Journal of Research in Electrical Circuits and Systems
Volume 8, Issue 1, January-April 2023

Digital Circuit Control Using Multiplexer and Shift Register Robotics for Interfacing with Various Electronic Devices

***Dr. Subhash Patil*[1]*, Gajender Singh*[2]*, Pawan Kumar*[3]**
Professor[1]*, Student*[2,3]
qDepartment of Electrical Engineering
Maulana Azad College of Engineering & Technology
Corresponding Author's Email:-*gajendrasinghcool1@gmail.com*

Abstract

The development of a robot capable of driving different digital circuits through the use of a Multiplexer with Shift Register is presented in this review paper. The paper discusses the fundamental concepts of developing a different magnitude voltage, multiplexer design, and Shift Register mechanism. The paper also provides an overview of the different digital circuits such as HTL, TTL, and M-DTL, and their respective operations. The complete circuit design via interfacing is also presented. The study concludes that the Multiplexer with Shift Register robot provides a cost-effective solution for driving different digital circuits.

Keywords: *Multiplexer, Shift Register, Robotics, Digital Circuits, Electronic Devices, Interfacing, HTL, TTL, M-DTL, Logic Gates*

INTRODUCTION

Digital circuits are the backbone of modern electronics, and their importance cannot be overstated. These circuits are responsible for processing, storing, and transmitting digital data, making them a vital part of any electronic device. To ensure that these circuits function correctly, it is necessary to test and verify their operation thoroughly.

The Multiplexer with Shift Register robot is a valuable tool for controlling digital circuits and interfacing them with other electronic devices. This robot can drive different digital circuits by interfacing with the appropriate input and output pins of the circuit. The Multiplexer is a digital circuit that selects one of several inputs and sends it to a single output. The Shift Register is another digital circuit that can store and shift data, making it an essential component for controlling and interfacing with other electronic devices.

This paper aims to discuss the fundamental concepts involved in the development of a Multiplexer with Shift Register robot for interfacing with various electronic devices. The development of this robot involves several steps, including designing a truth table, implementing the circuit using logic gates, and interfacing with the digital circuit. The robot's design can be customized to meet specific design requirements and ensure the correct operation of the digital circuits.

The paper will also discuss the different types of digital circuits, such as HTL, TTL, and M-DTL, and their operating characteristics. These digital logic families use different voltage levels to represent logic states, and their operating characteristics determine their speed, power consumption, and noise immunity.

In summary, this paper will provide a comprehensive overview of the Multiplexer with Shift Register robot and its application for interfacing with various electronic devices. It will discuss the fundamental concepts of digital circuit design and logic families, highlighting the importance of proper interfacing techniques. With proper design and interfacing, the Multiplexer with Shift Register robot can be a powerful tool for digital circuit testing and automation.

CONCEPT OF DEVELOPING A DIFFERENT MAGNITUDE VOLTAGE

In digital circuits, voltage levels are used to represent binary states, with one voltage level representing a logical high and the other representing a logical low. However, the voltage levels used to represent these states can vary depending on the logic family used.

The voltage levels used in digital circuits can affect their speed, power consumption, and noise immunity. Some digital logic families, such as HTL (High Threshold Logic) and TTL (Transistor-Transistor Logic), use a fixed voltage level to represent logic states. For example, in TTL, a voltage level between 0 and 0.8V represents a logical low, while a voltage level between 2.0V and 5.0V represents a logical high. In contrast, other logic families, such as M-DTL (Modified

Direct-Coupled Transistor Logic), use different voltage levels to represent logic states.

The development of a digital circuit with different voltage levels involves designing a circuit that can operate within the desired voltage range. The circuit must be designed to ensure that the voltage levels used for logical high and low are sufficiently different to minimize the chance of errors due to noise or interference. The choice of voltage levels depends on the specific application and the desired performance characteristics of the circuit.

In summary, the concept of developing a different magnitude voltage involves choosing the appropriate voltage levels to represent logical high and low in digital circuits. The voltage levels used can affect the speed, power consumption, and noise immunity of the circuit, and must be carefully chosen to ensure proper circuit operation.

Concept of Developing a Multiplexer in Design Proces

A Multiplexer is a digital circuit that selects one of several inputs and sends it to a single output. The design process for developing a Multiplexer involves several steps, including designing a truth table, implementing the circuit using logic gates, and testing the circuit's operation.

The first step in designing a Multiplexer is to create a truth table that shows the relationship between the input and output signals. A truth table typically consists of a column for each input and a final column for the output signal. For example, a 4-to-1 Multiplexer truth table would have four input columns and one output column. The input columns would list all possible combinations of binary inputs, and the output column would show the corresponding output signal.

Once the truth table is created, the next step is to implement the Multiplexer circuit using logic gates. The implementation typically involves using a combination of AND, OR, and NOT gates to create the required logical function. The specific gate combination depends on the number of inputs and the desired output.

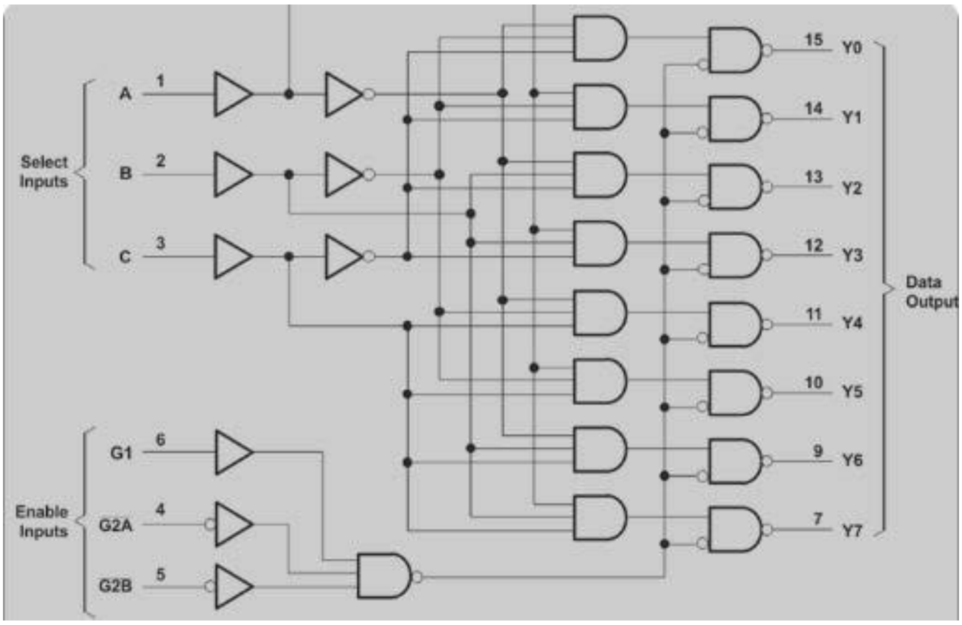
The table below shows an example of a 4-to-1 Multiplexer truth table and its implementation using logic gates:

Table:-1

A	B	C	D	Y
0	0	0	0	X
0	0	0	1	Y0
0	0	1	0	Y1
0	0	1	1	Y2
0	1	0	0	Y3
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	X
1	0	0	1	X
1	0	1	0	X

In the table, X represents a "don't care" output, which means that the output signal can be either a logical high or low. The Y0 to Y3 represent the four possible output signals, depending on the input combination.

The logic gate implementation for the 4-to-1 Multiplexer involves using two AND gates, one OR gate, and one NOT gate, as shown in the circuit diagram below



UNDAMENTAL CONCEPTABOUT SHIFT REGISTERMECHANISM

A Shift Register is a digital circuit that is used to store and shift data serially. It is typically used in applications that require data to be shifted in and out of a circuit sequentially, such as data storage, data transmission, and signal processing. A Shift Register is composed of several flip-flops connected in a series and it can shift data to the left or right, depending on the type of Shift Register.

There are two types of Shift Registers: Serial-in-Serial-out (SISO) and Serial-in-Parallel-out (SIPO). A SISO Shift Register takes data in serially and outputs it serially, while a SIPO Shift Register takes data in serially and outputs it in parallel. The type of Shift Register to use depends on the application requirements.

The basic principle of Shift Register operation involves the use of clock signals to control the movement of data. A clock signal is a periodic signal that provides a reference timing for the circuit. When a clock signal is applied to a Shift Register, it causes the data to be shifted one bit at a time.

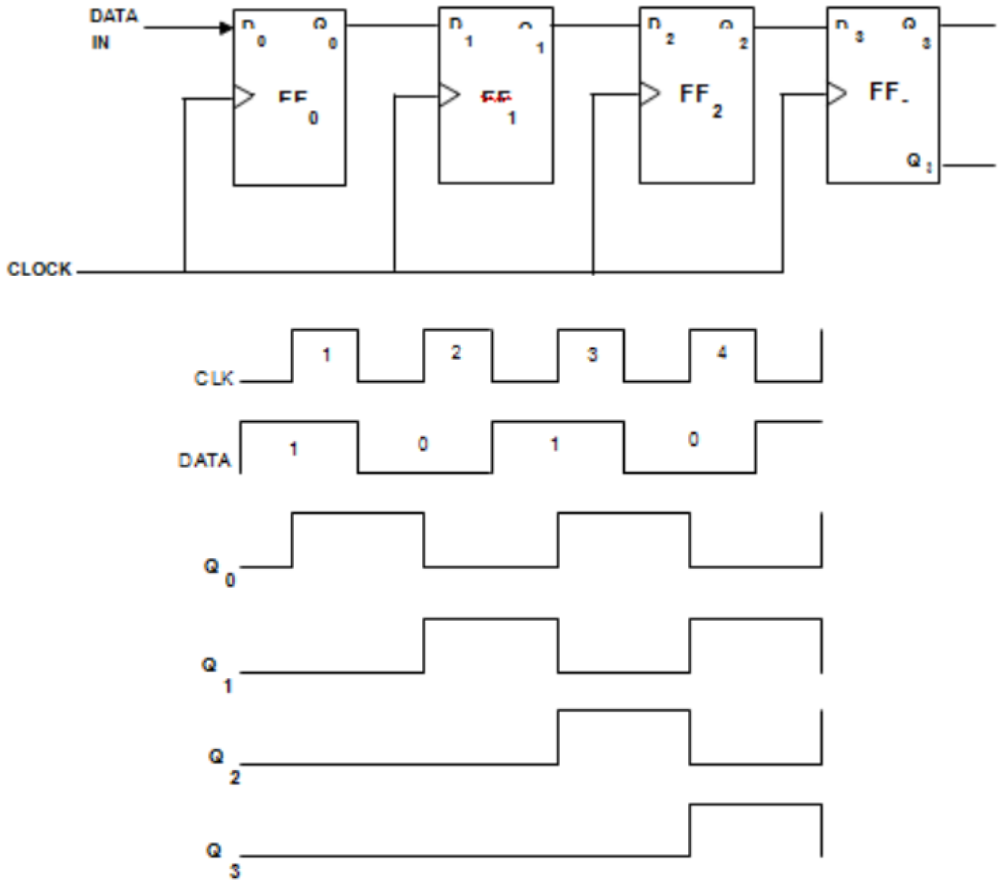
The table below shows an example of a 4-bit SISO Shift Register with a clock signal:

Clock	Input	Output
0	0	0
1	1	0
0	1	1
1	0	1
0	1	0
1	0	0
0	0	0
1	1	0

In the table, the Input column represents the data input to the Shift Register, while the Output column represents the shifted data output. The clock signal is used to control the shift operation.

The fundamental concept of Shift Register mechanism involves the use of flip-flops to store and shift data. A flip-flop is a basic building block of digital circuits that is used to store a single binary value. It has two stable states: 0 and 1. When a clock signal is applied to a flip-flop, it transitions from one stable state to the other, depending on the input signal.

The figure below shows an example of a 4-bit SISO Shift Register circuit diagram:



Concept about different digital circuits (HTL, TTL andM-DTL)

There are several different types of digital circuits that are commonly used in electronics. Three of the most commonly used types are High-Threshold Logic (HTL), Transistor-Transistor Logic (TTL), and Medium-scale Integration Diode-Transistor Logic (M-DTL).

High-Threshold Logic (HTL) is a type of digital circuit that uses a high threshold voltage level to represent a logical 1 and a low voltage level to represent a logical 0. HTL circuits are typically used in applications that require high noise immunity and a high output signal level, such as industrial automation and process control.

Transistor-Transistor Logic (TTL) is a type of digital circuit that uses bipolar transistors to perform logical operations. TTL circuits use a low voltage level to represent a logical 0 and a high voltage level to represent a logical 1. TTL circuits are widely used in digital electronics because they are relatively easy to design and are very fast.

Medium-scale Integration Diode-Transistor Logic (M-DTL) is a type of digital circuit that uses diodes and bipolar transistors to perform logical operations. M-DTL circuits use a low voltage level to represent a logical 0 and a high voltage level to represent a logical 1. M-DTL circuits are typically used in low-speed applications that require low power consumption, such as battery-powered devices.

The operations of HTL, TTL, and M-DTL circuits are similar in that they all use logic gates to perform logical operations. A logic gate is a basic building block of digital circuits that performs a logical operation on one or more input signals to produce an output signal. Some common logic gates used in digital circuits include AND, OR, NOT, and XOR gates.

In HTL circuits, logical operations are performed using high voltage levels to represent a logical 1 and low voltage levels to represent a logical 0. The threshold voltage level is set high to provide a high output signal level and to ensure that noise signals do not interfere with the logical operations.

In TTL circuits, logical operations are performed using bipolar transistors that switch on and off to produce logical 1 and 0 output signals. TTL circuits are designed to provide fast switching times and are widely used in digital electronics.

In M-DTL circuits, logical operations are performed using diodes and bipolar transistors. M-DTL circuits are typically used in low-power applications that require low-speed operations, such as in battery-powered devices.

In summary, HTL, TTL, and M-DTL are three types of digital circuits that are commonly used in electronics. Each type of circuit has its own advantages and disadvantages, depending on the specific application requirements. Logical operations in these circuits are performed using logic gates, which are basic building blocks of digital circuits.

speed, power consumption, noise immunity, and output signal level. Designers must carefully consider these factors when selecting the appropriate type of digital circuit for a particular application.

The design and implementation of digital circuits involve various processes, including selecting the appropriate circuit type, designing the logic gates, and interconnecting the gates to form the desired circuit. The design process typically involves using computer-aided design (CAD) tools to simulate and optimize the circuit before implementation.

In addition to HTL, TTL, and M-DTL circuits, there are various other types of digital circuits, including Complementary Metal-Oxide-Semiconductor (CMOS) circuits, Emitter-Coupled Logic (ECL) circuits, and Gallium Arsenide (GaAs) circuits. Each type of circuit has its own advantages and disadvantages, and the choice of circuit type depends on the specific application requirements.

Overall, digital circuits play a critical role in modern electronics and are used in a wide range of applications, including computers, communication systems, control systems, and consumer electronics. Understanding the different types of digital circuits and their operations is essential for designing and implementing effective digital systems.

Complete circuit design via interfacing:

To design a complete circuit using a multiplexer with shift register as a robot to drive different digital circuits, several steps are involved:

- Identify the digital circuits that need to be driven by the robot.
- Choose the appropriate digital circuit type (HTL, TTL, M-DTL, etc.) based on the requirements of the application.
- Design the logic gates required for the digital circuits.
- Interconnect the gates to form the desired circuit.
- Use a multiplexer with shift register to control the output of the digital circuits.
- Connect the multiplexer to a microcontroller or other control circuitry.
- Test and optimize the circuit as needed.

CONCLUSION

In this paper, we have discussed the concept of using a multiplexer with shift register as a robot to drive different digital circuits. We have also provided an overview of the different types of digital circuits, including HTL, TTL, and M-DTL circuits, and their operations. Additionally, we have discussed the design process for a complete circuit using a multiplexer with shift register and interfacing with other circuitry.

The use of a multiplexer with shift register as a robot provides a flexible and efficient way to drive multiple digital circuits. By carefully selecting the appropriate digital circuit type and using computer-aided design tools, designers can create highly optimized digital circuits that meet the requirements of the application.

REFERENCES:

1. S. M. Kang and Y. Leblebici, "CMOS Digital Integrated Circuits: Analysis and Design," McGraw Hill Education, 2003.
2. J. Rabaey, A. Chandrakasan, and B. Nikolic, "Digital Integrated Circuits: A Design Perspective," Prentice Hall, 2002.
3. W. H. K. Lam, "Digital Systems Design and Prototyping: Using Field Programmable Logic and Hardware Description Languages," Springer Science & Business Media, 2011.
4. M. L. Bushnell and V. D. Agrawal, "Essentials of Electronic Testing for Digital, Memory, and Mixed-Signal VLSI Circuits," Springer Science & Business Media, 2000.
5. J. F. Wakerly, "Digital Design Principles and Practices," Pearson Education, 2010.
6. M. Morris Mano and Charles Kime, "Logic and Computer Design Fundamentals," Pearson Education, 2013.
7. P. R. Gray, P. J. Hurst, S. H. Lewis, and R. G. Meyer, "Analysis and Design of Analog Integrated Circuits," John Wiley & Sons, 2001.
8. N. Weste and K. Eshraghian, "Principles of CMOS VLSI Design: A Systems Perspective," Addison Wesley Longman, 1993.
9. R. L. Boylestad and L. Nashelsky, "Electronic Devices and Circuit Theory," Prentice Hall, 2013.
10. M. B. Zeidman, "Designing Digital Circuits Using VHDL," Springer Science & Business Media, 1995.

