**Journal of Mines, Metals and Fuels,** 71(4): 520-522; 2023. DOI: 10.18311/jmmf/2023/33929

Print ISSN : 0022-2755

Journal of Mines, Metals and Fuels

Contents available at: www.informaticsjournals.com/index.php/jmmf

# Design, Synthesis, and Testbench Verification of High Order Decoder Circuits using VerilogHDL

**Abdulkhaliq Saeed[1], Amjed Ali[1], Abdulaziz Saeed[1], Ibrahim Hassan[1], Neeraj Kumar Shukla[1], M. Abdul Muqeet[1] and Shilpi Birla[2*]**

*[1]Electrical Engineering Department, King Khalid University Abha, Kingdom of Saudi Arabia.*
*[2]Department of ECE, Manipal University Jaipur, Jaipur, India. [*]E-mail: shilpi.birla@jaipur.manipal.edu*

## Abstract

*The decoder is an integral part of modern memory, processors. In this work, a 5 to 32 and 6 to 64 decoders have been designed and characterized using low-order decoders for the state-of-the-art digital systems. The Design and Synthesis work have been executed at the Register Transfer Level (RTL) and the Testbench based Verification has been performed to ensure the functional correctness of the designs with timing constraints. RTL Coding is done at VerilogHDL Std. IEEE 1364-2001 and IEEE 1364-2005. It is observed that the 5 to 32 and 6 to 64 order decoders utilize the highest 15% and 30% bonded IOBs where the 2 to 4 decode 2% only. It is found that the 6 to 64 decoders have 5% higher delay to 5 to 32 decoders whereas the 5 to 32 decoder has 5.5% more logic delay and 9.1% less route delay as compared to the 6 to 64 decoders.*

***Keywords:*** *Decoders, FPGA, IOB, Logic Block, Logic Delay, Rout Delay, VerilogHDL*

## 1.0 Introduction

In modern day's digital systems, the encoder and decoder systems are emerging as a new challenge when it comes to overall system's efficiency. The popular system which uses the high speed and low-power decoder systems includes, speed synchronization of multiple motors in industries, war field flying robot with a night vision flying camera, robotic vehicle with the metal detector, radio frequency-based home automation system, automatic health monitoring systems, etc. And at the microprocessors systems' level it includes memory systems to select different memory banks, input/output systems for selecting different peripheral and on-board devices, etc. The way a computer system decodes the addresses on the address bus to select peripheral devices, microprocessor instruction decoding (enabling different functional units), the instruction decoder logic (converts the op-code bits into settings for all the internal control lines), memory chips (enabling different rows of memory depending on address), etc. In[1,2], an efficient and robust 2 to 4 decoder design using three-input majority voters have been presented which is implemented in the Quantum-dot Cellular Automata (QCA). The design has been optimized by considerably reducing the number of cells counts and QCA wire crossings. The authors in[3] have proposed circuit techniques to reduce leakage power dissipation in memory cells (and memory array) by considering the input circuitry of memory architecture i.e., row and column decoder. In digital architectures, a decoder decodes the coded data primarily used in the memory circuits to access the data stored in the memory. In a processors-based system a decoder circuit is used for the selection of different memory banks, peripheral devices, etc. connected to the processor by decoding its address where the instruction decoding logic converts the op-code data for internal control logics by enabling different system functional units. In memory circuits, different array segments are enabled/disabled or accessed by the decoder logics.

In a digital circuit, for an n-bits binary code, a decoder

[*]*Author for correspondence*

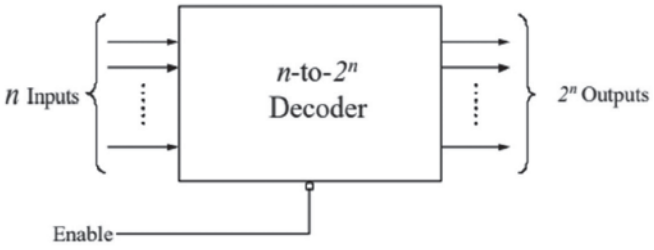


Figure 1. An n to $2^n$ Decoder with Enable Input

offers $2^n$ possible output codes. Figure 1, a typical decoder with n-inputs and $2^n$ outputs with enable input is depicted. Here, each of the $2^n$ outputs are corresponds to one of the possible n inputs and their combinations. The enable input performs no logical operations but is only responsible for making the decoder circuit active (Enable=1) inactive (Enable = 0)[4,5].

Various decoder circuit types are reported in the literature as 2 to 4, 3 to 8, 4 to 16, etc. with and without enable inputs for different requirements and applications. As an example, a 2 to 4 decoder has 2 input lines and 4 output lines. A 2 to 4 line decoder with enable input is shown in Figure 2. Figure 2, the decoder has two inputs $A_0$, $A_1$, four outputs $Y_0$, $Y_1$, $Y_2$, $Y_3$ and Enable signal "EN".

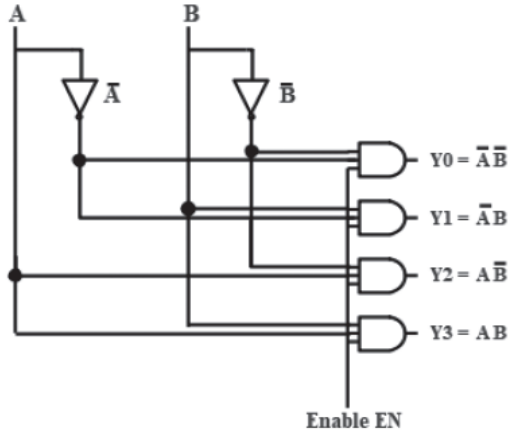


Figure 2: A 2 to 4 Line Decoder with Enable Input

In a 2 to 4 binary decoder, two inputs are decoded into four outputs. At a time, only one outputs $Y_0$,$Y_1$,$Y_2$, or $Y_3$ is at active high state for an input combination of $A_0A_1$ while the other outputs are maintained at active low state. A functional truth table for 2 to 4 decoder is presented in Table 1.

Here, Logic "0" and "1" are active low and active high state states, respectively. Logic "X" is a don't care state, i.e., any input combination of $A_0A_1$ will give output logic low when EN=0. The operation shown in the Table 1 is as, when EN=1, if $A_0A_1$=00, the Output Y0 shall be decoded. And when

**Table 1: Functional Truth Table of 2to4 Decoder with Enable input**

| Inputs | | | Outputs | | | |
|---|---|---|---|---|---|---|
| EN | $A_0$ | $A_1$ | $Y_3$ | $Y_2$ | $Y_1$ | $Y_0$ |
| 0 | | | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 |

EN=1, and $A_0A_1$ is 01,10, or 11, then outputs $Y_1$, $Y_2$, or $Y_3$ shall be decoded. This, 2 to 4 decoder is a popular choice for the design and development of higher order decoders as 3 to 8, 4 to 16, etc. using hierarchical level design approach and modelling. In the similar line, the decoders 3 to 8, 4 to 16, etc. can be designed and analyzed for their functional performances and characterization. In this work, the high order decoder circuits 5 to 32 and 6 to 64 decoder have been designed using these low order decoder circuits and their functional performance has been analyzed and verified using RTL Design and Verification Methodologies.

## 2.0 Integration of Lower Order Decoders

The decoders integration by instantiation (instantiation by name) approach is used for the design of a high order decoder using available lower order decoders for process simplicity. This hierarchical modelling approach offers an easy analysis and verification of decoder blocks in the design by ensuring a faster time to market and product tape-out. As an example, the 5 to 32 decoder has been implemented by using four 3 to 8 and one 2 to 4 decoders. The functional performance has been ensured by VerilogHDL Testbench Verification methodology[6,7]. The electrical level performance evaluation has been carried at the Digilent Spartan-3e Xilinx Starter Kit FPGA Development Board.

## 3.0 Results

The design simulation, synthesis has been done to estimate the device utilization using Spartan3AN FPGA. The chart shows the device summary for various decoders, Figure 3.

An increasing trend has been observed towards higher order decoders because of the high density of logic blocks in the design.

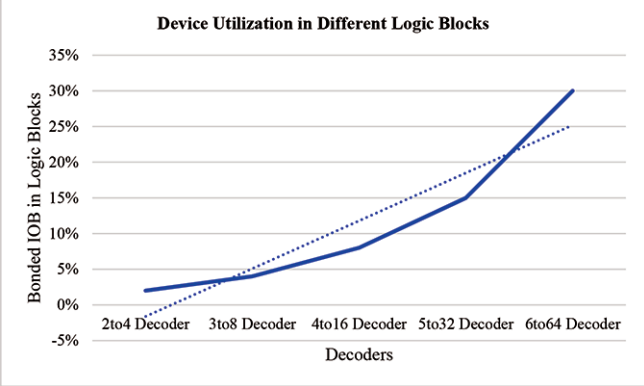


Figure 3: Device Utilization of various Decoders Circuits

## A. Delay and Performance Analysis

The performance analysis has been carried out by estimating the delay offered by various decoders circuits. The logic delay has been observed almost equal for various decoders as being a parallel architecture.

This design and analysis have been performed using VerilogHDL constructs and dataflow and behavioural modelling techniques. The designs have been verified for their functional correctness using testbench methodology. Thereafter, a FPGA implementation done to evaluate its electrical performance and power analysis. The synthesis operation provides information related to hardware utilization, e.g., LUTs, and IOBs.

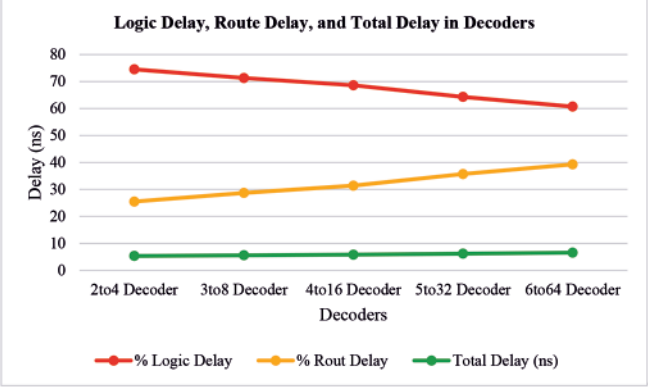


Figure 4: Logic Delay and Rout Delay in Various Decoders

# 4.0 Conclusions

In this work, the VerilogHDL Coding has been performed for various decoder architectures suing dataflow modelling approach at Xilinx platform and Synthesis is being done for hardware utilization, LUT, IOB. The FPGA implemented is underway to test the electrical performance, e.g., power and speed analysis. It is found in the synthesis that the 2 to 4 decoder, 3 to 8 decoder, 4 to 16 decoder, 5 to 32 decoder and 6 to 64 decoders have number IOBs (Input/Output Block to implement I/O functions) 2%, 4%, 8%, 15% and 30%. Here, the 6 to 64 decoders are occupying 30% area whereas 2 to 4 decoder has lowest 2%. It is also seen in the synthesis for 2 to 4 decoder total delay 5.362ns (3.996ns logic, 1.366ns route), (74.5% logic, 25.5% route), 3 to 8 decoder total delay 5.607ns (3.998ns logic, 1.609ns route), (71.3% logic, 28.7% route), 4 to 16 decoder total delay 5.827ns (3.996ns logic, 1.831ns route), (68.6% logic, 31.4% route), 5 to 32 decoder total delay 6.211ns (3.996ns logic, 2.215ns route), (64.3% logic, 35.7% route) and 6 to 64 decoder total delay 6.579ns (3.996ns logic, 2.583ns route), (60.7% logic, 39.3% route). Further, based on the applications and by improving the HDL synthesizable constructs, the area efficiency (lowering the IOBs) and performance (speed) can be further in future designs.

# 5.0 References

1. Sherizadeh, R., & Navimipour, N. J. (2018): Designing a 2-to-4 decoder on nanoscale based on quantum-dot cellular automata for energy dissipation improving. *Optik*, 158, 477-489.
2. Sherizadeh, R., & Navimipour, N. J. (2018): Designing a 2-to-4 decoder on nanoscale based on quantum-dot cellular automata for energy dissipation improving. *Optik*, 158, 477-489.
3. Kumar, M., & Sasamal, T. N. (2017): An Optimal design of 2-to-4 Decoder circuit in coplanar Quantum-dot cellular automata. Energy Procedia, 117, 450-457.
4. Gupta, N., Parihar, P., & Neema, V. (2018): Application of source biasing technique for energy efficient DECODER circuit design: memory array application. *Journal of Semiconductors*, 39(4), 045001.
5. Mano, M. M. (2017). Digital logic and computer design. Pearson Education India.
6. Smith, D. J. (1998): HDL Chip Design: A practical guide for designing, synthesizing and simulating ASICs and FPGAs using VHDL or Verilog. Doone publications.
7. Palnitkar, S. (2003): Verilog HDL: a guide to digital design and synthesis (Vol. 1). Prentice Hall Professional.