International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 3, Issue 9, September 2014

USB TRANSCEIVER MACROCELL INTERFACE WITH USB 3.0 APPLICATIONS USING FPGA IMPLEMENTATION

T Mahendra[1], N Mohan Raju[2], K Paramesh[3]

Abstract— The Universal Serial Bus(USB) Transceiver Macro cell Interface (UTMI) could be a 2 wire, bi-directional serial bus interface. The USB3.0 specifications outline 3 forms of UTMI implementations depends on knowledge transmission rates, those area unit Low Speed (1.5MHz) solely (LS), Full Speed (12MHz) solely (FS) and High Speed (480MHz)/Full speed (12MHz) (HS). UTMI consists of transmission and Receiving sections, during which the Transmitter of the UTMI sends knowledge to totally different USB devices through D+ and D- lines whereas the Receiver gets knowledge on a similar lines. This presentation reveals the FPGA implementation of UTMI with HS/FS transmission rate providing with USB two.0 specifications. more UTMI has been designed by exploitation VHDL code and simulated, synthesized and programmed to the targeted Spartan2 family of FPGA within the Xilinx surroundings.

Index Terms— **USB 3.0 Transceiver Macro Cell,Full Speed,Low Speed, Serial Interface Engine,Device Specific Logic,Bit Stuff,Bit Un Stuff,Encoder,Decoder**

I. INTRODUCTION

The Universal Serial Bus (USB) Transceiver Macrocell Interface (UTMI) may be a 2 wire, bi-directional serial bus interface between USB devices through D+ and D- lines. this can be one among the necessary useful blocks of USB controller, which might transmit and receive knowledge to or from USB devices. There square measure 3 useful blocks gift in USB controller; those square measure Serial Interface Engine (SIE), UTMI and Device Specific Logic (DSL). Figure one shows the diagram of UTMI. The parallel knowledge from SIE is taken into the transmit hold register and this knowledge is shipped to transmit register from wherever the information is reborn serially. This serial knowledge is bit stuffed to perform knowledge transitions for clock recovery and NRZI (1) encryption. Then the encoded knowledge is shipped on to the serial bus. once the information is received on the serial bus, it's decoded, bit unstuffed and is shipped to receive register. once the register is full, the information is shipped to receive hold register.

This knowledge are bestowed on the port wherever it's sampled by the SIE. The intent of the UTMI is to accelerate USB two.0 peripheral development.

T.MAHENDRA, *PG Scholar, Department Of ECE, Kottam College Of Engineering, JNTUA, Kurnool, A.P ,INDIA, 9491408401*

N.MOHAN RAJU, *Assistant Professor, Department Of ECE, Kottam College Of Engineering, , JNTUA, Kurnool, A.P ,INDIA, 9014111062.*

K.Paramesh*, Head, Assistant Professor, Department Of ECE, Kottam College Of Engineering, JNTUA, Kurnool, A.P ,INDIA ,9966043638.*

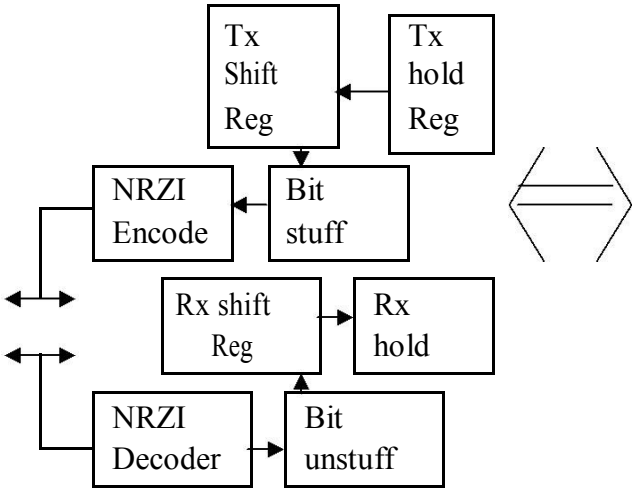


Figure 1: Block diagram of UTMI.

Features of the UTMI:

- ✓ Supports 480 Mbit/s High Speed (HS)/ twelve Mbit/s Full Speed (FS), FS solely knowledge transmission rates.
- ✓ Utilizes 8-bit parallel port to transmit and receive USB two.0 cable knowledge.
- ✓ SYNC/EOP generation and checking.
- ✓ Data and clock recovery from serial stream on the USB.
- ✓ Bit-stuffing/unstuffing and bit stuff error detection. Holding registers to stage transmit and receive knowledge.
- ✓ Ability to modify between FS and HS terminations/signaling

II. Design Aspects

The present UTMI has been designed consistent with the subsequent specifications provided by the USB two.0 protocol. correct and finish of Packet (EOP) generation by the transmitter. Correct and EOP detection by the receiver receive error reportage. enabling or disabling the bit stuffer and NRZI encoder depends on the operational mode. Suspension of the transceiver by the SIE.

Further the UTMI is split into 2 necessary modules that ar the Transmitter module and also the Receiver module. during this section the look issues of those modules are explained individually and integrated to urge prime level Transceiver (UTMI) module.

A. The Transmitter Module

The diagram of the UTMI transmitter is shown in Figure2. The transmitter module has been enforced by considering the subsequent specifications.
The synchronise pattern "01111110" should be transmitted like a shot once the transmitter is initiated by the SIE. once six consecutive '1's occur within the knowledge stream a zero to be inserted. the information ought to be encoded victimization Non come to Zero Invert on one (NRZI -1) cryptography technique. The EOP pattern 2 single over zeroes(D+ and D-lines ar carrying zero for 2 clock cycles) and somewhat one ought to be transmitted once every packet or once SIE suspends the transmitter

The transmitter logic facilitates synchronise transmission, holding parallel 8- bit knowledge from SIE, parallel to serial conversion of information, bit stuffing, NRZI cryptography , transmission of the information and EOP transmission on to the serial bus.

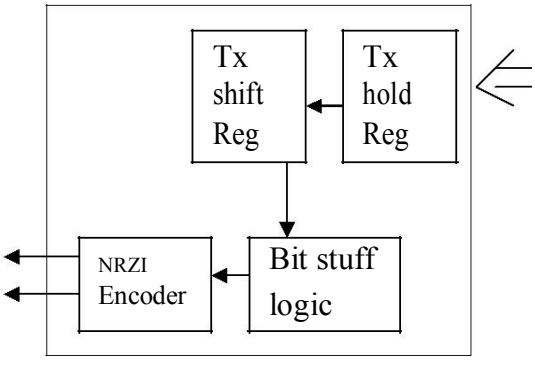


Figure 2: Block diagram of UTMI Transmitter.

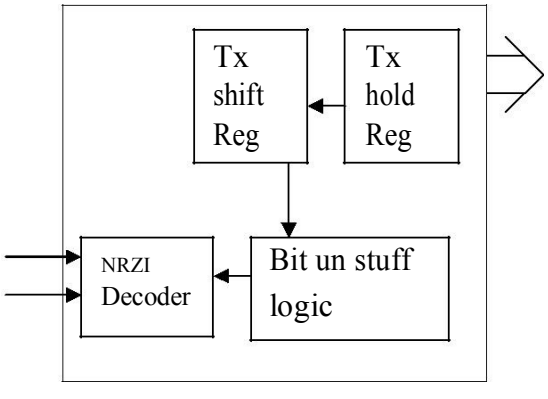


Figure 3: Block diagram of UTMI Receiver.

B. The Receiver Module

The diagram of the UTMI receiver is shown in Figure three. The receiver module has been enforced by considering the subsequent specifications.

When correct pattern is detected that ought to be intimated to the SIE.

If a zero isn't detected when six consecutive '1's a slip-up ought to be reported to the SIE.

When EOP pattern is detected that ought to be intimated to the SIE.

The receiver logic facilitates correct detection, NRZI decryption, bit unstuffing, serial to parallel conversion of knowledge, receive error reportage and EOP detection

C. The Transceiver Module

The transmitter and therefore the receiver modules area unit combined along to style the transceiver (UTMI) module. This transceiver met all the USB3.0 specifications thought-about on top of. The transceiver logic facilitates the output of the transmitter to feed to the input of the receiver for purposeful verification. The Transceiver module has been designed with the issues of individual modules of the transmitter and therefore the receiver Specifications.

Any the specified Transceiver module logic has been verified with the purposeful simulation followed by necessary Synthesis and perform Programming to the targeted FPGA Device.

III Simulation Results

The individual modules of the UTMI are designed using VHDL as stated above and they are simulated within the Xilinx based Model Sim 6.0 environment.

A. The Transmitter Module

The Figure 4 shows the Simulation results of UTMI transmitter. When TX valid signal goes high, encoded SYNC pattern "01010100"is transmitted and the signal tx ready is asserted. The data "10110100" present on the dataln bus is NRZI encoded and transmitted on to the txdp, txdm lines. The signal tx ready goes low when the data is sampled by the TX hold register.

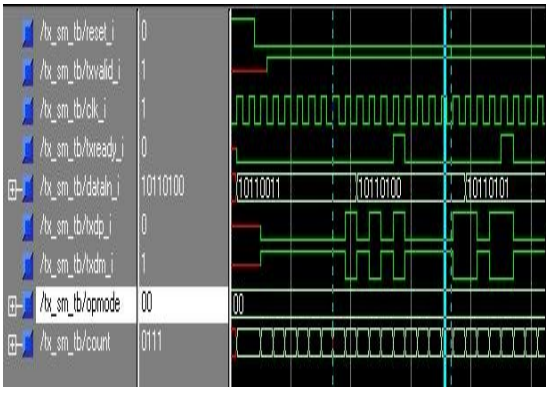
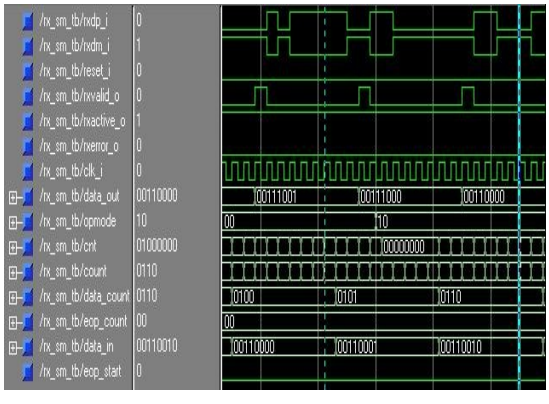


Figure 4: UTMI Transmitter Module

B. The Receiver Module

The Figure 5 shows the Simulation results of UTMI receiver. once correct is detected rxactive is declared. the info gift on rxdp, rxdm lines is decoded, serial to parallel born-again and sent to the SIE through knowledge out bus by declarative rxvalid signal.



C. The Transceiver Module

The Figure 6 shows the Simulation results of the Transceiver module that transmits and receives knowledge. once txvalid goes high, synchronize is transmitted. the information "00000000" gift on the data_bustx is NRZI encoded and transmitted on to the DP, dm lines.

When synchronize is detected by the receiver rxactive is declared by the UTMI. the information gift on the DP, dm lines is NRZI decoded and sent to the SIE through rxdata_bus by declarative rxvalid signal. Rxdata_bus contains "00000000" since the transmitted knowledge is fed back to the receiver.

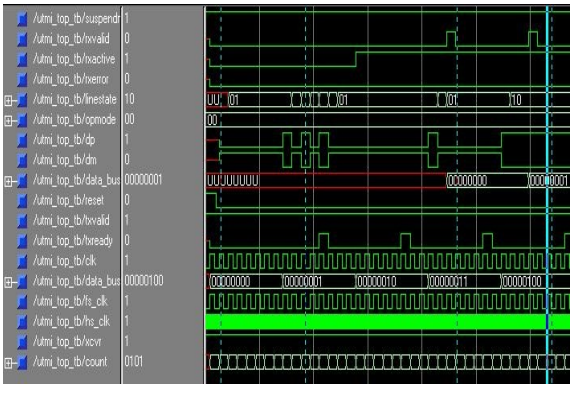


Figure 6: Transceiver (UTMI) Module.

IV FPGA Implementation

The top order module, UTMI is synthesized inside the Xilinx 8.1 ISE software package tool and it's programmed to the targeted SPARTAN 2 a pair of family of FPGA Device. the assorted levels pf implementation like Synthesis report, RTL View, Place and Route Report and Device Programming has been explained and unreal within the following sub sections.

MYUTMI Project Status			
Project File:	myutmi.ise	**Current State:**	Programming File Generated
Module Name:	utmi_top	**• Errors:**	No Errors
Target Device:	xc2s15-6cs144	**• Warnings:**	6 Warnings
Product Version:	ISE, 8.1i	**• Updated:**	Fri Sep 22 02:48:21 2006

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	73	384	19%	
Number of 4 input LUTs	122	384	31%	
Logic Distribution				
Number of occupied Slices	70	192	36%	
Number of Slices containing only related logic	70	70	100%	
Number of Slices containing unrelated logic	0	70	0%	
Total Number of 4 input LUTs	122	384	31%	
Number of bonded IOBs	29	86	33%	
IOB Flip Flops	2			
Number of GCLKs	1	4	25%	
Number of GCLKIOBs	1	4	25%	
Total equivalent gate count for design	1,344			
Additional JTAG gate count for IOBs	1,392			

Table 1: Synthesis Summary

A. RTL View

This section provides the image of Resister electronic transistor Logic (RTL) views within the type of schematic and Netlist diagrams that area unit shown in Figure seven and Figure eight severally. Figure seven which provides RTL schematic diagram reveals the pin diagram of prime order module with such that|the desired} specified notes whereas Figure eight reveals the Gate level logical diagram of prime order module with the desired input and output ports(Netlist view).

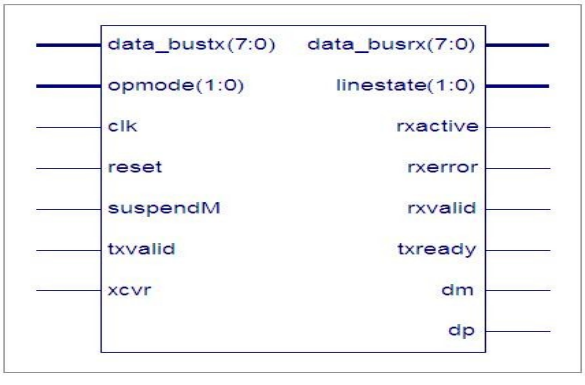


Figure 7: RTL Schematic diagram

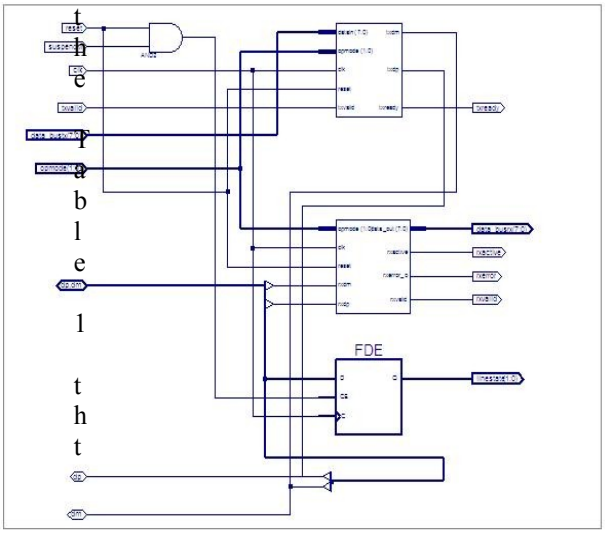


Figure 8: RTL netlist view

B. Place and Route Report

This section concentrates on course right direction|not off course} FPGA device utilization outline that reveals the knowledge needed for correct layout at the extent of producing in the variety of Place and Route report. additional it offers the temporal arrangement synchronization of electronic equipment with the important time environment.

Device Utilization Summary:

Number of GCLKs	1 out of 4	25%
Number of External GCLKIOBs	1 out of 4	25%
Number of LOCed GCLKIOBs	0 out of 1	0%
Number of External IOBs	29 out of 86	33%
Number of LOCed IOBs	0 out of 29	0%
Number of SLICEs	70 out of 192	36%

Total REAL time to Placer completion: 2 secs Total CPU time to Placer completion: 2 secs

C. Device Programming

After successful process of synthesis the Target device xc2s15 of Spartan2 is connected to the system through printer port. The pin assignment is specified in the User Constraint File (UCF). The functional verification is carried out by using a pattern generator.

Synthesis Report:

The below synthesis report is generated by Xilinx nine.2i ISE.

Release 9.2i - xst J.36

Copyright (c) 1995-2007 Xilinx, Inc. All rights reserved.

→ Parameter TMPDIR set to ./xst/projnav.tmp

CPU : 0.00 / 0.15 s | pass on : zero.00 / 0.00 s

→ Parameter xsthdpdir set to ./xst

CPU : 0.00 / 0.15 s | pass on : zero.00 / 0.00 s

→ Reading design: utmi_top.prj

V. CONCLUSION

The individual modules of UTMI are designed victimisation VHDL and verified functionally with the Model Sim 6.0.

The UTMI Transmitter is capable of changing parallel knowledge into serial bits, playing bit stuffing and NRZI secret writing.

The UTMI Receiver is capable of playing NRZI cryptography bitunstuffing and changing serial bits into parallel knowledge.

The purposeful simulation has been with success disbursed. the planning has been synthesized victimisation FPGA technology from Xilinx. This style is targeted to the device family_spartan2, device_xc2s15, and package_cs144 and speed ranked vi. The device belongs to the Vertex-E cluster of FPGAs from Xilinx. The UTMI is meant to support HS/FS, FS solely and LS solely UTM implementations. The 3 choices permit one SIE implementation to be used with any speed USB transceiver. A vender will select the transceiver performance that best meets their wants.

VI. FUTURE SCOPE

The UTMI has been enforced for 8-bit, however it may also be extended to 16- bit UTMI. It may also be designed to come up with CRCs for management and knowledge packets. If AN SIE and Device specific logic square measure designed, the mixture of UTMI, SIE and Device specific logic may be used as a controller of any USB device.

VII. APPLICATIONS

The UTMI has been developed into a standard code (Generalized USB Transceiver) which might be used for developing the whole USB device stack. a number of the Low speed and High speed USB devices, that square measure presently accessible within the market are:

1. Optical Mouse
2. Key Board
3. Printer
4. Scanner
5. Joy Stick
6. Memory Stick
7. Flash Memory
8. Mobiles
9. Video cameras.

REFERENCES

[1] Charles H Roth "Digital system using VHDL".2nd edition, Thomson publication
[2] Jayaram Bhasker "A VHDL Primer" 2nd edition, Prentice Hall publications
[3] Stephen Brown, Zvonko Vranesic "Fundamentals Digital logic with VHDL design". 2nd edition, McGraw-Hill, Hardcover, Published July 2004
[4] Zainalabedin Navabi "Vhdl Analysis and Modeling of Digital Systems" 2nd edition, McGraw- Hill, Hardcover, Published January 1998
[5] William Stallings, "Data and Computer Communications" ,McGraw-Hill Publications.
[6] Andrew S.Tannenbaum," Computer Networks", Pearson publications.
[7] Z.Kohavi, "Switching and finite Automata Theory", Tata Mcgraw-Hill Publications
[8] N.N.Biswas, "Logic design theory", Printice Hall of India Publications.
[9] Morris Mano, "digital design", Tata McGraw-Hill Publications.
[10] Lala, "Digital system Design Using PLDs", BSP Publications.

