**Part 2:**

**Theory:**

**-ALU:**

Arithmetic Logic Unite ( ALU ) is one components of the CPU that perform arithmatic and logic operation . The arithmatic or logic output is selected by a multiplexer that choice eithor arithmatic or logic unite .

**-Parity Generater:**

Parity generater offers a small amount of error checking to help detect data corruption that might occur during transmission. There are two type of parity even and odd parity. Even parity will produce 1 on the output if the data word contains an odd number of ones else the output will be 0. Odd parity is the opposite to even parity.

**Proceduer:**

**ALU:**



Fig 1: ALU

After we connected the circuit in fig 1 , we put it on logic operation by set M = 1.

(1) When S3S2S1S0= 0000, A3A2A1A0=0000 and B3B2B1B0=1111, the outputs F3F2F1F0= 1111 .

(2) When S3S2S1S0= 0000, A3A2A1A0=1100 and B3B2B1B0=1010, the outputs F3F2F1F0= 0011 .

\*\*The relationship between the input and output was F = A' .

(3) When S3S2S1S0=1001, A3A2A1A0=1100 and B3B2B1B0=0110, the outputs F3F2F1 F0= 0101 .

\*\*The relationship between the input and output was F = A XNOR B .

(4) When S3S2S1S0=1011, A3A2A1A0=0011 and B3B2B1B0=1001, the outputs F3F2F1 F0= 0001 .

\*\*The relationship between the input and output was F = A. B .

We set M to "0", Cn to "0". To perform the following arithmetic functions:

(1) When S3S2S1S0=1001, A3A2A1A0=B3B2B1B0=0100, the outputs

F3F2F1 F0 = 1001 and Cn+4 = 1 .

(2) When S3S2S1S0=1001, A3A2A1A0=1000 and B3B2B1B0=1100, the outputs F3F2F1F0 = 0101 and Cn+4 = 0 .

\*\*The relationship between the input and output was F = A+B+1 .

 (3) When S3S2S1S0=0011, A3A2A1A0=0100 and B3B2B1B0=0010, the outputs F3F2F1 F0 = 0000 .and Cn+4 = 0 .

(4) When S3S2S1S0=0011, A3A2A1A0=1010 and B3B2B1B0=1000, the outputs F3F2F1 F0 = 0000 .and Cn+4 = 0 .

\*\*The relationship between the input and output was F = ZERO .

(5) When S3S2S1S0=0000, A3A2A1A0=1010 and B3B2B1B0=0011 , the outputs F3F2F1 F0 = 1011 . and Cn+4 = 1 .

\*\*The relationship between the input and output was F = A+1.

After that we set M= "0" and Cn = "1" . To follow the input sequence in table 1.

|  |  |  |
| --- | --- | --- |
| FUNCTION | OUTPUT | INPUT |
|  | C4 F3F2F1F0 | A3A2A1A0 | B3B2B1B0 | S3S2S1S0 |
| A |  1 0001 | 0001 | 0001 | 0000 |
| A |  1 0011 | 0011 | 0010 | 0000 |
| -1 |  1 1111 | 0010 | 0001 | 0011 |
| -1 |  1 1111 | 0110 | 0100 | 0011 |
| A-B-1 |  1 1110 | 0100 | 0101 | 0110 |
| A-B-1 |  0 0000 | 0010 | 0001 | 0110 |
|  (A.B')' |  0 0000 | 0101 | 0110 | 0111 |
| (A.B')' |  0 0111 | 1000 | 0111 | 0111 |
| A.B-1 |  0 0111 | 1000 | 1000 | 1011 |
| A.B-1 |  0 0011 | 0110 | 1100 | 1011 |
| A-1 |  0 0110 | 0111 | 0110 | 1111 |
| A-1 |  0 0101 | 0110 | 0101 | 1111 |

Table 1

**Parity Generater:**

The circuit in fig 2 was connected and the output was as shown in table 2.



Fig 2: parity generater using XOR gates

|  |  |
| --- | --- |
| OUTPUT | INPUT |
| F | E D C B A |
| 0 |  0 0 0 0 0 |
| 1 |  0 0 0 1 0 |
| 0 |  0 0 0 1 1 |
| 1 |  0 0 1 0 0 |
| 0 |  0 0 1 0 1 |
| 1 |  0 1 1 1 0 |
| 0 |  1 1 0 0 0 |
| 1 |  1 1 0 1 0 |
| 0 |  1 1 1 1 0 |
| 1 |  1 1 1 1 1 |

Table 2

From result in table 2 we could noticed that parity generater is even type.

When we used the IC chips in fig 3 to had the parity generater we had the output in table 3.



Fig 3 : parity generater using IC chip

|  |  |
| --- | --- |
| OUTPUT | INPUT |
| Y1(odd) Y2(even) | L H G F E D C B A |
|  1 0 |  0 0 0 0 0 0 0 0 0 |
|  0 1 |  0 0 0 0 0 0 0 0 1 |
|  1 0 |  0 0 0 0 0 0 0 1 1 |
|  0 1 |  0 0 0 0 0 0 1 1 1 |
|  1 0 |  0 0 0 0 0 1 1 1 1 |
|  0 1 |  0 0 0 0 1 1 1 1 1 |
|  1 0 |  0 0 0 1 1 1 1 1 1 |
|  0 1 |  0 0 1 1 1 1 1 1 1 |
|  1 0 |  0 1 1 1 1 1 1 1 1 |
|  0 1 |  1 1 1 1 1 1 1 1 1 |
|  1 0 |  1 1 1 1 1 1 1 0 1 |
|  0 1 |  1 1 1 1 1 1 1 0 0 |
|  1 0 |  1 1 0 0 0 1 1 0 0 |

Table 3

**Conclusion:**

At the end of these experiment we know the the two component of the ALU which are logic and mathmaticl unites , and how we could change betweeen them . Also we know how parity generater worke in detect the error in data transmition , and its type . In addition we learned how we could built parity generater using XOR gates.