LABORATORY #2
HALF-ADDER IMPLEMENTATION

Austin Guerrero
ECE 221 Digital Logic Design | Fall 2026 | Dr. Sami Khorbotly
Department of Electrical and Computer Engineering
Valparaiso University

ABSTRACT

For this lab, I took the NOR-only half adder from Design Project #1 and built it on a breadboard using two 74LS02 quad NOR chips. Two switches fed the inputs, two LEDs showed the outputs, and all four input combinations were tested. An oscilloscope was then used to measure how long it takes a change on input Y to show up at the SUM output. The circuit worked for every case, the measured delay was 25 ns, and the whole build cost just under four dollars.

♦ INTRODUCTION

The objective of this lab is to implement the half adder from Design Project #1 using small scale integration (SSI) circuits while using as few integrated circuits (ICs) as possible. Simulations are nice, but a real circuit has to deal with real wires, real chips, and the occasional loose jumper.

A half adder adds two one-bit numbers, X and Y, and produces two outputs. SUM is the exclusive OR of the inputs, and COUT (carry out) is only a 1 when both inputs are 1. Since the NOR gate is a universal gate, the entire circuit can be built from one gate type. Standard AND, OR, and inverter chips would each only cover part of the logic; sticking with NOR gates keeps everything on fewer chips.

♦ PRE LAB

The design was drawn in OrCAD Capture before lab (Fig. 1 in the Appendix). It uses five two-input NOR gates split between two 74LS02 chips, U1 and U2. Each gate does the following:

i. **U1A: (X+Y)'**
ii. **U1C: (X+X)' = X'**
iii. **U1D: (Y+Y)' = Y'**
iv. **U2A: (X'+Y')' = XY = COUT**
v. **U1B: ((X+Y)'+XY)' = (X+Y)(X'+Y') = SUM**

The final SUM expression is the same POS form from Design Project #1, so the hardware and the earlier simulation agree with each other. U1 uses all four of its gates, while U2 only needs one; the other three are just along for the ride.

♦ PROCEDURE

The steps below follow the Laboratory #2 handout in the ECE 221 Laboratory Manual. The schematic and oscilloscope capture can be found in the Appendix.

Part A: Construction and Testing

I. Two 74LS02 chips were placed on the breadboard. Pin 14 of each chip was connected to +5 V, and pin 7 was connected to ground.

II. The five NOR gates were wired according to the pre-lab schematic in Fig. 1.

III. Inputs X and Y were connected to a DIP switch, with each input pulled up to +5 V through a 10 kΩ resistorr. An open switch (down) gives a logic 1, and a

closed switch (up) pulls the input to ground for a logic 0.

IV. SUM and COUT each drove a green LED through a 180 Ω current-limiting resistor.

V. All four input combinations were tested, the LED states were written down in Table 1, and the working circuit was demonstrated to the instructor.

Part B: Propagation Delay

I. X was set to logic 0 with its switch, and Y was unplugged from its switch.

II. The function generator was set to output a 100 kHz square wave. Before it touched the circuit, the signal was checked on the oscilloscope to make sure it swung from 0 V to +5 V and not −2.5 V to +2.5 V; the lab manual makes it very clear that the second option ends with starting over.

III. After checking the signal, the square wave was connected to the Y input and the half-adder ground.

IV. Channel 1 was placed on Y and channel 2 on SUM. Both channels were set to 2 V/div with a 100 ns/div time base, and the scope triggered on the rising edge of channel 1.

V. Cursors were lined up on the rising edge of each signal to measure the delay (Fig. 2).

Part C: Cost Analysis

I. Every part on the board was counted and priced using the distributor price list in the lab handout (Table 2).

♦ RESULTS

Functional Testing

Every input combination matched the half-adder truth table from Design Project #1, so the circuit was functionally verified. Table 1 shows the results, where a lit LED stands for a logic 1.

Table 1. Measured Half-Adder Outputs

X	Y	SUM	COUT
0	0	0 (off)	0 (off)
0	1	1 (on)	0 (off)
1	0	1 (on)	0 (off)
1	1	0 (off)	1 (on)

Propagation Delay

With X held at 0, gate U1C outputs a constant 1, which locks COUT (U2A) at 0. That leaves SUM equal to Y, and the signal only travels through two gates: U1A, then U1B. Cursor a on the rising input edge read 20.00 ns, and cursor b on the SUM output read 45.00 ns, which gives a propagation delay of 25.00 ns. That works out to about 12.5 ns per gate, which is right in line with the typical switching speed of LS-series TTL chips. In other words, the circuit could add a new pair of bits roughly 40 million times per second; my fingers on the switches were the real bottleneck.

Both traces show some overshoot and ringing right after the edge. This most likely comes from the probe ground leads and breadboard jumper wires acting like small inductors. The ringing dies out within a couple hundred nanoseconds, and the output settles near 3.5 V, which is still a valid TTL high, so the logic is not affected.

Cost

The total parts cost is $3.98, and the DIP switch alone is more than half of it; the chips doing all the actual work cost less than the switch. The breakdown is in Table 2.

Table 2. Parts Cost

Part	Qty	Unit	Total
SN74LS02N Quad NOR	2	$0.62	$1.24
5 mm LED	2	$0.073	$0.15
10-Pos. DIP Switch	1	$2.27	$2.27
10 kΩ Resistor	2	$0.08	$0.16
180 Ω Resistor	2	$0.08	$0.16
Total			**$3.98**

♦ QUESTIONS

1. Why is it important to try and minimize the number of ICs used?

Cost is the obvious reason, but it is not the only one. Every IC needs its own power and ground connections, takes up space on the board, and draws supply current whether its gates are doing anything or not More chips also means more wires, and every wire is another chance for a mistake that has to be hunted down during debugging. Fewer connections make a circuit more reliable, and when a product is built thousands of times, saving even one chip per board adds up quickly.

2. How many ICs would be needed using AND OR and INVERTER ICs and how would this change the cost?

In SOP form, SUM = X'Y + XY' and COUT = XY. SUM needs two inverters, two AND gates, and one OR gate, while COUT needs a third AND gate. That takes one 7404, one 7408, and one 7432, so three ICs instead of two. The IC cost goes from $1.24 to $0.62 + $0.62 + $0.68 = $1.92, which is $0.68 more, or about 55% more spent on chips. The total cost would climb from $3.98 to $4.66, and the extra chip brings more wiring with it.

♦ CONCLUSION

The NOR-only half adder from Design Project #1 was built with two 74LS02 chips and worked correctly for all four input combinations. The measured propagation delay from Y to SUM was 25 ns through two gate levels, and the total parts cost was $3.98. Compared to an AND/OR/INVERTER version, the NOR design saves a full chip, cuts the IC cost by about a third, and leaves fewer wires to get wrong. It takes a bit more thinking on paper, but that effort pays for itself once the circuit is on the breadboard.

• ACKNOWLEDGEMENTS

The Objective, Introduction, and Procedure were given and paraphrased from Dr. Sami Khorbotly in the ECE 221 Laboratory Manual, as referenced in the Procedure section.

APPENDIX

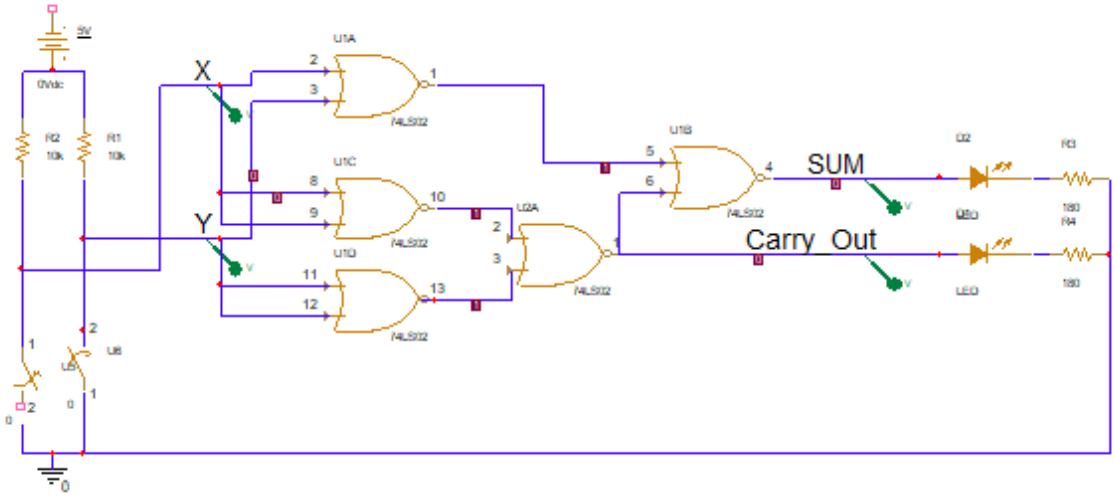


Fig 1. ***Half-Adder Pre-Lab Schematic (NOR Only, Two 74LS02 ICs)***



Fig 2. ***Oscilloscope Capture of Y to SUM Propagation Delay (Δ = 25.00 ns)***

APPENDIX

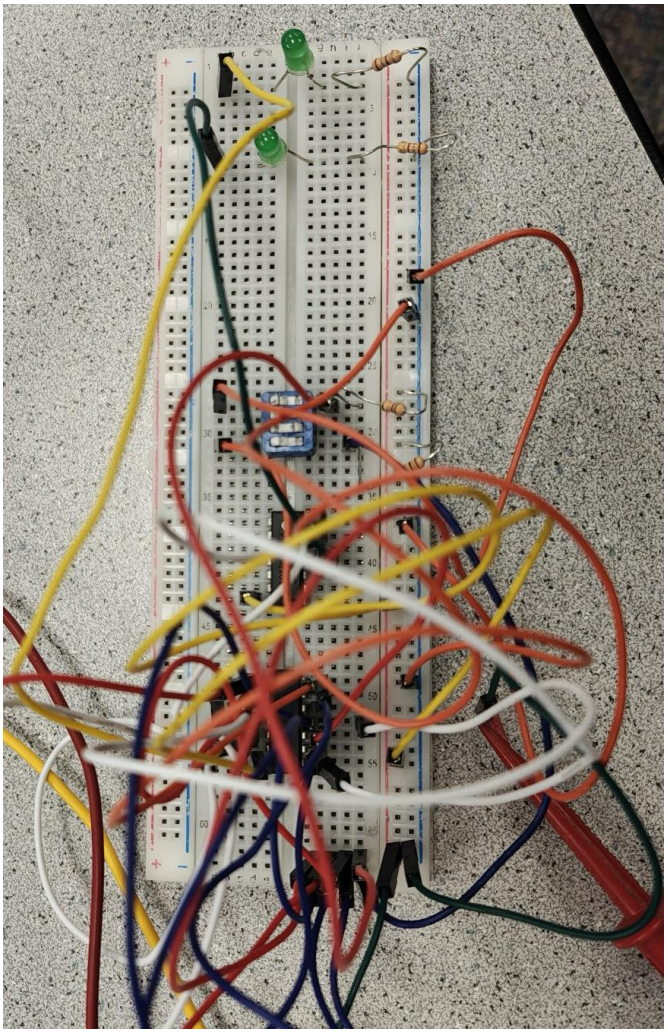


Fig 3. ***Breadboard Implementation of the Half Adder***

"I have neither given nor received, nor have I tolerated others' use of unauthorized aid."

Austin Guerrero