A Comprehensive study of Input Output [IO] functionalities of Contemporary IO

[1]Aruna Rao B P, [2]Dr. Shanthi Prasad. M.J
[1]Assistant Professor, [2]Professor
[1]KS Institute of Technology, Bengaluru, India,
[2]Cambridge Institute of Technology, North Campus, Bengaluru.

***Abstract* - As a part of on-going research towards Design, development & testing of next generation multi-functional High speed Universal IO, I undertook an exhaustive survey spanning hundreds of related research documents in the performance of IO with respect to Processors, Memories & Programmable Logic devices [FPGAs] & completed a comprehensive study of IO functionality in the above said category of devices.**

***keywords* - Boundary Scan, Boundary Scan Register, Programmable IO Buffers, BIST-BSR, Spartan IOB.**

NR1 RESEARCH REFERENCE MODULE [RRM]

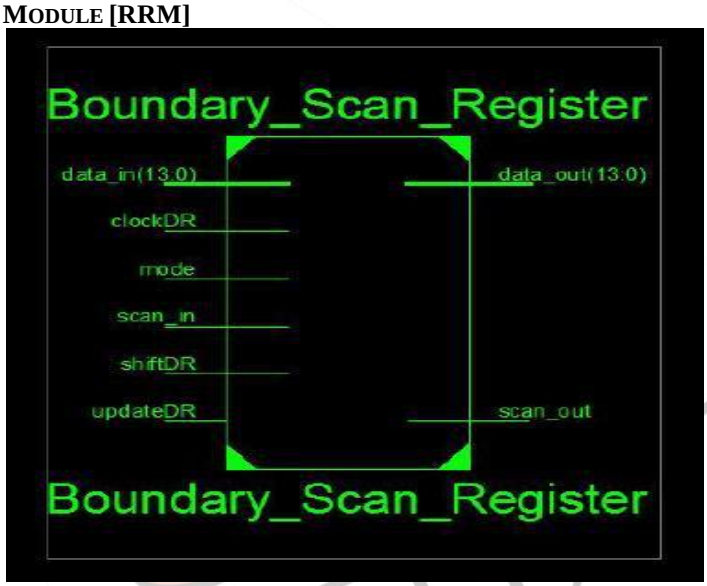

Fig.1 RTL schematic of Boundary Scan Register

In this paper, the authors claimed to have designed & implemented Boundary Scan register (BSR) for tracing & debugging a microcontroller as target device using JTAG protocol. Boundary Scan (BS) has been implemented in Verilog. They have implemented BSR whose RTL has been illustrated. This BSR has 14 bit data input & output, Clock_DR [clock for data register], mode signal to select between capture & shift mode [scan & primary input output], scan input, shift DR & Update DR & scan_out as output. They proposed to integrate this BSR with microcontroller core with an intension of debugging the interfaced microcontroller with BSR.

The technical aspect of this research that influenced my research is that BS can be interfaced with independent cores with the complexity extending up to latest microcontroller.

NR2

In this paper, a test procedure is designed & implemented to test BS cells & Test Access Port (TAP) controller on PLDs & FPGAs. The suggested Test procedure involves 3 steps: 1) configuration of PLs or FPGAs 2) Application of test vectors 3) Verification of the response to ensure high fault coverage. The configuration & application of test vectors in DUT is done through JTAG port. The author has explained in detail JTAG architecture with necessary diagrams & also explained BS cell in typical IO block. He also explained JTAG instructions like SAMPLE, PRELOAD, BYPASS, EXTEST, INTEST, RUNBIST, HIGHZ & also explained BSR & the test procedure to test DUT & also provided table of logic gates & operation performance with the application of test vectors after instruction INTEST is enabled. The table provides the TAP state during capture DR, shift DR & Update DR logic states.

The test data path for input & output cells are illustrated separately during the process of fault detection. A summarized has been provided to explain the tested logic in the IOB BS logic & corresponding struck at 0 & 1 faults.

This paper provided me with a complete knowledge of how BS logic is implemented & tested in any IOB of PLD or FPGA. My BS is an enhanced & customized version of this BS. My BS is a simple design to find out BS IO faults in the BS chain.

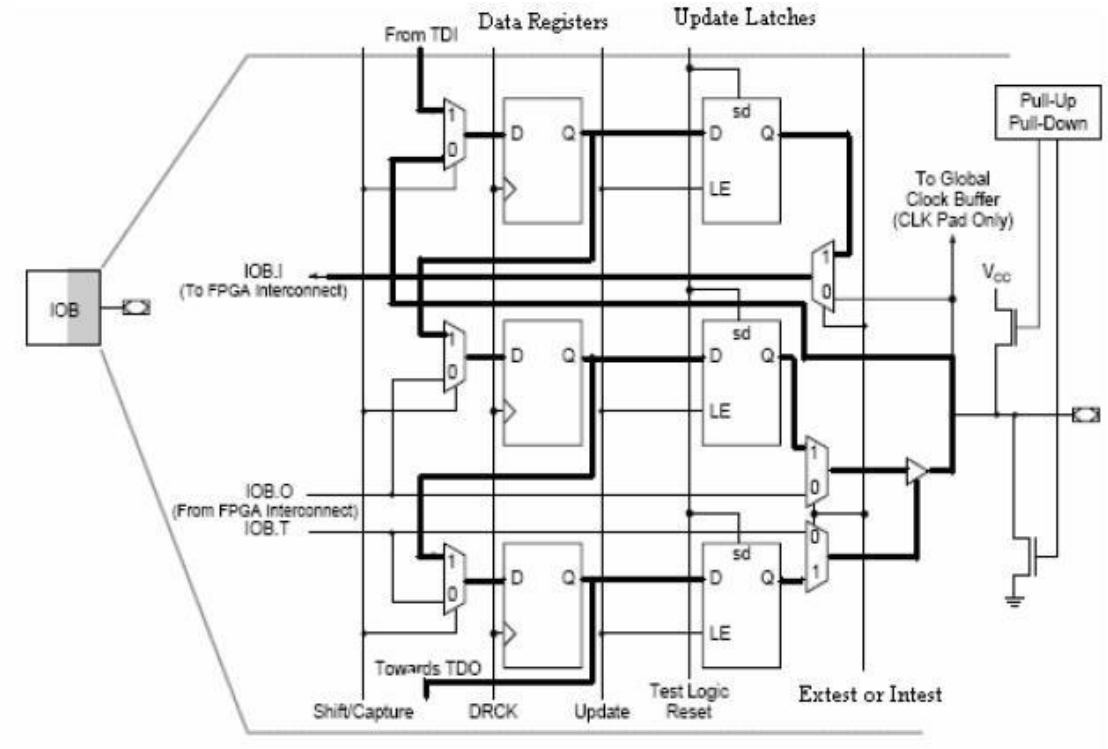

Fig.2 Tested Data lines in an input cell

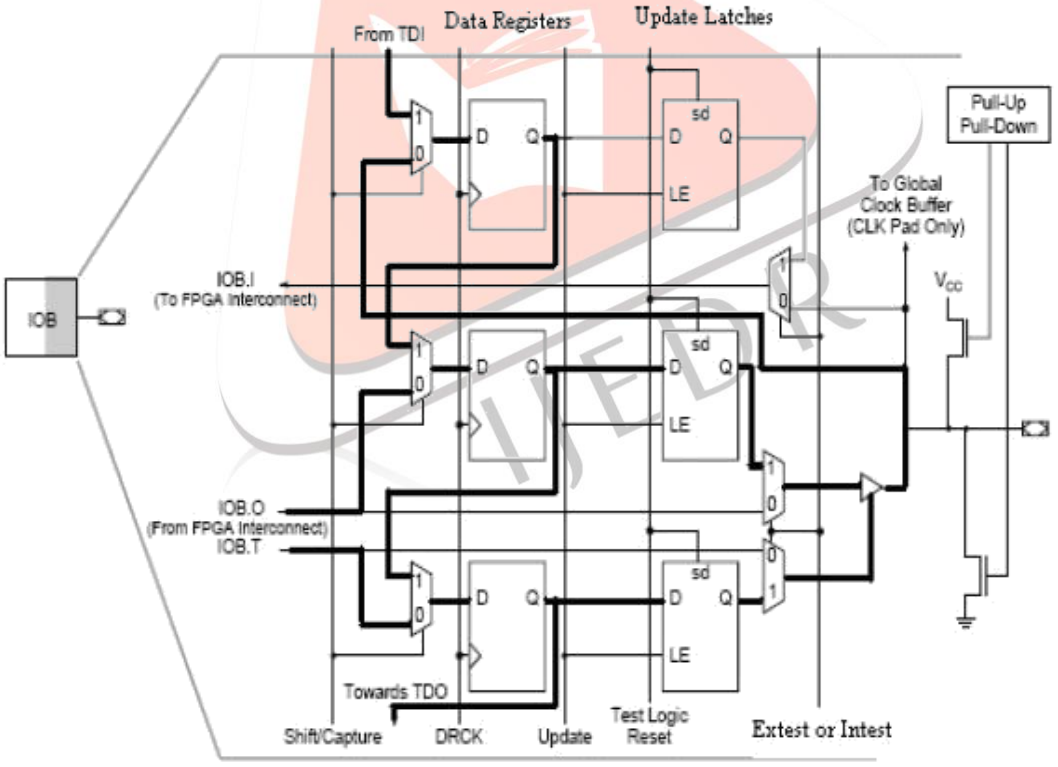

Fig.3 Tested Data lines in an output cell

TAP State	Logic States in Input Cells						Logic State in Output Cells						Operations Performed
	Data Registers			Update Latches			Data Registers			Update Latches			
	T	O	I	T	O	I	T	O	I	T	O	I	
Capture-DR	X	X	X	X	X	X	X	X	X	X	X	X	Loading data registers with undetermined states
Shift-DR	1	0	1	X	X	X	1	0	1	X	X	X	Shifting in 1st test vector
Update-DR	1	0	1	1	0	1	1	0	1	1	0	1	Applying test data to internal logic input and output buffer
Capture-DR	X	X	0	1	0	1	1	1	0	1	0	1	Loading data registers with the specified states
Shift-DR	1	1	0	1	0	1	1	1	0	1	0	1	Shifting in 2nd test vector, and shifting out the response to be verified
Update-DR	1	1	0	1	1	0	1	1	0	1	1	0	Applying test data to internal logic input and output buffer
Capture-DR	X	X	1	1	1	0	1	0	1	1	1	0	Loading data registers with the specified states
Shift-DR													Shifting out the response to be verified
T = the output of data register or update latch which drives the 3-state line of the output buffer.													
O = the output of data register or update latch which drives the input line of the output buffer.													
I = the output of data register or update latch which drives the input of the internal logic.													
X = undetermined logic state.													

Table 1. Logic states & operations performed during the application o two test vectors, and after INTEST has been selected

NR3

In this paper, the authors have presented the overall architecture of IOB array in SRAM based FPGA. They have also explained the various types of IOs present in FPGA like Power IO, Dedicated function IO, programmable IO, Dual purpose IO their functionalities of respective pins used for realization of respective functionalities. They have also illustrated the layout of Logic Blocks in 2 dimensional array of rows & columns surrounded by IOB associated with each pin.

The authors also have provided a detail circuit design of IOB with bidirectional tri-state functionality for input & output operations. They have also illustrated the BS path for JTAG protocol implementation (JTAG instructions). The authors have explained programmable io buffers wherein any specific io pad can be made to drive various logic families like TTL, CMOS etc [HV & LV versions] by implementing programmable pull up & pull down. The authors have only suggested the concept of programmable buffers but not designed or implemented.

The technical aspect of this paper impacting my research was the architecture of IOB with bidirectional, tri-state, BS & programmable IO interface with multiple logic families.

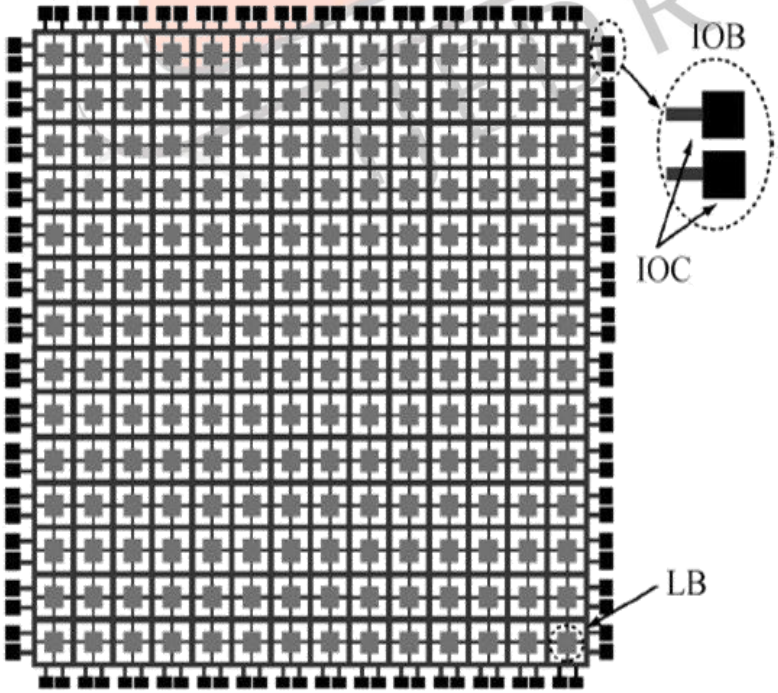

Fig.4 FPGA IO block

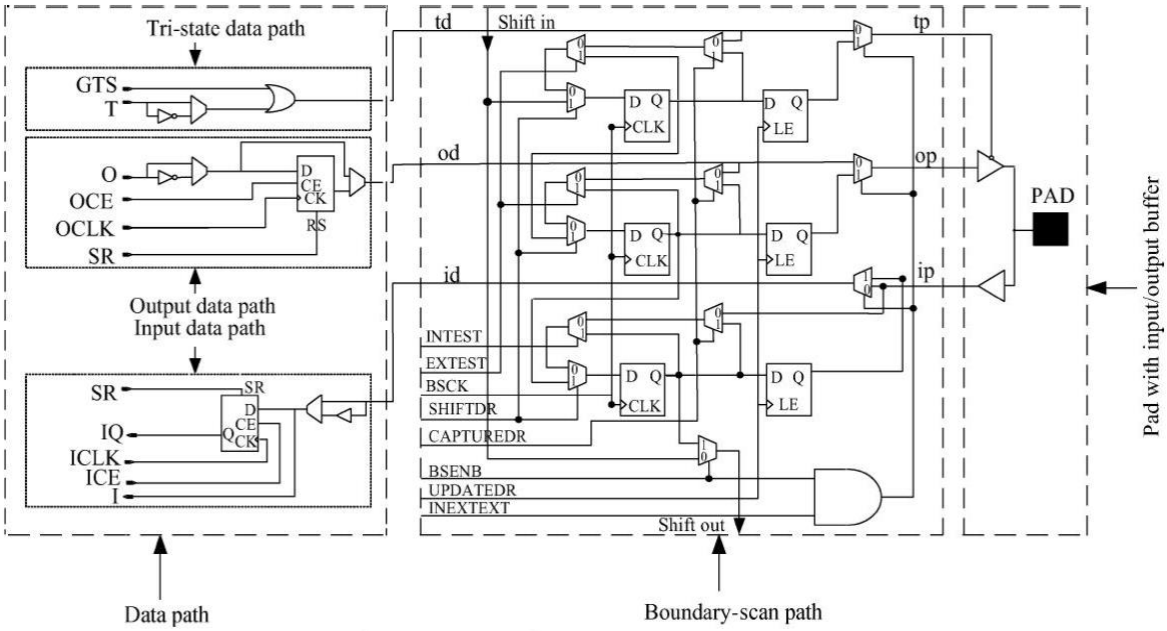

Fig.5 Simplified signal path circuit diagram

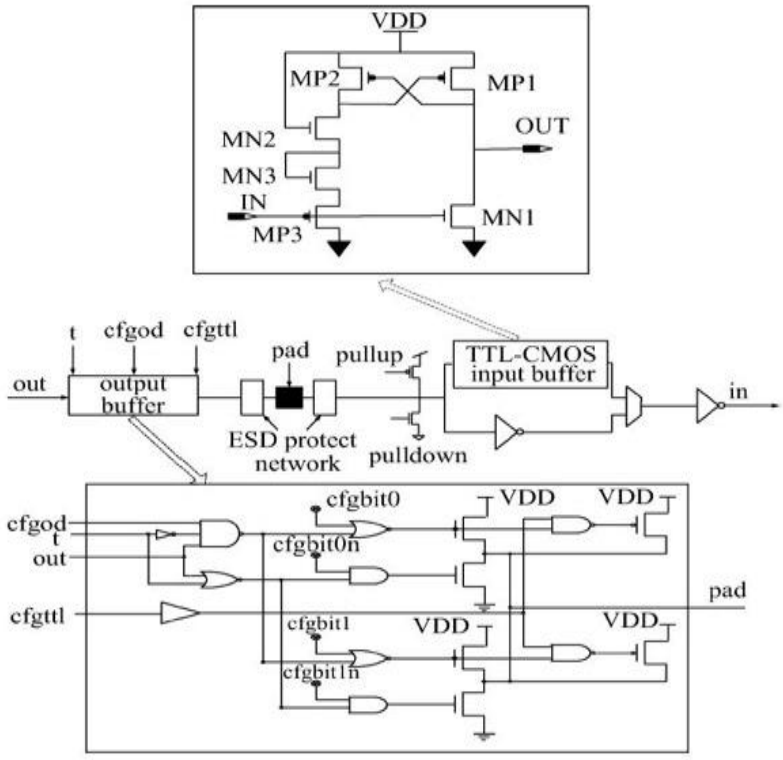

Fig.6 Programmable IO buffers

NR4

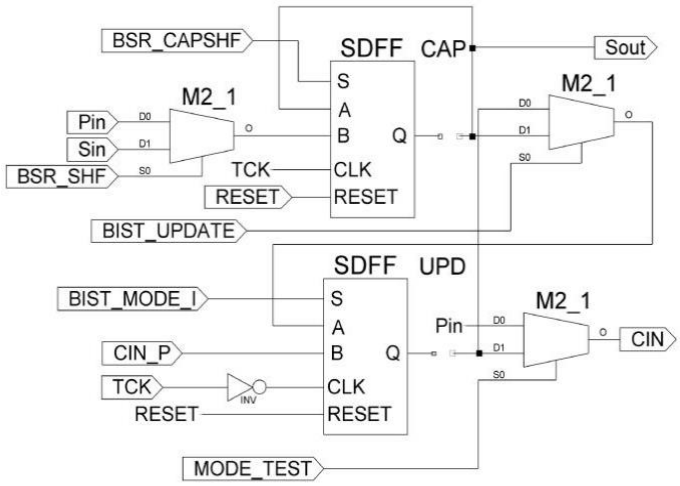

Fig.7 BIST-BSR Input cell

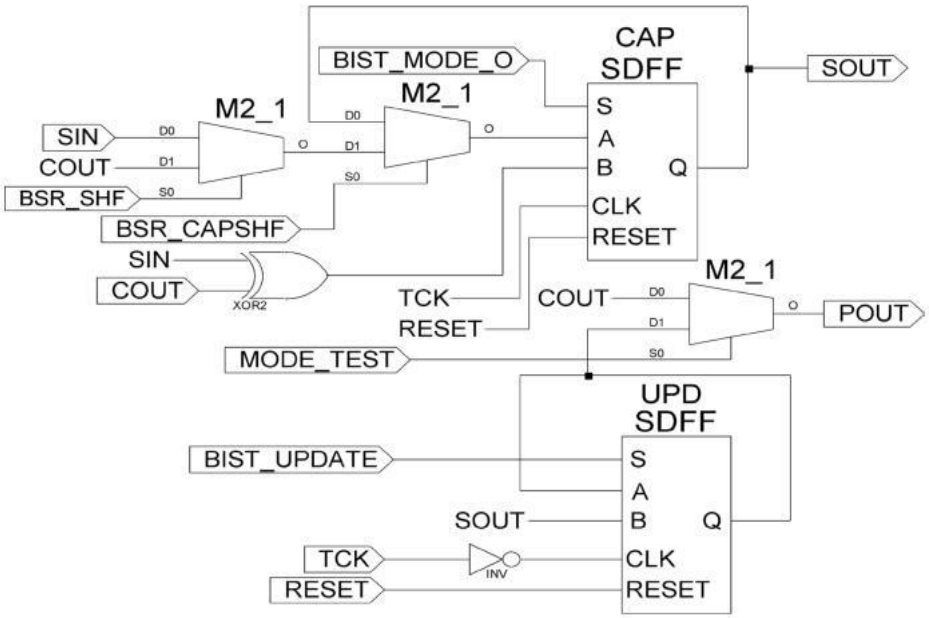

Fig.8 BIST-BSR Output cell

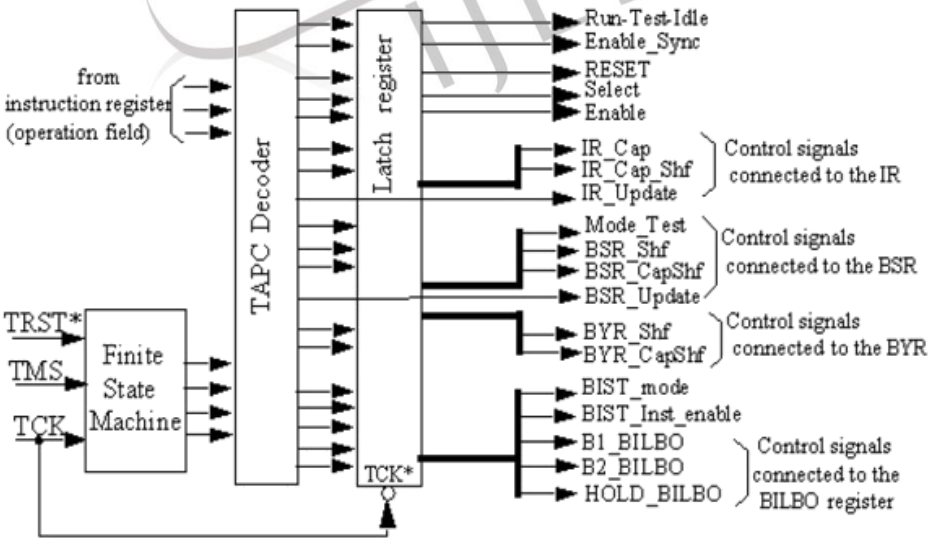

Fig.9 Block diagram of TAP Controller

In this paper, the authors have proposed a new pseudo-exhaustive BIST Boundary Scan architecture for Digital IC in industrial applications. They have explained the basic BS architecture with JTAG port [TDI, TD0, TRST, TCLK, TAP], basic concepts of

pseudo exhaustive BIST with illustrations of serial & parallel BIST. They have proposed a method to convert conventional scan FF into BILBO [Built in Logic Block Observer] & using this BILBO cells PET BIST architecture/test methodology is proposed. The authors have also proposed architecture of BIST-BSR & TAP controller with tabulated detailed explanation of each instruction. Also they have in detail explained test sequences of BIST BS & tabulated TAP controller [TAPC] decoder outputs like Shift DR, Capture DR, Update DR, Exit DR for BS process [EXTEST, INTEST, BYPASS] with FSM diagram. The authors have proposed single clock test [test clock] approach for all BS cells & have provided simulation timing diagram of various clocks [Tclk, Chip-clk, CUT clk] under BIST mode 0 & 1 as a case study to prove the functionality of a BIST BSR the authors have designed & implemented 16 bit parallel pipelined multiplier with BIST BSR embedded into the architecture of multiplier. The inspiring aspect of this technical paper to my research is the BS & TAP controller explanations & architecture & single clock test approach which I have also incorporated in my BS design & test module.

NR5

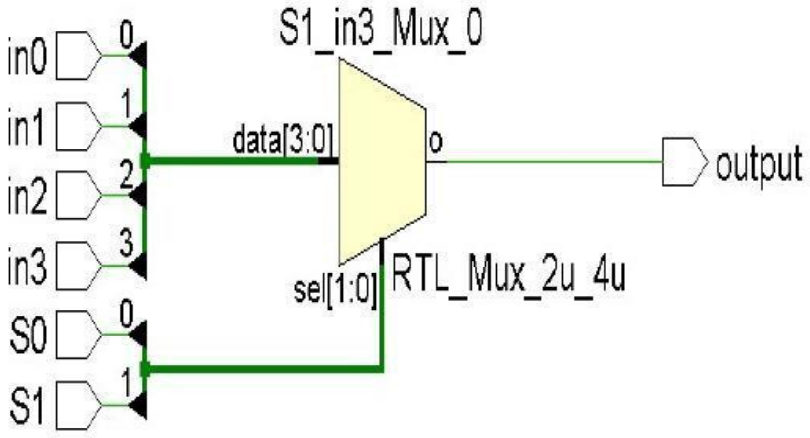

Fig.10 Design of Low power 4x1 Mux

In this paper, the authors have designed IO standard green multiplexer design & implementation of 40nm FPGA. The authors have done power estimation of standard IO logic families like SSTL, HSTL, LVPCI, LVCMOS etc. The authors have proposed mux to provide single output from a set of multiple inputs. They have done IO power estimation of 4x1 mux for IO power & leakage power & comparative IO power consumption of HSTL, SSTL logic families & sub families.

The authors have proposed mux design to multiplex multiple inputs to single output but not multiple logic families. They have also not done research experimentation on how to use mux to automatically provide corresponding pull up & pull downs for several logic families to be assigned any IOB of FPGA or IO of ASIC or ASSP. Also the authors have not spoken anything on usage of user constraint file or IO banks for assigning any IOB to any of the several logic families to provide IO standard interface.

This paper provides a basic mux design for power consumption of HSTL & SSTL logic families. But my paper provides for automatic enabling of internal pull ups & pull downs corresponding to logic family assigned in UCF for any of the pins & corresponding IOB i.e., my design & implementation realizes a programmable select IO with direct interface of all the IOs with any of the 40+ standard logic families.

NR6

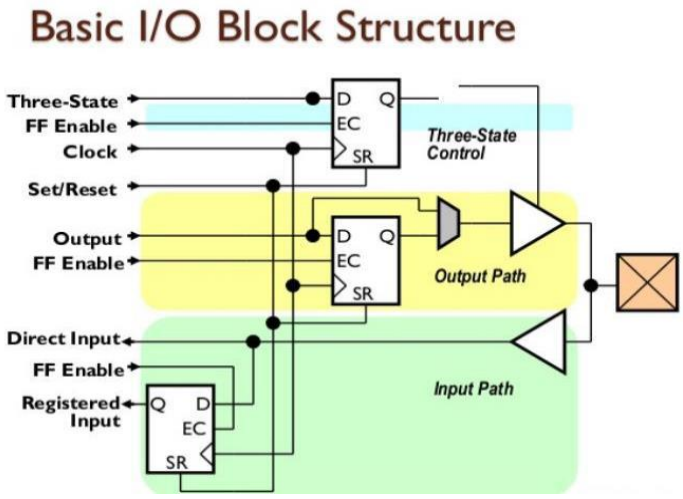

Fig.11 Basic I/O structure[6]

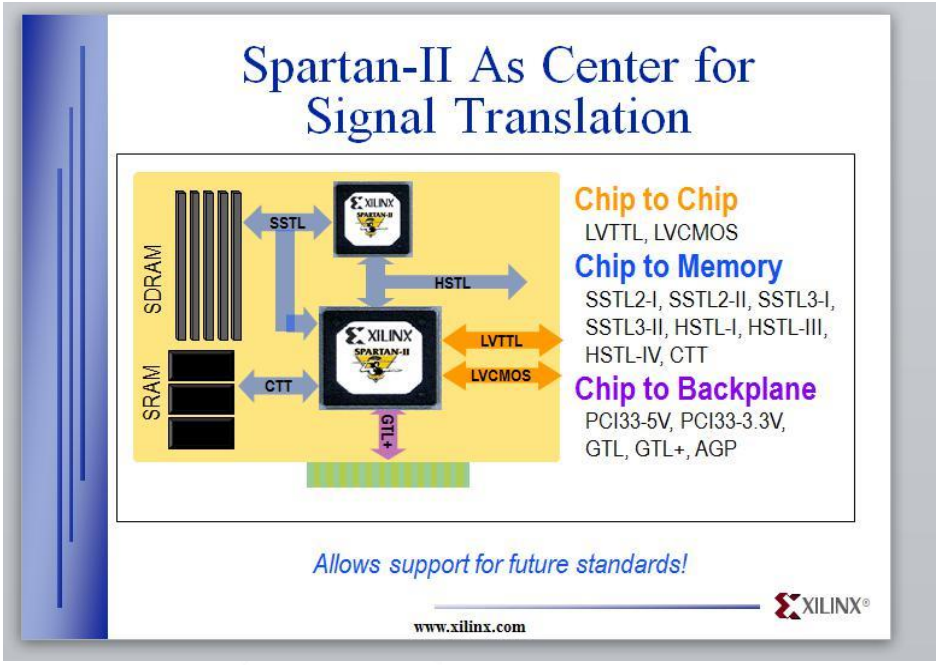

Fig.12 I/O Logic families[6]

In this document, Xilinx FPGA manufacturers have explained the basic IOB structure consisting of IO pad, IO buffers & tri-state output buffer, buffer flip-flops both at the input & output paths & a flip-flop to control tri-state. There is a Buffer-less or delay-less direct path for direct input & output & delay path or delay-less path can be selected using 2:1 mux. Some mux is also used to make clock positive or negative edge sensitive. Some delay latches are also there in the input path to provide signal synchronization of various signals arriving at different times at various IOB of same FPGA. There are also internal provision done at each IOB for slew rate control at output buffer i.e., any signal can be made to have FAST slew rate [dv/dt].

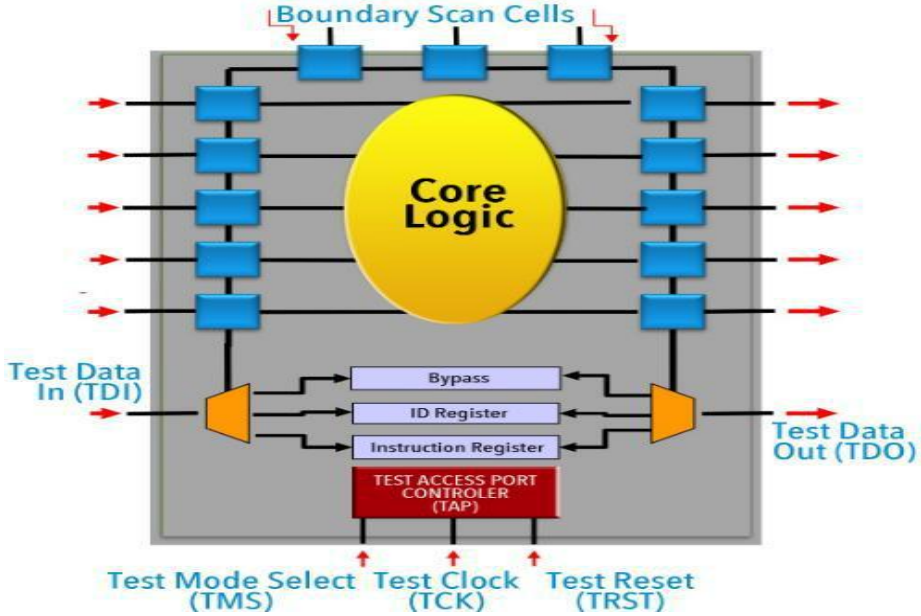

Fig.13 Boundary Scan I/O[6]

Also passive pull up & pull down facility is available for all IOBs of FPGA in order to interface FPGAs IOBs/pads to various other logic families like HSTL, SST SSTL, LVCMOS, LVTTL, RSDS, TMDS.

Programmable pull up & pull down is implemented on all IOBs of FPGA in order to interface any pin of FPGA to 40+ logic families [to comply voltage & protocol standards]. A group of IOB/pins of FPGA can be grouped into IO bank & all the above pins of such banks can be interfaced without any external pull up pull downs directly to any of the 40+standard logic families i.e., variable internal pull up pull downs will be automatically enabled according to the UCF specification of Bank interfacing logic families.

By providing such variable IOB facilities & direct interfacing with various logic families PCB design will be simplified & running high speed buses on a multilayered PCB will also get simplified by appropriate pin locking of FPGA.

The flip-flops in input & output path of IOB in FPGA will perform 2 functions:

1. Data buffering or delay synchronization of input or output signals or data.
2. They also act as BS flip-flops as a part of BS chain & will self test the FPGA using JTAG protocol during chips self test using BS chain.

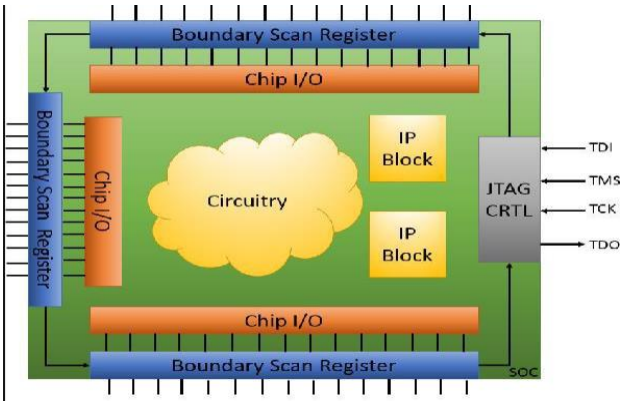

Fig.14 Boundary Scan I/O[6]

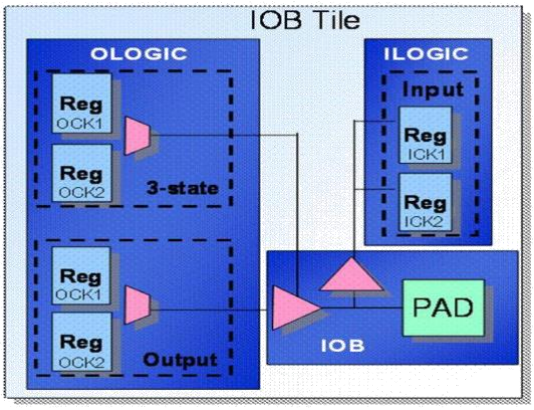

Fig.15 Spartan 6 IOB structure[6]

In Spartan 6 IOB tile architecture both input & output path there are 2 DDR registers & in the 3 state logic also 2 DDR registers are present. Separate clocks enable input & output, set & reset signals are common hence it becomes a compulsion to research on DDR based high speed IO. Spartan 2 IOB contains SDR flip-flops & Spartan 6 contains DDR flip-flops.
The Spartan 6 IO logic resources contain an IO logic block which in turn consists of IOSERDES block which acts as data Serializer [P to S] & De-Serializer [S to P], IO delay, SDR & DDR.

CONCLUSION

In line with my overall research objective of Design, implementation & testing of next generation multifunctional Universal IO, this technical survey paper provides a comprehensive explanation of all the major built-in functionalities of contemporary IO like Boundary Scan, BIST, JTAG, Bidirectional, Tri-state, Programmable select IO [interface with multiple logic families] etc.

REFERENCES

[1] Mr. Manjunath T.N, Sunil T.D, Dr. M. Z Kurian, Imran Rasheed, “Design and
[2] Implementation Boundary Scan Register for Trace and Debug of Controller” International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering Vol. 3, Issue 2, February 2014.
[3] Bashar Al-Khalifa “A Test Procedure for Boundary Scan Circuitry in PLDs and FPGAs”,The International Arab Journal of Information Technology, Vol. 7, No. 2, April 2010
[4] Zhao Yan,Wu Lihua, Han Xiaowei, Li Yan, Zhang Qianli, Chen Liang, Zhang Guoquan, Li Jianzhong, Yang Bo, Gao Jiantou, Wang Jian, Li Ming, Liu Guizhai, Zhang Feng, Guo Xufeng, Zhao Kai, Stanley L. Chen,Yu Fang, and Liu Zhongli, “An IO block array in a radiation-hardened SOI SRAM-based FPGA”, Journal of Semiconductors Vol. 33, No. 1 January 2012.
[5] Mohamed H. El-Mahlawy, Sherif Anas, Winston Waller, “New Built-In Self-Test Boundary Scan Architectures for Digital Integrated Circuits in Industrial Applications” International Journal of Computer Applications Volume 182- No.1, July 2018
[6] Bishwajeet Pandey, Rajendra Aaseri, Deepa Singh, Sweety, “IO standard green multiplexer design & implementation of 40nm FPGA” Research Gate,September 2013
[7] www.xilinx.com