyezni, hogy az SDA vezetéken jelváltás csak az SCL alacsony állapota alatt (ez az ábrán sárga színnel van jelölve) engedhető meg az adatbitek elküldése esetén.
 - a. Ha a master adatot ír, akkor adatküldés alatt az SDA vonalat a master hajtja meg.
 - b. Ha a master adatot olvas, akkor adatküldés alatt az SDA vonalat slave hajtja meg.
4. A nyolcadik adatbit átvitele után egy kilencedik, nyugtázó (acknowledge, ACK) bit átvitele történik meg, ehhez a master még egy órajelet generál az SCL vonalon.
 - a. Ha a master adatot ír, akkor az ACK alatt az SDA vonalat a slave hajtja meg
 - b. Ha a master adatot olvas, akkor az ACK alatt az SDA vonalat a master hajtja meg.
5. Amennyiben több mint egy bájtot viszünk át, a master további 8+1 órajelet generál, ebben az esetben a folyamat ismétlődik a 3. ponttól kezdődően. Ellenkező esetben az adatátvitel végén a master egy speciális STOP feltételt (STOP bitet) ad ki: az SCL magas állapota alatt az SDA vonalon felfutó élet generál. Ezután a busz nem használt (Idle) állapotba kerül.

5-21. ábra: Az I²C adatátvitel fázisai.

Általában elmondható, hogy az ACK bit értéke jelzi, hogy van-e szükség további bájt átvitelére. Ha az ACK bit értéke logikai 0, akkor az átvitel folytatódik, ha az ACK értéke logikai 1 (NACK) a megszólított eszköz nem válaszol, vagy nem akar több adatot átvinni.

A legelső átvitt bájt minden esetben egy speciális cím (address) bájt (5-22. ábra), amely segítségével megszólíthatunk egy slave eszközt és kijelölhetünk további kommunikációra. Így valósulhat meg, hogy egyetlen buszra akár több slave eszköz is felfűzhető. Fontos viszont, hogy minden slave eszköznek saját címmel kell rendelkeznie. Ahhoz, hogy azonos típusú eszközökből többet is használhassunk egyetlen buszon, a chipeket általában cím kiválasztó bemenetekkel is ellátják, így mindegyikhez külön-külön cím rendelhető hozzá.



5-22. ábra: Az I²C cím bájt formátuma.

Ha a master írni szeretne a kiválasztott slave-re, akkor az R/W bit értéke 1, ha olvasni szeretne, akkor pedig az R/W bit értéke 0. A cím bájt küldésekor az ACK bitnek kitüntetett szerepe van. A slave eszköz a nyugtázással tudja jelezni, hogy a buszra csatlakozott és képes az adat továbbítására. Amennyiben senki sem húzta le a buszt az ACK fázisban, akkor az ACK bit értéke 1 lesz az R_p felhúzó ellenállás miatt, így egyértelműen kiderül, hogy a megcímezett eszköz nem válaszol.

5.5.1.2 Az LM75 hőmérő chip használata

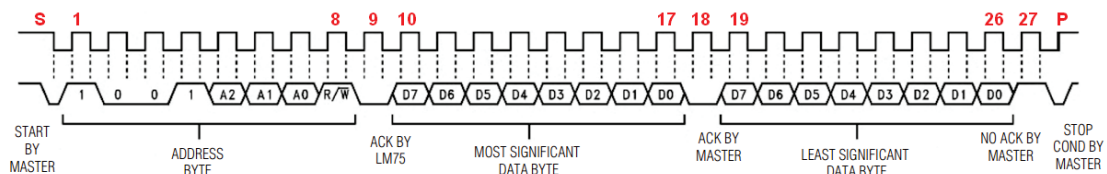
Az LM75 egy igen széles körben elterjedt I2C buszra csatlakoztatható, konfigurálható hőmérő chip. Az eszköz adatlapja letölthető a <http://datasheets.maxim-ic.com/en/ds/LM75.pdf> címről. Ebben az alfejezetben az LM75 áramkör elsődleges funkcióját, a hőmérséklet kiolvasását fogjuk bemutatni.

A jelen kialakításban egyetlen master és egyetlen slave eszköz kapcsolódik össze, így a master (FPGA) az SCL vonalat akár normál kimenetként is használhatja. Az SDA adatvezeték viszont minden esetben kétirányú, nyitott kollektoros meghajtást igényel. A modulon az LM75 A2-A0 címkiválasztó bemenetei földre vannak kötve, így az eszköz címe az I²C buszon 0x48, tehát olvasáskor a cím bájt értéke 0x91, íráskor pedig 0x90.

A mért hőmérséklet 0.5 °C-os felbontással (±2 °C pontossággal) egy belső 16 bites regiszterben tárolódik az alábbiak szerint: a felső 9 biten található a tárolt hőmérséklet 0.5 °C-os felbontással, kettes komplement számábrázolási formátumban, a regiszter alsó 7 bitje pedig nem használt (vegyük észre, hogy a felső bájt egyben a mért hőmérséklet érték 1 °C-os felbontással). Az adatbitek értelmezését és három minta hőmérséklet értéket az 5-28. táblázat tartalmazza. Az LM75 aktuális hőmérséklet regiszterének - bekapcsolás utáni - I²C buszon történő olvasási idődiagramja az 5-23. ábrán látható, az adatátvitel egyes fázisait részletesen az 5-29. táblázat tartalmazza.

5-28. táblázat: Az LM75 IC által használt számábrázolás.

	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	Egészrész								Tört-rész	Don't care						
+25.5 °C	0	0	0	1	1	0	0	1	1	X	X	X	X	X	X	X
+25 °C	0	0	0	1	1	0	0	1	0	X	X	X	X	X	X	X
-25 °C	1	1	1	0	0	1	1	1	0	X	X	X	X	X	X	X



5-23. ábra: Az aktuális hőmérséklet regiszter olvasása (bekapcsolás után).

5-29. táblázat: Az aktuális hőmérséklet regiszter olvasásának fázisai.

Órajel sorszáma	Az SDA vonalat meghajtja	Leírás
Start (S)	master (FPGA)	START feltétel generálása,
1-8	master (FPGA)	Cím bájt kiadása: b10010001 = 0x91 (olvasás).
9	slave (LM75)	Az LM75 ACK = 0 bittel jelzi, hogy készen áll az átvitelre.
10-17	slave (LM75)	A hőmérséklet adat felső bájtjának átvitele.
18	master (FPGA)	A master ACK=0 bittel jelzi, hogy további bájtot szeretne olvasni.
19-26	slave (LM75)	A hőmérséklet adat alsó bájtjának átvitele.
27	master (FPGA)	A master ACK=1 bittel jelzi, hogy nem olvas további bájtokat.
Stop (P)	master (FPGA)	Az átvitelt a master a STOP feltétel generálásával zárja le.

A korábban ismertetett aktuális hőmérséklet regiszteren túl az eszköz egyéb funkciókkal is rendelkezik. A chip összesen 4 regisztert tartalmaz, amelyet az 5-30. táblázat foglal össze. Az eszköz konfigurációs regiszterében található bitek funkcióit az 5-31. táblázat tartalmazza. Fontos megjegyezni, hogy a 16 bites regiszterek olvasásakor (pl. aktuális hőmérséklet regiszter) a master-nek minden esetben kötelező mind a két bájtot kiolvasnia, különben az LM75 blokkolja az I²C buszt. (Az első bájt olvasása után, az LM75 az SDA vonalat alacsonyan tartja, ezáltal lehetetlenné teszi a STOP bit generálását. Amikor megtörtént a második bájt kiolvasása, az LM75 engedi a buszciklust lezárni.)

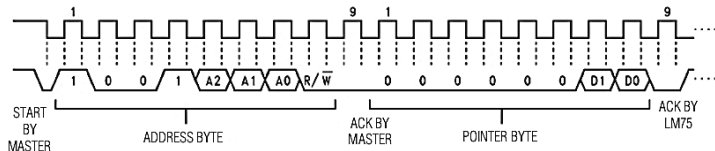
5-30. táblázat: Az LM75 áramkör regiszterei.

Regiszter cím	R/W	Szélesség	Név	Reset érték
b00	csak olvasható	16 bit	aktuális hőmérséklet regiszter	-
b01	írható/olvasható	8 bit	konfigurációs regiszter	0
b10	írható/olvasható	16 bit	T _{HYST} regiszter	75 °C
b11	írható/olvasható	16 bit	T _{OS} regiszter	80 °C

5-31. táblázat: A konfigurációs regiszter biteinek funkciói.

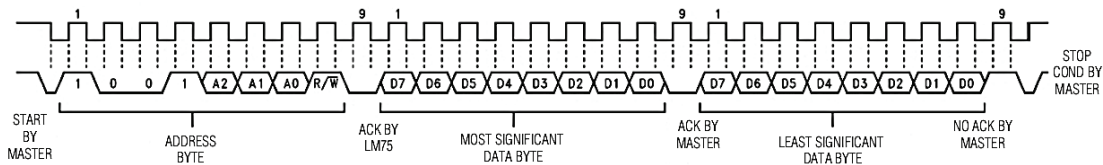
Bit	Név	Leírás
Bit 0	Készenlét	0: normál működés 1: az LM75 készenléti állapotba kerül
Bit 1	Cmp/Int	0: komparátor üzemmód 1: megszakítás üzemmód
Bit 2	OS polaritás	0: az OS kimenet aktív alacsony 1: az OS kimenet aktív magas
Bit 4-3	Megengedett hibák száma	A hőmérséklet hibák száma, mielőtt azt az O.S. kimenet jelezne: 00: jelzés az első hibánál (azonnali jelzés) 01: jelzés a második hibánál 10: jelzés a negyedik hibánál 11: jelzés a hatodik hibánál
Bit 7-5	Nem használt	Értéke kötelezően 0.

Azt, hogy a négy belső regiszter közül pontosan melyiket érjük el, egy belső kétbites pointer (mutató) regiszter tartalmazza. A pointer regiszter csak írható, reset után az értéke 0 (tehát az aktuális hőmérséklet regiszter kerül kiválasztásra). Ahhoz, hogy a többi regisztert is elérhessük, szükséges a pointer regiszter beállítása, melynek értékét az írási művelet során a cím bájtot követő bájtban kell megadni (5-24. ábra). A pointer regiszter a beállított értékét kikapcsolásig megőrzi, tehát ha ugyanazt a belső regisztert többször is olvassuk, nem szükséges előtte minden alkalommal a regiszter címet beállítani.

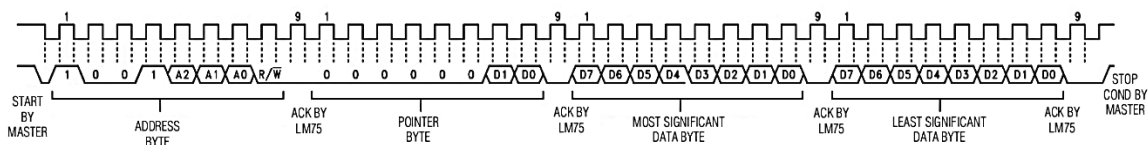


5-24. ábra: A pointer regiszter beállítása.

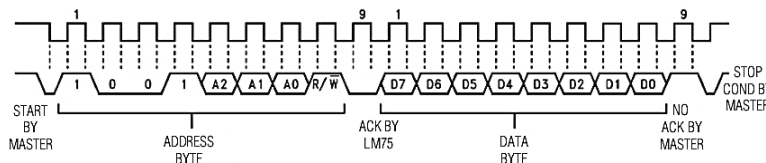
A 16 bites regiszterek olvasási idődiagramja az 5-25. ábrán látható, az írási idődiagramot az 5-26. ábra szemlélteti. A 8 bites konfigurációs regiszter olvasási idődiagramja az 5-27. ábrán látható, a regiszter írásának idődiagramját az 5-28. ábra mutatja. A regiszterek írása esetén a cím bájt kiadása után következik a pointer regiszter értéke, majd pedig a regiszter méretétől függően a beírandó 1 vagy 2 bájtnyi adat. Olvasás esetén a cím bájt kiadása után közvetlenül az 1 vagy 2 bájtos adat következik.



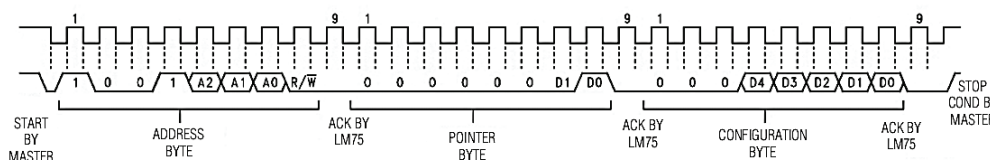
5-25. ábra: 16 bites regiszter olvasása (korábban beállított pointer regiszter mellett).



5-26. ábra: 16 bites regiszter írása.



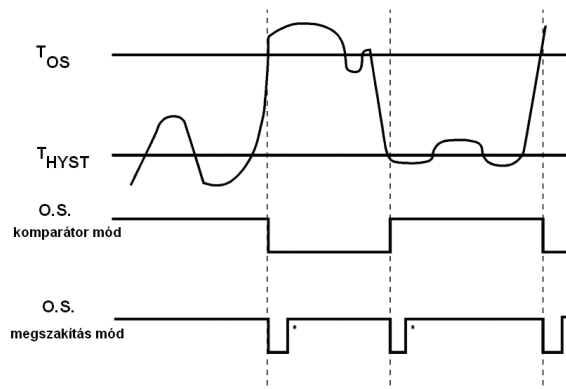
5-27. ábra: A 8 bites konfigurációs regiszter olvasása.



5-28. ábra: A 8 bites konfigurációs regiszter írása.

A hőmérséklet hiba kimenet (OS) komparátor módban felhasználható, mint egy egyszerű termosztát (pl. egy ventilátor vezérlésére). A felhasználó a T_{HYST} és a T_{OS} regiszterekkel tudja beállítani a kívánt kapcsolási hőmérsékletet. Mivel két hőmérsékletet kell definiálni, tetszőleges histerézis valósítható meg. Megszakítás módban az OS kimenetet egy mikroprocesszor

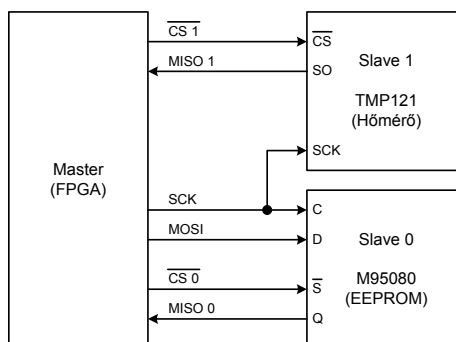
megszakítás lábához lehet kötni. Az aktív megszakítási állapot bármely regiszter kiolvasásával feloldható. Az OS kimenet kétféle üzemmódját az 5-29. ábra szemlélteti.



5-29. ábra: Hőmérséklet hiba kimenet használata

5.5.2 TMP121 – SPI buszos hőmérő

Az SPI (Serial Peripheral Interface) egy soros full-duplex adatátviteli interfész. Nagy előnye az I²C buszhoz képest, hogy csak egyirányú vonalakat használ, így akár több tíz MHz-es frekvencián is működhet. A buszra egy master és több slave eszköz kapcsolódhat. A bővítőmodulon két slave periféria kapott helyet (5-30. ábra): az egyik egy TMP121 típusú hőmérő chip, a másik egy M95080 típusú EEPROM memória. A master funkciót az FPGA tölti be.



5-30. ábra: A modul SPI egységei.

Az SPI interfész az alábbi jelekből áll:

- SCK (Serial Clock): órajel, forrása mindig a master egység.
- MOSI (Master Output Slave Input): a master egység soros adatkimenete.
- MISO (Master Input Slave Output): a slave egység soros adatkimenete.
- nCS/SS (Chip Select / Slave Select): aktív alacsony kiválasztó jel, a master egység adja ki.

Az SPI szabvány szerint minden slave egység saját nCS kiválasztó jellel rendelkezik, de egymással megosztott MISO kimenetet használnak. A modulon ezzel szemben minden slave egység saját kiválasztó jellel (nCS0, nCS1) és saját adatkimenettel (MISO0, MISO1) rendelkezik.

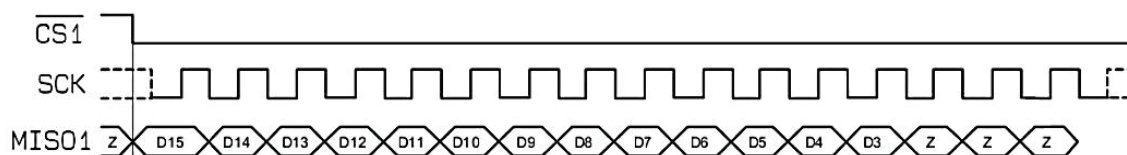
A TMP121 egy meglehetősen egyszerű integrált áramkör, amely csak a mért hőmérsékletet továbbítja a master felé a MISO1 vonalon át. Mivel a chipen semmilyen beállítási lehetőség nincs, ezért nincs is MOSI bemenete. Az TMP121 hőmérő áramkör adatlapja letölthető a <http://www.ti.com/product/tmp121> címről.

A mért hőmérséklet 0.0625 °C-os felbontással (± 1.5 °C pontossággal) egy belső 16 bites regiszterben tárolódik. A felső 13 biten található a tárolt hőmérséklet 0.0625 °C-os felbontással, kettes komplementes számbábrázolási formátumban, míg a regiszter alsó 3 bitje nem használt. A 13 biten tárolt hőmérséklet felső 9 bitje a mért hőmérséklet egészrésze, míg az alsó 4 bit a mért hőmérséklet törtrésze. Az adatbitek értelmezését és három minta hőmérséklet értéket az 5-32. táblázat tartalmazza.

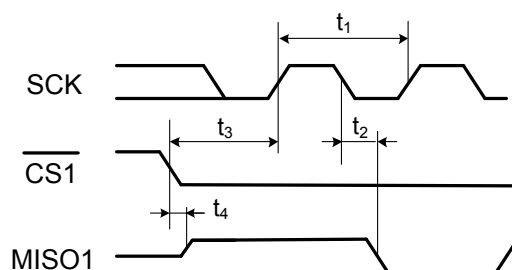
5-32. táblázat: A TMP121 chip által használt számbábrázolás.

	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	Egészrész									Törtrész				Don't care		
+25 °C	0	0	0	0	1	1	0	0	1	0	0	0	0	X	X	X
0.0625 °C	0	0	0	0	0	0	0	0	0	0	0	0	1	X	X	X
0 °C	0	0	0	0	0	0	0	0	0	0	0	0	0	X	X	X
-0.0625 °C	1	1	1	1	1	1	1	1	1	1	1	1	1	X	X	X
-25 °C	1	1	1	1	0	0	1	1	1	0	0	0	0	X	X	X

A TMP121 áramkörből a mért hőmérséklet értéket SPI interfészen lehet kiolvasni. Az olvasás idődiagramját az 5-31. ábra mutatja. Az SPI kiválasztó jel ($\overline{\text{nCS1}}$) aktív ideje alatt (logikai alacsony) 16 órajel (SCK) ütemben elküldésre kerül a 16 adatbit az IC soros adatkimenetén (MISO1). Az IC adatlapja szerint (és az ábrából láthatóan) az SPI kommunikáció során a TMP121 az SCK órajel lefutó élére shifteli ki a 16 bitet. Az áramkör SPI interfészének időzítési paramétereit az 5-32. ábra és az 5-33. táblázat mutatja.



5-31. ábra: A TMP121 adatkimenete.



5-32. ábra: A TMP121 SPI interfészének időzítése.

5-33. táblázat: A TMP121 SPI interfészének időzítési paramétereit.

Szimbólum	Paraméter	Min.	Max.	Egység
t_1	SCK órajel periódusideje	100	-	ns
t_2	MISO1 késleltetési ideje az SCK lefutó éle után	-	30	ns
t_3	$\overline{\text{nCS1}}$ előkészítési idő	40	-	ns
t_4	MISO1 késleltetési ideje az $\overline{\text{nCS1}}$ lefutó éle után	-	30	ns

Az IC a hőmérséklet mérését akkor végzi el, amikor a chip nincs kiválasztva, azaz az SPI kiválasztó jel ($\overline{\text{nCS1}}$) magas szintű. Egy konverzió időtartama 320 ms, az aktuális hőmérséklet kiolvasásához garantálnunk kell, hogy az $\overline{\text{nCS1}}$ jel legalább ennyi ideig magas állapotú legyen. Abban az esetben,

ha a kiválasztó jel huzamosabb ideig magas állapotban van, a chip 0,5 s periódusidővel frissíti a hőmérsékletet tároló belső regiszter értékét. A kiválasztó jel lefutó éle az esetlegesen futó aktuális mérési folyamat félbeszakad és a regiszterbe utoljára beírt érték olvasható ki.

5.5.3 M95080 – SPI buszos EEPROM

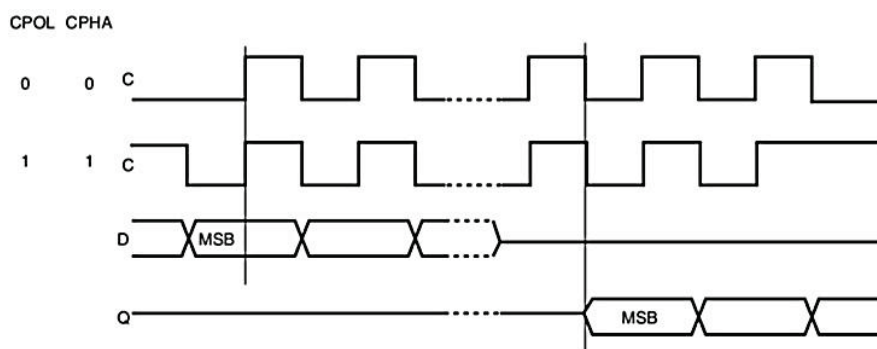
A modulon található másik SPI buszra csatlakozó eszköz az M95080 8 kbitos (1 kBájtos) EEPROM memória, amely valódi kétirányú SPI kommunikációt valósít meg. Az eszköz adatlapja letölthető a gyártó honlapjáról: <http://www.st.com>.

5.5.3.1 Az EEPROM SPI interfésze

A master és slave egységeknek meg kell állapodnia, hogy hogyan értelmezik az órajelet. Ennek megfelelően négyféle SPI adatátviteli mód lehetséges, amelyet az úgynevezett CPOL (órajel polaritás) és CPHA (órajel fázis) bitekkel szoktak jellemezni:

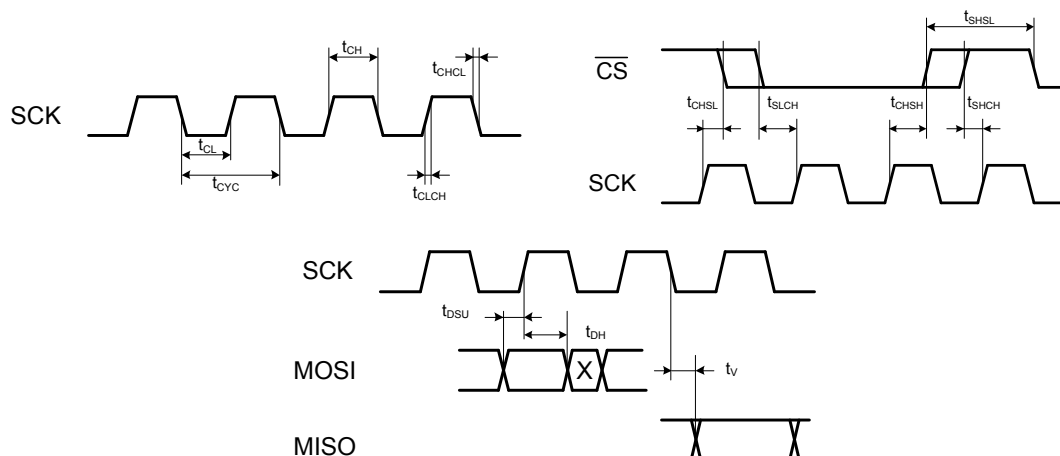
- Ha a CPOL üzemmód bit 0, akkor az SCK órajel alacsony szintű az inaktív állapotban.
 - Ha a CPHA üzemmód bit 0, akkor az adatok mintavételezése az SCK órajel felfutó élére történik, az adatok küldése pedig az SCK órajel lefutó élére.
 - Ha a CPHA üzemmód bit 1, akkor az adatok mintavételezése az SCK órajel lefutó élére történik, az adatok küldése pedig az SCK órajel felfutó élére.
- Ha a CPOL üzemmód bit 1, akkor az SCK vonal magas szintű az inaktív állapotban.
 - Ha a CPHA üzemmód bit 0, akkor az adatok mintavételezése az SCK órajel lefutó élére történik, az adatok küldése pedig az SCK órajel felfutó élére.
 - Ha a CPHA üzemmód bit 1, akkor az adatok mintavételezése az SCK órajel felfutó élére történik, az adatok küldése pedig az SCK órajel lefutó élére.

Az M95080 EEPROM memória a CPOL, CPHA = 00 vagy a CPOL, CPHA = 11 üzemmódokat támogatja (5-33. ábra). Tehát a MOSI (D) vonalon a bejövő adatokat mindig az órajel felfutó élénél mintavételezi és a MISO (Q) kimenetre pedig az órajel lefutó élének hatására helyez érvényes adatot. Az SPI interfészen az adatok kishiftelése mindig a legnagyobb helyiértékű bittel kezdődik (MSB first).



5-33. ábra: Az órajel időzítési viszonyai

Az M95080 áramkör SPI interfészének időzítési paramétereit az 5-34. ábra és az 5-34. táblázat szemlélteti.



5-34. ábra: Az M95080 EEPROM SPI interfészének idődiagramja.

5-34. táblázat: Az M95080 EEPROM SPI interfészének időzítési paraméterei.

Szimbólum	Paraméter	Min.	Max.	Egység
t_{CYC}	SCK órajel periódusideje	200	-	ns
t_{CL}/t_{CH}	SCK órajel alacsony/magas szintjének ideje	90	-	ns
t_{CLCH}/t_{CHCL}	SCK órajel felfutási/lefutási ideje	-	2	μs
t_{SLCH}/t_{CHSH}	nCS előkészítési idő	90	-	ns
t_{CHSL}/t_{SHCH}	nCS tartási idő	90	-	ns
t_{SHSL}	nCS magas (inaktív) szintjének ideje	100	-	ns
t_{DSU}	Adatbemenet (MOSI) előkészítési idő	90	-	ns
t_{DH}	Adatbemenet (MOSI) tartási idő	90	-	ns
t_V	Adatkimenet (MISO) késleltetési idő	-	100	ns

5.5.3.2 Az EEPROM státusz regisztere

Az EEPROM státusz regiszterében található bitek értelmezését az 5-35. táblázat mutatja. A státusz regiszter írható bitjei (SRWD, BP1 és BP0) az értéküket a tápfeszültség kikapcsolása után is megőrzik.

5-35. táblázat: Az EEPROM státusz regiszterének bitjei.

Bit	Név	R/W	Funkció
Bit 0	WIP	R	Write In Progress – Írás folyamatban Mivel az EEPROM írása 5 ms időt igényel, ezért a felhasználó ezen bit figyelésével győződhethet meg arról, hogy az írási ciklus már befejeződött-e vagy sem. 0: nincs aktív írási folyamat 1: írási művelet folyamatban
Bit 1	WEL	R	Write Enable Latch – Írás Engedélyezés Latch Minden írás típusú utasítás (WRITE, WRSR) kiadása előtt a WEL bitet egybe kell állítani a WREN utasítás kiadásával. Az alábbi események után a WEL bit ismét alapállapotba kerül: - Bekapcsolás - WRDI utasítás - WRSR utasítás befejezése - WRITE utasítás befejezése 0: EEPROM írás tiltva (alapállapot) 1: EEPROM írás engedélyezve

Bit 2-3	BP0/1	R/W	Az EEPROM memória egy része írásvédetté tehető ezen bitek segítségével. - BP[1:0] = 00 esetén nincs írásvédelem - BP[1:0] = 01 esetén a 0x300 – 0x3FF címtartomány írásvédett - BP[1:0] = 10 esetén a 0x200 – 0x3FF címtartomány írásvédett - BP[1:0] = 11 esetén a teljes memória írásvédett
Bit 4-6		R	Nem használt
Bit 7	SRWD	R/W	Status Register Write Disable – Státusz Regiszter Írás Tiltva Abban az esetben, ha az EEPROM chip nWE lába alacsony és az SRWD bit értéke egy, a státusz regiszter írhatósága letiltódik (hardveres védelem). Ezáltal lehet biztosítani a BP0 és BP1 bitek védelmét. Mivel a modulon az nWE láb fixen tápfeszültségre van kötve, ezért a hardveres védelem nem használható.

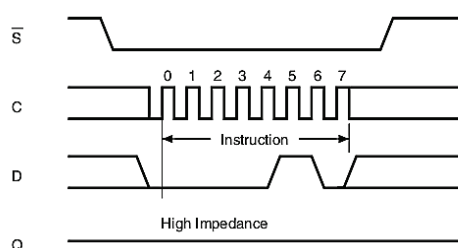
5.5.3.3 Az EEPROM parancskészlete

Az M95080 EEPROM memóriával parancsok segítségével kommunikálhatunk, amelyeket az SPI buszon keresztül továbbíthatunk az eszköznek. Az 5-36. táblázat foglalja össze a memóriának elküldhető parancsokat.

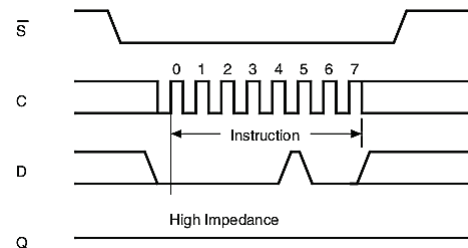
5-36. táblázat: Az M95080 EEPROM parancskészlete.

Parancs	Bináris	Hex	Funkció
WREN	b0000 0110	0x06	Írás engedélyezés
WRDI	b0000 0100	0x04	Írás tiltás
RDSR	b0000 0101	0x05	Státusz regiszter olvasás
WRSR	b0000 0001	0x01	Státusz regiszter írás
READ	b0000 0011	0x03	Memória olvasás
WRITE	b0000 0010	0x02	Memória írás

Minden írás típusú utasítást (WRITE, WRSR) meg kell előznie egy WREN (írás engedélyezés) utasításnak (5-35. ábra), amely hatására a státusz regiszter WEL bitjének értéke 1 lesz, ezáltal engedélyeződnek a WRITE és a WRSR utasítások. A WRDI utasítás (5-36. ábra) nem nagy gyakorlati jelentőséggel bír, hiszen az írási ciklusok végén a WEL bit ismét törlődik, tehát a memória írásvédetté válik.



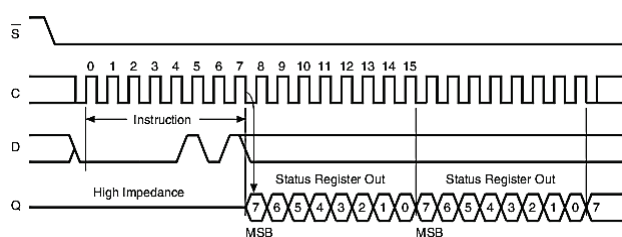
5-35. ábra: A WREN utasítás idődiagramja.



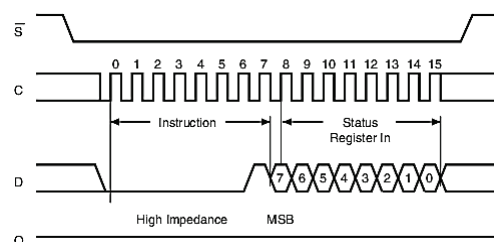
5-36. ábra: A WRDI utasítás idődiagramja.

Mivel az írási folyamat időtartama 5 ms, a státusz regiszter periodikus olvasásával és a WIP bit periodikus lekérdezésével vizsgálhatjuk meg, hogy az írás befejeződött-e. A státusz regiszter kiolvasása az RDSR utasítás kiadásával lehetséges (5-37. ábra). Az ábrán jól látható, hogy amennyiben a master egység további órajeleket biztosít, a státusz regiszter többször is kiolvasható egyetlen buszciklusban.

A WRSR parancs segítségével (5-38. ábra) írhatjuk a státusz regiszter SRWD, BP1 és BP0 bitjeit. Használata előtt ne felejtsük el kiadni a WREN parancsot. A státusz regiszter írásának időtartama 5 ms, az írási folyamat végét a WIP bit értékének vizsgálatával állapíthatjuk meg.

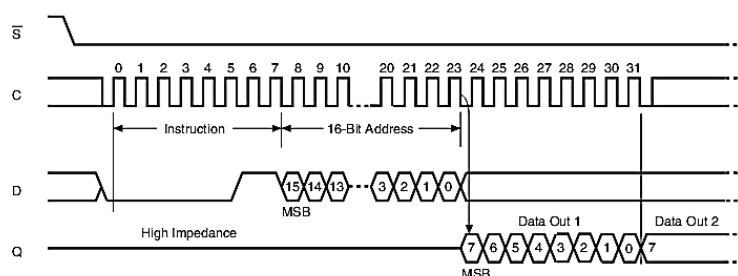


5-37. ábra: Az RDSR utasítás idődiagramja.



5-38. ábra: A WRSR utasítás idődiagramja.

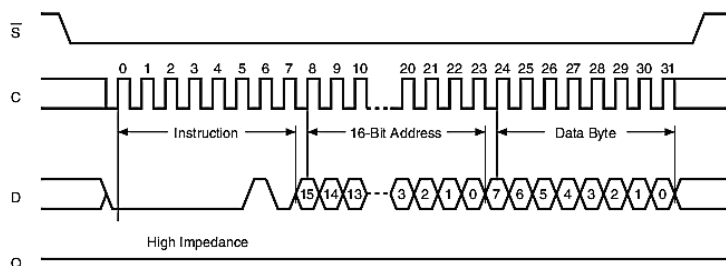
Az 5-39. ábrán az EEPROM olvasási ciklusa látható. A READ parancs kishiftelése után 16 biten továbbítjuk az olvasni kívánt memóriacella címét (az 1024 byte megcímezéséhez 10 bitre van szükség, a felső 6 címbit értéke nincs figyelembe véve). Ezek után a következő órajel periódustól kezdve az EEPROM már továbbítja a kért adatot a MISO0 (Q) soros kimeneten keresztül. Amennyiben a master egység további órajel pulzusokat is küld, az EEPROM a cím automatikus növelésével a következő memóriacella értékét is az adatvonalra helyezi. Az olvasási folyamat tetszőleges hosszan folytatható. Amennyiben a belső címszámláló eléri a végértéket, automatikusan túlcserélődik, és folytatja az olvasást a nullás címtől kezdődően. Fontos megjegyezni, hogy az írási művelet alatt READ parancs nem adható ki.



5-39. ábra: A READ utasítás idődiagramja.

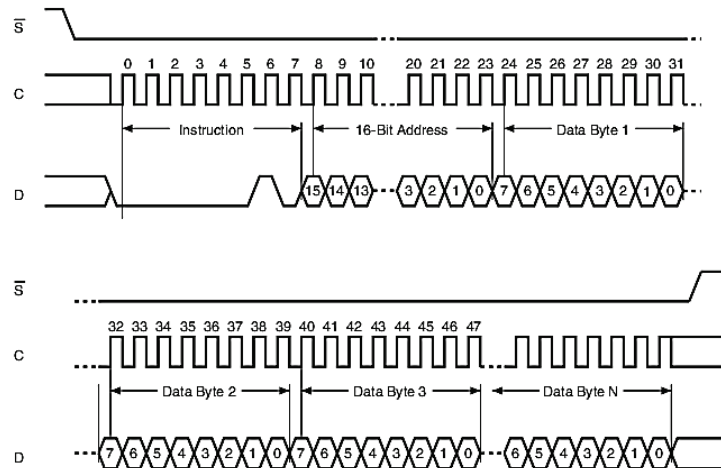
A WRITE parancs segítségével tudjuk a memória cellákat írni (5-40. ábra). A parancs kishiftelése után 16 biten továbbítjuk az írni kívánt memória cella címét, majd pedig a beírandó 8 bites adatot. Az írási folyamat időtartama 5 ms, melynek végét a WIP bit értékének vizsgálatával állapíthatjuk meg. Az adat beírása az alábbi esetekben nem történik meg:

- Ha a WEL bit értéke 0 (a WRITE utasítást meg kell előznie a WREN utasításnak).
- Ha a WIP bit értéke 1 (írási művelet van folyamatban).
- Ha BP1 és BP0 bitek által definiált módon írásvédett területre próbálunk írni.
- Ha az nCS alacsony szintje alatt kiadott órajel pulzusok száma nem osztható nyolccal.



5-40. ábra: A WRITE utasítás idődiagramja.

A memóriacellák 32 bájtos lapokra vannak osztva. Amennyiben az írási buszciklus folytatódik, további bájtok írhatók be egyetlen WRITE utasítással (5-41. ábra). Fontos azonban megjegyezni, hogy az automatikusan növekvő címszámláló csak az alsó 6 címvezetéken számol (a laphatáron belül), tehát egyszerre maximum 32 bájt, azaz egyetlen lap írható be. Amennyiben az eszköznek küldött adatok laphatárt lépnek át, a címszámláló ugyanazon lap kezdetére és nem pedig a következő lapra fog mutatni.



5-41. ábra: Több bájt írása egyetlen WRITE utasítással.

5.5.4 Kapcsolási rajz

