Journal Homepage: - www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

Article DOI: 10.21474/IJAR01/21461
DOI URL: http://dx.doi.org/10.21474/IJAR01/21461

RESEARCH ARTICLE

8 TO 3 PRIORITY ENCODERS

Ponnala Lakshmi Prasanna
1. Assistant Professor, IoT Department, SR Gudlavalleru Engineering College Gudlavalleru.

Manuscript Info

Manuscript History
Received: 23 May 2025
Final Accepted: 25 June 2025
Published: July 2025

Key words:-
Fsm,hdl,verilog,ieee,ic,fpga

Abstract

The Hardware Description Language(HDL)is a computer language use d to describe the structure and behavior of digital circuits and systems. HDLs are very important in the designing and development of digital circuits in the form of integrated circuits (ICs) and field-programmable gate arrays(FPGAs).Verilog is a hardware description language (HDL) used to model and design digital circuits.It's widely used in the semicon ductor industry for designing and verifying digital systems,especially in the context of Field Programmable Gate Arrays(FPGAs)and Applicatio n-Specific Integrated Circuits (ASICs) They allow designers to model, simulate,and synthesize hardware designs,effectively creating a bluepri nt for the physical hardware.In this paper we presented 8 to 3 Priority E ncoders with and without priority which was implemented using Verilo g.Here,the Verilog modules for 8 to 3 priority encoders with and witho ut priorities were written and Test bench modules were developed and then the designs were simulated and synthesized using Xilinx 14.7.

Introduction:-

Hardware Description Language (HDL) designers can model, simulates, and synthesizes hardware designs, effectively creating a blueprint for the physical hardware. HDLs are used to create models of digital circuits, allowing designers to simulate their behavior using Verilog or VHDL before physical fabrication.

HDL code can be automatically converted into physical implementation (e.g., an IC) using specialized software tools like Xilinx, Vivado, Modelsim/Questasim, HDL Designer, HDL coder, HDL Verifier, Icarus Verilog which is a process called synthesis. HDLs support various levels of abstraction like Behavioral Level abstraction, Register Transfer Level and Gate-Level abstractions.

Behavioral level concentrates on functionality rather than internal structure of a design. RTL focuses on data flow between registers. Gate level focuses on circuit which is an interconnection of logic gates. HDLs include two IEEE standards VHDL and Verilog.Verilog enables designers to describe the connections and behavior of digital circuits, allowing for a structured and systematic approach to hardware design.

Verilog creates test benches that check the functionality and correctness of a digital design. Test benches include various possible input combinations which checks the desired output.

Corresponding Author:- Ponnala Lakshmi Prasanna
Address:- Assistant Professor, IoT Department, SR Gudlavalleru Engineering College Gudlavalleru.

An encoder is a digital circuit that converts one form of information into another form. For example an encoder encodes text "ABCDEFGH" into "DEFGHIJL". In this encoding technique letter 'A' has been encoded into letter 'D', 'B' has been encoded into 'E' and so on. Similarly, in digital electronics, an 8 to 3 priority encoder without and priority encodes "00000001" into "000" based on some rule. There are encoders that can encode binary values, speech signals and video signals as well.

Generally, an encoder has 2^n input lines and n-output lines. Only one of the 2^n input lines is activated at a time and the corresponding encoded code appears at the output. There are several types of encoders available such as Binary encoders, priority encoders and encoders for Number systems like Hexadecimal to binary encoders, Octal to binary encoders, keyboard encoders, decimal to BCD encoders etc.

A set of examples of 2^n to n Priority encoders includes 4 to 2 priority encoders, 8 to 3 priority encoders, 16 to 4 priority encoders and 32 to 5 prioprity encoders etc.

RTL schematic:-
RTL (Register Transfer Level) schematic provides block diagram representation of a digital circuit at an abstract level. It focuses on how data moves between registers and the logical operations performed on that data, rather than on the detailed implementation using individual logic gates.

Technology schematic:-
Technology schematics are schematics that uses technology-specific or target specific FPGA components like LUTs (Look Up Tables), I/O buffers, carry logic and other technology-specific components.

Device Utilization Summary:-
In the context of VHDL, device utilization summary is a report generated by synthesis and implementation tools (e.g., Xilinx, Intel Quartus Prime, Vivado) after processing a Verilog module for a target ASIC or FPGA. That is device utilization summary is a synthesis report generated after simulating Verilog Test bench module using Behavioural model. A number of devices can be selected to generate synthesis reports. In this paper device 7a100tcsg324-3 is selected to generate synthesis report. This summary provides a detailed breakdown of how the logic described in the Verilog code translates into physical resources available on the chosen device.

The device utilization summary includes Slice Logic Utilization, Slice Logic Distribution, IO Utilization, Specific Feature Utilization, Partition Resource Summary, I/O Pin Utilization, and Clock Resources. The device utilization summary is required for Cost Estimation, Design Feasibility, Resource Management and Performance Analysis.

Timing Summary:-
The Timing Summary is a type of synthesis report that provides details about Minimum period, Minimum input arrival time before clock, Maximum output required time after clock, Maximum combinational path delay.
In Verilog, timing summaries gives overview of the timing behavior of a Verilog design generated during synthesis or static timing analysis (STA). They illustrate how signals propagate within the circuit, potential timing violations like hold time and setup issues. They summarize the timing characteristics of paths within the Verilog module.

Priority Encoder:-
A priority encoder is a combinational logic circuit with a priority function that encodes multiple input lines into a multiple output lines. It works based on the rule that the input which is active high, then the corresponding binary value will be assigned to the multiple output lines. This is different from a binary encoder. Hence it is named as priority encoder.

8 to 3 Priority Encoder without priority:-
A priority encoder 8 to 3 converts eight binary inputs into a 3-bit binary output without priority

Block Diagram of 8 to 3 Encoder without priority:-

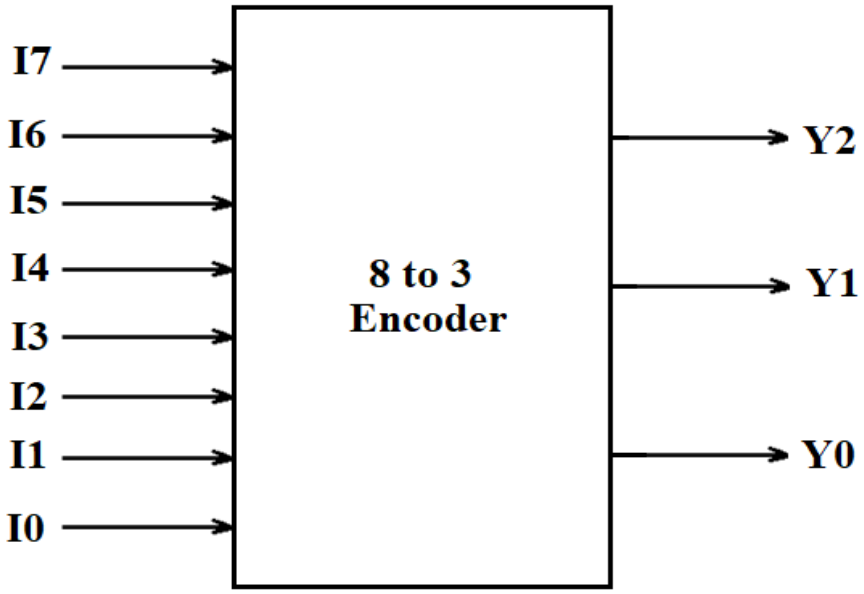


Fig: 8 to 3 Encoder without priority

The Block Diagram of 8 to 3 Encoder without priority consists of 8 inputs from $I_7 - I_o$ and outputs from $Y_2 - Y_o$.

Working of 8 to 3 Encoder without priority:-

The working of 8 to 3 Encoder without priority is as follows,
When the input I_o is active high($I_o = 1$) and all other inputs are active low($I_1I_2I_3I_4I_5I_6I_7 = 0000000$) then the output ($Y_2Y_1Y_0$) becomes 000.
When the input I_1 is active high($I_1 = 1$) and all other inputs are active low($I_0I_2I_3I_4I_5I_6I_7 = 0000000$) then the output ($Y_2Y_1Y_0$) becomes 001.
When the input I_2 is active high($I_2 = 1$) and all other inputs are active low($I_0I_1I_3I_4I_5I_6I_7 = 0000000$) then the output ($Y_2Y_1Y_0$) becomes 010.
When the input I_3 is active high($I_3 = 1$) and all other inputs are active low($I_0I_1I_2I_4I_5I_6I_7 = 0000000$) then the output ($Y_2Y_1Y_0$) becomes 011.
When the input I_4 is active high($I_4 = 1$) and all other inputs are active low($I_0I_1I_2I_3I_5I_6I_7 = 0000000$) then the output ($Y_2Y_1Y_0$) becomes 100.
When the input I_5 is active high($I_5 = 1$) and all other inputs are active low($I_0I_1I_2I_3I_4I_6I_7 = 0000000$) then the output ($Y_2Y_1Y_0$) becomes 101.
When the input I_6 is active high($I_6 = 1$) and all other inputs are active low($I_0I_1I_2I_3I_4I_5I_7 = 0000000$) then the output ($Y_2Y_1Y_0$) becomes 110,
Similarly, when the input I_7 is active high($I_7 = 1$) and all other inputs are active low($I_0I_1I_2I_3I_4I_5I_6 = 0000000$) then the output ($Y_2Y_1Y_0$) becomes 111.

Truth Table : 8 to 3 Encoder without priority:-

I7	I6	I5	I4	I3	I2	I1	I0	Y2	Y1	Y0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

Logic Diagram of 8 to 3 Encoder without priority:-

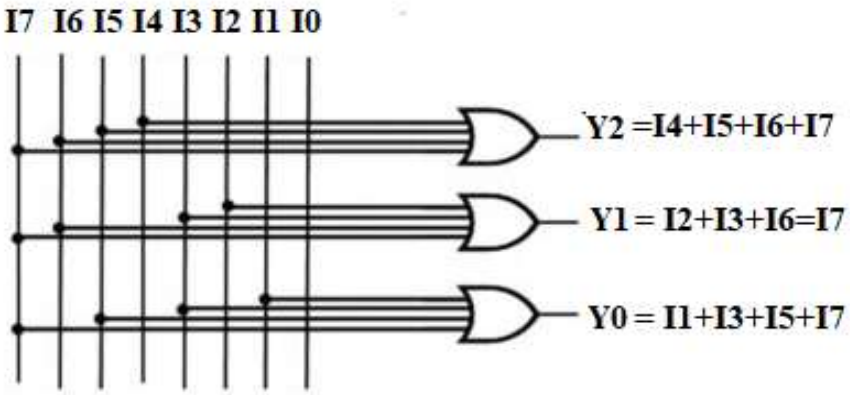
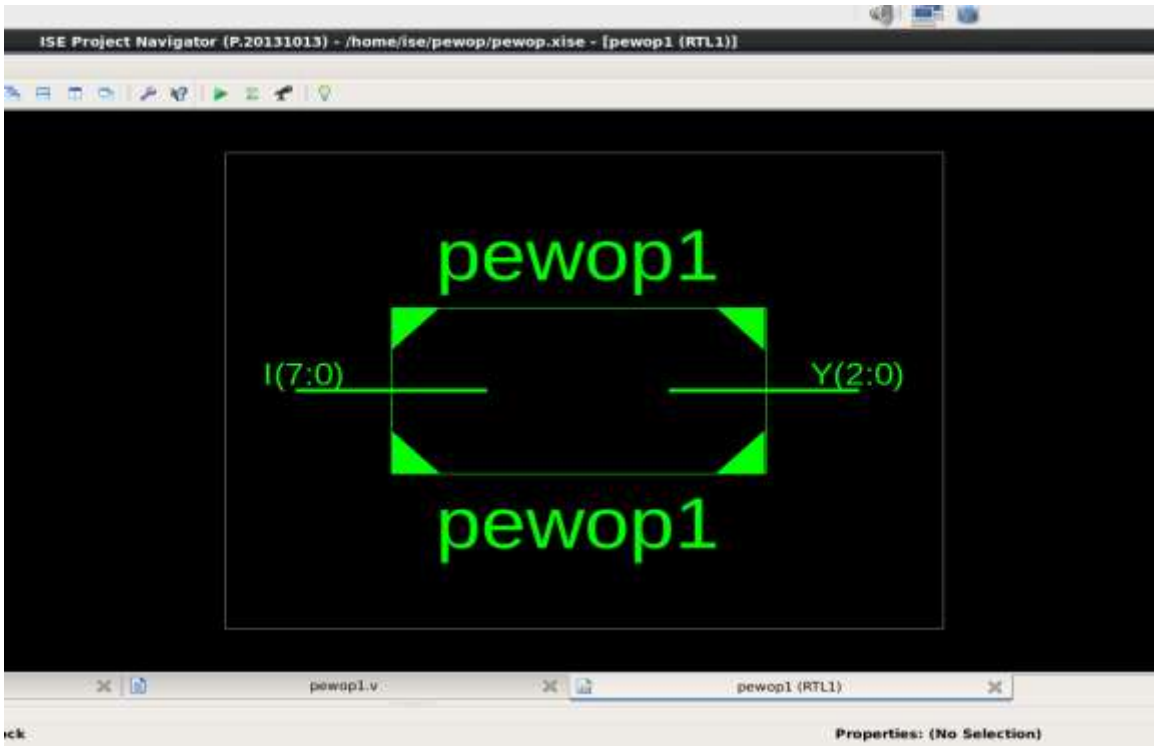
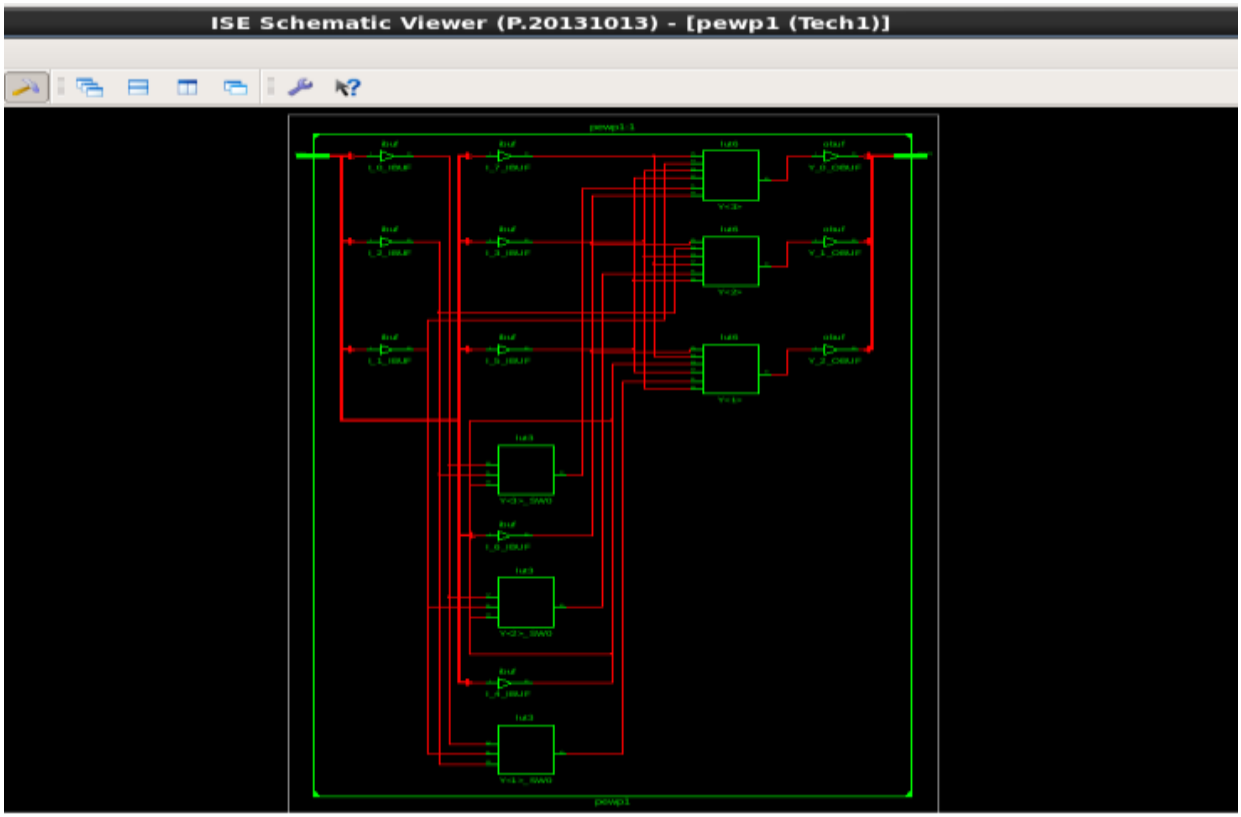


Fig: Logic Diagram of 8 to 3 Encoder without priority

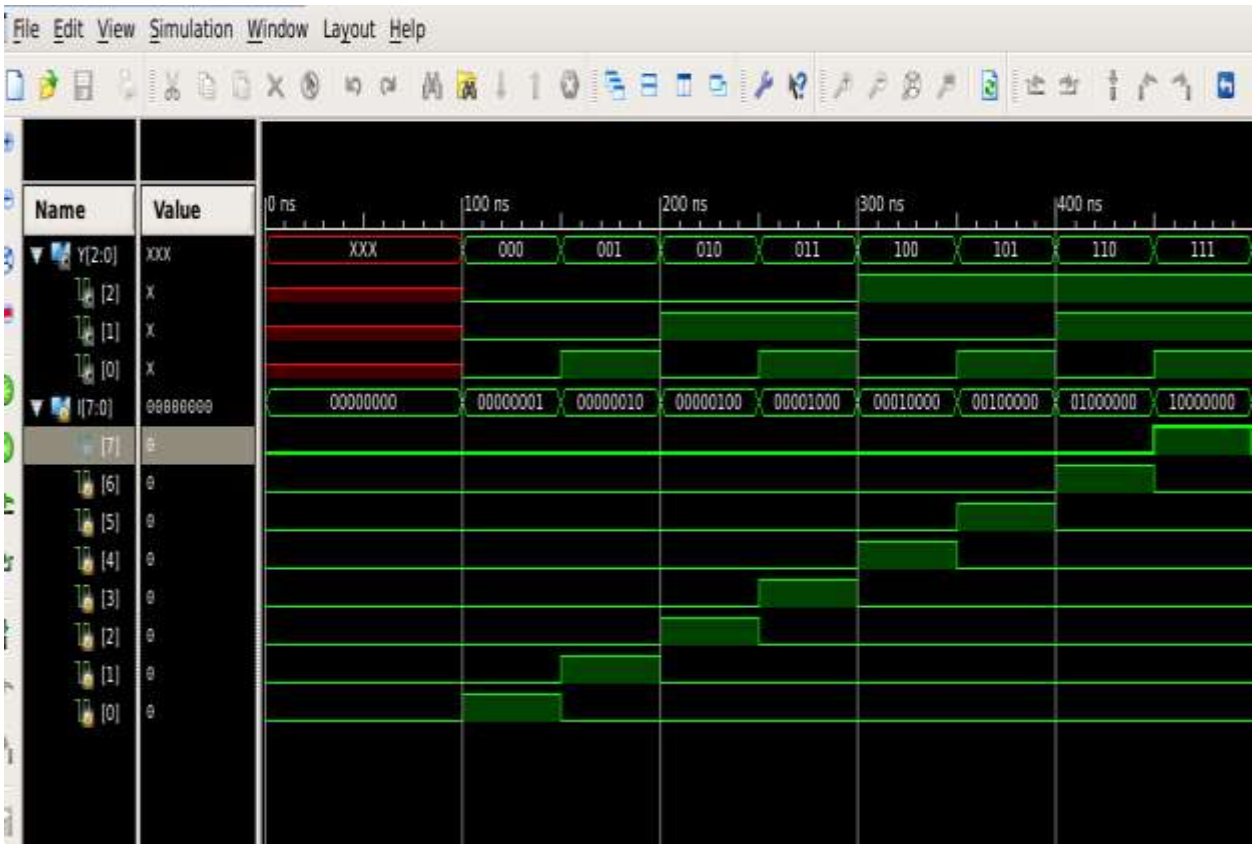
RTL schematic:- 8 to 3 Encoder without priority



Technology schematic:- 8 to 3 Encoder without priority



Output Waveform: - 8 to 3 Encoder without priority



Timing Summary:- 8 to 3 Encoder withput priority

Speed Grade: -3

Minimum period	No path found
Minimum input arrival time before clock	No path found
Maximum output required time after clock	No path found
Maximum combinational path delay	1.679ns

Device utilization summary:- 8 to 3 Encoder withput priority
Selected Device: 7a100tcsg324-3

Slice Logic Utilization:	
Number of Slice LUTs:	6 out of 63400 0%
Number used as Logic:	6 out of 63400 0%
Slice Logic Distribution:	

Number of LUT Flip Flop pairs used	6
Number with an unused Flip Flop	6 out of 6 100%
Number with an unused LUT	0 out of 6 0%
Number of fully used LUT-FF pairs	0 out of 6 0%
Number of unique control sets	0
IO Utilization:	
Number of IOs	11
Number of bonded IOBs	11 out of 210 5%

Primitive and Black Box Usage:- 8 to 3 Encoder withput priority

#BELS	6
#LUT3	3
#LUT6	3
# IO Buffers	11
# IBUF	8

8 to 3 Priority Encoder with priority:-

A priority encoder 8 to 3 converts eight binary inputs into a 3-bit binary output with priority.

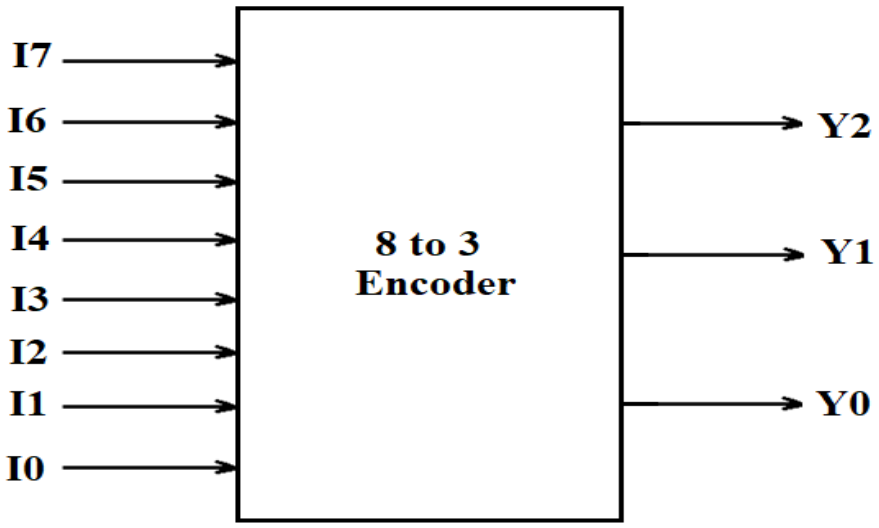


Fig: 8 to 3 Encoder with priority

Working of 8 to 3 Encoder with priority:-

The working of 8 to 3 Encoder with priority is as follows,

When the enble input is active low(en = 0):

When all the other inputs are active low($I_oI\ I_1I_2I_3I_4I_5I_6I_7$ = 00000000) then the output ($Y_2Y_1Y_0$) becomes ZZZ (High impedance state).

When the enble input is active high(en = 1):

When all the other inputs are active low($I_oI\ I_1I_2I_3I_4I_5I_6I_7$ = 00000000) then the output ($Y_2Y_1Y_0$) becomes XXX.

When the input I_o is active high(I_o = 1) and all other inputs are active low($I_1I_2I_3I_4I_5I_6I_7$ = 0000000) then the output ($Y_2Y_1Y_0$) becomes 000.

When the input I_1 is active high(I_1 = 1) and all other inputs are active low($I_0I_2I_3I_4I_5I_6I_7$ = X000000) then the output ($Y_2Y_1Y_0$) becomes 001.

When the input I_2 is active high(I_2 = 1) and all other inputs are active low($I_0I_1I_3I_4I_5I_6I_7$ = 0X00000) then the output ($Y_2Y_1Y_0$) becomes 010.

When the input I_3 is active high(I_3 = 1) and all other inputs are active low($I_0I_1I_2I_4I_5I_6I_7$ = 00X0000) then the output ($Y_2Y_1Y_0$) becomes 011.

When the input I_4 is active high(I_4 = 1) and all other inputs are active low($I_0I_1I_2I_3I_5I_6I_7$ = 000X000) then the output ($Y_2Y_1Y_0$) becomes 100.

When the input I_5 is active high(I_5 = 1) and all other inputs are active low($I_0I_1I_2I_3I_4I_6I_7$ = 0000X00) then the output ($Y_2Y_1Y_0$) becomes 101.

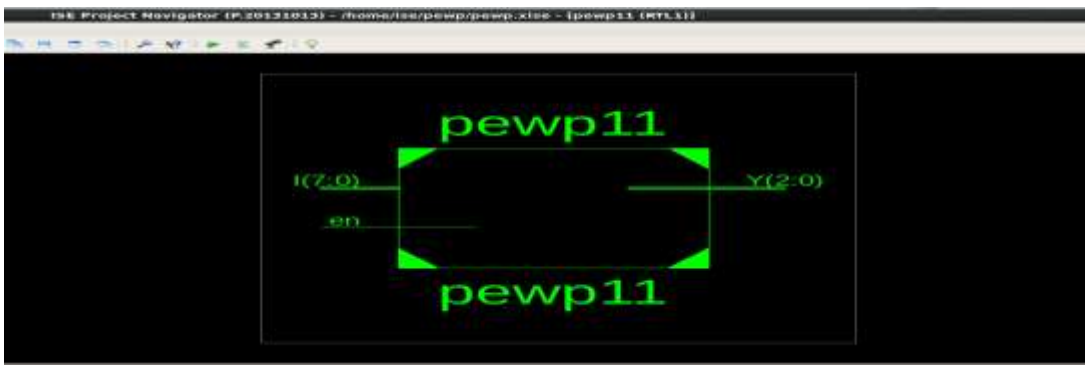
When the input I_6 is active high(I_6 = 1) and all other inputs are active low($I_0I_1I_2I_3I_4I_5I_7$ = 00000X0) then the output ($Y_2Y_1Y_0$) becomes 110,

Similarly, when the input I_7 is active high(I_7 = 1) and all other inputs are active low($I_0I_1I_2I_3I_4I_5I_6$ = 000000X) then the output ($Y_2Y_1Y_0$) becomes 111.

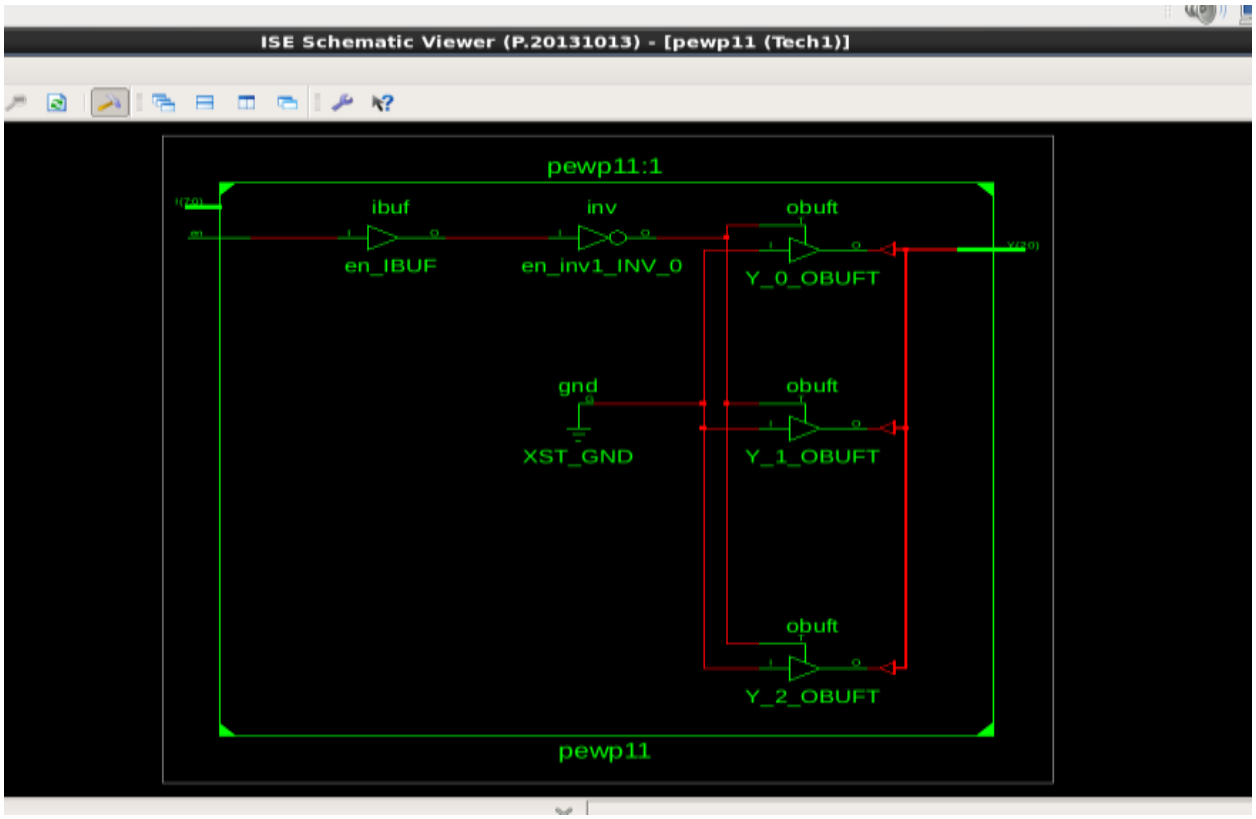
Truth Table :- 8 to 3 Encoder with priority

en	I7	I6	I5	I4	I3	I2	I1	I0	Y2	Y1	Y0
0	0	0	0	0	0	0	0	0	Z	Z	Z
1	0	0	0	0	0	0	0	0	X	X	X
1	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	1	X	0	0	1
1	0	0	0	0	0	1	X	X	0	1	0
1	0	0	0	0	1	X	X	X	0	1	1
1	0	0	0	1	X	X	X	X	1	0	0
1	0	0	1	X	X	X	X	X	1	0	1
1	0	1	X	X	X	X	X	X	1	1	0
1	1	X	X	X	X	X	X	X	1	1	1

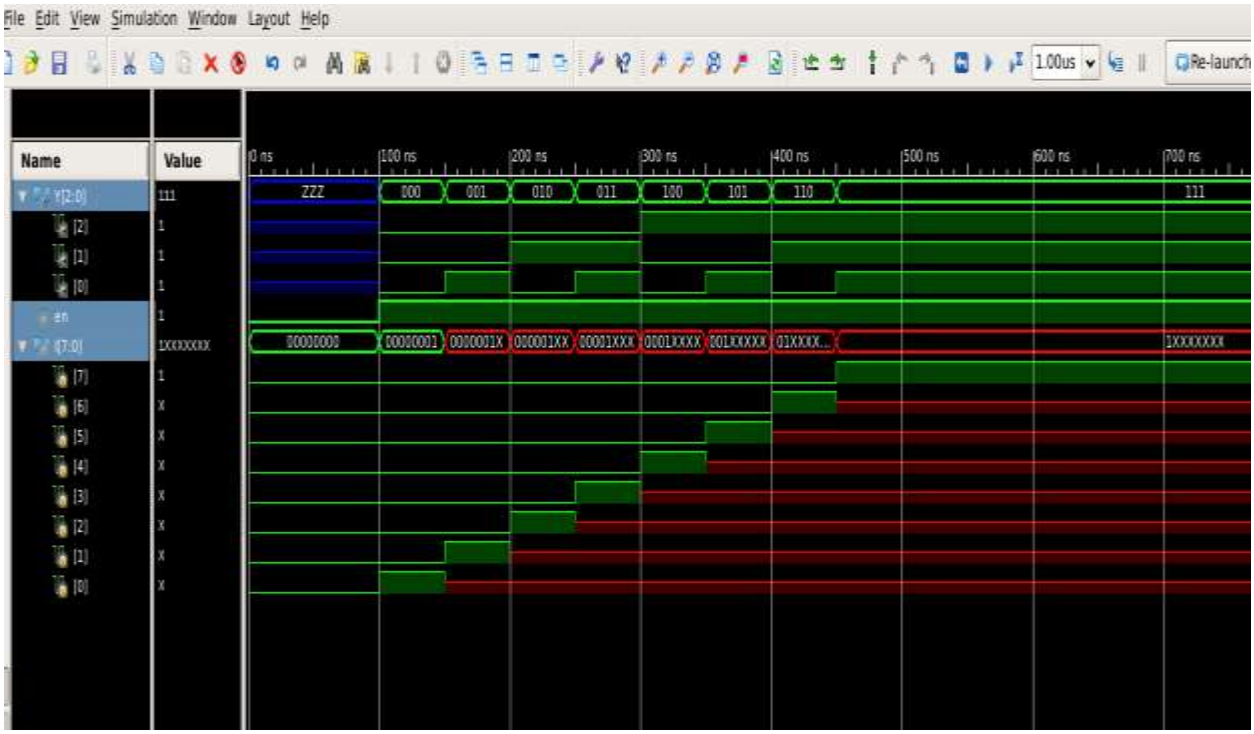
RTL schematic: - 8 to 3 Encoder with priority



Technology schematic: - 8 to 3 Encoder with priority



Output Waveform: - 8 to 3 Encoder with priority



Timing Summary: - 8 to 3 Encoder with priority

Speed Grade: -3

Minimum period	No path found
Minimum input arrival time before clock	No path found
Maximum output required time after clock	No path found
Maximum combinational path delay	0.682ns

Device utilization summary:- 8 to 3 Encoder with priority

Selected Device: 7a100tcsg324-3

Slice Logic Utilization:	
Number of Slice LUTs:	1 out of 63400 0%
Number used as Logic:	1 out of 63400 0%
Slice Logic Distribution:	
Number of LUT Flip Flop pairs used	1
Number with an unused Flip Flop	1 out of 1 100%
Number with an unused LUT	0 out of 1 0%
Number of fully used LUT-FF pairs	0 out of 1 0%
Number of unique control sets	0
IO Utilization:	
Number of IOs	12
Number of bonded IOBs	4 out of 210 1%

Primitive and Black Box Usage:- 8 to 3 Encoder with priority

#BELS	2
#GND	1
#INV	1
# IO Buffers	4
# IBUF	1
# OBUFT	3

Conclusion:-

In this paper priority encoders (8 to 3) with and without priorities was implemented and simulated and synthesized using Verilog. RTL and Technology schematics were obtained. Device Utilization Summary, Timing Summary and Primitive and Black Box Usage were summarized.

References:-

[1] J. G. Delgado-Frias, J. Nyathi, "A high-performance encoder with priority lookahead," IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 47, no. 9, pp. 1390–1292, 2000.
[2] A. Morgenshtein, A. Fish, and I. A. Wagner, "Gate-Diffusion Input (GDI): A Power Efficient Method for Digital Combinatorial Circuits ", IEEE Trans. VLSI Systems, Vol. 10, No. 5, pp- 566 to 581, Oct. 2002.
[3] S. A. Hafeez and S. Harb, "A VLSI high performance priority Encoder using standard CMOS library ", IEEE transactions on Circuits and System II, Vol. 53, pp- 597 to 601, No. 8, August 2006.
[4] Vishal Khandelwal and Ankur Srivastava, "Leakage Control Through Fine-Grained Placement and Sizing of Sleep Transistors ", IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, Vol. 26, No. 7, JULY 2007.
[5] H. F. Dadgour and K. Banerjee "A novel variation-tolerant keeper architecture for high-performance low-power wide fan-in dynamic OR gates", IEEE Trans. Very Large Intrgr. (VLSI) Syst., vol. 18, no. 11, pp. 1567–1577 2010.
[6] S. W. Huang, and Y. J. Chang, "A full parallel priority encoder design used in comparator," in Proc. of the 53rd IEEE International Midwest Symposium on Circuits and Systems, Seattle, USA, pp. 877–880, 2010.
[7] N. Mehta and B. Amrutur, "Dynamic supply and threshold voltage scaling for CMOS digital circuits using in-situ power monitor ", IEEE Trans. Very Large Scale Integr. (VLSI) Syst., Vol. 20, No. 5, pp- 892 to 901, May 2012.
[8] Amritesh Ojha, Rajesh Mehra, "low Power Layout design of Priority encoder using 65nm technology ", International Journal of Engineering Trends and Technology (IJETT) -Volume 23 Number 9 - May 2015.
[9] Ojha, Amritesh & Mehra, Dr. Rajesh. (2015). Low Power Layout Design of Priority Encoder Using 65nm Technology. International Journal of Engineering Trends and Technology. 23. 450–453. 10.14445/22315381/IJETT-V23P285.
[10] L. Dileshwar Rao, S. Dixit, K. Pachkor, and M. Aarthy, "FinFET based adiabatic logic design for low power applications," 2017 International conference on Microelectronic Devices, Circuits and Systems (ICMDCS), Vellore, India, 2017, pp. 1–6, doi: 10.1109/ICMDCS.2017.8211535.
[11] E. Norige, A. X. Liu, E. Torng, "A ternary unification framework for optimizing TCAM-based packet classification systems," IEEE/ACM Transactions on Networking, vol. 26, no. 2, pp. 657–670, 2018.
[12] C. -L. Hsieh, N. Weng and W. Wei, "Scalable Many-Field Packet Classification for Traffic Steering in SDN Switches," IEEE Transactions on Network and Service Management, vol. 16, no. 1, pp. 348–361, 2019.
[13] R. R. Vallabhuni, A. Karthik, C. V. Sai Kumar, B. Varun, P. Veerendra, and S. Nayak, "Comparative Analysis of8-Bit Manchester Carry Chain Adder Using FinFET at 18nm Technology," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Tho othukudi, India, 2020, pp. 1579–1583, doi: 10.1109/ICISS49785.2020.9316061.

[14] R. R. Vallabhuni, D. V. L. Sravya, M. S. Shalini, and G. U. Maheshwararao, "Design of Comparator using 18nm FinFET Technology for Analog to Digital Converters," 2020 7th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, 2020, pp. 1–6, doi: 10.1109/ICSSS49621.2020.9202164.
[15] M. Irfan, Z. Ullah, M.H. Chowdhury and R.C. Cheung, "RPE-TCAM: Reconfigurable power-efficient ternary content-addressable memory on FPGAs," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 28, no. 8, pp. 1925–1929, 2020.
[16] S. Asadrooz, M. Nassiri, M. Abbasi, and H. Abdoli, "A Traffic-Aware Packet Classification Method to Reduce Memory Accesses," Iranian Journal of Electrical and Computer Engineering, vol. 18, no. 3, pp. 185–196, 2021.
[17] J. Yang, H. Lee, J.H. Jeong, T.h. Kim, S.H. Lee, T. Song, A Practical Implementation of the Ternary Logic Using Memristors and MOSFETs, in: 2021 IEEE 51st International Symposium on Multiple-Valued Logic, ISMVL, 2021, pp. 183–188.
[18] A. Goenka, S. Mitra, N. Das, Design of Balanced Ternary Encoder and Decoder, in: 2022 6th International Conference on Computing Methodologies and Communication, ICCMC, 2022, pp. 482–490.
[19] V.S.M. Srinivasavarma, S. Pydi Sk. N. Mahammad, "Hardware-based multi-match packet classification in NIDS: an overview and novel extensions for improving the energy efficiency of TCAM -based classifiers," The Journal of Supercomputing, vol. 78, no. 11, pp. 13086–13121, 2022.
[20] M. Dhayalakumar, Sk. N. Mahammad, "TeRa: Ternary and Range based packet classification engine," Integration, vol. 96, pages. 102153, 2024.