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).

Important EDK Files

• MHS File:

- Describes all components and connections in a system.
- MSS File:
 - Describes all SW drivers associated with components of a system.
- UCF File:
 - Describes the connections of all top-level ports.
 - All top-level ports have connections to specific physical pins on the FPGA.

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	==×
<u>File Edit View Project Hardware Software Device Configuration Debug Simulation Window Help</u>	
▋▋▓▋ऄ▐ॏऄॺॏॗॵऄऄ₭ऄऄऀॵॴॻॖख़ख़ॕॣॖॏय़ॣॣॾऄक़ॣऻॾॵॾॵढ़ॎॾॻॖऄ	
Applications Project Software Project Software Project Create new or open existing project Create new or open existing project Boss © Base System Builder wizard (recommended) Create new or open a recent project Create new or open a recent project Frowse for More Projects Browse installed EDK examples (projects) here QK Qancel	
✓ Output Warnings Errors	<u>*</u>
🔹 🛙 jagron@shere-khan:-//388labs/ 🔲 jagron@shere-khan:-//388labs/ 📄 jagron@shere-khan:-//388labs/ 🐟 Xilinx Platform Studio - no proje	

XPS - Creating The Base System

Create New XPS Project using BS	B Wizard 🛛 🗶
New project	
<u>P</u> roject file	
users/jagron/388labs/newProject/system.xmp	<u>B</u> rowse
Use Repository paths	
	Bro <u>w</u> se
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 ×
Embedded Development Ki Platform Studio	na-
Welcome to the Base System Builder!	
This tool will lead you through the steps necessary	to create an embedded system.
Please begin by selecting one of the following opt	ions:
C I would like to load an existing .bsb settings fil	e (saved from a previous session)
	Browse
More Info	ack Next > Qancel

XPS - Board Selection

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

Select a target develo	Base System Bu pment board:	ilder - Select Board	×
Select board		felles in development based	
Board vendor: Xil	aate a system for the	following gevelopment board	•
Board name: XU	JP Virtex-II Pro Devel	opment System	•
Board revision: C Note: Visit the vers	dor website for additio	nal board support materials.	
Vendor's Website		Contact Info	
C I would like to cre	ty Board Definition Fill eate a system for a cu	es ustom board	
Board description			
The XUP Virtex-II Pro Development System provides an advanced hardware platform that consists of a high performance Virtex-II Pro Platform FPGA surrounded by a comprehensive collection of peripherals that can be used to create a complex system and to demonstrate the capability of the Virtex-II Pro Platform FPGA.			
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 v xc2vp30 v ff896 v -7 v
Select the processor you would like to use in this design:
Processors
C Min Director Serial
MicroBlaze Unit The
PowerPC C C C C C C C C C C C C C C C C C C
Attar r
Ethernet 1740
SPLAN
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.
peripheral bus (OPB).
More Info < Back 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.

¥	Base System Builder - Configure MicroBlaze	3
MicroBlaze		
-Sustam wide setting	e	
Reference clock	Processor-Bus clock	
frequency:	frequency:	
100.00 MHz	100.00 MHz	
Beset polarity:	Active LOW 🚬	
Processor configura	tion	-
Debug I/F		
@ On-chip H/W d	ebug module	
C YMD with SAM	debug at the	
MD with S/W	debug stub	
C No debug		
	Data and instantion	
Micro	(Use BBAM)	
	(Gae Brivin)	
	16 KB	
Cache setup		.
C No Casha	C Enable cache liek	
14 No Cache	 Enable cache jink 	
Enable floating p	oint unit (EPU)	
		_
More Info	< Back Next > Cance	ł

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 of	n your board:
Xilinx XUP Virtex-II Pro Development System Revision C	
Please select the IO devices which you would like to use:	
-IO devices	
- IF RS232_Uart_1	a and
Peripheral: OPB UARTLITE	Data Sheet
Baudrate (bits per seconds): 9600	
Data bit <u>s</u> : 8 - Parity: NONE -	
T Use interrupt	
Ethernet_MAC	Data Short
	Lota Mate
	NOTe
More Info < Back	<u>N</u> ext > <u>C</u> ancel

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 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:	
-IO devices	
Data Sheet]
Peripheral: OPB GPIO	
Use interrupt	
Peripheral: OPB GPIO	
☐ Use interrupt	
PushButtons_5Bit Data Sheet	1
Peripheral: OPB GPIO	
☐ Use interrupt	
More Info < Back Next > Canc	el

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 In	iterfaces	×
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		- I
DDR_512MB_64Mx64_rank2_row13_col10_cl2_5	Data Sheet	
	Noțe	
DDR_512MB_64MX64_rank1_row13_col11_cl2_5	Data Sheet	
	Note	
DDR_256MB_32MX64_rank1_row13_col10_cl2_5	<u>D</u> ata Sheet	
	Note	
DDR_128MB_16MX64_rank1_row13_col9_cl2_5	Data Sheet	
	Noțe	
More Info < Back Next	> <u>C</u> ancel	

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.

▼ Base System Builder - Add Internal Periph	erals
Add other peripherals that do not interact with off-chip components "Add Peripheral" button to select from the list of available peripher	s. Use the rais.
If you do not wish to add any non-IO peripherals, click the "Next"	button.
	Add Peripheral
Peripherals	
More Info	rt > Cancel
More mo	

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.

\sim	Base System Builder - Software Setup	×	
	Devices to use as standard input and standard output		
	STDIN: RS232_Uart_1		
	STDOUT: RS232 Uart 1		
1	Sample application selection	1	
	Select the sample C application that you would like to have generated. Each application will include a linker script.		
	F 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.		
ľ			
-		-	
[-	More Info < Back Next > Cancel		

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.

⊻ B	ase System Builder - Configure Memory Test Application ×
The simple Me read/write test t	mory Test application will illustrate system aliveness and perform a basic to your memory devices.
MemoryTest - Select the me sections:	emory devices which will be used to hold the following program
Instruction:	ilmb_cntlr
Data:	dimb_cntir 💌
Stack/Heap:	dimb_cntir 💌
WARNING If you have p memory, you before you ca	laced the Instruction or Data section of this program in an external must use a debugger, bootloader, or ACE file to initialize memory an run this program!
More Info	< Back Next > Cancel

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"...

Below is a summary of the system you have created. Please review the information below. If it is correct, hit <Generate> to enter the information into the XPS data base and generate the system files. Otherwise return to the previous page to make corrections.

Base System Builder - System Created

Processor: Microblaze

System clock frequency: 100.000000 MHz Debug interface: On-Chip HW Debug Module On Chip Memory : 16 KB

The address maps below have been automatically assigned. You can modify them using the editing features of XPS.

	Instance Name	Base Addr	High Addr	
pb_mdm	debug_module	0x41400000	0x4140FFFF	
pb_uartlite	RS232_Uart_1	0x40600000	0x4060FFFF	
pb_gpio	LEDs_4Bit	0x40000000	0x4000FFFF	
pb_gpio	DIPSWs_4Bit	0x40020000	0x4002FFFF	
pb_gpio	PushButtons_5Bit	0x40040000	0x4004FFFF	
MB Bus : LMB_V	/10 Inst. name: Ilmb	Attached Comp	ponents:	
Core Name	Instance Name	Base Addr	High Addr	
mb bram if cntlr	ilmb cntlr	0x00000000	0x00003FFF	
Core Name	Instance Name	Base Addr	High Addr	
Core Name	Instance Name	Base Addr	High Addr	
mb bram if cntlr	dlmb_cntlr	0x00000000	0x00003FFF	

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

📤 Applications Actions 🍣 🖉 🍞 🙋				🔴 Tue Ma	y 29, 1:35 PM 🔇
✓ Xilin:	x Platform Studio - /users/jagron/3	888labs/newProject/system.xmp	- [System Asse	embly View1]	
IP File Edit View Project Hardware Software Device C	onfiguration <u>D</u> ebug Si <u>m</u> ulation <u>W</u> i	ndow <u>H</u> elp			_8:
I 🗅 🔊 🖥 🕹 I 🗗 🕷 📬 I 🗢 🗠 🖄 I 🗟 🛝 I 🗟	3 🗗 😼 🖂 🖌 🛛 😖 👘 🖉] 🕫 📥 🛐 🏫 🗍 🗰 🏭 🔟 🐹	* 🛛 🖻 🛍 🖉	x x 🛱 🗄 🗖 🗖 🙀	
×	Filters				
Applications Project IP Catalog	OLL 😤 🕫 Bus Interface 🤆 F	orts 🤆 Addresses 🔭 Connec	tion Filters		
Platform	B B B Name	Bus Connection IP Type	IP Version		
Project Files		microblaze	5.00.c		
-MHS File: system.mhs		opb v20	1.10.0		
MSS File: system.mss	l	lmb v10	1.00.a		
UCF File: data/system.ucf	_ ⊕ ≪dlmb	Imb v10	1.00.a		
-iMPACT Command File: etc/download.cmd	■ → debug module	opb mdm	2.00.a		
-Implementation Options File: etc/fast_runtim	☐ ☐ → →dimb cntir	Imb bram if c	ntlr 2.00.a		
Bitgen Options File: etc/bitgen.ut		Imb bram if c	ntir 2.00.a		
Project Options		opb_uartlite	1.00.b		
-Device: xc2vp30ff896-7	LEDs_4Bit	opb_gpio	3.01.b		
Netlist: TopLevel	DIPSWs_4Bit	opb_gpio	3.01.b		
-Implementation: XPS		opb_gpio	3.01.b		
-HDL: VHDL	⊡ • > Imb_bram	bram_block	1.00.a		
L-Sim Model: BEHAVIORAL		dcm_module	1.00.a		
⊡ Reference Files					
-Log Files					
·Synthesis Report Files					
	unders Annenskie Viewel				
	ystem Assembly View1				
×					
					-
					<u> </u>
Output Warnings Errors					
iagron@shere-khan:~/388labs/basicMP	on@shere-khan:~/388lahs/hasicMP	iagron@shere-khan:-/388	abs/basicMP/mi	A Xiliny Platform Studio - /users/jagron/3881	
	one one entere knun. yooolabby basiemb	Jugrone shere knall. 7500	abs, busicing/illi		

XPS - IP Catalog Tab

📤 Applications 🛛 Actions 🔍 🗏 🍙 💋 🥑						🕕 Tue May 29, 1:35 PM 🔌
Xilinx Platfor	n Studio - /users/jagron/	388labs/newProje	ct/system.xmp -	[System Ass	embly View1]	*
Description File Edit View Project Hardware Software Device Configuration	on <u>D</u> ebug Si <u>m</u> ulation <u>W</u>	/indow <u>H</u> elp				<u>_18 ></u>
	🔽 🛛 🛛 🗛 🔝 🎨 🗍 🖉	🗾 uha 📩 🕅 🌧	琵虹 🏭 💢 💥	K 🛛 🖻 🖭 🛛	ΣΣ [6 B D 🗖 🕅	
	- Filters			- 1 1 -		
Applications Project IP Catalog OLL	C Bus Interface C	Ports C Address		n Filtore		
€ PMM	e Buo mondoo				1	
Θ	Name	Bus Connection	IP Type	IP Version		
Name 🗸 Version 🔺			microblaze	5.00.c		
- Analog ←	there are the second s		opb_v20	1.10.c		
⊕-Bus —	_ ± • > ilmb		lmb_v10	1.00.a		
🗄 Bus Bridge 🔶 🔶	_ ± ≪dimb		lmb_v10	1.00.a		
🗄 Clock Control	debug_module		opb_mdm	2.00.a		
Communication High-Speed	dimb_cntir		Imb_bram_if_cnt	lr 2.00.a		
Communication Low-Speed	utimb_cntir		Imb_bram_if_cnt	lr 2.00.a		
DMA	HS232_Uart_1		opp_uartiite	1.00.b		
E Debug	EDS_4Bit		opb_gpio	3.01.b		
FPGA Reconfiguration	Durk Dutterra		opp_gpio	3.01.D		
General Purpose IO	PushButtons_5B	L	opp_gpio	3.01.D		
Interrupt Control			dem medule	1.00.a		
⊕-Memory Block	dom_0		dcm_module	1.00.a		
Memory Controller						
⊕-PCI						
Peripheral Controller						
t Processor						
Ereset Control						
timer ↓ Imer						
System As	sembly View1					
×						
						<u></u>
						-
						<u> </u>
Output Warnings Errors						
iagron@shere-khan:~/388labs/basicMB iagron@sher	e-khan:~/388labs/basicMP	iagron@sh	ere-khan:~/388lah	s/basicMR/m	i 🕭 Xilinx Platform Studio - /use	rs/jagron/388ja
	c man y soonass, sasieme		ere anan. 7500lat	o, ousiend/m	· · · · · · · · · · · · · · · · · · ·	

XPS - Applications Tab

							🕕 Tue May 29, 1:36 PM 🔌
	Xilinx Platform	Studio - /users/jagron/	388labs/newProje	ct/system.xmp -	[System Asse	embly View1]	
<u>File</u> <u>Edit</u> <u>View</u> <u>Project</u> <u>Hardware</u> <u>Software</u> D	evice <u>C</u> onfiguratio	n <u>D</u> ebug Si <u>m</u> ulation <u>W</u>	(indow <u>H</u> elp				
] 🗋 🤌 🔚 🕹] 🔯 📸 🔂] 🏍 📿 🖁 🛍	M 🛛 🖻 🗗 🔂 I	57 🛛 🛛 🕸 🔛 🗞 🗍 🕫	Z 🗠 📩 🛓 🛛	[태태 2음]] 🐹 🔉	ƙ 🛛 🖻 🕅 🗍	ΣΣ]5800 🗗 😽	
×		Filters					
Applications Project IP Catalog	O L L	🐣 🖲 Bus Interface 🏻	Ports C Addresse	s Sr Connectio	n Filters		
Software Projects	P M M B B B	Nama	Bue Connection	LID Turne			
Add Software Application Project	***		Bus Connection	ir iype	TF Version		
Default: microblaze 0 bootloop	▌┝┼┼─	microblaze_0		microbiaze	5.00.C		
Default: microblaze 0 xmdstub	┃ ←	dqo_dm		opp_v20	1.10.0		
Project: TestApp Memory				Imb_v10	1.00.a		
Project: TestApp Peripheral	•			oph_mdm	2.00 a		
		the debug_module		Imb bram if ont	2.00.a		
				Imb_bram_if_cnt	Ir 2.00.a		
				oph uartite	1.00 h		
				opb_ganio	3.01.b		
		E ⊃DIPSWs 4Bit		opb gpio	3.01.b		
			ł	opb gpio	3.01.b		
				bram block	1.00.a		
	1	±		dcm module	1.00.a		
	1	-		-			
	1						
	1						
	1						
	1						
	1						
	1	1					
	🕩 System Ass	embly View1					
<u>×</u>							
Ī							
							-
	······						
4	_						<u> </u>
U Uutput Warnings Errors							<u></u>
Output Warnings Errors							

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.
 - Over the USB-based JTAG connection.

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.
- This may not seem like much, but...
 - You just created a SoC (System-On-Chip).
 - You just cross-compiled a program to run on "bare-metal" (no OS).

Questions

- 1) What is an FPGA?
- 2) What is an SoC? Why is it different from your desktop computer system?
- 3) What does soft-core IP mean?
- 4) What is an MHS, MSS, and UCF file?
- 5) What does cross-compile mean?
- 6) Why does it take so long to build the HW portion of your system?
- 7) How does the desktop computer program the FPGA, how does it monitor the FPGA?