**A close up of a logo

Description automatically generated**

**LAB MANUAL**

**Subject code: CS-404**

**SUBJECT NAME: computer ORGANIZATION**

**&ARCHITECTURE**

**SEMESTER: 4TH SEM**

**Session: JAN – JUNE’ 23**

**Department: CSE**

**LIST OF EXPERIMENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr.No.** | **Name of Experiments** | **Date** | **Sign** |
| 1 | Study of Multiplexer and Demultiplexer. |  |  |
| 2 | Study of Half Adder and Subtractor. |  |  |
| 3 | Study of Full Adder and Subtractor. |  |  |
| 4 | WAP to Add two 8-bit numbers. |  |  |
| 5 | WAP to Subtract two 8-bit numbers. |  |  |
| 6 | WAP to Add two 16-bit numbers. |  |  |
| 7 | WAP to Subtract two 16-bit numbers. |  |  |
| 8 | WAP to Multiply two 8-bit numbers. |  |  |
| 9 | WAP to multiply two 8-bit numbers stored at memory location 2000 and 2001 and stores the result at memory location 2000 and 2001. |  |  |
| 10 | WAP to Add two 8-bit numbers stored at memory location 2000 and 2001 and stores the result at memory location 2000 and 2001. |  |  |

**EXPERIMENT NO. 1**

**Aim: Study of Multiplexer and Demultiplexer.**

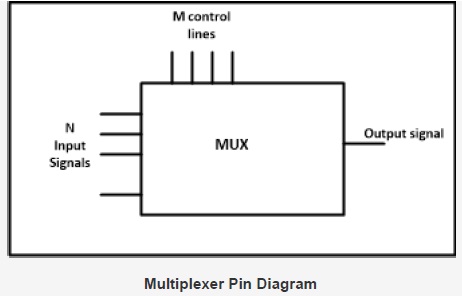
**Solution:**

A multiplexer is a circuit that accepts many input but give only one output. A demultiplexer function exactly in the reverse of a multiplexer, that is a demultiplexer accepts only one input and gives many outputs. Generally multiplexer and demultiplexer are used together, because of the communication systems are bi directional.

Mutliplexer:

Multiplexer means many into one. A multiplexer is a circuit used to select and route any one of the several input signals to a signal output. An simple example of an non electronic circuit of a multiplexer is a single pole multiposition switch.

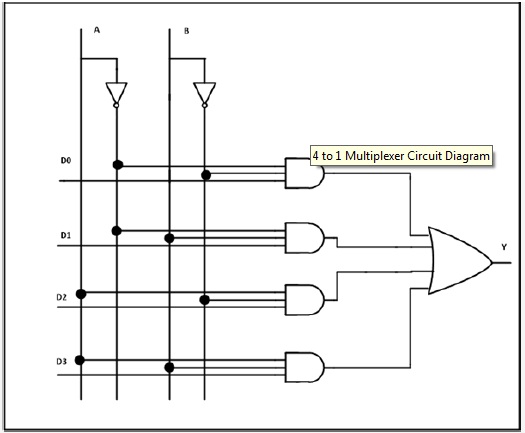
Multiposition switches are widely used in many electronics circuits. However circuits that operate at high speed require the multiplexer to be automatically selected. A mechanical switch cannot perform this task satisfactorily. Therefore, multiplexer used to perform high speed switching are constructed of electronic components. Multiplexer handle two type of data that is analog and digital. For analog application, multiplexer are built of relays and transistor switches. For digital application, they are built from standard logic gates. The multiplexer used for digital applications, also called digital multiplexer, is a circuit with many input but only one output. By applying control signals, we can steer any input to the output. Few types of multiplexer are 2-to-1, 4-to-1, 8-to-1, 16-to-1 multiplexer. Following figure shows the general idea of a multiplexer with n input signal, m control signals and one output signal. Multiplexer Pin Diagram



**Understanding 4-to-1 Multiplexer:**

The 4-to-1 multiplexer has 4 input bit, 2 control bits, and 1 output bit. The four input bits are D0,D1,D2 and D3. only one of this is transmitted to the output y. The output depends on the value of AB which is the control input. The control input determines which of the input data bit is transmitted to the output. For instance, as shown in fig. when AB = 00, the upper AND gate is enabled while all other AND gates are disabled. Therefore, data bit D0 is transmitted to the output, giving Y = Do.

**Multiplexer Circuit Diagram**



If the control input is changed to AB =11, all gates are disabled except the bottom AND gate. In this case, D3 is transmitted to the output and Y = D3.

- An example of 4-to-1 multiplexer is IC 74153 in which the output is same as the input.

- Another example of 4-to-1 multiplexer is 45352 in which the output is the compliment of the input.

- Example of 16-to-1 line multiplexer is IC74150.

**Applications of Multiplexer:**

Multiplexer are used in various fields where multiple data need to be transmitted using a single line. Following are some of the applications of multiplexers –

1. Communication system – Communication system is a set of system that enable communication like transmission system, relay and tributary station, and communication network. The efficiency of communication system can be increased considerably using multiplexer. Multiplexer allow the process of transmitting different type of data such as audio, video at the same time using a single transmission line.

2. Telephone network – In telephone network, multiple audio signals are integrated on a single line for transmission with the help of multiplexers. In this way, multiple audio signals can be isolated and eventually, the desire audio signals reach the intended recipients.

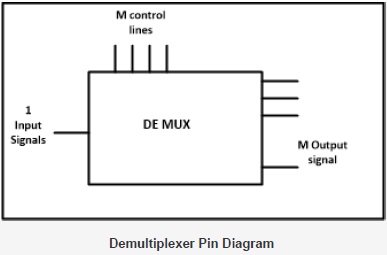
3. Computer memory - Multiplexers are used to implement huge amount of memory into the computer, at the same time reduces the number of copper lines required to connect the memory to other parts of the computer circuit.

4. Transmission from the computer system of a satellite – Multiplexer can be used for the transmission of data signals from the computer system of a satellite or spacecraft to the ground system using the GPS (Global Positioning System) satellites.

**Demultiplexer:**

Demultiplexer means one to many. A demultiplexer is a circuit with one input and many outputs. By applying control signal, we can steer any input to the output. Few types of demultiplexer are 1-to 2, 1-to-4,

1-to-8 and 1-to 16 demultiplexer. Following figure illustrate the general idea of a demultiplexer with 1 input signal, m control signals, and n output signals.



**Understanding 1- to-4 Demultiplexer:**

The 1-to-4 demultiplexer has 1 input bit, 2 control bit, and 4 output bits. An example of 1-to-4 demultiplexer is IC 74155. The 1-to-4 demultiplexer is shown in figure below-

**De-multiplexer**

The input bit is labelled as Data D. This data bit is transmitted to the data bit of the output lines. This depends on the value of AB, the control input.

When AB = 01, the upper second AND gate is enabled while other AND gates are disabled. Therefore, only data bit D is transmitted to the output, giving Y1 = Data.

If D is low, Y1 is low. IF D is high,Y1 is high. The value of Y1 depends upon the value of D. All other outputs are in low state.

If the control input is changed to AB = 10, all the gates are disabled except the third AND gate from the top. Then, D is transmitted only to the Y2 output, and Y2 = Data.

Example of 1-to-16 demultiplexer is IC 74154 it has 1 input bit, 4 control bits and 16 output bit.

**Applications of Demultiplexer:**

1. Demultiplexer is used to connect a single source to multiple destinations. The main application area of demultiplexer is communication system where multiplexer are used. Most of the communication system are bidirectional i.e. they function in both ways (transmitting and receiving signals). Hence, for most of the applications, the multiplexer and demultiplexer work in sync. Demultiplexer are also used for reconstruction of parallel data and ALU circuits.

2. **Communication System** - Communication system use multiplexer to carry multiple data like audio, video and other form of data using a single line for transmission. This process makes the transmission easier. The demultiplexers receive the output signals of the multiplexer and convert them back to the original form of the data at the receiving end. The multiplexer and demultiplexer work together to carry out the process of transmission and reception of data in communication system.

3. **ALU (Arithmetic Logic Unit)** – In an ALU circuit, the output of ALU can be stored in multiple registers or storage units with the help of demultiplexer. The output of ALU is fed as the data input to the demultiplexer. Each output of demultiplexer is connected to multiple register which can be stored in the registers.

4. **Serial to parallel converter** - A serial to parallel converter is used for reconstructing parallel data from incoming serial data stream. In this technique, serial data from the incoming serial data stream is given as

data input to the demultiplexer at the regular intervals. A counter is attach to the control input of the demultiplexer. This counter directs the data signal to the output of the demultiplexer where these data signals are stored. When all data signals have been stored, the output of the demultiplexer can be retrieved and read out in parallel.

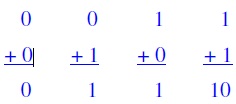
**EXPERIMENT NO. 2**

**Aim- Study of Half Adder and Subtractor.**

**Solution: -**Let's start with a **half (single-bit) adder** where you need to add single bits together and

get the answer. The way you would start designing a circuit for that is to first look at all

of the logical combinations. You might do that by looking at the following four sums:

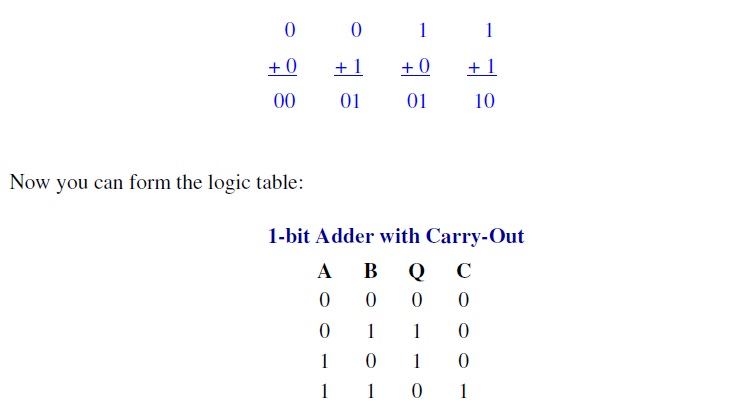


That looks fine until you get to 1 + 1. In that case, you have a **carry bit** to worry about. If

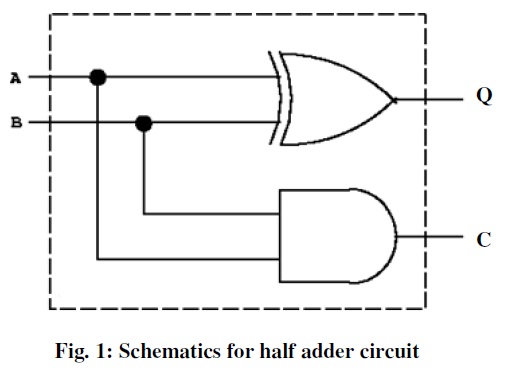
you don't care about carrying (because this is, after all, a 1-bit addition problem), then

you can see that you can solve this problem with an XOR gate. But if you do care, then

you might rewrite your equations to always include **2 bits of output**, like this:



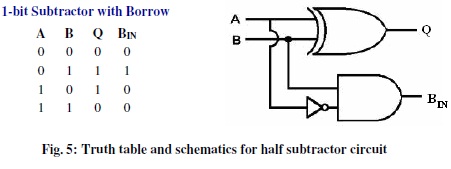
By looking at this table you can see that you can implement the sum Q with an XOR gate and C (carry-out) with an AND gate.



***Subtraction:***

In a similar fashion subtraction can be performed using binary numbers. The truthtable for a single bit or half-subtractor with inputs A and B is given below along with itscircuit diagram (Fig.5). A full substractor circuit accepts a minuend (A) and thesubtrahend (B) and a borrow (BIN) as inputs from a previous circuit.

A full subtractorcircuit can be realized by combining two half substractor circuits and an OR gate as shownin Fig.5.



However, it is possible to use the same circuit to perform addition and subtraction byreplacing the ‘NOT’ gate of the subtractor circuit by an ‘XOR’ as shown in the circuitdiagram below. Here, the second input (first input is from supply) for XOR gate decidesthe function of the circuit, i.e. addition or subtraction. This means if the second input forXOR is 0, the circuit will do addition and if 1, it will do subtraction.

**Circuit components/Equipments:**

1. Resistors (1K\_, 5 Nos)

2. ICs (XOR-7486, AND-7408, OR-7432, NOT-7404, )

3. A Surface mount dip switch

4. D.C. Power supply (5V)

5. Red/Green LEDs (2 Nos)

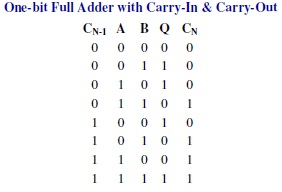
6. Connecting wires

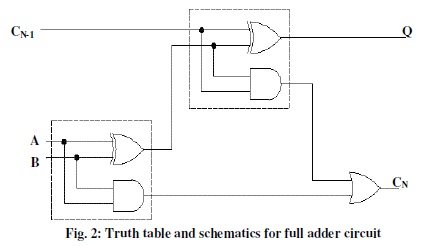
7. Breadboard

**EXPERIMENT NO. 3**

**Aim- Study of Full Adder and Subs-tractor.**

**Solution-**If you want to add two or more bits together it becomes slightly harder. In this case, weneed to create a full adder circuits. The difference between a full adder and a half adderwe looked at is that a full adder accepts inputs A and B plus a **carry-in** (CN-1) givingoutputs Q and CN. Once we have a full adder, then we can string eight of them together tocreate a byte-wide adder and cascade the carry bit from one adder to the next. The logictable for a full adder is slightly more complicated than the tables we have used before,because now we have **3 input bits**. The truth table and the circuit diagram for a full-adderis shown in Fig. 2. If you look at the Q bit, it is 1 if an odd number of the three inputs isone, i.e., Q is the XOR of the three inputs. The full adder can be realized as shown below.Notice that the full adder can be constructed from two half adders and an ***OR*** gate.





Now we have a piece of functionality called a"full adder", which can be combined with a halfadder to construct a 2-bit adder. Adders forarbitrarily large (say N-bit) binary numbers canbe constructed by cascading full adders. Theseare called a **ripple-carry** adder, since the carrybit “ripples” from one stage to the next. Theschematics for a 4-bit full adder circuit is shownbelow. This implementation has the advantage ofsimplicity but the disadvantage of speedproblems. In a real circuit, gates take time toswitch states (the time is of the order ofnanoseconds, but in high-speed computersnanoseconds matter). So 32-bit

or 64-bit ripplecarryadders might take 100 to 200 nanosecondsto settle into their final sum because of carry ripple.

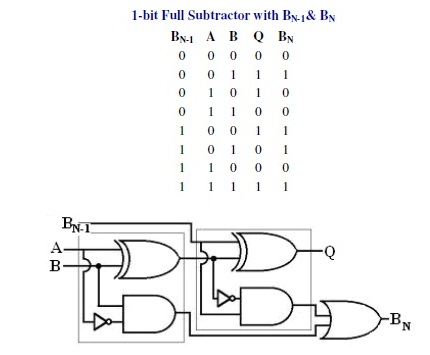
****

**Fig. 3: Schematics of 2-bit adder**



**Fig.4: Schematics of 4-bit adder**

A full subtractor circuit can be realized by combining two half subtractor circuits and an OR gate as shown in Fig. 6.



**Fig. 6: Truth table and schematics for full subtractor circuit**

**EXPERIMENT NO. 4**

**Aim- WAP to addition of two 8-bit number.**

**Solution-**

**Problem Analysis:**

1. **Load the numbers in registers.**
2. **Add the contents of registers.**
3. **Display the Sum on output device.**

**Flow Chart:**

LOAD TWO 8 BIT HEX NUMBERS IN TWO REGISTERS

ADD THE CONTENTS OF REGISTERS

DISPLAY THE SUM

**Assembly Language Program:**

2000 MOV AL,03

2001 MOV BL,03

2002 ADD AL, BL

2003 HLT

**OUTPUT:**

AX: FB06

**EXPERIMENT NO. 5**

**Aim- WAP to subtraction of two 8-bit number.**

**Solution-**

**Problem Analysis:**

1. **Load the numbers in registers.**
2. **Subtract the contents of registers.**
3. **Display the Subtraction on output device.**

**Flow Chart:**

LOAD TWO 8 BIT HEX NUMBERS IN TWO REGISTERS

SUBTRACT THE CONTENTS OF REGISTERS

DISPLAY THE SUBTRACTION

**Assembly Language Program:**

2000 MOV AL, 06

2001 MOV BL, 03

2002 SUB AL, BL

2003 HLT

**OUTPUT:**

AX: FB03

**EXPERIMENT NO. 6**

**Aim- WAP a program to addition of two 16-bit number.**

**Solution**-

**Problem Analysis:**

1. **Load the numbers in registers.**
2. **Add the contents of registers.**
3. **Display the Sum on output device.**

**Flow Chart:**

LOAD TWO 16 BIT HEX NUMBERS IN TWO REGISTERS

ADD THE CONTENTS OF REGISTERS

DISPLAY THE SUM

**Assembly Language Program:**

2000 MOV AX,1234

2001 MOV BX,1234

2002 ADD AX, BX

2003 HLT

**OUTPUT:**

AX: 2468

**EXPERIMENT NO. 7**

**Aim- WAP a program to subtraction of two 16-bit numbers.**

**Solution**-

**Problem Analysis:**

1. **Load the numbers in registers.**
2. **Subtract the contents of registers.**
3. **Display the Subtraction on output device.**

**Flow Chart:**

LOAD TWO 16 BIT HEX NUMBERS IN TWO REGISTERS

SUBTRACT THE CONTENTS OF REGISTERS

DISPLAY THE SUBTRACTION

**Assembly Language Program:**

2000 MOV AX,2468

2001 MOV BX,1234

2002 SUB AX, BX

2003 HLT

**OUTPUT:**

AX: 1234

**EXPERIMENT NO. 8**

**Aim- WAP to multiplication of two 8-bit number.**

**Solution-**

**Problem Analysis:**

1. **Load the numbers in registers.**
2. **Multiply the contents of registers.**
3. **Display the result on output device.**

**Flow Chart:**

LOAD TWO 8 BIT HEX NUMBERS IN TWO REGISTERS

MULTIPLY THE CONTENTS OF REGISTERS

DISPLAY THE RESULT

**Assembly Language Program:**

2000 MOV AL,03

2001 MOV BL,03

2002 MUL BL

2003 HLT

**OUTPUT:**

AX: FB09

**EXPERIMENT NO. 9**

**Aim- WAP a program to multiplication of two numbers and save that result on location 2000.**

**Solution**-

**Problem Analysis:**

1. **Load the numbers at given memory locations.**
2. **Move the numbers from given memory locations to registers.**
3. **Multiply the contents of registers.**
4. **Move the result at given memory location.**

**Flow Chart:**

LOAD TWO 8 BIT HEX NUMBERS AT GIVEN MEMORY LOCATION

MOVE THE DATA TO THE REGISTERS

MULTIPLY THE CONTENTS OF REGISTERS

STORE THE RESULT AT MEMORY LOCATION 2000

**Assembly Language Program:**

2000 MOV AL, [2000]

2001 MOV BL, [2001]

2002 MUL BL

2003 MOV [2000], AL

2004 HLT

**OUTPUT:**

TO FILL THE DATA ON GIVEN MEMORY LOCATIONS ( 2000 AND 2001)

PRESS F7

PRESS F

AT ADDRESS 2000 STORE THE FIRST DATA VALUE ( 03)

AT ADDRESS 2001 STORE THE SECOND DATA VALUE (02)

TO DISPLAY THE RESULT

PRESS F7

PRESS D

2000 01**06**

**EXPERIMENT NO. 10**

**Aim- WAP a program to addition of two numbers and save that result on location 2000.**

**Solution**-

**Problem Analysis:**

1. **Load the numbers at given memory locations.**
2. **Move the numbers from given memory locations to registers.**
3. **Multiply the contents of registers.**
4. **Move the result at given memory location.**

**Flow Chart:**

LOAD TWO 8 BIT HEX NUMBERS AT GIVEN MEMORY LOCATION

MOVE THE DATA TO THE REGISTERS

ADD THE CONTENTS OF REGISTERS

STORE THE RESULT AT MEMORY LOCATION 2000

**Assembly Language Program:**

2000 MOV AL, [2000]

2001 MOV BL, [2001]

2002 ADD AL,BL

2003 MOV [2000], AL

2004 HLT

**OUTPUT:**

TO FILL THE DATA ON GIVEN MEMORY LOCATIONS ( 2000 AND 2001)

PRESS F7

PRESS F

AT ADDRESS 2000 STORE THE FIRST DATA VALUE ( 03)

AT ADDRESS 2001 STORE THE SECOND DATA VALUE (02)

TO DISPLAY THE RESULT

PRESS F7

PRESS D

2000 01**05**