Hardware Lab 2: TTL Characteristics, Open-Collector and Three-State Buffers

Prerequisites: Before beginning this laboratory experiment you must be able to:

- Communicate with the Arduino board using the Firmata Test application. Refer to the instructional videos for Week 2 on how to download and install the Arduino IDE, upload the firmware, download and test Firmata.
- Use a prototyping board.
- Be able to describe how a multiplexer works..

Equipment: Personal computer, Arduino Uno (or equivalent) with Firmata firmware loaded, USB cable, breadboard, breadboarding wire bundle.

Integrated Circuits: You will need the following IC to complete this lab:

- (1) 7404 (Hex Inverters)
- (1) 7405 (Hex Open-Collector Inverter Buffers)
- (1) 74LS126(Three-State Buffers) (Not 74HC126)
- (1) 1000 Ohm Resistor

Objective: In this laboratory you will learn about different output stage topologies and how to electrically connect open-collector and three-state buffer circuits to drive a common communication bus.

Outcomes: When you have completed the tasks in this experiment you will be able to:

- Make and interpret electrical voltage measurements.
- Describe the difference between buffered, open-collector and three-state outputs of digital logic gates.
- Electrically connect open-collector and three-state buffer circuits to drive a common communication bus.
- Describe how open-collector and three-state buffer circuits must be controlled to drive a common bus to avoid data conflicts.
- Observe the effects of conflicting output signals on a common communication bus, a so-called “data conflict”, on the voltage levels.

Introduction

In this laboratory exercise you will gain knowledge on what the difference is between “regular” or so-called “totem pole” outputs of logic gates and open-collector and three-state outputs. You will also learn how to work with open-collector and three-state buffers to drive common communication lines, so-called “bus lines”. In addition, you will gain experience on relating voltage levels at gate outputs to logic states.

Looking at how a multiplexer is constructed in Figure 1, only one of the AND gates can actually have a logic one at the output. All other AND gates will be “de-selected” and will only output a logic zero. However, the outputs of all the AND gates have to be wired to an OR gate to produce the multiplexed output signal. The reason for the necessity of the OR gate is that we are not allowed to simply connect the outputs of the AND gates. Having an OR gate with a lot of inputs is no problem from a logic gate perspective, but in digital systems this can become an issue, because all the AND outputs have to be wired to a common location, being the OR gate.

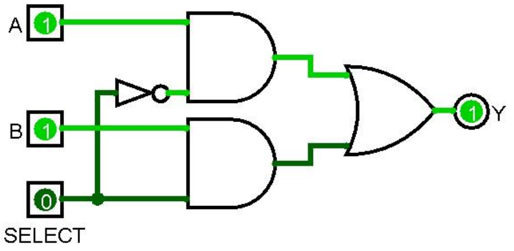
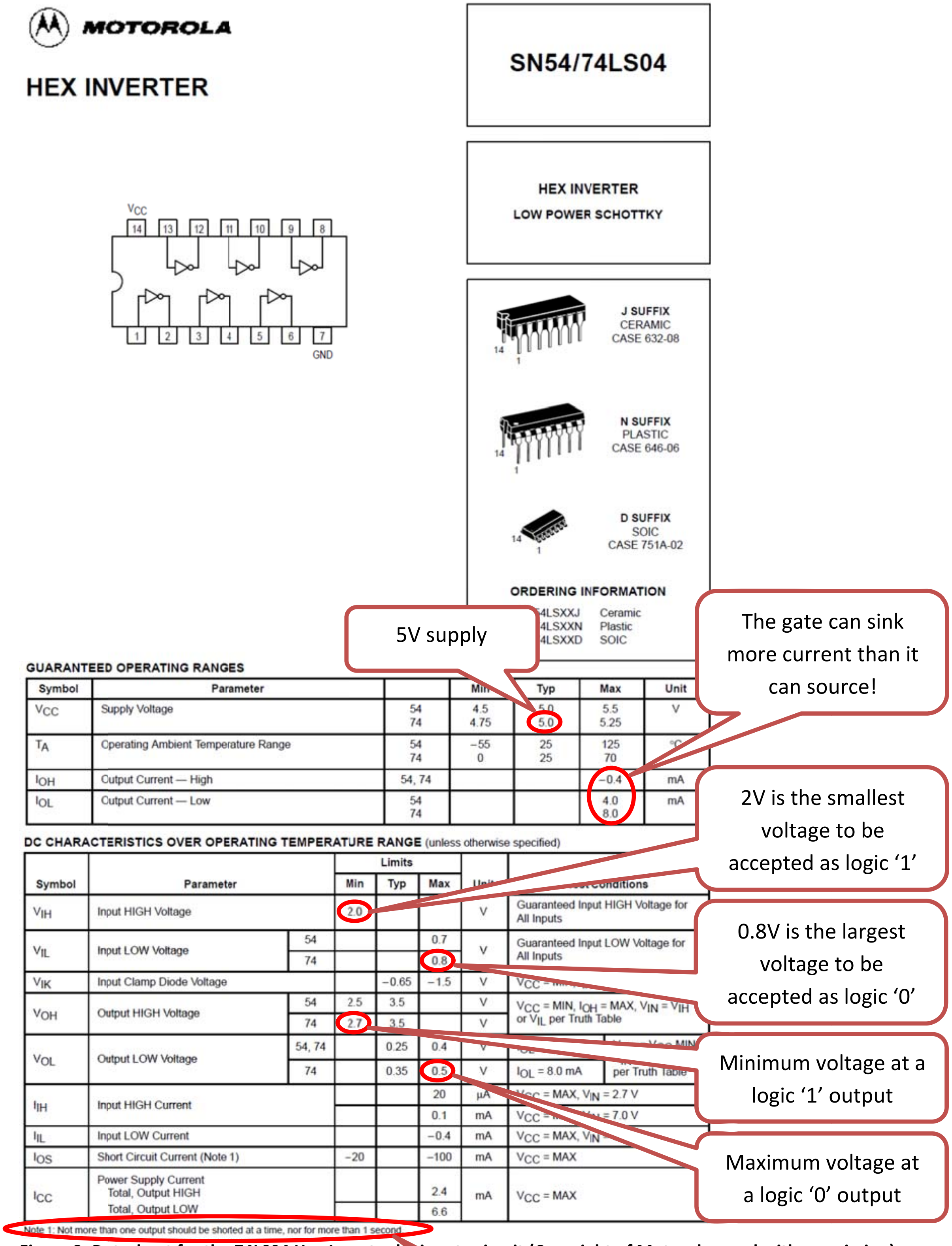


Figure 1: 2:1 Multiplexer using AND/OR/NOT gates. Only one of the AND gates will output logic one while the other gate will always output zero. The OR gate is only necessary to "combine" the signals.



MOTOROLA

HEX INVERTER

SN54/74LS04

HEX INVERTER

LOW POWER SCHOTTKY

VCC
14
13
12
11
10
9
8
1
2
3
4
5
6
7
GND

14
1

J SUFFIX
CERAMIC
CASE 632-08

N SUFFIX
PLASTIC
CASE 646-06

D SUFFIX
SOIC
CASE 751A-02

ORDERING INFORMATION

54LSXXJ	Ceramic
74LSXXN	Plastic
74LSXXD	SOIC

5V supply

The gate can sink more current than it can source!

GUARANTEED OPERATING RANGES

Symbol	Parameter		Min	Typ	Max	Unit
V_{CC}	Supply Voltage	54 74	4.5 4.75	5.0 5.0	5.5 5.25	V
T_A	Operating Ambient Temperature Range	54 74	−55 0	25 25	125 70	°C
I_{OH}	Output Current — High	54, 74			−0.4	mA
I_{OL}	Output Current — Low	54 74			4.0 8.0	mA

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

Symbol	Parameter		Limits			Unit	Test Conditions	
			Min	Typ	Max			
V_{IH}	Input HIGH Voltage		2.0			V	Guaranteed Input HIGH Voltage for All Inputs	
V_{IL}	Input LOW Voltage	54			0.7	V	Guaranteed Input LOW Voltage for All Inputs	
		74			0.8			
V_{IK}	Input Clamp Diode Voltage			−0.65	−1.5	V	V_{CC} =	
V_{OH}	Output HIGH Voltage	54	2.5	3.5		V	V_{CC} = MIN, I_{OH} = MAX, V_{IN} = V_{IH} or V_{IL} per Truth Table	
		74	2.7	3.5		V		
V_{OL}	Output LOW Voltage	54, 74		0.25	0.4	V		MIN
		74		0.35	0.5	V	I_{OL} = 8.0 mA	per Truth Table
I_{IH}	Input HIGH Current				20	µA	V_{CC} = MAX, V_{IN} = 2.7 V	
					0.1	mA	V_{CC} = = 7.0 V	
I_{IL}	Input LOW Current				−0.4	mA	V_{CC} = MAX, V_{IN}	
I_{OS}	Short Circuit Current (Note 1)		−20		−100	mA	V_{CC} = MAX	
I_{CC}	Power Supply Current Total, Output HIGH Total, Output LOW				2.4 6.6	mA	V_{CC} = MAX	

Note 1: Not more than one output should be shorted at a time, nor for more than 1 second.

2V is the smallest voltage to be accepted as logic '1'

0.8V is the largest voltage to be accepted as logic '0'

Minimum voltage at a logic '1' output

Maximum voltage at a logic '0' output

Figure 2: Datasheet for the 74LS04 Hex Inverter logic gate circuit (Copyright of Motorola, used with permission).

Outputs must not be shorted!

This is not just a problem for larger digital systems that rely on actual physical wiring, it also gets tricky when routing connections on a digital logic chip itself. The question is why do we have to abide by the rule that we are not supposed to connect two outputs of logic gates? The reason will become clear if we look at the internals and the datasheet of a digital integrated circuit that we have been using. Internally, a logic inverter consists of two electronic switches, one that connects the output to the positive voltage V_{CC} if the input is zero and another one that connects the output to GND if the input is one. This configuration is called a 'totem-pole output' because the switches are 'stacked,' one on top of the other. While both switches are never both on at the same time, there is always one of these switches conducting. If you connect two of these outputs together, everything will be fine as long as both outputs have the identical logic value, because then the same switches are active in both gates. However, if one gate outputs a zero and the other one outputs a one, the two logic signals will "mix". Looking at the datasheet in Figure 2, the 74LS04 can sink more current than it can source, meaning that the gate with the output of one will be shorted by the gate having a zero at the output. The resulting voltage level will be close to zero as well.

There are multiple issues that arise from this situation. For one, the electronic switch to GND will short the one connecting to V_{CC}, pulling the largest amount of current possible (I_{OL} in the datasheet), which in turn leads to high power dissipation and heating of the chip. The datasheet in Figure 2 specifies that "Not more than one output should be shorted at one time nor for more than one second". While this applies to a "real" short circuit, not one via an electronic switch of another gate, it is not a good circuit design practice and in the long run might lead to permanent damage of the electronic chip. This is why we are not allowed to connect two outputs of logic gates together.

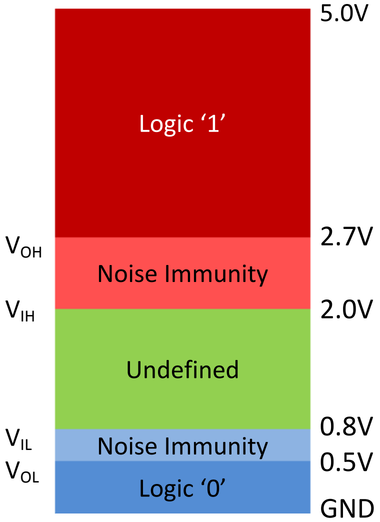
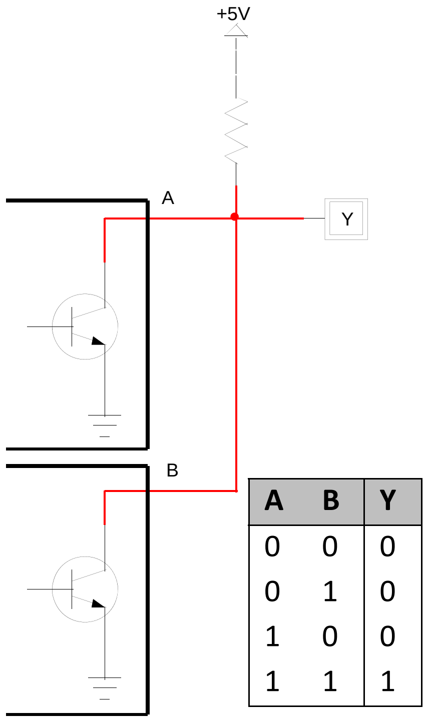


Figure 3: Logic level diagram for a TTL gate.

The other issue that arises when connecting two outputs together is that even if damage to the chip can be avoided, the voltage at that common point might be at a level that is outside of the voltage ranges for logic zero or logic one. The datasheet in Figure 2 defines four voltage values: V_{OL}, V_{IL}, V_{IH} and V_{OH}. V_{OL} is the typical voltage measured at a logic zero output (logic low) and V_{OH} the typical voltage measured at a logic one (logic high) output. V_{IL} is the highest voltage acceptable at an input pin to be interpreted by the logic gate as a logic low or zero input signal. Any voltage that is larger than V_{IL} does not guarantee correct functionality of the gate; the signal might be interpreted as logic low or as logic high. This depends on the manufacturing tolerances of the devices and even on the temperature of the chip itself. Only if the input voltage is above V_{IH} the gate will interpret the input as logic high or one. Figure 3 shows a diagram indicating the relation between the four voltage values for a TTL gate like the one shown in Figure 2. Thus, when connecting two outputs, it might happen that the output voltage falls within the "undefined" region and inputs of gates connected to this common point might not be able to correctly interpret that voltage value. This is another reason not to connect outputs of "regular" logic gates together.

How can we solve this problem of not being able to simply combine logic gate outputs? One solution is to "break up" the short circuit by removing the electronic switches that connect to V_{CC} from all the gate outputs that are supposed to be connected together. Gates that do no longer have the switch to V_{CC} are called "open collector" (or "open drain", depending on the transistor technology used). This indicates that the transistor switch connecting to GND is only wired to the output and nowhere else. How can we still achieve a logic one signal at the output? We have to replace the switch to V_{CC} with a resistor of such a large value that it is "safe" to short it to GND without having it heat up significantly, but small enough to ensure that gate inputs connected to this resistor will "see" a voltage that is above V_{IH}. This resistor is called a "pull-up" resistor and has a typical value of a few hundred ohms for low-voltage systems (< 5V) up to a few kilo-ohms.



A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

Figure 4: Open-collector configuration, realizing a "wired-AND" at the common connection point.

Now that we are able to connect outputs together, what did we gain by doing this? Let's look at two open-collector outputs that are connected together. If both output a zero, meaning that both output switches connect to GND, the resistor will be shorted to GND and the combined output will be zero. As long as one of the gates outputs a zero, the resistor will be shorted and the combined output will remain zero. Only if both gates output a one, the resistor can supply V_{CC} to the combined output as shown in Figure 4. We can now set up a truth table, which shows us that the simple connection of two open collector gate outputs performs a logic AND operation. We call this a "wired-AND". This way, we can save a complete logic gate and still get the same functionality.

Both the Arduino microcontroller and the gate array chip in the Digilent ADK can be programmed to have their output pins act like open collector outputs. It is possible to "disconnect" the V_{CC} switches and if desired connect internal pull-up resistors instead. No manual wiring is required to accomplish this.

What about the multiplexer? The multiplexer requires an OR gate at the output, yet the open collector combination only realizes an AND. We can use our knowledge how to change a sum-of-product implementation into a product-of-sums or we can apply deMorgan's rule: A + B = (A'B')' If we intend to realize an OR gate using deMorgan's rule, we have to invert the signals going to the wired-AND and invert the combined output signal again. Since this last inversion would defeat the purpose of a wired connection, the missing inverter is often shifted to the inputs that the combined output is connected to. The combined output is then called a "bus" with an "active low" signal, indicating that the signal should be inverted at the input nodes. A prominent example of an open collector bus is the I2C bus, which is used to control a wide variety of integrated circuits.

The drawback of an open-collector bus is the selection of the pull-up resistor: Too large of a value combined with the cable capacitance will make the bus slow and too small of a value will cause unnecessarily high power consumption. The original idea of having separate electronic switches to both GND and V_{CC} actually had its virtues. To prevent the short circuits from happening at connected outputs, we could simply switch off both of the electronic switches if that gate does not contribute any signal to the common output. A gate that has both of the electronic switches disconnected from the output is in a so-called "high impedance" or "high-Z" state, which is different from zero or one. This "disconnected" state is a third state, which is why gates that allow disconnecting the output are called three-state gates. The third state is then indicated by the letter 'Z'.

Three-state gates work just like regular logic gates, except that they have a (output) enable input, labeled EN, which controls if the output switches are both connected or disconnected. These gates are ideal for signal multiplexing, since they can replace the AND gates as signal selectors in the multiplexer and do not need an OR gate, since only one output will be enabled for a particular combination of select lines. However, making sure that only one output is enabled is critical for the correct operation of such a multiplexer, since otherwise a "data conflict" will occur on the common output, similar to the case of "regular" output gates.

The three-state outputs have become a standard in digital circuit design and both the Arduino chip and the ADK chip come with three-state outputs. When they are being powered on, all pins will be disabled by default and they have to be enabled to become an output. This also allows us to switch pins between inputs and outputs. If we want a pin to act as an output, all we are doing is enabling the three-state output.

Task 2-1: Gates with Common Outputs.

In this task we will check what really happens when we connect two outputs of "regular" inverters together. We already expect a data conflict to happen, but let's see how the voltage levels behave in a real-world scenario. Insert the 74LS04 integrated circuit into your breadboard and connect its V_{CC} and GND pins (IC Pins 14 and 7). Connect Arduino output Pin 2 to Pin 1 of the 74LS04. This will be the input A. Next, connect Arduino Pin 3 to Pin 3 of the 74LS04. This will be the input B. Set up Arduino Pin 7 to be an input and connect Arduino Pin 7 to IC Pin 2 (output Y). Finally, set up Arduino Analog In A0 (Pin 14) to serve as a voltmeter and connect it to IC Pin 2, in parallel with the previous connection. Finally, connect IC Pin 2 and IC Pin 4 as shown in Figure 5. For all four possible input combinations, record the output voltage in a table. How do the voltages compare with the values in the datasheet for V_{OL}, V_{IL}, V_{IH} and V_{OH}?

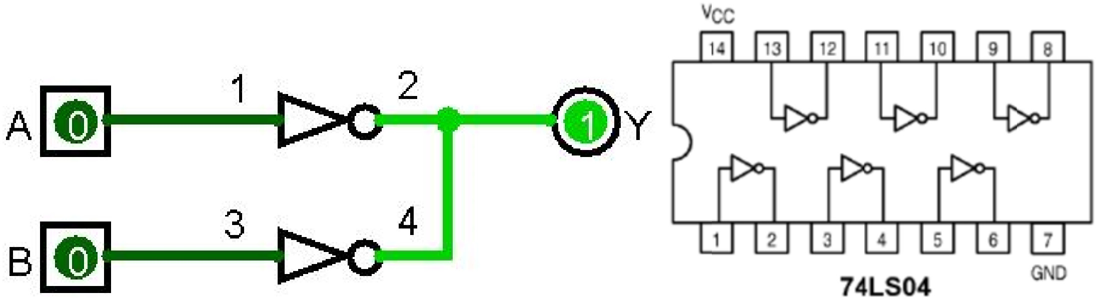


Figure 5: Schematic diagram for the circuit testing the effect of connecting two inverter outputs together. The right hand image shows the pin-out diagram for the 74LS05 integrated circuit (Copyright of Motorola, used with permission).

Task 2-2: Build and Test an Open-Collector Buffer Circuit

In Task 2-1, we verified that by connecting the outputs of two inverters together it creates a data conflict under certain input conditions. Now we replace the 74LS04 with the open collector version that is missing the transistor to V_{CC}. This logic gate is the 74LS05. You can simply swap one of the 74LS04 gates for a 74LS05, because V_{CC} and GND are connected to Pin 14 and Pin 7 as well. You can also leave Arduino Pin 2 connected to IC Pin 1, which is our A1 input signal. Now, connect Arduino Pin 3 to IC Pin 3, which will serve as our A2 input signal as shown in Figure 6. Connect the Arduino logic input Pin 7 to IC Pin 2 (output Y). In addition, set up Arduino Analog In A0 (Pin14) and connect it to IC Pin 2, in parallel with the previous connection to be able to measure the voltage at output Y. Finally, connect the two outputs of the IC together (Pins 2 and 4) and connect a 1000Ω resistor (color code brown-black-red) to Vcc +5V. Record the voltage and logic level of the output for all input combinations. What wired logic does the circuit perform (AND, OR, etc.)? Does the chip give the proper output voltages even though its outputs are connected together?

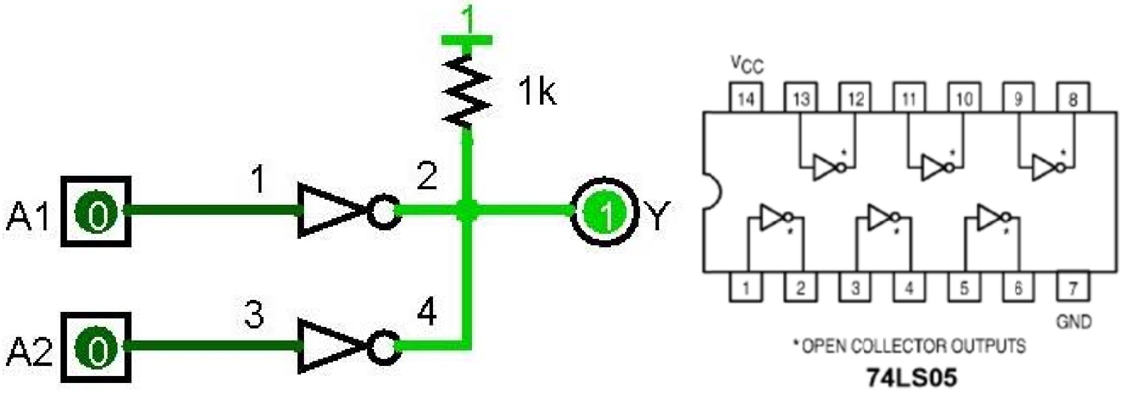


Figure 6: Schematic for the open-collector buffer circuit. The right hand image shows the pin-out diagram for the 74LS05 integrated circuit (Copyright of Motorola, used with permission).

Task 2-3: Build and Test a Three-State Buffer Circuit

In this task we will build a circuit similar to that in Task 2-3, but using three-state buffers instead of “regular” logic inverters. For this experiment, we should only use a TTL IC, such as the 74LS126 chip. DO NOT use a CMOS IC, such as the 74HC126 chip. Place the IC on the breadboard and wire it as follows: First, connect V_{CC} to Pin 14 of the IC and GND to Pin 7 of the IC. Then, wire Arduino Pin 2 to IC Pin 2 (input A1), Arduino Pin 3 to IC Pin 1 (EN1), Arduino Pin 4 to IC Pin 5 (input A2) and Arduino Pin 5 to IC Pin 4 (EN2). This sets up input signals in the sequence that is shown in Figure 7. Connect the Arduino logic input Pin 7 to IC Pin 3 (output Y). In addition, set up Arduino Analog In A0 (Pin14) and connect it to IC Pin 3, in parallel with the previous connection to be able to measure the voltage at output Y. Finally, connect the two three-state buffer outputs together by connecting a wire between IC Pin 3 and IC Pin 6.

The three-state buffers shown connected together in Figure 6 have two inputs each. The enable (EN) input determines whether the device is in the active state or in the inactive high-impedance state. If EN1 = 1 while EN2 = 0, the output is the same as the A1 input. If both EN1 and EN2 equal 0, both devices are in the high-impedance state and we cannot make a statement about the value of the output voltage at Y. This might cause the Arduino A0 input to fluctuate, so be aware of this when you record your measurements and observations in a table in your lab report. Using your table, answer the following questions: If the enable, EN, of one of the buffers is 0, does changing the input A of that gate change the output? If the enables of both gates are 0, what output do you get? If both EN values are 1, what outputs do you get for all combinations of A1 and A2? Careful: If you leave both EN inputs at 1 and you have different values for A1 and A2, the IC can heat up because of the data conflict, because it will act like the circuit in Task 2-1.

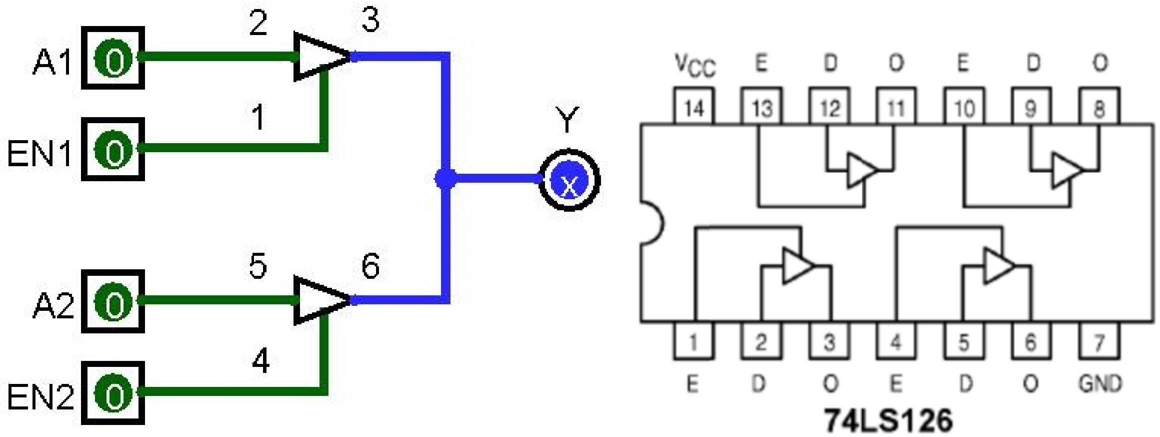


Figure 7: Schematic diagram of the three-state buffer circuit used in Task 2-5. The right hand diagram shows the pin-out of the three state buffer 74LS126 (Copyright of Motorola, used with permission).

Task 2-4: Take a Photo of your Completed Circuit

Before you rip everything apart, take a photo of your completed circuit and attach it to the lab template. This will serve as a replacement for the “attendance stamp” that the in-person students have to get on their paper lab submission.