The Xilinx EDK Toolset: Xilinx Platform Studio (XPS)

Building a Base System Platform

By Jason Agron

What is Xilinx EDK?

- EDK = Embedded Development Kit.
- It is a set of tools used to build embedded processing systems.
 - i.e. Systems-On-Chip (SoCs).
 - Processors (MicroBlaze, PowerPC).
 - Interconnect (PLB, OPB, FSL, Custom, etc.).
 - Memories (BRAM, DDR).
 - Peripherals (UART, GPIO, Ethernet, Custom, etc.).
- Provides a single environment for...
 - Simulation
 - Synthesis.
 - Compilation.

How Do I Use Xilinx EDK?

- Xilinx Platform Studio (XPS) the actual tool.
 - Design flow...
 - First, create the hardware platform.
 - Select all of the peripherals.
 - Connect all of the peripherals.
 - Second, create the software for the platform.
 - > Write SW to "make things work".
 - Iterate if needed.
- The FPGA has a malleable fabric...
 - So both SW and HW are flexible and can be changed...
 - At "compile-time".
 - At "run-time" (dynamic reconfiguration).

How To Get Started

- Open up XPS.
- Create a new project.
 - Select "File", "New Project"
 - Select "Base System Builder..."
 - Provides a wizard to help get basic system established.
 - Click OK.

XPS - Getting Started





XPS - New Project Creation

📥 Applications 🛛 Actions 🔍 🗏 🌀 🙋	🜗 Tue May 29, 1:30 PM 🔌
Xilinx Platform Studio - no project opened	- • ×
Eile Edit View Project Hardware Software Device Configuration Debug Simulation Window Help	
Applications Project Software Project So	
≤ Output Warnings Errors	<u>م</u> ح
💿 🕅 i agron@chara khan:/399lahe/ 🔲 i agron@chara khan:/399lahe/ 📄 i agron@chara khan:/399lahe/ 🚔 Viliny Distrom Studiono proje	

XPS - Creating The Base System

¥	Create New XPS Project using BSE	3 Wizard 🛛 🗙
New	project	
<u>P</u> roj	ect file	
lser	s/jagron/388labs/newProject/system.xmp	<u>B</u> rowse
Adva	anced options (optional: F1 for help)	
Γι	Jse Repository paths	
		Provisio
]		BI0 <u>W</u> 5C
	ОК	Cancel

- Now, create a directory for this EDK project.
 - Saved as a .xmp file.
- IMPORTANT NOTE!!!!
 - Make sure that the absolute path contains no spaces!!!!

XPS - Base System Builder

- The Base System Builder window will open.
- Select "Create a New Design…".
- Now we can select the base components of our custom SoC.

×	Base System Builder - Welcome	×
Embe Platf	dded Development Kit	
Welcome to th	e Base System Builder!	
This tool will lead y	you through the steps necessary to create an embedded sys	tem.
Please begin by	selecting one of the following options:	
I would like to	create a new <u>d</u> esign	
C I would like to	load an existing .bsb settings file (saved from a previous set	ession)
	Bro	wse
Mare Info	< Rack Next >	Cancel
More milo	< Dank Mext >	Zalicei

XPS - Board Selection

- We must select the platform we wish to use.
- In our case it is...
 - Vendor = Xilinx.
 - Board = XUP.
 - Rev # = C

Base System Builder - Select Board Select a target development based	1
Serect a target development board:	
Select board	1
Paret used as	1
Board vendor: Xilinx	1
Board name: XUP Virtex-II Pro Development System	1
Board revision: C	I
Note: Visit the vendor website for additional board support materials.	I
Vendor's Website Contact Info	I
C I would like to create a system for a custom board	I
	I
Board description	I
The XUP Virtex-II Pro Development System provides an advanced hardware platform that consists of a high performance Virtex-II Pro Platform FPGA	1
surrounded by a comprehensive collection of peripherals that can be used to	I
create a complex system and to demonstrate the capability of the Virtex-II Pro Platform FPGA.	1
	ł
1	l
	l
More Info < Back Next > Cancel	

XPS - Processor Selection

- Choice as to which processor to use in our SoC.
- PowerPC:
 - PPC405 Hard Core.
 - Physical CPU embedded within FPGA fabric.
- MicroBlaze:
 - Soft core.
 - Must be synthesized (implemented using the FPGA fabric).
- We will use the MicroBlaze.

Base System Builder - Select Processor
The board you selected has the following FPGA device:
Architecture: Device: Package: Speed grade:
virtex2p 👻 xc2vp30 👻 ff896 👻 -7 👻
Calest the presence you would like to use in this design:
Processors
MicroBlage Dips D
PowerPC
Crit Sroa
Processor description
The MicroBlaze(TM) 32-bit soft processor is a RISC-based engine with a 32
register by 32 bit LUT RAM-based Register File, with separate instructions for data and memory access. It supports both on-chip BlockRAM and/or external memory.
All peripherals are implemented on the FPGA fabric and operate off the on-chip
penpherai bus (OPB).
More Info < Back Next > Cancel

XPS - Processor Configuration

- Choose customizable CPU settings.
- In our case...
 - Bus freq. = 100 MHz.
 - Debug I/F = On-chip.
 - Local mem. = 16 KB.
 - No cache.
 - Disabled FPU.
- Simple, but highly effective.

~	3ase System Builder - Configure MicroBlaze
MicroBlaze	
System wide setting	15
Beference clock	Processor-Bus clock
frequency:	frequency
100.00 MHz	100.00 MHz
Beset polarity:	Active LOW 👻
Processor configura	tion
Debug I/F	
C On-chip H/W d	ahun medula
. Zuenbring	and mornie
C XMD with S/W	debug stub
C No debug	
Micro	Local memory Data and Instruction: (Use BRAM) 16 KB
Cache setup	
No Cache	C Enable cache link
Enable floating p	oint unit (EPU)
More Info	< Back Next > Cancel

XPS - I/O Interface Configuration

- Choose from available I/O interfaces.
- Ethernet.
 - allows boards to be networked.
- RS232 UART.
 - Serial protocol I/O for the board.
- In our case...
 - No Ethernet.
 - RS232 + UARTLite.
 - 9600 Baud. No parity
 - 8 Data bits.

Base System Builder - Configure IO	Interfaces ×
The following external memory and IO devices were found	on your board:
Xilinx XUP Virtex-II Pro Development System Revision C	
Please select the IO devices which you would like to use:	
10 devices	
RS232_Uart_1	Data Sheet
Peripheral: OPB UARTLITE	Terre outer
Baudrate (bits per seconds): 9600	
Data bit <u>s</u> : 8 Parit <u>y</u> : NONE	
Le interrupt	
Ethernet_MAC	Data Sheet
	Note
More Info < Back	Next > Cancel

XPS - I/O Interface Configuration

- Configure additional I/O interfaces.
- SysACE.
 - Allows for file storage.
- General Purpose I/O.
 - GPIO.
 - Used for LEDs, switches, and push buttons.
- In our case...
 - Only use the GPIOs for LEDs, DIPSWs, and PushButtons (No SysACE).

Base System Builder - Configure Additional 10 Interface	es 🗵
The following external memory and IO devices were found on your board:	
Xilinx XUP Virtex-II Pro Development System Revision C	
Please select the IO devices which you would like to use:	
-IO devices	
SysACE_CompactFlash	sta Sheet
Peripheral: OPB GPIO	ata Sheet
Use interrupt DIPSWs_4Bit Peripheral: OPB GPIO	ata Sheet
☐ Use interrupt	
Peripheral: OPB GPIO	ata Sheet
☐ Use interrupt	
More Info < Back Next >	Cancel

XPS - I/O Interface Configuration

- Configure additional memory interfaces.
- Different types of external memory.
- Often times DDR.
 - Large amount of storage.
 - Cheap.
 - Fast.
- For this system...
 - No external memory.

Base System Builder - Configure Additional IO Interfaces	×
The following external memory and IO devices were found on your board:	
Xilinx XUP Virtex-II Pro Development System Revision C	
Please select the IO devices which you would like to use:	
10 devices	- I
DDR_512MB_64Mx64_rank2_row13_col10_cl2_5	
Note	
DDR_512MB_64MX64_rank1_row13_col11_cl2_5	
DDR_256MB_32MX64_rank1_row13_col10_cl2_5	
Note	
DDR_128MB_16MX64_rank1_row13_col9_cl2_5	
	1
More Info < Back Next > Cance	el

XPS - Add Internal Peripherals

- Xilinx provides a large library of peripherals:
 - I/O
 - Debug
 - Busses
 - Memory
 - Timers
 - Interrupts
 - A/D
- In our case...
 - We will add internal peripherals at a later time.

~	Rat	e System Ruil	der - Add Inte	arnal Parin	herals		1
	Das	e system sun	uer - Auu ino	ernar r erip	merars		
Add othe	peripherals the	at do not intera	ct with off-chip he list of swai	o componer	nts. Use	the	
Huguda	not wish to ad	d any non IO n	vinharale alla	k the "Next	renaro.		
n you do	not wish to as	u any non-to p	mprierais, ciic	K LITE INCA	barron.		1
					Ad	d Peripheral	
Periphe	rals						٦
More In	lo		< Bac	k N	ext >	Cancel	
							-

XPS - Software Setup

- Configuration of softwarerelated properties of the system.
- I/O Configuration:
 - STDIN = UART.
 - STDOUT = UART.
- Include sample applications:
 - Memory test.
 - Peripheral self-test.
- In our case...
 - Use the defaults.

Base System Bullder - Software Setup	×
Devices to use as standard input and standard output	
STDIN: RS232_Uart_1	
STDQUT: RS232_Uart_1	
Sample application selection	
Select the sample C application that you would like to have generated. Each application will include a linker script.	
P Memory test	
Illustrate system aliveness and perform a basic read/write test to each memory in your system	
Peripheral selftest	
Perform a simple self-test for each peripheral in your system.	
More Info	J

XPS - Application Configuration

- Configuration of application-related properties of the system.
- Choose where instructions and data are stored in the system.
- In our case...
 - Our system is simple a single memory for instructions and a single memory for data.
 - Use the defaults.

Base System Builder - Configure Memory Test Application
The simple Memory Test application will illustrate system aliveness and perform a basic read/write test to your memory devices.
MemoryTest
Select the memory devices which will be used to hold the following program sections:
Instruction: ilmb_cntlr
Data: dimb_entir
Stack/Heap: dlmb_cntlr
Boot section resides in ilmb_cnt/r
WARNING If you have placed the Instruction or Data section of this program in an external memory, you must use a debugger, bootloader, or ACE file to initialize memory before you can run this program!
More Info < Back Next > Qancel

XPS - System Created

- System configuration is complete.
- Displays all of the included components and their associated address spaces.
- What are *address spaces*?
- Why is knowing the address space of a device useful?
- Click "Generate"...



XPS - BSB Complete

- Congratulations!!!!
 - You have just created a custom SoC!
- Now click on "Finish", and you can begin...
 - Using the system.
 - Developing custom HW and SW for the system.
- BSB has just generated a .mhs file for your system.
 - A file that lists all components and how they are configured and connected.
 - This file can be translated directly to VHDL or Verilog, and synthesized to the FPGA.



XPS - Project Tab

interpretations Actions 🍣 🖉 🎯 🙋							Tue May 29, 1:35 PM 🛛 🄇
	Xilinx Platform	Studio - /users/jagron/	388labs/newProject/	/system.xmp -	[System Asse	mbly View1]	
De Eile Edit View Project Hardware Software Dev	vice <u>C</u> onfiguration	<u>D</u> ebug Si <u>m</u> ulation <u>W</u>	(indow <u>H</u> elp				_ 8
ID 🖻 🖥 🕹 🛛 🖥 📬 🖬 🗢 🗠 🗶 🖻 🕅	6 🛛 🖻 🕫 😼 🛛] 🛛 🛛 🛤 🔡 🏀 🗍 🖁	Z 🗠 📥 🐚 🏤 🗍	離 🎥 🛛 🐹 🚿	\$ 🛛 🖻 🕅 🗍	ΣΣ [5 8 0 5 12 12	
X		Filters		1			
Applications Project IP Catalog	O L L	🖱 🖲 Bus Interface 🤇	Ports C Addresses	Sonnection 📲	on Filters		
Platform	BBB	Name	Bus Connection	P Type	IP Version		
Project Files	• • • •	t microblaze 0	1240 00111011011	nicroblaze	5.00 c		
-MHS File: system.mhs			0	pb v20	1.10.c		
MSS File: system.mss			In	nb v10	1.00.a		
UCF File: data/system.ucf		🕀 🗢 dimb	In	nb_v10	1.00.a		
-iMPACT Command File: etc/download.cmd		■ >debug_module	o	pb_mdm	2.00.a		
-Implementation Options File: etc/fast_runtime			In	nb_bram_if_cntl	lr 2.00.a		
Bitgen Options File: etc/bitgen.ut			In	nb_bram_if_cntl	lr 2.00.a		
Project Options		⊕ • ● RS232_Uart_1	o	pb_uartlite	1.00.b		
- Device: xc2vp30ff896-7		■ ◆LEDs_4Bit	o	pb_gpio	3.01.b		
-Netlist: TopLevel		■ ◆DIPSWs_4Bit	o	pb_gpio	3.01.b		
-Implementation: XPS			t o	pb_gpio	3.01.b		
-HDL: VHDL		⊞ • ≫lmb_bram	b	ram_block	1.00.a		
Sim Model: BEHAVIORAL		± • ≪dcm_0	d	cm_module	1.00.a		
⊡ Reference Files							
-Log Files							
Synthesis Report Files							
	Svetom Asse	mbly View1					
	Gr System Asse	Indiy viewi					
×							-
							F
Output Warnings Errors							
Utput Warnings Errors							

XPS - IP Catalog Tab

📤 Applications 🛛 Actions 🔍 🗏 🍞 💋						🕕 Tue May 29, 1:35 PM	٩
Xilinx Platform	Studio - /users/jagron/	388labs/newProje	ct/system.xmp - [System Ass	embly View1]	-) 🗆 🗙
De Edit View Project Hardware Software Device Configuration	n <u>D</u> ebug Si <u>m</u> ulation <u>N</u>	<u>í</u> indow <u>H</u> elp				-	-1립>
	I 🛛 🛛 🗛 🔛 🏀 🗍	Z 🗠 🛓 💁 🏫	「 新 辞 🐹 🂥	: 🛛 🖻 🛍 🗍	ΣΣ]6800 🗖 🕅		
	Filters		_				
Applications Project IP Catalog OLL PMM	😇 🤄 Bus Interface 🤇	Ports C Addresse	s 🖹 Connection	n Filters			
	Name	Bus Connection	IP Type	IP Version			
Name Version	⊕- ≪microblaze_0		microblaze	5.00.c			
	⊕- ≫ mb_opb		opb_v20	1.10.c			
	l ⊕- ≫ilmb		lmb_v10	1.00.a			
H-Bus Bridge	⊕- ⊘ dImb		lmb_v10	1.00.a			
T Clock Control	⊞-≪debug_module		opb_mdm	2.00.a			
E Communication High-Speed	⊞-≪dimb_cntir		Imb_bram_if_cntl	r 2.00.a			
The Communication Low-Speed	⊞- ⊘ ilmb_cntlr		Imb_bram_if_cntl	r 2.00.a			
	⊕- ◇ RS232_Uart_1		opb_uartlite	1.00.b			
E Debug	⊕ ≪LEDs_4Bit		opb_gpio	3.01.b			
EPGA Reconfiguration	⊕-		opb_gpio	3.01.b			
⊕ General Purpose IO	⊕ • ● PushButtons_5Bi	t	opb_gpio	3.01.b			
B-Interrupt Control	🗄 🗢 Imb_bram		bram_block	1.00.a			
Memory Block	±-∽dcm_0		dcm_module	1.00.a			
Memory Controller							
⊕-PCI							
Peripheral Controller							
⊕-Processor							
· Reset Control							
+ Timer							
La Later Ass	embly View1						
							<u></u>
							Ŧ
Output Warnings Errors							
							5
🐗 🗍 🗋 jagron@shere-khan:~/388labs/basicMB 🛛 jagron@shere-	khan:~/388labs/basicME	jagron@she	ere-khan:~/388lab	s/basicMB/mi	🗢 Xilinx Platform Studio - /users,	/jagron/388la 🔤 🗌 🔛	

XPS - Applications Tab

📤 Applications Actions 🍣 🗏 🍞 🙋					🕕 Tue May 29, 1:36 PM 🔇
	Xilinx Platform Studio - /users/jagron/3	88labs/newProject/system.xmp - [System Ass	embly View1]	
De Eile Edit View Project Hardware Software Dev	vice <u>C</u> onfiguration <u>D</u> ebug Si <u>m</u> ulation <u>W</u> i	ndow <u>H</u> elp			<u></u>
🗋 🤌 🔚 🦆 🕼 👘 🗳 🗠 🎮 🔏 🗎 🕼	🛝]] 🖻 🗝 🛃 🔽 🔜 関 🐘 🗍 🌌) 🗠 📥 🐚 🏫 🗍 🎬 🏭 📓 🌋	🖻 だ	ΣΣ]58005%	
×	Filters				
Applications Project IP Catalog	OLL BOBINTERFACE CP	orts C Addresses	Filters		
Software Projects	B B B Name	Rus Connection IR Type	IP Version		
Add Software Application Project		misrepleze	5.00 c		
Default: microblaze 0 bootloop		micropiaze	5.00.c		
Default: microblaze 0 xmdstub		lmb_v10	1.00.2		
Project: TestApp Memory		Imb_v10	1.00.a		
Project: TestApp Peripheral		oph mdm	2.00.2		
		Imb bram if onth	2.00.a		
		Imb_bram_if_cnth	2.00.a		
		opb uartlite	1.00.b		
	E-→LEDs 4Bit	opb gpio	3.01.b		
	DIPSWs 4Bit	opp dqo	3.01.b		
	PushButtons 5Bit	opb gpio	3.01.b		
		bram_block	1.00.a		
		dcm_module	1.00.a		
	System Assembly View1				
<u>ــــــــــــــــــــــــــــــــــــ</u>					-
۱ (۱					€
Output Warnings Errors	•				<u> </u>
🐲 👔 🛄 jagron@ shere-khan:~/388labs/basicMB] jagron@shere-khan:~/388labs/basicMB	jagron@shere-khan:~/388lab	s/basicMB/m	i 💎 Xilinx Platform Studio - /use	rs/jagron/388la

XPS Interface

- System Assembly View:
 - Graphical view of system.
 - Can edit configurations, port connections, bus connections, and memory spaces for all components.
- Tabs:
 - Project Tab.
 - Project info (.mhs, logs, etc.).
 - IP Catalog Tab.
 - Available peripherals that can be added to the system.
 - Application Tab.
 - Available SW projects that can be run on the system.

How To Run An Application

- Select the application of choice.
 - Compile the sources for the application.
 - Right-click and select "Build Application".
- Execute the test on the base system platform.
 - This requires the following to be combined...
 - Hardware bitstream (.bit)
 - Software executable (.elf)
 - This is done by selecting "Device Configuration".
 - "Update Bitstream" combines HW/SW (.bit + .elf).
 - "Download Bitstream" downloads the configuration to the board.

Monitoring Software Execution On The FPGA

- How do you see what is happening on the FPGA?
 - Normally in software you use print() statements.
 - The output goes to the screen.
- In this system, STDIN/STDOUT are routed to the serial port.
 - We must monitor the serial port from an external host to see what is happening.
- In order to "see" what is executing...
 - Open up a terminal windows.
 - Minicom (Linux) or Hyperterminal (Windows).
 - Setup the correct communication parameters
 - Baud rate = 9600.

Creating New SW Applications

- Select "Software"...
 - Click on "Add Software Application Project".
- Enter the new project name.
 - Also choose which CPU to run the application on.
- Now a new application tab entry will appear.
 - You can now add/edit sources for this application.
- In order to run this new application...
 - Right-click on it.
 - Select "Mark to initialize BRAMs".
 - Instructs the tool that this application is to be "combined" with the bitstream.
 - Now, when updating the bitstream, this application will be used.

Integration of IP Cores



- EDK allows one to add IP cores to a system.
 - Pre-built cores from Xilinx.
 - Custom cores.
- Additionally, there is a Create/Import Core Wizard that...
 - Allows one to quickly create bus interfaces for custom IP cores.
 - PLB, OPB, FSL interfaces.
 - Imports cores into an EDK repository.
 - So that the IP cores can be added into a system.

How to Create/Import Cores

- Select "Hardware"...
 - Click on "Create/Import Peripheral".
- Creating a new peripheral...
 - Peripheral name.
 - Bus interface type (PLB, OPB, FSL).
 - Interface features:
 - Master/Slave.
 - Interrupts and Resets.
 - Registers.
 - Accessible Signals.
- Importing a peripheral...
 - Not recommended, much easier to do via cut/copy from the command line.

Initial Assignment

- Create a new SW project...
 - Make it do the following...
 - Print "Hello <yourName>!"
- Run the SW on the board and demonstrate that it works correctly.