بسم الله الرحمن الرحيم



Computer Systems Engineering Department

Digital Electronics and Organization Computer Lab

ENCS.211

Report for Experiment No.2

Adders, Subtractors, Comparators

Instruction: Dr. Khader Mohammad

Teaching Assistant: ENG. Haleema Hmedan.

Students Names & ID:

Nafe Abubaker, 1200047

Mohammad Abushama, 1200270

Mohammad Salem 1200651,

Date: 4.5.2023

Section (3)

# Abstract

The aim of this experiment is :

* to understand the construction and operating principle of digital comparators.
* To construct comparators with basic gates and ICs
* To implement half- and full adders using basic logic gates and ICs
* To implement a 4-bit adder unit(s)/ICs to add 4-bit numbers
* To understand the theory of complements
* To construct half- and full- subtractor circuits

Table of Contents

[Abstract I](#_Toc134128125)

[**Theory** 1](#_Toc134128126)

[**Half Adder** 1](#_Toc134128127)

[**Full Adder** 1](#_Toc134128128)

[**Half** **Subtractor** 1](#_Toc134128129)

[**Full** **Subtractor** 1](#_Toc134128130)

[**Comparator** **Circuit** 2](#_Toc134128131)

[**Procedure** 3](#_Toc134128132)

[Constructing Comparator with Basic Logic Gates : 3](#_Toc134128133)

[Constructing Comparator with TTL IC 3](#_Toc134128134)

[**Half- and Full-Adder Circuits** 4](#_Toc134128135)

[**Half- and Full Subtractor Circuits** 6](#_Toc134128136)

[**Constructing 4-Bit Full-Adder with IC** 7](#_Toc134128137)

[**Constructing 4-Bit Full-Subtractor with IC** 8](#_Toc134128138)

[Constructing BCD Adder 9](#_Toc134128139)

[Constructing BCD Adder 11](#_Toc134128140)

[**High-Speed Adder Carry Generator Circuit** 12](#_Toc134128141)

[**Conclusion** 13](#_Toc134128142)

[**References** 14](#_Toc134128143)

# **Theory**

A digital system's logic circuit could be sequential or combinatorial. We want to find out more about adders and comparators in this experiment. Combinational circuits include four-bit parallel adders, half-adders, full-adders, and simple and sophisticated comparators. At these circuits, the output is decided only by the input, regardless of prior input.

**Half Adder**:

Two binary digits are accepted as inputs by the half adder, which also outputs two binary digits together with a carry bit and a sum bit. The half-adder is an example of a simple, functional digital circuit built from two logic gates. The half-adder multiplies binary values with one bit (AB). The two bits (S) and the carry (C) are added to generate the output.

**Full Adder:**

The full adder produces a sum output and an output carry from two input bits and an input carry. The full-adder circuit produces two one-bit binary numbers, a sum (S), and a carry (Cout), after adding three one-bit binary numbers (Cin, A, and B). The complete adder typically functions as one of a series of adders that add binary digits 8, 16, and so on.

**Half** **Subtractor:**

The 2's complement is typically used to execute binary subtraction. The 2's complement can be obtained in two steps. A "1" becomes a "0," and a "0" becomes a "1" as part of the subtrahend's first inversion to 1's complement. Second, the least important portion of the complement of 1 is given a "1" addition. Regardless of whether the minimum value is bigger or less than the subtrahend, a half-subtractor executes the process of subtraction one bit at a time. There is no consideration for "borrow" from earlier subtraction

**Full** **Subtractor:**

full subtractor is a combinational circuit that performs subtraction of two bits, one is minuend and other is subtrahend, taking into account borrow of the previous adjacent lower minuend bit. This circuit has three inputs and two outputs. The three inputs A, B and Bin, denote the minuend, subtrahend, and previous borrow, respectively. The two outputs, D and Bout represent the difference and output borrow, respectively. Although subtraction is usually achieved by adding the complement of subtrahend to the minuend, it is of academic interest to work out the Truth Table and logic realization of a full subtractor; x is the minuend; y is the subtrahend; z is the input borrow; D is the difference; and B denotes the output borrow. The corresponding maps for logic functions for outputs of the full subtractor namely difference and borrow.

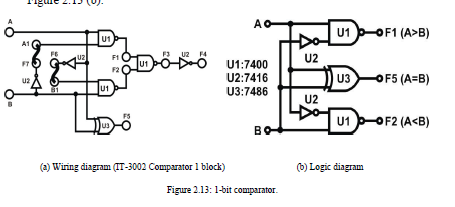
**Comparator** **Circuit:**

Any comparison must use a minimum of two numbers. The comparator's two inputs are its most basic configuration. There are three potential outputs if the two inputs are A and B: B > A, B = A, and A. In actual applications, 4-bit comparators are used most often. Each bit in a 4-bit comparator stands for 20, 21, 22, and 23. The "A>B" output will be in the high state if input A is bigger than input B at the 23 bits of the comparison, which starts at the comparison's most important bit (23). The comparison will be made at the next highest bit (22), if A and B are equal at the 23 bits. If this bit still yields no results, the procedure is repeated at the following bit. If the inputs are still equal at bit (20), the "A=B" output will be in the high state..

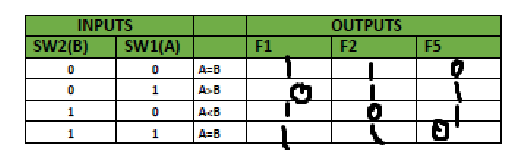
# **Procedure**

## Constructing Comparator with Basic Logic Gates :

As a first step, we Set the KL-26001 Module on the KL-22001, and locate block a, then we completed the connections by referring to wiring diagram.



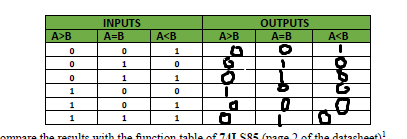
The inputs were active high. We Connected inputs A and B to Data Switches SW1 and SW2 (The outputs are active low). Then we Connected outputs F1, F2, F5 to logic Indicators L1, L2, L3, respectively and we Applied +5 VDC from the Fixed Power on KL-26001 Module. We followed, the input sequences in the following table. And we recorded the outputs we observed, and got the following results.



## Constructing Comparator with TTL IC



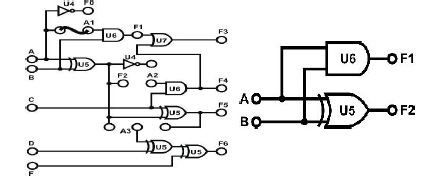
We Set the KL26005 Module on the KL-22001, and locate it in the above block, we apply +5VDC from the Fixed Power on KL-22001 Lab to KL26005 Module. U6 is a 7485 4-bit comparator IC. Its pin assignment and function table are given below:



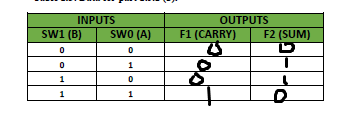
## **Half- and Full-Adder Circuits**

We Set the KL-26002 Module on the KL-22001, and locate block a. we completed the connections by referring to the wiring diagram in this figure, we Applied +5VDC from Fixed Power on the KL-22001 Lab to KL-26002 Module

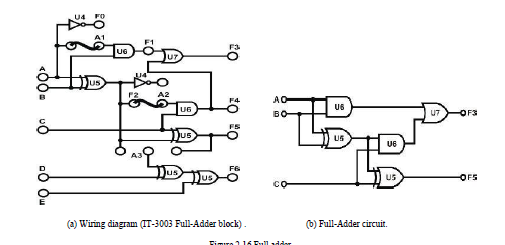
Half Adder :



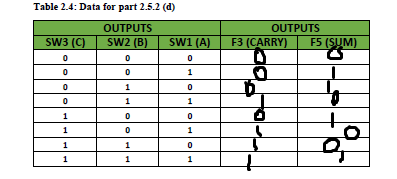
And we reached to this result by controlling of input and observing the output after we connecting the wires as shown in the next table



Full adder :



And after connecting the wires we got the following results :



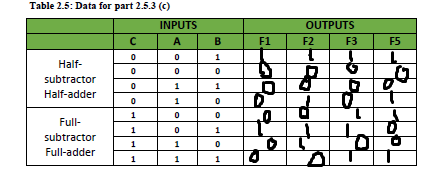
## **Half- and Full Subtractor Circuits**



As the same connection we made from the last experiment, we

1. Connected inputs A~C to Data Switches SW0~SW2; Outputs F2 to Logic Indicator L1; F1 to L2; F3 to L3; and F5 to L4. When C=0 the circuit is a half-subtractor. F1 is the borrow output; F2 is the difference and F5=F2; F4=0; F3=F1. When C=1 the circuit is a full-subtractor. F3 is the borrow output and F5 is the difference output.

And the following table shows the results.

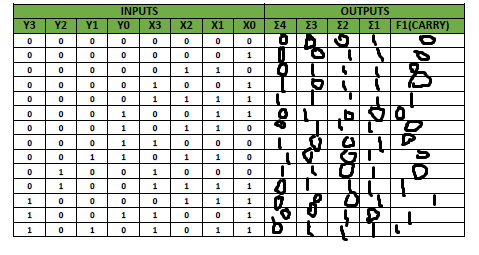


1. .

## **Constructing 4-Bit Full-Adder with IC**

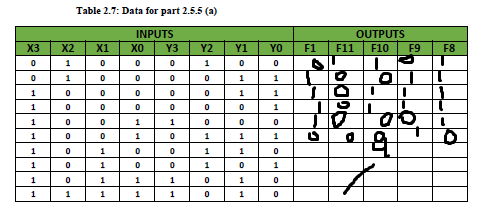


By reading the lab manual we got the following results in the table :



## **Constructing 4-Bit Full-Subtractor with IC**





## Constructing BCD Adder



|  |  |  |  |
| --- | --- | --- | --- |
| *X3 X2 X1 X0* | *Y3 Y2 Y1 Y0* | *OUTPUTS (U5)*  *F1 F11  F10 F9 F8* | *FINAL*  *F2 F3 F7 F6 F5 F4* |
| *0 0 0 0* | *0 0 0 0* | *0 0 0 0 0* | *0 0 0 0 0 0* |
| *0 0 0 1* | *0 0 1 1* | *0 0 1 0 0* | *0 0 0 1 0 0* |
| *0 0 1 1* | *0 1 0 0* | *0 0 1 1 1* | *0 0 0 1 1 1* |
| *0 0 1 0* | *0 0 1 0* | *0 0 1 0 0* | *0 0 0 1 0 0* |
| *0 0 1 0* | *1 0 0 0* | *0 1 0 1 0* | *1 1 0 0 0 0* |
| *0 0 1 1* | *0 1 1 0* | *0 1 0 0 1* | *0 0 1 0 0 1* |
| *0 1 0 0* | *0 0 1 0* | *0 0 1 1 0* | *0 0 0 1 1 0* |
| *0 1 0 0* | *0 1 0 1* | *0 1 0 0 1* | *0 0 1 0 0 1* |
| *0 1 0 0* | *0 1 1 0* | *0 1 0 1 0* | *0 1 0 0 0 0* |
| *0 1 0 1* | *0 1 1 0* | *0 1 0 1 1* | *0 1 0 0 0 1* |
| *0 1 1 0* | *0 1 1 1* | *0 1 1 0 1* | *0 1 0 0 1 1* |
| *0 1 1 1* | *1 0 0 0* | *0 1 1 1 1* | *0 1 1 0 0 0* |
| *0 1 1 1* | *1 0 0 1* | *1 0 0 0 0* | *1 1 0 1 0 1* |
| *1 0 0 0* | *1 0 0 1* | *1 0 0 0 1* | *1 1 0 1 1 1* |
| *1 0 0 1* | *1 0 0 1* | *1 0 0 1 0* | *1 1 1 0 0 0* |
| *1 0 1 0* | *1 0 1 0* | *1 0 1 0 0* | *1 1 1 0 1 0* |
| *1 0 1 0* | *1 0 1 1* | *1 0 1 0 1* | *1 1 1 0 1 1* |
| *1 0 1 0* | *1 1 0 0* | *1 0 1 1 0* | *1 1 1 1 0 0* |
| *1 0 1 1* | *1 1 1 0* | *1 1 0 0 1* | *1 1 1 1 1 1* |
| *1 1 1 1* | *1 1 1 1* | *1 1 1 1 0* | *1 0 0 1 0 0* |

## Constructing BCD Adder



## **High-Speed Adder Carry Generator Circuit**

The time of the lab was not enough to complete it.

# **Conclusion**

The experiment went smoothly with no complications, it took me about hour and a half to finish it, and the results satisfied the theory part of the report, the experiment helped with understanding more about comparators and adders and subtractors

The time of the lab was not enough for some experiments, I suggest extending the time of the lab.

# **References**

Digital Lab manual