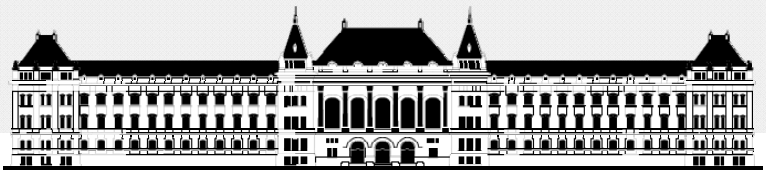


BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
VILLAMOSMÉRNÖKI ÉS INFORMATIKAI KAR
MÉRÉSTECHNIKA ÉS INFORMÁCIÓS RENDSZEREK TANSZÉK

Rendszerarchitektúrák labor
Xilinx EDK

Raikovich Tamás

BME MIT

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)

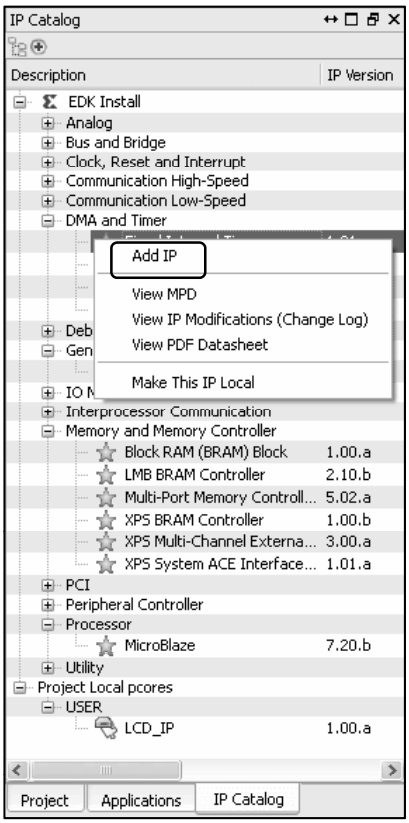
Témakörök

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

Perifériák hozzáadása a rendszerhez

Intellectual Property (IP) katalógus:

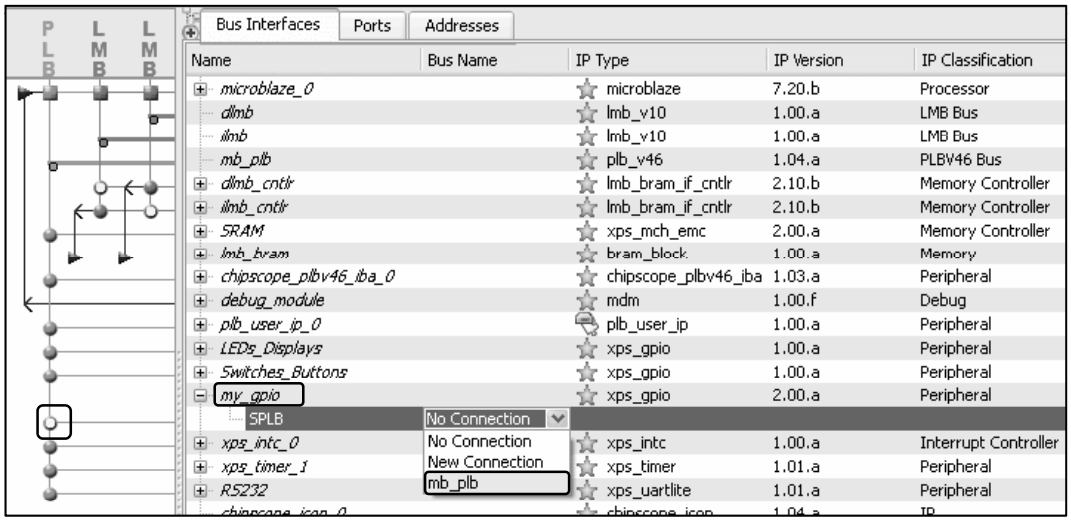
- **Az elérhető IP modulok listája**
- **Bal oldalon az *IP Catalog* fül**
- **Ingyenes IP modulok**
- **Fizetős IP modulok: korlátozások**
  - Időkorlátosan használható
  - Csak szimuláció megengedett
  - Stb.
- **IP modul hozzáadása a rendszerhez**
  - Jobb kattintás a modul nevén
  - ***Add IP*** menüpont kiválasztása



Perifériák hozzáadása a rendszerhez

Az IP modul (pl. GPIO) csatlakoztatása a rendszerhez:

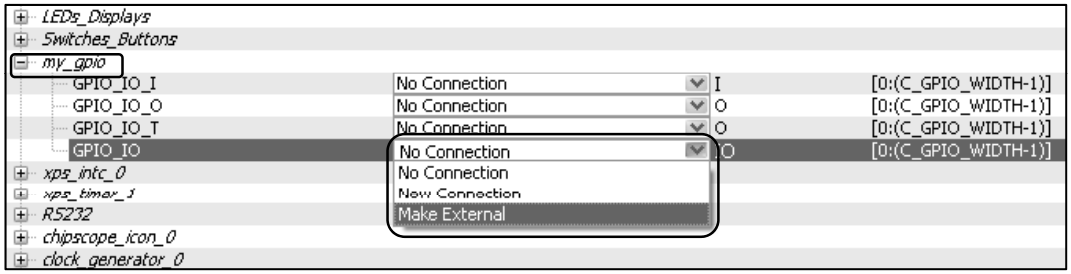
- **Átnevezés: kattintsunk az IP modul nevére, majd írjuk át**
- **Csatlakoztatás a buszra: *Bus Interfaces* fül**
  - A legördülő menüből válasszuk ki a megfelelő buszt (***mb_plb***), vagy
  - Kattintsunk az IP modultól balra lévő üres körre



Perifériák hozzáadása a rendszerhez

Az IP modul (pl. GPIO) csatlakoztatása a rendszerhez:

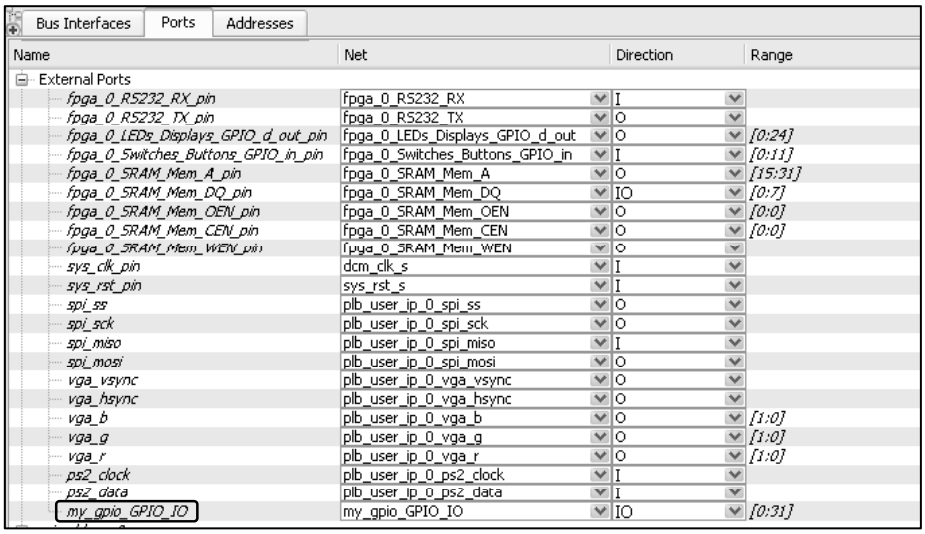
- **A portok bekötése: *Ports* fül**
  - A legördülő menüből válasszuk ki a megfelelő elemet
  - ***No Connection***: az adott port nincs bekötve
  - ***New Connection***: adott portra csatlakozó új vonal létrehozása
  - ***Make External***: az adott port kivezetése az FPGA I/O lábakra
  - ***Meglévő vonal kiválasztása***



Perifériák hozzáadása a rendszerhez

Az IP modul (pl. GPIO) csatlakoztatása a rendszerhez:

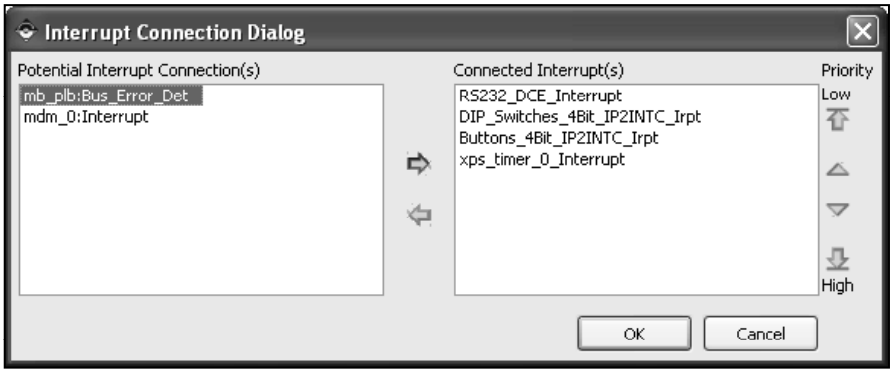
- **A portok bekötése: *Ports* fül**
  - ***Make External***: új elemmel bővül a külső portok listája
  - Átnevezés: kattintsunk a külső port nevére, majd írjuk át
  - Az UCF fájlban az itt megadott portneveket kell használni



Perifériák hozzáadása a rendszerhez

Megszakításkérő vonal bekötése a megszakítás vezérlőbe:

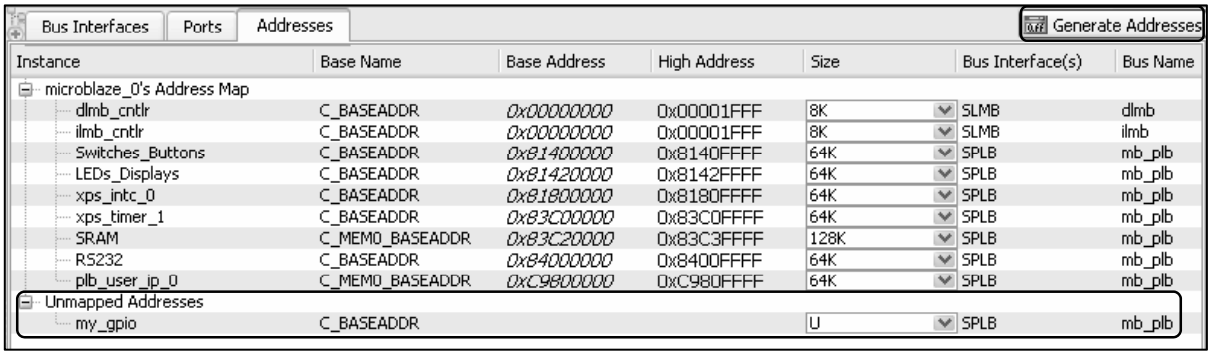
- **Periféria megszakításkérő vonala: *New Connection***
- **Megszakítás vezérlő *Intr* portja: kattintsunk a gombra → ablak**
  - Baloldali lista: a még nem csatlakoztatott IRQ vonalak
  - Jobboldali lista: a már csatlakoztatott IRQ vonalak
  - Megszakításkérő vonal csatlakoztatása
  - Megszakításkérő vonal eltávolítása
  - Megszakítások priorításának beállítása



Perifériák hozzáadása a rendszerhez

Az IP modul (pl. GPIO) csatlakoztatása a rendszerhez:

- **Cím hozzárendelés: *Addresses* fül**
  - ***Generate Addresses*** gomb: a címek újragenerálása
  - A báziscím módosítása: kattintsunk rá és írjuk át
  - A címtartomány méretének módosítása: a legördülő menüből válasszuk ki az új méretet

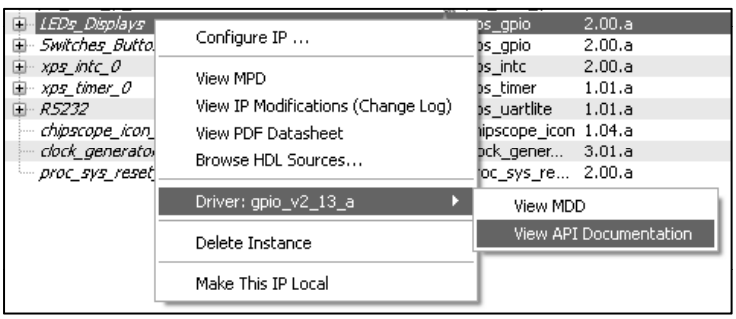


Instance	Base Name	Base Address	High Address	Size	Bus Interface(s)	Bus Name
microblaze_0's Address Map						
dlmb_cntlr	C_BASEADDR	0x00000000	0x00001FFF	8K	SLMB	dlmb
ilmb_cntlr	C_BASEADDR	0x00000000	0x00001FFF	8K	SLMB	ilmb
Switches_Buttons	C_BASEADDR	0x81400000	0x8140FFFF	64K	SPLB	mb_plb
LEDs_Displays	C_BASEADDR	0x81420000	0x8142FFFF	64K	SPLB	mb_plb
xps_intc_0	C_BASEADDR	0x81800000	0x8180FFFF	64K	SPLB	mb_plb
xps_timer_1	C_BASEADDR	0x83C00000	0x83C0FFFF	64K	SPLB	mb_plb
SRAM	C_MEM0_BASEADDR	0x83C20000	0x83C3FFFF	128K	SPLB	mb_plb
RS232	C_BASEADDR	0x84000000	0x8400FFFF	64K	SPLB	mb_plb
plb_user_ip_0	C_MEM0_BASEADDR	0xC9800000	0xC980FFFF	64K	SPLB	mb_plb
Unmapped Addresses						
my_gpio	C_BASEADDR			U	SPLB	mb_plb

Perifériák hozzáadása a rendszerhez

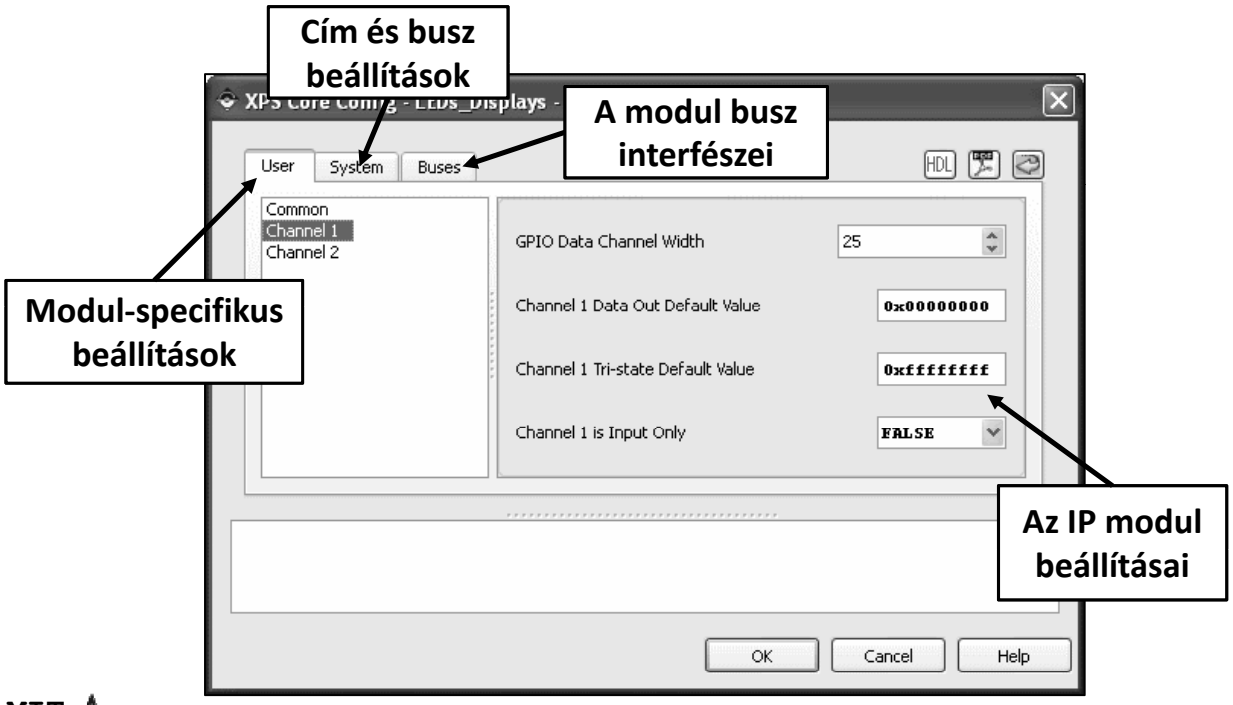
Az IP modul (pl. GPIO) konfigurálása:

- **A System Assembly nézet → jobb kattintás az IP nevén → menü**
  - Az IP modul konfigurálása
  - Az IP modul leíró fájl megtekintése/szerkesztése
  - A változások megtekintése
  - Az IP modul adatlapjának megtekintése
  - A HDL forrásfájlok megtekintése/szerkesztése
  - Eszközmeghajtó
    - A meghajtó leíró fájl megtekintése/szerkesztése
    - API dokumentáció
  - Az IP modul törlése



Perifériák hozzáadása a rendszerhez

Az IP modul (pl. GPIO) konfigurálása: *Configure IP…* menüpont



Perifériák hozzáadása a rendszerhez

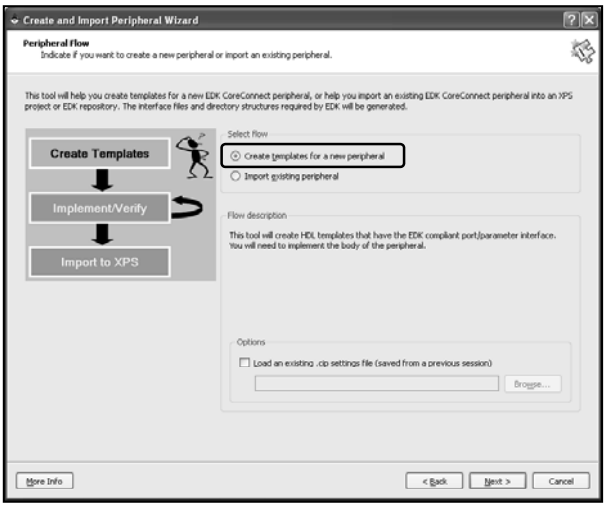
Az IP modul konfigurációs beállításai:

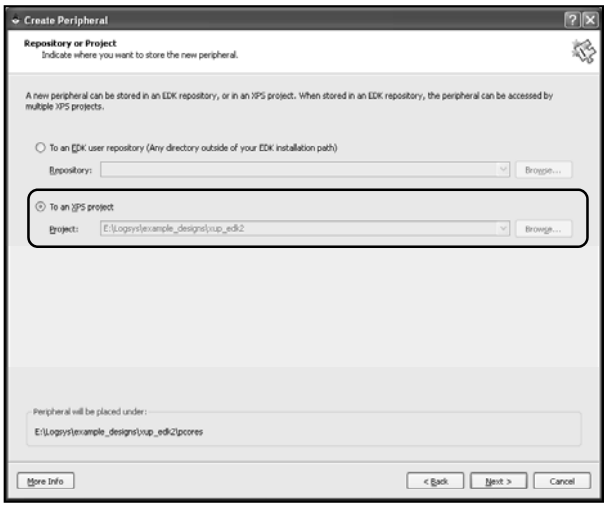
- **A példában használt GPIO modul beállításai**
  - Megszakítás engedélyezése
  - I/O csatornák száma: 1 vagy 2
    - Szélesség: 1 – 32 bit
    - Az adatregiszter alapértelmezett értéke
    - Az irányregiszter alapértelmezett értéke
    - Kétirányú vagy csak bemenet
- **Részletek az IP modulok adatlapjaiban**

Saját periféria létrehozása

Create and Import Peripheral Wizard:

- ***Hardware*** **menü →** ***Create or Import Peripheral…*** **vagy**
- **A**  **gomb a toolbar-on**
- **Új periféria létrehozása**
- **A perifériát az XPS projekt könyvtárában tároljuk**

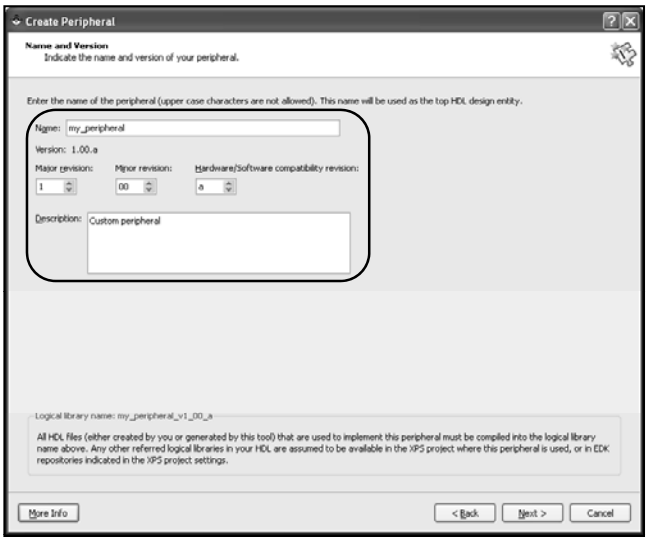


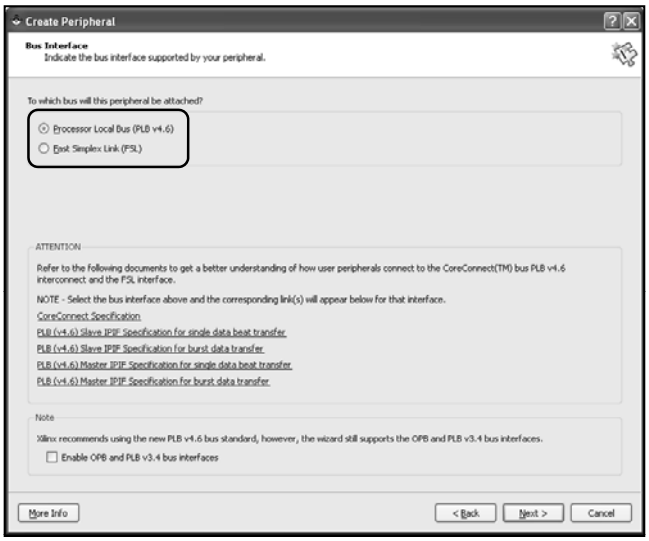


Saját periféria létrehozása

Create and Import Peripheral Wizard:

- **A periféria nevének, verziójának és leírásának megadása**
- **A busz interfész kiválasztása: *PLB v4.6* vagy *FSL***

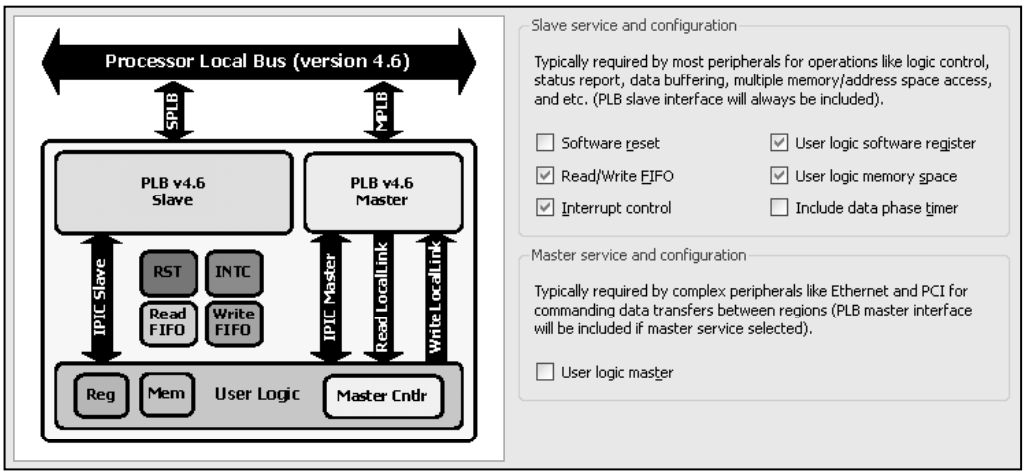




Saját periféria létrehozása

Create and Import Peripheral Wizard:

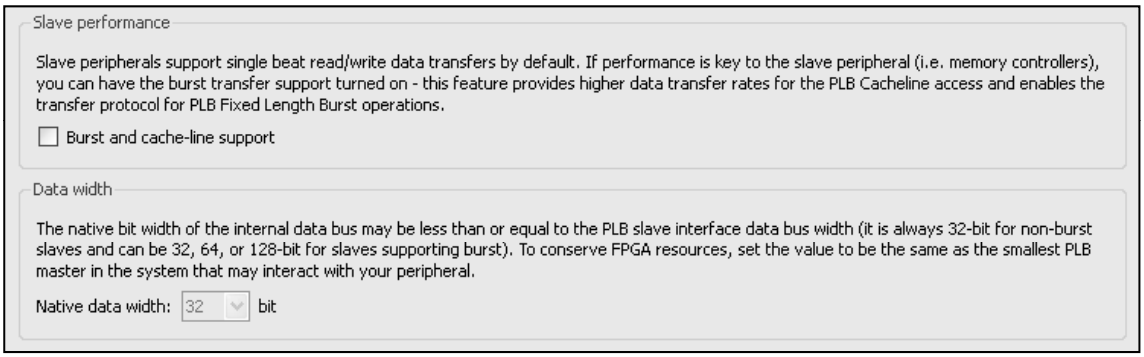
- **IPIF slave szolgáltatások**
  - Szoftveres reset
  - Regiszterek, memória tartomány, FIFO
  - Megszakítás vezérlő
- **IPIF master szolgáltatások**
  - Tipikusan összetett perifériák (pl. Ethernet) esetén, **nem fogjuk használni**



Saját periféria létrehozása

Create and Import Peripheral Wizard:

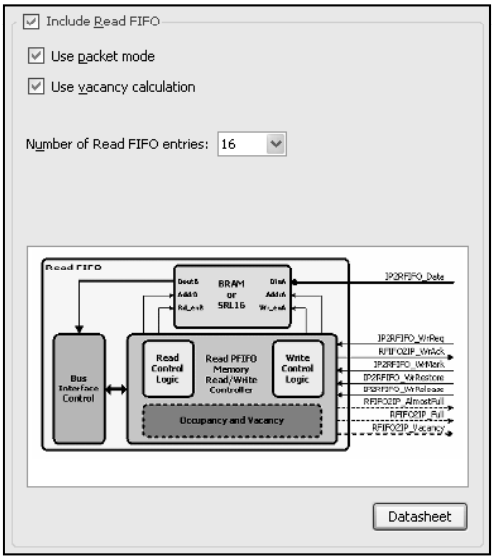
- **Master/slave interfész teljesítménye: burst adatátvitel**
  - Nagyteljesítményű perifériák (pl. memória vezérlő) esetén
  - Nem fogjuk használni
- **Adatbusz szélessége**
  - Nem burst adatátvitel esetén: mindig 32 bit
  - Burst adatátvitel esetén: 32, 64 vagy 128 bit



Saját periféria létrehozása

Create and Import Peripheral Wizard:

- **Slave szolgáltatások: FIFO**
  - Pufferelés biztosítása a PLB busz és az IP modul között
  - Írás és olvasás a PLB IPIF regiszterein keresztül
  - Részletek az adatlapban
- **A FIFO-k beállításai**
  - Írási és olvasási FIFO engedélyezése
  - Csomag mód engedélyezése
  - Szabad helyek számának jelzése
  - FIFO mérete → LUT/BRAM



Saját periféria létrehozása

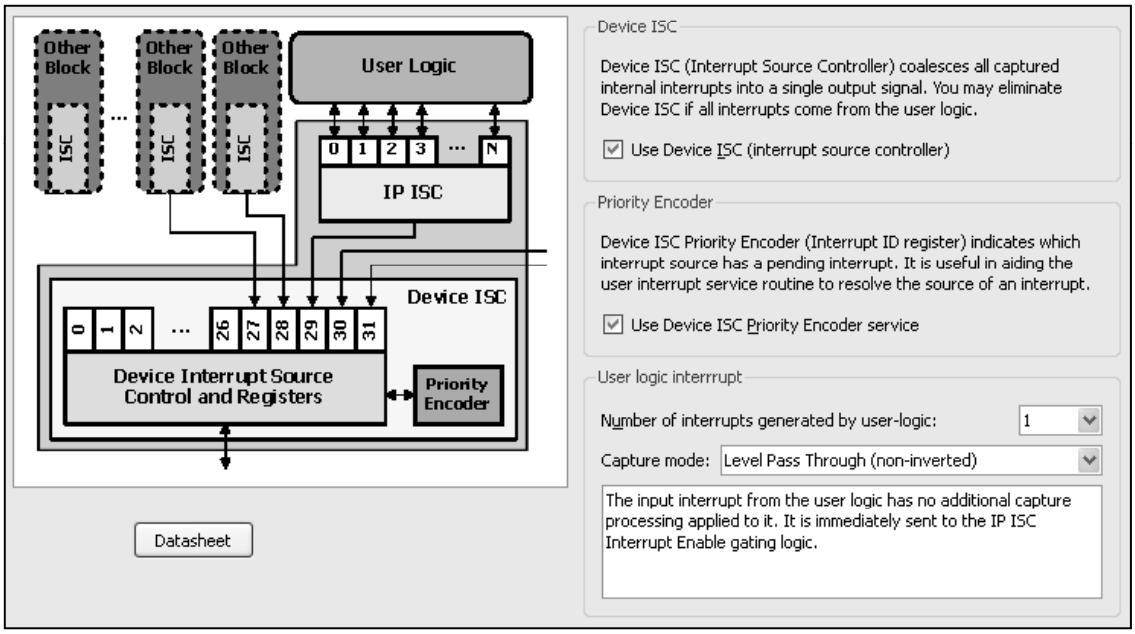
Create and Import Peripheral Wizard:

- **Slave szolgáltatások: megszakítás vezérlő a perifériában**
  - IP modul megszakítások → egy vonal a CPU vagy az INTC felé
  - Vezérlés a PLB IPIF regiszterein keresztül
  - Részletek az adatlapban
- **A megszakítás vezérlő beállításai**
  - Device Interrupt Source Controller: belső IPIF megszakítások kezelése
  - Prioritás enkóder: Device ISC megszakítások azonosítása
  - Felhasználói logika megszakítások száma
  - Felhasználói logika megszakítások feldolgozása
    - Nincs feldolgozás: normál vagy invertált
    - Szint detektálás: alacsony aktív vagy magas aktív
    - Él detektálás: felfutó vagy lefutó
- **Egyszerű perifériák esetén:**
  - Nincs szükség többszintű megszakítás rendszerre
  - Az IRQ vonala(ka)t közvetlenül a CPU-ba vagy az INTC-be vezetjük

Saját periféria létrehozása

Create and Import Peripheral Wizard:

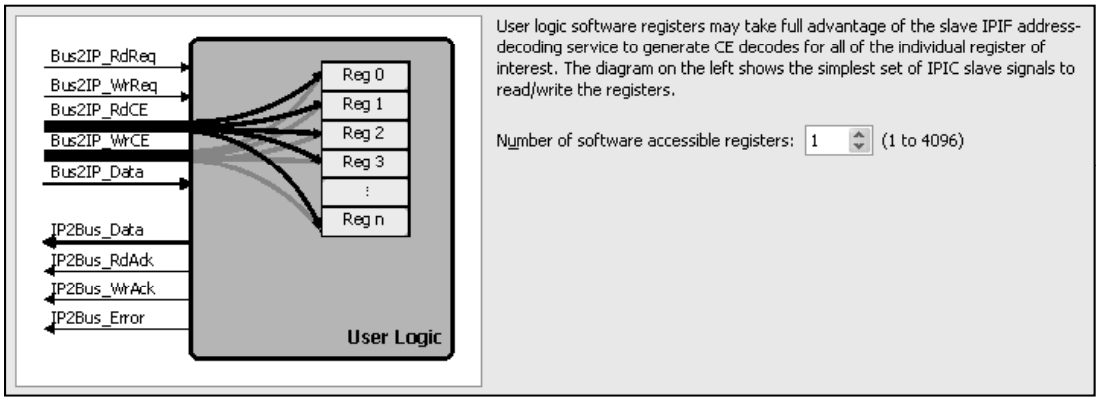
- **Slave szolgáltatások: megszakítás vezérlő**



Saját periféria létrehozása

Create and Import Peripheral Wizard:

- **Slave szolgáltatások: regiszterek**
  - Regiszterek száma: 1 - 4096
  - Címdekódolás az IPIF-ben: minden regiszterhez külön írás (***Bus2IP_WrCE***) és olvasás (***Bus2IP_RdCE***) engedélyező jel



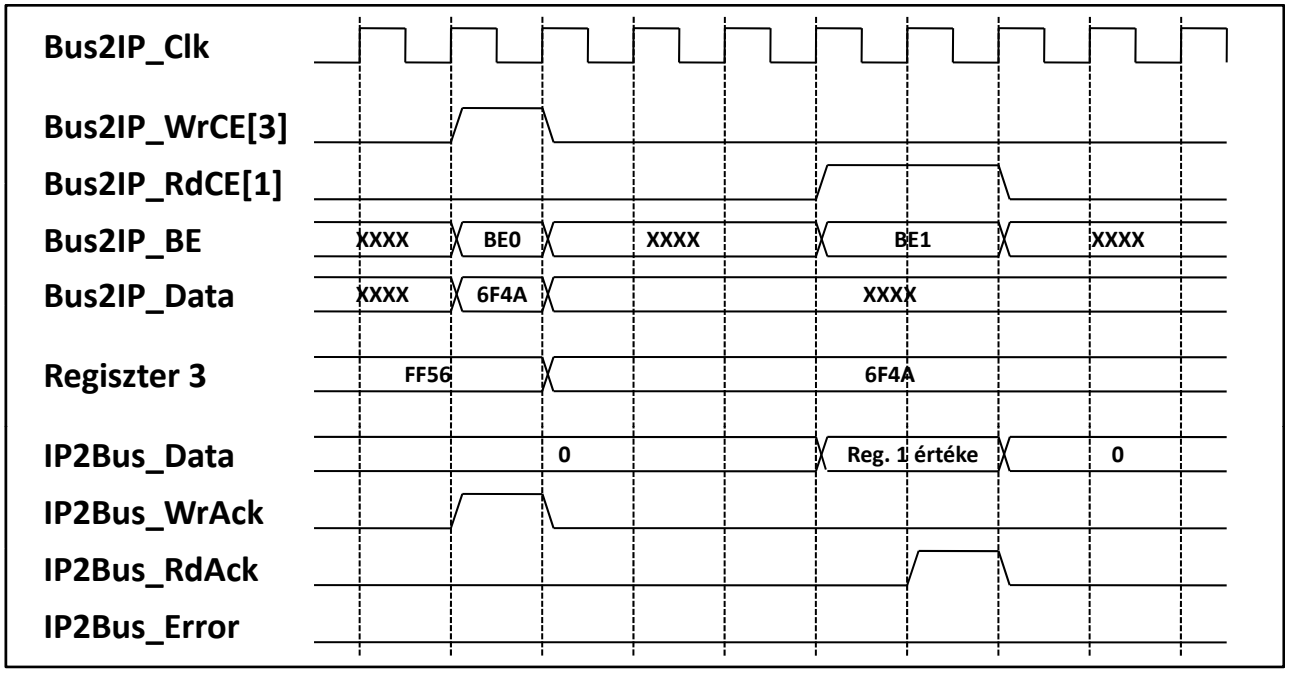
Saját periféria létrehozása

Regiszter interfész:

- **Jelek az IPIF-től a felhasználói modul felé**
  - ***Bus2IP_WrCE***: írás engedélyező jel(ek)
  - ***Bus2IP_RdCE***: olvasás engedélyező jel(ek)
  - ***Bus2IP_BE***: bájt engedélyező jelek
  - ***Bus2IP_Data***: 32 bites írási adatbusz
- **Jelek a felhasználói modultól az IPIF felé**
  - ***IP2Bus_Data***: 32 bites olvasási adatbusz
  - ***IP2Bus_WrAck***: írási műveletek nyugtázó jele
  - ***IP2Bus_RdAck***: olvasási műveletek nyugtázó jele
  - ***IP2Bus_Error***: hiba jelzése

Saját periféria létrehozása

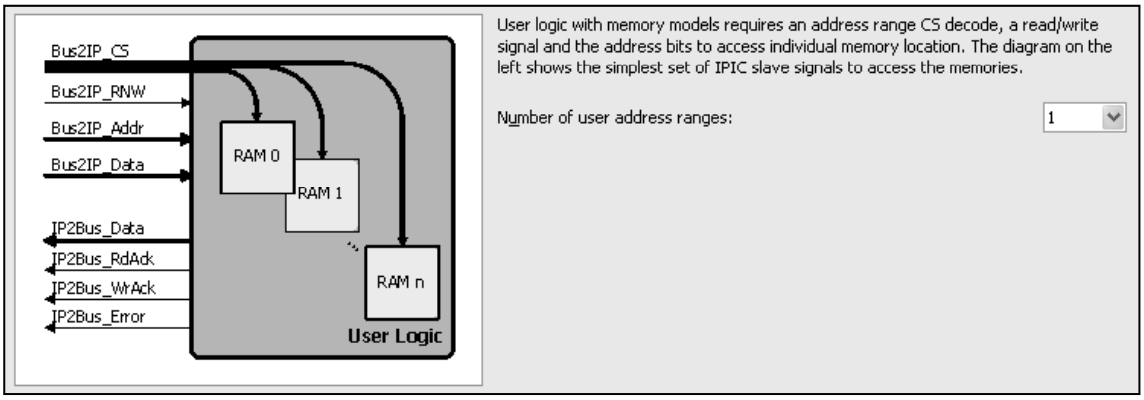
Regiszter interfész: írás és olvasás



Saját periféria létrehozása

Create and Import Peripheral Wizard:

- **Slave szolgáltatások: memória címtartomány**
  - Címtartományok száma: 1 – 8
  - Minden címtartományhoz külön kiválasztó jel (***Bus2IP_CS***)



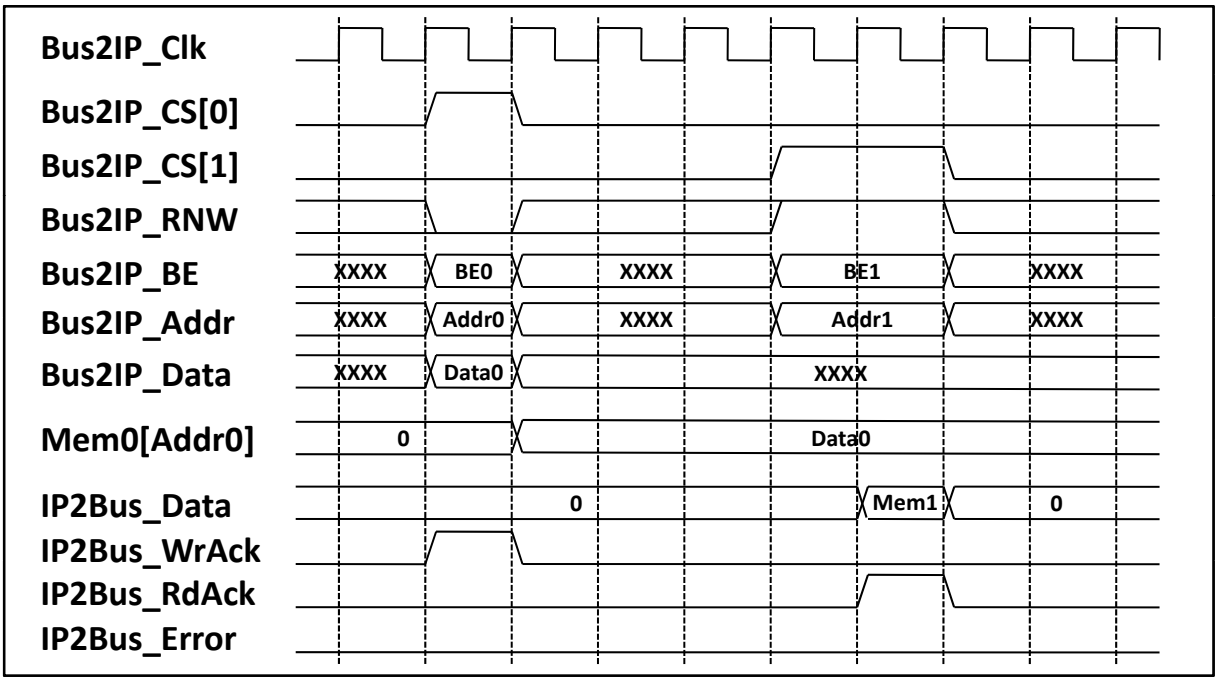
Saját periféria létrehozása

Memória címtartományok:

- **Jelek az IPIF-től a felhasználói modul felé**
  - ***Bus2IP_CS***: memória címtartomány kiválasztó jel(ek)
  - ***Bus2IP_RNW***: írás (0) / olvasás (1) kiválasztó jel
  - ***Bus2IP_Addr***: 32 bites címbusz
  - ***Bus2IP_BE***: bájt engedélyező jelek
  - ***Bus2IP_Data***: 32 bites írási adatbusz
- **Jelek a felhasználói modultól az IPIF felé**
  - ***IP2Bus_Data***: 32 bites olvasási adatbusz
  - ***IP2Bus_WrAck***: írási műveletek nyugtázó jele
  - ***IP2Bus_RdAck***: olvasási műveletek nyugtázó jele
  - ***IP2Bus_Error***: hiba jelzése

Saját periféria létrehozása

Memória címtartományok: írás és olvasás



Saját periféria létrehozása

A bájt engedélyező jelek (*Bus2IP_BE*) értelmezése:

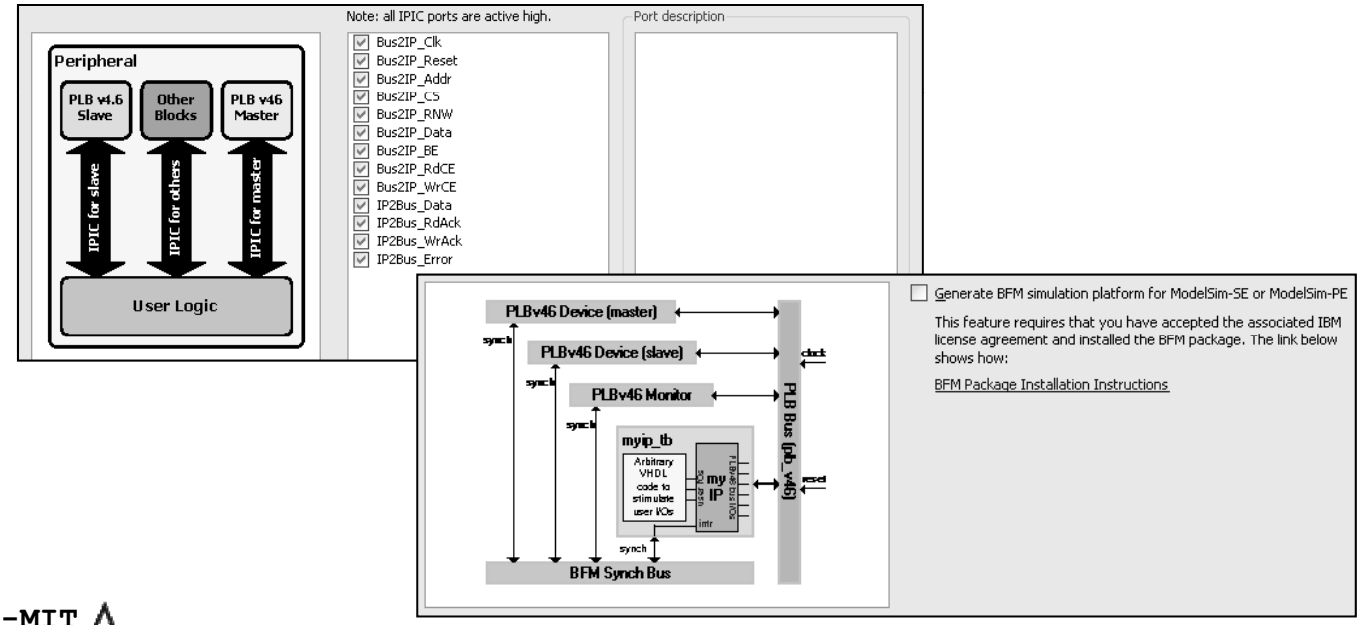
- **Big-Endian formátum, fordított bit indexelés (MSb a 0. bit)**
- **Szavas címzés → az alsó két címbitet (30. és 31.) nem vesszük figyelembe, helyettük vannak a bájt engedélyező jelek**
  - Írásnál: értelmezés az alábbi táblázat szerint
  - Olvasásnál: nincs értelmezve, a processzor rendezi át a bájtokat

		Write Data Bus Bytes			
Byte_Enable [0:3]	Transfer Size	Byte0	Byte1	Byte2	Byte3
0001	byte				rD[24:31]
0010	byte			rD[24:31]	
0100	byte		rD[24:31]		
1000	byte	rD[24:31]			
0011	halfword			rD[16:23]	rD[24:31]
1100	halfword	rD[16:23]	rD[24:31]		
1111	word	rD[0:7]	rD[8:15]	rD[16:23]	rD[24:31]

Saját periféria létrehozása

Create and Import Peripheral Wizard:

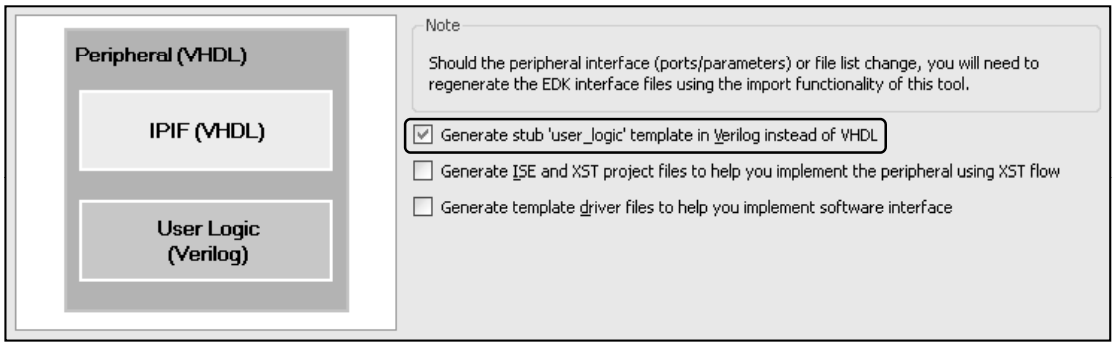
- **IP Interconnect (IPIC) vonalak kiválasztása**
- **Szimulációs modell generálása a perifériához (ModelSim)**



Saját periféria létrehozása

Create and Import Peripheral Wizard:

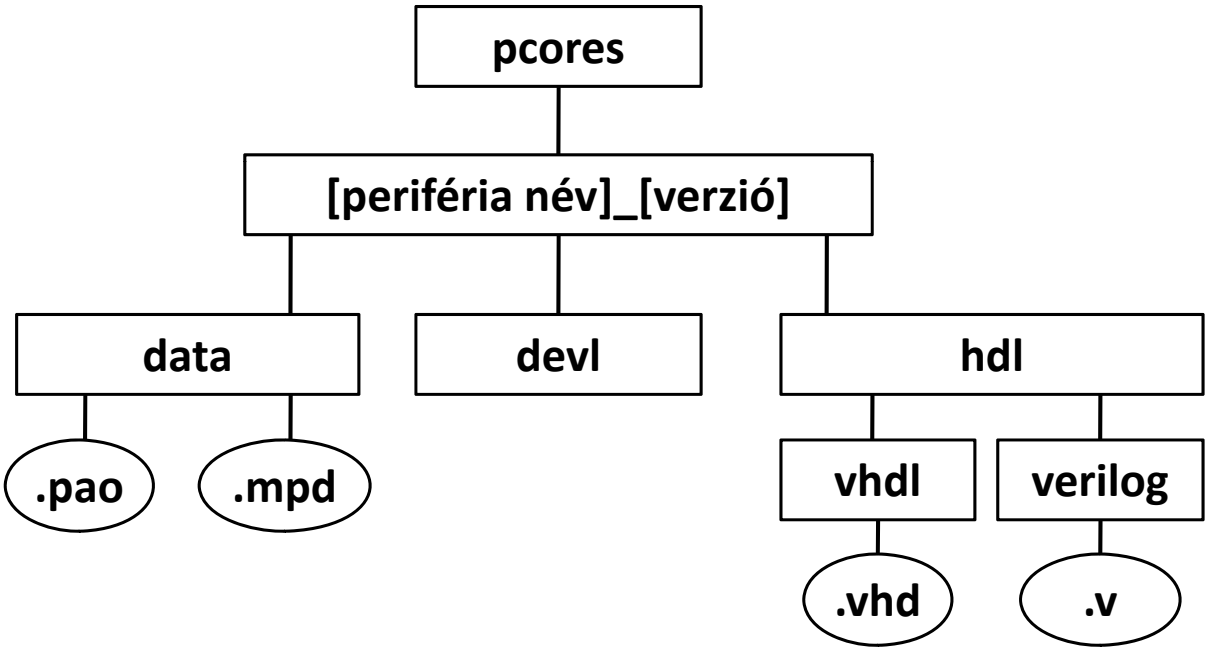
- **Az *user_logic* modul HDL forráskódjának nyelve**
  - VHDL (a periféria top-level modulja mindig VHDL nyelvű)
  - Verilog
- **ISE projekt létrehozása a perifériához**
- **Eszközmeghajtó sablon létrehozása a perifériához**



- **Összegzés a létrehozandó perifériáról**

Saját periféria létrehozása

A perifériákhoz tartozó könyvtárstruktúra:



Saját periféria létrehozása

Microprocesspr Peripheral Description (MPD) fájl:

- **A periféria leírását tartalmazza**
  - Paraméterek és alapértelmezett értékeik
  - Busz interfész(ek)
  - Portok
- **Számunkra lényeges**
  - Paraméterek hozzáadása
  - Portok hozzáadása
    - Normál kimenet, bemenet
    - Háromállapotú kimenet, I/O vonal
    - Megszakításkérő vonal

Saját periféria létrehozása

Microprocesspr Peripheral Description (MPD) fájl:

- **Részletes szintaxis: EDK\doc\usenglish\psf_rm.pdf**
- **Paraméter hozzáadása: *PARAMETER* kulcsszó**
  PARAMETER név = alapértelmezett_érték
  - Az egyes opciókat vesszővel kell elválasztani
  - Adattípus megadása
    DT = integer, real, string, stb.
  - Értéktartomány megadása
    - Tartomány: ***RANGE = (alsó érték:felső érték)***
    - Felsorolás: ***RANGE = (8,16,32,64)***
    - Vegyes: ***RANGE = (1:4,8,16)*** → 1, 2, 3, 4, 8, 16

Saját periféria létrehozása

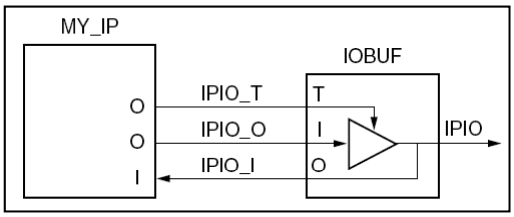
Microprocesspr Peripheral Description (MPD) fájl:

- **Port hozzáadása: *PORT* kulcsszó**
  PORT név = ""
  - Az egyes opciókat vesszővel kell elválasztani
  - Irány megadása
    DIR = I, O vagy ***IO***
  - Szélesség megadása (A és B: nemnegatív egészek)
    VEC = [A:B]
  - Megszakításkérő kimenet (1 bites kimeneti port esetén)
    SIGIS = INTERRUPT
    SENSITIVITY = EDGE_RISING, (felfutó élre)
    EDGE_FALLING, (lefutó élre)
    LEVEL_HIGH, (magas szintre)
    LEVEL_LOW (alacsony szintre)

Saját periféria létrehozása

Microprocesspr Peripheral Description (MPD) fájl:

- **Port hozzáadása: *PORT* kulcsszó**
  PORT név = ””
  - Háromállapotú kimenet (kimeneti vagy I/O port esetén)
    THREE_STATE = TRUE
    - Közös kimenet engedélyező jel: ***ENABLE = SINGLE***
    - Egyedi kimenet engedélyező jel: ***ENABLE = MULTI***
  - Háromállapotú kimenet vagy I/O esetén a modul portjai
    - Kimenet: ***[port név]_O***
    - Bemenet: ***[port név]_I*** (csak I/O port esetén)
    - Kimenet engedélyezés: ***[port név]_T***
      - ALACSONY AKTÍV JEL(EK)!
      - Közös: 1 bites
      - Egyedi: a port szélességével egyező vektor



Saját periféria létrehozása

Microprocesspr Peripheral Description (MPD) fájl:

```
## Saját paraméterek
PARAMETER PORT_WIDTH = 8, DT = INTEGER, RANGE = (1:32)
PARAMETER CLK_FREQ = 50000000, DT = INTEGER

## Megadható szélességű I/O port (egyedi kimenet engedélyezés)
## output wire [PORT_WIDTH-1:0] my_io_O
## input  wire [PORT_WIDTH-1:0] my_io_I
## output wire [PORT_WIDTH-1:0] my_io_T
PORT my_io = ””, DIR = O, VEC = [(PORT_WIDTH-1):0], THREE_STATE = TRUE,
ENABLE = MULTI

## 4 bites normál bemenet
## input  wire [0:3] my_input
PORT my_input = ””, DIR = I, VEC = [0:3]

## Megszakításkérő kimenet (megszakítás felfutó élre)
## output wire my_irq
PORT my_irq = ””, DIR = O, SIGIS = INTERRUPT, SENSITIVITY = EDGE_RISING
```

Saját periféria létrehozása

Peripheral Analyze Order (PAO) fájl:

- **A szintézishez szükséges fájlok listáját tartalmazza**
- **Saját HDL fájl hozzáadása**
  lib [könyvtár név] [fájl név] [nyelv]
  - Könyvtár név: esetünkben a periféria neve és verziója
  - Fájl név: a HDL forrásfájl neve (a kiterjesztés opcionális)
  - Nyelv: vhdl vagy verilog

```
lib plbv46_slave_single_v1_01_a plbv46_slave_single vhdl
lib my_peripheral_v1_00_a user_logic verilog
lib my_peripheral_v1_00_a my_peripheral vhdl

## Saját HDL forrásfájlok
lib my_peripheral_v1_00_a fifo verilog
lib my_peripheral_v1_00_a ps2_interface vhdl
```


Saját periféria létrehozása (példa)

GPIO periféria az alábbi paraméterekkel:

- **A port szélessége legyen paraméterben megadható**
  - 1 és 32 bit közötti érték
- **Az egyes bitekhez tartozó kimeneti meghajtók legyenek egyedileg engedélyezhetők, illetve tilthatók**
- **Megszakításkérés, ha megváltozik egy bemenet értéke**
- **Regiszterkészlet: 32 bites regiszterek**

Bázis + 0x00	**R/W**	**Adatregiszter: a kimeneten megjelenő adat**
Bázis + 0x04	**R/W**	**Irányregiszter: a kimenetek engedélyezése**
Bázis + 0x08	**R**	**Az I/O kábak aktuális értéke**
Bázis + 0x0C	**R/W**	**Megszakítás engedélyező regiszter (bitenként)**

Saját periféria létrehozása (példa)

A periféria generálása a varázslóval:

- **Beállítások**
  - Slave szolgáltatások
    - Regiszter: 4 darab regiszter szükséges
    - A többi szolgáltatás (FIFO, memória, stb.) nem kell
    - Burst adatátvitelre nincs szükség
  - Master szolgáltatások: nem kell
  - A ***user_logic*** modul nyelve: Verilog
- **Adjuk hozzá a létrehozott perifériát a rendszerhez**
  - Így szerkeszteni tudjuk a szükséges fájlokat az XPS-ből.
  - A busz és a portok bekötése majd később
- **Módosítani kell**
  - A ***user_logic.v*** fájlt
  - A ***[periféria név].vhd*** fájlt (top-level modul)
  - A ***[periféria név]_v2_1_0.mpd*** fájlt

Saját periféria létrehozása (példa)

A user_logic.v fájl módosítása:

- **Jobb kattintás a periféria nevére → Browse HDL Sources…**
- **Modul fejléc: saját portok megadása**

```
module user_logic (
   // -- ADD USER PORTS BELOW THIS LINE ---------------
   gpio_O,
   gpio_I,
   gpio_T,
   irq,
   // -- ADD USER PORTS ABOVE THIS LINE ---------------
```

- **Saját paraméterek megadása**

```
// -- ADD USER PARAMETERS BELOW THIS LINE ------------
parameter GPIO_WIDTH = 8;
// -- ADD USER PARAMETERS ABOVE THIS LINE ------------
```

Saját periféria létrehozása (példa)

A user_logic.v fájl módosítása:

- **Saját portok típusának és szélességének definiálása**

```
// -- ADD USER PORTS BELOW THIS LINE -----------------
output reg  [GPIO_WIDTH-1:0] gpio_O;
input  wire [GPIO_WIDTH-1:0] gpio_I;
output reg  [GPIO_WIDTH-1:0] gpio_T;
output wire                  irq;
// -- ADD USER PORTS ABOVE THIS LINE -----------------
```

- **A nyugtázó- és hibajelek meghajtása: nem kell késleltetés**
  - A beírandó adatot azonnal be tudjuk írni a regiszterbe
  - A beolvasandó adat azonnal rendelkezésre áll (regiszter olvasás)

```
//A nyugtázó- és hibajelek meghajtása.
assign IP2Bus_WrAck = |Bus2IP_WrCE;
assign IP2Bus_RdAck = |Bus2IP_RdCE;
assign IP2Bus_Error = 0;
```


Saját periféria létrehozása (példa)

A user_logic.v fájl módosítása:

- **Az írási adatbusz bit sorrendjének megfordítása**
  - Célszerű a fordított bit indexelés miatt (MSb a 0. bit, LSb a 31. bit)

```
//Az írási adatbusz bitjeinek megfordítása
reg [C_SLV_DWIDTH-1:0] wr_data;
integer i;

always @(*)
   for (i = 0; i < C_SLV_DWIDTH; i = i + 1)
      wr_data[i] <= Bus2IP_Data[C_SLV_DWIDTH-i-1];
```

- **Az adatregiszter: a *gpio_O* port *reg* típusú, csak 32 bites írás**

```
//A GPIO port adatregisztere
always @(posedge Bus2IP_Clk)
   if (Bus2IP_Reset)
      gpio_O <= 0;
   else
      if (Bus2IP_WrCE[0] && (Bus2IP_BE == 4'b1111))
         gpio_O <= wr_data[GPIO_WIDTH-1:0];
```

Saját periféria létrehozása (példa)

A user_logic.v fájl módosítása:

- **Az irányregiszter: a *gpio_T* port *reg* típusú, csak 32 bites írás**

```
//A GPIO port irányregisztere (reset: minden vonal bemenet).
always @(posedge Bus2IP_Clk)
   if (Bus2IP_Reset)
      gpio_T <= 32'hffffffff;
   else
      if (Bus2IP_WrCE[1] && (Bus2IP_BE == 4'b1111))
         gpio_T <= ~wr_data[GPIO_WIDTH-1:0];
```

- **A megszakítás engedélyező regiszter: csak 32 bites írás**

```
//A megszakítás engedélyező regiszter.
reg [GPIO_WIDTH-1:0] irq_enable;

always @(posedge Bus2IP_Clk)
   if (Bus2IP_Reset)
      irq_enable <= 0;
   else
      if (Bus2IP_WrCE[3] && (Bus2IP_BE == 4'b1111))
         irq_enable <= wr_data[GPIO_WIDTH-1:0];
```


Saját periféria létrehozása (példa)

A user_logic.v fájl módosítása:

- **A bemenet szinkronizálása: mert aszinkron az órajelhez képest**
- **A megszakításkérő jelzés előállítása**
  - Az adott bit bemenet
  - Az adott bitre engedélyezve van a megszakítás
  - Az adott bit értéke megváltozott

```
//A bemenet szinkronizálása (mert aszinkron az órajelhez képest).
reg [GPIO_WIDTH-1:0] sample0, sample1, sample2;

always @(posedge Bus2IP_Clk)
   if (Bus2IP_Reset)
      {sample2, sample1, sample0} <= 0;
   else
      {sample2, sample1, sample0} <= {sample1, sample0, gpio_I};

//A megszakításkérő jelzés előállítása (megszakítás felfutó élre).
assign irq = |((sample1 ^ sample2) & gpio_T & irq_enable);
```

Saját periféria létrehozása (példa)

A user_logic.v fájl módosítása:

- **Az olvasási adatbusz meghajtása: ha nincs olvasás, akkor értéke 0**

```
//Az olvasási adatbusz meghajtása.
reg [0:C_SLV_DWIDTH-1] rd_data;

always @(*)
   case (Bus2IP_RdCE)
      4'b1000: rd_data <= gpio_O;
      4'b0100: rd_data <= ~gpio_T;
      4'b0010: rd_data <= sample1;
      4'b0001: rd_data <= irq_enable;
      default: rd_data <= 0;
   endcase

assign IP2Bus_data = rd_data;

endmodule
```


Saját periféria létrehozása (példa)

A top-level modul (VHDL fájl) módosítása:

- **Az alábbi módosításokat két helyen kell végrehajtani**
  - ***entity [periféria név] is…***
  - ***componenet user_logic is…***
- **Paraméterek hozzáadása**

```
-- ADD USER GENERICS BELOW THIS LINE ---------------
GPIO_WIDTH : integer := 8;
-- ADD USER GENERICS ABOVE THIS LINE ---------------
```

- **Portok hozzáadása**

```
-- ADD USER PORTS BELOW THIS LINE ------------------
gpio_O : out std_logic_vector(GPIO_WIDTH-1 downto 0);
gpio_I : in  std_logic_vector(GPIO_WIDTH-1 downto 0);
gpio_T : out std_logic_vector(GPIO_WIDTH-1 downto 0);
irq    : out std_logic;
-- ADD USER PORTS ABOVE THIS LINE ------------------
```

Saját periféria létrehozása (példa)

A top-level modul (VHDL fájl) módosítása:

- **Az alábbi módosításokat egy helyen kell végrehajtani**
  - ***USER_LOGIC_I : component user_logic…***
- **Paraméterek leképzése**

```
-- MAP USER GENERICS BELOW THIS LINE ---------------
GPIO_WIDTH => GPIO_WIDTH,
-- MAP USER GENERICS ABOVE THIS LINE ---------------
```

- **Portok leképzése**

```
-- MAP USER PORTS BELOW THIS LINE ------------------
gpio_O => gpio_O,
gpio_I => gpio_I,
gpio_T => gpio_T,
irq    => irq,
-- MAP USER PORTS ABOVE THIS LINE ------------------
```


Saját periféria létrehozása (példa)

Az MPD fájl módosítása:

- **Paraméterek hozzáadása**
  - Közvetlenül a már meglévő paraméterek elé írjuk be

```
## Saját paraméterek
PARAMETER GPIO_WIDTH = 8, DT = INTEGER, RANGE = (1:32)
```

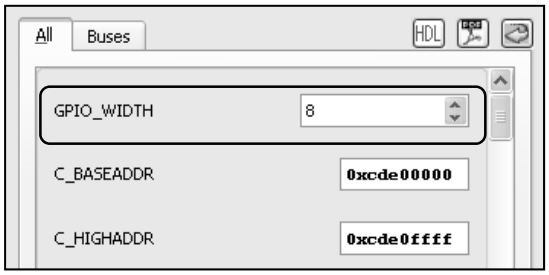
- **Portok hozzáadása**
  - Közvetlenül a már meglévő portok elé írjuk be

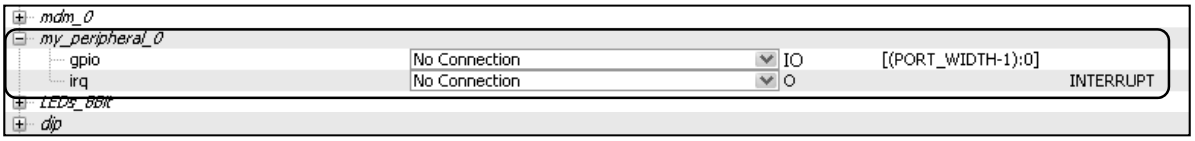
```
## Saját portok
PORT gpio = "", DIR = IO, VEC = [(PORT_WIDTH-1):0], THREE_STATE = TRUE,
ENABLE = MULTI
PORT irq = "", DIR = O, SIGIS = INTERRUPT, SENSITIVITY = EDGE_RISING
```

Saját periféria létrehozása (példa)

A saját periféria beillesztése a rendszerbe:

- **A módosítások figyelembe vétele**
  - ***Project*** menü → ***Rescan User Repositories*** vagy
  - A gomb a toolbar-on
- **Ezután a szokásos módon lehetséges:**
  - Paraméterek beállítása
  - Csatlakoztatás a buszra
  - Portok bekötése
  - Címek kiosztása





Perifériák tesztelése

Xilinx Microprocessor Debug (XMD) alkalmazás:

- **Alapfunkciók tesztelése: memória írás és olvasás**
- **Az XMD indítása:**
  - ***Debug*** menü → ***Launch XMD…*** vagy
  - A gomb a toolbar-on
- **Első indításnál: a debug opciók beállítása**
  - Az alapértelmezett beállítások megfelelőek
  - ***Connection type***: Hardware
  - ***JTAG Cable***: Auto
  - ***Auto-Discover JTAG Chain Definition***: engedélyezve

Perifériák tesztelése

Fontosabb Xilinx Microprocessor Debug (XMD) parancsok:

- **Memória írása:**
  mwr [cím] [adat] <w|h|b>
  - Adatformátum: w (32 bit, alapértelmezett), h (16 bit), b (8 bit)
  - A címet az adatformátumnak megfelelő határra kell igazítani
- **Memória olvasása:**
  mrd [cím] <olvasások_száma> <w|h|b>
  - Az olvasások számának megadása nem kötelező
  - Több olvasásnál a cím növekszik az adatformátum szerint
- **Programkód letöltése: *dow [elf_fájl_név]***
  - Az elérési út megadásakor a \ karakter helyett a / karakter kell
- **A processzor elindítása: *run***
- **A processzor leállítása: *stop***
- **A rendszer alapállapotba hozása: *rst***
- **Kilépés az XMD programból: *exit***