PS2-VGA Peripheral based Character Display using FPGA

B. Murali Krishna Assistant professor Dept of ECE K L University V. G. Santhi Swaroop M.Tech, student Dept of ECE K L University

Habibullah Khan, PhD. Professor Dept of ECE K L University. K. Gnana Deepika M.Tech, student Dept of ECE K L University

ABSTRACT

The aim of this paper is to print characters by providing the interfaces to connect a VGA port for graphical output and a PS/2 port for keyboard input. This system is then to be synthesized on an FPGA board, and should be capable of running a program to display characters that takes input from the keyboard and prints the results to a VGA monitor.

This paper is developed under the environment of Xilinx ISE and Platform Studio. It includes Xilinx provided hardware, VHDL IP core and MICROBLAZE processor, the characters inputted by ps2 keyboard, and will be displayed on the VGA monitor. Based on Spartan 3E Starter Board Hardware Architecture, Software Development mainly consists of two parts: one is that PS/2 keyboard scan code receiving and processing program, the other one is VGA displaying program.

Keywords: Xilinx platform studio, Xilinx-c, IP core, MICROBLAZE processor

1. INTRODUCTION

Embedded Development Kit (EDK) is a suite of tools and Intellectual Property (IP) that enables you to design a complete embedded processor system for implementation in a Xilinx Field Programmable Gate Array (FPGA) device. Xilinx Platform Studio (XPS) is the development environment used for designing the *hardware* portion of your embedded processor system using MICROBLAZE processor.

The MICROBLAZE has a versatile interconnect system to support a variety of embedded applications. MICROBLAZE'S primary I/O bus, the Core Connect PLB_bus, is a traditional system-memory mapped transaction bus with master/slave capability. The MICROBLAZE processor is a 32-bit Harvard Reduced Instruction Set Computer (RISC architecture optimized for implementation in Xilinx FPGAs with separate 32-bit instruction and data buses running at full speed to execute programs and access data from both on-chip and external memory at the same time. The backbone of the architecture is a single-issue, 3-stage pipeline with 32 generalpurpose register, an Arithmetic Logic Unit (ALU), a shift unit, and two levels of interrupt. This basic design can then be configured with more advanced features to tailor to the exact needs of the target embedded application such as: barrel shifter, divider, multiplier, single precision floating-point unit (FPU), instruction and data caches, exception handling, debug logic, Fast Simplex Link (FSL) interfaces and others. This

flexibility allows the user to balance the required performance of the target application against the logic area cost of the soft processor. Figure 1 shows a view of a MICROBLAZE system. The items in white are the backbone of the MICROBLAZE architecture while the items shaded gray are optional features available depending on the exact needs of the target embedded application. Because MICROBLAZE is a soft-core microprocessor, any optional features not used will not be implemented and will not take up any of the FPGAs resources.

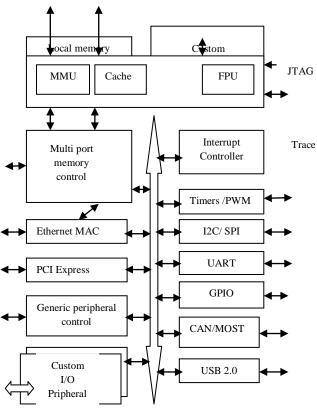


Fig 1. MicroBlaze Processor System Architecture

Writing software to control the MicroBlaze processor must be done in C/C++ language. Using C/C++ is the preferred method by most people and is the format that the Xilinx Embedded Development Kit (EDK) software tools expect. The EDK tools have built in C/C++ compilers to generate the necessary machine code for the MicroBlaze processor. The MICROBLAZE processor is useless by itself without some type of peripheral devices to connect to and EDK comes with a large number of commonly used peripherals. Here we are using PS2 and VGA peripherals to develop the application in EDK-XPS using MICROBLAZE processor in SPARTAN 3E starter board.

2. PS2 PRIPHERAL

The Spartan-3E FPGA Starter Kit board includes a PS/2 mouse/keyboard port and the standard 6-pin mini-DIN connector, labeled J14 on the board. Only pins 1 and 5 of the connector attach to the FPGA.

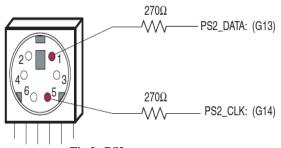


Fig 2. PS2 connector

Both a PC mouse and keyboard use the two-wire PS/2 serial bus to communicate with a host device, the Spartan-3E FPGA in this case. The PS/2 bus includes both clock and data. Both a mouse and keyboard drive the bus with identical signal timings and both use 11-bit words that include a start, stop and odd parity bit. However, the data packets are organized differently for a mouse and keyboard. Furthermore, the keyboard interface allows bidirectional data transfers so the host device can illuminate state LEDs on the keyboard.

Ps/2 DIN Pin	Signal	FPGA Pin		
1	Data (PS2_DATA)	G13		
2	Reserved	G13		
3	GND	GND		
4	+5v			
5	CLK	G14		
6	Reserved	G13		

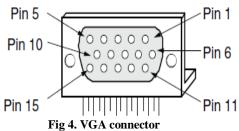
Fig 3. PS2 connector pinout

The keyboard uses open-collector drivers so that either the keyboard or the host can drive the two-wire bus. If the host never sends data to the keyboard, then the host can use simple input pins. A PS/2-style keyboard uses scan codes to communicate key press data. Nearly all keyboards in use today are PS/2 style. Each key has a single, unique scan code that is sent whenever the corresponding key is pressed. If the key is pressed and held, the keyboard repeatedly sends the scan code every 100 ms or so. When a key is released, the keyboard sends an "F0" key-up code, followed by the scan code, regardless if a key has different shift and non-shift characters and regardless whether the Shift key is pressed or not. The host determines which character is intended.

Some keys, called extended keys, send an "E0" ahead of the scan code and furthermore, they might send more than one scan code. When an extended key is released, an "E0 F0" keyup code is sent, followed by the scan code. The keyboard sends commands or data to the host only when both the data and clock lines are High, the Idle state. Because the host is the bus master, the keyboard checks whether the host is sending data before driving the bus. The clock line can be used as a clear to send signal. If the host pulls the clock line Low, the keyboard must not send any data until the clock is released.

3. VGA PERIPHERAL

The Spartan 3E FPGA Starter Kit board includes a VGA display port via a DB15 connector. Connect this port directly to most PC monitors or flat-panel LCDs using a standard monitor cable. The below figure shows VGA connector.



The Spartan-3E FPGA directly drives the five VGA signals via resistors. Each color line has a series resistor, with one bit each for VGA_RED, VGA_GREEN, and VGA_BLUE. The VGA_HSYNC and VGA_VSYNC signals using LVTTL or LVCMOS33 I/O standard drive levels.

Drive the VGA_RED, VGA_GREEN, and VGA_BLUE signals High or Low to generate the eight colors shown below

VGA_RED	VGA_GREEN	VGA_BLUE	RESULTING
			COLOUR
0	0	0	black
0	0	1	blue
0	1	0	green
0	1	1	cyan
1	0	0	red
1	0	1	magenta
1	1	0	yellow
1	1	1	white

Fig 5. 3 bit binary color codes

4. IMPLEMENTATION4.1 PS2-VGA initialization using XPS

XPS-PS2 is the IP core that is provided by XILINX plat form studio after creating a sample project. Xps_ps2 is added by selecting IP catalog tab in project information tab in XPS. The selected IP core will appear in bus interface tab of system assembly view. The ps2 controller will be connected to mb_plb i.e. the processor local bus connection in order to perform communication with MICROBLAZE processor.

Vga_0 custom IP core created using create or import peripheral wizard in XPS. The "Create/Import Peripheral Wizard" in the Xilinx Embedded Development Kit (EDK) assists increasing a custom peripheral that is accessible from a MICROBLAZE processor.

👌 Xilina Platform Studio - CH, sersivity Desktopi puaroopi pystemump - Bystem Assembl	(ieu)	- 6 . .
🔶 File Edit View Project Hardware Software DeviceConfiguration Debug	Simulation Window Help	_ 6 X
0) 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	90 3 5 3 3 3 5 5 5 6 6 6 7 5 6 6 6 6 6 6 6 6 6 6 6 6 6 	
PCablog ↔ D d	X P L L Bus Interfaces Ports Addresses	Bus Interface Filters
28	L M N Name BusName PType Pilesion PClassification	🗄 By Connection
Description Pilesion Pilype Status	nicoblaze 7.21a Processor	🗑 Connected
e EKintal - C'Alimilia)	dină 🕂 Îmb di 1.00a UMBBus	- 🖉 Unconnected
E Analog	drb drb UI 1JJa UMBas	🕀 By Bus Standard
B Bus and Bridge	- mb ob 🚽 ob v45 1,84a PLBV458us	- 🛛 UMB
E Oock, Reset and Interrupt	0-+0- 🗟 dinb ontr 🙀 Inb bran i 2206 Menory Controller	🗸 PL8V45
Communication High-Speed	👍 🛶 🕀 dró otír 👘 kró bran i 2016 🛛 Menory Cottoller	- 🖉 FSL
- Communication Low-Speed	ind box w bram block 1.10.a Memory	🖉 Xilinx Point To Point
🛉 XPSIIC Interface 🛛 201.a xps.jic 🖕 PREFERRE	andn 0 grada 100e Debug	 By Interface Type
- 🐈 IPS PS2 interface 🛛 IILa 🛛 ups ps2 🚽 PREFERRE	🖶 igo 🖉 👷 ya 🛛 Illia Perpheal	🗸 Saves
🛉 XPSSPI Interface 2.01.a xps.spi 🛉 PREFERRE	913 mb pb	🖉 Masters
- 🐈 XPS UART (19550-s., 201.a 🛛 xps uart16550 🖕 FREFERRE	🖶 🗄 Autors, Alt 👷 ups gaio 🛛 2010a 🛛 Peripheral	- 🖉 Master Slaves
🛉 XPSUART (Lite) 🛛 1.0La 🛛 xps_uardite 🖕 PREFERRE	🖯 ups.ps2,0 👷 ups.ps2 11La Peripheral	🖉 Monitors
B- DMA and Timer	- 913 mb da	🖉 Targets
E Debug	🛛 🖗 KIZZLOCE 👷 ups warfite 1.11.a Peripheral	🛛 Initiators
🗟 General Purpose 10	R222,DTE ps wafte 1.1La Peripheal	
∃ 10 Modules	- dod gener. 👷 clock gene 300.a 🛛 P	
E Interprocessor Communic	- projejje. 👷 projejje. 201a Perpheal	
Memory and Memory Con		
B PO		
Perpheral Controller		
8 Processor		
B Utility		
🕀 Project Local pcores Cilliser.		
⊕ USER		
- 👷 vga 🛛 1.01.a vga 👷 PREFERRE		
C		
Project Applications IP Catalog	🔶 System Assembly View 🔶 Book Diagram 🗵 Design Summary 📄 Tiestikop, Memory	а С
Consie		+□8×
make -f system.make TestApp Hemory microblaze 0 program	artei	4
make: Nothing to be done for 'TestApp Memory microblaze 0	program'.	
Ime!		
<		÷
Console Illenings Brous		

Fig6. IP core creation for PS2 & VGA peripherals

4.2 Making PS2-VGA ports externals

After the initialization of PS2-VGA IP cores the respective ports must be made external. After making the external the ports of the peripherals connected to the MICROBLAZE processer local bus.

° Catalog			⇔⊡ð×	aus interfaces Ports	Addresses			🚀 Add External Port	Port Filters
ŧ				Name	Net	Direction	Range	Class *	B-By Connection
escription	P lesion	IP Type	Status	e ano					- Defaults
EDKInstall C\Xiimi1.1\				🕞 ânb					- 🛛 Connected
R Araloo				dq_dn ⊕					🛛 🛛 Unconnected
Bus and Bridge				- dinb_ontir					8-By Class
Clock Reset and Interrupt				- and_critir					🛛 🗹 Clocks
Communication High-Spee	d			- Inb_brom					- 🛛 Resets
R Communication Low-Speed				0 mbm 🖲					🛛 🛛 Interrupts
1951C Interface		105.IC	+ PREFERRED	⊖ vga_0					🛛 💟 Others
75 PS2 Interface		105.052	+ PREFERRED	- H6jt	vga,0,H5,t	• 0			B-ByDirection
1PS SPI Interface		105,50	PREFERRED	- VS_t	vga 0,VS,t	.0			– 🔽 Inputs
🚽 XPS UART (16550-s.		xps uart16550	+ PREFERRED	RED_t	vga_0_RED_t	.0			- 🛛 Outputs
🐇 XPS UART (Lite)		xps uatite	PREFERRED	GREEN t	vga_0_GREEN_t	.0			- 🛛 hOuts
R DNA and Timer		1000		BLUE)	vga_0_BLUE_t	• 0		8	
# Debug				🕀 Buttons_450					
General Purpose 10				⊖ xps_ps2,0					
# 10 Modules				IP2INTC_lipt_1	No Connection	.0		INTERSUPT	
🗧 Interprocessor Communic.				PS2,1,DATA,I	No Connection	• 1			
Memory and Memory Con.				PS2_1_DATA_O	No Connection	.0			
R PCI				PS2,1,DATA,T	No Connection	.0			
+ Peipheral Controller				- P\$2,1,0,K)	No Connection	•1			
- Processor				- PS2_1_CLK_0	No Connection	.0			
8 Utility				PS21.CX(T	No Connection	.0			
Project Local pcores Ci User.				PS2_1_DATA	aps_ps2_0_PS2_1_DATA	• 10			
in USER				- PS2,1,CLK	aps_ps2.0_PS2.1_CLK	• 10		-	
- 1 100	100a	103	+ PREFERSED	⊕ RS232_DCE					
A 12*	2004	194	Automat	⊕ #\$232_DTE					
	1		•	e)	4 E
Project Applications IP Catal	y I			👌 Sisten Assenbly Ver	👌 Bod Dagram	Σ Design	Sumary	Testikap Nenorikis	
	3				v			, any series	-
ssie									*
Memory Initialization	completed	successfully.							

Fig7 . PS2-VGA ports are made external

4.3 Adress generation

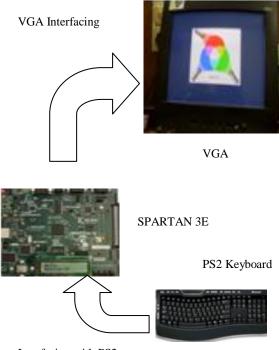
After making the ports externals of these two peripherals the adress will generate at adress tab in system assemly view. It gives the base adress and high adress for the both PS2 and VGA peripherals.

After the adress genaration MHS(Microprocessor hardware specification) and MPD(Microprocessor peripheral descripton) must be edit depends upon peripheral device configuration. The UCF(user constraint file) must be given at the system.ucf in project information tab.

P Catalog			+□8×	Bus Interfaces	Parts	Address	8					🔝 Generate Add
8				Instance	Base Na		Base Address	High Address	Sce	BusInterface(s)	Rer Dires	IP Version
Description	P Version	P Type	Status	e microblaze		110	PERCHARINES	IIIyi Ausos	712	anomereisi	ano n ilhe	a result
EDK Install C/Xiim11.1		21		- dinb c		INNP	A000000	(MOOD/FFF	32(J SLMB	d. 🛉 inb ban i.	210.6
Analco				- imb crth			0.0000000	0.00017FFF	320	SUMB	il. 🛊 Inb bran i.	
Bus and Bridge				Buttons			0.6140000	0x8140FFFF	54K	- 98	m tras colo	
 Clock, Reset and Interrupt 				- RS232 D.			0.6400000	0x8400FFFF	64(• 9B	m., y us uatite	
 Communication High-Sper 	u.			- RS232 D.			0.6402000	0x8402FFFF	64K	• 9B	m., y us uatite	
E Communication Low-Spee									54K	• 96		1.00.e
	201.a		+ PREFERRED	- mdm_0			0x94400007	0x8440FFFF			m 🛊 mdm	
- VPSIC menace		xps_ic	PREFERSED	nte ber			0.6540000	0.86AUFFFF	64(₽ \$PLB	m 👷 ups_ps2	18La
		xps_ps2		- vga_0	C BASE	4UUK	DACCENNOO	0xCC60FFFF	64K	🖌 SPLB	m 👷 vga	1.00.a
👷 XPS SPI Interface		xps_spi	REFERRED									
- 👷 XPS UART (06550-s		xps_uart16550	T PREFERRED									
- 🙀 XPS UART (Lite)	101 <i>a</i>	xps_uartite	🛊 PREFERRED									
🗟 DMA and Timer												
🗟 Debug												
🗟 General Purpose 10												
🖶 10 Modules												
🗟 Interprocessor Communic.												
Memory and Memory Con												
🖶 PCI												
Peripheral Controller												
Processor												
# Utility												
Project Local pcores Cillser												
in USER												
-1/19	100a	ndis	🙀 PREFERRED									
	1		•									
Project Applications IP Catal	9			👌 System Ass	enbly View	٢	Block Diagram	Σ	Design Summa	ry 🖹 Testa	pp_Memory.c	
insie	_		2011723100			****						+ 🗍
Memory Initialization	completed :	successfully.										
Done!												

5. EXPERIMENTAL SETUP

The total experiment is done by using spartan 3e starter kit (XC3S500E-Device family,FG 320 package and 4-speed grade). The interface between PS2 and VGA is done by using Xilinx Platform Studio.The total application is developed by writing required logic in Embedded-C.



Interfacing with PS2

Fig9. Experimental setup

6. EXPERIMENTAL RESULTS



Fig10 .Hardware setup for PS2-VGA Peripherals



Fig 12 . Display of characters on vga using PS2



Fig13. Deteting the characters using backspace key



Fig14 . Display of characters using RGB colours

		XONAABREE				RE EE					
lapications - D & x	õ	Design Dramlew Summary	- }/	anation in the	Ner 27 1238 18 20 12	ju -	_	14,1758.00	101.0.02020	-	
Software Projects	ŏ	Summary 108 Properties	1								
Add Software Application Project	0	- S Module Level Utilization		XPS Synthesis Summary							
	Timing Constraints		Report	Generated	thp tiops Used	LUTs Used	SRAUG Used	trors			
🗧 🧾 Project: TestApp Memory microblaz.		Clock Report	83	satter.	lifed Har 28 15:00:55 2012	1905	2987	20			
Processor microbilize_0				roa 0 seaport	Wed Har 28 15:03:38 2012	200	405				
Executable: C/Users/admin/Desktop/	1	Platgen Messages		108 282 0 MINOR	Hed Him 28 15:00:29 2012	361	415				
Sources	A	- Simper Messages		proc sus reset & wrapper	Hed Har 28 15:00:13 2012	67	53				
C//Uses/admirl/Desktop/gs2vga/	194	Bithit Messages	Ę	rade. D. wrapper.	10ed Har 28 15:03:30 2012	10	140		-		
Headers Project: TestApp_Peripheral_microbia.		XPS Reports Platgen Log File		dads generator 0 wrapper	Wed Her 28 15:03:05 2012	5	2				
8 Processon microbilace 0		Libgen Log File		rs232.dze.wrapper	Hed Har 28 15:03:01 2012	3%	134				
Executable: C/UsesLadmini/Desktop/		Segan Log File Bitlet Log File Brost and Remany: Translation Messages Man Message Man Message Months Enknowed Design Summer		inb bram snapper	Med Har 28 15:02:50 2012			16			
E Sources				inb oth water	Hed Har 28 15:02:47 2012	2					
I Headers				dirà cità vracori	lifed Har 28 15:02:44 2012	2	6				
				deb anapper.	Hed Har 28 15:02:41 2012	1	1				
				inb wratter	Hed Mar 28 15:02:38 2012	1	1				
		- 🗵 Display Incremental Messages		the all wrapper.	Hed Har 28 15:02:35 2012	154	383				
		Enable Message Filtering Optional Design Surremary Contents		microbiane O vractor	10ed Her 28 15:02:23 2012	\$17	101				
		Show Clock Report									
		Show Failing Constraints Show Warrings		Device Utilization Semmany							
		- Show literangs		Logic Utilization	Used	Available	Utilizat	ion Note	(4)	F	
	-	Show Partition Data			-	10		1971			
Project Applications IP Catalog	۲	System Americky Vera 🛛 🔶		Sock Diagram	E Design Summery	B 1	stApp_Hemory.c				
ansde										• 1	
INFO:INFACT - '1': Programing INFO:INFACT - '1': Checking d Done!											

Fig15. Design summery

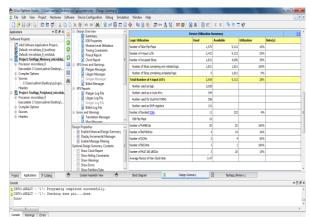


Fig16. Device utilization summary

7. CONCLUSION

The Xilinx platform studio under MICROBLAZE processor is used to design embedded processor to develop applications with peripheral devices using fpga. Some files are written depends upon the peripheral configuration and specifications those are MHS,MPD,MSS and UCF to develop the interface between peripherals and intilization. We can run any of the application which is working through PS2-VGA peripherals. After the peripheral is imported, user can use the peripheral in the design by using the XPS flow process. The software application can be written in a "C or C++" to control I/O peripherals through serial communication then the complete embedded processor system for user application will be completed and verified by generating and downloading the bit file into actual hardware.

International Journal of Computer Applications (0975 – 888) Volume 48– No.9, June 2012

8. REFERENCES

- [1] "EDK Concepts, Tools, and Techniques" by Xilinx Corporation.
- [2] "EDK MICROBLAZE Tutorial" by Xilinx Corporation.
- [3] Spartan-3E FPGA Starter Kit Board User Guide http://www.xilinx.com/support/documentation/boards_an d_kits/ug230.pdf
- [4] "Create or Import Peripheral wizard on EDK" by Xilinx Corporation.
- [5] Wikipedia http://en.wikipedia.org/wiki/Fpga.
- [6] www.computerengineering.org/ps2keyboard/scancodes2

- [7] "A Control Design Approach for Controlling an Autonomous Vehicle with FPGAs" JOURNAL OF COMPUTERS, VOL. 5, NO. 3, MARCH 2010.
- [8] "Soft core processors and Embedded processing : a survey and analysis" by Humberto Calderón, Christian Elena and Stamatis Vassiliadis.
- [9] "Experiences with Soft-Core Processor Design" Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium.
- [10] IEEE "Efficient Design and Implementation on FPGA of a MicroBlaze Peripheral for Processing Direct Electrical Networks Measurements" by J. Viejo, M.J. Bellido, A. Millan, E. Ostua, J. Juan, P. Ruiz-de-Clavijo and D. Guerrero.