م. أشرف بقيرات
مختبر المنطق الرقمي
لهندسة الحاسوب
والكهرباء

The Hashemite University

Computer Engineering Department

Logic Circuits Lab

Manual

Prepared by Eng. Ashraf Hasan Al-Bqerat

FEB 2014

م. أشرف سقيران
مختبر المنطق الرقمي
الهندسة الحاسوبية

The Hashemite University

Department of Computer Engineering

Logic Circuits Laboratory Manual

Prepared by Eng. Ashraf Hasan- Feb 2014

Table of Contents:

General Lab Rules

- Be **PUNCTUAL** for your laboratory session.
- Foods, drinks and smoking are NOT allowed.
- Sandals or open-toed shoes are NOT allowed.
- The lab timetable must be strictly followed. Prior permission from the Lab Supervisor must be obtained if any change is to be made.
- Experiment must be completed within the given time.
- Respect the laboratory and its other users. Noise must be kept to a minimum.
- Workspace must be kept clean and tidy at all time. Points might be taken off on student/group who fail to follow this.
- Handle all apparatus with care.
- All students are liable for any damage to equipment due to their own negligence.
- At the end of your experiment make sure to switch off all the instruments.
- Students are strictly PROHIBITED from taking out any items from the laboratory without permission from the Lab Supervisor.
- Students are NOT allowed to work alone in the laboratory.
- Please consult the Lab Supervisor if you are not sure on how to operate the laboratory equipment.
- Report immediately to the Lab Supervisor if any injury occurred.
- Report immediately to the Lab Supervisor any damages to equipment, hazards, and potential hazards.
- Please refer to the Lab Supervisor should there be any concerns regarding the laboratory.

Grading Policy

The total mark for this lab is distributed as follows

WorkSheet+ prelab + quizzes + attendance	40%
Mid-term Exam	20%
Final	40%

Guidelines for Writing Lab Report

The most effective way to acquire the practical skills in engineering studies is usually by experimenting in a laboratory. The process of experimentation involves organization, observation, familiarization with various pieces of equipment, working with others, writing, and communicating ideas and information. These are the skills required of an engineer.

In a practical situation, such as that in the industry or university research, experiments are designed for the purpose of clarifying research questions or conflicting theories by means of collecting a series of data. The conclusions drawn from that data can be used to validate a theory or sometimes to develop a theory that explains the behavior of an engineering object. The report for this kind of experiments must includes an introduction to the topic and purpose of the experiment, the theory, method, procedure, equipment used in the experiment, the data presented in an organized manner, and the conclusions based on the data gathered.

In engineering education, lab experiments are usually designed to enhance the understanding in engineering topics. Students are supposed to "dirty their hand" in preparing the experiment setup, organize the experiment flow, and learn to observe the salient features as well as to spot any unexpected occurrence as part of the training to acquire the practical skill to become an engineer. Although the introduction and the procedure are usually given in the lab handouts, students should practice writing a proper lab report which includes all the necessary sections, targeting at a reader who does not have any prior knowledge about the experiment. This is to develop the skill in documenting the laboratory work and communicating that experience to others. This write-up gives some guidelines on what to write in each section in preparing laboratory reports for engineering curricula.

Title Page

The title page should contain the title of the experiment, the code and title of the course, the name of the writer, the date when the experimental work was performed, submission date of the lab report, the name of lecturer for whom the report is prepared for, Day's of the lab, name of the experiment, number of the experiment.

Introduction or Objectives

An introduction is necessary to give an overview of the overall topic and the purpose of the report. The motivation to the initialization of the experimental work can be included. Its content should be general enough to orientate the reader gracefully into the subject materials.

Theoretical Background

This section is to discuss the theoretical aspects leading to the experiment. Typically, this involves the historical background of the theories published in the research literature and the questions or ambiguities arose in these theoretical work. Citations for the sources of information should be given in one of the standard bibliographic formats (for example, using square brackets with the corresponding number [2] that points to the List of References). Explore this background to prepare the readers to read the main body of the report. It should contain sufficient materials to enable the readers to understand why the set of data are collected, and what are the salient features to observe in the graph, charts and tables presented in the later sections.

Depending on the length and complexity of the report, the introduction and the theoretical background may be combined into one introductory section.

Experimental Method, Procedure and Equipment

This section describes the approach and the equipment used to conduct the experiment. It explains the function of each apparatus and how the configuration works to perform a particular measurement. Students should not recopy the procedures of the experiment from the lab handout, but to summarize and explain the methodology in a few paragraphs.

Observations, Data, Findings, Results

The data should be organized and presented in the forms of graphs, charts, or tables in this section, without interpretive discussion. Raw data which may take up a few pages, and most probably won't interest any reader, could be placed in the appendices.

Calculations and Analysis

The interpretation of the data gathered can be discussed in this section. Sample calculations may be included to show the correlation between the theory and the measurement results. If there exists any discrepancy between the theoretical and experimental results, an analysis or discussion should follow to explain the possible sources of error.

The experimental data and the discussions may also be combined into one section, for example, under the heading called "Discussion of Experimental Results".

Conclusions

The conclusions section closes the report by providing a summary to the content in the report. It indicates what is shown by the experimental work, what is its significance, and what are the advantages and limitations of the information presented. The potential applications of the results and recommendations for future work may be included.

Appendices

The appendices are used to present derivations of formulae, computer program source codes, raw data, and other related information that supports the topic of the report.

List of References

The sources of information are usually arranged and numbered according to the order they are cited in the report. The reference materials may be entered in the following formats:

[1] Author, "Title of the book", 2nd edition, New York: Publisher, 1989.
[2] Author, "Title of the paper", Journal name, Vol. 2, No. 3, Jan 1990, pg. 456-458.
[3] Author, "Title of the paper", Proceedings of Conference 1991, pg. 5-6.
[4] Author, "Title of the thesis", Ph.D. thesis, Rice University, Houston, May 1973.

Lab Session 1
Logic Gates

1.1 Objectives:

1. Determine experimentally the truth tables for OR, AND, NAND, NOR, Inverter and XOR gates.
2. Wire and operate 2-input, 3-input and 4-input AND gates.
3. Wire and operate 2-input, 3-input and 4-input NAND gates.
4. Wire and operate the Inverter.
5. Wire and operate 2-input OR gate, 2-input XOR gate and 2-input NOR gate.
6. Use the logic gates to implement Boolean functions.

1.2 Reference Reading:

- Floyd, Digital Fundamentals, chapter 3.

1.3 Materials Needed:

- ICs: 7400,7402,7404,7410,7411,7421 & 7486

1.4 Basic Information:

Logic deals with only two normal conditions: logic "1" or logic "0." These conditions are like the yes or no answers to a question. Either an event has occurred (1) or it hasn't (0); and so on.
In Boolean logic, 1 and 0 represent conditions. In positive logic, 1 is represented by the term *HIGH* and 0 is represented by the term *LOW*. In positive logic, the more positive voltage is 1 and the less positive voltage is 0. Thus, for positive TTL logic, a voltage of 2.4 V is 1 and a voltage of 0.4 V is 0.

1.4.1 Logic Gates

a. The Inverter
The inverter (NOT circuit) performs the operation called inversion or complementation. The inverter changes one logic level to the opposite level. In terms of bits it changes a 1 to a 0 and a 0 to a 1.

Standard logic symbols for the inverter are shown in Figure 1. Part (a) shows the distinctive shape symbols, and part (b) shows the rectangular outline symbols. In the logic circuits lab, distinctive shape symbols are used.

(a) Distinctive Shape Symbol

(b) Rectangular Outline Symbol

Figure 1: Standard Logic Symbols For The Inverter

b. The AND Gate

The AND gate is one of the basic gates from which all logic functions are constructed. An AND gate can have two or more inputs.

The term gate is used to describe a circuit that performs a basic logic operation. The AND gate is composed of two or more inputs and a single output, as indicated by the standard logic symbols in Figure 2. Inputs are on the left, and the output is on the right in each symbol.

(a) Distinctive Shape Symbol

(b) Rectangular Outline Symbol

Figure 2: Standard Logic Symbols For The AND Gate Showing Two Symbols

c. The OR Gate

The OR gate is another of the basic gates from which all logic functions are constructed. An OR gate can have two or more inputs and performs what is known as logical addition.

(a) Distinctive Shape Symbol

(b) Rectangular Outline Symbol

Figure 3: Standard Logic Symbols For The OR Gate Showing Two Symbols

d. The NAND Gate

The NAND gate is a popular logic element because it can be used as a universal gate; that is, NAND gates can be used in combination to perform the AND, OR, and inverter operations.

(a) Distinctive Shape Symbol (b) Rectangular Outline Symbol

Figure 4: Standard Logic Symbols For The NAND Gate Showing Two Symbols

e. The NOR Gate

The NOR gate, like the NAND gate, is a useful logic element because it can also be used as a universal gate; that is, NOR gates can be used in combination to perform AND,OR, and inverter operations.

(a) Distinctive Shape Symbol (b) Rectangular Outline Symbol

Figure 5: Standard Logic Symbols For The NOR Gate Showing Two Symbols

f. The EXCLUSIVE-OR Gate

The EXCLUSIVE-OR gate are actually formed by a combination of other gates already discussed, because of their fundamental importance in many applications, these gates are treated as basic logic elements with their own unique symbols.

(a) Distinctive Shape Symbol (b) Rectangular Outline Symbol

Figure 6: Standard Logic Symbols For The XOR Gate Showing Two Symbols

1.4.2 Practical TTL Logic Gates

A popular type of IC is illustrated in Figure ٨.This case style is referred to as a *dual in-line package* (DIP) by IC manufacturers. This particular IC is called a 14-pin DIP IC. Just counterclockwise from the **notch** on the IC is pin 1. A **dot** on the top of the DIP IC is another method used to locate pin 1.

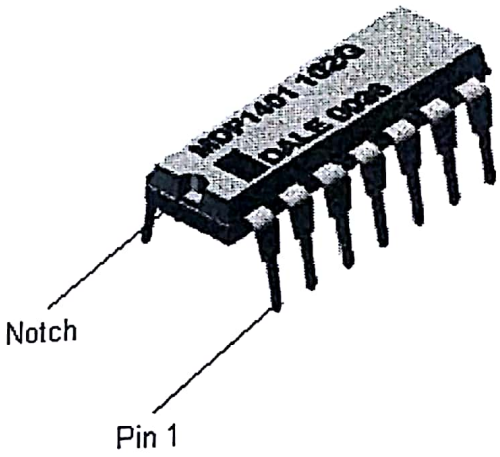


Figure 8

1.4.2.1 Part Number

Part number is divided into three sections:

- The prefix: the manufacturer's code.
- Core part number: which determine the technology"TTL or CMOS" . The device series and the function of a digital IC.
- The trailing letter(s) "the suffix" which is a code used by several manufacturers to designate the DIP.

For example, the Part Number of:

SN 74 LS 08 J

SN: stand for the manufacturer "Texas Instruments"
74: 7400 TTL series
LS: Low schottky type
08: function of a digital IC
J: Ceramic dual in line package

1.4.3 Integrated-Circuit Gates

Hex Inverter:
The 74LS04 hex inverter is consisting of six inverters in a 14-pin package.

AND Gates:
Several configurations of AND gates are available in IC form.
The 74LS08 has four 2-input AND gates (it is called a quad 2-input AND).
The 74LS11 has three 3-input AND gates (a triple 3-input AND).
The 74LS21 has two 4-input AND gates(a dual 4-input AND).

NAND Gates:
A variety of NAND gates are available, including:
The 74LS00 has four 2-input gates.
The 74LS10 has three 3-input gates.
The 74LS20 has two 4-input gates.
The 74LS30 has one 8-input gate.

OR Gates:
The 74LS32 has four 2-input OR gates.

NOR Gates:
74LS02 has four 2-input gates.
74LS27 has three 3-input gates.

Exclusive-OR Gates:
74LS86 has four 2-input XOR gates.

1.5 PreLab:

1- Obtain logic diagram, the truth table and needed ICs of the following function: $F(A, B, C) = \overline{A}.C + B\overline{C}$. Using AND,OR, and NOT gates.

Lab Session 2
Boolean Algebra, DeMorgan's Theorem & Karnaugh-map

2.1 Objectives:

1. Verify and Prove Boolean Laws.
2. Verify DeMorgan's Theorem.
3. Write sum-of-product (SOP) logic expressions for functions defined in given truth tables then implement these expressions by using the logic gates.
4. Use K-map technique to simplify a logic expression in SOP format.

2.2 Reference Reading:

- Floyd, Digital Fundamentals, chapter 4.

2.3 Materials Needed:

- ICs: 7400, 7404, 7408, 7410, 7411

2.4 Information Summary:

2.4.1 Boolean Expressions

When two or more logic gates are connected to perform a specified function, we have a gate network. Boolean algebra provides a concise way to express the operation of a gate network so that we can readily determine what the output will be for various combinations of input levels.
The form of a given expression determines the how many logic gates are used, what type of gates are needed, and how they are connected together.

2.4.2 Boolean Algebra

Boolean algebra consists of a set of laws that govern logical relationships. Unlike ordinary algebra, where an unknown can take any value, the elements of Boolean algebra are binary variables and can have only one of two values: 1 or 0. Symbols used in Boolean algebra include the overbar, which is the NOT or complement; the connective +, which implies logical addition and is read "OR"; and the connective '.', which implies logical multiplication and is read "AND." The dot is frequently eliminated when logical multiplication is shown. Thus $A.B$ is written AB.

The laws & theorems of Boolean Algebra are listed below:

Commutative Law: Rearranging the order of the expression

$A+B = B+A \qquad A.B = B.A$

Associative Law: Grouping terms or variables using parenthesis:

$A + (B + C) = (A+B) + C$

$A.(B.C) = (A.B).C$

Distributive Law: The process of ANDing a single variable over each term of an OR expression or factoring out a common term form an OR expression:

$A(B+C) = A.B + A.C$

$A + B.C = (A+B).(A+C)$

Absorption Law: $A+A.B = A \qquad A.(A+B) = A$

The Basic rules of Boolean Algebra are:

$A + 0 = A$	$A . 0 = 0$	$\overline{\overline{A}} = A$
$A + 1 = 1$	$A . 1 = A$	$A + AB = A$
$A + \overline{A} = 1$	$A.\overline{A} = 0$	$(A+B).(A+C) = A + BC$
$A + A = A$	$A.A = A$	$A+\overline{A}B = A + B$

In addition to the basic rules of Boolean algebra, there are two additional rules called De-Morgan's theorems that allow simplification of logic expressions that have an over bar covering more than one quantity. DeMorgan wrote two theorems for reducing these expressions. His first theorem is The complement of two or more variables ANDed is equivalent to the OR of the complements of the individual variables. Algebraically, this can be written as:

$$\overline{X.Y} = \overline{X} + \overline{Y}.$$

His second theorem is The complement of two or more variables ORed is equivalent to the AND of the complements of the individual variables. Algebraically, this can be written as:

$$\overline{X+Y} = \overline{X}.\overline{Y}$$

As a memory aid for DeMorgan's theorems, some people like to use the rule "Break the bar and change the sign." The dot between ANDed quantities is implied if it is not shown, but it is given here to emphasize this idea.

Many times in the application of Boolean algebra, we have to reduce a particular expression to its simplest form or change its form to a more convenient one to implement the expression most efficiently.

The reason that we are interested in simplifying Boolean expressions: is to use the fewest gates possible to implement a given expression.

2.4.3 SOP (Sum of Product) Expressions

The most commonly used format for writing logic expressions is a standard form called sum-of-product (SOP). SOP expressions can be quickly written from a truth table and they are easily implemented using a two-level (not counting inverters that may be needed) gate network. Conversely, these standard circuits that are used to implement SOP functions can be quickly analyzed just by inspection. A sum-of-product expression consists of two or more product (AND) terms that are ORed together. The SOP expression is obtained from a truth table by writing down all of the product terms (also called minterms) whose outputs are high for the desired function and then ORing them together. The resultant SOP expression can be directly implemented with either AND/OR or NAND/NAND circuit designs.

Examples of SOP format:-

- $AB + BCD$
- $A\overline{B}C + \overline{D}EF + AEG$
- $A + BC$

2.4.4 Karnaugh Mapping

The basic procedure for combinational logic circuit design is to develop first the truth table that defines the desired function and then from the table, write a simplified SOP expression. The expression can be simplified using various techniques (such as Boolean algebra, Karnaugh mapping, etc.). Karnaugh mapping is a simple and fast procedure for reducing SOP logic expressions and thereby also reducing the implemented circuit's complexity and cost. In Karnaugh mapping, the function is defined graphically. The relationships between the function's inputs and the output are plotted in a Karnaugh map (K map). This will be the same information that would be listed in the truth table for the function. The input variables must be labeled on the K-map in a very systematic fashion. If the K map is not properly labeled, the function cannot be correctly simplified and the resulting design will be wrong. With K mapping, the function reduction is accomplished by forming appropriate groupings of 1s in the output. Then identify the common input variables for the group and write the indicated product term. Karnaugh mapping can best be applied to functions with 5 or fewer input variables.

The effectiveness of algebraic simplification depends on your familiarity with all the laws, rules, and theorems of Boolean algebra and on your ability to apply them. Cleverness is often an important factor in algebraic simplification.

The Karnaugh Map (K map), on the other hand, is easier than algebraic simplification because it doesn't need a strong background of rules and laws like algebraic simplification.

2.5 Prelab

1- By using Boolean Algebra technique, simplify the following function as much as possible: $F(X, Y, Z) = \overline{X}.Y.Z + \overline{X}.\overline{Z}$. Then obtain the Logic diagram and the truth table for this function.

2- Use K-map to simplify the Boolean function F together with the don't care conditions d in Sum of products
$F(X, Y, Z, W) = \sum(3,5,7,13) + \sum d(6,11,14,15)$. Obtain the logic diagram for SOP.

3- Obtain the truth table of the following function:- $F(A, B, C) = A\overline{B} + \overline{ABC}$. Then determine the needed ICs and obtain the logic diagram if you use the NAND and inverter gates only to implement this function.

Lab Session 3
Multiplexers, Decoders

3.1 Objectives:

1. Check the functions of the multiplexers (MUXs) and the decoders.
2. use the multiplexer to implement a logic function
3. use the decoders to implement a logic function.

3.2 Reference Reading:

- Floyd, Digital Fundamentals, chapter 6, "Functions of Combinational Logic"

3.3 Materials Needed:

- ICs: 7400,7404, 7408, 7410,7411

3.4 Information Summary:

3.4.1 Multiplexers

A multiplexer (MUX) is a logic circuit that channels two or more input data lines to one output data line. A MUX is also called a data selector. The routing of particular data input to the output is controlled by the SELECT (or ADDRESS) inputs. Generally speaking, a MUX has N select inputs (address bits), data inputs, and one data output. For example, an-8-to-1 MUX has eight data inputs, three select inputs and one data output.

Multiplexers are widely used in digital and data communications systems. They can perform data selection, data routing, operation sequencing, parallel-to-serial conversion, waveform generation, and logic-function generation, Multiplexers make it possible for several streams of digital data to be sent over one physical cable is a system called TDM (time-division-multiplexing)

One useful application for MUXs is implementation of combinational logic functions directly from the truth table.

3.4.1.1 Internal Gating of the Multiplexer

The internal gating of the multiplexer (MUX) chips is very simple. The multiplexer, for example, consists of a number of AND gates, with enable/disable functions provided by control inputs. Only one AND gate is enabled at a time, allowing only one input to reach the output at a time.

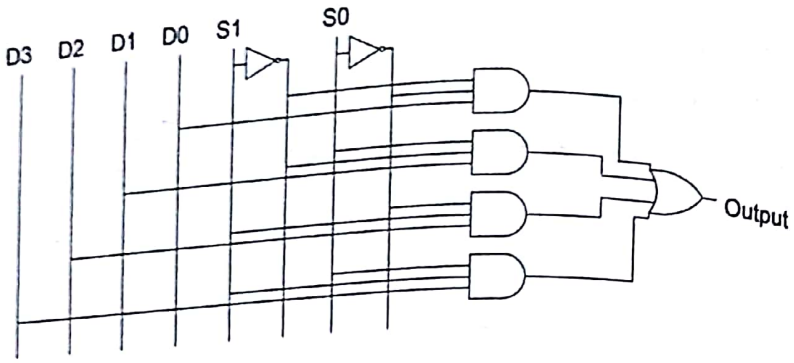


Figure 3.1 Internal Gating Of 4:1 MUX

3.4.1.2 The 74150 Sixteen-Input Data Selector / Multiplexer

The 74150 has sixteen data inputs and four data-select lines. In this case four bits are required to select and one of the sixteen data inputs ($2^4 = 16$). There is also an active-LOW $\overline{\text{STROBE}}$ input. On this particular device, only the complement of the output is available.

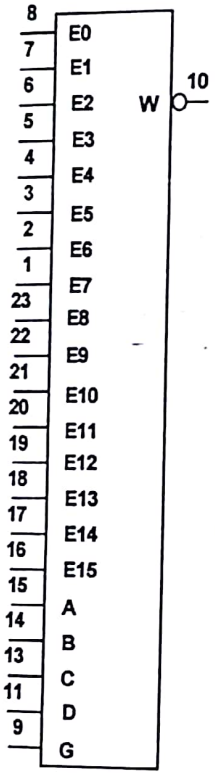


Figure 3.2: Logic Symbol of 74LS150

3.4.1.3 The 74151 Eight-Input Data Selector/Multiplexer

The 74151 has eight data inputs and, therefore, three data-select input lines. Three bits are required to select any one of the eight data inputs ($2^3 = 8$). A LOW on the $\overline{\text{STROBE}}$ input allows the selected input data to pass through to the output. Notice that the data output and its complement are both available. The Logic symbol is shown in Figure 3.3.

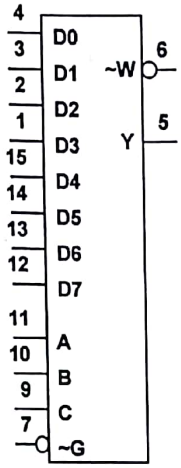


Figure 3.3: Logic Symbol Of 74LS151

3.4.1.4 The multiplexer as a Logic Function Generator

One of useful applications of the data selector/multiplexer is the generation of combinational logic functions in sum-of-products form. When used in this way, the device can replace discrete gates, can often greatly reduce the number of ICs, and can make design changes much easier.

To illustrate, an 8:1 MUX can be used to implement any specified 3-variable logic function if the variables are connected to the data-select inputs and each data input is set to the logic level required in the truth table for that function.

For example, if the function is 1 when the variable combination is $\overline{X}.Y.\overline{Z}$, the 2 input (selected by 010) is connected to a HIGH. This HIGH is passed through to the output when this particular combination of variables occurs on the data-select lines. An example will help clarify this application.

The circuit in figure 3.4 use a 74151 (8:1 MUX) to implement the function $F(X,Y,Z) = \sum(1,3,4,7)$

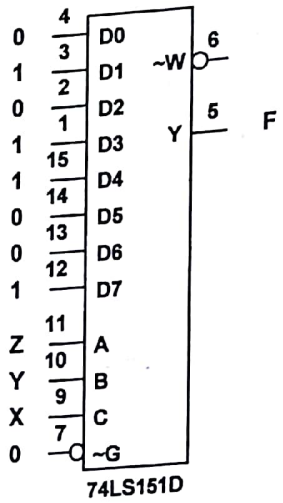


Figure 3.4: Use An 8:1 MUX To Implement A Logic Function

3.4.2 Decoders

A decoder is a multiple-input, multiple-output logic circuit that converts coded inputs into coded outputs, where the input and output codes are different. Then input code generally has fewer bits than the output code. The enable inputs, if present, must be asserted for the decoder to perform its normal mapping function. Otherwise, the decoder maps all input code words into a single, "disabled", output code word.

The decoder is a logic circuit that converts an N-Bit binary input code into M out lines such that each output will be activated for only one of the possible combination of inputs.

n : 2^n decoder has n control inputs and 2^n outputs. Depending on the value of the control inputs, only one of the output lines becomes *active*.

If an **active-HIGH** output is desired for each decoded number. The output of the decoded number will be HIGH and all of the other outputs are LOW.

If an **active-LOW** output is desired for each decoded number. The output of the decoded number will be LOW and all of the other outputs are HIGH..

3.4.2.1 Internal Gating of the Decoder

A 2-to-4 line decoder with an enable input constructed with NAND gates is shown in figure 3.4. All outputs are equal to 1 if enable input E is 1, regardless of the values of inputs A & B. When the enable input is 0, the circuit operates as a decoder as an Active-Low output. The truth table in table 3.1 lists these conditions.

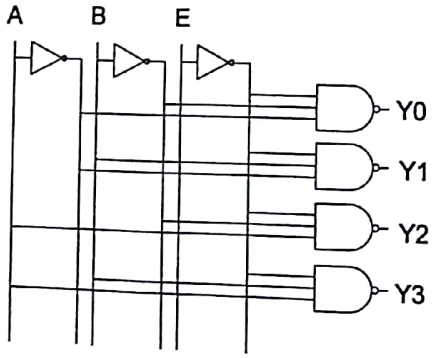


Figure 3.4:- 2:4 Low-Active Decoder With Enable Line

E	A	B	Y0	Y1	Y2	Y3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

Table 3.1: Truth Table Of A 2:4 Low-Active Decoder With Enable Line.

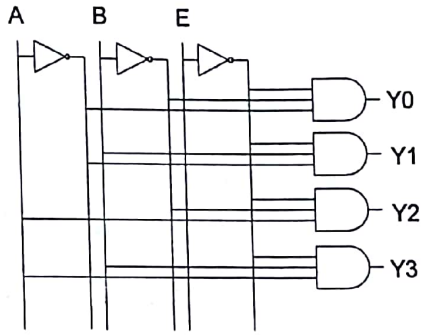


Figure 3.5:- 2:4 High--Active Decoder With Enable Line

E	A	B	Y0	Y1	Y2	Y3
1	X	X	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	1	0	0
0	1	0	0	0	1	0
0	1	1	0	0	0	1

Table 3.2: Truth Table Of A 2:4 High-Active Decoder With Enable Line.

3.4.2.2 The 74139 Dual 2-to-4 Decoder

Two independent and identical 2-to-4 decoders are contained in a single IC, the 74139. The logic symbol 74139 is shown in figure 5. Notice that the outputs and the enable inputs of the 74139 are active LOW. Most ICs decoders were originally designed with active-low outputs, since TTL inverting gates are generally faster than noninverting ones.

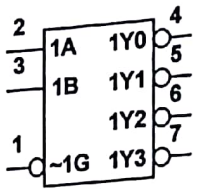


Figure 3.6:Logic Symbol Of 74LS139 – Section A

3.4.2.3 The 74138 3-to-8 Decoder

The 74LS138 is a commercially available IC 3-to-8 decoder whose logic symbol is shown in Figure 6. its truth table is shown in the data sheets of 74LS138. Like the 74139, the 74LS138 has active-low outputs, and it has three enable inputs (G1, $\overline{G2A}$, $\overline{G2B}$), all of which must be asserted for the selected output to be asserted.

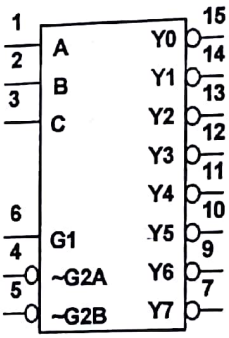


Figure 3.7: Logic Symbol Of 74LS138

3.4.2.4 The 74154 4-to-16 Decoder

The 74LS154 is a 4-to-16 TTL IC decoder. Its logic symbol is shown in figure 7. If the enable inputs is not activated (LOW on both inputs), then all sixteen decoder outputs will be HIGH regardless of the states of the four input variables.

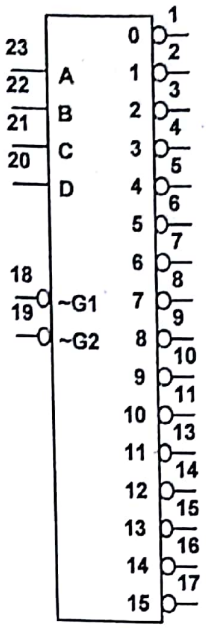


Figure 3.8: Logic Symbol Of 74LS154

3.4.2.5 The 7442 BCD-to-Decimal Decoder

The BCD-to-decimal decoder converts each BCD code word into one of ten possible decimal digit indicators. It frequently referred to as a 4-line-to-10-line decoder. The method of implementation is essentially the same as for the 4-line-to-16line decoder previously discussed, the logic is identical to that of the first ten decoding gates in the 4-line-to-16-line decoder.

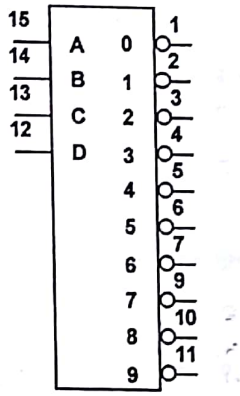


Figure 3.9: Logic Symbol Of 74LS42

3.4.2.6 Applications of the Decoders

Decoders are used in many types of applications. One example is computers for input/output selection. Computers must communicate with a variety of external devices called peripherals by sending and/or receiving data through what is known as input/output (I/O) ports. These external devices include printers, modems, scanners, external disk drives, keyboard, video monitors, and other computers. The decoder is used to select the I/O port as determined by the computer so that data can be sent or received from a specific external device.

Each I/O port has a number, called an address, which uniquely identifies it. When the computer wants to communicate with a particular device, it issues the appropriate address code for the I/O port to which that particular device is connected. This binary port address is decoded and the appropriate decoder output is activated to enable the I/O port.

3.4.2.7 The Decoder as a Logic Function Generator

As the multiplexer is used to generate a logic function. One of the useful applications of the decoder is the generation of combinational logic functions in sum-of-products form. When used in this way, the device can replace discrete gates, can often greatly reduce the number of ICs, and can make design changes much easier.

To illustrate, an 3:8 Decoder can be used to implement any specified 3-variable logic function if the variables are connected to the data-select inputs and all of the data outputs which represent the logic 1 are connected to NAND gate if the Decoder is a low-active output or connected to AND gate if the Decoder is a high-active output.

Another implementation; is connect all of the data output which represent to the logic 0 to an AND gate if the Decoder is low-active output or to NAND gate if the decoder is a high-active output.

For example, the circuit in figure 3.10 and 3.11 use the 3:8 Decoder IC; 74138 to implement the function $F(X,Y,Z) = \sum(1,3,4,7)$ by the two ways.

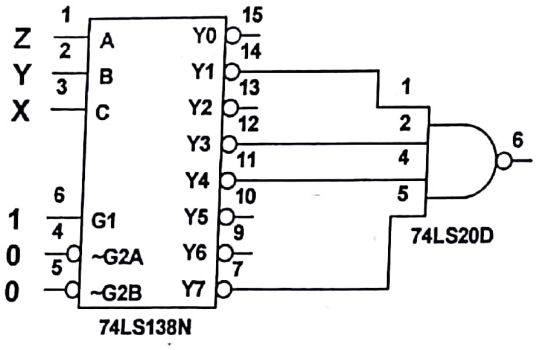


Figure 3.10: Use The NAND Gate To Implement A Function From A 3:8 Low-Active Output Decoder

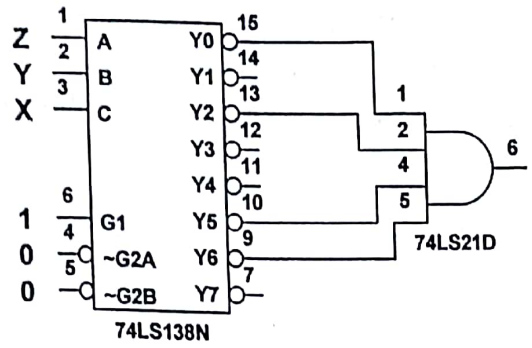


Figure 3.11: Use The AND Gate To Implement A Function From A 3:8 Low-Active Output Decoder

3.4.2.8 Use The Decoder To Implement More Than Logic Function

One of the useful applications of the decoder that you can implement more than logic function from the same decoder. For example, you want to implement two functions: $F(X,Y,Z) = \sum(1,3,4,7)$ and $G(X,Y,Z) = \sum(0,2,6)$. The circuit in figure 3.12 illustrates this principle.

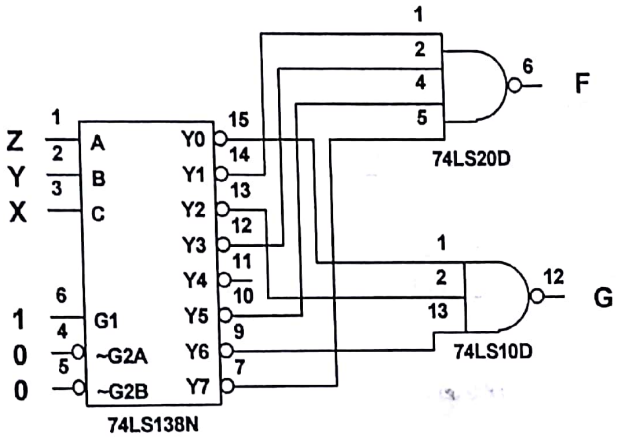


Figure 3.12: Implement Two Functions From The Same Decoder

3.5 Prelab

1- Use an 8:1 Multiplexer to implement the function $F(X,Y,Z,W) = \sum(0,2,4,6,9,10,13,14)$. Obtain the implementation table, logic diagram and required ICs to implement this function.

2- Show how a 3:8 decoder is used to determine the function. $F(X,Y,Z) = \sum(0,3,7)$

Lab Session 4
Adders and Comparators

4.1 Objectives:

1- introduces two important MSI circuits—the 4-bit adder and the 4-bit magnitude comparator.
2- Use a 4-bit parallel adder as subtractor.
3- Use 4-bit magnitude comparator.

4.2 Materials Needed:

- ICs: 7483, 7485, 7486, and 7404.

4.3 Reference Reading

- Floyed, Digital Fundamentals, chapter 6.

4.4 Information Summary:

4.4.1 Adders

Adders are very important in many types of digital systems in which numerical data are processed. An understanding of their basic operation is fundamental to a thorough grasp of digital systems concepts.

4.4.1.1 Full-Adder

The full-Adder accepts three inputs and generates a sum output and a carry output. So the basic difference between a full-adder and a half-adder is that the full-adder accepts an additional input, which allows it to handle input carries.

A logic symbol for a full-adder is shown in Figure 4.1 and the truth table in Table 4.1 shows the operation of a full adder.

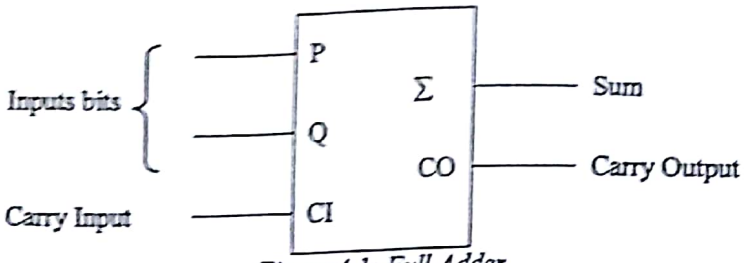


Figure 4.1: Full Adder

P	Q	CI	CO	Σ
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Table 4.1: Truth Table Of The Full Adder

4.4.1.2 Four Bit Parallel Adders

The block diagram of a four-bit binary parallel adder is shown in figure 4.2. Notice that the carry output of each adder is connected to the carry input of the next higher-order adder.

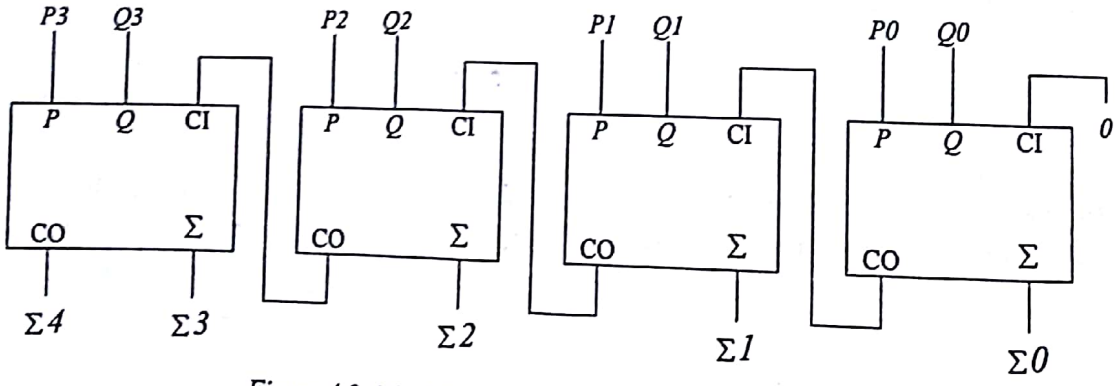


Figure 4.2: Block Diagram Of A 4 Bit-Parallel Adder

4.4.1.3 7483 4-Bit Parallel Adder

IC manufacturers produce several adders. One elementary arithmetic IC is the TTL 7483 4-bit binary full adder. The logic symbol of 7483 is same as the logic symbol of 4-bit parallel adder in figure 4.3.

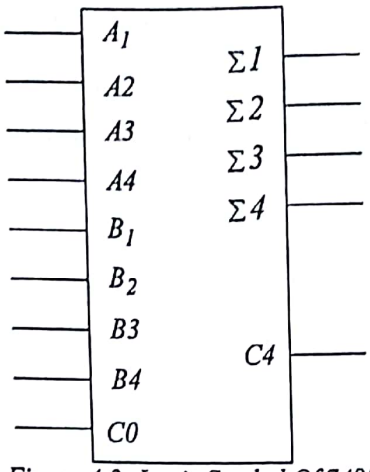


Figure 4.3: Logic Symbol Of 7483

Internally the 7483 IC adder is organized very much like the block diagram in Figure 4.2. The 7483 IC adder can by cascaded by connecting the C4 output of the first IC to the C0 carry input of the next 7483 IC. With two 7483 IC adders cascaded, an 8-bit binary adder is produced.

The 7483 IC adder can by cascaded by connecting the C4 output of the first IC to the C0 carry input of the next 7483 IC. With two 7483 IC adders cascaded, an 8-bit binary adder is produced.

Other 4-bit adders that function the same as the 7483 IC but have a different pin configuration is 74LS283. A more complex arithmetic chip is the 74LS181 IC and its relatives, the 74LS381, are described as arithmetic-logic units / function generators. These units perform many of the tasks of the ALUs in simple microprocessors and microcontrollers.

4.4.1.4 Over Flow

Fixed-point signed numbers are stored in most computers in the manner illustrated in Table 4.2. Positive numbers are stored in true form and negative numbers are stored in 2's complement form. If two numbers with the same sign are added, the answer can be too large to be represented with the number of bits available. This condition, called *overflow*, occurs when an addition operation causes a carry into the sign bit position. As a result, the sign bit will be in error, a condition easy to detect. When two numbers with the opposite sign are added, over-flow cannot occur, so the sign bit will always be correct. Example 1 illustrates overflow for 4-bit numbers.

4.4.2 Comparators:

The basic function of a comparator is to compare the magnitude of two quantities to determine the relationship of those quantities. In its simplest form, a comparator circuit determines whether two numbers are equal, or not.

The exclusive-OR gate can be used as a basic comparator because its output is a 1 if its two input bits are *not equal* and a 0 if the inputs are *equal.* In order to compare binary numbers containing two bits each, an additional exclusive-OR gate is necessary.

Some Integrated circuit comparators provide additional outputs that indicate which of the two numbers being compared is the larger. There is an output that indicates when number A is greater than number B (A > B) and an output that indicates when number A is less than number B (A < B).

4.4.2.1 The 7485 Four-Bit Magnitude Comparator:

The 7485 is a representative integrated circuit comparator in the 54/74 family. The logic symbol is shown in Figure 4.4.

Notice that this device has three cascading inputs. (<,=,>). These inputs allow several comparators to be cascaded for comparison of any number of bits greater than four. To expand the comparator, the A < B, A = B, and A > B outputs of the less significant comparator are connected to the corresponding cascading inputs of the next higher comparator. The least significant comparator must have a HIGH on the = input and LOWs on the < and > inputs.

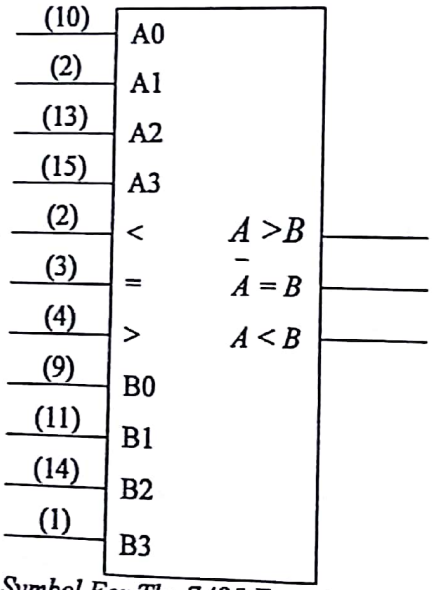


Figure 4.4: Logic Symbol For The 7485 Four-Bit Magnitude Comparator

Prelab

1- Build a **Adder** circuit which add two 4-bit numbers by using a 4-bit parallel adder. (Use the 7483).

Lab Session 5
Design of Sequential Circuits

5.1 Objectives:

1. To learn how to operate the flip-flops.
2. To utilize the basic flip-flops to build asynchronous counters.
3. Use JK flip-flops to design.

5.2 Materials Needed:

- ICs: 7493, 7408, 7400 and 74107

5.3 Reference Reading

- Digital Fundamentals, Floyd, 8th edition. Section: 8 and Section 9.

5.4 Information Summary:

logic circuits are classified into two classes:

1. **Combination logic circuits:** using AND, OR, and NOT gates. The basic building block for combinational logic circuits is the *logic gate*.

2. **Sequential logic circuits:** involve timing and memory devices. The basic building block for sequential logic circuits is the *flip-flop* (FF). Flip-flops are wired to form *counters*, *shift registers*, and *various memory devices*.

5.4.1 Flip Flops:

Flip-flop is a basic storage circuit that can store only one bit at a time. In this subsection we will introduce different types of flip-flops.

5.4.1.1 Clocked J-K Flip-Flop

The J-K flip-flop is considered the universal flip-flop, having the features of all the other types of flip-flops. The logic symbol for the J-K flip-flop is illustrated in figure 5.4. Its unique feature is the toggle mode of operation so useful in designing counters. When the J-K flip-flop is wired for use only in the toggle mode, it is commonly called a T flip-flop. Table 5.1 illustrates the truth table of the J-K flip-flop

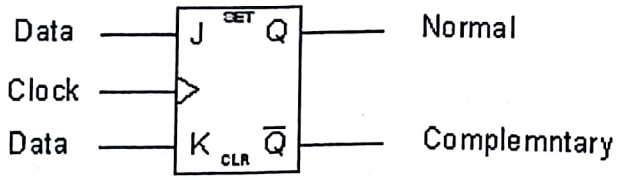


Figure 5.1: Logic Symbols Of J-K Flip-Flop.

Mode of Operation	***Inputs***			***Outputs***		*Effects on output Q*
	CLK	*J*	*K*	$Q_{(t+1)}$	$\overline{Q}_{(t+1)}$	
Hold		0	0	$Q_{(t)}$	$\overline{Q}_{(t)}$	No Change
Set		0	1	0	1	Reset or Cleared to 0
Reset		1	0	1	0	Set to 1
Prohibited		1	1	$\overline{Q}_{(t)}$	$Q_{(t)}$	Toggling

Table 5.1: Truth Table For A Clocked J-K Flip-Flop

5.4.1.2 Clocked D Flip-Flop

The logic symbol for the D *flip-flop* is shown in figure 5.2. it has only one data input (D) and a clock input (CLK) . the outputs are labeled Q and $\overline{Q}$. Table 5.2 illustrates the truth table of the D flip-flop.

D flip-flops are wired together to form shift registers and storage registers. The D flip-flop is often called a *delay flip-flop*. The word "delay" describes what happens to the data, at input D. Actually the data (a 0 or 1) at input D is delayed one clock pulse from getting to output Q. D flip-flops are some times also called data flip-flops or D-type latches.

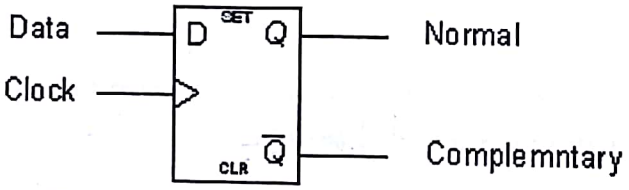


Figure 5.2: Logic Symbols Of D Flip-Flop.

Mode of Operation	***Inputs***		***Outputs***		*Effects on output Q*
	CLK	*D*	$Q_{(t+1)}$	$\overline{Q}_{(t+1)}$	
Reset		0	0	1	Reset to 0
Set		1	1	0	Set to 1

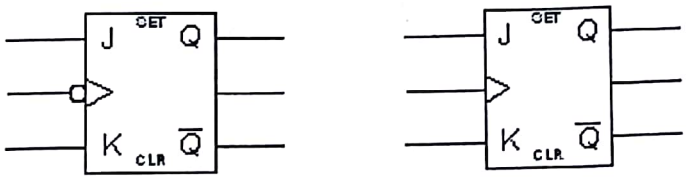
Table 5.2: Truth Table For A Clocked D Flip-Flop

5.4.2 Triggering Flip-Flops:

Flip-flops are classified as *synchronous* or *asynchronous* depending on their operation. Synchronous flip-flops are all those having a clock input. Asynchronous flip-flops are all those that have an enable-line which are called *Latches*.

The term synchronous means that the output changes state only at a specified point on a triggering input called the clock; that is, changes in the output occur with the clock. When using manufacturers' data manuals you will notice that many synchronous flip-flops are also classified as either *edge-triggered* or *pulse-triggered (master-slave)*.

An **edge-triggered** flip-flop changes state either at the positive edge (rising edge) or at the negative edge (falling edge) of the clock pulse and is sensitive to its inputs only at this transition of the clock.



Negative edge triggered *Popsitive edge triggered*

Figure 5.3: Positive And Negative Edge Triggered Flip-Flops.

The **pulse-triggered (master-slave)** means that data are entered into the flip-flop on the leading edge of the clock pulse but the output does not reflect the input state until the trailing edge of the clock pulse is end. The inputs must be setup prior to the clock pulse's leading edge, but the output is postponed until the trailing edge of the clock.

A major restriction of the pulse-triggered flip-flop is that the data inputs must not change while the clock pulse is HIGH. Because the flip-flop is sensitive to any change of input levels during this time.

5.4.3 74LS74: Dual D-Type Flip-Flops With Preset And Clear

This device contains two identical flip-flops that are independent of each other except for sharing VCC and GND. The flip-flops are positive-edge triggered and have active low asynchronous preset and clear inputs. The logic symbols for individual flip-flop within the package are shown in figure 5.4.
A low level at the preset or clear inputs sets or resets the outputs regardless of the levels of other inputs.

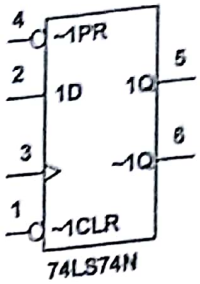


Figure 5.4: 7474 Flip-Flop

5.4.3 74LS76A Dual JK-Type Flip-Flops With Preset And Clear

This device contains two identical flip-flops that are independent of each other except for sharing VCC and GND. The flip-flops are negative-edge triggered and have active low asynchronous preset and clear inputs. The logic symbols for individual flip-flop within the package are shown in figure 5.5.
A low level at the preset or clear inputs sets or resets the outputs regardless of the levels of other inputs.

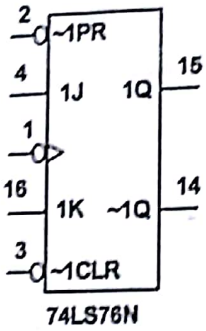


Figure 5.5: 7476 Flip-Flop

5.4.3 Definition Of State Machine

Let's begin with a general definition of a sequential circuit (state machine): A general sequential circuit consists of a combinational logic section and a memory section (flip-flops) as shown in Figure 5.6. In a clocked sequential circuit, there is a clock input to the memory section as indicated.

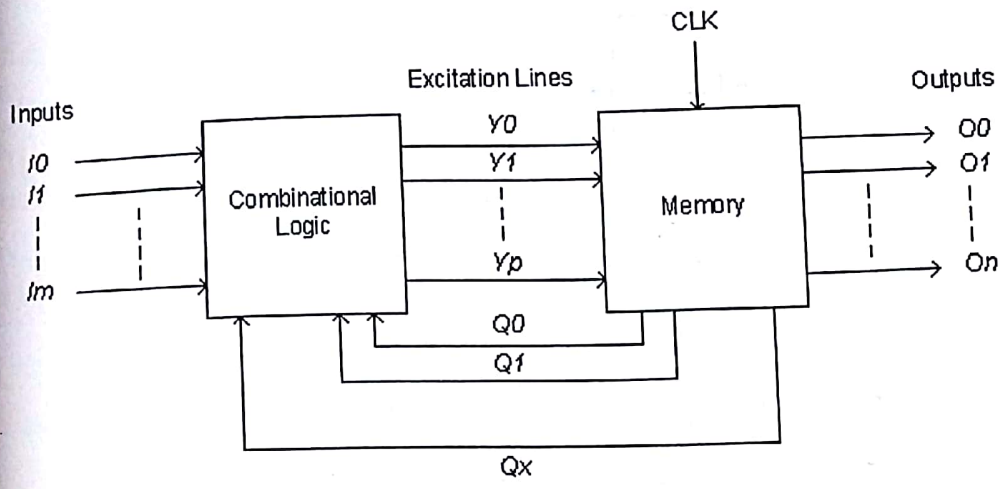


Figure 5.6: State Diagram

The information stored in the memory section, as well as the inputs to the combinational logic $(I_0, I_1, ..., I_m)$, is required for proper operation of the circuit. At any given time, the memory is in a state called the present state and will advance to a next state on a clock pulse as determined by conditions on the excitation lines $(Y_0, Y_1, ..., Y_P)$.
The present state of the memory is represented by the state variables $(Q_0, Q_1, ..., Q_x)$.
These state variables, along with the inputs $(I_0, I_1, ..., I_m)$, determine the system outputs $(O_0, O_1, ..., O_n)$.
The system outputs are the next state of the flip-flops.
Not all sequential circuits have input and output variables as in the general model just discussed. However, all have excitation variables and state variables.

5.4.4 General Design Procedure For Sequential Circuits

The general design procedure for sequential circuits will be explained by these series of steps:

Step 1: State Diagram
A counter is first described by a state diagram, which shows the progression of states through which the sequential circuit advances when it is clocked.

Step2: Next-State Table
Once the sequential circuit is defined by a state diagram, the second step is to derive a next-state table, which lists each state of the counter (present state) along with the corresponding next state. *The next state is the state that the counter goes go from its present state upon application of a clock pulse.*

Step 3: Flip-Flop Transition Table
Table 5.3 is a transition table for the J-K flip-flop. All possible output transitions are listed by showing the Q output of the flip-flop going from present state to next states. Q_N

is the present state of the flip-flop (before a clock pulse) and Q_{N+1} is the next state (after a clock pulse). For each output transitions, the J and K inputs that will cause the transitions to occur are listed. The Xs indicate a "don't care" (the input can be either a 1 or a 0).

Output Transitions		Flip-Flop Inputs	
Q_N	Q_{N+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Table 5.3: Transition Table of JK Flip-Flop

Step 4: Karnaugh Maps

Karnaugh maps are used to determine the logic required for the J and K inputs of each flip-flop in the sequential circuit. There is a Karnaugh map for the J input and a Karnaugh map for the K input of each flip-flop. Each cell in a Karnaugh map represents one of present states in the sequential circuit (Q_0, Q_1, ..., Q_x) and the inputs (*if its available*).

Step 5: Logic Expression For Flip-Flops Inputs

From the karnaugh maps. The expressions for the J and K inputs of each flip-flop must be obtained.

Step 6: Circuit Implementation:

The final step is to implement the combinational logic from the expressions for the J and K inputs and connect the flip-flops to form the complete sequential circuit.

5.5 Prelab:

The state transition table of a state machine is given below. Design this machine using JK flip-flops.

Present State		Input	Next State	
A	B	X	A*	B*
0	1	1	1	1
1	1	1	1	0
1	0	1	0	0
0	0	0	1	1
1	1	0	0	1

Table 5.4

Lab Session 6
Synchronous Counters

6.1 Objectives:

1. To utilize the basic flip-flops to build synchronous counters.
2. To utilize the 74LS161 Synchronous Binary Counter to get specific counting sequence.
3. To utilize the 74LS193 UP/DOWN Binary Counter to get specific counting sequence.

6.2 Materials Needed:

- ICs: 7476, 7411, 74161, 74193

6.3 Reference Reading

- Floyd, Digital Fundamentals

6.4 Information Summary:

The term Synchronous as applied to counter operation means that the counter is clocked in such a way that all flop-flops in the counter are triggered at the same time. This arrangement is accomplished by connecting the clock line to each stage of the counter.

6.4.1 Two Bit Synchronous Binary Counter:

Figure 6.1 shows a two-stage counter. Notice that an arrangement different from that for the asynchronous counter is applied.

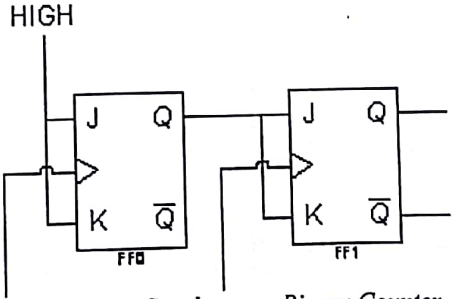


Figure 6.1 A Two Bit Synchronous Binary Counter.

The operation of this counter is as follows:

1. We will assume that the counter is initially in the binary 0 state and each of the flip-flops is a positive edge trigger flip-flop.

2. When the positive edge of the first clock pulse is applied, FF0 will toggle, and Q_0 will therefore go HIGH. When the leading edge of the first clock pulse is applied, $J_1 = K_1 = 0$. This is a no change state. Therefore Q_1 is still 0.

3. After CLK_1, $Q_0 = 1$ and $Q_1 = 0$. At the leading edge of CLK_2, FF0 will toggle, and Q_0 will go LOW. Since FF1 "sees" at HIGH on its J and K inputs when the triggering edge of this clock pulse occurs, the flip-flop toggles, and Q_1 goes HIGH.

4. After CLK_2, $Q_0 = 0$ and $Q_1 = 1$. At the leading edge of CLK_3, FF0 again toggles to 1, and FF1 remains 1 because its J and K inputs are both 0.

5. After CLK_3, $Q_0 = 1$ and $Q_1 = 1$. at the leading edge of CLK_4. Q_0 and Q_1 go LOW because they both have a toggle condition on their J and K inputs.

6.4.2 Three Bit Synchronous Binary Counter:

A three bit synchronous binary counter is shown in Figure 6.2, and its state sequence in table 6.1.

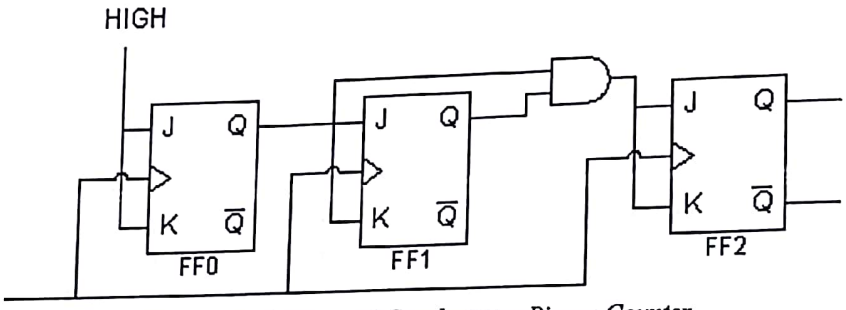


Figure 6.2 A Three Bit Synchronous Binary Counter.

Clock Pulse	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

Table 6.1 State Sequence Of Three Bit Synchronous Binary Counte

The operation of this counter is as follows:

1. Let us look at Q_0. Notice that Q_0 changes on each clock pulse as the counter progress because FF0 is held in the toggle mode by constant HIGH on its J & K inputs.

2. Let us look at Q_1. Notice that it goes to the opposite state following each time Q_0 is a 1. This change occurs at CLK_2, CLK_4, CLK_6, and CLK_8. The CLK_8 pulse causes the counter to recycle. To produce this operation, Q_0 is connected to the J and K inputs of FF1.
3. Let us look at Q_2. Notice that both times Q_2 changes state, it is preceded by the unique condition in which both Q_0 and Q_1 are HIGH . This condition is detected by the AND gate and applied to the J and K inputs of FF2, whenever both Q_0 and Q_1 are HIGH, the output of the AND gate makes the J and K inputs of FF2 HIGH. And FF2 toggles on the following clock pulse.

6.4.3 Synchronous Decade Counter:

A BCD counter counts in binary-coded decimal from 0000 to 1001 and back to 0000. Because of the return to 0 after a count of 9, a BCD counter does not have a regular pattern as in a straight binary count. The counter operation can be understood by examining the sequence of states in Table 6.2.

Clock Pulse	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	0	0	0	0

Table 6.2 *State Sequence Of Synchronous Decade Counter.*

The operation of this counter is as follows:

1. Notice that FF0 toggles on each clock pulse, so the logic equation for its J and K inputs is

$$J_0 = K_0 = 1$$

2. Notice that FF1 changes on the next clock pulse each time $Q_0 = 1$ and $Q_3 = 0$. so the logic equation for its J and K inputs is

$$J_1 = K_1 = Q_0 \overline{Q_3}$$

3. FF2 changes on the next clock pulse each time both $Q_0 = 1$ and $Q_1 = 1$. This requires an input logic equation as follows:

$$J_2 = K_2 = Q_0 Q_1$$

4. FF3 changes to the opposite state on the next clock pulse each time $Q_0 = 1$, $Q_1 = 1$, and $Q_2 = 1$ (count 7), or when $Q_0 = 1$ and $Q_3 = 1$ (count 9) .The equation for this is as follows:

$$J_3 = K_3 = Q_0 Q_1 Q_2 + Q_0 Q_3$$

6.4.4 UP/DOWN Synchronous Counter:

An up/down counter is one that is capable of progressing in either direction through a certain sequence. An up/down counter, sometimes called a *bidirectional counter*, can have any specified sequence of states. For example, a three-bit binary counter that advances upward through its sequence (0,1,2,3,4,5,6,7) and then can be reversed so that it goes through the sequence in the opposite direction (7,6,5,4,3,2,1,0) is an illustration of up/down sequential operation.

6.4.5 The 74LS161 Synchronous Binary Counter:

The 74LS161 is an example of an integrated circuit synchronous binary counter. A logic symbol is shown in Figure 6.4. This counter has several features in addition to the basic function previously discussed for the general synchronous binary counter.
First, the counter can be synchronously preset to any four-bit binary number by applying the proper levels to the data inputs. When a LOW is applied to the $\overline{\text{LOAD}}$ input, the counter will assume the data inputs on the next clock pulse. Thus, the counter sequence can be started with any four-bit binary number.

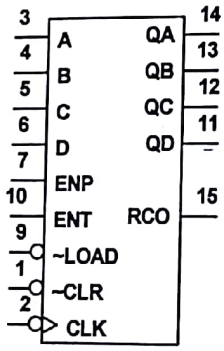


Figure 6.4: The 74LS161 Four Bit Synchronous Binary Counter.

Also, there is an active-LOW clear input ($\overline{\text{CLR}}$), which synchronously resets all four flip-flops in the counter.

There are two Enable inputs, ENP and ENT. These inputs must both be high for the counter to sequence through its binary states. When at least one is LOW, the counter is disabled.

The ripple clock output (RCO) goes HIGH when the counter reaches the last state in the sequence. Fifteen (1111).

6.4.6 The 74LS193 UP/DOWN Binary Counter

Figure 6.4 shows a logic symbol for the 74193, an UP/DOWN binary counter. The clock inputs are UP and DOWN pins. The direction of the count is determined by which clock input is pulsed while the other clock input is HIGH. When the UP input is HIGH and the DOWN input is pulsed. The counter counts down.

This device can be preset to any desired binary digit as determined by the states of the data inputs when the LOAD input is LOW.

The carry output ($\overline{CO}$) produces a LOW pulse when the terminal count fifteen (1111) is reached in the up mode. The borrow output ($\overline{BO}$) produces a LOW pulse when the terminal count zero (0000) is reached in the down mode.

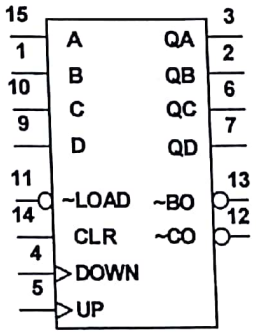


Figure 6. 4: The Logic Diagram For 74LS192 And 74LS193

6.4.7 The 74LS192 UP/DOWN Decade Counter

Figure 6.4 shows a logic diagram for the 74192, an UP/DOWN decade counter. The carry output ($\overline{CO}$) produces a LOW pulse when the terminal count nine (1001) is reached in the up mode. While the borrow output ($\overline{BO}$) produces a LOW pulse when the terminal count zero (0000) is reached in the down mode.

6.4.8 Buffers

The symbol in figure 6.5 is that of a non-inverting buffer/driver. The noninverting buffer serves no logical purpose it does not invert, but is used to supply greater drive current at its output than normal for any regular gate since regular digital ICs have limited driving current capabilities, the noninverting buffer/driver is very important when interfacing ICs with other devices such as LEDs, lamps, and others. Buffer/drivers are available in both noninverting and inverting form.

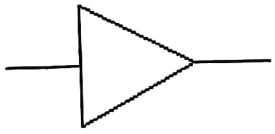


Figure 6.5: Non-Inverting Buffer/Driver Logic Symbol.

6.4 Prelab

1- By using 3 individuals negative -edge JK flip-flops. Design a synchronous counter with the counting sequence 2,3,4,5 then return back to 2.

Lab Session 8
Registers

8.1 Objectives:

- Introducing the different types of registers.
- Introducing some ICs registers.

8.2 Materials Needed:

- 7474, 74164, 74166, 74194

8.3 Reference Reading

- Digital Fundamentals, Floyd

8.4 Information Summary

Shift registers consist of an arrangement of flip-flops and are important in applications involving the storage and transfer of data in a digital system. The basic difference between a register and a counter is that a register has no specified sequence of states, except in certain very specialized applications. A register, in general, is used solely for storing and shifting data (1s and 0s) entered into it from an external source and possesses no characteristic internal sequence of states.

A register is a digital circuit with two basic functions: data storage and data movement. The storage capability of a register makes it an important type of memory device.

The storage capacity of a register is the number of bits (1s and 0s) of digital data it can retain. Each stage (flip-flop) in a shift register represents one bit of storage capacity; therefore, the number of stages in a register determines its total storage capacity. Registers are implemented with flip-flops or other storage devices.
The shifting capability of a register permits the movement of data from stage to stage within the register or into or out of the register upon application of clock pulses.

8.4.1 Serial In/ Serial Out Shift Registers

The serial in/serial out shift register accepts data serially – that is, one bit at a time on a single line. It produces the stored information on its output also in serial form.

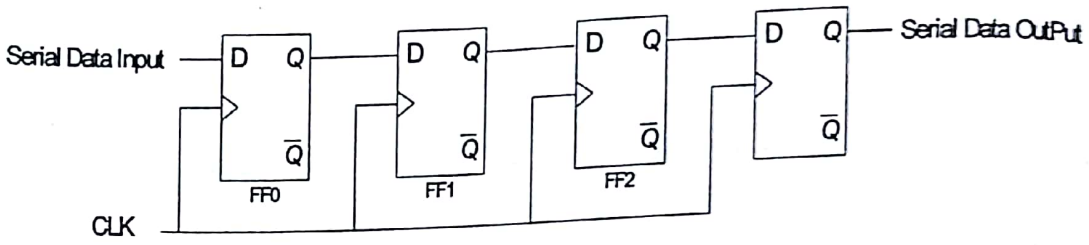


Figure 8.1: Serial in/Serial out shift register

8.4.2 Serial In/ Parallel Out Shift Registers

Data bits are entered serially (right-most bit first) into this type of register in the same manner as Serial in/Serial out shift register. The difference is the way in which the data bits are taken out of the register; in the parallel output register, the output of each stage is available. Once the data are stored, each bit appears on its respective output line, and all bits are available simultaneously, rather than on a bit-by-bit basis as with the serial output.

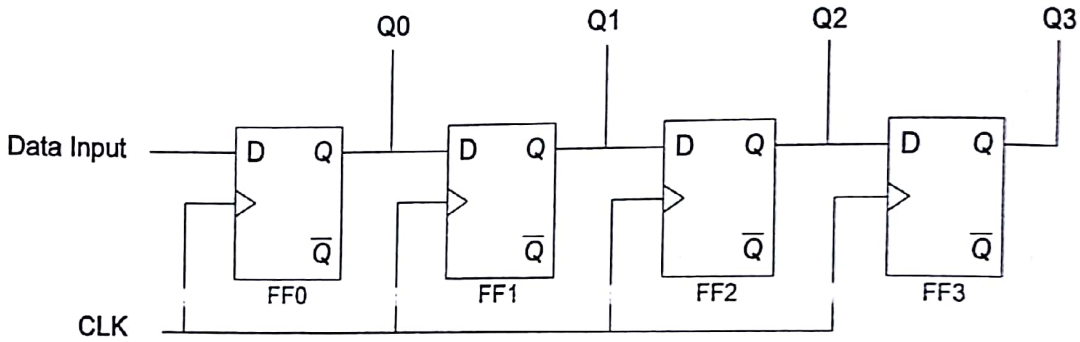


Figure 8.2: Serial In/Parallel Out Shift Register

8.4.3 Parallel In/Serial Out Shift Register

For a register with parallel data inputs, the bits are entered simultaneously into their respective stages on parallel lines rather on a bit-by-bit basis on one line as with serial data inputs.
Figure 8.3 illustrates a four-bit parallel in-serial out register. Notice that there are four data-input lines, D_0, D_1, D_2, and D_3. And a SHIFT/ $\overline{\text{LOAD}}$ input.

SHIFT/$\overline{\text{LOAD}}$ input allows four bits to be entered in parallel into the register. When SHIFT/ $\overline{\text{LOAD}}$ input is LOW. And each data bit input is applied to the D input of its respective flip-flop. When SHIFT/ $\overline{\text{LOAD}}$ is HIGH. Allowing the data bits to shift right from one stage to the next. The OR gates allow either the normal shifting operation or the parallel data-entry operation, depending on which AND gates are enabled by the level on the SHIFT/$\overline{\text{LOAD}}$.

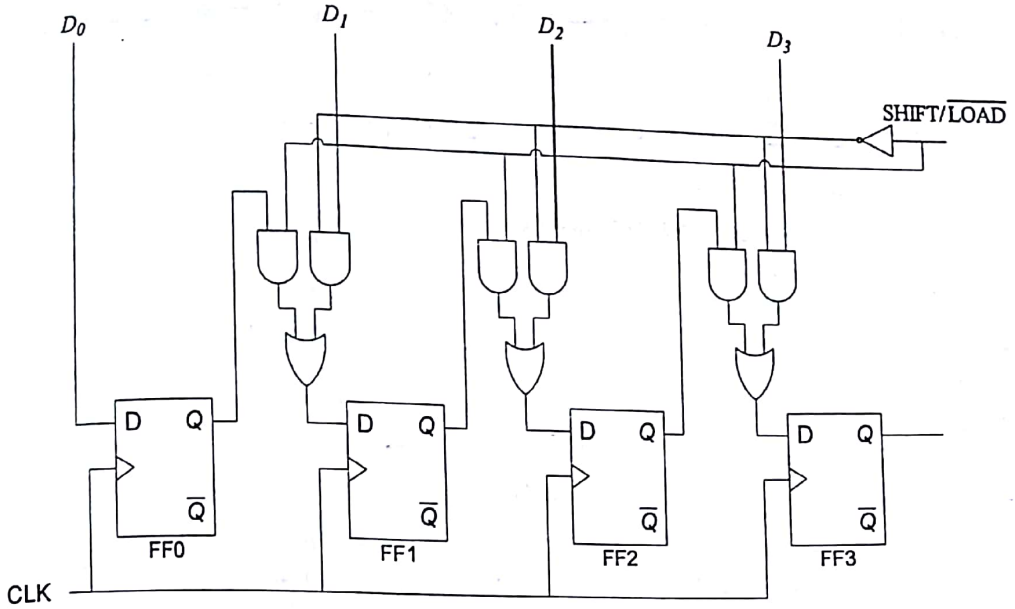


Figure 8.3: Parallel In/Serial Out Shift Register

8.4.4 Parallel In/Parallel Out Shift Register.

Figure 8.4 shows a parallel in/parallel out shift register.

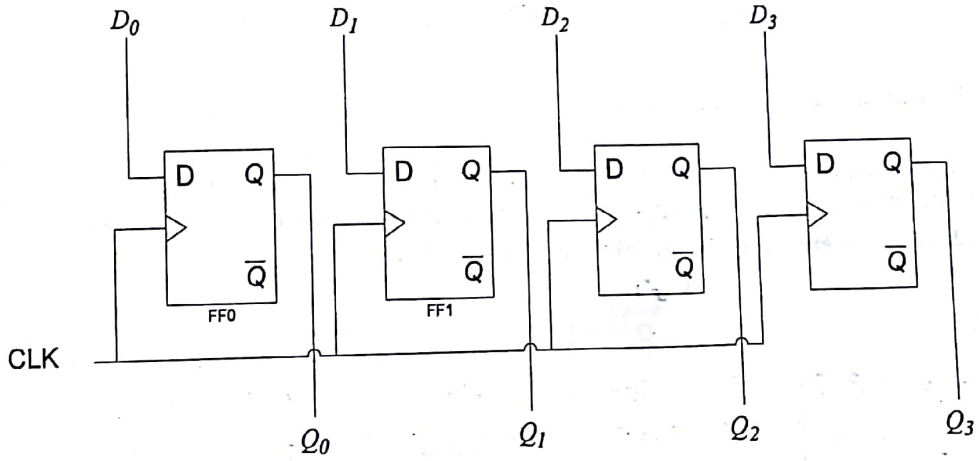


Figure 8.4: Parallel In/Parallel Out Shift Register

8.4.5 Bidirectional Shift Register

A bidirectional shift register is one which the data can be shifted either left or right. It can be implemented by using gating logic that enabled the transfer of a data bit from one stage to the next stage to the right or to the left, depending on the level of a control line.

8.4.6 74LS164 8-Bit Serial In/Parallel Out Shift Register

The 74LS164 is an example of an IC shift register having serial in/parallel out operation. The logic symbol is shown in figure 8.5.

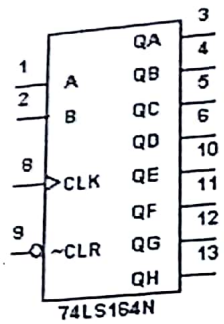


Figure 8.5: Logic Symbol of 74LS164

Notice that this device has tow gated serial inputs, A and B, Tow serial inputs that are ANDed internally. That is, both A and B inputs must be 1 for a 1 to be shifted into the first bit of the register.

Clear input ($\overline{CLR}$) is active-LOW and The parallel outputs are Q_0 through Q_7.

Return back to the logic diagram of 74LS164 in the datasheets and read it carefully.

8.4.7 74LS166 8-Bit Parallel In/Serial Out Shift Register

The 74LS166 is an example of an IC shift register that has a parallel in/serial out operation (It can also be operated as serial in/serial out). The logic symbol is shown in figure 8.6.

This device shifts when SH/$\overline{LD}$ is 1 and loads new data otherwise. The 166 has an unusual clocking arrangement called a "gated clock"; it has two clock inputs that are connected to the internal flip-flops as shown in figure 8.6. The designers of the 166 intended for CLK to be connected to a free-running system clock and for CLKINH to be asserted to inhibit CLK, so that neither shifting nor loading occurs on the next clock tick, and the current register contents are held. However for this to work, CLKINH must be changed only when CLK is 1; otherwise undesired clock edges occur on the internal flip-flops.

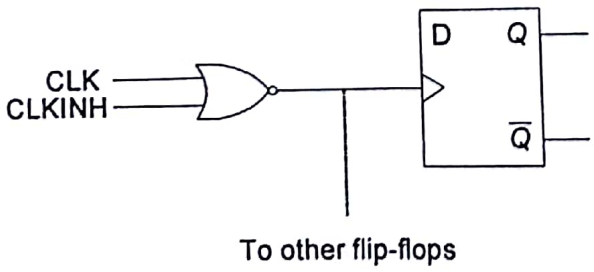
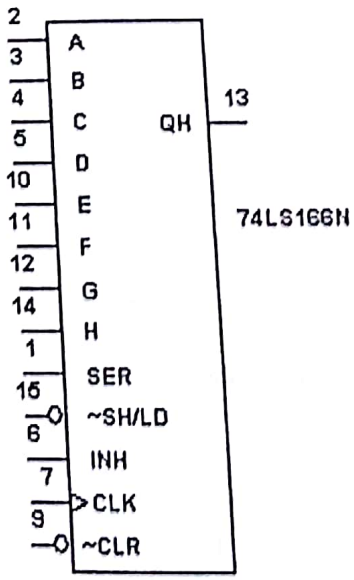


Figure 8.6: 74166 IC



8.4.8 The 74LS194 4-Bit Bidirectional Universal Shift Register.

The 74LS194 is an example of a universal bidirectional shift register in integrated circuit form. A universal shift register has both serial and parallel input and output capability. The mode operation of the 74LS194 is determined by the **mode control inputs:** S_0 and S_1.

When $S_0 = S_1 = 1$: Synchronous parallel loading is accomplished by applying the four bits of data and taking both mode control inputs. The data are loaded into the associated flip-flops and appear at the outputs after the positive transition of the clock input, During loading, serial data flow is inhibited.

When $S_0 = 1, S_1 = 0$: Shift right is accomplished synchronously with the rising edge of the clock pulse. Serial data for this mode is entered at the shift-right data input.

When $S_0 = 0, S_1 = 1$: Shift left is accomplished synchronously with the rising edge of the clock pulse. Serial data for this mode is entered at the shift-left data input.

When $S_0 = 0, S_1 = 0$: Clocking of the flip-flop is inhibited.

The Texas instruments data manual contains the descriptions, diagrams, and tables. Read the manual for a good overview of what this shift register can do.

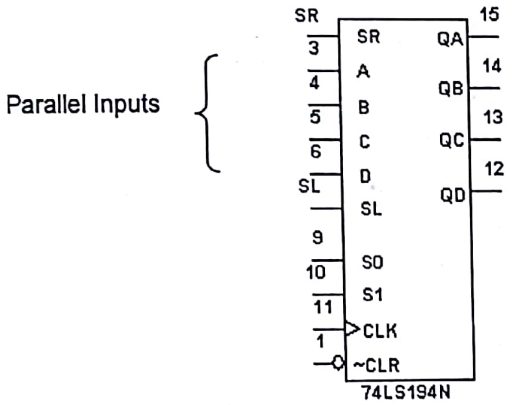


Figure 8.7: Logic Symbol Of Shift Register 74LS194

Exp. 9

ALU Implementation

Any computer system consists of 3 main parts:

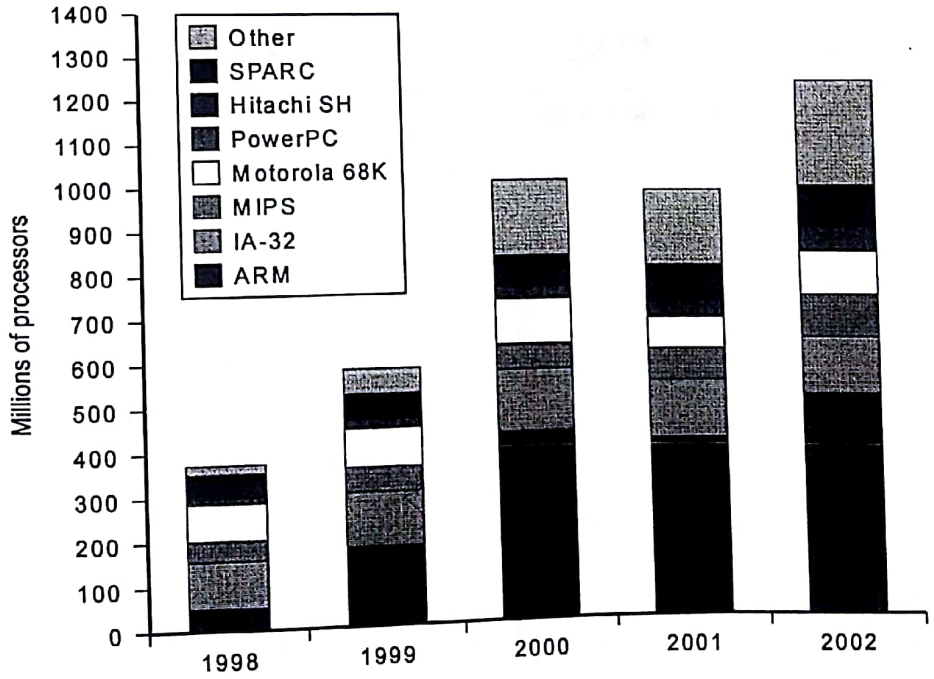
1. CPU
2. Memory subsystem
3. I/O subsystem

So any system contains these 3 main parts can be called a computer.

CPU is the heart of the computer system. One of basic components of the CPU is the ALU (Arithmetic Logic Unit) it is responsible for instructions execution. The execution of an instruction is called operation the operation can be either

A. arithmetic operations (addition, subtraction, multiplication and division)
B. logic operations (AND, NOT, OR, XOR)
C. Bit-shifting operations (shifting or rotating a word by a specified number of bits to the left or right, with or without sign extension). Shifts can be interpreted as multiplications by 2 and divisions by 2.

There are two types of signals (inputs to the ALU) which are the Data Signals and Control Signals.



Inputs and outputs

The inputs to the ALU are the data to be operated on (called operands) and a code from the control unit indicating which operation to perform. Its output is the result of the computation.

In many designs the ALU also takes or generates as inputs or outputs a set of condition codes from or to a status register. These codes are used to indicate cases such as carry-in or carry-out, overflow, divide-by-zero, etc.

1- 1-Bit ALU

The logical operations are easiest, because they map directly onto the hardware components in Figure1
The 1-bit logical unit for AND and OR looks like Figure 1 the multiplexor on the right then selects *a* AND *b* or *a* OR *b*, depending on whether the value of *Operation* is 0 or 1. The line that controls the multiplexor is shown in color to distinguish it from the lines containing data. Notice that we have renamed the control and output lines of the multiplexor to give them names that reflect the function of the ALU.
The next function to include is addition. An adder must have two inputs for the operands and a single-bit output for the sum. There must be a second output to pass on the carry, called *CarryOut*.Since the CarryOut from the neighbor adder must be included as an input, we need a third input. This input is called *CarryIn*. Figure B.5.2 shows the inputs and the outputs of a 1-bit adder. Since we know

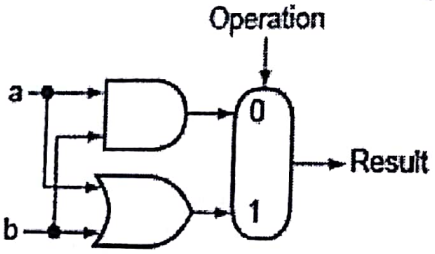


FIGURE 1 **The 1-bit logical unit for AND and OR.**

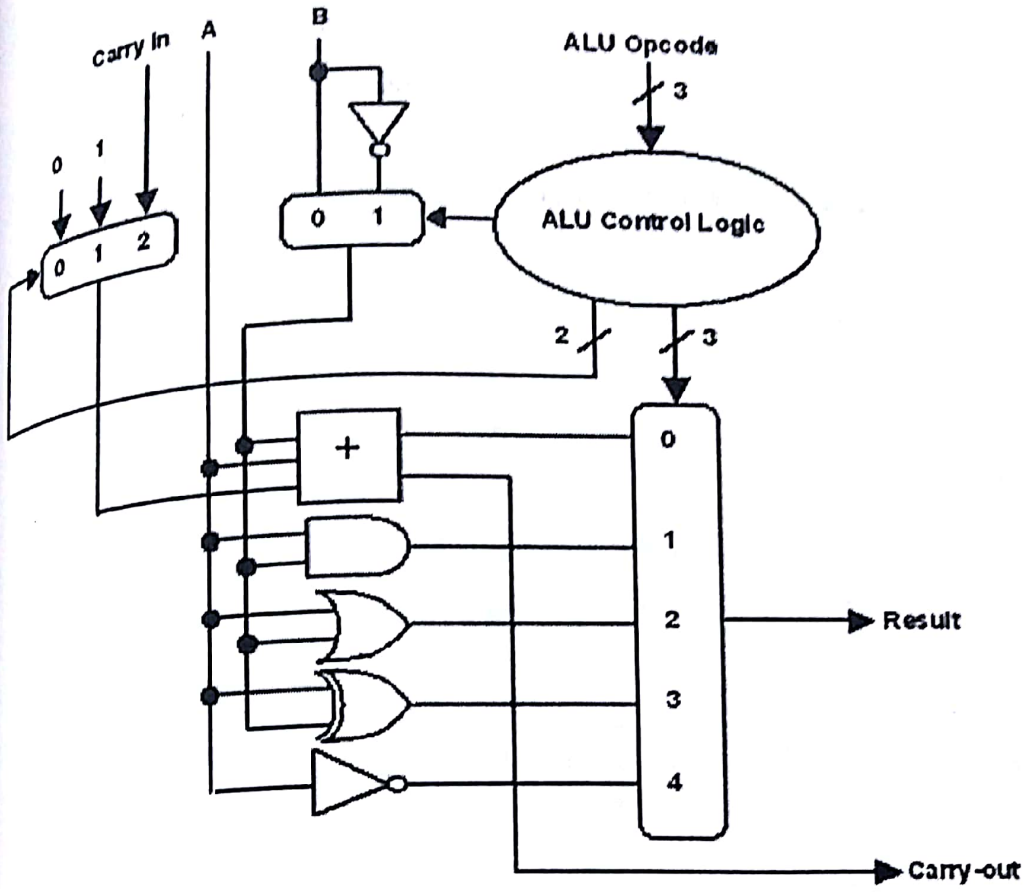


Figure 2 1 bit ALU logical and Arithmetic

2- 32-bit ALU

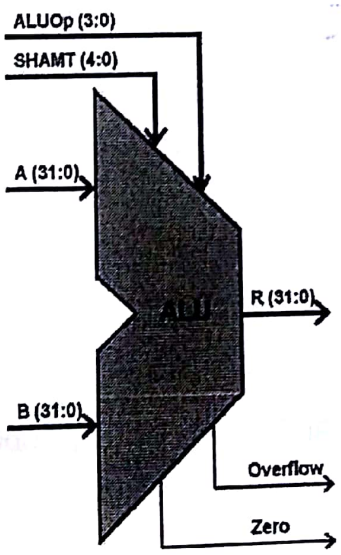


Figure 3 ALU Block Diagram

A 32 bit ALU can be implemented by combining 32 of the above 1 bit ALU Sequentially. By feeding the carry out as a carry in for the next stage

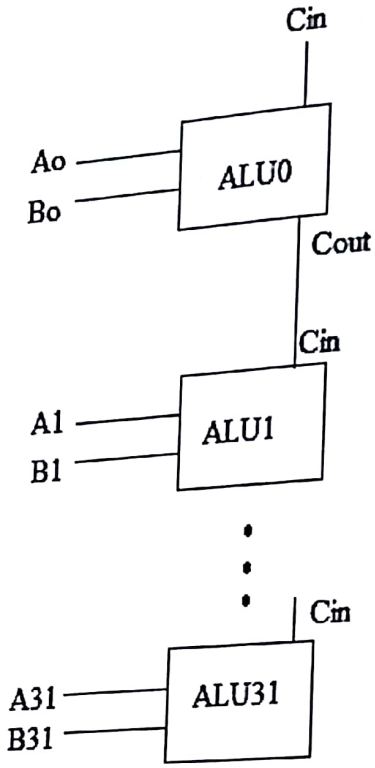


Figure 4 32 bit Sequential ALU

Lab Work

Design and implement a simple 1 bit ALU that do the following functions: And, Or, Inverter, Xor

Electrical & Computer Engineering Department

Microprocessors & Logic Circuits Lab

ICs Data Sheets

Prepared by Eng. [redacted]

- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

description

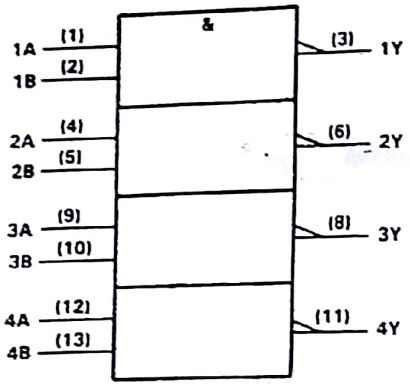
These devices contain four independent 2-input NAND gates.

The SN5400, SN54LS00, and SN54S00 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN7400, SN74LS00, and SN74S00 are characterized for operation from 0°C to 70°C.

FUNCTION TABLE (each gate)

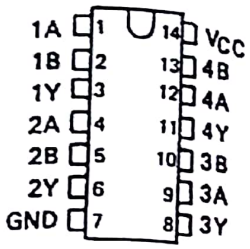
INPUTS		OUTPUT
A	B	Y
H	H	L
L	X	H
X	L	H

logic symbol†

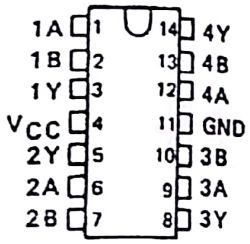


†This symbol is in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, and N packages.

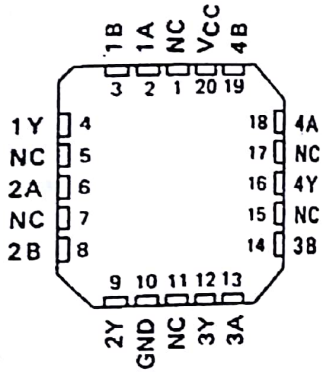
SN5400 . . . J PACKAGE
SN54LS00, SN54S00 . . . J OR W PACKAGE
SN7400 . . . N PACKAGE
SN74LS00, SN74S00 . . . D OR N PACKAGE
(TOP VIEW)



SN5400 . . . W PACKAGE
(TOP VIEW)

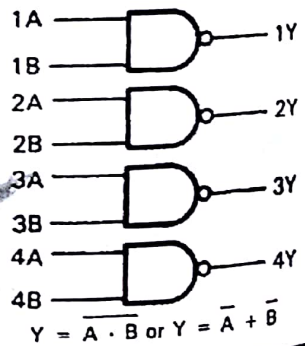


SN54LS00, SN54S00 . . . FK PACKAGE
(TOP VIEW)



NC - No internal connection

logic diagram (positive logic)



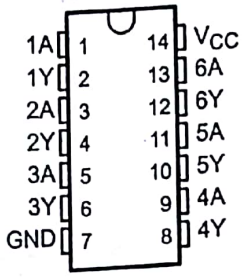
$Y = \overline{A \cdot B}$ or $Y = \bar{A} + \bar{B}$

- Dependable Texas Instruments Quality and Reliability

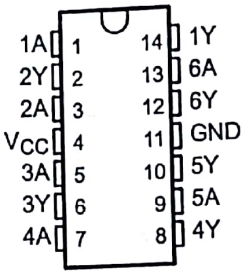
description

These devices contain six independent inverters.

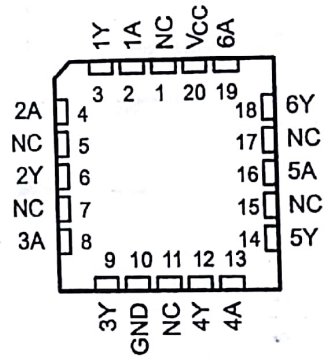
SN5404 . . . J PACKAGE
SN54LS04, SN54S04 . . . J OR W PACKAGE
SN7404 . . . D, N, OR NS PACKAGE
SN74LS04 . . . D, DB, N, OR NS PACKAGE
SN74S04 . . . D OR N PACKAGE
(TOP VIEW)



SN5404 . . . W PACKAGE
(TOP VIEW)



SN54LS04, SN54S04 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

description

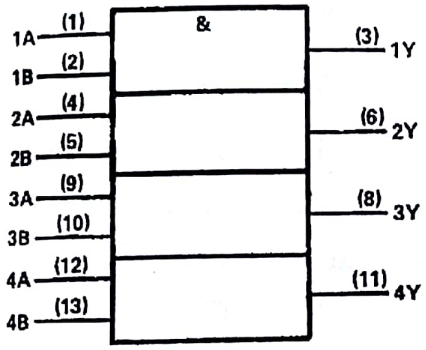
These devices contain four independent 2-input AND gates.

The SN5408, SN54LS08, and SN54S08 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN7408, SN74LS08 and SN74S08 are characterized for operation from 0° to 70°C.

FUNCTION TABLE (each gate)

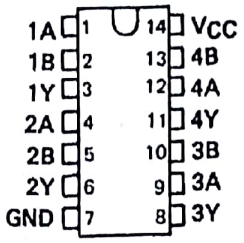
INPUTS		OUTPUT
A	B	Y
H	H	H
L	X	L
X	L	L

logic symbol†

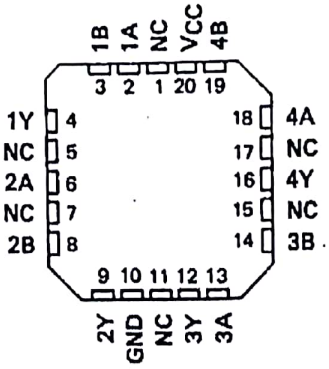


†This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

SN5408, SN54LS08, SN54S08 . . . J OR W PACKAGE
SN7408 . . . J OR N PACKAGE
SN74LS08, SN74S08 . . . D, J OR N PACKAGE
(TOP VIEW)

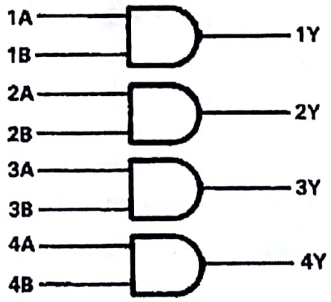


SN54LS08, SN54S08 . . . FK PACKAGE
(TOP VIEW)



NC—No internal connection

logic diagram (positive logic)

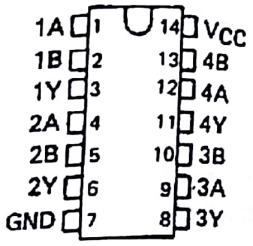


$Y = A \cdot B$ or $Y = \overline{\overline{A} + \overline{B}}$

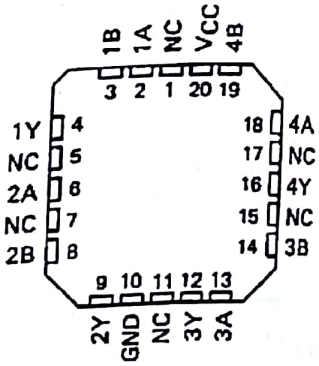
SDLS100

- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

SN5432, SN54LS32, SN54S32 . . . J OR W PACKAGE
SN7432 . . . N PACKAGE
SN74LS32, SN74S32 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS32, SN54S32 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

description

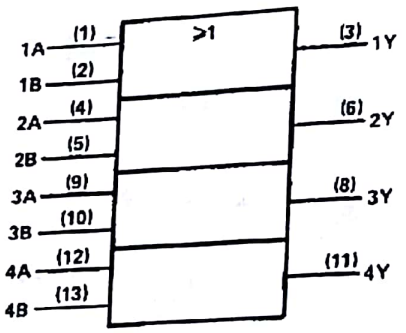
These devices contain four independent 2-input OR gates.

The SN5432, SN54LS32 and SN54S32 are characterized for operation over the full military range of −55°C to 125°C. The SN7432, SN74LS32 and SN74S32 are characterized for operation from 0°C to 70°C.

FUNCTION TABLE (each gate)

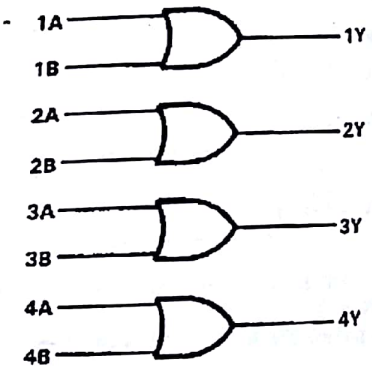
INPUTS		OUTPUT
A	B	Y
H	X	H
X	H	H
L	L	L

logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, or W packages.

logic diagram



positive logic

$Y = A + B$ or $Y = \overline{\overline{A} \cdot \overline{B}}$

- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

description

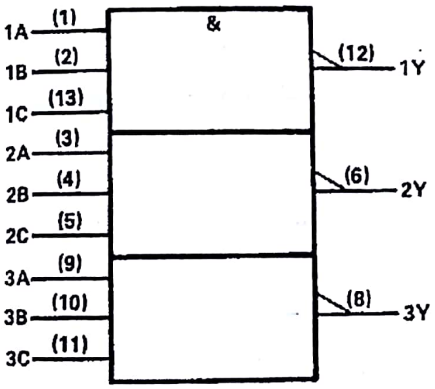
These devices contain three independent 3-input NAND gates.

The SN5410, SN54LS10, and SN54S10 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN7410, SN74LS10, and SN74S10 are characterized for operation from 0°C to 70°C.

FUNCTION TABLE (each gate)

INPUTS			OUTPUT
A	B	C	Y
H	H	H	L
L	X	X	H
X	L	X	H
X	X	L	H

logic symbol†

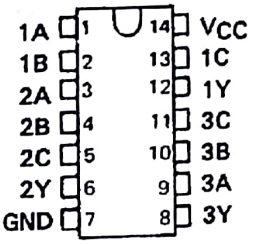


†This symbol is in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, and N packages.

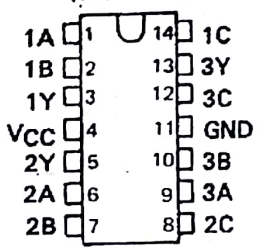
positive logic

$$Y = \overline{A \cdot B \cdot C} \text{ or } Y = \overline{A} + \overline{B} + \overline{C}$$

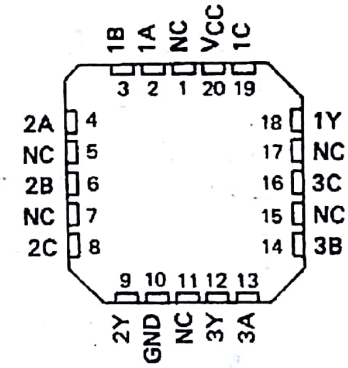
SN5410 . . . J PACKAGE
SN54LS10, SN54S10 . . . J OR W PACKAGE
SN7410 . . . N PACKAGE
SN74LS10, SN74S10 . . . D OR N PACKAGE
(TOP VIEW)



SN5410 . . . W PACKAGE
(TOP VIEW)

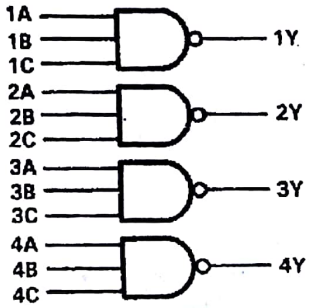


SN54LS10, SN54S10 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

logic diagram (positive logic)



FAIRCHILD SEMICONDUCTOR™

August 1986
Revised March 2000

DM74LS11
Triple 3-Input AND Gate

General Description

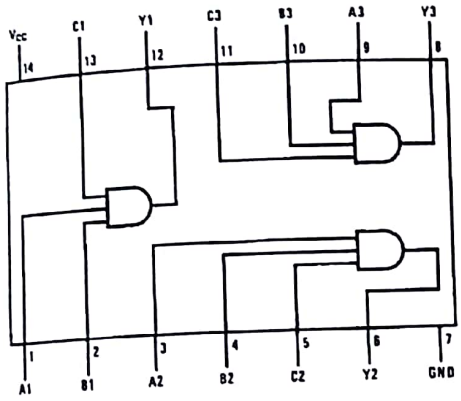
This device contains three independent gates each of which performs the logic AND function.

Ordering Code:

Order Number	Package Number	Package Description
DM74LS11M	M14A	14-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-120, 0.150 Narrow
DM74LS11N	N14A	14-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide

Devices also available in Tape and Reel. Specify by appending the suffix letter "X" to the ordering code.

Connection Diagram



Function Table

Y = ABC

Inputs			Output
A	B	C	Y
X	X	L	L
X	L	X	L
L	X	X	L
H	H	H	H

H = HIGH Logic Level
L = LOW Logic Level
X = Either LOW or HIGH Logic Level

 DS006350

SDLS079

- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

description

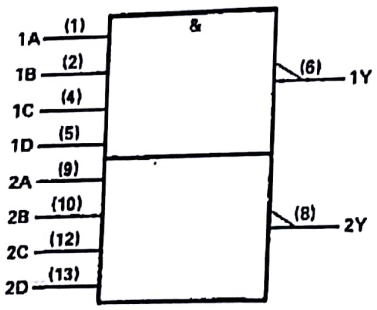
These devices contain two independent 4-input NAND gates.

The SN5420, SN54LS20, and SN54S20 are characterized for operation over the full military range of −55°C to 125°C. The SN7420, SN74LS20, and SN74S20 are characterized for opertion from 0°C to 70°C.

FUNCTION TABLE (each gate)

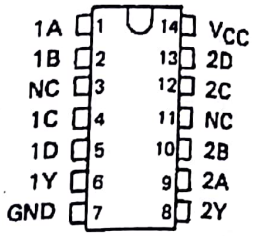
INPUTS				OUTPUT
A	B	C	D	Y
H	H	H	H	L
L	X	X	X	H
X	L	X	X	H
X	X	L	X	H
X	X	X	L	H

logic symbol†

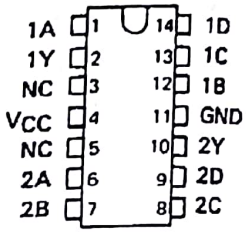


†This symbol is in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

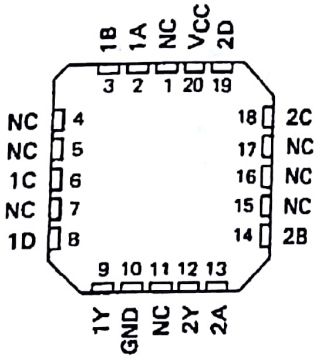
SN5420 . . . J PACKAGE
SN54LS20, SN54S20 . . . J OR W PACKAGE
SN7420 . . . N PACKAGE
SN74LS20, SN74S20 . . . D OR N PACKAGE
(TOP VIEW)



SN5420 . . . W PACKAGE
(TOP VIEW)

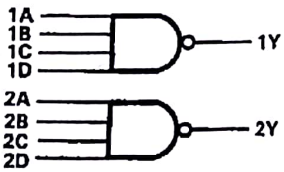


SN54LS20, SN54S20 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

logic diagram



positive logic $Y = \overline{A \cdot B \cdot C \cdot D}$ or $Y = \overline{A} + \overline{B} + \overline{C} + \overline{D}$

- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

description

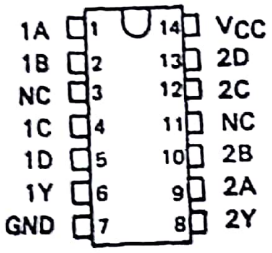
These devices contain two independent 4-input AND gates.

The SN54LS21 is characterized for operation over the full military temperature range of −55°C to 125°C. The SN74LS21 is characterized for operation from 0°C to 70°C.

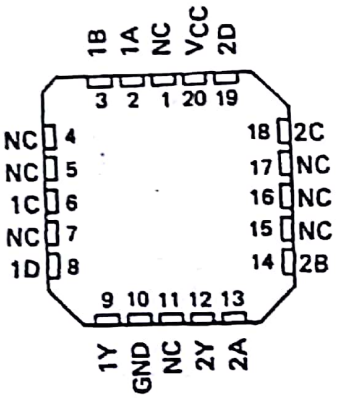
FUNCTION TABLE (each gate)

INPUTS				OUTPUT
A	B	C	D	Y
H	H	H	H	H
L	X	X	X	L
X	L	X	X	L
X	X	L	X	L
X	X	X	L	L

SN54LS21 . . . J OR W PACKAGE
SN74LS21 . . . D OR N PACKAGE
(TOP VIEW)

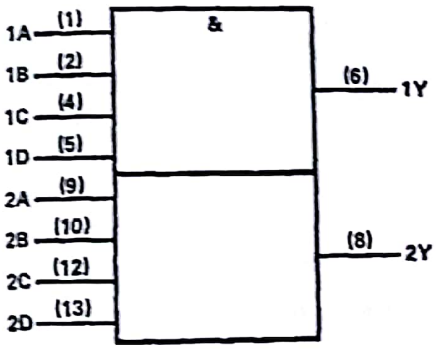


SN54LS21 . . . FK PACKAGE
(TOP VIEW)



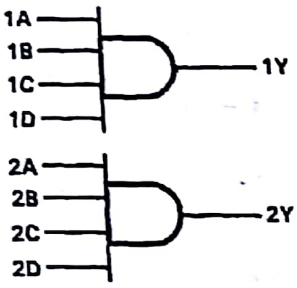
NC—No internal connection

logic symbol†



†This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

logic diagram



(positive logic) $Y = A \cdot B \cdot C \cdot D$ or $Y = \overline{\overline{A} + \overline{B} + \overline{C} + \overline{D}}$

SN5432, SN54LS32, SN54S32, SN7432, SN74LS32, SN74S32
QUADRUPLE 2-INPUT POSITIVE-OR GATES

DECEMBER 1983 – REVISED MARCH 1988

- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

description

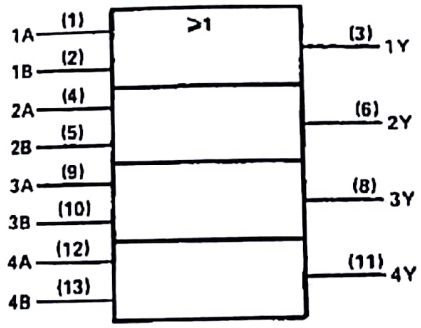
These devices contain four independent 2-input OR gates.

The SN5432, SN54LS32 and SN54S32 are characterized for operation over the full military range of −55°C to 125°C. The SN7432, SN74LS32 and SN74S32 are characterized for operation from 0°C to 70°C.

FUNCTION TABLE (each gate)

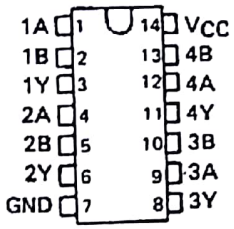
INPUTS		OUTPUT
A	B	Y
H	X	H
X	H	H
L	L	L

logic symbol†

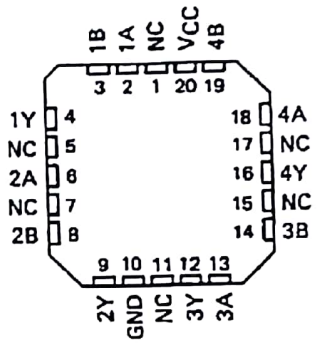


† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, or W packages.

SN5432, SN54LS32, SN54S32 . . . J OR W PACKAGE
SN7432 . . . N PACKAGE
SN74LS32, SN74S32 . . . D OR N PACKAGE
(TOP VIEW)

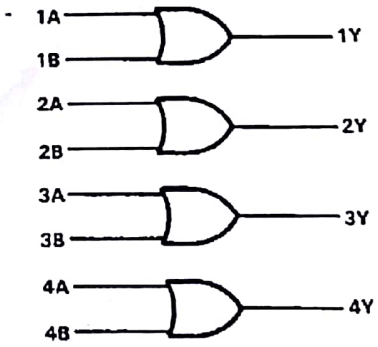


SN54LS32, SN54S32 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

logic diagram



positive logic

$$Y = A + B \text{ or } Y = \overline{\overline{A} \cdot \overline{B}}$$

TEXAS INSTRUMENTS
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

SDLS109 – MARCH 1974 – REVISED MARCH 1988

- All Outputs Are High for Invalid Input Conditions
- Also for Application as
  4-Line-to-16-Line Decoders
  3-Line-to-8-Line Decoders
- Diode-Clamped Inputs

TYPES	TYPICAL POWER DISSIPATION	TYPICAL PROPAGATION DELAYS
'42A	140 mW	17 ns
'LS42	35 mW	17 ns

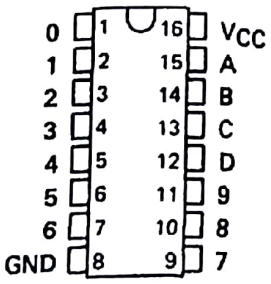
description

These monolithic BCD-to-decimal decoders consist of eight inverters and ten four-input NAND gates. The inverters are connected in pairs to make BCD input data available for decoding by the NAND gates. Full decoding of valid input logic ensures that all outputs remain off for all invalid input conditions.

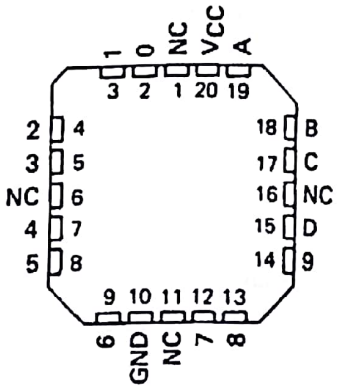
The '42A and 'LS42 feature inputs and outputs that are compatible for use with most TTL and other saturated low-level logic circuits. DC noise margins are typically one volt.

The SN5442A and SN54LS42 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN7442A and SN74LS42 are characterized for operation from 0°C to 70°C.

SN5442A, SN54LS42 . . . J OR W PACKAGE
SN7442A . . . N PACKAGE
SN74LS42 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS42 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

'46A, '47A, 'LS47 feature

- Open-Collector Outputs Drive Indicators Directly
- Lamp-Test Provision
- Leading/Trailing Zero Suppression

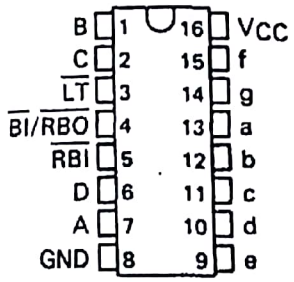
'48, 'LS48 feature

- Internal Pull-Ups Eliminate Need for External Resistors
- Lamp-Test Provision
- Leading/Trailing Zero Suppression

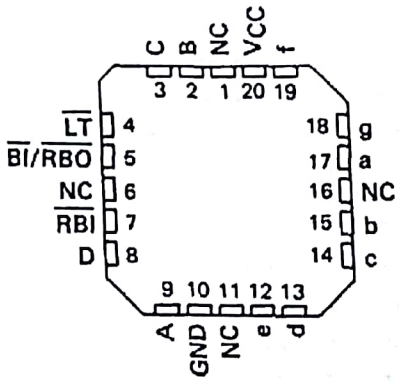
'LS49 feature

- Open-Collector Outputs
- Blanking Input

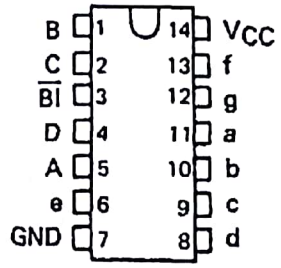
SN5446A, SN5447A, SN54LS47, SN5448, SN54LS48 . . . J PACKAGE
SN7446A, SN7447A, SN7448 . . . N PACKAGE
SN74LS47, SN74LS48 . . . D OR N PACKAGE
(TOP VIEW)



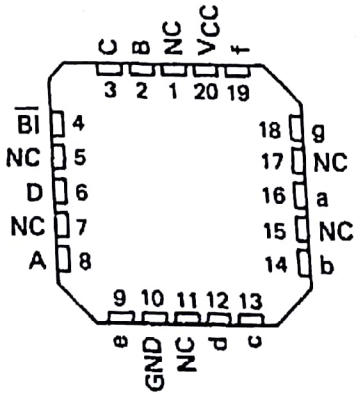
SN54LS47, SN54LS48 . . . FK PACKAGE
(TOP VIEW)



SN54LS49 . . . J OR W PACKAGE
SN74LS49 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS49 . . . FK PACKAGE
(TOP VIEW)



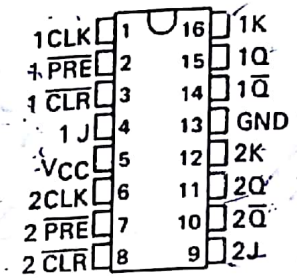
NC – No internal connection

DUAL J-K FLIP-FLOPS WITH PRESET AND CLEAR

SDLS121 – DECEMBER 1983 – REVISED MARCH 1988

- **Package Options Include Plastic and Ceramic DIPs and Ceramic Flat Packages**
- **Dependable Texas Instruments Quality and Reliability**

SN5476, SN54LS76A . . . J PACKAGE
SN7476 . . . N PACKAGE
SN74LS76A . . . D OR N PACKAGE
(TOP VIEW)



description

The '76 contains two independent J-K flip-flops with individual J-K, clock, preset, and clear inputs. The '76 is a positive-edge-triggered flip-flop. J-K input is loaded into the master while the clock is high and transferred to the slave on the high-to-low transition. For these devices the J and K inputs must be stable while the clock is high.

The 'LS76A contain two independent negative-edge-triggered flip-flops. The J and K inputs must be stable one setup time prior to the high-to-low clock transition for predicatble operation. The preset and clear are asynchronous active low inputs. When low they override the clock and data inputs forcing the outputs to the steady state levels as shown in the function table.

The SN5476 and the SN54LS76A are characterized for operation over the full military temperature range of −55°C to 125°C. The SN7476 and the SN74LS76A are characterized for operation from 0°C to 70°C.

'76
FUNCTION TABLE

INPUTS					OUTPUTS	
$\overline{PRE}$	$\overline{CLR}$	CLK	J	K	Q	$\overline{Q}$
L	H	X	X	X	H	L
H	L	X	X	X	L	H
L	L	X	X	X	H†	H†
H	H	⎍	L	L	Q_0	$\overline{Q}_0$
H	H	⎍	H	L	H	L
H	H	⎍	L	H	L	H
H	H	⎍	H	H	TOGGLE	

'LS76A
FUNCTION TABLE

INPUTS					OUTPUTS	
$\overline{PRE}$	$\overline{CLR}$	CLK	J	K	Q	$\overline{Q}$
L	H	X	X	X	H	L
H	L	X	X	X	L	H
L	L	X	X	X	H†	H†
H	H	↓	L	L	Q_0	$\overline{Q}_0$
H	H	↓	H	L	H	L
H	H	↓	L	H	L	H
H	H	↓	H	H	TOGGLE	
H	H	H	X	X	Q_0	$\overline{Q}_0$

† This configuration is nonstable; that is, it will not persist when either preset or clear returns to its inactive (high) level.

FAIRCHILD SEMICONDUCTOR™

August 1986
Revised March 2000

DM74LS83A
4-Bit Binary Adder with Fast Carry

General Description

These full adders perform the addition of two 4-bit binary numbers. The sum (Σ) outputs are provided for each bit and the resultant carry (C4) is obtained from the fourth bit. These adders feature full internal look ahead across all four bits. This provides the system designer with partial look-ahead performance at the economy and reduced package count of a ripple-carry implementation.

The adder logic, including the carry, is implemented in its true form meaning that the end-around carry can be accomplished without the need for logic or level inversion.

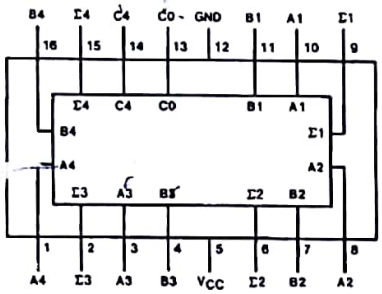
Features

- Full-carry look-ahead across the four bits
- Systems achieve partial look-ahead performance with the economy of ripple carry
- Typical add times
  - Two 8-bit words 25 ns
  - Two 16-bit words 45 ns
- Typical power dissipation per 4-bit adder 95 mW

Ordering Code:

Order Number	Package Number	Package Description
DM74LS83AN	N16E	16-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide

Connection Diagram



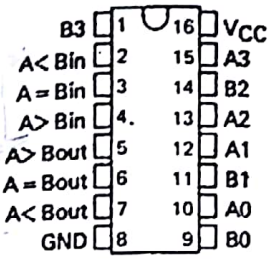
 DS006378

TYPE	TYPICAL POWER DISSIPATION	TYPICAL DELAY (4-BIT WORDS)
'85	275 mW	23 ns
'LS85	52 mW	24 ns
'S85	365 mW	11 ns

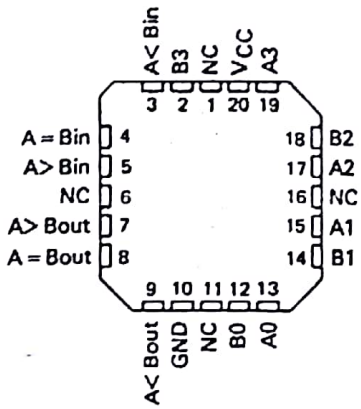
description

These four-bit magnitude comparators perform comparison of straight binary and straight BCD (8-4-2-1) codes. Three fully decoded decisions about two 4-bit words (A, B) are made and are externally available at three outputs. These devices are fully expandable to any number of bits without external gates. Words of greater length may be compared by connecting comparators in cascade. The A > B, A < B, and A = B outputs of a stage handling less-significant bits are connected to the corresponding A > B, A < B, and A = B inputs of the next stage handling more-significant bits. The stage handling the least-significant bits must have a high-level voltage applied to the A = B input. The cascading paths of the '85, 'LS85, and 'S85 are implemented with only a two-gate-level delay to reduce overall comparison times for long words. An alternate method of cascading which further reduces the comparison time is shown in the typical application data.

SN5485, SN54LS85, SN54S85 . . . J OR W PACKAGE
SN7485 . . . N PACKAGE
SN74LS85, SN74S85 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS85, SN54S85 . . . FK PACKAGE
(TOP VIEW)



NC - No internal connection

FUNCTION TABLE

COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A3, B3	A2, B2	A1, B1	A0, B0	A > B	A < B	A = B	A > B	A < B	A = B
A3 > B3	X	X	X	X	X	X	H	L	L
A3 < B3	X	X	X	X	X	X	L	H	L
A3 = B3	A2 > B2	X	X	X	X	X	H	L	L
A3 = B3	A2 < B2	X	X	X	X	X	L	H	L
A3 = B2	A2 = B2	A1 > B1	X	X	X	X	H	L	L
A3 = B3	A2 = B2	A1 < B1	X	X	X	X	L	H	L
A2 = B3	A2 = B2	A1 = B1	A0 > B0	X	X	X	H	L	L
A3 = B3	A2 = B2	A1 = B1	A0 < B0	X	X	X	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	H	L	L	H	L	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	H	L	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	X	X	H	L	L	H
A3 = B3	A2 = B2	A1 = B1	A0 = B0	H	H	L	L	L	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	L	L	H	H	L

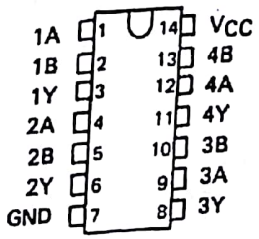
QUADRUPLE 2-INPUT EXCLUSIVE-OR GATES

SDLS124 – DECEMBER 1972 – REVISED MARCH 1988

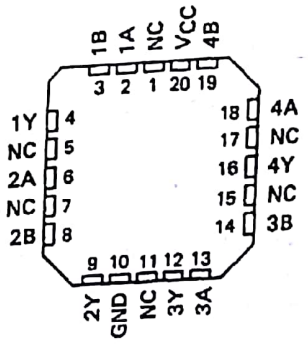
- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Standard Plastic and Ceramic 300-mil DIPs
- Dependable Texas Instruments Quality and Reliability

TYPE	TYPICAL AVERAGE PROPAGATION DELAY TIME	TYPICAL TOTAL POWER DISSIPATION
'86	14 ns	150 mW
'LS86A	10 ns	30.5 mW
'S86	7 ns	250 mW

SN5486, SN54LS86A, SN54S86 . . . J OR W PACKAGE
SN7486 . . . N PACKAGE
SN74LS86A, SN74S86 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS86A, SN54S86 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

description

These devices contain four independent 2-input Exclusive-OR gates. They perform the Boolean functions $Y = A \oplus B = \overline{A}B + A\overline{B}$ in positive logic.

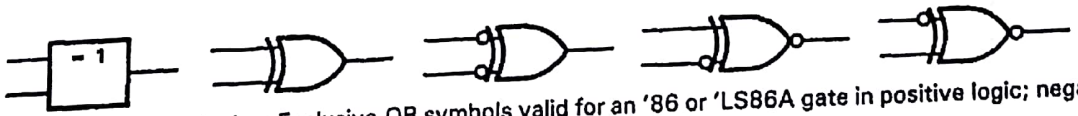
A common application is as a true/complement element. If one of the inputs is low, the other input will be reproduced in true form at the output. If one of the inputs is high, the signal on the other input will be reproduced inverted at the output.

The SN5486, 54LS86A, and the SN54S86 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN7486, SN74LS86A, and the SN74S86 are characterized for operation from 0°C to 70°C.

exclusive-OR logic

An exclusive-OR gate has many applications, some of which can be represented better by alternative logic symbols.

EXCLUSIVE-OR

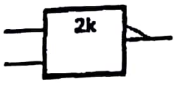


These are five equivalent Exclusive-OR symbols valid for an '86 or 'LS86A gate in positive logic; negation may be shown at any two ports.

LOGIC IDENTITY ELEMENT

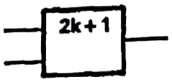
The output is active (low) if all inputs stand at the same logic level (i.e., A = B).

EVEN-PARITY



The output is active (low) if an even number of inputs (i.e., 0 or 2) are active.

ODD-PARITY ELEMENT



The output is active (high) if an odd number of inputs (i.e., only 1 of the 2) are active.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

'90A, 'LS90 . . . Decade Counters

'92A, 'LS92 . . . Divide By-Twelve Counters

'93A, 'LS93 . . . 4-Bit Binary Counters

TYPES	TYPICAL POWER DISSIPATION
'90A	145 mW
'92A, '93A	130 mW
'LS90, 'LS92, 'LS93	45 mW

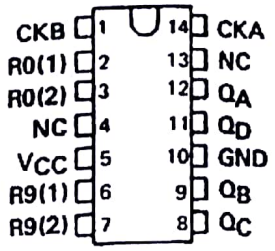
description

Each of these monolithic counters contains four master-slave flip-flops and additional gating to provide a divide-by-two counter and a three-stage binary counter for which the count cycle length is divide-by-five for the '90A and 'LS90, divide-by-six for the '92A and 'LS92, and the divide-by-eight for the '93A and 'LS93.

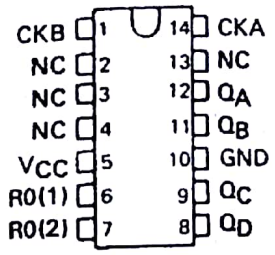
All of these counters have a gated zero reset and the '90A and 'LS90 also have gated set-to-nine inputs for use in BCD nine's complement applications.

To use their maximum count length (decade, divide-by-twelve, or four-bit binary) of these counters, the CKB input is connected to the Q_A output. The input count pulses are applied to CKA input and the outputs are as described in the appropriate function table. A symmetrical divide-by-ten count can be obtained from the '90A or 'LS90 counters by connecting the Q_D output to the CKA input and applying the input count to the CKB input which gives a divide-by-ten square wave at output Q_A.

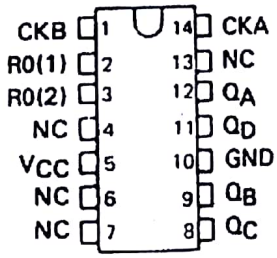
SN5490A, SN54LS90 . . . J OR W PACKAGE
SN7490A . . . N PACKAGE
SN74LS90 . . . D OR N PACKAGE
(TOP VIEW)



SN5492A, SN54LS92 . . . J OR W PACKAGE
SN7492A . . . N PACKAGE
SN74LS92 . . . D OR N PACKAGE
(TOP VIEW)



SN5493A, SN54LS93 . . . J OR W PACKAGE
SN7493 . . . N PACKAGE
SN74LS93 . . . D OR N PACKAGE
(TOP VIEW)



'90A, 'LS90
BCD COUNT SEQUENCE
(See Note A)

COUNT	OUTPUT			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H

'90A, 'LS90
BI-QUINARY (5-2)
(See Note B)

COUNT	OUTPUT			
	Q_A	Q_D	Q_C	Q_B
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	H	L	L	L
6	H	L	L	H
7	H	L	H	L
8	H	L	H	H
9	H	H	L	L

'92A, 'LS92
COUNT SEQUENCE
(See Note C)

COUNT	OUTPUT			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	H	L	L	L
7	H	L	L	H
8	H	L	H	L
9	H	L	H	H
10	H	H	L	L
11	H	H	L	H

'90A, 'LS90
RESET/COUNT FUNCTION TABLE

RESET INPUTS				OUTPUT			
$R_{0(1)}$	$R_{0(2)}$	$R_{9(1)}$	$R_{9(2)}$	Q_D	Q_C	Q_B	Q_A
H	H	L	X	L	L	L	L
H	H	X	L	L	L	L	L
X	X	H	H	H	L	L	H
X	L	X	L	COUNT			
L	X	L	X	COUNT			
L	X	X	L	COUNT			
X	L	L	X	COUNT			

'92A, 'LS92, '93A, 'LS93
RESET/COUNT FUNCTION TABLE

RESET INPUTS		OUTPUT			
$R_{0(1)}$	$R_{0(2)}$	Q_D	Q_C	Q_B	Q_A
H	H	L	L	L	L
L	X	COUNT			
X	L	COUNT			

'93A, 'LS93
COUNT SEQUENCE
(See Note C)

COUNT	OUTPUT			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

NOTES: A. Output Q_A is connected to input CKB for BCD count.
B. Output Q_D is connected to input CKA for bi-quinary count.
C. Output Q_A is connected to input CKB.
D. H = high level, L = low level, X = irrelevant

SDLS014

DECEMBER 1972—REVISED MARCH 1988

- Designed Specifically for High-Speed:
  Memory Decoders
  Data Transmission Systems
- 3 Enable Inputs to Simplify Cascading and/or Data Reception
- Schottky-Clamped for High Performance

description

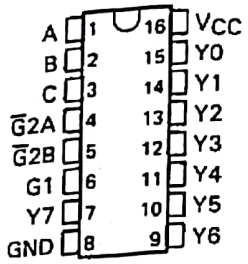
These Schottky-clamped TTL MSI circuits are designed to be used in high-performance memory decoding or data-routing applications requiring very short propagation delay times. In high-performance memory systems, these docoders can be used to minimize the effects of system decoding. When employed with high-speed memories utilizing a fast enable circuit, the delay times of these decoders and the enable time of the memory are usually less than the typical access time of the memory. This means that the effective system delay introduced by the Schottky-clamped system decoder is negligible.

The 'LS138, SN54S138, and SN74S138A decode one of eight lines dependent on the conditions at the three binary select inputs and the three enable inputs. Two active-low and one active-high enable inputs reduce the need for external gates or inverters when expanding. A 24-line decoder can be implemented without external inverters and a 32-line decoder requires only one inverter. An enable input can be used as a data input for demultiplexing applications.

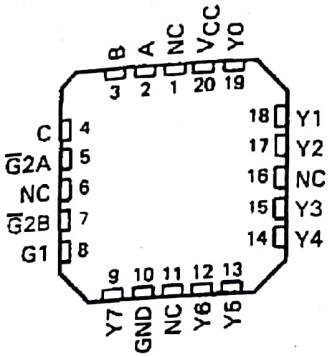
All of these decoder/demultiplexers feature fully buffered inputs, each of which represents only one normalized load to its driving circuit. All inputs are clamped with high-performance Schottky diodes to suppress line-ringing and to simplify system design.

The SN54LS138 and SN54S138 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN74LS138 and SN74S138A are characterized for operation from 0°C to 70°C.

SN54LS138, SN54S138 . . . J OR W PACKAGE
SN74LS138, SN74S138A . . . D OR N PACKAGE
(TOP VIEW)

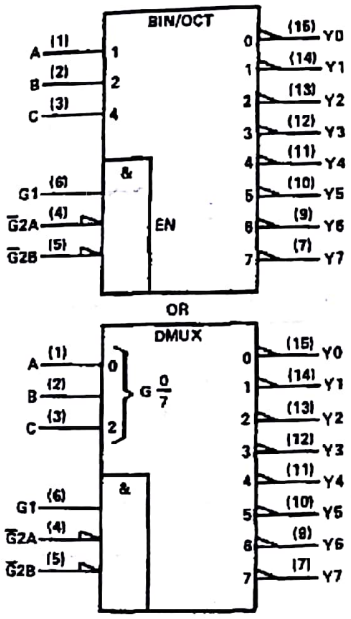


SN54LS138, SN54S138 . . . FK PACKAGE
(TOP VIEW)



NC—No internal connection

logic symbols†



†These symbols are in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

SN54LS139A, SN54S139, SN74LS139A, SN74S139A
DUAL 2-LINE TO 4-LINE DECODERS/DEMULTIPLEXERS

DECEMBER 1972 – REVISED MARCH 1988

- **Designed Specifically for High-Speed:**
  Memory Decoders
  Data Transmission Systems
- **Two Fully Independent 2- to 4-Line Decoders/Demultiplexers**
- **Schottky Clamped for High Performance**

description

These Schottky-clamped TTL MSI circuits are designed to be used in high-performance memory-decoding or data-routing applications requiring very short propagation delay times. In high-performance memory systems, these decoders can be used to minimize the effects of system decoding. When employed with high-speed memories utilizing a fast-enable circuit, the delay times of these decoders and the enable time of the memory are usually less than the typical access time of the memory. This means that the effective system delay introduced by the Schottky-clamped system decoder is negligible.

The circuit comprises two individual two-line to four-line decoders in a single package. The active-low enable input can be used as a data line in demultiplexing applications.

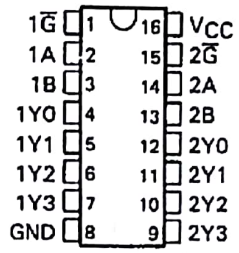
All of these decoders/demultiplexers feature fully buffered inputs, each of which represents only one normalized load to its driving circuit. All inputs are clamped with high-performance Schottky diodes to suppress line-ringing and to simplify system design. The SN54LS139A and SN54S139 are characterized for operation range of −55°C to 125°C. The SN74LS139A and SN74S139A are characterized for operation from 0°C to 70°C.

FUNCTION TABLE

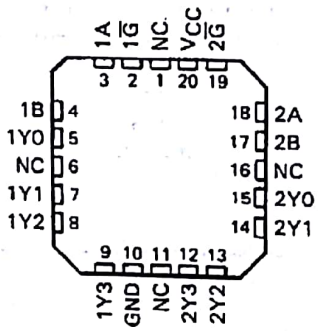
INPUTS			OUTPUTS			
ENABLE $\overline{G}$	SELECT B	SELECT A	Y0	Y1	Y2	Y3
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	L	H	H	L	H	H
L	H	L	H	H	L	H
L	H	H	H	H	H	L

H = high level, L = low level, X = irrelevant

SN54LS139A, SN54S139 . . . J OR W PACKAGE
SN74LS139A, SN74S139A . . . D OR N PACKAGE
(TOP VIEW)

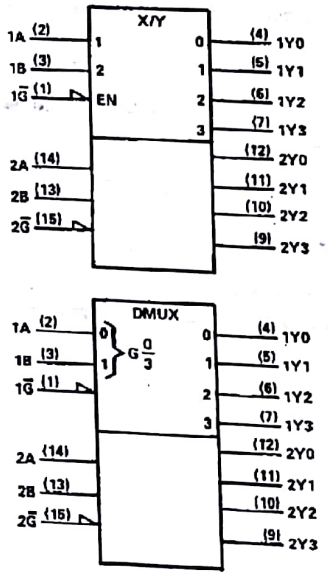


SN54LS139A, SN54S139 . . . FK PACKAGE
(TOP VIEW)



NC—No internal connection

logic symbols (alternatives)†



†These symbols are in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

TEXAS INSTRUMENTS
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

SDLS054

- '150 Selects One-of-Sixteen Data Sources
- Others Select One-of-Eight Data Sources
- All Perform Parallel-to-Serial Conversion
- All Permit Multiplexing from N Lines to One Line
- Also For Use as Boolean Function Generator
- Input-Clamping Diodes Simplify System Design
- Fully Compatible with Most TTL Circuits

TYPE	TYPICAL AVERAGE PROPAGATION DELAY TIME DATA INPUT TO W OUTPUT	TYPICAL POWER DISSIPATION
'150	13 ns	200 mW
'151A	8 ns	145 mW
'LS151	13 ns	30 mW
'S151	4.5 ns	225 mW

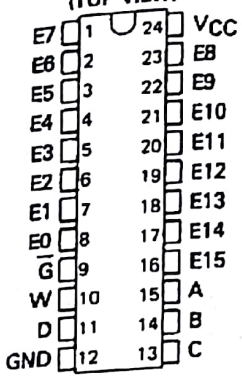
description

These monolithic data selectors/multiplexers contain full on-chip binary decoding to select the desired data source. The '150 selects one-of-sixteen data sources; the '151A, 'LS151, and 'S151 select one-of-eight data sources. The '150, '151A, 'LS151, and 'S151 have a strobe input which must be at a low logic level to enable these devices. A high level at the strobe forces the W output high, and the Y output (as applicable) low.

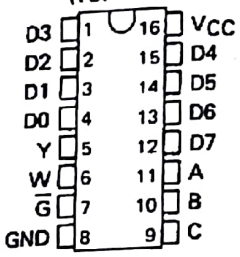
The '150 has only an inverted W output; the '151A, 'LS151, and 'S151 feature complementary W and Y outputs.

The '151A and '152A incorporate address buffers that have symmetrical propagation delay times through the complementary paths. This reduces the possibility of transients occurring at the output(s) due to changes made at the select inputs, even when the '151A outputs are enabled (i.e., strobe low).

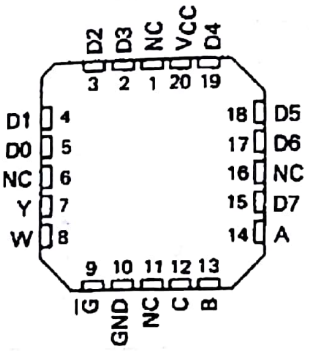
SN54150 . . . J OR W PACKAGE
SN74150 . . . N PACKAGE
(TOP VIEW)



SN54151A, SN54LS151, SN54S151 . . . J OR W PACKAGE
SN74151A . . . N PACKAGE
SN74LS151, SN74S151 . . . D OR N PACKAGE
(TOP VIEW)



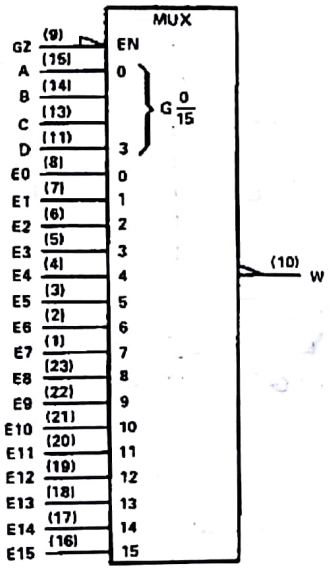
SN54LS151, SN54S151 . . . FK PACKAGE
(TOP VIEW)



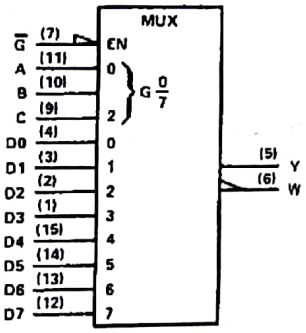
NC - No internal connection

logic symbols†

'150



'151A, 'LS151, 'S151



†These symbols are in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.
Pin numbers shown are D, J, N, and W packages.

'150

FUNCTION TABLE

INPUTS					OUTPUT
SELECT				STROBE	
D	C	B	A	$\overline{G}$	W
X	X	X	X	H	H
L	L	L	L	L	$\overline{E0}$
L	L	L	H	L	$\overline{E1}$
L	L	H	L	L	$\overline{E2}$
L	L	H	H	L	$\overline{E3}$
L	H	L	L	L	$\overline{E4}$
L	H	L	H	L	$\overline{E5}$
L	H	H	L	L	$\overline{E6}$
L	H	H	H	L	$\overline{E7}$
H	L	L	L	L	$\overline{E8}$
H	L	L	H	L	$\overline{E9}$
H	L	H	L	L	$\overline{E10}$
H	L	H	H	L	$\overline{E11}$
H	H	L	L	L	$\overline{E12}$
H	H	L	H	L	$\overline{E13}$
H	H	H	L	L	$\overline{E14}$
H	H	H	H	L	$\overline{E15}$

'151A, 'LS151, 'S151

FUNCTION TABLE

INPUTS				OUTPUTS	
SELECT			STROBE		
C	B	A	$\overline{G}$	Y	W
X	X	X	H	L	H
L	L	L	L	D0	$\overline{D0}$
L	L	H	L	D1	$\overline{D1}$
L	H	L	L	D2	$\overline{D2}$
L	H	H	L	D3	$\overline{D3}$
H	L	L	L	D4	$\overline{D4}$
H	L	H	L	D5	$\overline{D5}$
H	H	L	L	D6	$\overline{D6}$
H	H	H	L	D7	$\overline{D7}$

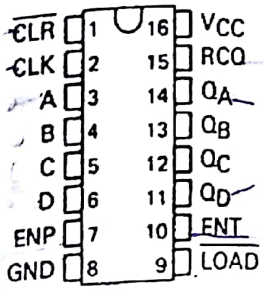
H = high level, L = low level, X = irrelevant
$\overline{E0}$, $\overline{E1}$ . . . $\overline{E15}$ = the complement of the level of the respective E input
D0, D1 . . . D7 = the level of the D respective input

'160,'161,'LS160A,'LS161A . . . SYNCHRONOUS COUNTERS WITH DIRECT CLEAR
'162,'163,'LS162A,'LS163A,'S162,'S163 . . . FULLY SYNCHRONOUS COUNTERS

- Internal Look-Ahead for Fast Counting
- Carry Output for n-Bit Cascading
- Synchronous Counting
- Synchronously Programmable
- Load Control Line
- Diode-Clamped Inputs

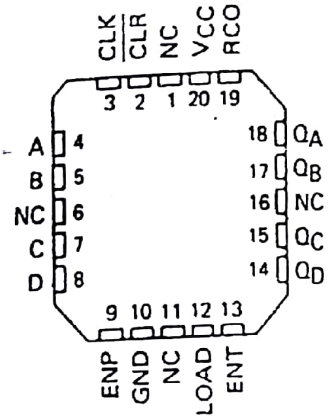
TYPE	TYPICAL PROPAGATION TIME, CLOCK TO Q OUTPUT	TYPICAL MAXIMUM CLOCK FREQUENCY	TYPICAL POWER DISSIPATION
'160 thru '163	14 ns	32 MHz	305 mW
'LS162A thru 'LS163A	14 ns	32 MHz	93 mW
'S162 and 'S163	9 ns	70 MHz	475 mW

SERIES 54', 54LS' 54S' . . . J OR W PACKAGE
SERIES 74' . . . N PACKAGE
SERIES 74LS', 74S' . . . D OR N PACKAGE
(TOP VIEW)



NC—No internal connection

SERIES 54LS', 54S' . . . FK PACKAGE
(TOP VIEW)



NC—No internal connection

description

These synchronous, presettable counters feature an internal carry look-ahead for application in high-speed counting designs. The '160,'162,'LS160A,'LS162A, and 'S162 are decade counters and the '161,'163,'LS161A,'LS163A, and 'S163 are 4-bit binary counters. Synchronous operation is provided by having all flip-flops clocked simultaneously so that the outputs change coincident with each other when so instructed by the count-enable inputs and internal gating. This mode of operation eliminates the output counting spikes that are normally associated with asynchronous (ripple clock) counters, however counting spikes may occur on the (RCO) ripple carry output. A buffered clock input triggers the four flip-flops on the rising edge of the clock input waveform.

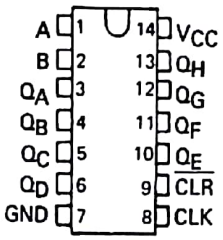
These counters are fully programmable; that is, the outputs may be preset to either level. As presetting is synchronous, setting up a low level at the load input disables the counter and causes the outputs to agree with the setup data after the next clock pulse regardless of the levels of the enable inputs. Low-to-high transitions at the load input of the '160 thru '163 should be avoided when the clock is low if the enable inputs are high at or before the transition. This restriction is not applicable to the 'LS160A thru 'LS163A or 'S162 or 'S163. The clear function for the '160, '161,'LS160A, and 'LS161A is asynchronous and a low level at the clear input sets all four of the flip-flop outputs low regardless of the levels of clock, load, or enable inputs. The clear function for the '162,'163,'LS162A,'LS163A, 'S162, and 'S163 is synchronous and a low level at the clear input sets all four of the flip-flop outputs low after the next clock pulse, regardless of the levels of the enable inputs. This synchronous clear allows the count length to be modified easily as decoding the maximum count desired can be accomplished with one external NAND gate. The gate output is connected to the clear input to synchronously clear the counter to 0000 (LLLL). Low-to-high transitions at the clear input of the '162 and '163 should be avoided when the clock is low if the enable and load inputs are high at or before the transition.

MARCH 1974 – REVISED MARCH 1988

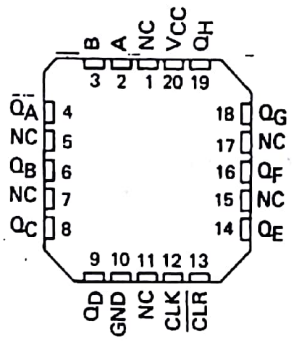
- Gated Serial Inputs
- Fully Buffered Clock and Serial Inputs
- Asynchronous Clear

TYPE	TYPICAL MAXIMUM CLOCK FREQUENCY	TYPICAL POWER DISSIPATION
'164	36 MHz	21 mW per bit
'LS164	36 MHz	10 mW per bit

SN54164, SN54LS164 . . . J OR W PACKAGE
SN74164 . . . N PACKAGE
SN74LS164 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS164 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

description

These 8-bit shift registers feature gated serial inputs and an asynchronous clear. The gated serial inputs (A and B) permit complete control over incoming data as a low at either input inhibits entry of the new data and resets the first flip-flop to the low level at the next clock pulse. A high-level input enables the other input which will then determine the state of the first flip-flop. Data at the serial inputs may be changed while the clock is high or low, but only information meeting the setup-time requirements will be entered. Clocking occurs on the low-to-high-level transition of the clock input. All inputs are diode-clamped to minimize transmission-line effects.

The SN54164 and SN54LS164 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN74164 and SN74LS164 are characterized for operation from 0°C to 70°C.

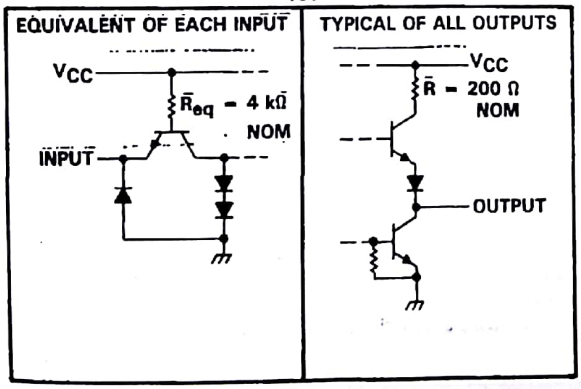
FUNCTION TABLE

INPUTS				OUTPUTS			
CLEAR	CLOCK	A	B	Q_A	Q_B	...	Q_H
L	X	X	X	L	L		L
H	L	X	X	Q_{A0}	Q_{B0}		Q_{H0}
H	↑	H	H	H	Q_{An}		Q_{Gn}
H	↑	L	X	L	Q_{An}		Q_{Gn}
H	↑	X	L	L	Q_{An}		Q_{Gn}

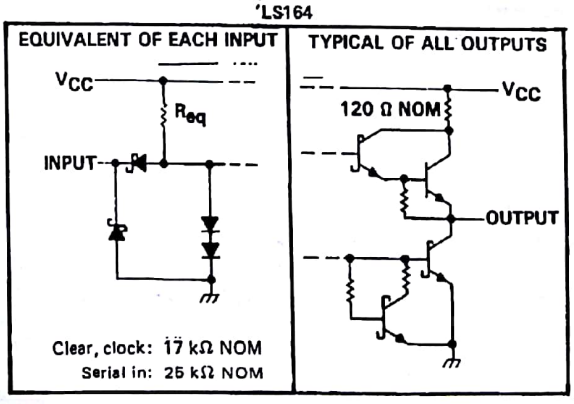
H = high level (steady state), L = low level (steady state)
X = irrelevant (any input, including transitions)
↑ = transition from low to high level.
Q_{A0}, Q_{B0}, Q_{H0} = the level of Q_A, Q_B, or Q_H, respectively, before the indicated steady-state input conditions were established.
Q_{An}, Q_{Gn} = the level of Q_A or Q_G before the most-recent ↑ transition of the clock; indicates a one-bit shift.

schematics of inputs and outputs

'164



'LS164



OCTOBER 1976 — REVISED MARCH 1988

- **Synchronous Load**
- **Direct Overriding Clear**
- **Parallel to Serial Conversion**

TYPE	TYPICAL MAXIMUM CLOCK FREQUENCY	TYPICAL POWER DISSIPATION
'166	35 MHz	360 mW
'LS166A	35 MHz	100 mW

description

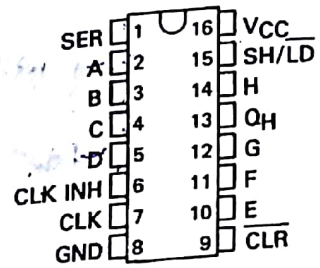
The '166 and 'LS166A 8-bit shift registers are compatible with most other TTL logic families. All '166 and 'LS166A inputs are buffered to lower the drive requirements to one Series 54/74 or Series 54LS/74LS standard load, respectively. Input clamping diodes minimize switching transients and simplify system design.

These parallel-in or serial-in, serial-out shift registers have a complexity of 77 equivalent gates on a monolithic chip. They feature gated clock inputs and an overriding clear input. The parallel-in or serial-in modes are established by the shift/load input. When high, this input enables the serial data input and couples the eight flip-flops for serial shifting with each clock pulse. When low, the parallel (broadside) data inputs are enabled and synchronous loading occurs on the next clock pulse. During parallel loading, serial data flow is inhibited. Clocking is accomplished on the low-to-high-level edge of the clock pulse through a two-input positive NOR gate permitting one input to be used as a clock-enable or clock-inhibit function. Holding either of the clock inputs high inhibits clocking; holding either low enables the other clock input. This, of course, allows the system clock to be free-running and the register can be stopped on command with the other clock input. The clock inhibit input should be changed to the high level only while the clock input is high. A buffered, direct clear input overrides all other inputs, including the clock, and sets all flip-flops to zero.

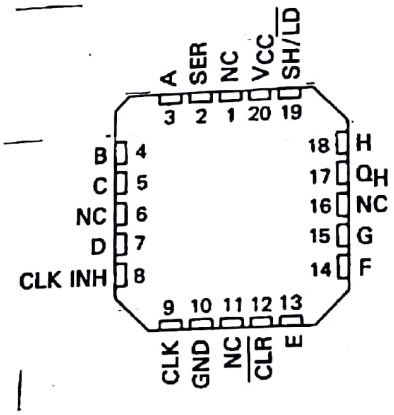
FUNCTION TABLE

INPUTS						INTERNAL OUTPUTS		OUTPUT
CLEAR	SHIFT/LOAD	CLOCK INHIBIT	CLOCK	SERIAL	PARALLEL A...H	Q_A	Q_B	Q_H
L	X	X	X	X	X	L	L	L
H	X	L	L	X	X	Q_{A0}	Q_{B0}	Q_{H0}
H	L	L	↑	X	a...h	a	b	h
H	H	L	↑	H	X	H	Q_{An}	Q_{Gn}
H	H	L	↑	L	X	L	Q_{An}	Q_{Gn}
H	X	H	↑	X	X	Q_{A0}	Q_{B0}	Q_{H0}

SN54166, SN54LS166A . . . J OR W PACKAGE
SN74166 . . . N PACKAGE
SN74LS166A . . . D OR N PACKAGE
(TOP VIEW)

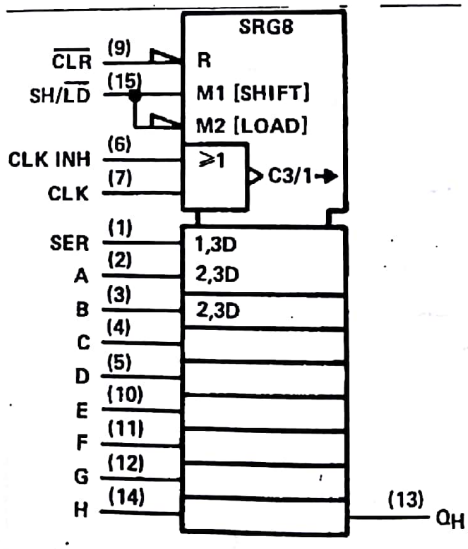


SN54LS166A . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

logic symbol†



†This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

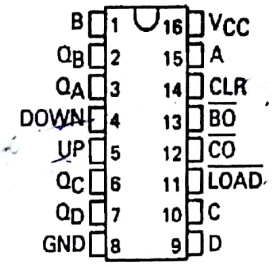
SN54192, SN54193, SN54LS192, SN54LS193, SN74192, SN74193, SN74LS192, SN74LS193
SYNCHRONOUS 4-BIT UP/DOWN COUNTERS (DUAL CLOCK WITH CLEAR)

SDLS074 – DECMEBER 1972 – REVISED MARCH 1988

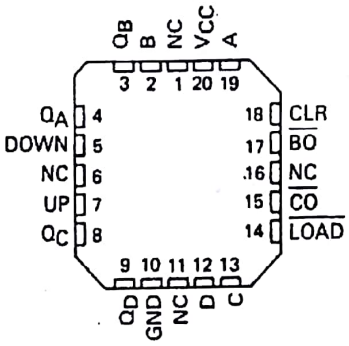
- Cascading Circuitry Provided Internally
- Synchronous Operation
- Individual Preset to Each Flip-Flop
- Fully Independent Clear Input

TYPES	TYPICAL MAXIMUM COUNT FREQUENCY	TYPICAL POWER DISSIPATION
'192,'193	32 MHz	325 mW
'LS192,'LS193	32 MHz	95 mW

SN54192, SN54193, SN54LS192, SN54LS193 . . . J OR W PACKAGE
SN74192, SN74193 . . . N PACKAGE
SN74LS192, SN74LS193 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS192, SN54LS193 . . . FK PACKAGE
(TOP VIEW)



NC - No internal connection

description

These monolithic circuits are synchronous reversible (up/down) counters having a complexity of 55 equivalent gates. The '192 and 'LS192 circuits are BCD counters and the '193 and 'LS193 are 4-bit binary counters. Synchronous operation is provided by having all flip-flops clocked simultaneously so that the outputs change coincidently with each other when so instructed by the steering logic. This mode of operation eliminates the output counting spikes which are normally associated with asynchronous (ripple-clock) counters.

The outputs of the four master-slave flip-flops are triggered by a low-to-high-level transition of either count (clock) input. The direction of counting is determined by which count input is pulsed while the other count input is high.

All four counters are fully programmable; that is, each output may be preset to either level by entering the desired data at the data inputs while the load input is low. The output will change to agree with the data inputs independently of the count pulses. This feature allows the counters to be used as modulo-N dividers by simply modifying the count length with the preset inputs.

A clear input has been provided which forces all outputs to the low level when a high level is applied. The clear function is independent of the count and load inputs. The clear, count, and load inputs are buffered to lower the drive requirements. This reduces the number of clock drivers, etc., required for long words.

These counters were designed to be cascaded without the need for external circuitry. Both borrow and carry outputs are available to cascade both the up- and down-counting functions. The borrow output produces a pulse equal in width to the count-down input when the counter underflows. Similarly, the carry output produces a pulse equal in width to the count-up input when an overflow condition exists. The counters can then be easily cascaded by feeding the borrow and carry outputs to the count-down and count-up inputs respectively of the succeeding counter.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

	SN54'	SN54LS'	SN74'	SN74LS'	UNIT
Supply voltage, V_{CC} (see Note 1)	7	7	7	7	V
Input voltage	5.5	7	5.5	7	V
Operating free-air temperature range	−55 to 125		0 to 70		°C
Storage temperature range	−65 to 150		−65 to 150		°C

NOTE 1: Voltage values are with respect to network ground terminal.

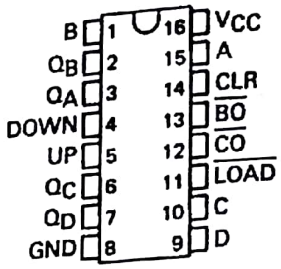
SYNCHRONOUS 4-BIT UP/DOWN COUNTERS (DUAL CLOCK WITH CLEAR)

SDLS074 – DECMEBER 1972 – REVISED MARCH 1988

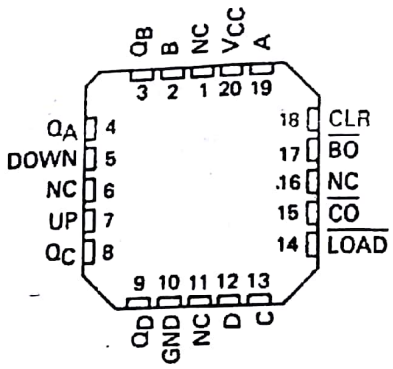
- Cascading Circuitry Provided Internally
- Synchronous Operation
- Individual Preset to Each Flip-Flop
- Fully Independent Clear Input

TYPES	TYPICAL MAXIMUM COUNT FREQUENCY	TYPICAL POWER DISSIPATION
'192,'193	32 MHz	325 mW
'LS192,'LS193	32 MHz	95 mW

SN54192, SN54193, SN54LS192, SN54LS193 . . . J OR W PACKAGE
SN74192, SN74193 . . . N PACKAGE
SN74LS192, SN74LS193 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS192, SN54LS193 . . . FK PACKAGE
(TOP VIEW)



NC - No internal connection

description

These monolithic circuits are synchronous reversible (up/down) counters having a complexity of 55 equivalent gates. The '192 and 'LS192 circuits are BCD counters and the '193 and 'LS193 are 4-bit binary counters. Synchronous operation is provided by having all flip-flops clocked simultaneously so that the outputs change coincidently with each other when so instructed by the steering logic. This mode of operation eliminates the output counting spikes which are normally associated with asynchronous (ripple-clock) counters.

The outputs of the four master-slave flip-flops are triggered by a low-to-high-level transition of either count (clock) input. The direction of counting is determined by which count input is pulsed while the other count input is high.

All four counters are fully programmable; that is, each output may be preset to either level by entering the desired data at the data inputs while the load input is low. The output will change to agree with the data inputs independently of the count pulses. This feature allows the counters to be used as modulo-N dividers by simply modifying the count length with the preset inputs.

A clear input has been provided which forces all outputs to the low level when a high level is applied. The clear function is independent of the count and load inputs. The clear, count, and load inputs are buffered to lower the drive requirements. This reduces the number of clock drivers, etc., required for long words.

These counters were designed to be cascaded without the need for external circuitry. Both borrow and carry outputs are available to cascade both the up- and down-counting functions. The borrow output produces a pulse equal in width to the count-down input when the counter underflows. Similarly, the carry output produces a pulse equal in width to the count-up input when an overflow condition exists. The counters can then be easily cascaded by feeding the borrow and carry outputs to the count-down and count-up inputs respectively of the succeeding counter.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

	SN54'	SN54LS'	SN74'	SN74LS'	UNIT
Supply voltage, V_{CC} (see Note 1)	7	7	7	7	V
Input voltage	5.5	7	5.5	7	V
Operating free-air temperature range	−55 to 125		0 to 70		°C
Storage temperature range	−65 to 150		−65 to 150		°C

NOTE 1: Voltage values are with respect to network ground terminal.

SN54194, SN54LS194A, SN54S194, SN74194, SN74LS194A, SN74S194
4-BIT BIDIRECTIONAL UNIVERSAL SHIFT REGISTERS

MARCH 1974—REVISED MARCH 1988

- Parallel Inputs and Outputs
- Four Operating Modes:
  - Synchronous Parallel Load
  - Right Shift
  - Left Shift
  - Do Nothing
- Positive Edge-Triggered Clocking
- Direct Overriding Clear

TYPE	TYPICAL MAXIMUM CLOCK FREQUENCY	TYPICAL POWER DISSIPATION
'194	36 MHz	195 mW
'LS194A	36 MHz	75 mW
'S194	105 MHz	425 mW

description

These bidirectional shift registers are designed to incorporate virtually all of the features a system designer may want in a shift register. The circuit contains 46 equivalent gates and features parallel inputs, parallel outputs, right-shift and left-shift serial inputs, operating-mode-control inputs, and a direct overriding clear line. The register has four distinct modes of operation, namely:

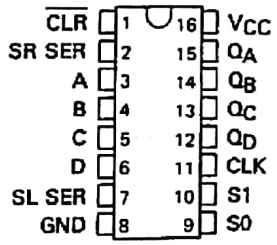
Inhibit clock (do nothing)
Shift right (in the direction Q_A toward Q_D)
Shift left (in the direction Q_D toward Q_A)
Parallel (broadside) load

Synchronous parallel loading is accomplished by applying the four bits of data and taking both mode control inputs, S0 and S1, high. The data are loaded into the associated flip-flops and appear at the outputs after the positive transition of the clock input. During loading, serial data flow is inhibited.

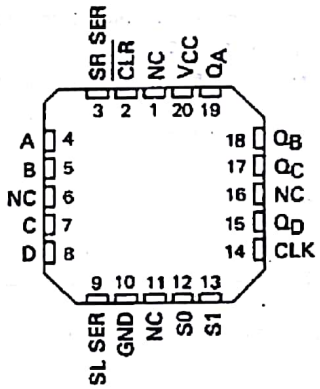
Shift right is accomplished synchronously with the rising edge of the clock pulse when S0 is high and S1 is low. Serial data for this mode is entered at the shift-right data input. When S0 is low and S1 is high, data shifts left synchronously and new data is entered at the shift-left serial input.

Clocking of the shift register is inhibited when both mode control inputs are low. The mode controls of the SN54194/SN74194 should be changed only while the clock input is high.

SN54194, SN54LS194A, SN54S194 . . . J OR W PACKAGE
SN74194 . . . N PACKAGE
SN74LS194A, SN74S194 . . . D OR N PACKAGE
(TOP VIEW)

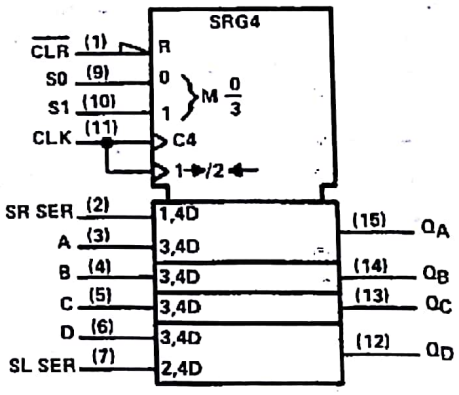


SN54LS194A, SN54S194 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

logic symbol†



†This symbol is in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

SN54LS240, SN54LS241, SN54LS244, SN54S240, SN54S241, SN54S244
SN74LS240, SN74LS241, SN74LS244, SN74S240, SN74S241, SN74S244
OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

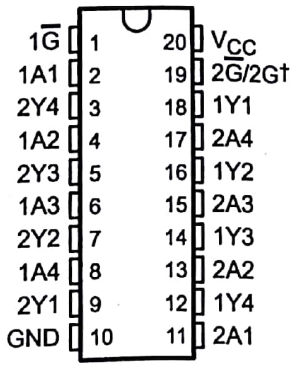
SDLS144B – APRIL 1985 – REVISED FEBRUARY 2002

- 3-State Outputs Drive Bus Lines or Buffer Memory Address Registers
- PNP Inputs Reduce DC Loading
- Hysteresis at Inputs Improves Noise Margins

description

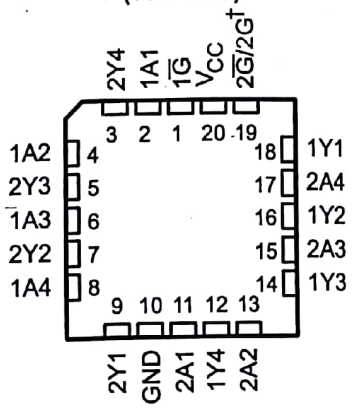
These octal buffers and line drivers are designed specifically to improve both the performance and density of three-state memory address drivers, clock drivers, and bus-oriented receivers and transmitters. The designer has a choice of selected combinations of inverting and noninverting outputs, symmetrical, active-low output-control ($\overline{G}$) inputs, and complementary output-control (G and $\overline{G}$) inputs. These devices feature high fan-out, improved fan-in, and 400-mV noise margin. The SN74LS' and SN74S' devices can be used to drive terminated lines down to 133 Ω.

SN54LS', SN54S' . . . J OR W PACKAGE
SN74LS240, SN74LS244 . . . DB, DW, N, OR NS PACKAGE
SN74LS241 . . . DW, N, OR NS PACKAGE
SN74S' . . . DW OR N PACKAGE
(TOP VIEW)



† 2G for 'LS241 and 'S241 or $2\overline{G}$ for all other drivers.

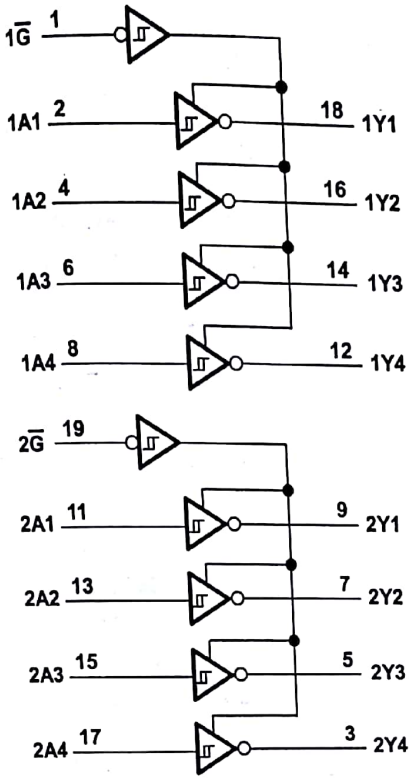
SN54LS', SN54S' . . . FK PACKAGE
(TOP VIEW)



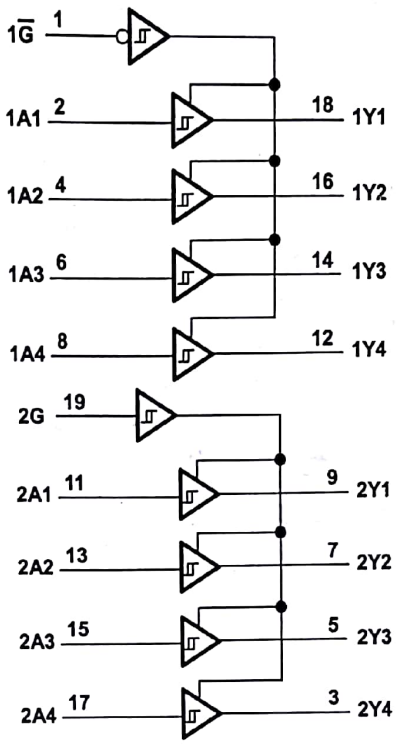
† 2G for 'LS241 and 'S241 or $2\overline{G}$ for all other drivers.

logic diagram

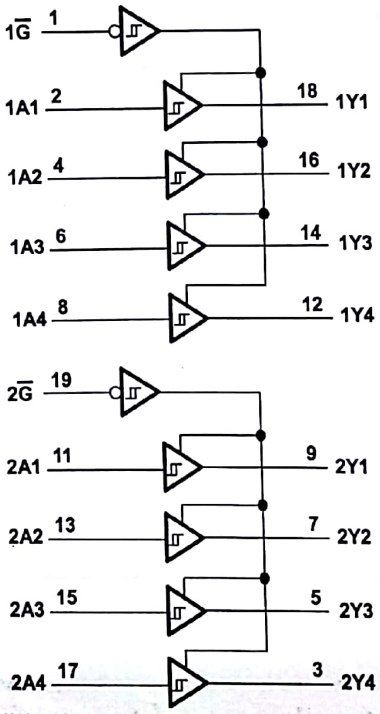
'LS240, 'S240



'LS241, 'S241



'LS244, 'S244



Pin numbers shown are for DB, DW, J, N, NS, and W packages.

م. أشرف سقيرات
مختبر المنطق الرقمي
للحاسوب

PreLabs

By Eng. Ashraf Hasan Al-Bqerat (Feb 2014)

PreLab Exp.1:

1- Obtain logic diagram, the truth table and needed ICs of the following function: $F(A,B,C)=\overline{A}.C+B\overline{C}$. Using AND,OR, and NOT gates.

PreLab Exp.2:

1- By using Boolean Algebra technique, simplify the following function as much as possible: $F(X,Y,Z)=\overline{X}.Y.Z+\overline{X}.\overline{Z}$. Then obtain the Logic diagram and the truth table for this function.

2- Use K-map to simplify the Boolean function F together with the don't care conditions d in Sum of products $F(X,Y,Z,W)=\sum(3,5,7,13)+\sum d(6,11,14,15)$. Obtain the logic diagram for SOP.

3- Obtain the truth table of the following function:- $F(A,B,C)=A\overline{B}+\overline{ABC}$. Then determine the needed ICs and obtain the logic diagram if you use the NAND and inverter gates only to implement this function.

PreLab Exp.3:

1- Use an 8:1 Multiplexer to implement the function $F(X,Y,Z,W)=\sum(0,2,4,6,9,10,13,14)$. Obtain the implementation table, logic diagram and required ICs to implement this function.

2- Show how a 3:8 decoder is used to determine the function. $F(X,Y,Z)=\sum(0,3,7)$

PreLab Exp.4:

1- Build a **Adder** circuit which add two 4-bit numbers by using a 4-bit parallel adder. (Use the 7483).

PreLab Exp.5:

1- The state transition table of a state machine is given below. Design this machine using JK flip-flops.

Present State		Input	Next State	
A	B	X	A*	B*
0	1	1	1	1
1	1	1	1	0
1	0	1	0	0
0	0	0	1	1
1	1	0	0	1

Table 5.4

PreLab Exp.6:

1- By using 3 individuals negative -edge JK flip-flops. Design a synchronous counter with the counting sequence 2,3,4,5 then return back to 2.