Faculty of Engineering and Technology
Electrical and Computer Engineering Department

DIGITAL ELECTRONICS AND COMPUTER ORGANIZATION LABORATORY
ENCS2110

Experiment No. 7
Constructing Memory Circuits Using Flip−Flops
&
Experiment No. 11
Arithmetic Elements

Prepared by:

Rasha Mansour	1210773.

Partners:

Fatima Azazmah	1200400
Mariam Awwad	1213069

Instructor: Dr.Khader Mohammad
Teaching assistant: Haleema Hmedan

Section:7
Date: 6/12/2023

Abstract

This report documents the findings and observations from two experiments conducted in the digital lab. Experiment 7 focuses on constructing memory circuits using flip-flops, specifically exploring the structure and functionality of a 64-bit Random Access Memory (RAM). Experiment 11 delves into the realm of Arithmetic Logic Units (ALUs), exploring their functions and applications through the use of the 74181 ALU IC.

Table of Contents:

Table of Figures

List of Tables

1. Theory

1.1 Experiment 7 - Constructing Memory Circuits Using Flip-Flops

1.1.1 Objectives

- Understand the basic structure of Random-Access Memory (RAM).
- Understand and test the circuit of 64-bit Random Access Memory (RAM).

1.1.2 Equipment Required

- IT-3000 Basic Electricity Circuit Lab
- IT-3011 Memory Circuits.

1.1.3 Theory

- Constructing Random Access Memory (RAM) with D Flip-Flop
- 64-Bit Random Access Memory (RAM) Circuit

1.1.4 Procedure

- RAM block with D Flip-Flop
- Latches and Flip-flops

Procedure CONSTRUCTING RANDOM ACCESS MEMORY (RAM) WITH D FLIP-FLOP [1]

“The input and output are not separated. There are two control terminals: one is the R/W terminal (R for READ or OUTPUT, W for WRITE or INPUT) and the other one is the ENABLE terminal.

When CS=0, tri-state gates U1 and U2 do not operate, so data input is not possible, the flip- flop output Q is not sent to the I/O terminal.

- When CS=1, W/R controls the D flip-flop.

When W/R' =1, U1 opens but U2 does not, I/O will accept data input.

If W/R'=0, the exact opposite will happen and I/O act as the data output. another connection that will increase the RAM capacity.

When CS1=1, RAM1 I/O1 and I/O2 are selected.

Address line A is used to select between RAM1 and RAM2.

Since there is only one address line, we can only select from 2 RAMs, each CS can only select a 2-bit RAM, so the total capacity is 2×2. 1), a 2-address (2-bit) RAM circuit has independent input and output. When Address=0, input D1 is enabled and the content of D1 will be made available at the output. When Address=1, input D2 is enabled and the content of D2 will be made available at the output. The ENABLE terminal must be activated to allow the output to correspond with the constantly changing inputs.

Figure 1: one bit memory using D-flip flop, two-bit memory with 1-2 decoder.

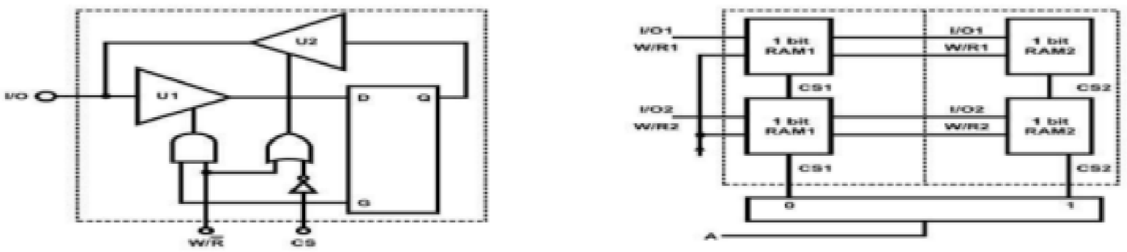
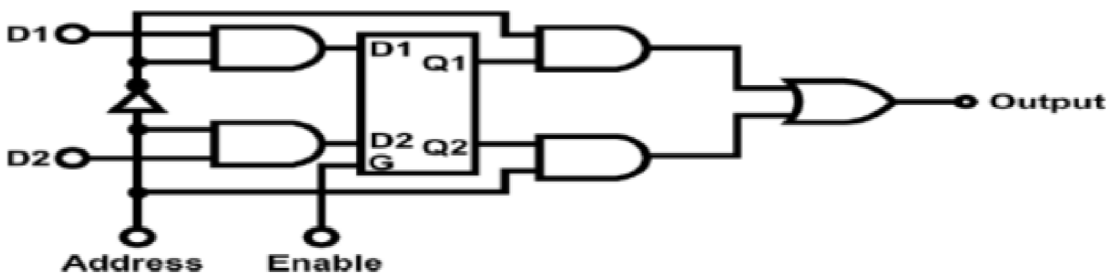
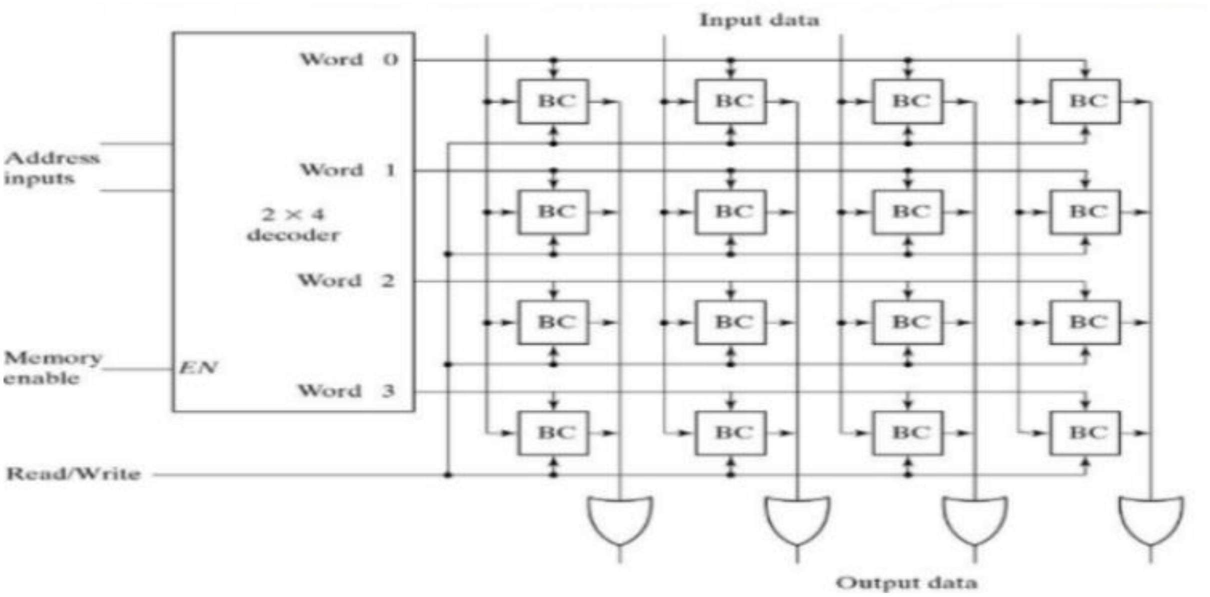


Figure 2: implementation of the two-bit memory.



Commercial random-access memories may have a capacity of thousands of words and each word may range from. The logical construction of a large capacity memory would be a direct extension of the configuration. The two address inputs go through a 2x4 decoder to select one of the four words." [1]

*Figure 3: logical construction of 4*4 RAMS.*



64-BIT RANDOM ACCESS MEMORY (RAM) CIRCUIT "Like ROM, RAM is also a memory element.

The data selection process is controlled by the address selectors. The length of data is related to the number of data variations. For example, if there are 4 data then 2^4 or 16 data variations exist. The number of address lines determines the number of locations. If there are 4 address lines, then 2^4or 16 locations exist. A 4-bit data can be stored in each location, since the total capacity is 16×4, where the 4 is the number of data while 16 is the number of address lines.".[1]

*Figure 4: 16*4 RAM chip.*

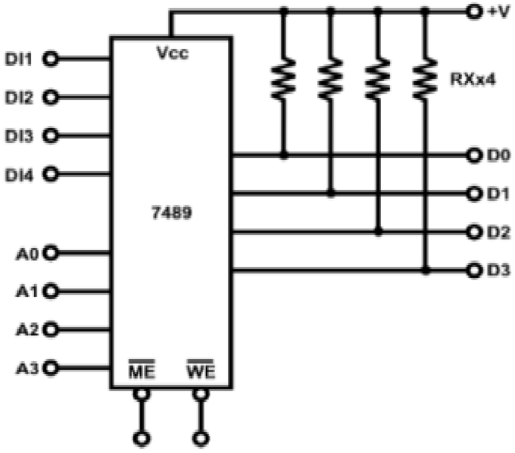
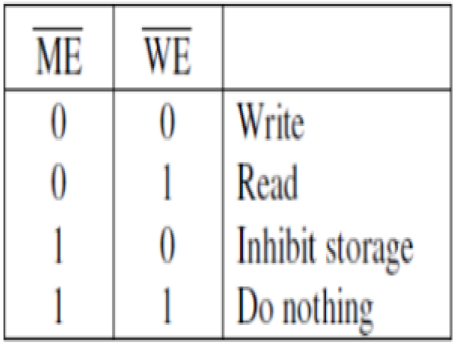


*Table1: 16*4 RAM chip.*



$\overline{ME}$	$\overline{WE}$	
0	0	Write
0	1	Read
1	0	Inhibit storage
1	1	Do nothing

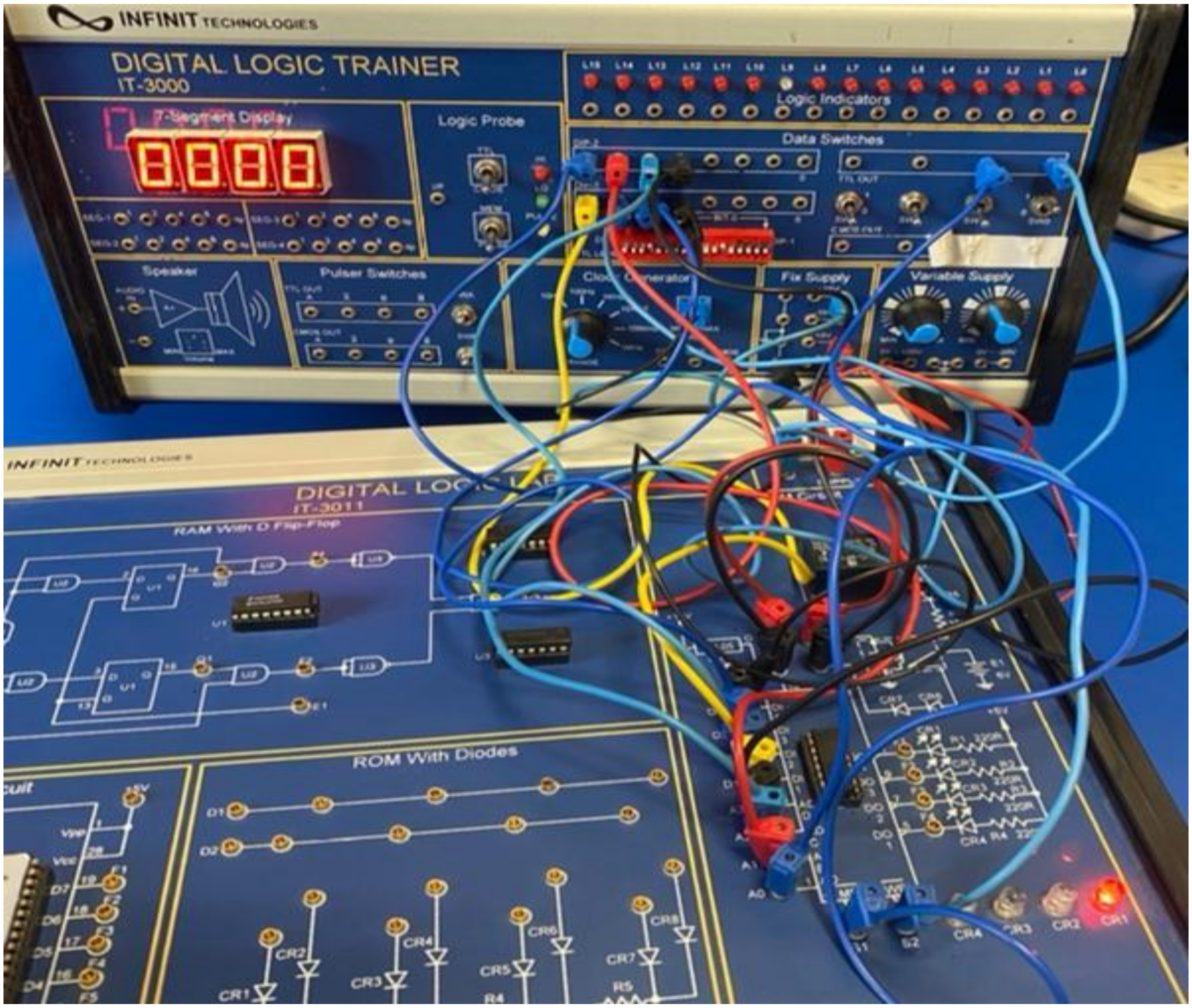
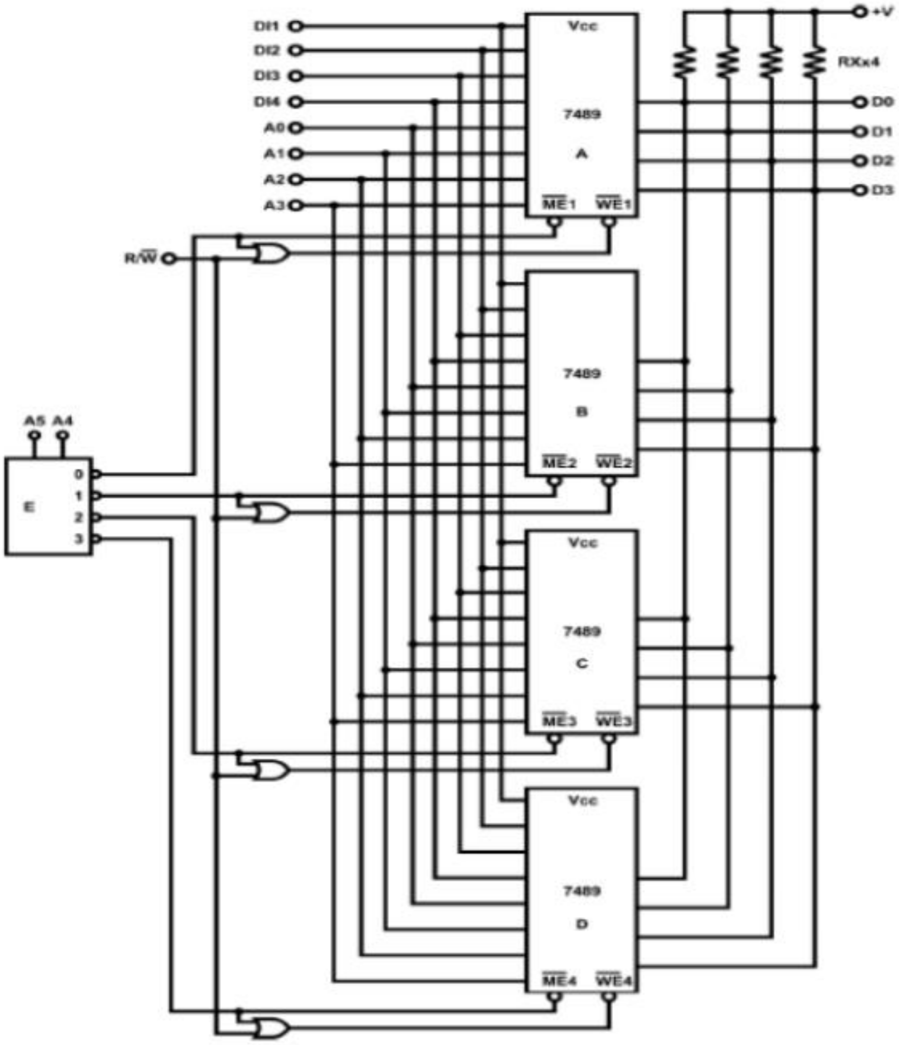


*Figure5: RAM 16*4 circuit*

*Figure6: Block diagram of RAM16*4 chip.*



Procedure Latches and Flip Flops [2]

A- we use RAM block with D Flip-Flop of module IT-3011, Connect E1, S1, D2, and D1 to Data Switch. And Connect outputs F1, F2, and F3 to Logic Indicators. Then Refer to the input sequence.

B- Connect E1, S1, D2, D1 to Data Switch SW0~SW3 respectively. Connect outputs F1, F2, F3 to Logic Indicators L1~L3.

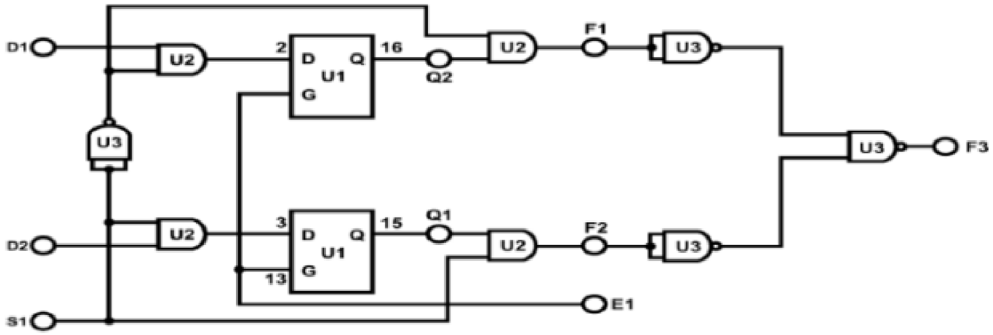


Figure 7: RAM block with D flip flop of module IT-3011.

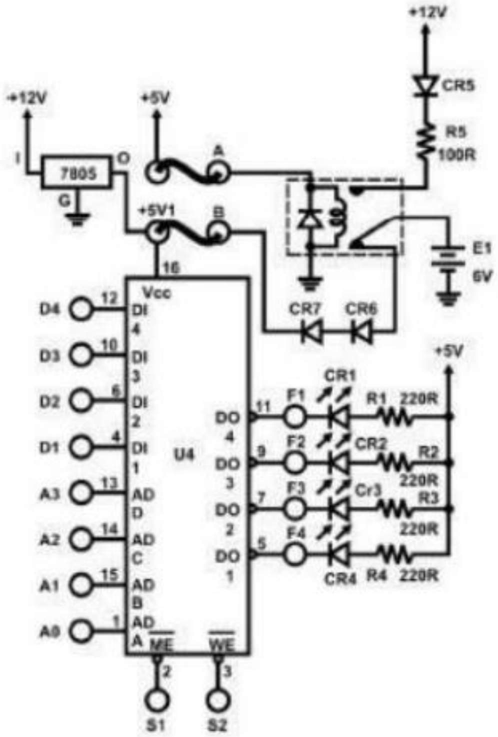
Table2: Storing.

Input				output		
E1	S1	D1	D2	F3	F2	F1
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	0	0	0
0	0	1	1	0	0	0
0	1	0	0	0	0	0
0	1	0	1	0	0	0
0	1	1	0	0	0	0
0	1	1	1	0	0	0
1	0	0	0	0	0	0
1	0	0	1	1	0	1
1	0	1	0	0	0	0
1	0	1	1	1	0	1
1	1	0	0	0	0	0
1	1	0	1	0	0	0
1	1	1	0	1	1	0
1	1	1	1	1	1	0

When E = 0 them the output will be zero, but when it becomes one, the outputs depend on other inputs value, when E = 1 and S1 = 0 the output F1 become like D1. when E1 = 1 and S1 = 1 the output F2 will be equal to D2.

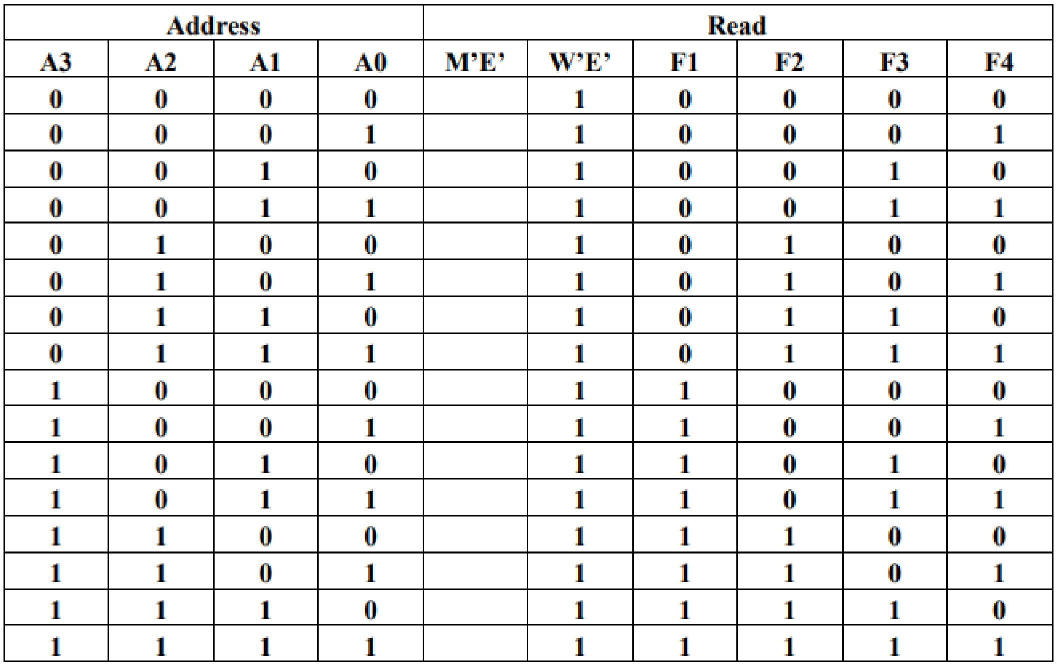
C- Then, set module IT-3011 and locate block RAM Circuit. Insert connection clip, connect +5V, +15V of module IT-3011 to the +5V, 15V output of fixed power supply respectively.

*Figure 8: open collector 4*4 RAM.*



Address				Write					
A3	A2	A1	A0	M’E’	W’E’	D1	D2	D3	D4
0	0	0	0		0	0	0	0	0
0	0	0	1		0	0	0	0	1
0	0	1	0		0	0	0	1	0
0	0	1	1		0	0	0	1	1
0	1	0	0		0	0	1	0	0
0	1	0	1		0	0	1	0	1
0	1	1	0		0	0	1	1	0
0	1	1	1		0	0	1	1	1
1	0	0	0		0	1	0	0	0
1	0	0	1		0	1	0	0	1
1	0	1	0		0	1	0	1	0
1	0	1	1		0	1	0	1	1
1	1	0	0		0	1	1	0	0
1	1	0	1		0	1	1	0	1
1	1	1	0		0	1	1	1	0
1	1	1	1		0	1	1	1	1

Table3: Write.



Address				Read					
A3	A2	A1	A0	M’E’	W’E’	F1	F2	F3	F4
0	0	0	0		1	0	0	0	0
0	0	0	1		1	0	0	0	1
0	0	1	0		1	0	0	1	0
0	0	1	1		1	0	0	1	1
0	1	0	0		1	0	1	0	0
0	1	0	1		1	0	1	0	1
0	1	1	0		1	0	1	1	0
0	1	1	1		1	0	1	1	1
1	0	0	0		1	1	0	0	0
1	0	0	1		1	1	0	0	1
1	0	1	0		1	1	0	1	0
1	0	1	1		1	1	0	1	1
1	1	0	0		1	1	1	0	0
1	1	0	1		1	1	1	0	1
1	1	1	0		1	1	1	1	0
1	1	1	1		1	1	1	1	1

Table4: Read.

When switch W’E’ = 0 the input will work to store data and write it at the memory, and when it’s 1 it will work to show data who stored at the memory.

1.2 Experiment 11 - Arithmetic Elements

1.2.1 Objectives

- Understand functions and applications of the ALU (arithmetic logic unit).

- Perform arithmetic and logic operations using the 74181 ALU IC.

1.2.2 Equipment Required

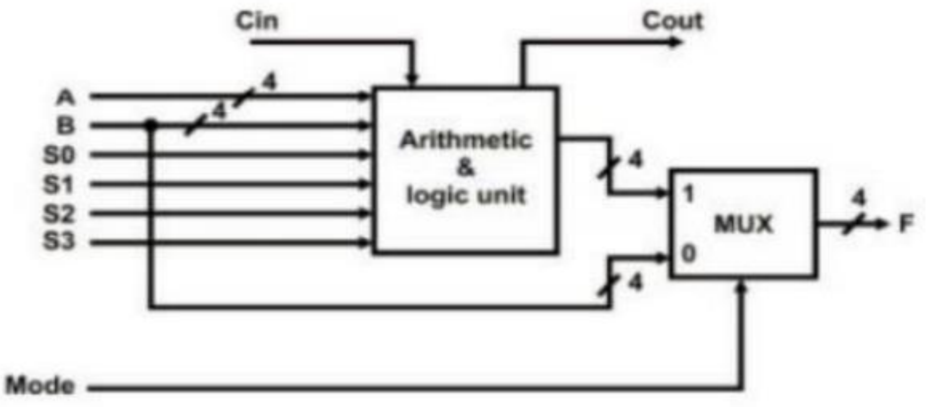
- IT-3000 Basic Electricity Circuit Lab

- IT-3003 Module.

1.2.3 Theory

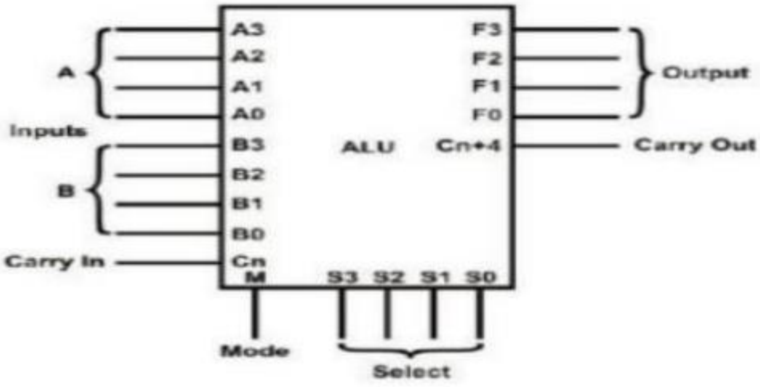
- Arithmetic Logic Unit (ALU) Circuit

Figure 9: logical diagram of the ALU.



“The circuit has two 4-bit inputs A and B, as well as a “carry-in” (Cn) input. There is a mode control input (M) and 4 function-select lines S0, S1, S2, S3 forming logic or arithmetic, operations, Also, it has a 4-bit output (F3~F0); a “carry-out” or “Cn+4” output. The biggest advantage of the design is its ability to perform arithmetic functions such as addition, subtraction. multiplication; and logic functions such as AND, XOR functions. The mode control input (M) and function-select lines (S0~S3) determines which function it will perform.” [1]

Figure10: Pin assignment.



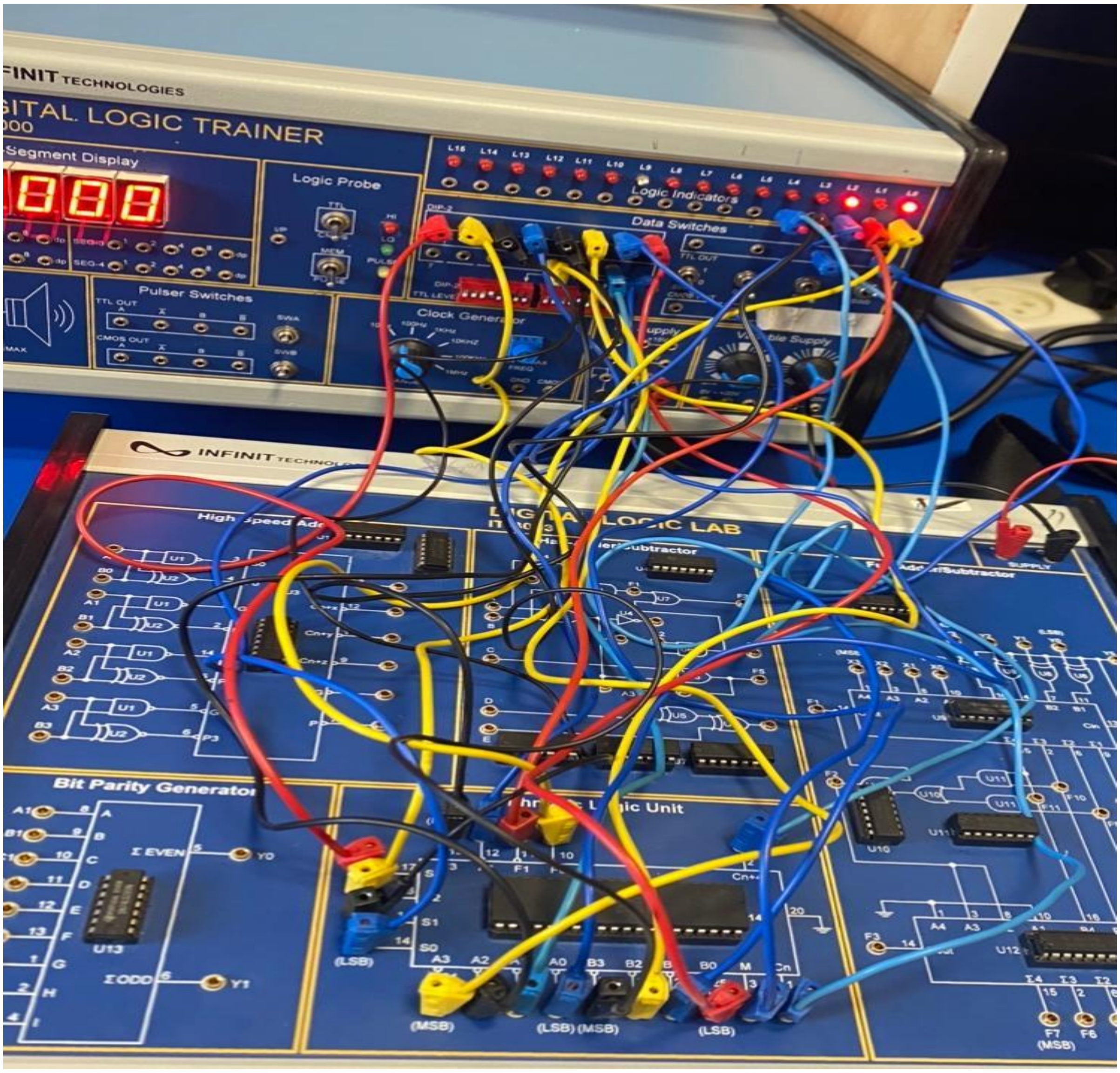


Figure11: ALU circuit.

1.2.4 Procedure

- ALU Circuit setup and connections

- Set Cn to “0” and ignore the previous carry. When S3S2S1S0=0000 perform the addition.

What is the output when A3A2A1A0=0000 and B3B2B1B0=1111?

F3F2F1F0= 1111; Cn+4= 0

What is the output when A3A2A1A0=1001 and B3B2B1B0=0100?

F3F2F1F0= 1101; Cn+4= 0

- Set Cn to “1” and add the previous carry.

When S3S2S1S0=0000 perform the addition. What is the output when A3A2A1A0=0000 and B3B2B1B0=1111?

F3F2F1F0= 0000; Cn+4= 1

What is the output when A3A2A1A0=1001 and B3B2B1B0=0100?

F3F2F1F0= 1110; Cn+4= 0

- Set Cn to “0”.

When S3S2S1S0=0001 perform the subtraction. What is the output when A3A2A1A0=0000 and B3B2B1B0=1111?

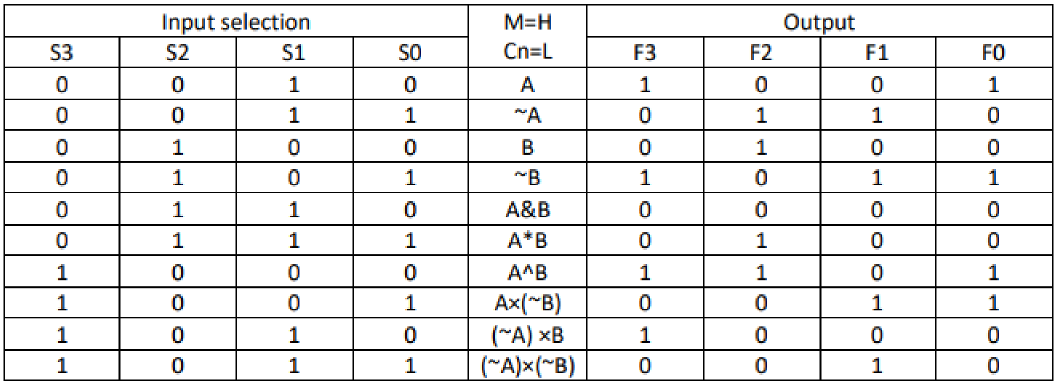
F3F2F1F0= 1111; Cn+4= 0

What is the output when A3A2A1A0=1001 and B3B2B1B0=0100?

F3F2F1F0= 0101; Cn+4= 0

**The asthmatic operation on the ALU, addition and subtraction and the operation determined by the value of the switch if its = 0000 addition and if its = 0001 subtraction.

set M to “1” to perform the following arithmetic and logic Set inputs sequence A0~A3=A, B0~B3=B from DIP switches.



Input selection				M=H	Output			
S3	S2	S1	S0	Cn=L	F3	F2	F1	F0
0	0	1	0	A	1	0	0	1
0	0	1	1	~A	0	1	1	0
0	1	0	0	B	0	1	0	0
0	1	0	1	~B	1	0	1	1
0	1	1	0	A&B	0	0	0	0
0	1	1	1	A*B	0	1	0	0
1	0	0	0	A^B	1	1	0	1
1	0	0	1	A×(~B)	0	0	1	1
1	0	1	0	(~A) ×B	1	0	0	0
1	0	1	1	(~A)×(~B)	0	0	1	0

Table 5: ALU.

This ALU do 10 Logical operations, so it has a four-bit selection to implement them, the value of A and B

we start with:

//A => 1001

//B => 0100

2. Discussion

2.1 Experiment 7 - Constructing Memory Circuits Using Flip-Flops

2.1.1 Observations and Results

The construction of Random-Access Memory (RAM) using D Flip-Flops provided valuable insights into the functionality of memory circuits. We observed that the separation of input and output terminals, controlled by the R/W (Read/Write) and ENABLE terminals, plays a crucial role in managing the data flow. The 64-bit RAM circuit exhibited the expected behavior based on the address selectors, with the number of address lines determining the storage capacity.

2.1.2 Circuit Performance

The 7489 IC, representing a 16x4 RAM chip, demonstrated effective memory operations. The control sequences ME' (Memory Enable) and WE' (Write Enable) dictated the read and write processes. The open-collector output terminals, allowed for parallel connection, enhancing the flexibility of data retrieval.

2.1.3 Discussion

The procedure allowed us to observe the behavior of the RAM circuit under different conditions. The WRITE and READ tasks demonstrated the circuit's ability to store and retrieve data reliably. The data retention test confirmed the stability of stored information even after power cycles.

2.2 Experiment 11 - Arithmetic Elements

2.2.1 ALU Operations

The Arithmetic Logic Unit (ALU) experiment with the 74181 IC proved to be insightful. The ALU, with its capability to perform various arithmetic and logic operations, showcased its versatility. The control inputs (M and S0-S3) effectively determined the nature of operations, and the 4-bit inputs A and B, along with the carry input (Cn), contributed to the successful execution of these operations.

2.2.2 Functionality Verification

Through a series of carefully controlled scenarios, we validated the ALU's ability to handle addition, subtraction, multiplication, AND, and XOR operations. The experiment not only affirmed the theoretical underpinnings but also provided a hands-on understanding of how these operations are executed at the circuit level.

2.2.3 Discussion

The procedure allowed us to systematically explore the ALU's capabilities. By manipulating the control inputs and function-select lines, we successfully executed various arithmetic and logic operations. The recorded outputs provided empirical evidence of the ALU's performance.

3. Conclusion

In conclusion, both experiments provided valuable hands-on experience and insights into the construction and operation of memory circuits and arithmetic logic units. The RAM circuit demonstrated reliable data storage and retrieval, while the ALU showcased its versatility in performing diverse operations. These experiments contribute to a deeper understanding of digital electronics and computer organization, laying a foundation for further exploration in the field.

4. References

- [1] Manual for Digital Electronics and Computer Organization Lab, 2023, Birzeit University.

https://ritaj.birzeit.edu/bzu-msgs/attach/2375682/Digital+lab+manual+2023.pdf