UART Interrupt Creation on Spartan 3A

This tutorial will demonstrate the UART Interrupt based application. To show this we will build a simple Interrupt application that will use the hyper-terminal to create an interrupt.

The second half of the tutorial will show how to convert from the standalone OS to the Xilkernel OS.

This application will use the Spartan 3A board. The design will be created using EDK 10.1.03i.

Step 1: Create the Interrupt Project.

To create the project, the Base System Builder (BSB) in XPS will be used. To launch XPS, go to Start -> Run and type "XPS", this will launch the XPS tool.

Follow the steps seen below to build the BSB design:

Select Base System Builder, Press Next to continue.

📀 Xilinx	Platform Studio			×
Create	e new or open existing project			
BSE	Base System Builder wizard (recommended)			
ľ	C Blank XPS project			
R	Open a recent project			
Brov	vse for More Projects			•
Browse	installed EDK examples (projects) <u>here</u>	ОК	Cancel	Help

Browse to where you want to save your interrupt project, Press Next to continue.

📚 Create New XPS Project using BSB Wizard	×
New project Project file	
C:/Interrupt_Tutorial/system.xmp	Browse
Advanced options (optional: F1 for help)	
	Browse
OK	Cancel

Select "I would like to create a new design", Click Next to continue.

Base System Builder - Welcome			×
Embedded Developm Platform Studio	ent Kit		
Welcome to the Base System	n Builder!		
This tool will lead you through the steps necessar	y to create an emi	bedded system.	
Please begin by selecting one of the following o	options:		
I would like to create a new design			
C I would like to load an existing .bsb settings	file (saved from a	previous session)	
			Browse
More Info	< Back	Next >	Cancel

Select the Spartan 3A Board as shown below, Select Next to Continue

📀 Base System	n Builder - Select Board			? ×
Select a target d	evelopment board:			
-Select board-				
I would like	e to create a system for the follo	owing developmer	nt board	
Board vendor:	Xilinx			•
Board name:	Spartan-3AN Starter Kit			
Board revision	: D			-
Note: Visit the	vendor website for additional l	poard support mat	erials.	
Vendor's Web	<u>site</u>	Contact Info		
Download Thi	rd Party Board Definition Files			
🔿 I would like	e to create a system for a custo	m board		
Board descripti	ion			
XC3S700AN four push but button rotary Ethernet port button South	FGG484 device. The board in tons, eight LEDs, VGA port, 16 encoder, SPI analog to digital , 2-16 Mbit SPI flash, 4 MB of p (RESET) is used as system res	cludes two RS232 5 character 2 line L converter, SPI dig parallel flash and 5 et.	2 serial ports, four DIF _CD display, PS/2 po ital to analog conver i12 MB DDR2 SDR4	-owin P switches, ort, push ter, 10/100 M. Push
More Info		< Back	Next >	Cancel

Select the Microprocessor, Press Next to Continue.

📀 Base System Builder - Select Processo	r		? ×
The board you selected has the following FP0 Architecture: Device: spartan3a xc3s700an Use stepping	GA device: Package: fgg484	Speed gra	de:
Select the processor you would like to use in th Processors MicroBlaze PowerPC Not supported by this device	is design:		
Processor description The MicroBlaze(TM) 32-bit soft processor is LUT RAM-based Register File, with separate supports both on-chip BlockRAM and/or ex the FPGA fabric.	a RISC-based engin e instructions for dat ternal memory. All p	e with a 32 register a and memory acce eripherals are impler	by 32 bit ss. It nented on
More Info	< Back	Nevts	Cancel

Select 8KB memory for the Local Memory, Press Next to continue.

🗇 Base System Builder - Configure Microl	Blaze Processs	or	<u>? ×</u>
Reference clock frequency: Processor-B	us clock frequend	sy:	
50.00 MHz 62.50	▼ MHz		
Reset polarity: Active High 💌			
Processor configuration			
Debug I/F On-chip H/W debug module XMD with S/W debug stub No debug MicroBlaze Cache setup Enable Enable Enable Enable floating point unit (FPU)	cal memory ta and Instruction se BRAM) KB		
More Info	< Back	Next >	Cancel

Select the peripherals as shown below, Press Next to Continue.

he following external memory and IO devices were found on your board:	
ilinx Spartan-3AN Starter Kit Revision D	
lease select the IO devices which you would like to use:	
ID devices	
RS232_DTE	Data Sheet
Peripheral: XPS UARTLITE	
Baudrate (bits	
per seconds): 9600 💌	
Data bits: 8	
Parity: NONE 💌	
Use interrupt	
	Data Sheet
	Data Sheet
More Info Kext :	> Cancel

Select the peripherals as shown below, Press Next to Continue.

📀 Base System Builder - Configure IO Int	erfaces (2 of 3)		? ×
The following external memory and IO devices v	vere found on your	board:	
Xilinx Spartan-3AN Starter Kit Revision D			
Please select the IO devices which you would li	ike to use:		
-10 devices			
DIPs_4Bit		Data	Sheet
BTNs_4Bit		Data	Sheet
Ethernet_MAC		Data	: Sheet lote
		Data	Sheet
More Info	< Back	Next >	Cancel

Select the peripherals as shown below, Press Next to Continue.

🗇 Base System Builder - Configure IO Interfaces (3 of 3)	<u>? ×</u>
The following external memory and IO devices were found on your board:	
Xilinx Spartan-3AN Starter Kit Revision D	
Please select the IO devices which you would like to use:	
10 devices	
SPI_FLASH	Data Sheet
FLASH	Data Sheet
Peripheral: MPMC	Data Sheet
More Info ABack Next	> Cancel

Select Next to continue

🗇 Base System Builder - Add Internal Peripherals (1 of 1)	×
Add other peripherals that do not interact with off-chip components. Use th	e
"Add Peripheral" button to select from the list of available peripherals.	
In you do not wish to dad dry horno perpricials, click the fresh bacon.	Add Peripheral
Perinherals	Had Foliphora
T onpriorais	
More Info < Back Nex	t> Cancel

In software setup, de-select the button for the two Application, shown below. Press Next -> Generate -> Finish to finish

	builder - Software Setu	ф —	
Devices to use	as standard input, standard	output, and boot memory	
STDIN:	RS232_Uart	•	
STDOUT:	RS232_Uart	•	
Boot Memory:	ilmb_cntlr	•	
Sample applical Select the sam	ion selection ble C application that you we	ould like to have generated. E	Each application will
include a linker	script.		
Memory tes	t		
Illustrate sy:	stem aliveness and perform	a basic read/write test to eacl	h memory in your system
Peripheral s	elftest		
1 0/10/11/1 0 0/	inple cell test for edoir perip	nordi in your oyotom.	
Below are othe	r software applications found	d for your board. In order to se rements - See "More Details"	elect an application,
Below are othe please ensure y ML403 Cyp	r software applications found our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	elect an application,
Below are othe please ensure y ML403 Cyp	r software applications found our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	elect an application, More Details
Below are othe please ensure y	r software applications found our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	elect an application, More Details
Below are othe please ensure y ML403 Cyp	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	elect an application,
Below are othe please ensure y ML403 Cyp	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	ect an application,
Below are othe please ensure y ML403 Cyp	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	ect an application,
Below are othe please ensure y ML403 Cyp	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	ect an application,
Below are othe please ensure y	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	ect an application,
Below are othe please ensure y	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	elect an application,
Below are othe please ensure y	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	ect an application,
Below are othe please ensure y	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	ect an application,
Below are othe please ensure y	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	Nore Details
Below are othe please ensure y	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	Hect an application,
Below are othe please ensure y ML403 Cyp	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	Nore Details
Below are othe please ensure y ML403 Cyp	r software applications foun our system satisfy the requir ress USB Application	d for your board. In order to se rements. See "More Details".	Nore Details

The next thing to do is to build the software application. This software application will be a simple application to show how to register the interrupt and to register the handler. To create a new application, Click on the Application tab on the left and right click on "Add Software Application Project" seen below:

Project	Applications	IP Catalog	
Software	Projects		
	Add Softw	are Application) Project
🌇 De	efault: microblaze	_0_xmdstub	

In the pop-up box, name your Application "Interrupt_Test" as shown below. Select OK to continue:

📀 Add Software Applical	ion Project		×			
Project Name Interrupt_Te	st					
Note: Project Name cannot	have spaces					
Processor microblaze_0						
🕞 🗖 Project is an ELF-only	Project is an ELF-only Project					
Choose an ELF file.						
	_	Browse				
The ELF file is assumed to	be generate	ed outside XPS				
Default ELF name is <sw< td=""><th>project name</th><th>>/executable.elf</th><td></td></sw<>	project name	>/executable.elf				
	ОК	Cancel				

This will create an empty project for you. To create the source code for the Interrupt_Test application, right click on "source" in the application and select "Add New File..." This is seen below:

Project	Applicat	ions	IP Catalog				
Software	Projects						
🔁 Ad							
🌄 Default: microblaze_0_bootloop							
🌄 D e	efault: micr	oblaze	_0_xmdstub				
🗄 - 🎇 Pr	oject: In	terrup	t_Test				
i‡⊷ Pi	rocessor: r	nicrobla	aze_O				
E:	kecutable:	C:\ca:	ses\Test_Inter	rupl	t\Interrupt_Test\exec		
⊕- <u>Compiler Options</u>							
<mark>S</mark>							
	Add New File						

Browse to the Interrupt_Test folder, if it isn't there create one making sure you name it correctly. Name the source file "Test.c", making sure you use a lower case c for the file. This is seen below:

Source/Header F	ile to create and	add to Project			<u>? ×</u>
Save in:	Conterrupt_Tes	t	•	(+ 🗈 💣 🎫	-
My Recent Documents					
Desktop					
My Documents					
My Computer					
i		-			
My Network Places	File name: Save as type:	Test.c	.c;*.c++;*.cpp;*.cc	▼ ;*.cxx) ▼	Cancel

The next step is to copy the code seen below into the empty Test.c file. This code is shown below:

```
#include <xintc_l.h>
#include <xparameters.h>
#include <xuartlite l.h>
#define Interrupt O
#define Mask OX000001
   //debug interrupt service routine
   void debug_int_handler(void *baseaddr_p)
   {
     char c;
     while (!XUartLite mIsReceiveEmpty(XPAR UARTLITE O BASEADDR))
      {
         c = XUartLite RecvByte(XPAR UARTLITE O BASEADDR);
        xil printf("You Entered %c\r\n",c);
      }
   }
   int main()
   {
      microblaze enable interrupts();
      XIntc_RegisterHandler(XPAR_INTC_SINGLE_BASEADDR,Interrupt,
      (XInterruptHandler) debug_int_handler, (void *) XPAR_UARTLITE_O_BASEADDR);
      XIntc_mMasterEnable(XPAR_XPS_INTC_O_BASEADDR);
      XIntc mEnableIntr(XPAR XPS INTC O BASEADDR, Mask);
      XUartLite mEnableIntr(XPAR UARTLITE O BASEADDR);
      xil printf("Interrupt Setup Finished....\r\n");
      //Wait for interrupts to occur
      while (1)
      {
      }
     return 0;
   }
```

Now that the code has been written, the project can be intialized into the BRAMS. To do this right click on the Interrupt_Test application and select "*Mark to initialize brams*..." This is seen below:

Project	Appl	ications	IP Catalog				
Software	Projec	ts					
🔁 Ad	ld Soft	ware Appli	cation Project	t			
🎇 De	efault: i	microblaze	_0_bootloop				
🔛 De	fault: i	microblaze	_0_xmdstub				
🖻 🞇 Pr	ojest						
i Pr	oc	Set Compiler Options					
E1	(et	Mark to Initialize BRAMs iterrupt_Test\executa					
t⊒∼U(t∓⊡S(om Sul	Build Project					
Т H	ea	Clean Project					
		Delete Project					
		Make Project Inactive					
		Generat	e Linker Scrip	t			

Next Generate the Linker script for the Interrupt_Test application. To do this right click on the Interrupt_Test application and select "*Generate Linker Script*" Place all sections into the DDR2 SDRAM, this is seen below:

🖗 Generate Linker	Script						2
Sections View:				Heap and Stack View	W:		
Section	Size (bytes)	Memory		Section	Size (bytes)	Memory	
.text	0x00000F84	DDR2_SDRAM_		Неар	0x400	DDR2_SDRAM_	
.rodata	0x00000422	DDR2_SDRAM_		Stack	0x400	DDR2_SDRAM_	1
.sdata2	0x00000002	DDR2_SDRAM_					
.sbss2	0x00000000	DDR2_SDRAM_					
.data	0x0000012C	DDR2_SDRAM_		I			
.sdata	0x00000004	DDR2_SDRAM_					
.sbss	0x00000000	DDR2_SDRAM_		Memories View:			
.bss	0x00000024	DDR2_SDRAM_		Memory	Start Address	Length	
,			1	ilmb_cntlr_dlmb_cntl	0x00000000	8K	
		Add Section	Delete Section	DDR2_SDRAM_C_I	0x8C000000	65536K	
Boot and Vector Sect	tions:						
Section	Address	Memory		1			
.vectors.reset	0x00000000	ilmb_cntlr_dlmb_cntl		ELE file used to popu	late section informatio		
.vectors.sw_excepti	0x00000008	ilmb_cntlr_dlmb_cntl				л. 	
.vectors.interrupt	0x00000010	ilmb_cntlr_dlmb_cntl		[U:\cases\759219_p	hillip\Spartan3_uart_i	nterrupt\interrupt_test	Nexecutable.elf
.vectors.hw_excepti	0x00000020	ilmb_cntlr_dlmb_cntl					
				Output Linker Script:	h3_uart_interrupt\int	errupt_test\interrupt_t	est_linker_script.ld
						OK Can	cel Help

UART Interrupt Tutorial 23/10/2008

- The next step is to compile the Libraries and BSP's, to do this select Software -> Generate Libraries and BSP's.
- Then build the Interrupt_Test application, to do this go to Software -> Build all user applications...
- Now build the bitstream, to do this go to Hardware -> generate bitstream.
- Now update the bitstream with the software, to do this go to Device Configuration -> Update bitstream.
- Finally, download the bitstream to the board, to do this go to Device Configuration -> Download bitstream.
- Open the Hyper-Terminal if you haven't done so and set the baud rate to 9600
- Open the XMD, type the follow commands:

dow interrupt_test/executable.elf
run

• When you type something on the keyboard you should see it on the hyperterminal, as seen below:



Updating design from Standalone OS to Xilikernel OS

Firstly, download the Mircoblaze application see on the page above. For this example the Spartan 3AN will be used:

http://xirweb/~stephenm/Projects/Spartan_3AN_Uart_interrupt.zip

Step 1: Add timer, connect to plb and give it an address space.

A timer will need to be added, as this is needed by the Xilkernel OS when using Microblaze. To add the timer, go to the IP Catalog and drop the DMA and Timer list. Double click the XPS timer/Counter, seen below:

Project Applications (IP Catalog						
<u>i</u> -•	•					
Description	IP Version	IP Type				
🖕 🗶 EDK Install C:\Xilinx\10.1\ED						
🕂 Analog						
🕂 - Bus and Bridge						
Clock, Reset and Interrupt						
Communication High-Speed						
Communication Low-Speed						
E-DMA and Limer						
€ ★ XPS Timer/Counter	1.00.a	xps_timer				
	1.00.Ь	xps_timebase_wdt				
🔔 XPS Watchdog Timer	1.00.a	xps_timebase_wdt				
	2.00.Ь	xps_central_dma				
🤼 XPS Central DMA Cont	2.00.a	xps_central_dma				
🔔 XPS Central DMA Cont	1.00.a	xps_central_dma				
💛 OPB Timer/Counter	1.00.Ь	opb_timer				
😑 OPB Watchdog Timer	1.00.a	opb_timebase_wdt				
😑 OPB Central DMA Cont	1.00.c	opb_central_dma				
🖙 🛧 Fixed Interval Timer	1.01.a	fit_timer				

UART Interrupt Tutorial 23/10/2008

To connect the xps_timer_0 to the PLB, In system Assembly view select the Bus interface tab, drop down the xps_timer_0 and select mb_plb. Shown below:

🗧 Bus Interfaces 🛛 F	Ports Addresses		
Name	Bus Connection	IP Type	IP Version
🗄 🥯 microblaze_0		microblaze	7.10.d
🗢 dimb		lmb_v10	1.00.a
🧼 ilmb		lmb_v10	1.00.a
🕂 🗢 mb_plb		plb_v46	1.03.a
😟 🥯 dimb_cnth		lmb_bram_if_cntlr	2.10.a
🕂 🗢 ilmb_cnth		lmb_bram_if_cntlr	2.10.a
🗄 🗢 DDR2_SDRAM		mpmc	4.03.a
🗄 🗢 lmb_bram		bram_block	1.00.a
🗄 🥯 debu <u>a_</u> module		mdm	1.00.d
🗄 🗢 xps_intc_0		xps_intc	1.00.a
🚊 🥯 xps_timer_0		xps_timer	1.00.a
SPLB	mb plb 🗾 💌		
₫- <i>≪R\$232_DTE</i>	No Connection	xps_uartlite	1.00.a
	New Connection	clock_generator	2.01.a
i…	mb_plb	proc_sys_reset	2.00.a

To give the xps_timer_0 an address space, In system Assembly view select the Addresses tab, drop down the xps_timer_0 and select 2K. Then click Generate Addresses. Shown below:

Bus Interfaces	Borts Addresses					🚟 Generate Addresses
Instance	Name 🛆	Base Address	High Address	Size	Bus Interface(s)	Bus Connection L
dlmb_ontlr	C_BASEADDR	0x00000000	0x00001fff	8K 🗾	SLMB	dlmb 🕻
ilmb_cntlr	C_BASEADDR	0x00000000	0x00001fff	8K 🗾	SLMB	ilmb 🕻
debug_module	C_BASEADDR	0x84400000	0x8440ffff	64K 💌	SPLB	mb_plb 🖸
mb_plb	C_BASEADDR			U 🔽	Not Applicable	C
xps_intc_0	C_BASEADDR	0x81800000	0x8180ffff	64K 💌	SPLB	mb_plb 🖸
xps_timer_0	C_BASEADDR	0x88208000	0x882087ff	2K 💌	SPLB	mb_plb 🕻
RS232_DTE	C_BASEADDR	0x84000000	0x8400ffff	512 🔺	SPLB	mb_plb 🕻
DDR2_SDRAM	C_MPMC_BASEADDR	0x8c000000	0x8fffffff	1К	SPLBO	mb_plb 🖸
				2K		
				4K		
				18K		
				165		
				52N 64K		
				128K		
				256K 🗾		

Step 2: Connect Timer to interrupt controller.

In the System Assembly view, click the Ports tab then drop down the timer ports. Make a new connection for the timer interrupt port, seen below:

🔁 Bus Interfaces Ports Address	ses	
Name	Net	Dire
i ⊕- →External Ports		
⊕ · → microblaze_0		
🗄 🗢 alimb		
🕂 🗢 ilmb		
🗄 🗢 mb_plb		
🗄 🗢 dimb_cntir		
🗄 🗢 ilmb_cnth		
t∲- ∽DDR2_SDRAM		
🗄 🗢 Imb_bram		
🕂 🗢 debu <u>a_</u> module		
ite_ ∽ xps_intc_0		
i≑· → xps_timer_0		
Freeze	No Connection 📃	Ι
Interrupt	No Connection 📃 💌	0
PWM0	No Connection	О
GenerateOut1	New Connection	0
GenerateOut0	Make External	0
CaptureTrig1	No Connection 📃 💌	1
¹ CaptureTrig0	No Connection 📃	I

Then select the interrupt controller, Select the intr port. This should open the GUI. Select the xps_timer_0_interrupt signal and click on the green cross. Seen below:



Step 3: Change the OS from Standalone to Xilkernel

Go to Software -> Software Platform Settings. Change the OS to Xilkernel, seen below;

📀 Software Platforn	n Settin	gs									×
Processor Information	n										
Processor Instance:	microbl	laze_0 💌									
Software Platform		Processor Settings -									
OS and Libraries	CPU Dr	iver: cpu	- CF	PU Driver	Version: 1	.11.b 💌					
Drivers	Process	sor Parameters:									
	Name		Current Val	ue	Default	Value	Туре	Description			
	Ė- mie	croblaze_0	-		-		navinkaval iv	utana Dahua ania	leaved to be seend with a	una ala tu da	
		extra_compiler_flags	rione :-g		-g		string	Extra compil	er flags used in BSP a	nd library generati	ion.
		archiver	mb-ar		mb-ar		string	Archiver use	ed to archive libraries fo	or both BSP gene	rati
		··· compiler	mb-gcc		mb-gcc		string	Compiler use	ed to compile both BSF	'/Libraries and Ap	plic
	•										•
		00.4130									
	_	US & Library Settings				Con 1					_
	OS: 🗐	ilkernel 🗾	Version:	4.00.a	•	scheduling	itware platform. I synchronization	time etc. Requires	kernel services such a an XPS timer on a Mici	oBlaze platform.	-
	Use	Library	∙ √ersion		Description	,					=
		xilmfs 1	1.00.a	- ×	ilinx Memory	File System					
		xilisf	1.00.a	- ×	ilinx In-syster	n and Serial I	Flash Library				
		xilflash	1.02.a	- ×	ilinx Flash lib	rary for Intel/	AMD CFI complia	ant parallel flash			
		xilfatfs	1.00.a		rovides read	/write routine	is to access files	stored on a FAT16/			
		Wip .	3.00.a			tack library v	/3.UU.a IulDut 20 Vilia	u adapter ut 00 a			
		Improo	1.00.a		VIE I CEVIE 3	CACK IIDIAIY.	IWIF ¥1.3.0, AIIII	x auapter v1.00.a			
	1										
								OK	Cancel	Help	

UART Interrupt Tutorial 23/10/2008

Step 4: Set-up the Xilkernel system

Go to Software -> Software Platform Settings. In OS an Libraries

For sysintc_spec, select xps_intc_0. For stdout and stdin, select RS232_DTE. Seen below:

⊡- xilkernel		
	xps into 0	•
stdout	RS232 DTE	•
stdin	RS232 DTE	V
•		

For sched_type, select SCHED_PRIO

🖯 config_sched	true 🗾
max_readyq	10
n_prio	32
sched_type	SCHED PRIO 🗾

For config_pthread_support, double click on Edit (seen below). In the GUI, select Add And enter the thread name "CreateThreads" with a priority of 1 (seen below)

⊖- config_pthread_support	true	•
static_pthread_table	Edit	
max_pthread_mutex_wa	iitq <i>10</i>	
max_pthread_mutex	10	
config_pthread_mutex	false	•
pthread_stack_size	1000	
max_pthreads	10	

Add/Delete List of Parameter-Values	×			
Parameter Name: static_pthread_table				
Parameter Description: Static specification of pthreads. These threads will be created at Xilkernel startup				
pthread start func othread prio	-			
CreateThreads 1				
To add an element to the parameter list, click "Add"				
To delete an element, select the fow and click. Delete				
Add Delete OK Cancel	1			

For the systmr_spec, in the systmr_dev, select the xps_timer_0

Ė~ systmr_spec		
systmr_interval	10	
systmr_freq	100000000	
systmr_dev	xps timer 0	-

The MSS file should be same as below:

```
BEGIN OS

PARAMETER OS_NAME = xilkernel

PARAMETER OS_VER = 4.00.a

PARAMETER PROC_INSTANCE = microblaze_0

PARAMETER sysintc_spec = xps_intc_0

PARAMETER stdout = RS232_DTE

PARAMETER stdin = RS232_DTE

PARAMETER stdin = RS232_DTE

PARAMETER sched_type = SCHED_PRIO

PARAMETER systmr_dev = xps_timer_0

PARAMETER static_pthread_table = ((CreateThreads,1))

END
```

Step 5: Run LibGen

To run LibGen, go to Software -> Generate Libraries and BSP's

Step 6: Build the Xilkernel Application

#include "xmk.h"
#include <os_config.h>
#include <sys/process.h>
#include <pthread.h>
#include <sys/intr.h>
#include <xparameters.h>
#include <xuartlite_l.h>
#include <xstatus.h>

//Variables	
static pthread_t	tid;
<pre>static pthread_attr_t</pre>	attr;
static struct sched_param	spar;

//Funtion Prototype

void* uart_thread();

```
/* uartlite interrupt service routine */
void uart_int_handler(void *baseaddr_p) {
    char c;
    /* till uart FIFOs are empty */
    while (!XUartLite_mIsReceiveEmpty(XPAR_RS232_DTE_BASEADDR)) {
        /* read a character */
        c = XUartLite_RecvByte(XPAR_RS232_DTE_BASEADDR);
        /* print character on hyperterminal (STDOUT) */
        xil_printf ("Character: %c \r\n", c);
    }
}
void main()
```

```
xilkernel_main();
```

```
}
```

{

```
void* CreateThreads(void* dummy)
{
      pthread_attr_init (&attr);
      spar.sched_priority = 1;
      pthread_attr_setschedparam(&attr,&spar);
      pthread_create (&tid, &attr, (void*)uart_thread, NULL);
      return 0;
}
void* uart_thread()
ł
      int_id_t UartIntrId = XPAR_INTC_0_UARTLITE_0_VEC_ID;
      register_int_handler(UartIntrId,uart_int_handler,NULL);
      enable_interrupt(UartIntrId);
      XUartLite_mEnableIntr(XPAR_RS232_DTE_BASEADDR);
      /* Wait for interrupts to occur */
      while (1);
```

```
}
```

UART Interrupt Tutorial 23/10/2008

Step 7: Set compiler options to compile Xilkernel library files.

Right click on the Interrupt_Test application then select Set Compiler Options, seen below:



Under the Paths and Options tab, type Xilkernel into the libraires to link against field, shown below:

📀 Compiler Options	×
Compiler Tools: mb-gcc	
Environment Debug and Optimization Paths and Options)
All paths should be relative to project directory. Separate multiple of space.	options with a
-Search Paths	
Library (-L)	Browse
Include (-I)	Browse
Libraries to Link against (-I)	
xilkernel	
Other Compiler Options to Append These options will be appended to the compiler command line	
OK Cancel	Help

Step 8: Generate the linker script, and download application.

Place all section in the DDR2_SDRAM. Also, increase the heap to 0x400. Shown below:

Sections View:			Heap and Stack Vie	ew:		
Section	Size (bytes)	Memory	Section	Size (bytes)	Memory	
text	0x00003C6C	DDR2_SDRAM_	Heap	0x400	DDR2_SDRAM_	
rodata	0x00000582	DDR2_SDRAM_	Stack	0x400	DDR2_SDRAM_	1
sbss2	0x00000000	DDR2_SDRAM_				-
data	0x00000140	DDR2_SDRAM_				
sbss	0x00000000	DDR2_SDRAM_				
bss	0x0000372C	DDR2_SDRAM_				
			Memories View:			
			Memory	Start Address	Length	
	Add See	ction Delete Section	ilmb entit dimb er	ป กรุกกกกกกกก	8K	
			Tump_cria_amp_cr	000000000	OIX .	
			DDR2_SDRAM_C	_I 0x8C000000	65536K	
oot and Vecto	or Sections:		DDR2_SDRAM_C	_ 0x8C000000	65536K	
oot and Vecto	or Sections: Address	Memory	DDR2_SDRAM_C	_ 0x8C000000	65536K	
oot and Vecto Section vectors.reset	or Sections: Address 0x00000000	Memory ilmb_cntlr_dlmb_cntl	DDR2_SDRAM_C	0x8C000000	65536K	
oot and Vecto Section vectors.reset vectors.sw_ex	or Sections: Address 0x00000000 xceptii 0x00000008	Memory ilmb_cntlr_dlmb_cntl ilmb_cnttr_dlmb_cntt	DDR2_SDRAM_C		65536K	
oot and Vecto Section vectors.reset vectors.sw_ex vectors.interru	or Sections: Address 0x00000000 xceptii 0x0000008 apt 0x00000010	Memory ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl	ELF file used to pop	UNACCOUNCE	65536K	
oot and Vecto Section vectors.reset vectors.sw_ex vectors.interru vectors.hw_ex	or Sections: Address 0x00000000 xcepti 0x0000008 upt 0x00000010 xcepti 0x00000020	Memory ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl	ELF file used to pop	ulate section inform. terrupt_mb_101%ilk	65536K ation: ternel_Spartan_3AN\Inte	errupt_Test\executable.
oot and Vecto Section vectors.reset vectors.sw_ex vectors.interru vectors.hw_ex	or Sections: Address 0x00000000 xcepti 0x0000008 apt 0x00000010 xcepti 0x00000020	Memory ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl	ELF file used to pop	ulate section inform. terrupt_mb_101\Xilk	65536K etion: ernel_Spartan_3ANVInte	errupt_Test\executable.
oot and Vecto Section vectors.reset vectors.sw_ex vectors.interru vectors.hw_e:	or Sections: Address 0x00000000 xcepti 0x0000008 apt 0x00000010 xcepti 0x00000020	Memory ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl	ELF file used to pop	ulate section inform. terrupt_mb_101\Xilk	65536K etion: terruel_Spartan_3AN\Inte	errupt_Test\executable. est_linker_script.ld
oot and Vecto Section vectors.reset vectors.sw_ex vectors.interru vectors.hw_ex	or Sections: Address 0x00000000 xcepti 0x0000008 upt 0x00000010 xcepti 0x00000020	Memory ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl ilmb_cntlr_dlmb_cntl	ELF file used to pop	UNACCOURCE 0x8C000000 ulate section inform. terrupt_mb_101\Xilk : [.Spartan_3AN\In	65536K etion: terruel_Spartan_3AN\Inte terrupt_Test\Interrupt_T	errupt_Test\executable. est_linker_script.ld
toot and Vector Section vectors.reset vectors.sw_ex vectors.interru vectors.hw_ex	or Sections: Address 0x00000000 xcepti 0x0000008 upt 0x00000010 xcepti 0x00000020	Memory imb_cntir_dimb_cnti imb_cntir_dimb_cnti imb_cntir_dimb_cnti imb_cntir_dimb_cnti	ELF file used to pop	ulate section inform terrupt_mb_101\Xilk	65536K 65536K ation: terrupt_Test\Interrupt_T	errupt_Test\executable. est_linker_script.ld

- Then build the Interrupt_Test application, to do this go to Software -> Build all user applications...
- Now build the bitstream, to do this go to Hardware -> generate bitstream.
- Now update the bitstream with the software, to do this go to Device Configuration -> Update bitstream.
- Finally, download the bitstream to the board, to do this go to Device Configuration -> Download bitstream.
- Open the Hyper-Terminal if you haven't done so and set the baud rate to 9600
- Open the XMD, type the follow commands:

dow interrupt_test/executable.elf run

UART Interrupt Tutorial 23/10/2008

• When you type something on the keyboard you should see it on the hyperterminal, as seen below:

шт	era Term ¥	'eb 3.1	- COM1 V	т		_ 🗆 🗵
File	Edit Setup	Web	Control	Window	Help	
You You You You You	Entered Entered Entered Entered	h 1 1			reih	

The Xilkernel project can be downloaded from the link below:

http://xirweb/~stephenm/Projects/Xilkernel Spartan 3AN.zip