paMinistry of Higher Education and Scientific Research

Badji Mokhtar Annaba University

Faculty of Technology

وزارة التعليم العالي و البحث العلمي

جـامعة باجـي مختـار- عنابـة

كليـة الــتكنـولوجيا

Department of Electrical engineering

Educational Handout

Title:

Practical work on combinatorial and sequential logic

Practical work intended for students of :

Specialization & level : Electrical Engineering, 2nd Year Common Core Science and Technology (ST 2 ,S4)

Written by Dr. Abdelhamid KSENTINI

Year : 2025-2026

Table of Contents

PW4 : Study and realization of an arithmetic combinational logic circuit

PW 5 : Study and creation of a combinational logic circuit

PW6 : Study and creation of a combinational logic circuit

PW7 : Study and implementation of counter circuits

PW 8 : Study and creation of registers

List of figures **Page**

List of tables **Page**

Lists of symbols

TTL:	Transistor Transistor Logic
COMS:	Complementary Metal-Oxide-Semiconductor
MUX :	Multiplexer
DMUX :	Demultiplexer
DC:	Direct current
Vcc:	Positive supply voltage
Vdd:	Drain to drain voltage
XOR:	OR exclusif
R:	Resistance
A,b,c:	Inputs
F1,F2,F3:	Outputs
0:	False
1:	True
SWi :	Switchs
L :	Indicator

Semestre: 4
Unité d'enseignement: UEM 2.2
Matière 3: TP Logique combinatoire et séquentielle
VHS: 22h30 (TP: 1h30)
Crédits: 2
Coefficient: 1

Objectifs de l'enseignement:

Consolider les connaissances acquises pendant le cours de la matière "Logique Combinatoire et Séquentielle" par des travaux pratiques pour mieux comprendre et assimiler le contenu de cette matière.

Connaissances préalables recommandées

Logique Combinatoire et Séquentielle.

Contenu de la matière :

L'enseignant choisit parmi cette liste de TP entre 4 et 6 TP à réaliser et traitant les deux types de circuits logiques (combinatoire et séquentiel).

TP1 : Technologie des circuits intégrés TTL et CMOS.
Appréhender et tester les différentes portes logiques

TP2 : Simplification des équations logiques par la pratique
Découvrir les règles de simplification des équations dans l'algèbre de Boole par la pratique

TP3 : Etude et réalisation de fonctions logiques combinatoires usuelles
Exemple : les circuits d'aiguillage (MUX, DMUX), les circuits de codage et de décodage, …

TP4 : Etude et réalisation d'un circuit combinatoire arithmétique
Réalisation d'un circuit additionneur et /ou soustracteur de 2 nombres binaires à 4 bits.

TP5 : Etude et réalisation d'un circuit combinatoire logique
Réalisation d'une fonction logique à l'aide de portes logiques. Exemple un afficheur à 7 segments et/ou un générateur du complément à 2 d'un nombre à 4 bits et/ou générateur du code de Gray à 4 bits, …

TP6 : Etude et réalisation d'un circuit combinatoire logique
Etude complète (Table de vérité, Simplification, Logigramme, Montage pratique et Essais) d'un circuit combinatoire à partir d'un cahier de charge.

TP7 : Etude et réalisation de circuits compteurs
Circuits compteurs asynchrones incomplets à l'aide de bascules, Circuits compteurs synchrones à cycle
irrégulier à l'aide de bascules

TP8 : Etude et réalisation de registres

Mode d'évaluation :

Contrôle continu : 100 %

Références bibliographiques:

1. J. Letocha, Introduction aux circuits logiques, Edition Mc-Graw Hill.
2. J.C. Lafont, Cours et problèmes d'électronique numérique, 124 exercices avec solutions, Edition Ellipses.

❖ Introduction

The combinational and sequential logic (CSL) module is based on essential prerequisites such as: basic concepts relating to integrated circuits, logic gates, the design of combinational logic functions, as well as the creation of arithmetic, logic and counter combinational circuits.

❖ Module description:

The objective of this module is to familiarize the student with the manipulation and implementation of different logic gates, as well as with the wiring of various combinational and sequential circuits, constituting a fundamental discipline of electronics.

This teaching material has been developed in accordance with the program of the course in combinatorial and sequential logic (LCS) included in the Bachelor's degree course in several specialties of the electrical engineering department (**UEM 2.2**).

❖ Some Tips:

Preparation of the practical work (PW)

Preparing for the lab session is very important.
- You must have prepared your work before coming to the lab session.
- A practical work session is not a button-pressing session, nor a stack of curves without legend or commentary.

Work during the practical session :

a. The material :

You will have a workbench. Each workbench is equipped with at least:

- A stabilized 5V DC power supply.
- A signal generator.
- A test model.
- Connection cables.

b. The realization :

- The work will be carried out in pairs or threes. Behavior during the practical work, including independence, problem-solving skills, and efficiency, will be taken into account in the assessment.

To avoid wiring errors, it is very important to:

- Draw the circuit diagram, indicating the pin numbers,
- Check that all circuits used are powered (a common mistake is to have one or more circuits without power).

The report must include for each practical work :

- A theoretical section,
- Schematics (flowcharts) with standardized symbols and pinouts (pin numbers).
- An analysis of the results obtained.

✓ Finally, the presentation itself of any work is important.

Control (Laboratory work defense):

The assessment will take place at the end of the practical work. It lasts **1.5 hours** and will be individual.

Final grade of the practical work:

The final grade awarded is the average of the lab preparation grade, participation (effectiveness during the lab), the report, and the assessment.

This brochure of practical work in combinatorial and sequential logic is intended for second-year students (LMD) in Science and Technology in ELT.

The brochure includes 8 PWs distributed as follows:

PW1: TTL and CMOS Integrated Circuit Technology

PW2: Simplifying Logic Equations Through Practice

PW3: Study and Implementation of usual common Combinational Logic Functions

PW4: Study and Implementation of an Arithmetic Combinational Logic Circuit

PW5: Study and Implementation of a Combinational Logic Circuit

PW6: Study and Implementation of a Combinational Logic Circuit

PW7: Study and Implementation of Counter Circuits

PW8: Study and Implementation of Registers

PW 1: TTL and CMOS integrated circuit technology

1. Purpose of the manipulation :

1) Understand basic digital integrated circuits and their uses in building combinational and sequential logic circuits.
2) Verify the truth tables of basic logic gates using the corresponding integrated circuits and manipulate certain rules of Boolean algebra.
3) Build combinational circuits using basic logic gates.

2. Theoretical reminder:

- Logic integrated circuits are a collection of resistors, diodes, and transistors fabricated directly into a semiconductor substrate.
- Logic integrated circuits (ICs) are often referred to by the complexity of the circuit they carry, measured by the number of equivalent logic gates embedded in the substrate.
- Digital circuits (logic circuits) are designed to provide output voltages within the voltage ranges corresponding to the binary values 1 and 0. These circuits respond in a predictable manner to input voltages within defined ranges for 0 and 1. The way in which a digital circuit responds to an input signal is called the circuit's logic.
- Digital circuits (logic circuits) are manufactured using several integrated circuit manufacturing technologies, the most common being TTL, CMOS, NMOS, and ECL. Each is distinguished by the type of circuit used to perform the desired logic operation. TTL (Transistor – Transistor – Logic) circuits use bipolar transistors as the main circuit elements, while CMOS (Complementary – Metal – Oxide – Semiconductor) technologies mainly employ enhancement-mode transistors (MOSFETs) (field-effect transistors), and ECL (Emitter-Coupled – Logic) technology enables high switching speeds.

  An integrated circuit mainly has 3 types of pins:

  - ***Power pins.***
  - ***Input pins:*** Used to receive voltages corresponding to logic levels **0** and **1.**
  - ***Output pins:*** Produce HIGH or LOW logic level voltages **(0** and **1).**

Note well:

- Avoid connecting the outputs to a power supply terminal or any other output or voltage.
- For **TTL i**ntegrated circuits, **Vcc** is set to **+5V**.
- For **CMOS** integrated circuits, **Vdd** is between **+3V** and **+18V**.

✓ **Industrial applications of TTL integrated circuits :**

- Programmable logic controllers (PLCs).
- Control and regulation systems.
- Test and measurement equipment.
- Smart sensors.
- Signal conversion and processing.

3. Required equipment :

KL-31001 Logic Digital Lab Simulation Model; **KL-33002** Module; Connecting Wires; Oscilloscope.

Fig.1 KL-31001 Logique Digital Lab

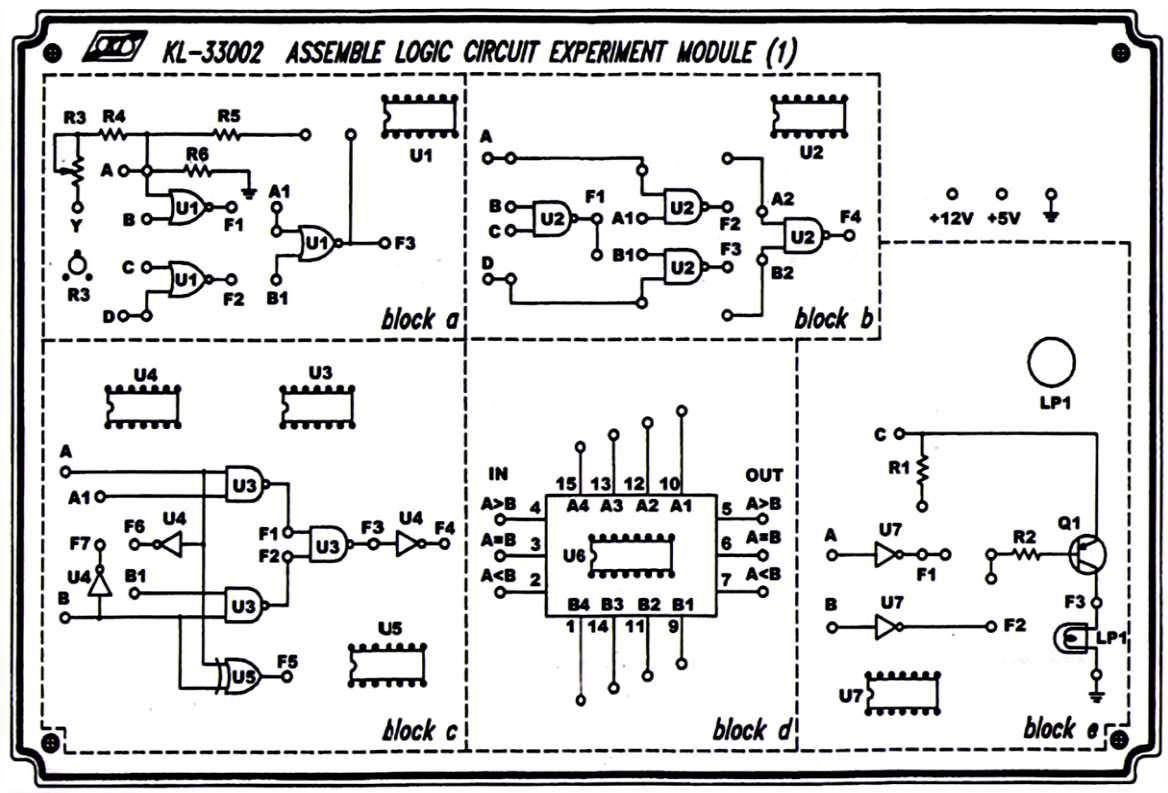


Fig.2 KL-33002 module

Let block a of module KL33002 be:

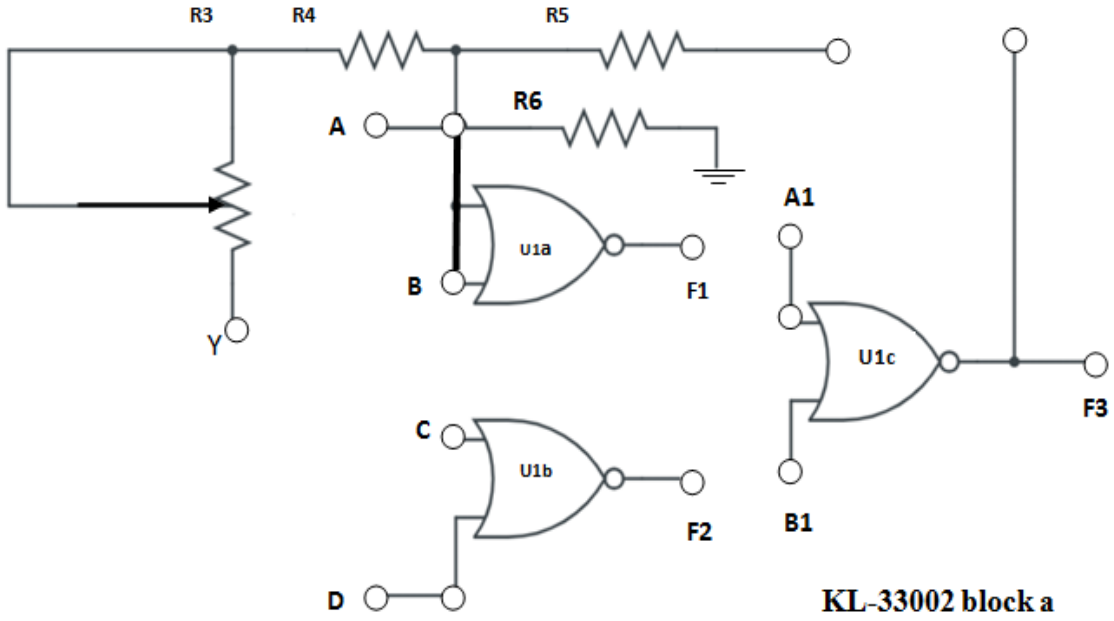


Fig.3 Module KL-33002 block a

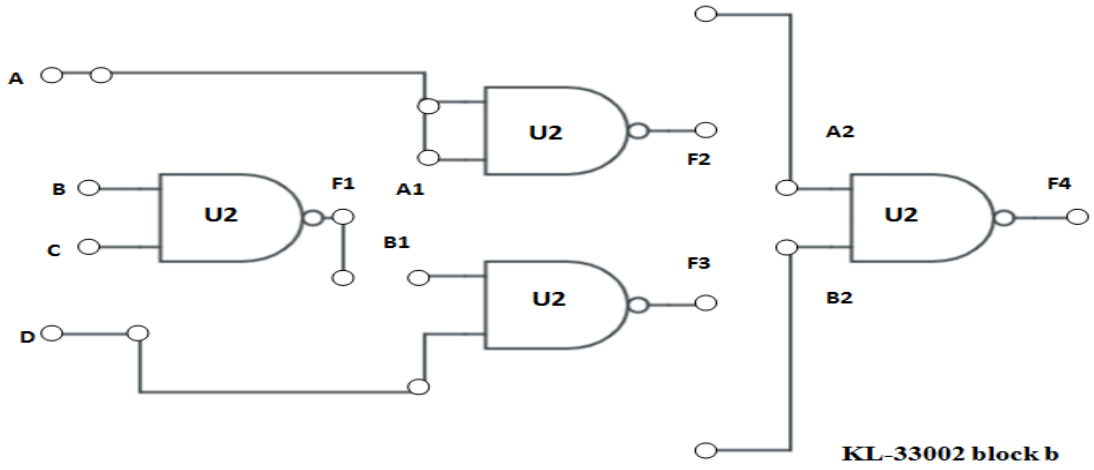


Fig.4 Module KL-33002 block b

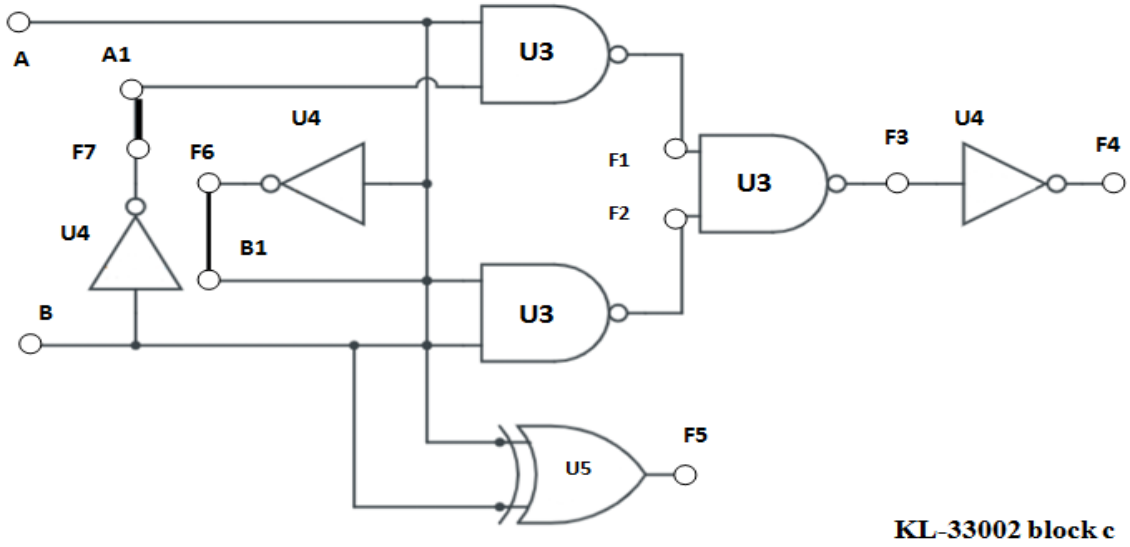
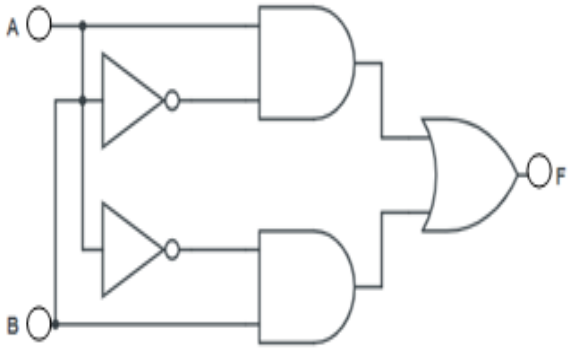


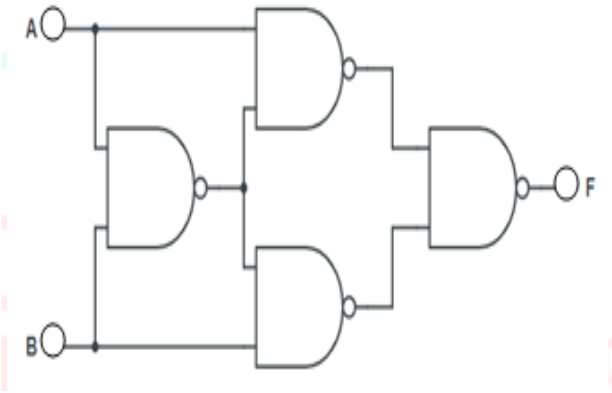
Fig.5 Module KL-33002 block c

4. Work requested :

1) Using the three available modules, establish the different truth tables of the basic logic gates: <<NOT>> gate (NOT); <<AND>> gate (AND); <<OR>> gate (OR); <<NOR>> gate (NOT-OR); <<NAND>> gate (NOT-AND).

2) Using the basic logic gates, construct the combinational logic circuit in the figure below and establish its truth table, which is that of an XOR gate constructed with basic NOT; NAND; AND and OR gates.

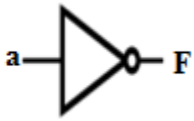




5. *Symbolic representation of logic gates :*

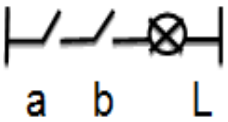
Logic gate AND

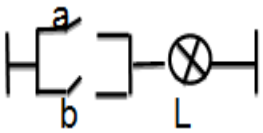
Logic gate OR

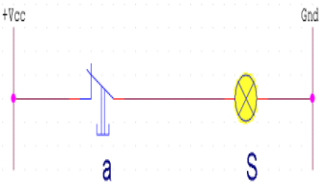


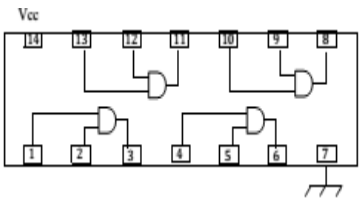
Logic gate NOT

5.1. Electrical Circuit :

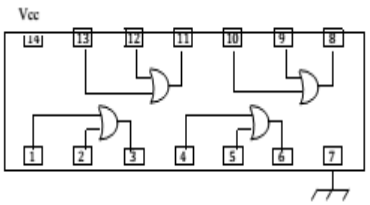




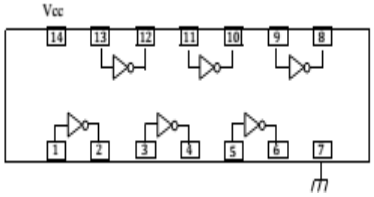




Integrated circuit 7408



Integrated circuit 7432



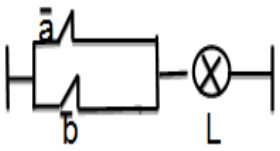
Integrated circuit7404

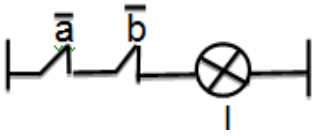
Logic gate NAND

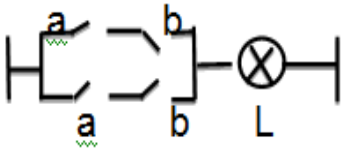
Logic gate NOR

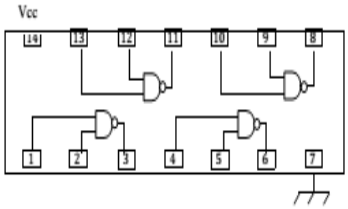
Logic gate XOR(OR exclusif)

5.2. Electrical Circuit :

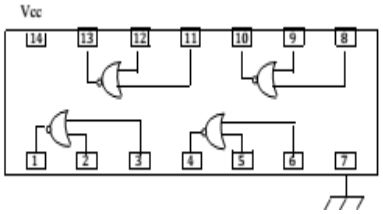




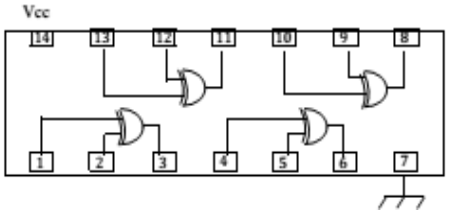




Integrated circuit 7400



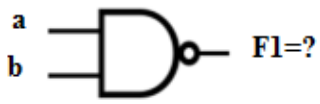
Integrated circuit 7402

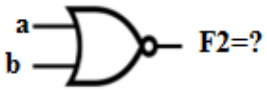


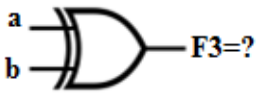
Integrated circuit 7486

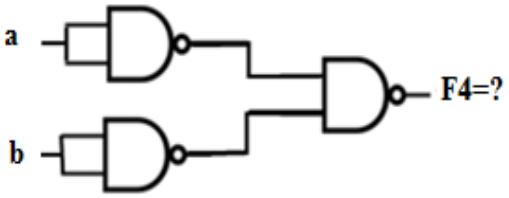
Application Exercise :

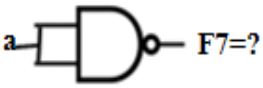
- ✓ Establish the truth tables of the combinational Boolean logic circuits below:

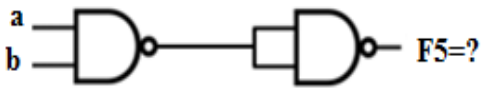


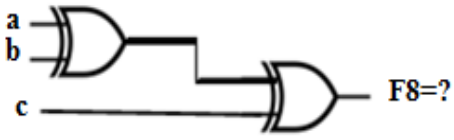


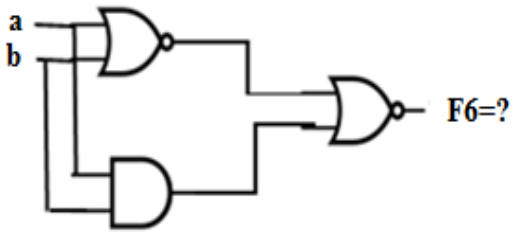


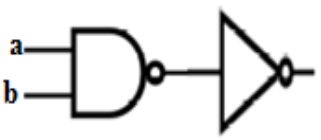












6. Propagation time in a logic gate :

Connect any logic gate and see the delay between input signal and output signal using dual trace oscilloscope compare both technology TTL Logic block(d) and CMOS Logic block(c).

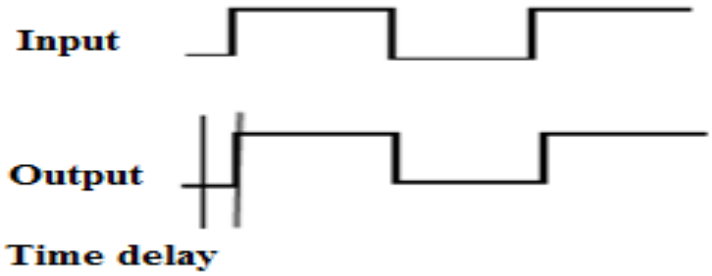


Fig.6 Propagation time in a logic gate

References :

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan
- Letocha ; Introduction aux circuits logiques ; Edition Mc-Graw Hill.
- J.C. Lafont ; Cours et problèmes d'électronique numérique, 124 exercices avec solutions; Edition Ellipses.
- R. Delsol ; Electronique numérique, Tomes 1 et 2 ; Edition Berti
- P. Cabanis ; Electronique digitale ; Edition Dunod.
- M. Gindre ; Logique combinatoire ; Edition Ediscience.
- H. Curry, Combinatory Logic II. North-Holland, 1972
- J-P. Ginisti, La logique combinatoire, Paris, PUF (coll. « Que sais-je? » n°3205), 1997.

PW2: Simplifying Logic Equations Through Practice

1. Objective :

The purpose of this second practical assignment is to:

- Acquire initial proficiency in the simulation tool used.
- Perform experimental validation of certain logic gates using integrated circuits and the KL-33002 module.
- Compare the simplified equations obtained through simulation with their practical verification.

2. De Morgan's Laws :

De Morgan's two theorems are fundamental tools in Boolean logic. They allow the relationship between a logical function and its complement to be established. These laws are widely used in logic circuit design and synthesis software, as they facilitate the choice of the simplest and most suitable logical representation for solving a given problem. They are stated as follows:

$$\overline{A + B} = \overline{A}.\overline{B} \qquad \overline{A.B} = \overline{A} + \overline{B}$$

It should be noted that in the above expressions, the variables (A) and (B) can themselves represent more complex logical expressions. Behind these formulas, which at first glance appear elementary, sometimes lay algebraic developments and logical calculations of some importance.

✓ **Examples of industrial applications :**

- Industrial programmable logic controllers (PLCs)
- Sorting and quality control systems
- Industrial motor and actuator controls
- Industrial security systems

3. Manipulation :

Using the KL-33002 module, you are asked to create the wiring corresponding to the following logic gates on the simulator:

- A NOT gate implemented using only a NOR gate.
- A NOT gate implemented using only a NAND gate.
- An AND gate built using only NOR gates.
- An AND gate built using only NAND gates.
- An OR gate built using only NAND gates.

- An OR gate built using only NOR gates.
- An XOR gate built using only NAND gates.

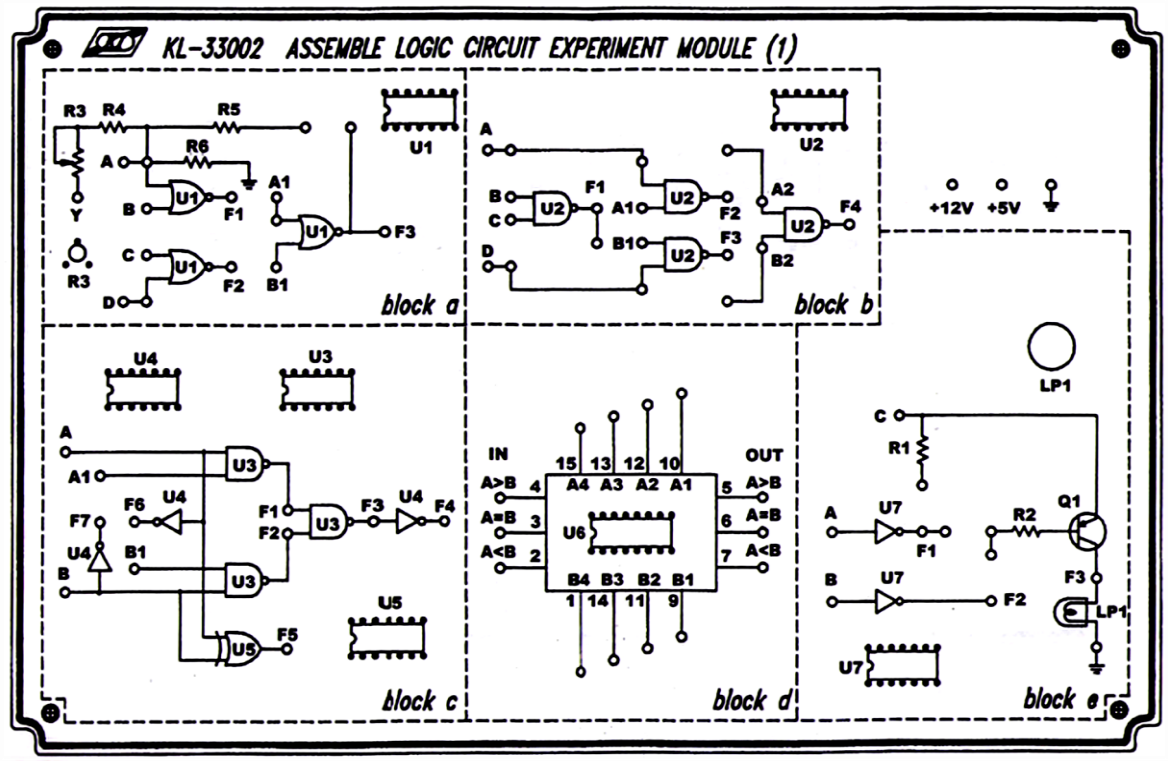


Fig.7 KL-33002 module

Note: The simulator's switches should be used as inputs, while the LEDs should be used as output indicators.

Example: A NOT gate can be implemented using either a NOR gate or a NAND gate, using the KL-33002 block a.

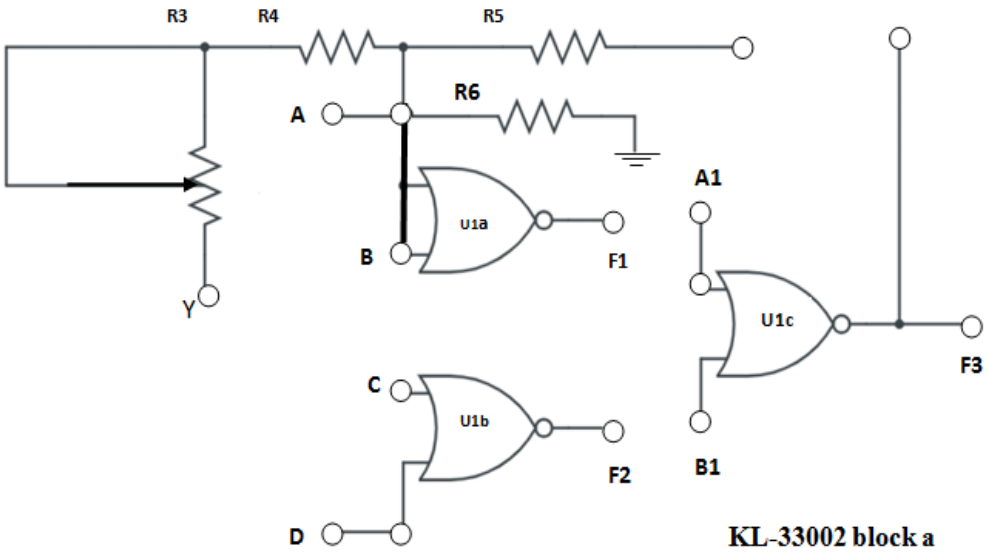


Fig.8 Creating a NOT gate using a NOR gate

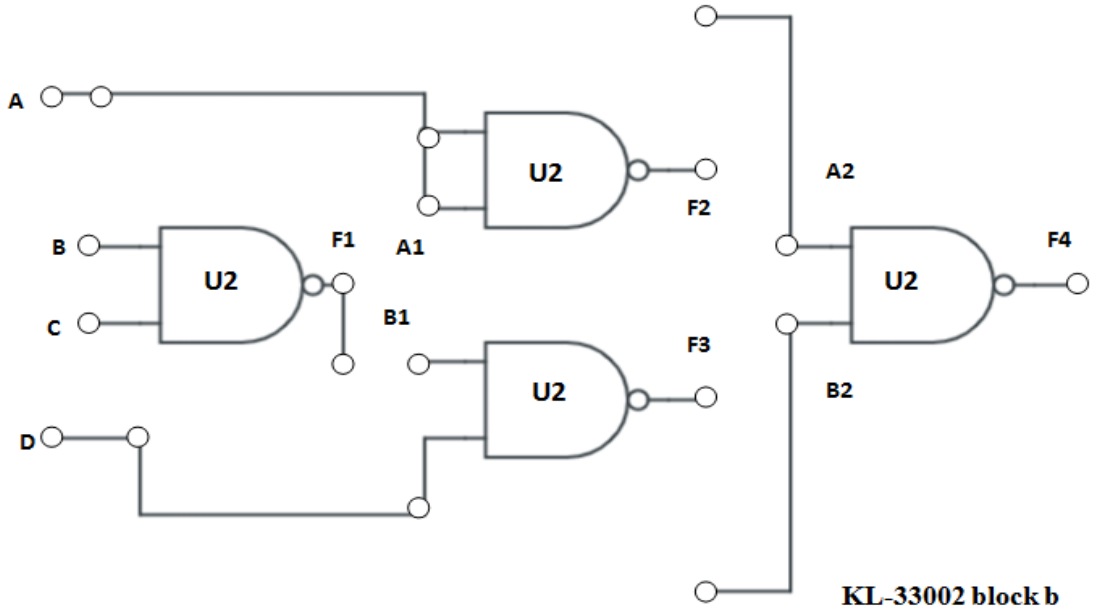


Fig. 9 Realizing a NOT gate using a NAND gate

4. *Application exercises:*

4.1.Exercise 01:

Consider the following truth table, corresponding to a logical function with two input variables and one output variable.

Table 1 Truth table (2/1)

A	B	F
SW0	SW1	L1
0	0	0
0	1	0
1	0	0
1	1	1

- Determine the logic expression (S) using only NAND gates.
- Then, implement this expression on the simulator (KL-33002 module) by performing the corresponding wiring, and then experimentally verify the correct operation of the assembly.
- Perform the wiring on the simulator and verify its correct operation.
- Conclude

4.2.Exercise 02 :

Let the truth table of a function with three inputs and one output be:

Table 2 Truth table (3/F)

SW0	SW1	SW2	L0
A	B	C	F3
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

- Provide a simplified logic diagram using only NAND gates

- Perform the wiring on a simulator and verify that it is working properly.

4.3.Exercise 03 :

Given the following function:

F(a,b,c,d) = sum (0, 1, 2, 4, 5, 6, 8, 9, 10, 14).

- Generate the inverse of this function in its simplified form on a simulator using the synthesis results.

- Perform the wiring on a simulator using only NAND gates.

- Perform the wiring on a simulator using only NOR gates.

References :

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan
- Letocha ; Introduction aux circuits logiques ; Edition Mc-Graw Hill.
- J.C. Lafont ; Cours et problèmes d'électronique numérique, 124 exercices avec solutions; Edition Ellipses.
- R. Delsol ; Electronique numérique, Tomes 1 et 2 ; Edition Berti
- P. Cabanis ; Electronique digitale ; Edition Dunod.
- M. Gindre ; Logique combinatoire ; Edition Ediscience.
- H. Curry, Combinatory Logic II. North-Holland, 1972
- J-P. Ginisti, La logique combinatoire, Paris, PUF (coll. « Que sais-je? » n°3205), 19

PW3 : Study and implementation of usual common combinatorial logic functions

1. Objective:

- Understand and implement basic combinational logic functions such as switching circuits called Multiplexers.
- Create truth tables verifying the operation of each Multiplexer.
- Create truth tables verifying the operation of each Multiplexer.
- Understand and implement combinational logic circuits for Encoding/Decoding.
- Create various truth tables verifying the operation of combinational logic circuits for Encoding/Decoding.

2. Theoretical reminder :

*2.1.***Logical function:** is a mathematical or computer function that associates **logical values** (true / false, or **1 / 0**) at the input with a logical value at the output, according to a well-defined rule.

*2.2.***A multiplexer (MUX):** The multiplexer is a combinational system whose function is to select one of 2^n inputs and transmit it to the output.

*2.3.***Switch (SW0, SW1, SW2,..):** it is a hardware that connects several electronic or computer components together, the interconnections being configurable.

*2.4.***Demultiplexer:** The demultiplexer is a combinational system whose function is to transmit an **input to one** of the **2^n outputs**.

*2.5.***Decoder:** is a combinational system whose function is to activate one of the 2^n outputs.

✓ **Examples of industrial applications :**

- Industrial lighting controls and signaling
- Automated dosing or mixing systems
- Industrial metering and display

3. Required equipment :

KL-31001 Digital Logic Lab Simulation Model; **KL-33004 / KL-33005** and **KL-33006** Modules.

KL33004 / KL33005 and **KL-33006** Modules, Connecting Wires :

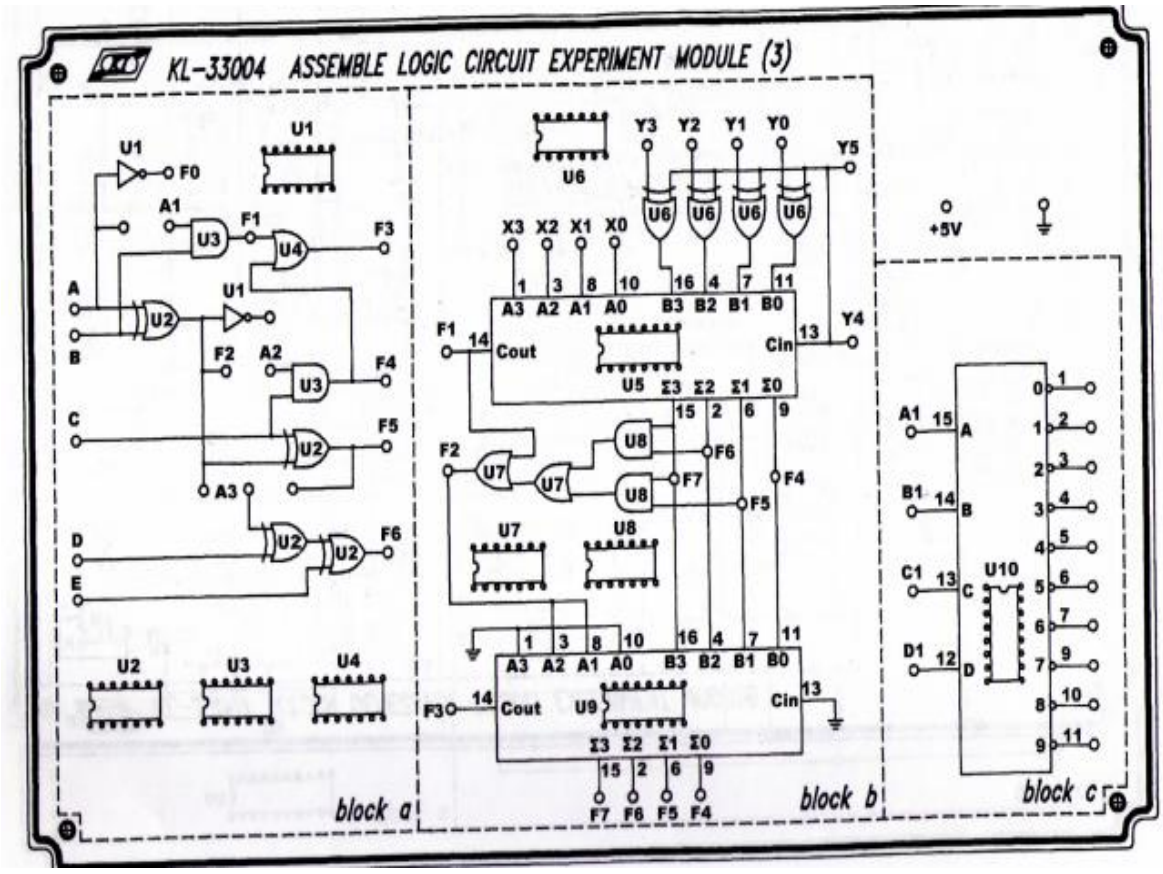


Fig.10 KL-33004 module

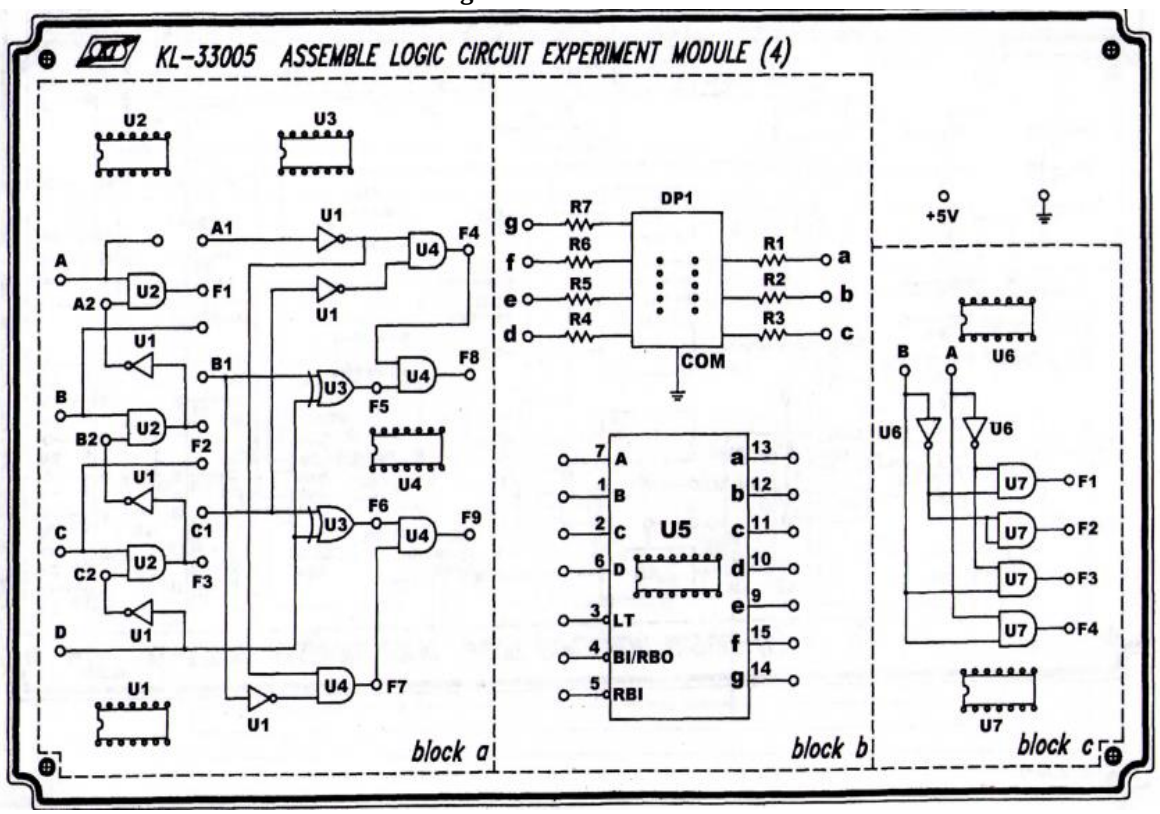


Fig.11 KL-33005 module

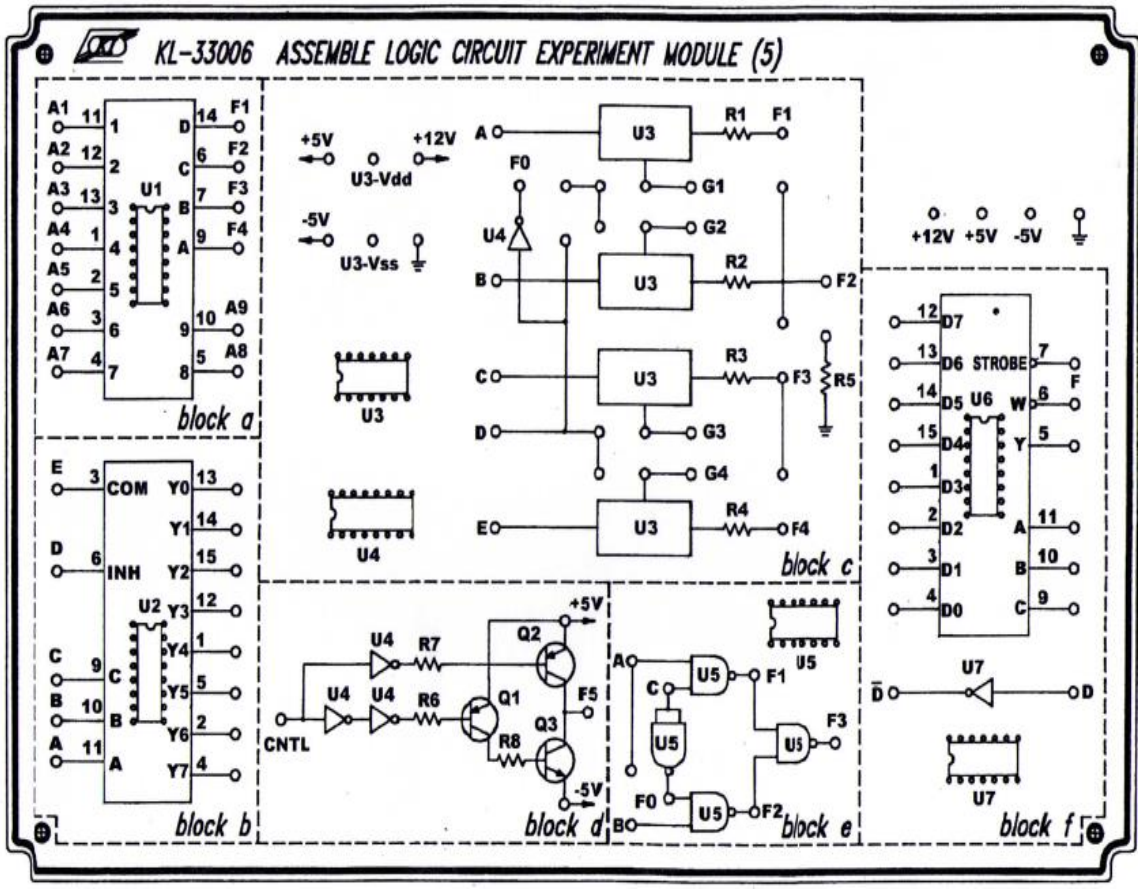


Fig.12 KL-33006 module

4. Manipulation:

4.1. Study and construction of a Multiplexer:

In this case, we use a **2-to-1 multiplexer** with NAND gates.

For this operation, we use Block (e) of the KL-3306 module.

- Let's make the connections according to the figure below. :

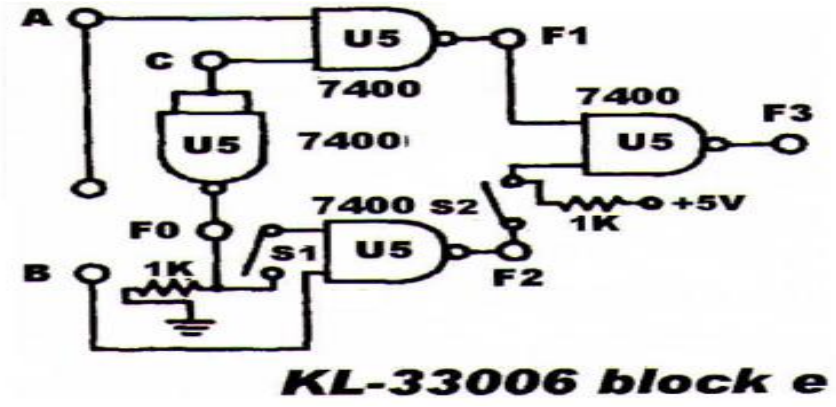


Fig.13 Module KL-33006 block e

- Connect the module's **+5V** terminal to the **+5V** output of the fixed power supply.
- Connect the module's **grounds** to the console's.
- Close switches **S1** and **S2** (position 1).

- Connect inputs **A, B**, and **C** to switches **SW0, SW1**, and **SW2**.
- Connect the single output **F3** to logic indicator **L0.**
- Turn on the **KL-33001** module.
- Create the Multiplexer truth table.

Table3 Multiplexer truth table

SW2	SW1	SW0	L0
C	B	A	F3
0	0	0	?
0	0	1	?
0	1	0	?
0	1	1	?
1	0	0	?
1	0	1	?
1	1	0	?
1	1	1	?

The logical equation: $F3 = AC + B\overline{C}$

4.2. Study and construction of a Demultiplexer :

4.2.1. Objective :

- Understand the operating principles and construction of demultiplexer circuits.
- In this case, we use a 1-to-2 demultiplexer with NAND gates.
- For this experiment, we use Block (e) of the KL-3306 module.

4.2.2. Carry out a 1 to 2 DEMUX on a simulator and check that it is working correctly.

For this manipulation we use the **KL-33006** block e module:

- Insert the connector clip as shown below.
- Connect A to data switch **SW0, C** to **SW3, F1** and **F2** to logic indicators **L0** and **L1**, respectively.
- Close switches **S1** and **S2** (position 1).
- Turn on the **KL-33001** module.
- Create the demultiplexer truth table.
- Connect **A** with **B**

Table 4 Demultiplexer truth table

C	A	F2	F1
SW3	SW0	L1	L0
0	0	?	?
0	1	?	?
1	0	?	?
1	1	?	?

4.2.3. Study and production of a four-input, two-output encoder :

Using block (a) of the following **KL-33005** Digital Logic Lab module :

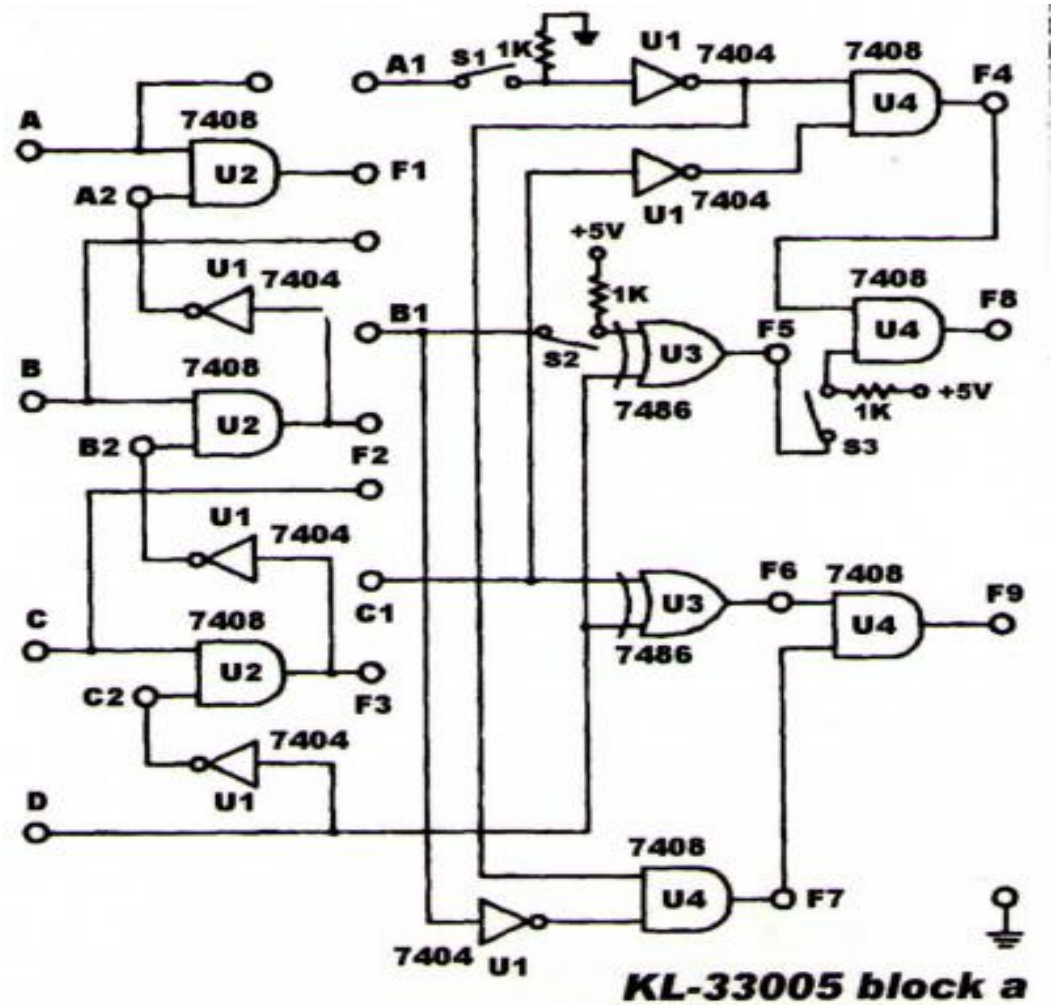


Fig.14 Module KL-33005 block a

- **A, B, C,** and **D** as inputs, and **F8** and **F9** as outputs.
- Connect the points: **A=A1**, **B=B1**, and **C=C1**
- Close the switches: **S1, S2**.
- Connect outputs **F8** and **F9** to logic indicators **L1 and L2**.

- Turn on the **KL-33001** module.

- Fill in the truth table for outputs **F8** and **F9** based on the inputs. :

Table 5 Truth table of a four-input, two-output encoder (circuit N° 1)

D	C	B	A	F8	F 9
0	0	0	0	?	?
0	0	0	1	?	?
0	0	1	0	?	?
0	0	1	1	?	?
0	1	0	0	?	?
0	1	0	1	?	?
0	1	1	0	?	?
0	1	1	1	?	?
1	0	0	0	?	?
1	0	0	1	?	?
1	0	1	0	?	?
1	0	1	1	?	?
1	1	0	0	?	?
1	1	0	1	?	?
1	1	1	0	?	?
1	1	1	1	?	?

- Redo the truth table and compare the result with this time (**A1** connected to **F1**, **B1** to **F2**, and **C1** to **F3**):

Table 6 Truth table of a four-input, two-output encoder (circuit N° 2)

D	C	B	A	F8	F 9
0	0	0	0	?	?
0	0	0	1	?	?
0	0	1	0	?	?
0	0	1	1	?	?
0	1	0	0	?	?
0	1	0	1	?	?
0	1	1	0	?	?
0	1	1	1	?	?
1	0	0	0	?	?
1	0	0	1	?	?
1	0	1	0	?	?
1	0	1	1	?	?
1	1	0	0	?	?
1	1	0	1	?	?
1	1	1	0	?	?
1	1	1	1	?	?

4.2.4. Study and construction of a two-input, four-output decoder :

Using block (c) of the following **KL-33005** Digital Logic Lab module:

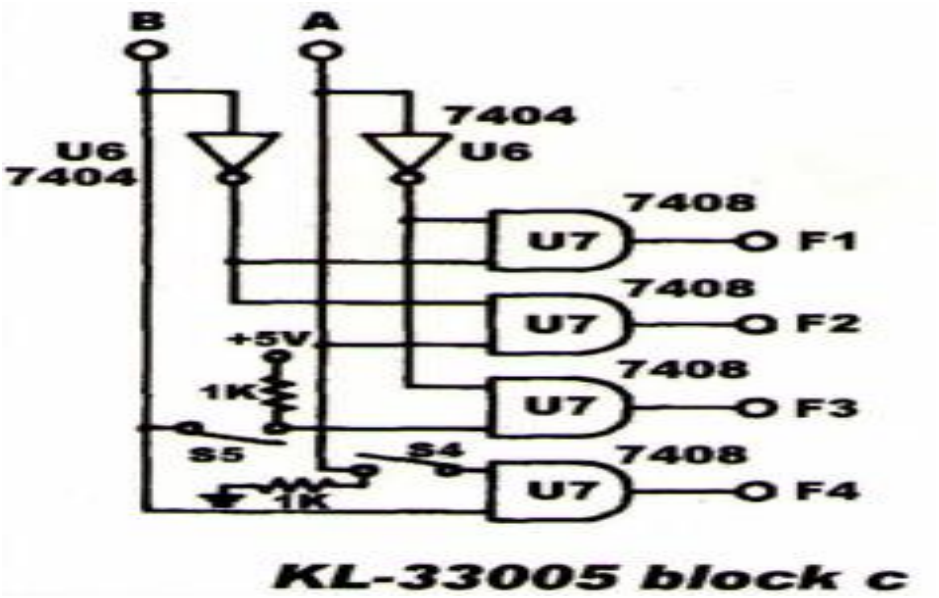


Fig.15 Module KL-33005 block C

- Connect inputs **A** and **B** to switches **SW0** and **SW1** (data switches).
- Close switches **S4** and **S5**.
- Connect outputs **F1, F2, F3**, and **F4** to logic indicators **L0, L1, L2**, and **L3**.
- Turn on the **KL-33001** module.
- Complete the truth table (Table 7):

Table 7 Truth table of a two-input, four-output decoder

B	A	F1	F2	F3	F4
0	0	?	?	?	?
0	1	?	?	?	?
1	0	?	?	?	?
1	1	?	?	?	?

- Conclude ?

5. Study and production of a seven (7) segment BCD decoder :

The BCD code was mainly used in the early days of calculating machines. It is sometimes still used in systems that often need to display numerical information to the user or for very

simple systems that handle money. The BCD code encodes the number to be represented in a very direct way. Each digit of the number is encoded on 4 bits. The binary possibilities from 10 to 15 are not used. The seven light segments, arranged as follows, allow the writing of all the digits :

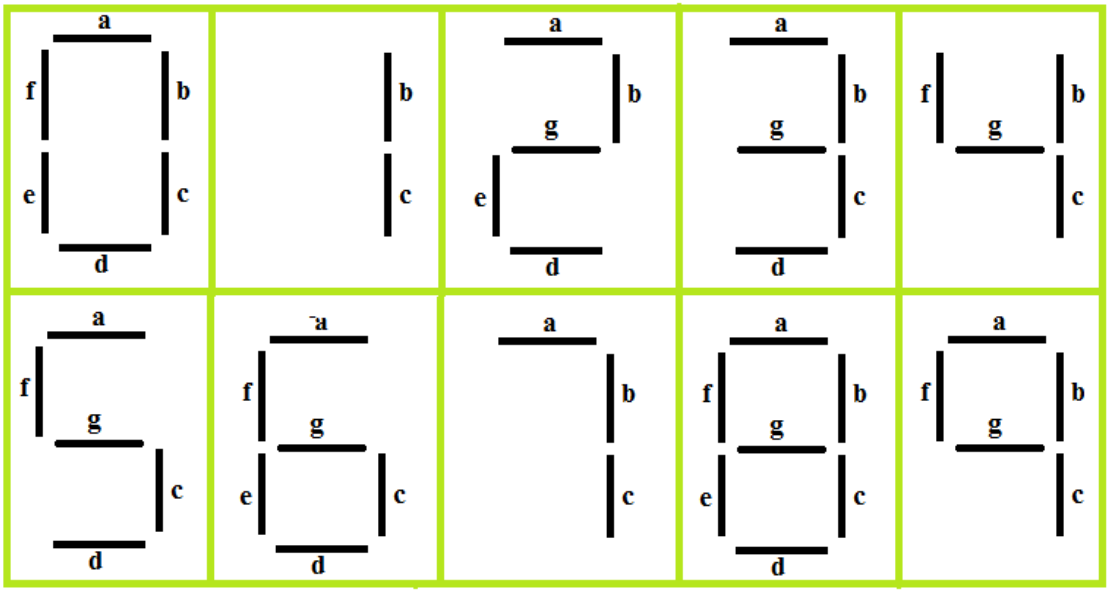


Fig.16 7-segment display.

Using block (**c**) of the following **KL-33005** Digital Logic Lab module :

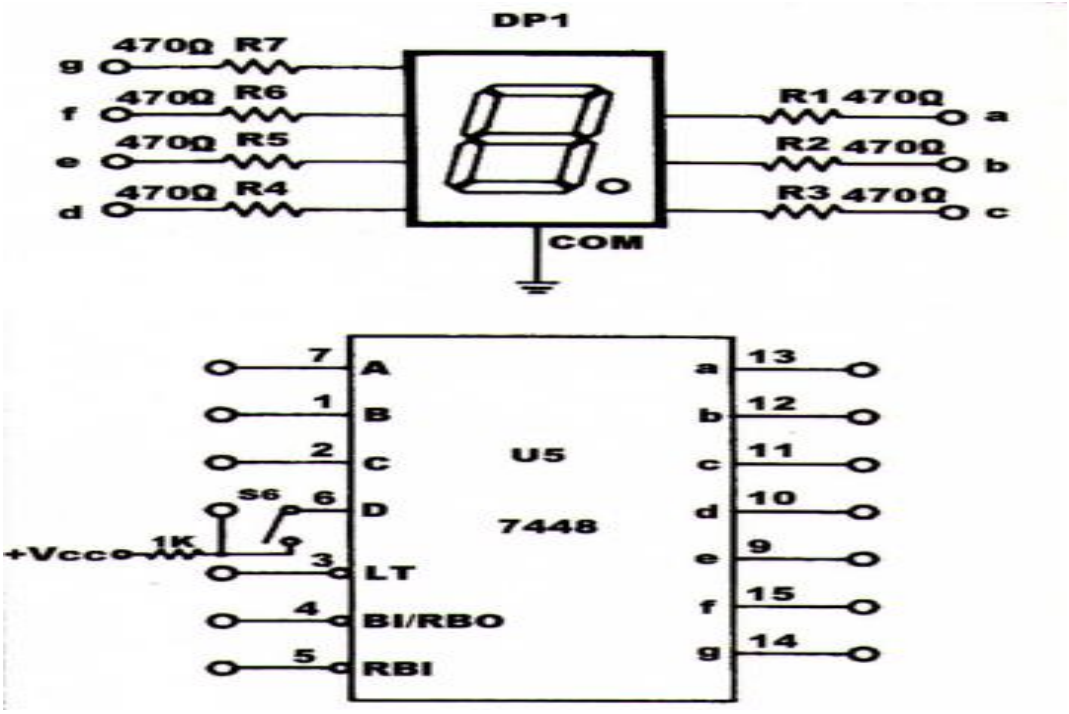


Fig.17 Module KL-33005 block b

- Connect inputs **A, B, C,** and **D** to switches **SW0, SW1, SW2**, and **SW3**.
- Close switch **S6 (OFF)**.

- Connect outputs **a, b, c, d, e, f**, and **g** to logic indicators **L0, L1, L2, L3, L4, L5**, **L6**.
- Turn on the KL-33001 module.
- Check the following truth table:

Table 8 Truth table of a seven (7) segment BCD decoder

D	C	B	A	a	b	c	d	e	f	g	Number
0	0	0	0	1	1	1	1	1	1	0	**0**
0	0	0	1	0	1	1	0	0	0	0	**1**
0	0	1	0	1	1	0	1	1	0	1	**2**
0	0	1	1	1	1	1	1	0	0	1	**3**
0	1	0	0	0	1	1	0	0	1	1	**4**
0	1	0	1	1	0	1	1	0	1	1	**5**
0	1	1	0	0	0	1	1	1	1	1	**6**
0	1	1	1	1	1	1	0	0	0	0	**7**
1	0	0	0	1	1	1	1	1	1	1	**8**
1	0	0	1	1	1	1	0	0	1	1	**9**

6. Conclusion : In this lab, we learned how to implement combinational logic functions using the KL-33006 module and the KL-31001 Digital Logic Lab. We also learned how to establish the truth table of the multiplexer and, more importantly, how to determine the logic function from this table. We noted that the multiplexer's role is to direct several inputs to a single output, the latter being selected by addressing. The capacity of the multiplexer depends on the number of bits used. Finally, we noticed that the demultiplexer works in the opposite way to the multiplexer.

References :

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan
- Letocha ; Introduction aux circuits logiques ; Edition Mc-Graw Hill.
- J.C. Lafont ; Cours et problèmes d'électronique numérique, 124 exercices avec solutions; Edition Ellipses.
- R. Delsol ; Electronique numérique, Tomes 1 et 2 ; Edition Berti
- P. Cabanis ; Electronique digitale ; Edition Dunod.
- M. Gindre ; Logique combinatoire ; Edition Ediscience.
- H. Curry, Combinatory Logic II. North-Holland, 1972
- J-P. Ginisti, La logique combinatoire, Paris, PUF (coll. « Que sais-je? » n°3205), 1997.

PW4 : Study and realization of an arithmetic combinational logic circuit

1. Objective:

-Understand and implement arithmetic combinational logic functions.

-Establish various truth tables to verify the operation of the arithmetic combinational logic circuit(s).

2. Principle of addition :

A computer can only perform a few basic operations: addition and subtraction, using logic gates (and, or, etc.), but each operation lasts very little time (<1 ns = one billionth of a second). Certainly much better than a human being, and without fatigue.Addition and subtraction circuits with suitable algorithms are more than sufficient to perform any mathematical calculation in the world. If asked to add 12 + 10, the computer must convert these numbers to binary. Application: convert the decimals 12 + 10 and 22 to binary. Write the addition corresponding to the decimal addition 12 + 10 = 22 in binary.

Ultimately, this amounts to performing elementary additions where digits appear that are the result of the sum s and possibly carryover digits r.

3. Required equipment :

KL 31001 Digital Logic Lab; KL-33003 / KL-33004 Module.

Or the KL33003 / KL33004 module:

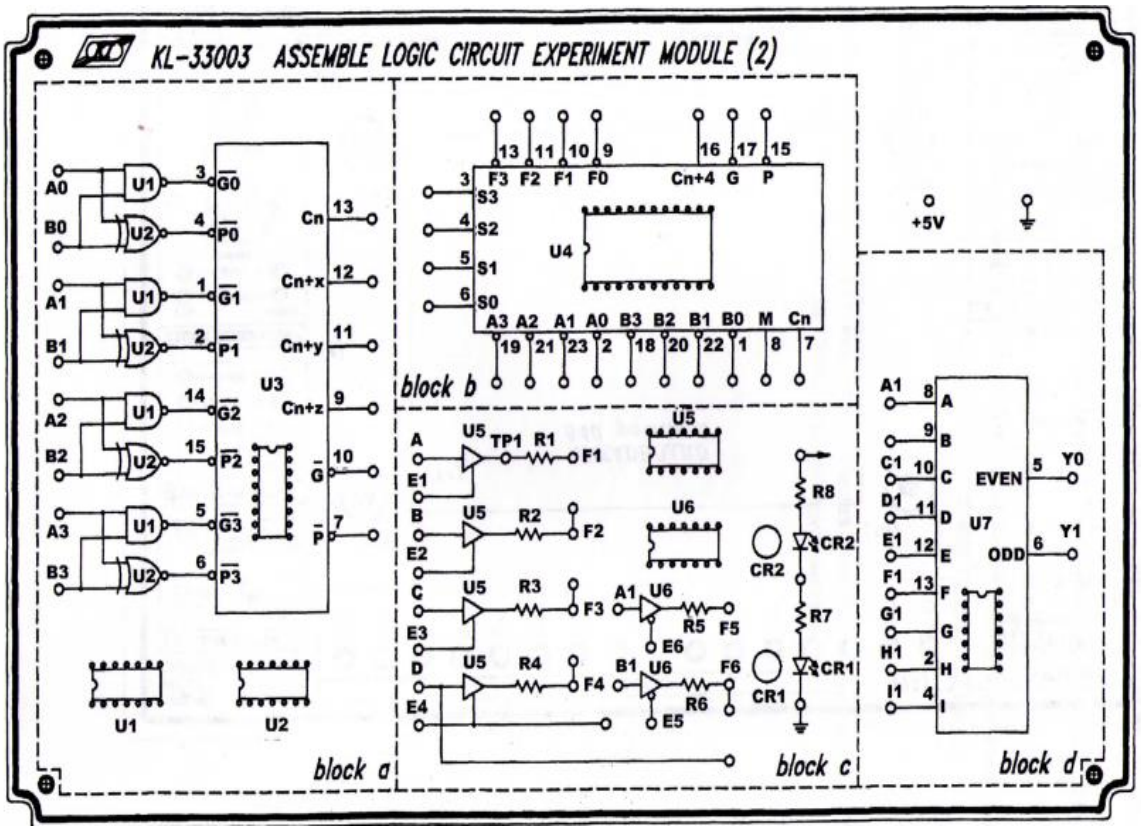


Fig.18 KL-33003 module

4. Manipulation:

4.1.Study and realization of a half-adder and a full adder :

Using the basic logic gates already studied and block (a) of the following KL-33004 Digital Logic Lab module:

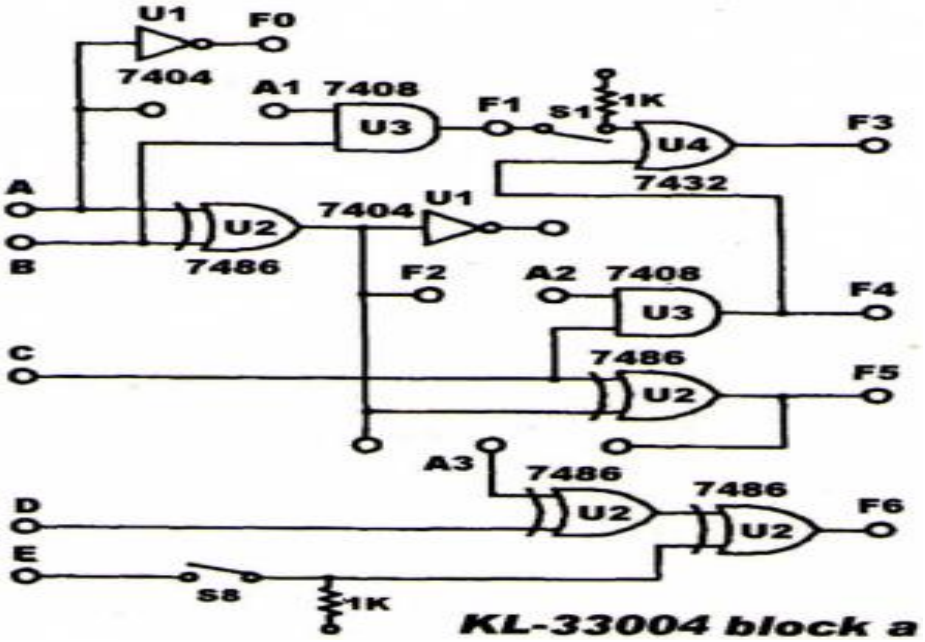


Fig.19 Module KL-33004 block a

- ***Half adder:***

Design a circuit capable of adding two bits (input variables) and which can generate their sum **S** and their carryover **D** (output variables).

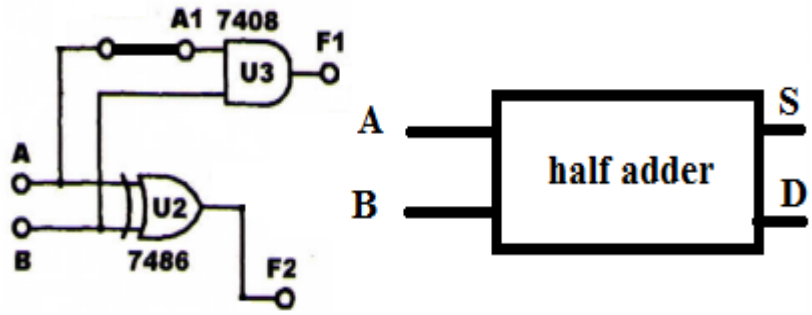


Fig.20 Half adder block

- Connect the points: **A** to **A1**, **A2 to F2.**
- Connect inputs **A** and **B** to switches **SW0** and **SW1** (Data switches KL-33001).
- Connect outputs **F1** and **F2** to logic indicators **L0** and **L1**.
- Turn on the **KL-33001** module.

- Check the truth table of a half-adder:

Table 9 Truth table of a half-adder

Input		Output	
B	**A**	**S**	**D**
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

- ***Full adder:***

The full adder consists of two one-bit half adders and generates a carry of the half additions **F1 and F4** and the sums at **F2 and F5**.

The full adder can perform additions of numbers more than two bits wide.

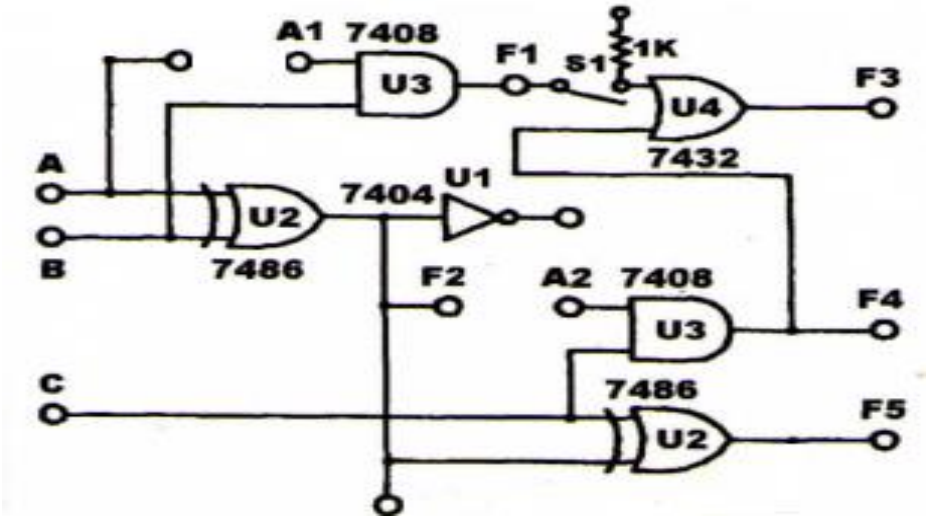


Fig.21 Circuit of Full adder (KL-33004 block a)

- Connect the points: **A1 to A** and **A2 to F2**.
- Close switch **S1**.
- Connect inputs **A, B**, and **C** to switches **SW0, SW1**, and **SW2** (data switches).
- Connect outputs **F3** and **F5** to logic indicators **L0** and **L1**.
- Turn on the **KL-33001** module.

- Verify the truth table of a full adder:

Table 10 Truth table of a full adder

Input			Output	
C	**B**	**A**	**S**	**D**
0	0	0	**?**	**?**
0	0	1	**?**	**?**
0	1	0	**?**	**?**
0	1	1	**?**	**?**
1	0	0	**?**	**?**
1	0	1	**?**	**?**
1	1	0	**?**	**?**
1	1	1	**?**	**?**

4.2.Study and realization of a half-subtractor and a complete subtractor :

- ***Half-subtractor :***

Design a circuit that subtracts one bit from another and generates their difference **D** and their borrowing **E**.

Formulas:

- Difference : $\boldsymbol{D = \bar{A}B + A\bar{B}}$

- Borrowing : $\boldsymbol{E = \bar{A}B}$

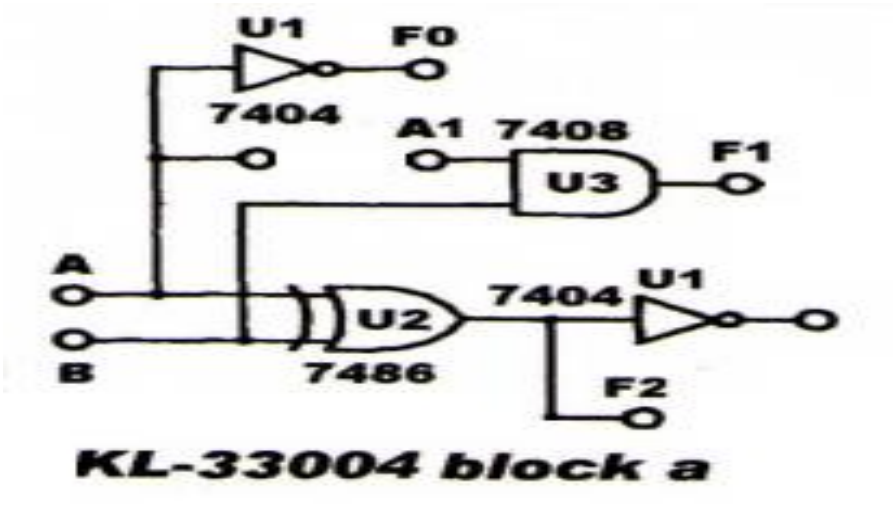


Fig.22 Circuit of Half-subtractor (KL-33004 block a)

- Connect point **F0 to A1**.

- Connect inputs **A and B** to switches **SW0 and SW1** (Data switches).
- Connect outputs **F1=E** and **F2=D** to logic indicators **L0 and L1**.
- Turn on the **KL-33001** module.
  - Check the truth table of a one-bit half-subtractor:

Table 11 Truth table of a half-subtractor

Input		Output	
B	**A**	**D**	**E**
0	**0**	0	0
0	**1**	1	1
1	**0**	1	0
1	**1**	0	0

➢ ***Subtractor :***

To take into account the borrowing from the previous stage, it is therefore necessary to design a complete subtractor with three input variables and two output variables.

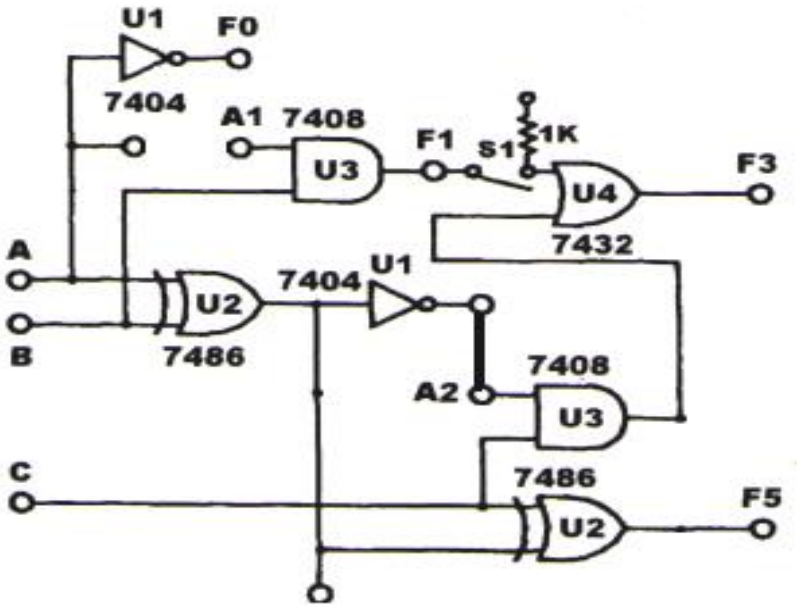


Fig.23 Circuit of Subtractor (KL-33004 block a)

- Connect point **F0 to A1**.
- Close switch **S1** (position 1).
- Connect inputs **A, B**, and **C** to switches **SW0, SW1, and SW2**.
- Connect outputs **F3=E** and **F5=D** to logic indicators **L1 and L2**.
- Turn on the **KL-33001** module.
  - Check Truth Table of a One-Bit Subtractor:

Table 12 Truth table of a subtractor

Input			Output	
C	**B**	**A**	**F5 =D**	**F3=E**
0	**0**	**0**	0	0
0	**0**	**1**	1	1
0	**1**	**0**	1	0
0	**1**	**1**	0	0
1	**0**	**0**	1	1
1	**0**	**1**	0	1
1	**1**	**0**	0	0
1	**1**	**1**	1	1

4.3.Practical construction of a complete four-bit binary adder/subtractor:

Using the **KL-33004** block, it takes as inputs two 4-bit numbers, the first number being **Y0, Y1, Y2**, and **Y3**, and the second number being **X0, X1, X2**, and **X3.** The output will be a 4-bit number **F8, F9, F10**, and **F11**.

Note: Output F1 is reserved for the carry.

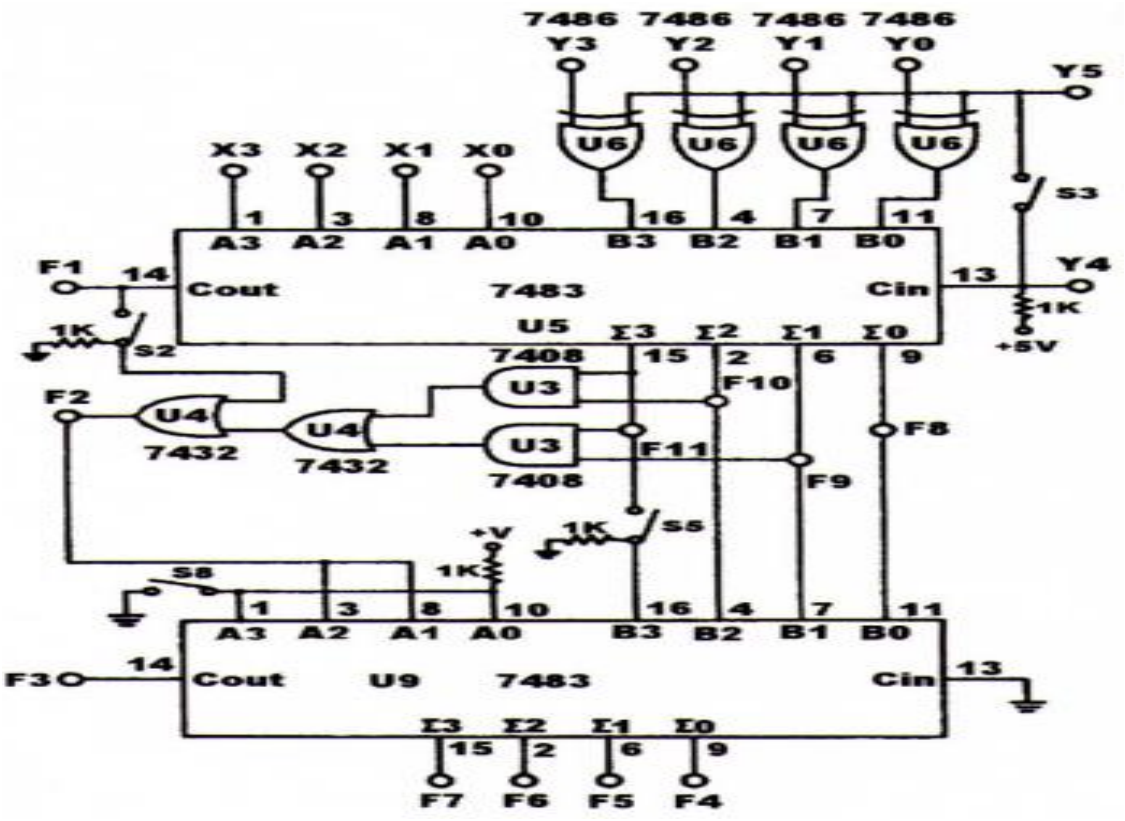


Fig.24 Module KL-33004 block b

- Close switches **S2, S3, S5**, and **S8** (ON).
- Turn on the KL-33001 module.
- Create the truth table. Conclusion?

Reference :

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan
- Letocha ; Introduction aux circuits logiques ; Edition Mc-Graw Hill.
- J.C. Lafont ; Cours et problèmes d'électronique numérique, 124 exercices avec solutions; Edition Ellipses.
- R. Delsol ; Electronique numérique, Tomes 1 et 2 ; Edition Berti
- P. Cabanis ; Electronique digitale ; Edition Dunod.
- M. Gindre ; Logique combinatoire ; Edition Ediscience.
- J-P. Ginisti, La logique combinatoire, Paris, PUF (coll. « Que sais-je? » n°3205), 1997.

PW5: Study and implementation of a Combinational Logic Circuit

1. Objective :

Analysis of some representative combinatorial subsystems:

- Two's complement generation.
- Gray/binary conversion.
-

2. Introduction :

The implementation of a logic function can be done using logic gates. In the following, we will present two examples illustrating the implementation of logic functions from these gates.

3. Two's complement :

In two's complement coding, the negative of a number is obtained by inverting all the bits in its codeword, then adding 1, without retaining any carry. For an n-bit word, the representable values range from $\mathbf{-2^{n-1}}$ **to** $\mathbf{2^{n-1} - 1}$, with only one possible representation for the value zero. This coding system is of great importance because it allows signed arithmetic operations to be performed without having to explicitly handle the sign.

3.1.Example :

Implement the two's complement of a three-bit number.

Table 13 Two's complement truth table

SW0	SW1	SW2	L2	L1	L0
C	B	A	F3	F2	F1
0	0	0	0	0	0
0	0	1	1	1	1
0	1	0	1	1	0
0	1	1	1	0	1
1	0	0	1	0	0
1	0	1	0	1	1
1	1	0	0	1	0
1	1	1	0	0	1

To determine the equations corresponding to this truth table, it is necessary to establish the Karnaugh table associated with each output.

3.2.Manipulation :

Using the **KL-33004 module, block a**:

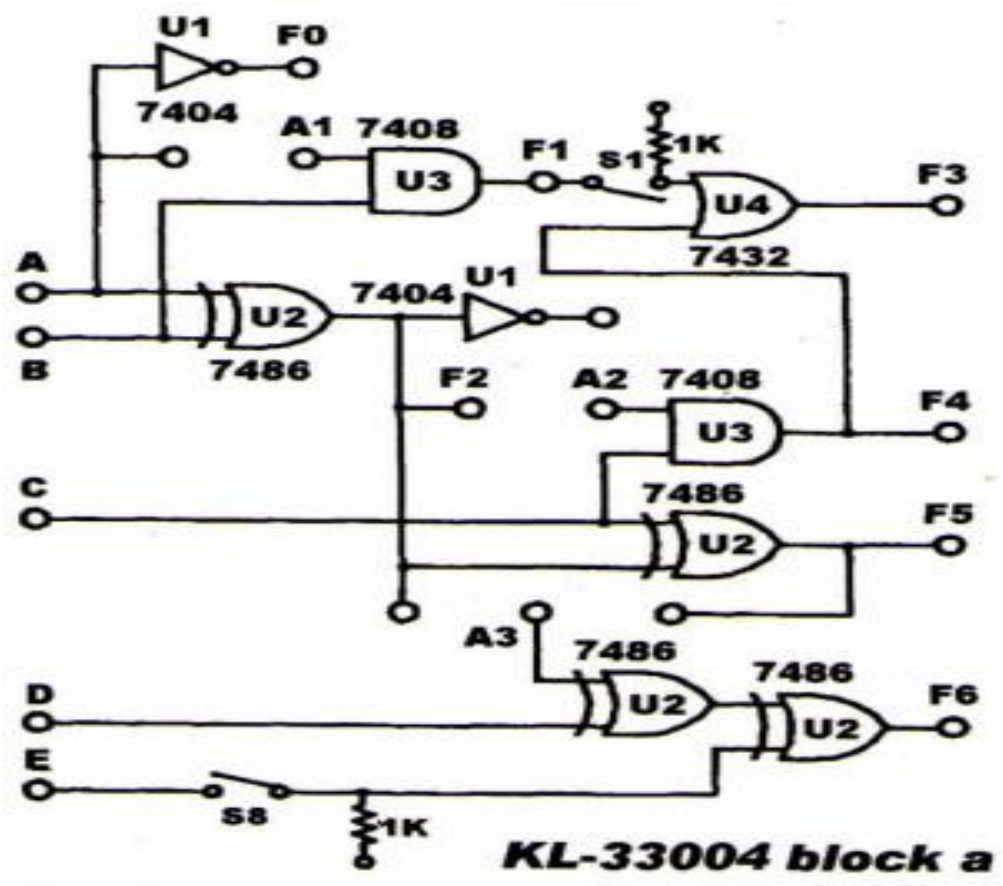


Fig.25 Circuit of natural gray- binary converter (KL-33004 block a)

- Create the truth table.
- Find the logic functions.
- Provide their flowcharts and wire the functions on a simulator.

4. Natural Gray-Binary Converter

Gray code was designed to avoid problems associated with transitions when moving from one word to the next. In its sequential order, only one bit varies between two consecutive words, making them adjacent. Furthermore, this code exhibits a cyclical nature.

Example: Conversion of a three-bit Gray code to binary code. The corresponding truth table is given below:

Table 14 Truth table GRAY code

Gray code inputs			Binary code outputs		
G3	G2	G1	B3	B2	B1
0	0	0	0	0	0
0	0	1	0	0	1
0	1	1	0	1	0
0	1	0	0	1	1
1	1	0	1	0	0
1	1	1	1	0	1
1	0	1	1	1	0
1	0	0	1	1	1

To derive the equations from this truth table, it is necessary to construct the Karnaugh table corresponding to each output.

4.1.Manipulation :

Use the **KL-33004 block a** module.

Build the transcoder to transform numbers expressed in the three-bit Gray code into their equivalent binary values using only exclusive-OR gates.

References :

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan
- Letocha ; Introduction aux circuits logiques ; Edition Mc-Graw Hill.
- J.C. Lafont ; Cours et problèmes d'électronique numérique, 124 exercices avec solutions; Edition Ellipses.
- R. Delsol ; Electronique numérique, Tomes 1 et 2 ; Edition Berti
- P. Cabanis ; Electronique digitale ; Edition Dunod.
- M. Gindre ; Logique combinatoire ; Edition Ediscience.
- H. Curry, Combinatory Logic II. North-Holland, 1972
- J-P. Ginisti, La logique combinatoire, Paris, PUF (coll. « Que sais-je? » n°3205), 1997.

PW6 : Study and creation of a combinational logic circuit

1. Objective :

- The study of truth table, simplification of assembly of a combinational circuit from a specification of a comparator.

2. Theoretical reminder on comparators:

Comparators are electronic circuits that perform various functions. Note that there are different types of comparators:

- Voltage comparators
- Current comparators
- Digital or Logic comparators

As previously noted, our presentation here will focus on the study and implementation of logic comparators.

A logic comparator is a logic circuit that compares two binary numbers, generally denoted **A** and **B**. It has three outputs denoted **A = B**, **A > B**, and **A < B**. All these considerations are reflected in the following truth table:

Table 15 Truth table of a comparator

inputs		**Outputs**		
B	**A**	**A=B**	**A>B**	**A<B**
0	**0**	**1**	**0**	**0**
0	**1**	**0**	**0**	**1**
1	**0**	**0**	**1**	**0**
1	**1**	**1**	**0**	**0**

We are thus led to the logic diagram of the figure below which provides the three signals **A < B, A = B** and **A > B** from bits **A** and **B**:

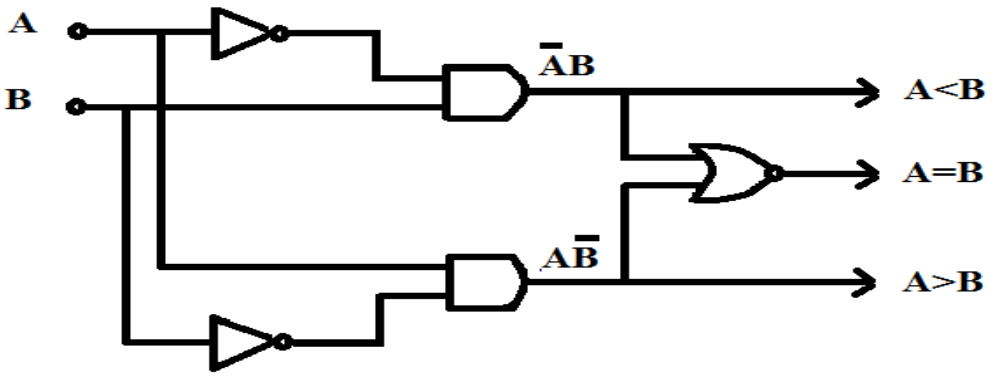


Fig.26 Comparator logic diagram

3. Required equipment :

KL 31001 Digital Logic Lab; KL-33002 Module, Connection Wires.

- Or the KL33002 module

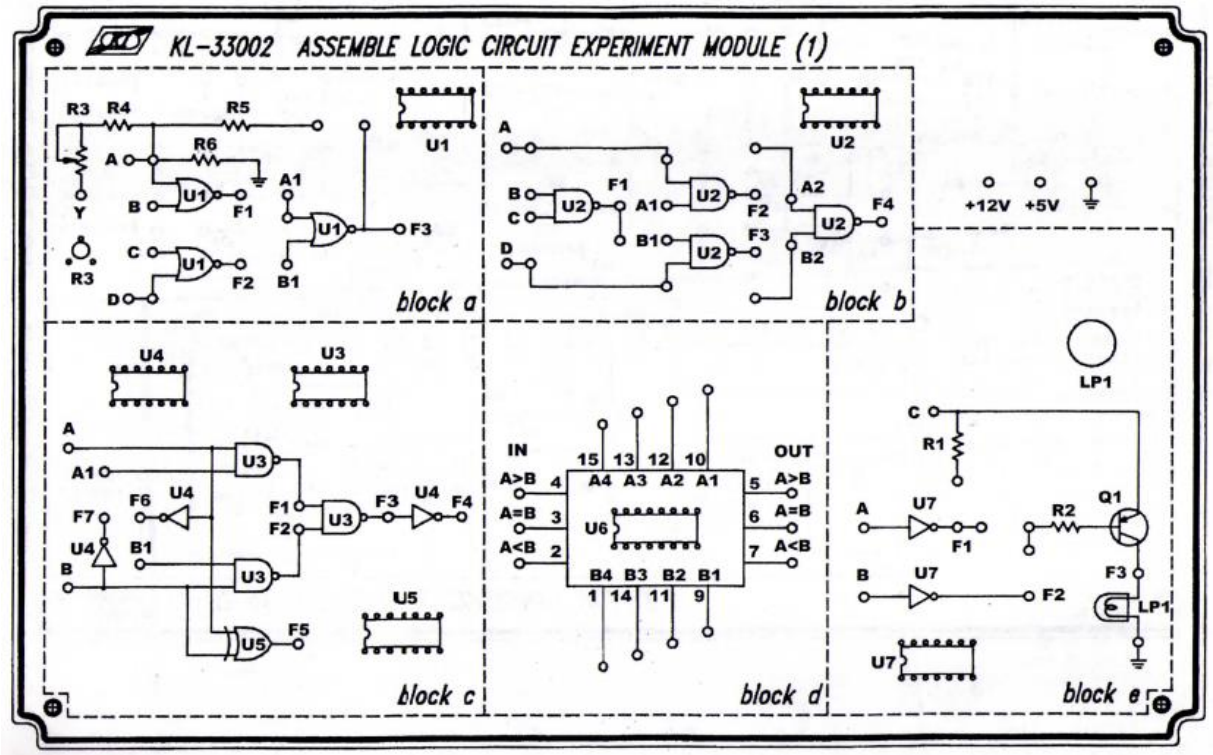


Fig.27 KL33002 module

4. Manipulation :

- ***Part 1 :***

For this manipulation, we use **Block (e)** of the **KL-33002** module.

Let's make the connections according to the figure below:

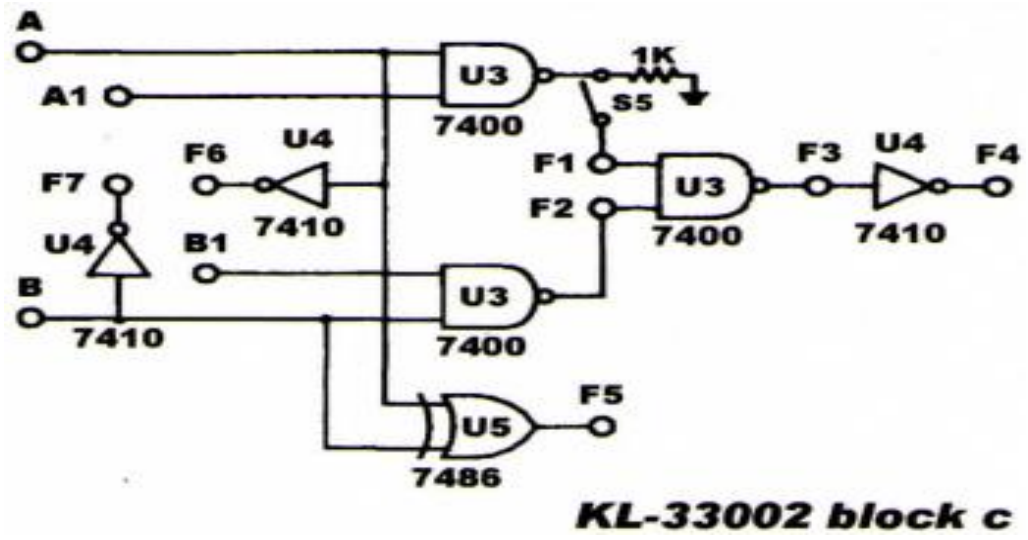


Fig.28 Module KL33002 block C

- Connect the module's **+5V** terminal to the **+5V** output of the fixed power supply.
- Connect the module's grounds to the console's.
- Close switch **S5**.
- Connect **A1, F7** and **B1, F6**.
- Connect inputs **A and B** to switches **SW0 and SW1** (Data switches).
- Connect outputs **F1, F2, and F5** to logic indicators **L0, L1, and L2**.
- Turn on the **KL-33001** module.
- Draw the truth table for a comparator of two one-bit numbers.
- Provide the logic diagram created by these functions.
- Verify its proper operation on a simulator.

Table 16 Truth table of a comparator (part 1)

SW1	SW0		F1	F2	F5
B	A		L0	L1	L2
0	0	A=B	1	0	0
0	1	A>B	0	0	1
1	0	A<B	0	1	0
1	1	A=B	1	0	0

- ***Part 2 :***

- Create a circuit that compares two numbers **A and B**, each written on four bits.
- For this experiment, use the **KL-33002 block (d)** module.

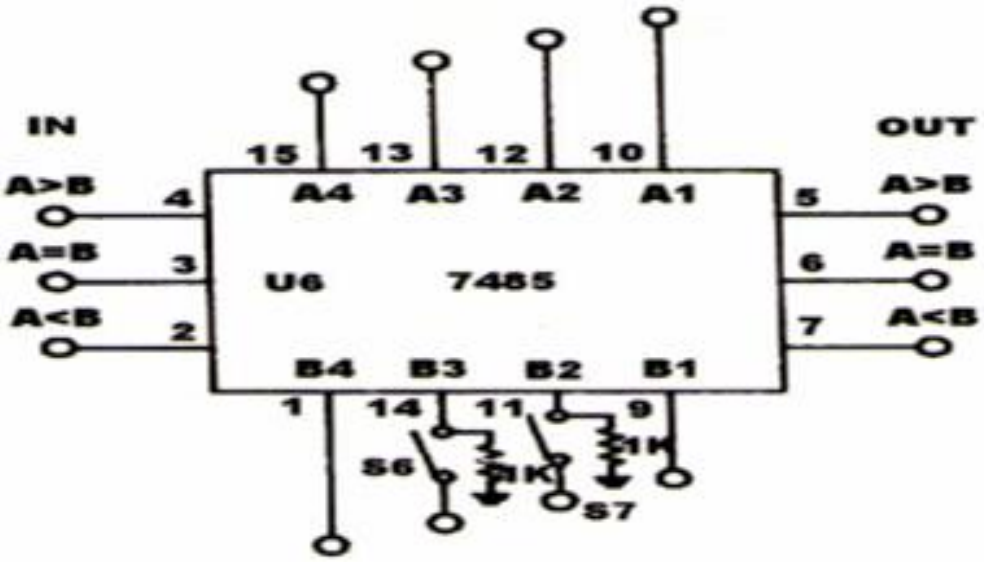


Fig.29 Module KL33002 block d

- Connect the module's **+5V** terminal to the **+5V** output of the fixed power supply.
- Connect the module's grounds to the console's.
- Close switches **S6, S7**.
- Set inputs **B2, B3** to position (OFF).
- Set IN (**A>B** and **A<B** to **0**, **A=B** to **1**).
- Turn on the KL-33001 module.
- Verify its proper operation on a simulator.
- Provide the flowchart created by these functions.
- Draw the truth table for a comparator of two one-bit numbers.

Table 17 Truth table of a comparator of two one-bit numbers

Inputs Number				Inputs Cascading			Outputs		
A4, B4	**A3, B3**	**A2, B2**	**A1, B1**	**A>B**	**A<B**	**A=B**	**A>B**	**A<B**	**A=B**
A4>B4	**X**	**X**	**X**	**X**	**X**	**X**	**1**	**0**	**0**
A4<B4	**X**	**X**	**X**	**X**	**X**	**X**	**0**	**1**	**0**
A4=B4	**A3>B3**	**X**	**X**	**X**	**X**	**X**	**1**	**0**	**0**
A4=B4	**A3<B3**	**X**	**X**	**X**	**X**	**X**	**0**	**1**	**0**
A4=B4	**A3=B3**	**A2>B2**	**X**	**X**	**X**	**X**	**1**	**0**	**0**
A4=B4	**A3=B3**	**A2<B2**	**X**	**X**	**X**	**X**	**0**	**1**	**0**
A4=B4	**A3=B3**	**A2=B2**	**A1>B1**	**X**	**X**	**X**	**1**	**0**	**0**
A4=B4	**A3=B3**	**A2=B2**	**A1<B1**	**X**	**X**	**X**	**0**	**1**	**0**
A4=B4	**A3=B3**	**A2=B2**	**A1=B1**	**1**	**0**	**0**	**1**	**0**	**0**
A4=B4	**A3=B3**	**A2=B2**	**A1=B1**	**0**	**1**	**0**	**0**	**1**	**0**
A4=B4	**A3=B3**	**A2=B2**	**A1=B1**	**0**	**0**	**1**	**0**	**0**	**1**
A4=B4	**A3=B3**	**A2=B2**	**A1=B1**	**X**	**X**	**1**	**0**	**0**	**1**
A4=B4	**A3=B3**	**A2=B2**	**A1=B1**	**1**	**1**	**0**	**0**	**0**	**0**
A4=B4	**A3=B3**	**A2=B2**	**A1=B1**	**0**	**0**	**0**	**1**	**1**	**0**

References :

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan
- Letocha ; Introduction aux circuits logiques ; Edition Mc-Graw Hill.

- J.C. Lafont ; Cours et problèmes d'électronique numérique, 124 exercices avec solutions; Edition Ellipses.
- R. Delsol ; Electronique numérique, Tomes 1 et 2 ; Edition Berti
- P. Cabanis ; Electronique digitale ; Edition Dunod.
- M. Gindre ; Logique combinatoire ; Edition Ediscience.
- H. Curry, Combinatory Logic II. North-Holland, 1972
- J-P. Ginisti, La logique combinatoire, Paris, PUF (coll. « Que sais-je? » n°3205), 1997.

PW7 : Study and implementation of counter circuits

1. Objective :

- Study, verification, and implementation of SR flip-flops using logic gates.
- Study, verification, and implementation of D flip-flops.
- Study, verification, and implementation of T flip-flops.
- Study, verification, and implementation of JK flip-flops.

2. Theoretical reminder :

2.1.Definition of logic flip-flop:

A flip-flop or latch is a logic integrated circuit with one output and one or more inputs. The output can be at logic level **0** or **1**.

2.2.SR Flip-flop :

The SR flip-flop is a memory element whose output can be set to an active or inactive state by forcing temporary conditions on the S and R inputs.

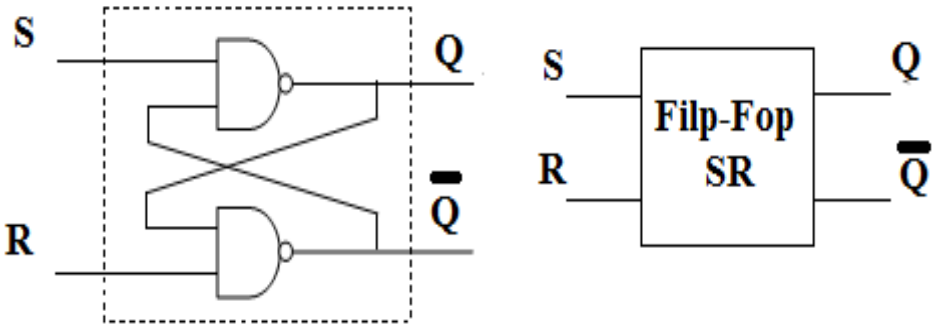


Fig.30 Realization of an SR flip-flop using NAND gates

2.3.D Flip-flop:

The D flip-flop is the most intuitive since its excitation is equal to its output. The clock is active on a transition and the D input is copied to the output at that time.

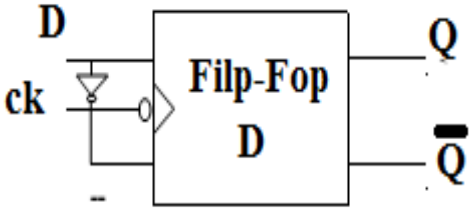


Fig.31 D Flip-flop

2.4.T Flip-flop:

T flip-flop is a memory element with only one input T. During a clock transition, the output will be inverted if T is active and will retain its state otherwise. It is a flip-flop that injects the outputs to the input.

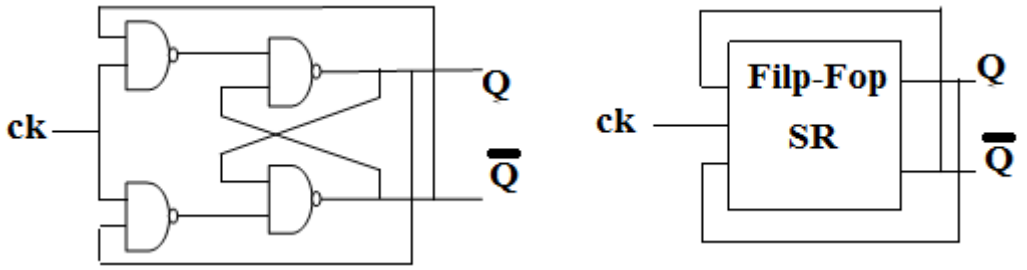


Fig.32 Creation of a T flip-flop from the SR flip-flop

2.5.JK Flip-flop :

It is a flip-flop with an overlapping clock input and two inputs to control j , k and two opposite outputs Q and $\overline{Q}$

k: Mode at 0 (output Q=0)

J: Mode at 1 (output Q=1)

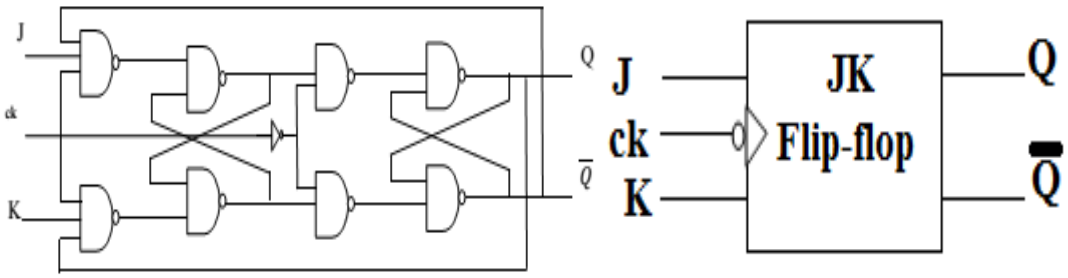
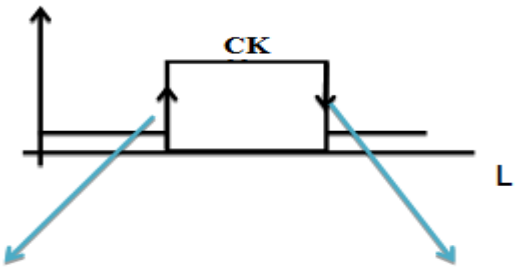


Fig.33 JK Flip-flop

- *Definition of a clock signal:* It is a series of periodic or non-periodic pulses. The clock signal input is represented by:

- CK(CLOCK)
- High logic level CK(1)
- Low logic level L(0)



Rising edge falling edge

✓ **Examples of industrial applications :**

- Machine cycle control

- Time measurement or frequency
- Batch packaging

3. Required equipment:

KL 31001 Digital logic lab: KL-33008/KL-33009 module; Connection wires.

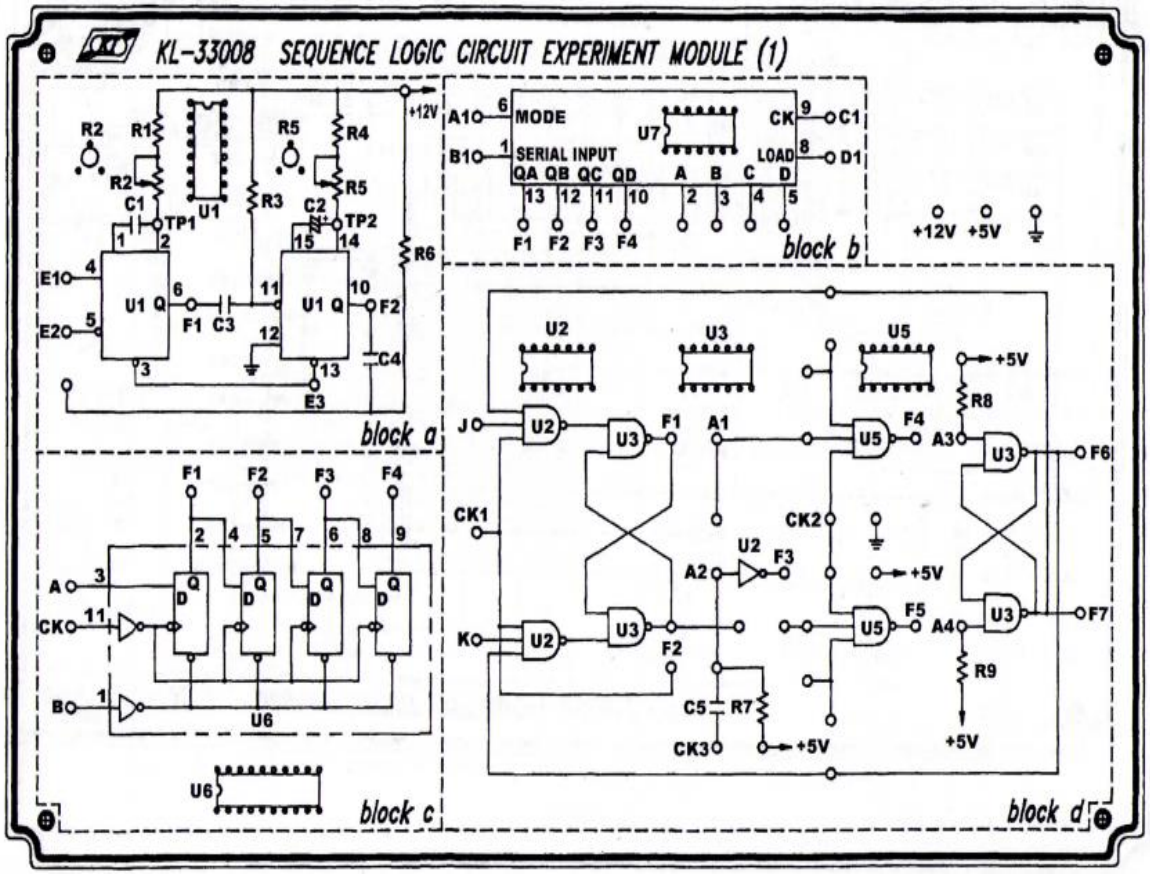


Fig.34 KL33008 module

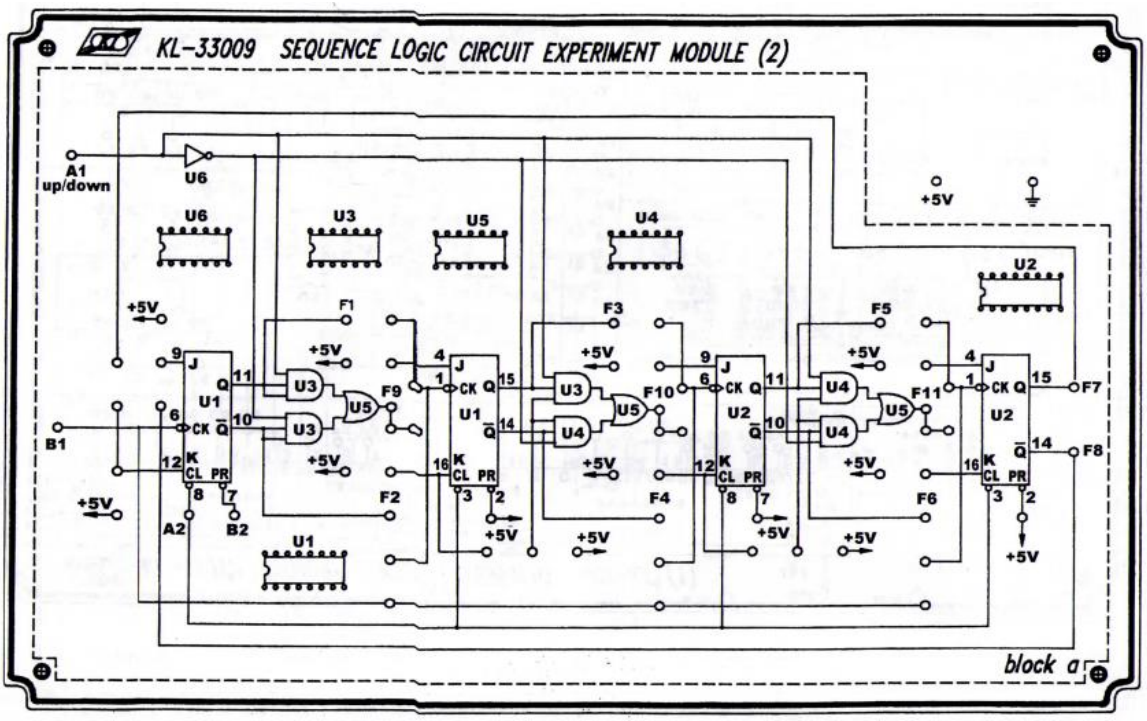


Fig.35 KL33009 module

4. Manipulation of different types of flip-flop :

4.1.SR Flip-Flop Manipulation:

Using the **KL-33008 block d** module, carry out the following assembly:

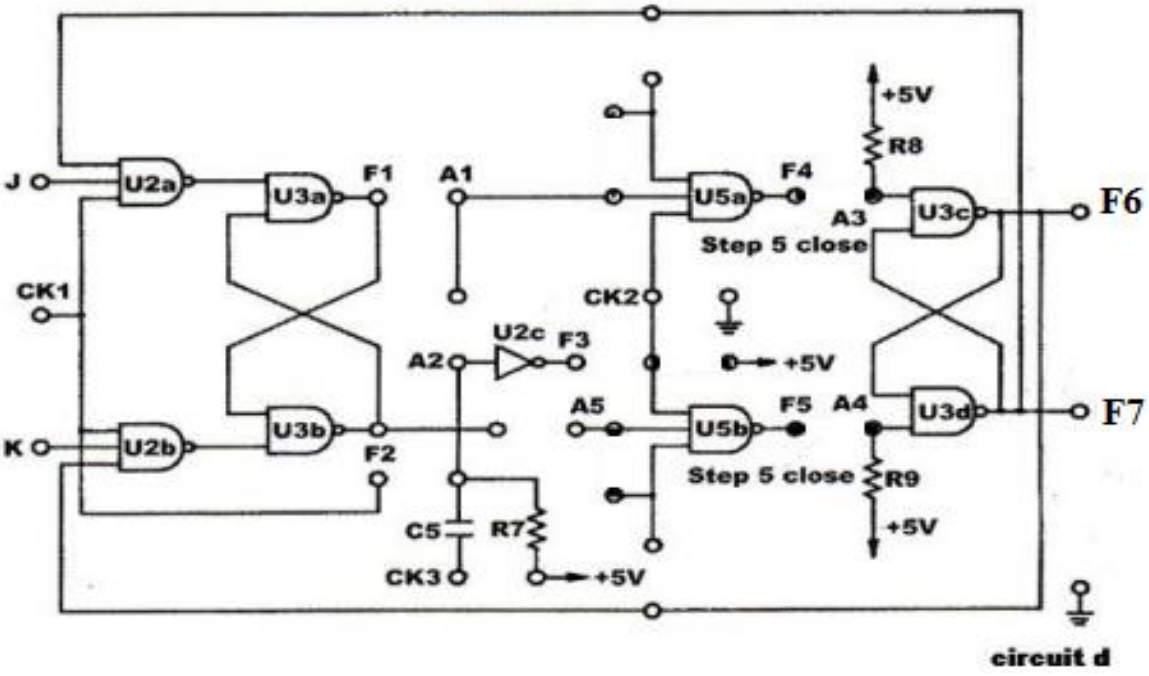


Fig.36 Circuit of SR Flip-Flop (KL33008 block d)

- Using **A3 and A4** as inputs, **F6 and F7** as outputs.

4.2.Manipulation of the D flip-flop:

Using the KL-33008 block module, the following assembly was carried out:

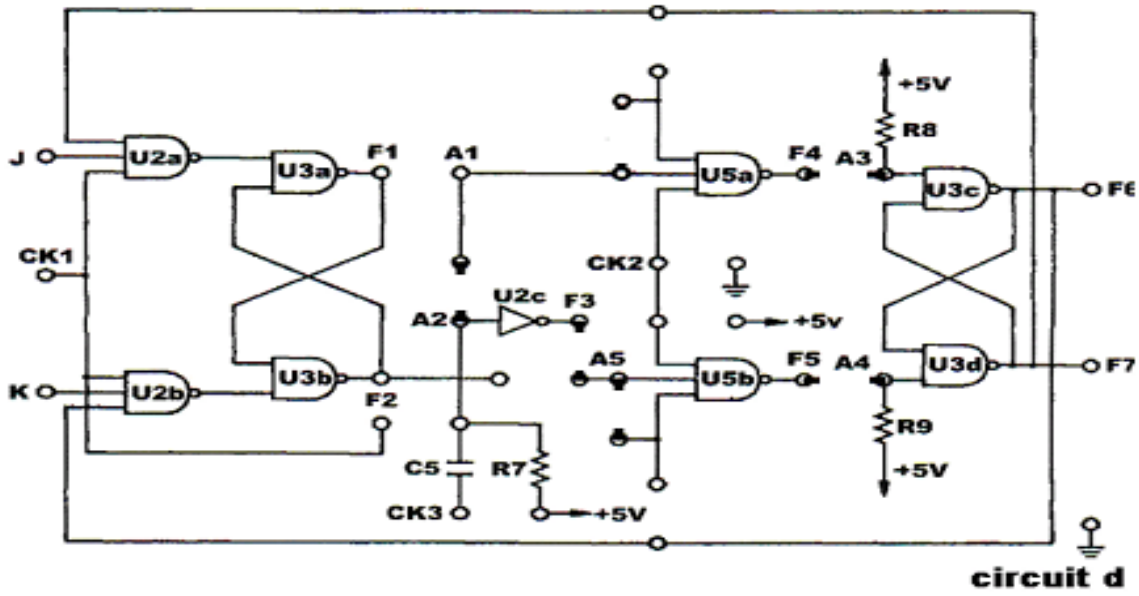


Fig.37 Circuit of D flip-flop (KL33008 block d)

- Using **A1** as input (connect **A1, A2, and F3, A5**), and **F6** as output.

4.3.Manipulation of the T flip-flop:

Using the KL-33008 block d module, carry out the following assembly:

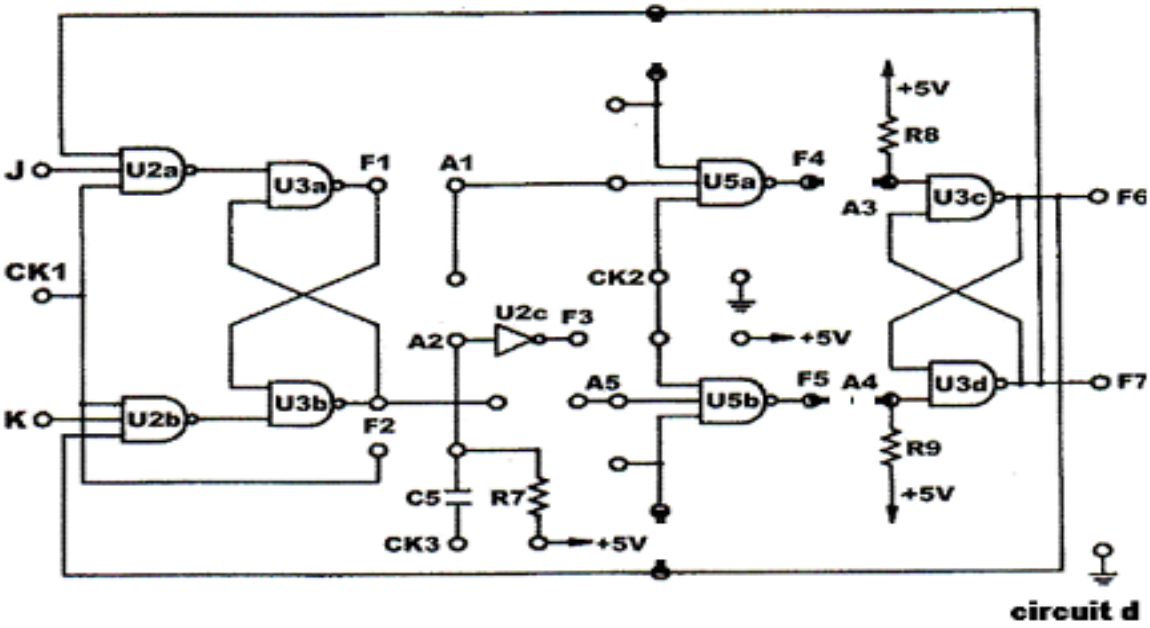


Fig.38 Circuit of T flip-flop (KL33008 block d)

- Using **A1 and A5** as inputs (connect F4, **A3 and F5, A4**), the outputs are **F6 and F7**.

4.4.Manipulation of the JK flip-flop:

Using the KL-33008 block d module, carry out the following assembly:

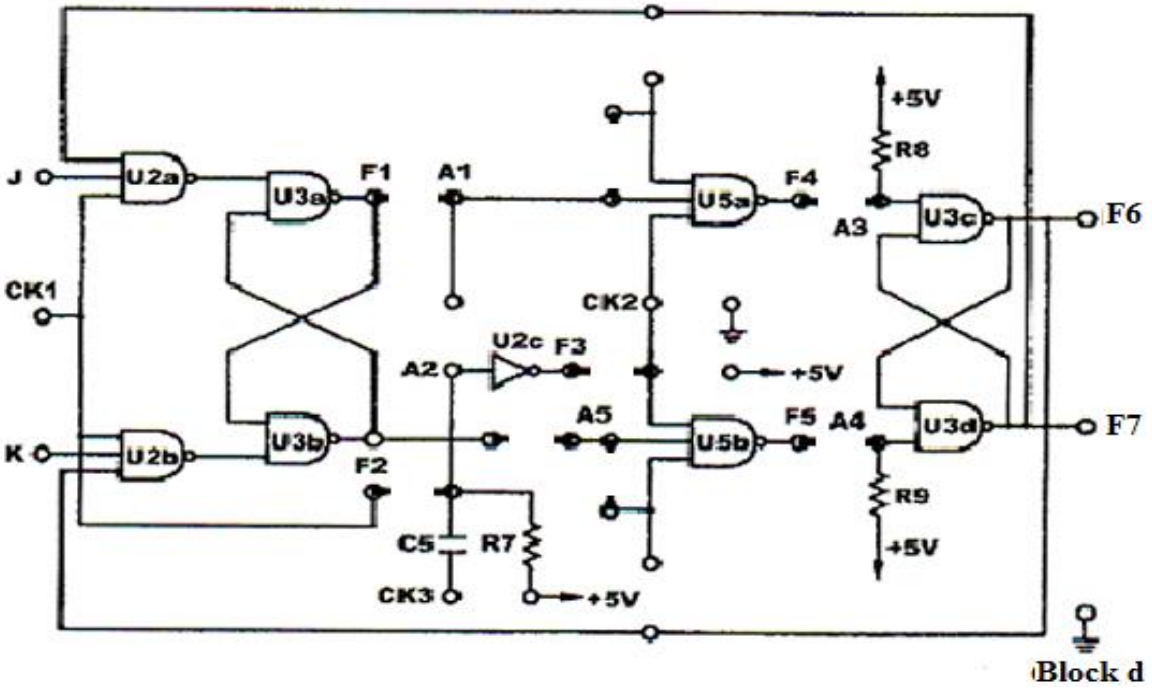
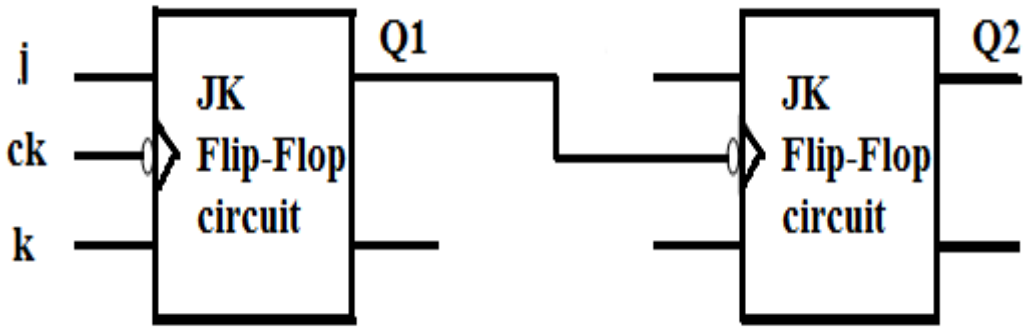


Fig.39 Circuit of JK flip-flop (KL33008 block d)

- SW0 (J) SW1 (K) are the inputs of the flip-flop (connect F1, A1 and F2, A5), the outputs are F1, F2, F6, and F7.

5. Counter :

It is a divider by two, to obtain a divider by four we adopt the following diagram:



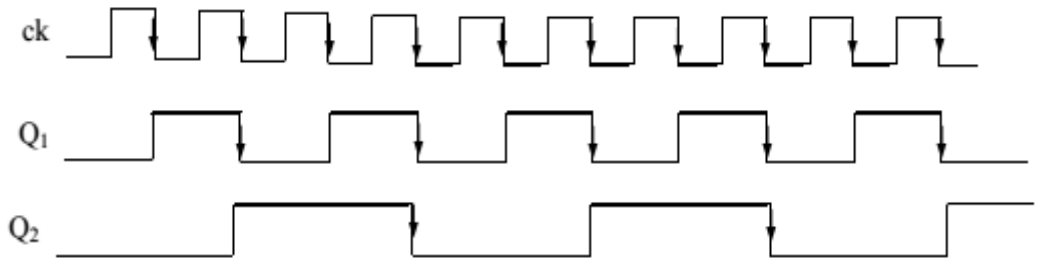


Fig.40 Construction of a counter based on a JK flip-flop (divider by four)

5.1.Manipulating an asynchronous counter:

Using KL 31001 Digital logic lab, module **KL-33009 block a**, carry out the assembly shown in the figure below:

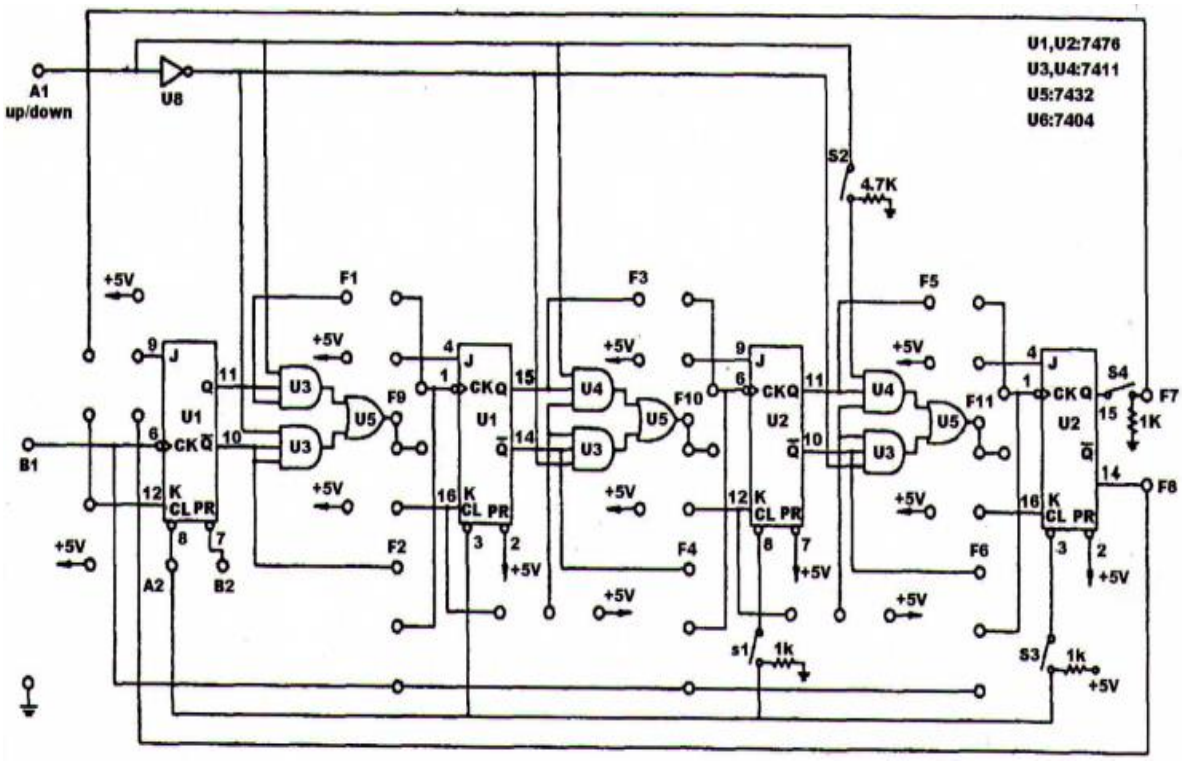


Fig.41 Module KL33009 block a

- Connect **A2 (Clear) to SW0**, **A1 (up/down) to +5V**, **B2 to +5V**, **B1 (CK) to a 1** Hz clock signal, and **F1, F3, F5,** and **F7** to the outputs.
- Initially set **SW0** to **"0"** to initialize the outputs, then set **SW0** to **"1"** to start counting and determine the modulo of this counter.

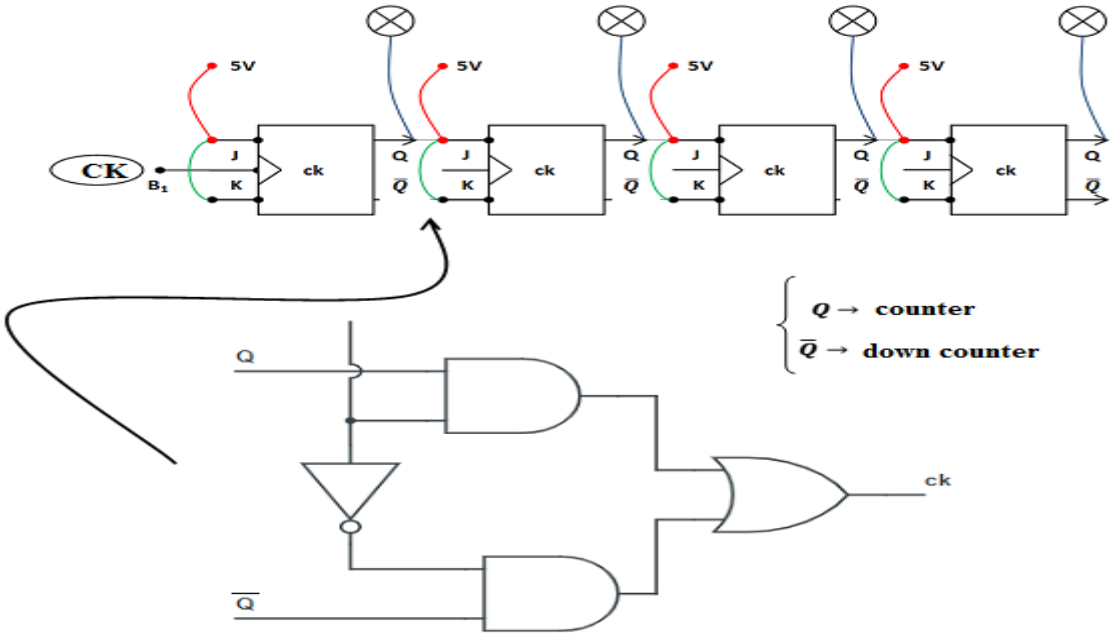


Fig.42 Simplified diagram of a JK flip-flop chain (counter)

- Check the operation of the counter according to the following table and see the clock signals and outputs **Q1, Q2, Q3, Q4** on the oscilloscope.

Table 18 Truth table of a counter

X_1	X_2	X_3	X_4	L	Q
0	0	0	0		
1	0	0	0		
0	1	0	0		
1	1	0	0		
0	0	1	0		
1	0	1	0		
0	1	1	0		
1	1	1	0		
0	0	0	1		
1	0	0	1		
0	1	0	1		
1	1	0	1		
0	0	1	1		
1	0	1	1		
0	1	1	1		
1	1	1	1		

6. *Down counter :*

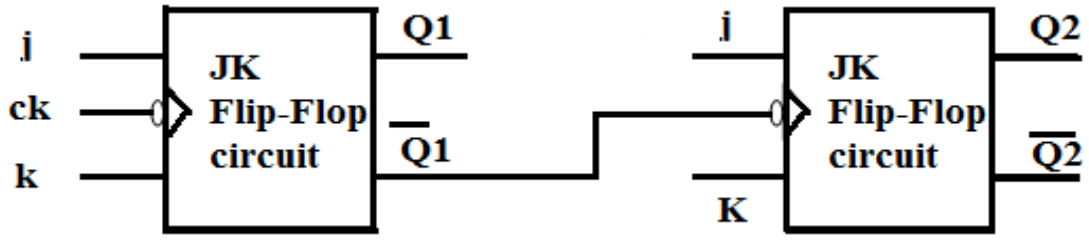


Fig.43 Simplified diagram of the JK flip-flop (down counter)

6.1.Asynchronous downcounter handling :

Using KL 31001 Digital logic lab, module KL-33009 block a, carry out the assembly shown in the figure below:

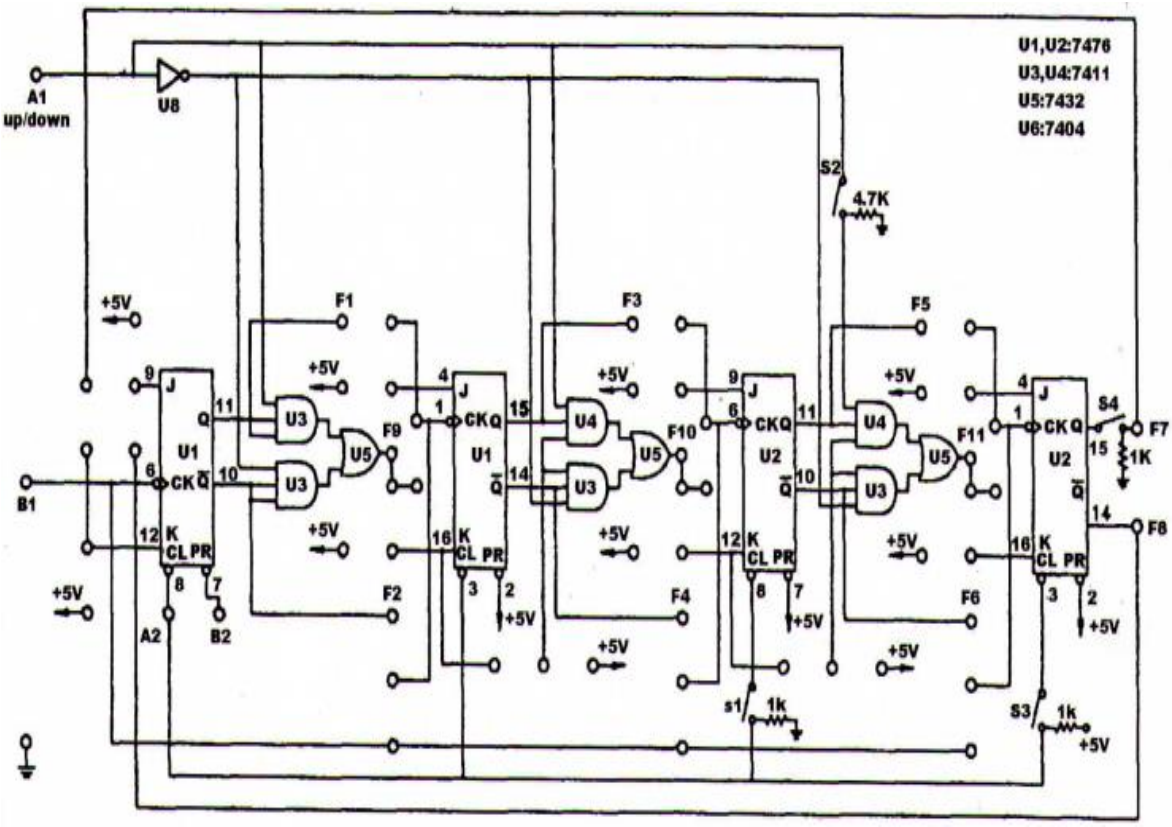


Fig.44 Module KL-33009 block a

- Connect **A2 (Clear) to SW0**, **A1 (up/down) to +5V**, **B2 to +5V**, B1 (CK) to a 1 Hz clock signal, **F2, F4, F6**, and **F7** to the outputs.
- Initially set **SW0 to "0"** to initialize the outputs, then set **SW0 to "1"** to start the downcounter and determine the modulo of this downcounter.

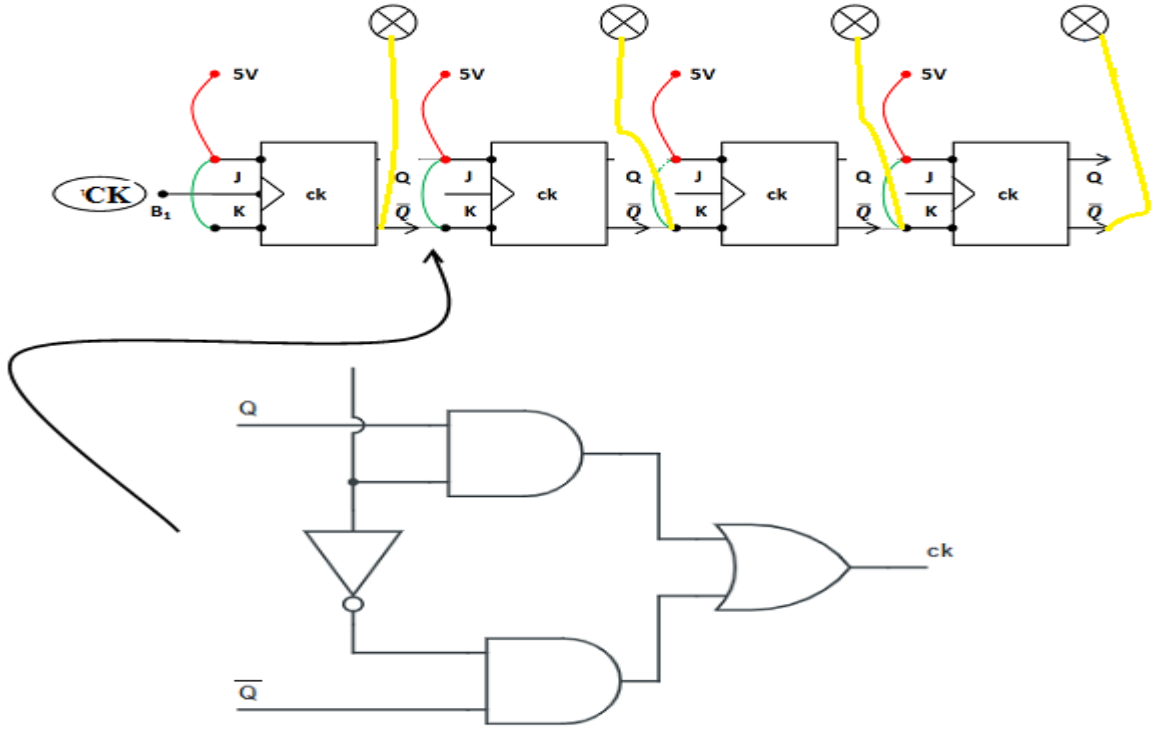


Fig.45 Simplified diagram of a JK flip-flop chain (down counter)

- Check the down counter operation according to the following table and see the clock signals and outputs $\overline{Q1}$, $\overline{Q2}$, $\overline{Q3}$, $\overline{Q4}$ on the oscilloscope.

Table 19 Truth table of a down counter

x_1	x_2	x_3	x_4	L	$\overline{Q}$
0	0	0	0		
1	0	0	0		
0	1	0	0		
1	1	0	0		
0	0	1	0		
1	0	1	0		
0	1	1	0		
1	1	1	0		
0	0	0	1		
1	0	0	1		
0	1	0	1		
1	1	0	1		
0	0	1	1		
1	0	1	1		
0	1	1	1		
1	1	1	1		

***References* :**

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan
- Letocha ; Introduction aux circuits logiques ; Edition Mc-Graw Hill.
- J.C. Lafont ; Cours et problèmes d'électronique numérique, 124 exercices avec solutions; Edition Ellipses.
- R. Delsol ; Electronique numérique, Tomes 1 et 2 ; Edition Berti
- P. Cabanis ; Electronique digitale ; Edition Dunod.
- M. Gindre ; Logique combinatoire ; Edition Ediscience.
- H. Curry, Combinatory Logic II. North-Holland, 1972
- J-P. Ginisti, La logique combinatoire, Paris, PUF (coll. « Que sais-je? » n°3205), 1997.

PW 8: Study and creation of registers

1. Objective:

- The study of the operation of shift registers.
- Understanding and implementing sequential logic circuits with memory register.
- Establishing various tables verifying the operation of sequential logic circuits with memory.

2. Theoretical reminders:

A logic register is a sequential circuit made up of flip-flops used to store and transfer binary information. Controlled by a clock signal, it allows a binary word to be temporarily stored and moved either in parallel (all bits at the same time) or in series (bit by bit). Depending on its structure, a register can perform different functions such as serial/parallel conversion, shifting bits to the left or right (shift register), or even introducing digital delays. Logic registers thus occupy an essential place in digital systems, since they are involved in the temporary storage of data, the transmission of information and the creation of more complex circuits such as counters and communication devices.

✓ **Industrial Applications**

- Signal Synchronization in a Production Line.
- Temporary data storage for motor control.
- Interface between analog sensors and industrial processor
- Serial to parallel data transmission.

3. Required equipment:

KL 31001 Digital logic lab: KL-33008 block c module; Connection wires.

- Connect **A to +5V.**
- Connect outputs **F1, F2, F3**, and **F4** to **L1, L2, L3,** and **L4.**
- Set **CK** to **clock generator (TTL output).**
- Connect **B** to **SW1 (TTL).**

- Build and verify the operation of the four-bit shift register and view the clock signals and outputs Q1, Q2, Q3, and Q4 on the oscilloscope:

The **KL 33008 block b** module represents a shift register:

- Connect **A, B, C, and D** to the inputs (**SW1, SW2, SW3, and SW4**)
- **F1, F2, F3, and F4** to the outputs (**L1, L2, L3, and L4**)

Connect **D1** (Load) to SWA output A, C1 (CK) to SWB output B, B1 to DIP2.0, and A1 (MODE) to DIP2.1

Following the A1 input sequence (0 0 1 1 1 0 1 1), observe and record the outputs.

***Reference* :**

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan
- Letocha ; Introduction aux circuits logiques ; Edition Mc-Graw Hill.
- J.C. Lafont ; Cours et problèmes d'électronique numérique, 124 exercices avec solutions; Edition Ellipses.
- R. Delsol ; Electronique numérique, Tomes 1 et 2 ; Edition Berti
- P. Cabanis ; Electronique digitale ; Edition Dunod.
- M. Gindre ; Logique combinatoire ; Edition Ediscience.
- H. Curry, Combinatory Logic II. North-Holland, 1972
- J-P. Ginisti, La logique combinatoire, Paris, PUF (coll. « Que sais-je? » n°3205), 1997.
- Logique combinatoire et séquentielle' Djamal Gozim,Kamel Guesmi, Licence. Al gérie. 2019, hal-02927680

Annexes :

- KL-300 Digital Logic Lab Manuels, by K & H MFG CO., Ltd Taiwan