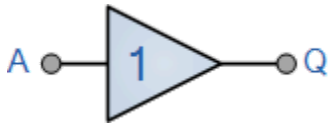


Digital Buffer Tutorial

Digital Buffers and Tri-state Buffers can provide current amplification in a digital circuit to drive output loads

In a previous tutorial we looked at the digital **Not Gate** commonly called an inverter, and we saw that the NOT gates output state is the complement, opposite or inverse of its input signal.

So for example, when the single input to NOT gate is "HIGH", its output state will NOT be "HIGH". When its input signal is "LOW" its output state will NOT be "LOW", in other words it "inverts" its input signal, hence the name "Inverter".

But sometimes in digital electronic circuits we need to isolate logic gates from each other or have them drive or switch higher than normal loads, such as relays, solenoids and lamps without the need for inversion. One type of single input logic gate that allows us to do just that is called the **Digital Buffer**.

Unlike the single input, single output inverter or NOT gate such as the TTL 7404 which inverts or complements its input signal on the output, the "Buffer" performs no inversion or decision making capabilities (like logic gates with two or more inputs) but instead produces an output which exactly matches that of its input. In other words, a digital buffer does nothing as its output state equals its input state.

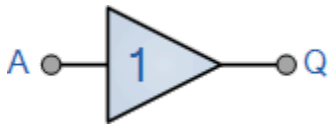
Then digital buffers can be regarded as Idempotent gates applying Boole's Idempotent Law because when an input passes through this device its value is not changed. So the digital buffer is a "non-inverting" device and will therefore give us the Boolean expression of: $Q = A$.

Then we can define the logical operation of a single input digital buffer as being:

"Q is true, only when A is true"

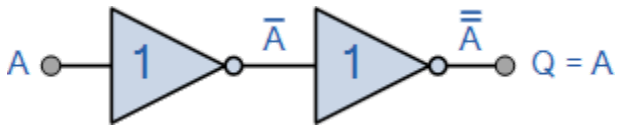
In other words, the output ("Q") state of a buffer is only true (logic "1") when its input "A" is true, otherwise its output is false (logic "0").

The Single Input Digital Buffer

Symbol	Truth Table	
 The Digital Buffer	A	Q
	0	0
	1	1
Boolean Expression **Q = A**	Read as: **A** gives **Q**	

The **Digital Buffer** can also be made by connecting together two NOT gates as shown below. The first will "invert" the input signal A and the second will "re-invert" it back to its original level performing a double inversion of the input.

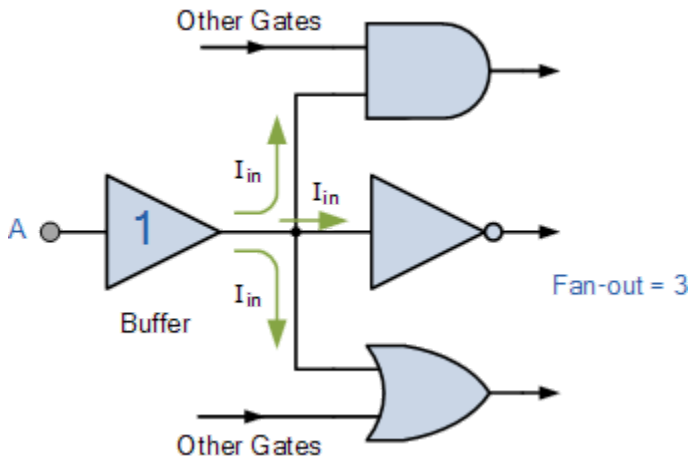
Double Inversion using NOT Gates



You may be thinking, well what's the point of a Digital Buffer if it does not invert or alter its input signal in any way, or make any logical decisions or operations like the AND or OR gates do, then why not just use a piece of wire instead, and that's a good point. But a non-inverting Digital Buffer does have many uses in digital electronics with one of its main advantages being that it provides digital amplification.

Digital Buffers can be used to isolate other gates or circuit stages from each other preventing the impedance of one circuit from affecting the impedance of another. A digital buffer can also be used to drive high current loads such as transistor switches because their output drive capability is generally much higher than their input signal requirements. In other words buffers can be used for power amplification of a digital signal as they have what is called a high "fan-out" capability.

Digital Buffer Fan-out Example



The **Fan-out** parameter of a buffer (or any digital IC) is the output driving capability or output current capability of a logic gate giving greater power amplification of the input signal. It may be necessary to connect more than just one logic gate to the output of another or to switch a high current load such as an LED, then a Buffer will allow us to do just that.

Generally the output of a logic gate is usually connected to the inputs of other gates. Each input requires a certain amount of current from the gate output to change state, so that each additional gate connection adds to the load of the gate. So the fan-out is the number of parallel loads that can be driven simultaneously by one digital buffer of logic gate. Acting as a current source a buffer can have a high fan-out rating of up to 20 gates of the same logic family.

If a digital buffer has a high fan-out rating (current source) it must also have a high "fan-in" rating (current sink) as well. However, the propagation delay of the gate deteriorates rapidly as a function of fan-in so gates with a fan-in greater than 4 should be avoided.

Then there is a limit to the number of inputs and outputs than can be connected together and in applications where we need to decouple gates from each other, we can use a **Tri-state Buffer** or tristate output driver.

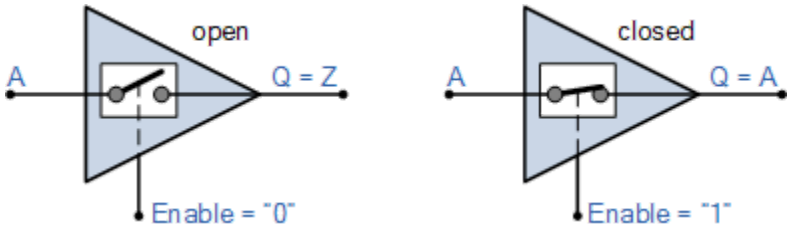
The "Tri-state Buffer"

As well as the standard Digital Buffer seen above, there is another type of digital buffer circuit whose output can be "electronically" disconnected from its output circuitry when required. This type of Buffer is known as a 3-State Buffer or more commonly a Tri-state Buffer.

A Tri-state Buffer can be thought of as an input controlled switch with an output that can be electronically turned "ON" or "OFF" by means of an external "Control" or "Enable" (EN) signal input. This control signal can be either a logic "0" or a logic "1" type signal resulting in the Tri-state Buffer being in one state allowing its output to operate normally producing the required output or in another state were its output is blocked or disconnected.

Then a tri-state buffer requires two inputs. One being the data input and the other being the enable or control input as shown.

Tri-state Buffer Switch Equivalent

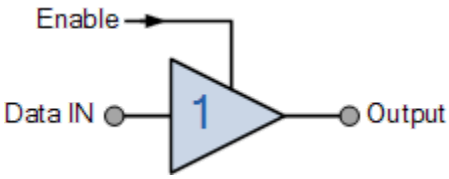


When activated into its third state it disables or turns “OFF” its output producing an open circuit condition that is neither at a logic “HIGH” or “LOW”, but instead gives an output state of very high impedance, **High-Z**, or more commonly Hi-Z. Then this type of device has two logic state inputs, “0” or a “1” but can produce three different output states, “0”, “1” or ” Hi-Z ” which is why it is called a “Tri” or “3-state” device.

Note that this third state is NOT equal to a logic level “0” or “1”, but is an high impedance state in which the buffers output is electrically disconnected from the rest of the circuit. As a result, no current is drawn from the supply.

There are four different types of Tri-state Buffer, one set whose output is enabled or disabled by an “**Active-HIGH**” control signal producing an inverted or non-inverted output, and another set whose buffer output is controlled by an “**Active-LOW**” control signal producing an inverted or non-inverted output as shown below.

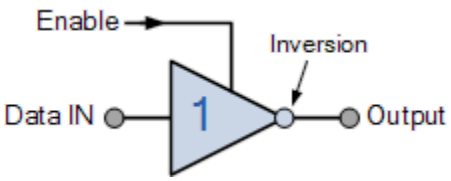
Active “HIGH” Tri-state Buffer

Symbol	Truth Table		
 Tri-state Buffer	Enable	IN	OUT
	0	0	Hi-Z
	0	1	Hi-Z
	1	0	0
	1	1	1
Read as Output = Input if Enable is equal to “1”			

An **Active-high** Tri-state Buffer such as the 74LS241 octal buffer, is activated when a logic level “1” is applied to its “enable” control line and the data passes through from its input to its output. When the enable control line is at logic level “0”, the buffer output is disabled and a high impedance condition, Hi-Z is present on the output.

An active-high tri-state buffer can also have an inverting output as well as its high impedance state creating an active-high tri-state inverting buffer as shown.

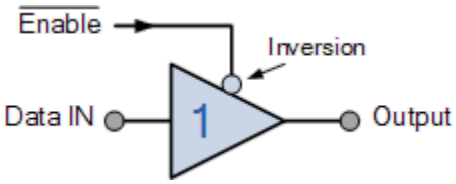
Active “HIGH” Inverting Tri-state Buffer

Symbol	Truth Table		
 Inverting Tri-state Buffer	Enable	IN	OUT
	0	0	Hi-Z
	0	1	Hi-Z
	1	0	1
	1	1	0
Read as Output = Inverted Input if Enable equals “1”			

The output of an active-high inverting tri-state buffer, such as the 74LS240 octal buffer, is activated when a logic level “1” is applied to its “enable” control line. The data at the input is passes through to the output but is inverted producing a complement of the input. When the enable line is LOW at logic level “0”, the buffer output is disabled and at a high impedance condition, Hi-Z.

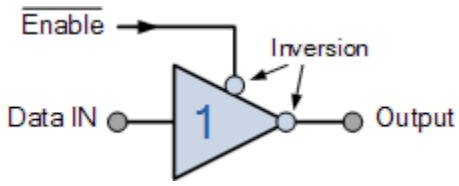
The same two tri-state buffers can also be implemented with an active-low enable input as shown.

Active “LOW” Tri-state Buffer

Symbol	Truth Table		
 Tri-state Buffer	Enable	IN	OUT
	0	0	0
	0	1	1
	1	0	Hi-Z
	1	1	Hi-Z
Read as Output = Input if Enable is **NOT** equal to “1”			

An **Active-low** Tri-state Buffer is the opposite to the above, and is activated when a logic level “0” is applied to its “**enable**” control line. The data passes through from its input to its output. When the enable control line is at logic level “1”, the buffer output is disabled and a high impedance condition, Hi-Z is present on the output.

Active “LOW” Inverting Tri-state Buffer

Symbol	Truth Table		
 Inverting Tri-state Buffer	Enable	IN	OUT
	0	0	1
	0	1	0
	1	0	Hi-Z
	1	1	Hi-Z
Read as Output = Inverted Input if Enable is **NOT** equal to "1"			

An **Active-low** Inverting Tri-state Buffer is the opposite to the above as its output is enabled or disabled when a logic level "0" is applied to its "**enable**" control line. When a buffer is enabled by a logic "0", the output is the complement of its input. When the enable control line is at logic level "1", the buffer output is disabled and a high impedance condition, Hi-Z is present on the output.

Tri-state Buffer Control

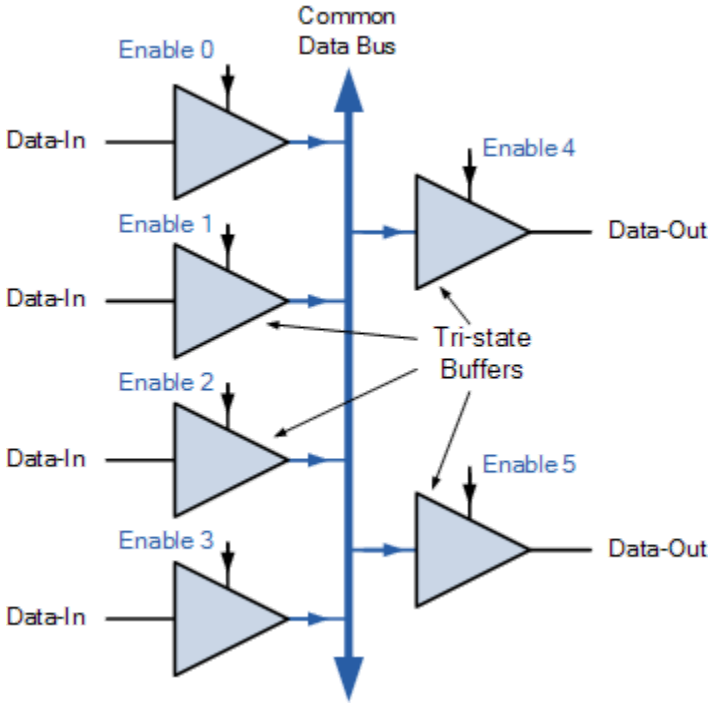
We have seen above that a buffer can provide voltage or current amplification within a digital circuit and it can also be used to invert the input signal. We have also seen that digital buffers are available in the tri-state form that allows the output to be effectively switched-off producing a high impedance state (Hi-Z) equivalent to an open circuit.

The Tri-state Buffer is used in many electronic and microprocessor circuits as they allow multiple logic devices to be connected to the same wire or bus without damage or loss of data. For example, suppose we have a data line or data bus with some memory, peripherals, I/O or a CPU connected to it. Each of these devices is capable of sending or receiving data to each other onto this single data bus at the same time creating what is called a contention.

Contention occurs when multiple devices are connected together because some want to drive their output high and some low. If these devices start to send or receive data at the same time a short circuit may occur when one device outputs to the bus a logic "1", the supply voltage, while another is set at logic level "0" or ground, resulting in a short circuit condition and possibly damage to the devices as well as loss of data.

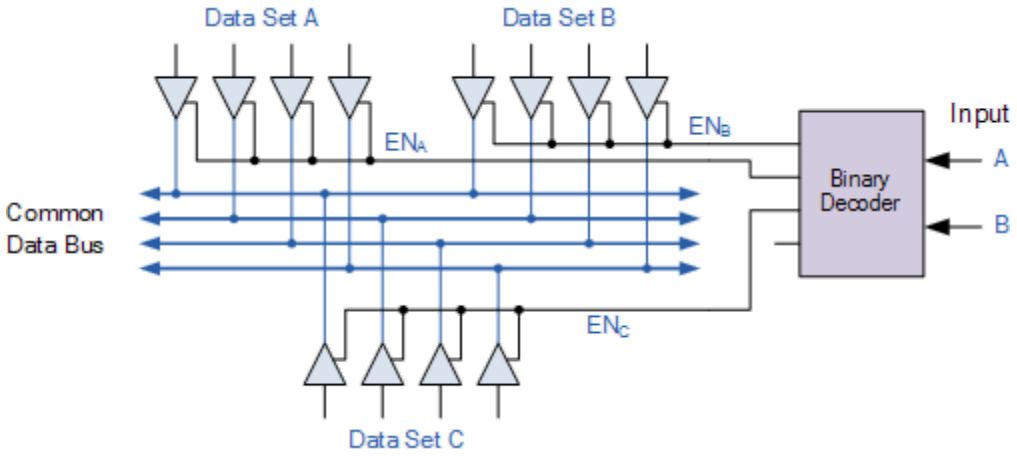
Digital information is sent over these data buses or data highways either serially, one bit at a time, or it may be up to eight (or more) wires together in a parallel form such as in a microprocessor data bus allowing multiple tri-state buffers to be connected to the same data highway without damage or loss of data as shown.

Tri-state Buffer Data Bus Control



Then, the **Tri-state Buffer** can be used to isolate devices and circuits from the data bus and one another. If the outputs of several Tri-state Buffers are electrically connected together Decoders are used to allow only one set of Tri-state Buffers to be active at any one time while the other devices are in their high impedance state. An example of Tri-state Buffers connected to a 4-wire data bus is shown below.

Tri-state Buffer Control



This basic example shows how a binary decoder can be used to control a number of tri-state buffers either individually or together in data sets. The decoder selects the appropriate output that corresponds to its binary input allowing only one set of data to pass either a logic “1” or logic “0” output state onto the bus. At this time all the other tri-state outputs connected to the same bus lines are disabled by being placed in their high impedance Hi-Z state.

Then data from data set "A" can only be transferred to the common bus when an active HIGH signal is applied to the tri-state buffers via the Enable line, EN_A. At all other times it represents a high impedance condition effectively being isolated from the data bus.

Likewise, data set "B" only passes data to the bus when an enable signal is applied via EN_B. A good example of tri-state buffers connected together to control data sets is the TTL 74244 Octal Buffer.

It is also possible to connect Tri-state Buffers "back-to-back" to produce what is called a **Bi-directional Buffer** circuit with one "active-high buffer" connected in parallel but in reverse with one "active-low buffer".

Here, the "enable" control input acts more like a directional control signal causing the data to be both read "from" and transmitted "to" the same data bus wire. In this type of application a tri-state buffer with bi-directional switching capability such as the TTL 74245 can be used.

We have seen that a **Tri-state buffer** is a non-inverting device which gives an output (which is same as its input) only when the input to the Enable, (EN) pin is HIGH otherwise the output of the buffer goes into its high impedance, (Hi-Z) state. Tri-state outputs are used in many integrated circuits and digital systems and not just in digital tristate buffers.

Both digital buffers and tri-state buffers can be used to provide voltage or current amplification driving much high loads such as relays, lamps or power transistors than with conventional logic gates. But a buffer can also be used to provide electrical isolation between two or more circuits.

We have seen that a data bus can be created if several tristate devices are connected together and as long as only one is selected at any one time, there is no problem. Tri-state buses allow several digital devices to input and output data on the same data bus by using I/O signals and address decoding.

Tri-state Buffers are available in integrated form as quad, hex or octal buffer/drivers in both uni-directional and bi-directional forms, with the more common being the TTL 74240, the TTL 74244 and the TTL 74245 as shown.

Commonly available **Digital Buffer** and **Tri-state Buffer** IC's include:

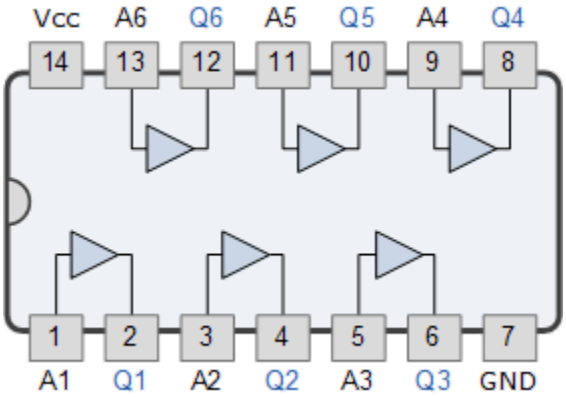
TTL Logic Digital Buffers

74LS07 Hex Non-inverting Buffer
74LS17 Hex Buffer/Driver
74LS244 Octal Buffer/Line Driver
74LS245 Octal Bi-directional Buffer

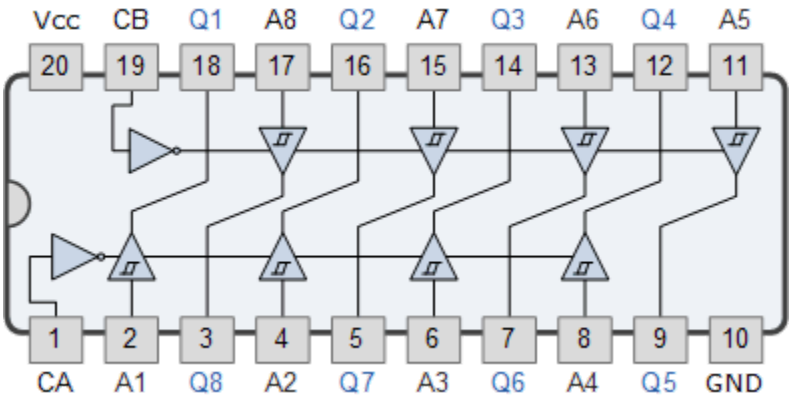
CMOS Logic Digital Buffers

CD4050 Hex Non-inverting Buffer
CD4503 Hex Tri-state Buffer
HEF40244 Tri-state Octal Buffer

74LS07 Digital Buffer



74LS244 Octal Tri-state Buffer



22 Comments

Join the conversation

Your Name

Email Address

Write your comment here

SUBMIT

P Pallavi b

Good explanation

Posted on July 10th 2018 | 2:07 pm

Reply

J Jono

What's a reinstate bufferbgents

Posted on January 08th 2018 | 11:31 am

Reply

A Anil veera

am electrician

Posted on April 17th 2017 | 6:42 am

Reply

F Frank Pajao

Does anyone know how many feet of cable can be driven by a 74ls245?
Thanks

Posted on April 07th 2017 | 5:05 am

Reply

V Varun

Suppose CPU wants to write a byte (8-bits) of data to a particular address(16-bit) location in RAM(64Kbyte).

1. CPU will assert 16 bit address onto address bus.
2. CPU will then assert 1 byte data onto the bus and assert RAM write signal.
3. RAM enabled by CPU will read the data from data bus written by CPU and write that into address read from address bus

For this communication to happen on shared data bus, both CPU and RAM must be connected simultaneously to the data bus. So in this case, the tristate buffers of both CPU and RAM must be simultaneous enabled. Is that correct ?

Posted on December 10th 2016 | 8:15 pm

Reply

W Wayne Storr

For the CPU to send or receive data to the RAM it must first select the memory location using the Address Bus, and indicate the direction of data transfer along the Data Bus. The shared data bus is bi-directional, thus both devices can send and receive data and to prevent bus conflicts with other devices connected to the data bus Tri-state Buffers are used.

The output of a tri-state buffer can be enabled or disabled (open-circuited) and when disabled the buffer behaves as though it output has been disconnected from the data bus (high impedance). Both the CPU and the RAM will have input and output tri-state buffers so that they are not actively connected to the shared bus until enabled or selected using their chip-select (CS) inputs. Then at any one time only ONE output buffer and ONE input buffer is actively connected to the shared data bus depending on the direction of data flow, CPU to RAM, or RAM to CPU.

Posted on December 11th 2016 | 9:33 am

Reply

---

k k

is there any device capable of “inverting” the HiZ state to either a HIGH or LOW state?

Posted on October 24th 2016 | 10:13 am

Reply

w weid guy

pull-up or pull-down resistors

Posted on September 30th 2017 | 10:07 am

Reply

E Esteban

Very didactic explanation tri-state buffer.

Posted on October 11th 2016 | 6:18 pm

Reply

---

h hitesh

Nice explanation about logic gates

Posted on August 27th 2016 | 1:06 pm

Reply

---

s sebina

Hi guys,
I want to use SN74HC245PWR to drive a 10mA load but I cant understand whats the amount of current output in that chip?

In datasheet highlights it has: +-6mA output drive 5V but
in absolute maximum it has : +-35mA for continues current

Posted on August 15th 2016 | 7:54 am

Reply

More

---

s sahal muhd

hm wanna join

Posted on June 10th 2016 | 5:14 pm

Reply

View More