

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
VILLAMOSMÉRŐNKI ÉS INFORMATIKAI KAR
MÉRÉSTECHNIKA ÉS INFORMÁCIÓS RENDSZEREK TANSZÉK

Rendszerarchitektúrák labor

Xilinx EDK

Raikovich Tamás

BME MIT



BME-MIT

FPGA labor

Labor tematika (Xilinx EDK)

- **1. labor:**
 - A Xilinx EDK fejlesztői környezet ismertetése
- **2. labor:**
 - Egyszerű processzoros rendszer összeállítása
 - Egyszerű szoftver alkalmazások készítése
- **3. labor:**
 - Saját periféria illesztése
 - Megszakításkezelés
 - Egyidejű HW/SW fejlesztés (debugger, ChipScope)

BME-MIT

FPGA labor

Témakörök

- Beágyazott rendszerek
- MicroBlaze processzor
- EDK alapok
- Gyári és saját IP-k hozzáadása
- Szoftverfejlesztés
- HW és SW együttes fejlesztése



Embedded Development Kit

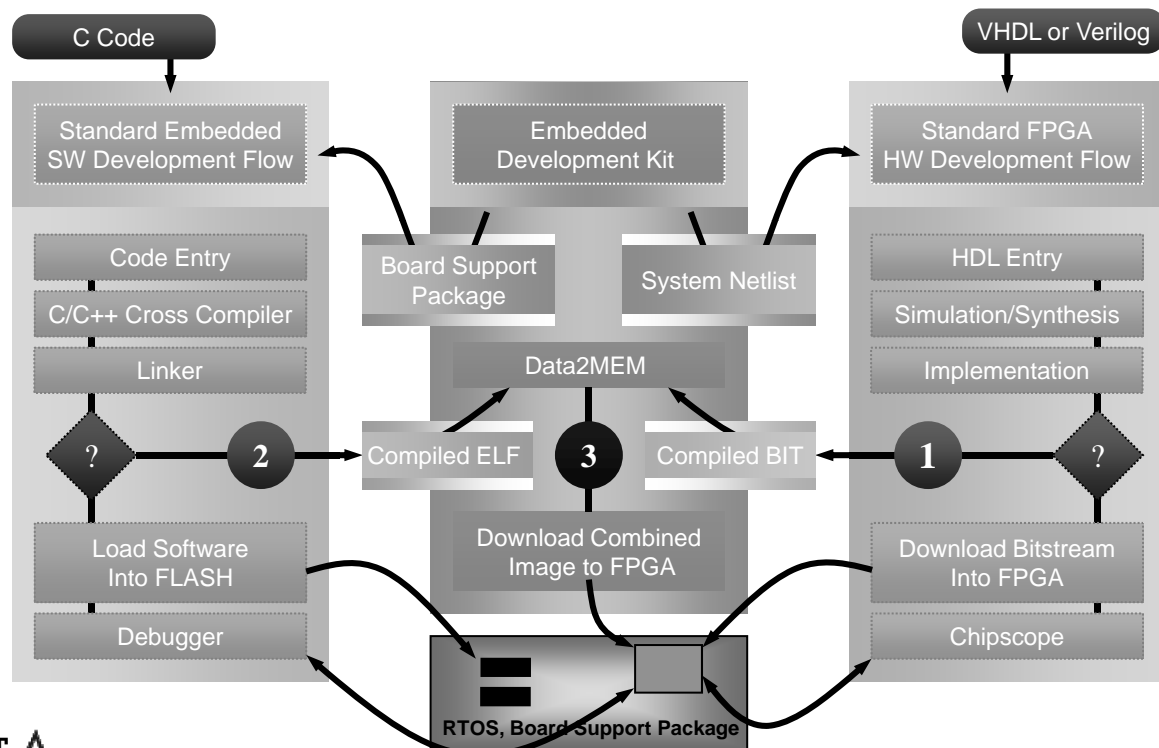
Mi is az Embedded Development Kit (EDK)?

- Az EDK a Xilinx fejlesztői környezete az FPGA alapú beágyazott rendszerekhez
- Tartalmaz minden eszközt, dokumentációt és IP-t, amely szükséges a PowerPC és a MicroBlaze processzort használó rendszerek létrehozásához
- Lehetővé teszi a beágyazott rendszerek hardver és szoftver komponenseinek integrálását



Embedded Development Kit

A fejlesztési folyamat áttekintése:



Embedded Development Kit

Beágyazott hardver fejlesztés:

- A *Base System Builder Wizard* segítségével gyorsan létrehozható a hardver rendszer egy adott eszközre
- A hardver rendszer kibővítése
 - Gyári perifériák hozzáadása az *IP katalógusból*
 - Saját periféria: *Create/Import Pheripherals Wizard*
 - ChipScope beillesztése: *Debug Configuration Wizard*
- HDL huzalozási lista generálása: *PlatGen*
- HDL szimuláció valamilyen szimulátor használatával
 - Szimulációs modellek generálása: *SimGen*

Embedded Development Kit

Beágyazott szoftver fejlesztés:

- Eszközmeghajtók és könyvtárak generálása: *LibGen*
- Szoftverfejlesztés: *Xilinx Platform Studio (XPS)* vagy *Software Development Kit (SDK)*
 - Program lefordítása: *GNU C/C++ fordító (gcc)*
 - Kapcsolódás a célrendszerhez:
Xilinx Microprocessor Debug (XMD)
 - Debugger: *GNU debugger (gdb)*

Embedded Development Kit

Hardver és szoftver integrálás:

- A bitfolyam létrehozása és az FPGA konfigurálása
 - A *BitInit* frissíti a belső Blokk-RAM-ok tartalmát a futtatható kóddal
- Külső FLASH memória konfigurálása
 - Írás külső FLASH memóriába: *Flash Writer*
 - Compact FLASH konfigurációs fájl létrehozása:
System ACE File generator (GenACE)

Embedded Development Kit

- **Szoftver könyvtárak**
 - **lwIP Library**: hálózati könyvtár
 - **LibXil MFS**: memória fájlrendszer
 - **LibXil FATfs**: FAT fájlrendszer
 - **LibXil Flash**: FLASH memória támogatás
 - Standard C könyvtárak (libc, libm)
- **Operációs rendszer Board Support Package-k (BSP)**
 - Önálló operációs rendszer (Standalone OS)
 - Xilinx MicroKernel (XMK)
 - Linux: nem része az EDK-nak, külön kell telepíteni

Xilinx Platform Studio

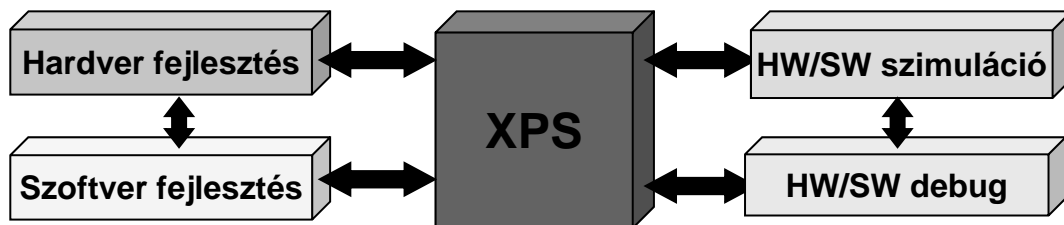
The screenshot displays the Xilinx Platform Studio interface in System Assembly View. The central pane shows a hierarchical tree of IP blocks and bus interfaces. A table lists these components with their names, bus names, IP types, and versions. Callout boxes provide annotations for key features:

- Projektfájlok elérése**: Points to the Project pane on the left.
- SW alkalmazás fejlesztés**: Points to the Applications pane.
- IP-k kiválasztása a katalógusból**: Points to the IP Catalog pane at the bottom.
- Hardver elemek összekapcsolása**: Points to the central System Assembly View diagram.
- A rendszer blokkvázlatának megtekintése**: Points to the Block Diagram pane at the bottom.
- A busz interfész és a port nézet szűrése**: Points to the Bus Interfaces and Ports panes.
- Információk a tervről, jelentések és üzenetek megtekintése**: Points to the Design Summary pane at the bottom.

Name	Bus Name	IP Type	IP Vers
microblaze_0		microblaze	7.20.b
DLMB	dlimb		
ILMB	lmb		
DPLB	mb_plb		
IPLB	mb_plb		
DXCL	microblaze_0_DXCL		
IXCL	microblaze_0_IXCL		
DEBUG	microblaze_0_mdm_bus		
TRACE	microblaze_0_TRACE		
dlimb_cntlr			
lmb_v10		lmb_v10	1.00.a
lmb		lmb_v10	1.00.a
mb_plb		plb_v46	1.04.a
lmb_bram_if...		lmb_bram_if...	2.10.b
lmb_cntlr			
SPLB0	dlimb		
BRAM_P...	dlimb_cntlr_BRAM_PORT		
lmb_bram			
SPLB	lmb		
BRAM_P...	lmb_cntlr_BRAM_PORT		
DDR_SDRAM			
SPLB0	mb_plb	mipmc	5.02.a
lmb_bram			
PORTA	lmb_port		
PORTB	dlimb_port		
mdm_0			
SPLB	mb_plb		
MFSLO	No Connection		
MBDEBU...	mdm_0_MBDEBUG_0		
XMTC	mdm_0_XMTC		



Xilinx Platform Studio

- **Projekt menedzsment**
 - Microprocessor Hardware Specification (MHS) fájl létrehozása
 - Microprocessor Software Specification (MSS) fájl létrehozása
 - Xilinx Microprocessor Project (XMP) fájl
- **Platform menedzsment**
 - A fejlesztési folyamat eszközeinek beállítása
 - Szoftver platform beállítások
 - A fejlesztési folyamat eszközeinek meghívása
 - Hibakeresés és szimuláció
- **Szoftver alkalmazás menedzsment**



Xilinx Platform Studio

Projekt létrehozása/megnyitása:

- **Új projekt létrehozása**
 - **File** → **New Project** vagy a  gomb a toolbar-on
 - Base System Builder Wizard
 - Blank XPS Project (üres projekt)
- **Meglévő projekt megnyitása**
 - **File** → **Open Project** vagy a  gomb a toolbar-on
 - Az XMP fájl kiválasztása
 - **File** → **New Project** vagy a  gomb a toolbar-on
 - Open a Recent Project, majd az XMP fájl kiválasztása
- **A projekt információk az XMP fájlba mentődnek el**

Base System Builder Wizard

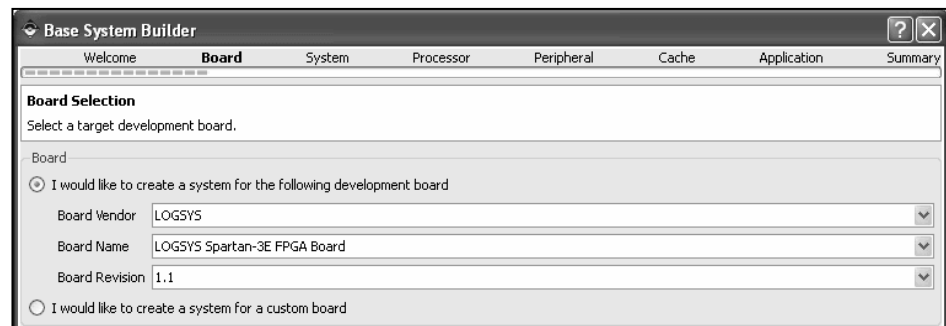
A *Base System Builder Wizard* segítségével gyorsan létre lehet hozni egy működőképes rendszert egy adott FPGA kártyára:

1. A cél FPGA kártya kiválasztása
2. A processzoros rendszer kiválasztása
3. A processzor(ok) konfigurálása
4. A perifériák kiválasztása és konfigurálása
5. A cache konfigurálása
6. A szoftver alkalmazások konfigurálása
7. A rendszer létrehozása

Base System Builder Wizard

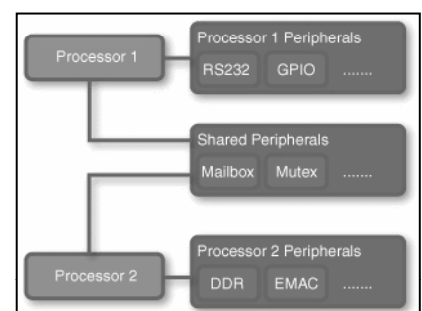
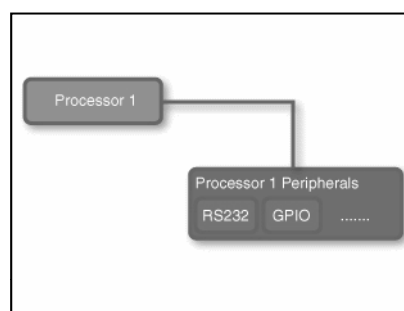
1. A cél FPGA kártya kiválasztása

- Gyári
- Egyedi



2. A processzoros rendszer kiválasztása

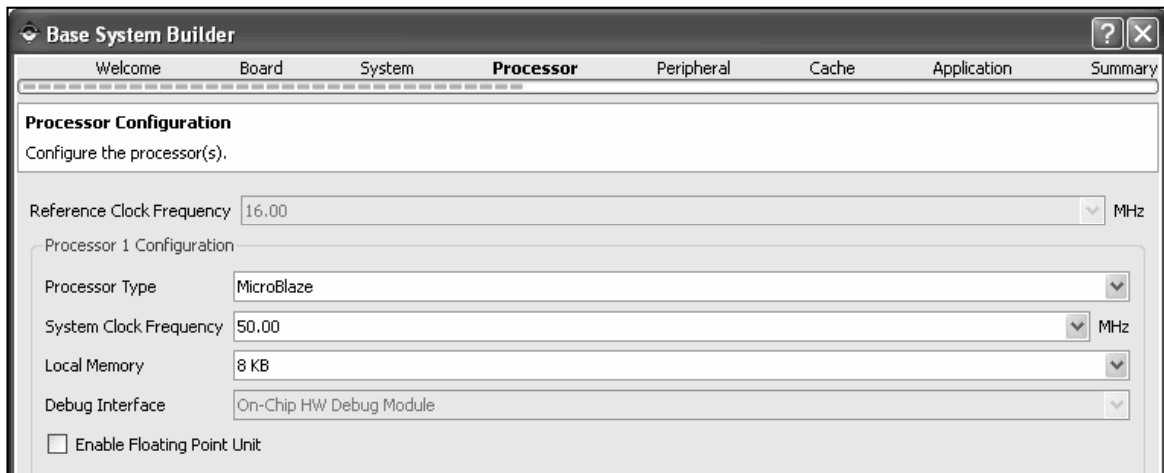
- Egy processzor
- Két processzor



Base System Builder Wizard

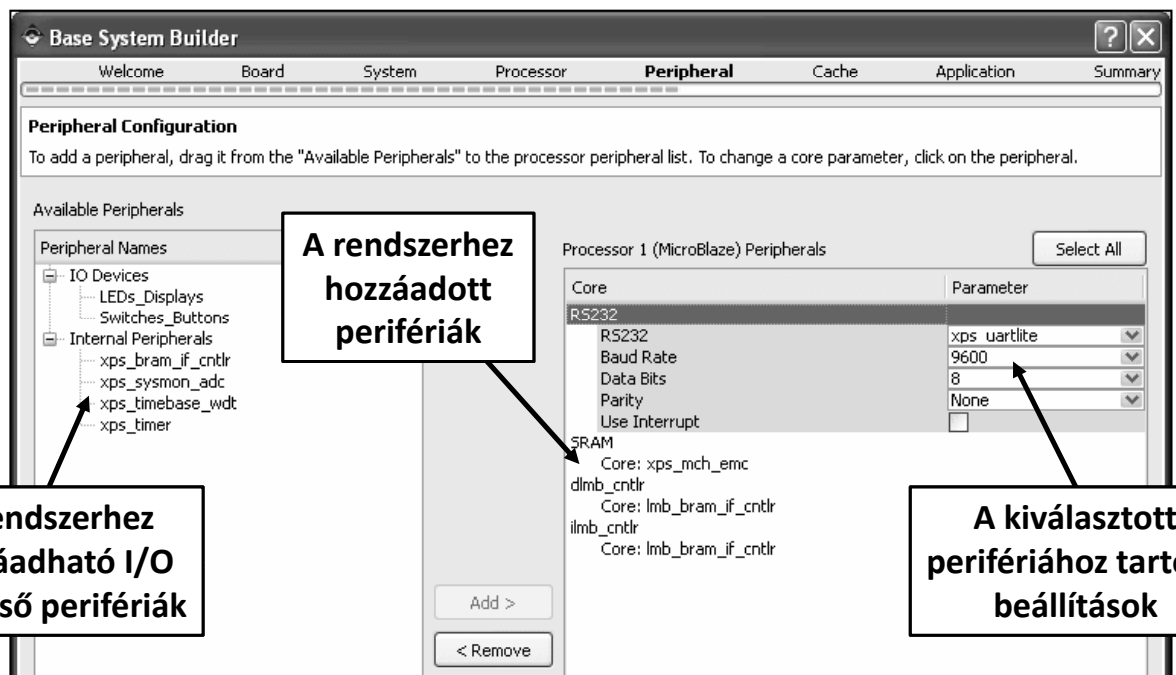
3. A processzor(ok) konfigurálása

- Processzor típusa: MicroBlaze, PowerPC
- Referencia órajel és rendszer órajel frekvencia
- Az LMB buszra csatlakozó Blokk-RAM mérete
- FPU engedélyezése



Base System Builder Wizard

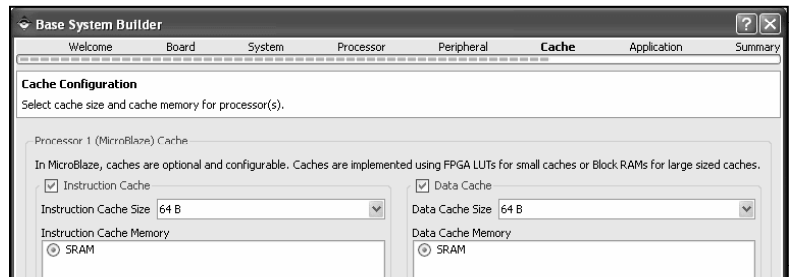
4. A perifériák kiválasztása és konfigurálása



Base System Builder Wizard

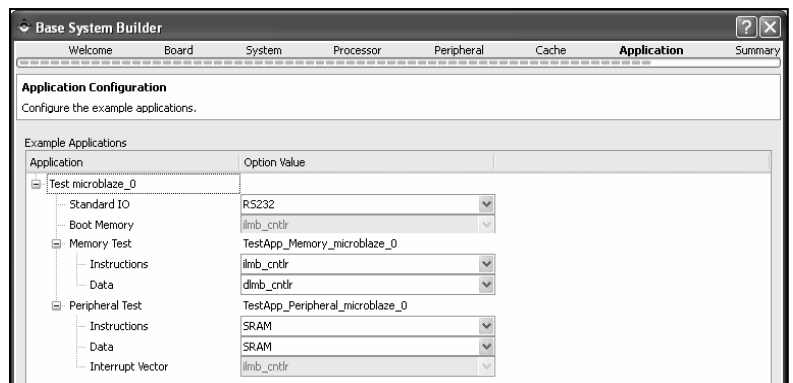
5. A cache konfigurálása

- Utasítás cache
 - Méret → LUT/BRAM
 - Gyorsítótárazott mem.
- Adat cache méret
 - Méret → LUT/BRAM
 - Gyorsítótárazott mem.



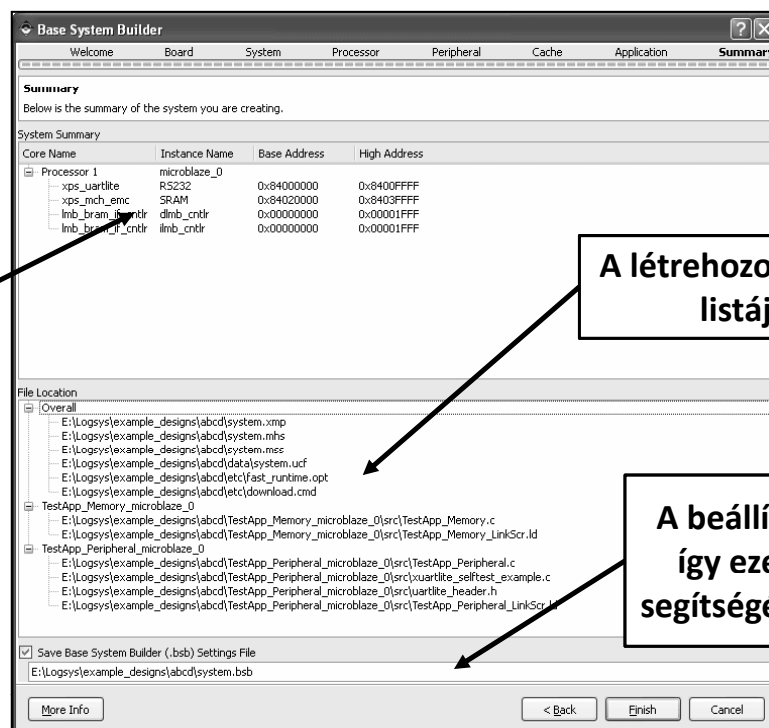
6. A szoftver alkalmazások konfigurálása

- Standard I/O
- Boot memória:
 - Blokk-RAM
 - FLASH memória
- Teszt alkalmazások
 - Utasítás memória
 - Adat memória
 - Megszakítás vektorok



Base System Builder Wizard

7. A rendszer létrehozása



A rendszerhez hozzáadott perifériák és a cím kiosztás

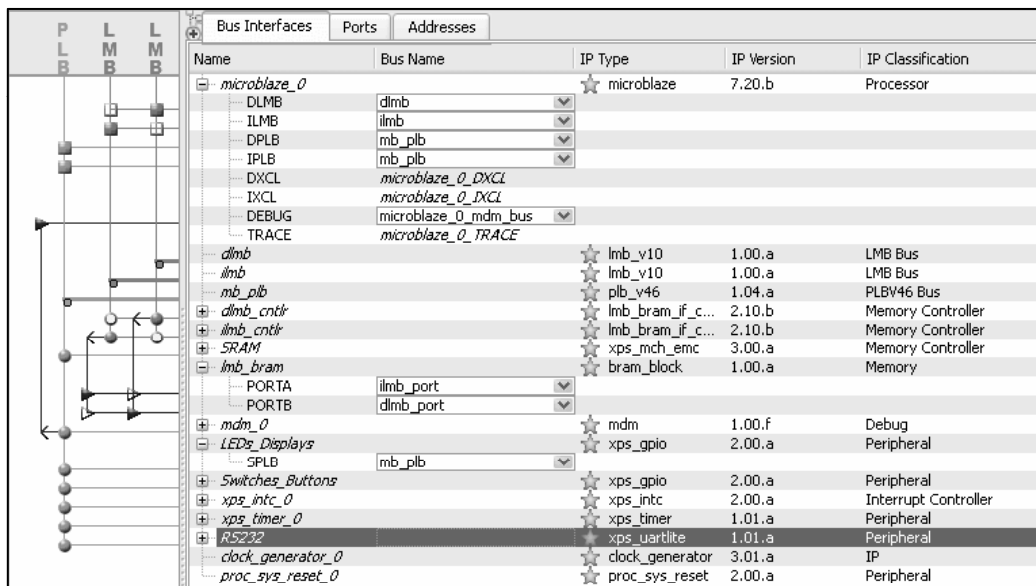
A létrehozott fájlok listája

A beállítások elmentése, így ezek később a BSB segítségével módosíthatók

Base System Builder Wizard

A létrejött rendszer: busz interfészek

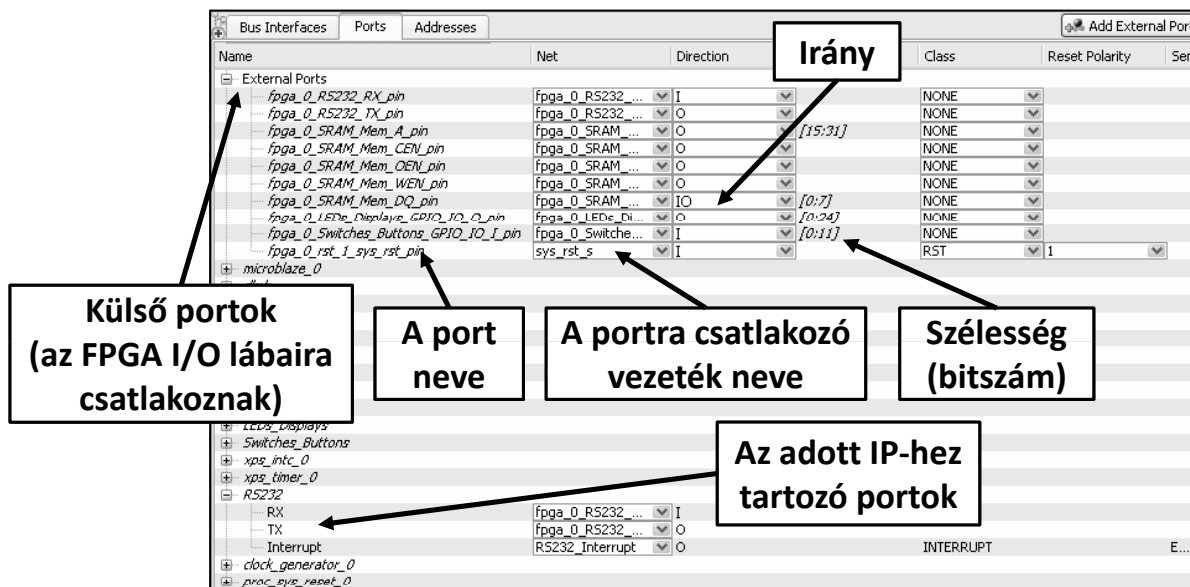
- *System Assembly* nézet → *Bus Interfaces* fül
- Itt módosíthatók a busz interfész összeköttetések



Base System Builder Wizard

A létrejött rendszer: portok

- *System Assembly* nézet → *Ports* fül
- Port összeköttetések módosítása, portok hozzáadása és törlése



Base System Builder Wizard

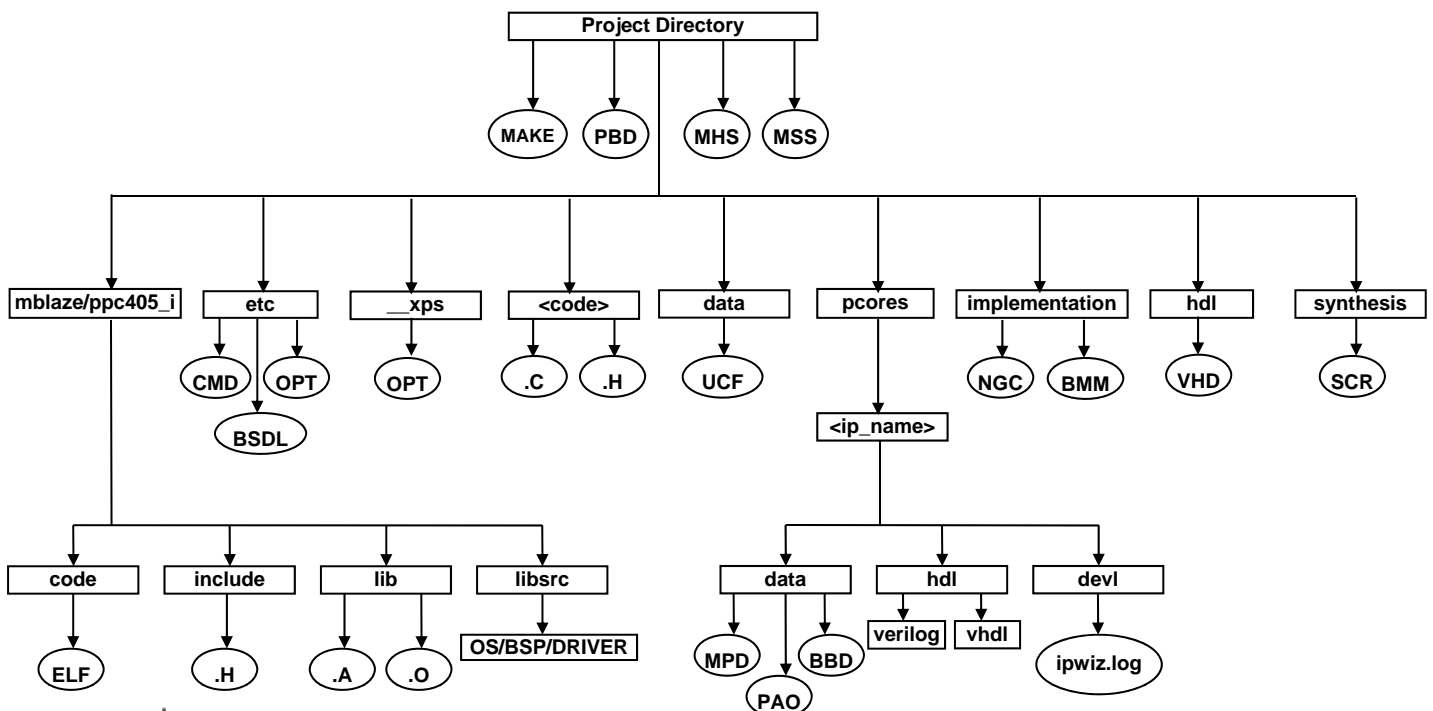
A létrejött rendszer: címek

- *System Assembly* nézet → *Addresses* fül
- Címtartomány hozzárendelése a perifériákhoz
- A címek újragenerálása: *Generate Addresses* gomb

Instance	Base Name	Base Address	High Address	Size	Bus Interface(s)	Bus Name
microblaze_0's Address Map						
dlmb_cnlr	C_BASEADDR	0x00000000	0x00001FFF	8K	SLMB	dlmb
ilmb_cnlr	C_BASEADDR	0x00000000	0x00001FFF	8K	SLMB	ilmb
Switches_Buttons	C_BASEADDR	0x81400000	0x8140FFFF	64K	SPLB	mb_plb
LEDs_Displays	C_BASEADDR	0x81420000	0x8142FFFF	64K	SPLB	mb_plb
xps_intc_0	C_BASEADDR	0x81800000	0x8180FFFF	64K	SPLB	mb_plb
xps_timer_0	C_BASEADDR	0x83C00000	0x83C0FFFF	64K	SPLB	mb_plb
RS232	C_BASEADDR	0x84000000	0x8400FFFF	64K	SPLB	mb_plb
SRAM	C_MEM0_BASEADDR	0x84020000	0x8403FFFF	128K	SPLB	mb_plb
mdm_0	C_BASEADDR	0x84400000	0x8440FFFF	64K	SPLB	mb_plb

Xilinx Platform Studio projekt

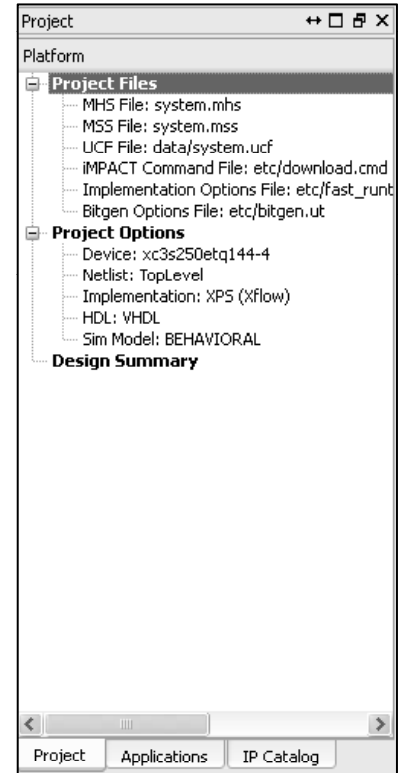
XPS projekt struktúra



Xilinx Platform Studio projekt

Projekt fájlok és beállítások

- Bal oldalon a **Project** fül
- **Project Files**
 - A rendszert leíró fájlok
 - Beállításokat tároló fájlok
- **Project Options**
 - A projekt beállításainak módosítása
- **Design Summary**
 - Összegzés a rendszerről
 - Az egyes fázisokhoz tartozó jelentések és üzenetek megtekintése



Xilinx Platform Studio projekt

Microprocessor Hardware Specification (MHS) fájl:

- A hardver terv struktúráját írja le
 - Külső portok
 - Rendszert felépítő modulok
 - Modul paraméterek
 - Modul busz interfészek és ezek kapcsolódása
 - Modul portok és ezek kapcsolódása
- Nem kell manuálisan módosítani, az XPS frissíti, ha a hardver tervben változások történtek

Xilinx Platform Studio projekt

Microprocessor Hardware Specification (MHS) fájl:

```
PARAMETER VERSION = 2.1.0
```

```
PORT fpga_0_RS232_RX_pin = fpga_0_RS232_RX_pin, DIR = I
PORT fpga_0_RS232_TX_pin = fpga_0_RS232_TX_pin, DIR = O
PORT fpga_0_SRAM_Mem_A_pin = fpga_0_SRAM_Mem_A_pin_vslice_15_31_concat, DIR = O, VEC = [15:31]
PORT fpga_0_SRAM_Mem_CEN_pin = fpga_0_SRAM_Mem_CEN_pin, DIR = O
PORT fpga_0_SRAM_Mem_OEN_pin = fpga_0_SRAM_Mem_OEN_pin, DIR = O
PORT fpga_0_SRAM_Mem_WEN_pin = fpga_0_SRAM_Mem_WEN_pin, DIR = O
PORT fpga_0_SRAM_Mem_DQ_pin = fpga_0_SRAM_Mem_DQ_pin, DIR = IO, VEC = [0:7]
PORT fpga_0_LEDs_Displays_GPIO_IO_0_pin = fpga_0_LEDs_Displays_GPIO_IO_0_pin, DIR = O, VEC = [0:24]
PORT fpga_0_Switches_Buttns_GPIO_IO_I_pin = fpga_0_Switches_Buttns_GPIO_IO_I_pin, DIR = I, VEC = [0:11]
PORT fpga_0_rst_1_sys_rst_pin = sy
```

Külső portok

MicroBlaze processzor példány

```
BEGIN microblaze
PARAMETER INSTANCE = microblaze_0
PARAMETER C_FAMILY = spartan3e
PARAMETER C_AREA_OPTIMIZED = 1
PARAMETER C_INTERCONNECT = 1
PARAMETER C_DEBUG_ENABLED = 1
PARAMETER HW_VER = 7.20.b
BUS_INTERFACE DLMB = dlmb
BUS_INTERFACE ILMB = ilmb
BUS_INTERFACE DPLB = mb_plb
BUS_INTERFACE IPLB = mb_plb
BUS_INTERFACE DEBUG = microblaze_0_mdm_bus
PORT MB_RESET = mb_reset
PORT INTERRUPT = microblaze_0_interrupt
END
```

Modul paraméterek

Busz interfészek (pl.: az IPLB és a DPLB az mb_plb buszra csatlakozik)


Portok (pl.: az INTERRUPT portra a microblaze_0_interrupt vonal csatlakozik)

BME-MIT

FPGA labor

Xilinx Platform Studio projekt

Microprocessor Software Specification (MHS) fájl:

- A szoftver platform beállításokat tartalmazza
 - Processzor
 - Operációs rendszer és szoftver könyvtárak
 - A perifériákhoz tartozó eszközmeghajtók
- Módosítás az XPS GUI-ból:
 - **Software** → **Software Platform Settings** vagy a  gomb a toolbar-on

```
PARAMETER VERSION = 2.2.0
```

```
BEGIN OS
PARAMETER OS_NAME = standalone
PARAMETER OS_VER = 2.00.a
PARAMETER PROC_INSTANCE = microblaze_0
PARAMETER STDIN = RS232
PARAMETER STDOUT = RS232
END
```

```
BEGIN PROCESSOR
PARAMETER DRIVER_NAME = cpu
PARAMETER DRIVER_VER = 1.12.a
PARAMETER HW_INSTANCE = microblaze_0
PARAMETER COMPILER = mb-gcc
PARAMETER ARCHIVER = mb-ar
END
```

```
BEGIN DRIVER
PARAMETER DRIVER_NAME = uartlite
PARAMETER DRIVER_VER = 1.14.a
PARAMETER HW_INSTANCE = RS232
END
```

```
BEGIN DRIVER
PARAMETER DRIVER_NAME = emc
PARAMETER DRIVER_VER = 2.00.a
PARAMETER HW_INSTANCE = SRAM
END
```

```
BEGIN DRIVER
PARAMETER DRIVER_NAME = gpio
PARAMETER DRIVER_VER = 2.13.a
PARAMETER HW_INSTANCE = LEDs_Displays
END
```

BME-MIT

FPGA labor

Xilinx Platform Studio projekt

User Constraints File (UCF):

- **A felhasználó által előírt megkötéseket tartalmazza**
 - Külső portok hozzárendelése az FPGA I/O lábaihoz
 - Időzítési adatok, előírások
 - Elhelyezési megkötések, előírások
 - Stb.
- **A Base System Builder Wizard automatikusan generálja nem egyedi FPGA kártya esetén**
- **Manuálisan kell módosítani**
 - Egyedi FPGA kártya vagy üres projekt esetén
 - Ha manuálisan adtunk hozzá IP-t a rendszerhez
- **Itt csak a portoknak az FPGA lábakhoz való hozzárendelését tárgyaljuk (további részletek a Xilinx dokumentációban)**

Xilinx Platform Studio projekt

User Constraints File (UCF):

- **A külső portok hozzárendelése az FPGA I/O lábaihoz**
 - Egybites port esetén
NET [port_név] LOC=[fpga_láb];
 - Vektor esetén minden bithez
NET [port_név<bit_index>] LOC=[fpga_láb];
- **Az I/O lábakhoz kapcsolódó egyéb megkötések**
 - Felhúzó ellenállás
NET [port_név[<bit_index>]] PULLUP;
 - Lehúzó ellenállás
NET [port_név[<bit_index>]] PULLDOWN;
 - I/O szabvány (pl. LVCMOS33, LVTTTL33, stb.)
NET [port_név[<bit_index>]] IOSTANDARD=[io_standard];
 - Meghajtás erőssége [mA] (I/O szabvány függő)
NET [port_név[<bit_index>]] DRIVE=[ma_érték];
 - Jelváltozási sebesség (SLOW, FAST)
NET [port_név[<bit_index>]] SLEW=SLOW/FAST;

Xilinx Platform Studio projekt

User Constraints File (UCF):

- Azonos porthoz tartozó megkötések összevonása: | karakter
`NET [port név] LOC=[fpga láb] | PULLUP;`
- Kommentek: # karakter
- UCF fájl példa:

```
*****
#* VGA interfész. *
*****
NET vga_red<0> LOC = P134 | IOSTANDARD = LVCMOS33 | DRIVE = 16 | SLEW = FAST;
NET vga_red<1> LOC = P7 | IOSTANDARD = LVCMOS33 | DRIVE = 16 | SLEW = FAST;
NET vga_green<0> LOC = P135 | IOSTANDARD = LVCMOS33 | DRIVE = 16 | SLEW = FAST;
NET vga_green<1> LOC = P5 | IOSTANDARD = LVCMOS33 | DRIVE = 16 | SLEW = FAST;
NET vga_blue<0> LOC = P139 | IOSTANDARD = LVCMOS33 | DRIVE = 16 | SLEW = FAST;
NET vga_blue<1> LOC = P4 | IOSTANDARD = LVCMOS33 | DRIVE = 16 | SLEW = FAST;
NET vga_hsync LOC = P140 | IOSTANDARD = LVCMOS33 | DRIVE = 16 | SLEW = FAST;
NET vga_vsync LOC = P3 | IOSTANDARD = LVCMOS33 | DRIVE = 16 | SLEW = FAST;

*****
#* USRT interfész. *
*****
NET usrt_clk LOC = P129 | IOSTANDARD = LVCMOS33 | PULLDOWN;
NET usrt_rxd LOC = P120 | IOSTANDARD = LVCMOS33 | PULLUP;
NET usrt_txd LOC = P143 | IOSTANDARD = LVCMOS33;
```

Xilinx Platform Studio projekt

iMPACT Command File (etc/download.cmd):

- Az FPGA konfigurálásához szükséges parancsokat tartalmazza
- Ha több eszköz is van a JTAG láncban, akkor meg kell adni, hogy hányadik eszköz a konfigurálandó FPGA
– p paraméter: 1 → 1. eszköz, 2 → 2. eszköz, stb.

```
setMode -bscan
setCable -p auto
identify
assignfile -p 1 -file implementation/download.bit
program -p 1
quit
```

Xilinx Platform Studio projekt

BitGen Options File (etc/bitgen.ut):

- A konfigurációs fájl generálásánál felhasznált beállítások
- Ha a BIT fájlt konfigurációs memóriába programozzuk
 - Indító órajel: JTAGCLK helyett CCLK
 - Egyébként nem fog elindulni az FPGA

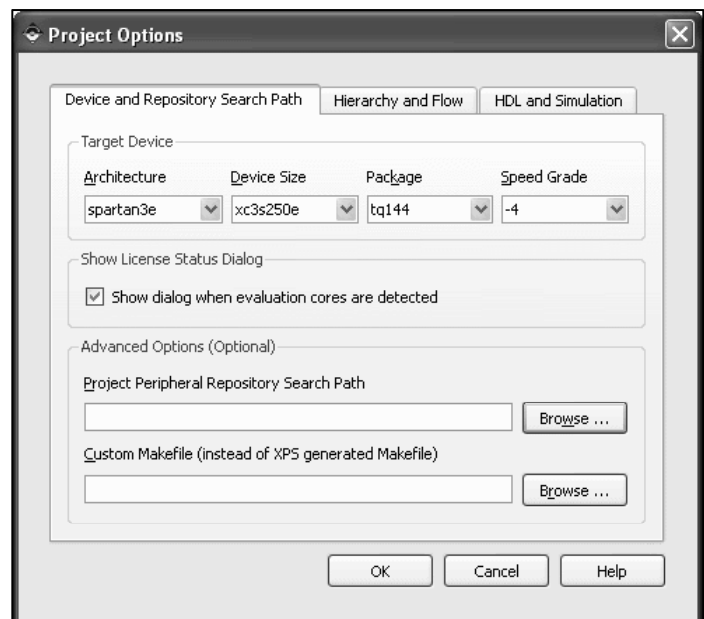
```
-g CclkPin:PULLUP
-g TdoPin:PULLNONE
-g M1Pin:PULLDOWN
-g DonePin:PULLUP
-g StartUpClk:JTAGCLK
-g M0Pin:PULLUP
-g M2Pin:PULLUP
-g ProgPin:PULLUP

-g TckPin:PULLUP
-g TdiPin:PULLUP
-g TmsPin:PULLUP
-g LCK_cycle:NoWait
-g Security:NONE
#-m
-g Persist:No
```

Xilinx Platform Studio projekt



Projekt beállítások:

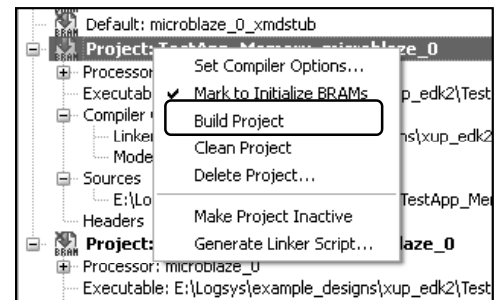
- FPGA eszköz kiválasztása
- A projekt IP-k elérési útvonala
- Az időzítési eltérések hibaként kezelése
- Generált HDL fájlok nyelve
 - VHDL vagy Verilog
- Szimulációs tesztkörnyezet létrehozása
- A generált szimulációs modell
 - Viselkedési
 - Strukturális
 - Időzítési



Xilinx Platform Studio



Az elkészült rendszer kipróbálása:

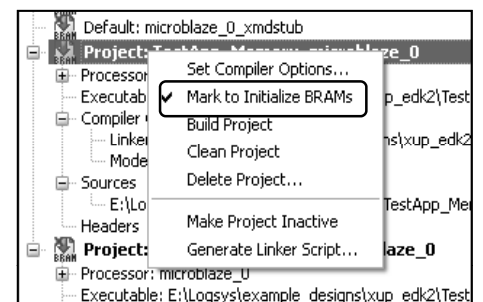
- A Base System Builder Wizard által generált memória teszt alkalmazást célszerű használni (**TestApp_Memory_microblaze_0**)
- A huzalozási lista (netlist) létrehozása
 - **Hardware** → **Generate Netlist** vagy a  gomb a toolbar-on
 - HDL fájlok generálása (PlatGen)
 - Szintézis
- Az FPGA konfigurációs fájl létrehozása
 - **Hardware** → **Generate Bitstream** vagy a  gomb a toolbar-on
 - Translate, Map, Place & Route
 - Konfigurációs bitfolyam generálás
- A memória teszt alkalmazás lefordítása
 - **Applications** fül → **TestApp_Memory**
 - Jobb kattintás, majd **Build Project**



Xilinx Platform Studio

Az elkészült rendszer kipróbálása:

- A memória teszt alkalmazás lefordítása
 - Első alkalommal lefordulnak a meghajtók és a könyvtárak is
 - Létrejön az ELF fájl
- A Blokk-RAM inicializálása az alkalmazással
 - **Mark to Initialize BRAMs** opció legyen bejelölve
 - **Device Configuration** → **Update Bitstream** vagy a  gomb
 - Létrejön a download.bit fájl
- Csatlakoztassuk az FPGA kártyát a soros porton keresztül a PC-hez
- Indítsuk el egy terminál alkalmazást (pl. HyperTerminal)
 - 115200 bit/s, 8 adatbit, nincs paritás
 - 1 stop bit, nincs átvitelvezérlés
- Az FPGA konfigurálása
 - **Device Configuration** → **Download Bitstream**
 - Vagy a  gomb a toolbar-on



Xilinx Platform Studio

Az elkészült rendszer kipróbálása:

- A memória teszt futása során a terminál ablakban megjelenő üzenet

```
-- Entering main() --  
Starting MemoryTest for DDR_SDRAM:  
  Running 32-bit test...PASSED!  
  Running 16-bit test...PASSED!  
  Running 8-bit test...PASSED!  
-- Exiting main() --
```