Lab 8

Lab 8 assignment is due on Friday, April 10, by 9pm in Blackboard.

Please ask all questions about this assignment during the office hours or post questions on piazza.

This lab requires to assemble a digital circuit on a prototyping board. This requires access to a hardware lab, which is not possible at the moment. Therefore, we will not be assembling any hardware at this time. Instead, please carefully read the provided lab materials and complete and submit the lab worksheet provided at the end of this page.

Combinational logic implementation with TTL

In this lab, you will implement the circuit you designed last week in Lab 7 as a working prototype on your protoboard. The circuit must be implemented using NAND or NOR gates only, with as few inverters as possible. (Yes, you can use NOT gates as needed, but try to use as few as possible.) Your circuit must implement the specification given in Lab 7 and generate two output signals "Accept coin" (A) and "Paid in full (dispense Product)" (P).

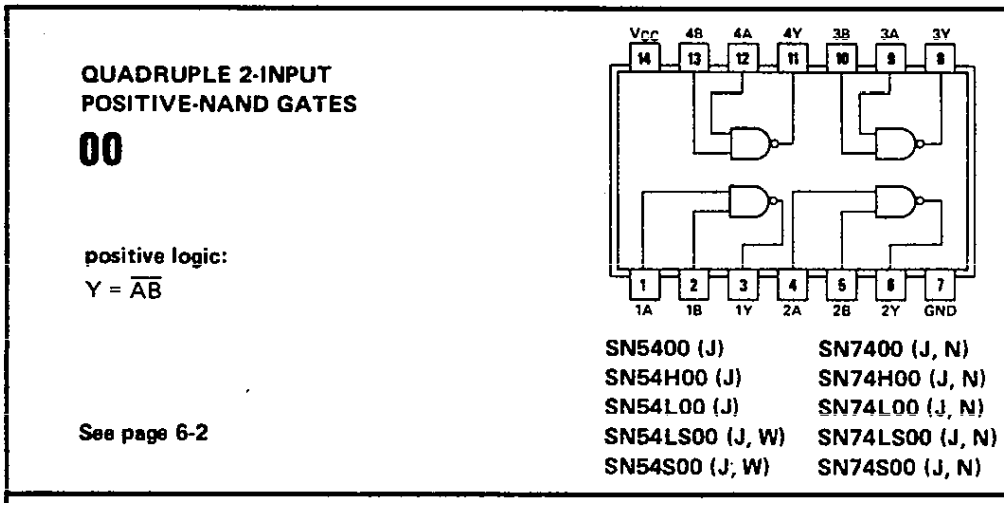
- A = 1, when the coin that has just been inserted should be accepted. A = 0, when the coin that has just been inserted should be rejected.
- P = 1, when the product should be dispensed. P = 0, when the product should not be dispensed

Instructions

1. Use this lab worksheet to draw your truth table and the layout of your TTL DIPs before implementing them on the protoboard.
    a. Your TTL DIPs, LEDs, and DIP switches should all be across a channel separating two blocks on the breadboard, just as you did in Lab 6.
    b. Try to layout your design for minimal clutter. This will not only help you debug if problems arise, but it will also make it easier to grade.
    c. Label each TTL DIP with a letter (e.g., A, B).
    d. Label each used pin on your TTL DIPs with the pin number (1,2,...14). This will help when you go to wire the actual circuit.
    e. Only labeling is required, but showing some wiring may be useful for you when assembling the circuit. In fact, you do not need to show entire wiring, just signal labeling may be sufficient.
2. Label your NAND or NOR Altera Quartus circuit diagram from Lab 7. Label each input and output with a DIP letter and pin number as a schematic for how you will connect the chips on your protoboard.
    a. The TTL chipset you received allows you to use only NOT, 2-input NAND, 2-input NOR, 3-input NAND, 3-input NOR, and 4-input NAND gates. Make sure you can implement your circuit with just those gates.
    b. Use the datasheets below to choose which TTL DIPs you will use and to determine how you will select which pins to use.
3. Implement, test, and debug your circuit on the protoboard by using your TTL DIPs, switches, and LEDs.
    a. You will need to use at last 3 switches and 2 LEDs to demonstrate the functionality of your circuit. We recommend using more LEDs to help you debug.
    b. During your testing and demonstration, bring your worksheet and use the switches to send all possible combinations of inputs into your circuit. Keep track of whether the LEDs show the behavior you specified with your Boolean expressions or truth tables.

TTL DIP data

Use the datasheets below to find out which TTL DIP to use for the different types of gates (2-input NAND, 2-input NOR, 3-input NAND, etc.) and to determine the pin assignments for your schematic. The title in the top left of each datasheet tells you what types of gates are in the DIP. The schematic on the top right of each datasheet shows you how the inputs and outputs of the various gates are connected to the pins of the TTL DIP. The text on the bottom right tells you which labels on your TTL DIPs to look for. The different serial numbers represent slightly different implementations of the same basic design.



QUADRUPLE 2-INPUT POSITIVE-NAND GATES

00

positive logic:

$Y = \overline{AB}$

See page 6-2

SN5400 (J)	SN7400 (J, N)
SN54H00 (J)	SN74H00 (J, N)
SN54L00 (J)	SN74L00 (J, N)
SN54LS00 (J, W)	SN74LS00 (J, N)
SN54S00 (J, W)	SN74S00 (J, N)

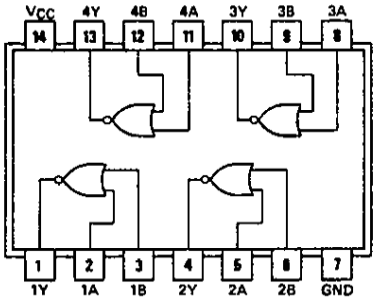
QUADRUPLE 2-INPUT POSITIVE-NOR GATES

02

positive logic:

$Y = \overline{A+B}$

See page 6-8



SN5402 (J)	SN7402 (J, N)
SN54L02 (J)	SN74L02 (J, N)
SN54LS02 (J, W)	SN74LS02 (J, N)
SN54S02 (J, W)	SN74S02 (J, N)

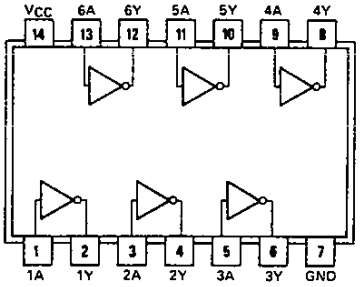
HEX INVERTERS

04

positive logic:

$Y = \overline{A}$

See page 6-2



SN5404 (J)	SN7404 (J, N)
SN54H04 (J)	SN74H04 (J, N)
SN54L04 (J)	SN74L04 (J, N)
SN54LS04 (J, W)	SN74LS04 (J, N)
SN54S04 (J, W)	SN74S04 (J, N)

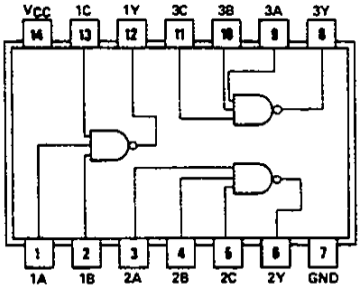
TRIPLE 3-INPUT POSITIVE-NAND GATES

10

positive logic:

$Y = \overline{ABC}$

See page 6-2



SN5410 (J)	SN7410 (J, N)
SN54H10 (J)	SN74H10 (J, N)
SN54L10 (J)	SN74L10 (J, N)
SN54LS10 (J, W)	SN74LS10 (J, N)
SN54S10 (J, W)	SN74S10 (J, N)

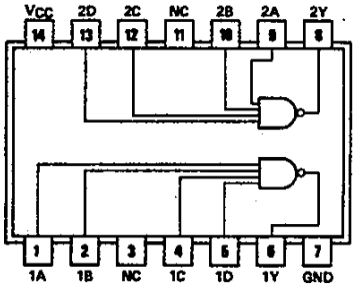
DUAL 4-INPUT POSITIVE-NAND GATES

20

positive logic:

$Y = \overline{ABCD}$

See page 6-2



SN5420 (J) **SN7420 (J, N)**
SN54H20 (J) **SN74H20 (J, N)**
SN54L20 (J) **SN74L20 (J, N)**
SN54LS20 (J, W) **SN74LS20 (J, N)**
SN54S20 (J, W) **SN74S20 (J, N)**

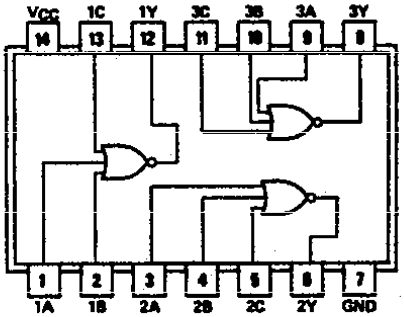
TRIPLE 3-INPUT POSITIVE-NOR GATES

27

positive logic:

$Y = \overline{A+B+C}$

See page 6-8



SN5427 (J, W) **SN7427 (J, N)**
SN54LS27 (J, W) **SN74LS27 (J, N)**

Refer to this video for tips for debugging this lab, courtesy of the ECE Student Advancement Committee.

What to Submit

- Completed lab worksheet.