Basic data transmission multiplexing via 2-to-1 Mux and 1-to-2 Demux

Lyle Kenneth M. Geraldez*

National Institute of Physics, University of the Philippines - Diliman, Philippines
*Corresponding author: lgeraldez@nip.upd.edu.ph

Abstract

Multiplexing is used in a wide array of complex network data transmission applications. A basic multiplexing configuration consists of a 2-to-1 multiplexer acting as a transmitter to choose was signal to send and a 1-to-2 demultiplexer acting as a receiver to send the chosen data into the appropriate output line. The paper aims to design, simulate, and assess a basic data transmission multiplexing circuit using logic gates and an Arduino UNO as an input signal proxy. By separately tackling the mechanisms of the transmitter and receiver part, the paper shows the working mechanism of multiplexing piece-by-piece building upon individual building blocks and, then, shows the final data transmission mechanism by connecting these blocks.

Keywords: multiplexing, data transmission

1 Data transmission multiplexing

Complex network array data transmission often uses *multiplexing* to consolidate multiple signals into a single composite single which is capable of being fed into a single serial line. This composite single could, then, be transported through a single medium such as fiber optics or radio waves [1]. The common serial line containing the composite signal from different input signals is then fed into another circuit construction capable of recovering the separate signals and sending them again to the appropriate individual lines. The ability to consolidate different signals into a single composite one is called *multiplexing*; combinational circuit construction capable of this mechanism is called a *multiplexer* (or mux). The ability to recover the individual signals and send them to the appropriate lines is called *demultiplexing* and the combinational circuit construction capable of this mechanism is called a demultiplexer (or demux). A variety of multiplexing techniques is being used in networks yet the working mechanism for all these variations is essentially the same - converting parallel data to serial (muxing) transmitted through a share medium and then back to parallel (demuxing) on separate lines to be used by a variety of operations. One commonly used technique is the time-division multiplexing (TDM) - method of multiplexing where each signal crossing the common serial line appears for only a fraction of time in an alternating pattern. TMD is capable of transmitting two or more digital signals if the bit rate of the transmission medium exceeds the signal sources. This multiplexing technique is commonly found on digital telephones in the second half of 20th century [2]. In this paper, we aim to construct a basic multiplexed 2-bit data transmission and reception using 2-to-1 multiplexer on the transmitting end and 1-to-2 demultiplexer on the receiving end.

2 2-to-1 multiplexer

At the transmitting end, a 2-to-1 multiplexer was used to compactify the multiple signals (in this case, two signals - hence, 2-to-1) , with different frequencies into a single composite one sharing the same serial line of transmission. To construct the mux, two AND gates, one OR gate, and one NOT gate is used as shown in the following circuit.

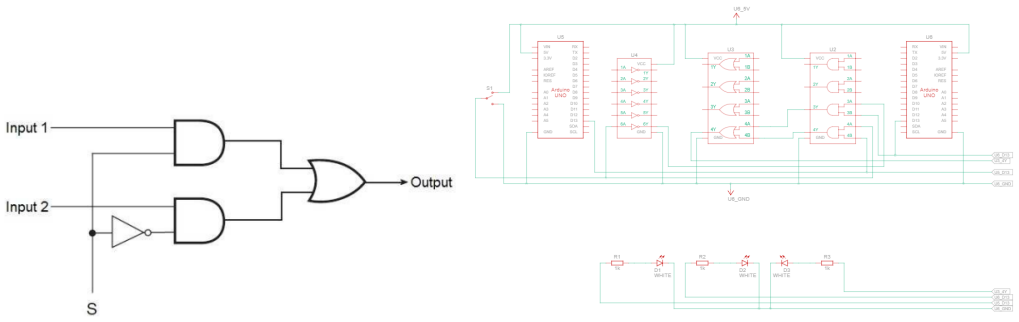

Figure 1: To implement the logic, a quad AND gate (74HC08), a quad OR gate (74HC32), and a quad hex inverter (74HC04) were used as shown on the right circuit diagram.

The circuit design above was constructed using *Tinkercad* and controlled using Arduino UNO. The two input signals possess different frequencies. For Input 1, two DigitalWrite() functions corresponding to HIGH and low states were separated by a delay(333) or 3 Hz. or Input 2, two DigitalWrite() functions corresponding to HIGH and low states were separated by a delay(1000) or 1 Hz. The other (inverted) leg of an AND gate was fed by a switched +5V - Ground input using a toggle switch corresponding to the SELECT pin of the mux. The system contains two different states depending on the toggle state of the SELECT pin switch. To illustrate the operating mechanism of the 2-to-1 multiplexer, the following shows the truth table of the system.

SELECT	Input 1	Input 2	Output
0	0	x	0
0	1	x	1
1	x	0	0
1	x	1	1

To see the operation in action, see the following movie: https://github.com/schwarzschlyle/electronics-and-instrumentation/blob/master/mux-demux/mux-sim.gif. From the truth table, it can be observed that a 2-to-1 multiplexer clearly shows a data selection mechanism by acting like an SPDT switch. Switching the SELECT pin to a LOW state, the output follows the signal from input 1 immaterial of the state of the input 2, The converse is true if the SELECT pin is set to a HIGH state. In this case, the output follows the signal from input 2 immaterial of the state of input 1. Concisely put, 2-to-1 multiplexer controls which input signal travels the common serial line in a multiplexed data transmission application serving as its transmission section.

3 1-to-2 demultiplexer

At the receiving end, a 1-to-demultiplexer was constructed which serves to convert the transmitted serial data from the 2-to-1 multiplexer back to the appropriate parallel and originally multiple signals. To do so, two AND gates and one NOT gate were used as shown in the following circuit configuration

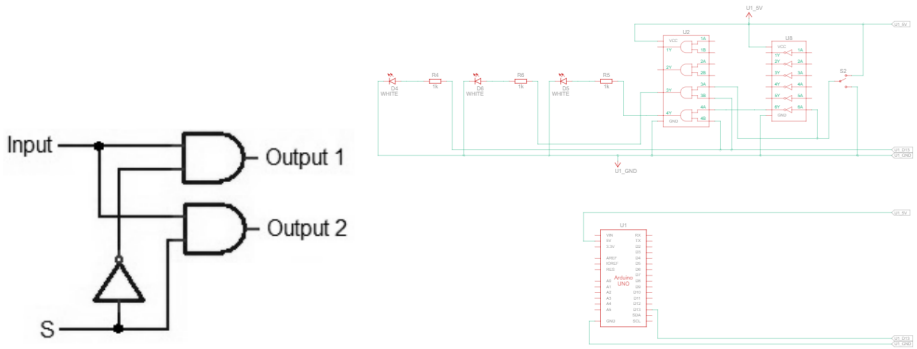

Figure 2: To implement the logic, a quad AND gate (74HC08) and a quad hex inverter (74HC04) were used as shown on the right circuit diagram.

In order to first separately analyze the working mechanism of the 1-to-2 demultiplexer, a proxy signal was fed into the Input pin using an Arduino UNO. The serial input signal proxy was set to have a delay(1000) corresponding to a frequency of 1 Hz. Similarly, the SELECT pin of the 1-to-2 demux was a switch between +5V and Ground using a toggle switch. To illustrate the working mechanism, we can similarly plot the corresponding truth table of the different SELECT states.

SELECT	Input	Output 1	Output 2
0	0	0	0
0	1	0	1
1	0	0	0
1	1	1	0

Similarly, to see the operation in action, see the following movie: https://github.com/schwarzschlyle/electronics-and-instrumentation/blob/master/mux-demux/demux-sim.gif. Observe from the movie and the truth table that the 1-to-2 demultiplexer receives the input signal and "routes" the signal on two different parallel paths. The serial signal is sent into different output lines depending on the state of the SELECT pin. At the LOW SELECT pin state, the input signal was sent into the Output 2 line. At the HIGH SELECT pin state, the input signal was sent into the Output 1 line. This serves as the receiving end of the basic 2-to-1 - 1-to-2 multiplexed data transmission.

4 Basic mux-demux multiplexing demonstration

Using the two combination circuit configuration - the transmitter and the receiver, we can now construct the full basic multiplexed data transmission line converting the two different signals with different frequencies, transmitting them into a single serial line, receiving it at the receiver, and converting them again back into their original parallel signal form. To do so, we simply connect the output of the 2-to-1 multiplexer into the input of the 1-to-2 demultiplexer.

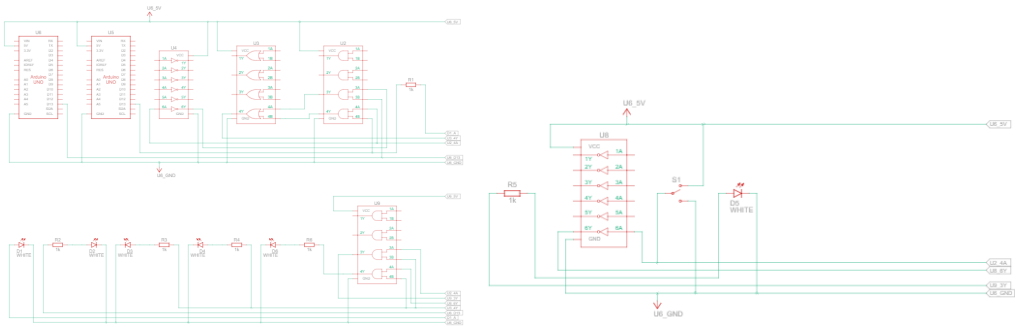

Figure 3: Figure shows circuit diagram of connected basic mux-demux configuration with incorporated LED pins to track the states at each input and output.

To cleanup and finalize the desired circuit applications, we simply interchange the labels of Output 1 and Output 2 at the demux part of the system. From the previous discussion, the transmitter part chooses the data signal to be transmitted and the receiver part chooses what line to send the signal. Connecting the SELECT pin to share a common state, the truth table is as follows:

SELECT	Input 1	Input 2	Output 1	Output 2
0	0	x	0	0
0	1	x	1	0
1	x	0	0	0
1	x	1	0	1

The working mechanism can be seen here: https://github.com/schwarzschlyle/electronics-and-instrumentation/blob/master/mux-demux/mux-demux-sim.gif. From the truth table, the basic multiplexing working mechanism can be inferred. For LOW SELECT pin states, the data signal from Input 1 gets sent to the Output 1 line while data from Input 2 remains at the input line as can be observed from the LOW states of the Output 2 pin. Conversely, switching into a HIGH select pin state, the data signal from Input 2 gets set to the Output 2 lien while data from Input 1 remains at the input

line as can be observed from the LOW states of the Output 1 pin. Clearly, the function of this basic mux-demux data transmission multiplexing is to transmit multiple data signal using only one serial line. Of course, this configuration can be improved to handle large amounts of input and output signals and is commonly used to improve the efficiency of data transmission in computer circuit or telecommunication applications.

5 Conclusion

The paper aims to construct a basic mux-demux data transmission multiplexing. This is done by, first, constructing a transmitter 2-to-1 multiplexer circuit using AND, OR, and NOT gates and then constructing a receiver 1-to-2 demultiplexer circuit using AND and NOT gates. By connecting this individual circuit configuration, a final 2-to-1 - 1-to-2 data transmission muxing was designed which is capable of choosing which signal to send and where appropriate output line to send it. This forms the basis of a plethora of multiplexing applications such as time-division multiplexing.

References

[1] T. Contributor, What is time-division multiplexing? (2021), https://www.techtarget.com/whatis/definition/time-division-multiplexing-TDM.

[2] R. Teja, Multiplexer and demultiplexer circuit diagrams and applications (2021), https://www.electronicshub.org/multiplexer-and-demultiplexer.

6 Codes and Raw Data

All files regarding the experiment can be found here: https://github.com/schwarzschlyle/electronics-and-instrumentation/tree/master/mux-demux