**Digital Logic Design Homework**

A (A1 A0)

B(B1 B0)

A= 0 0

B= 0 0

F1 (A=B)─> 1

A=11

B= 0 0

F2(A>B) ─> 1

A= 0 0 ,B=11

F3(B>A) ─> 1

**1.**

A 2-bit comparator compares two binary numbers, each of two bits and produces their relation such as one number is equal or greater than or less than the other. The figure below shows the block diagram of a two-bit comparator which has four inputs and three outputs.

The first number A is designated as A = A1A0 and the second number is designated as B = B1B0. This comparator produces three outputs as F2= ( A>B), F1= (A = B) and F3= (A<B).

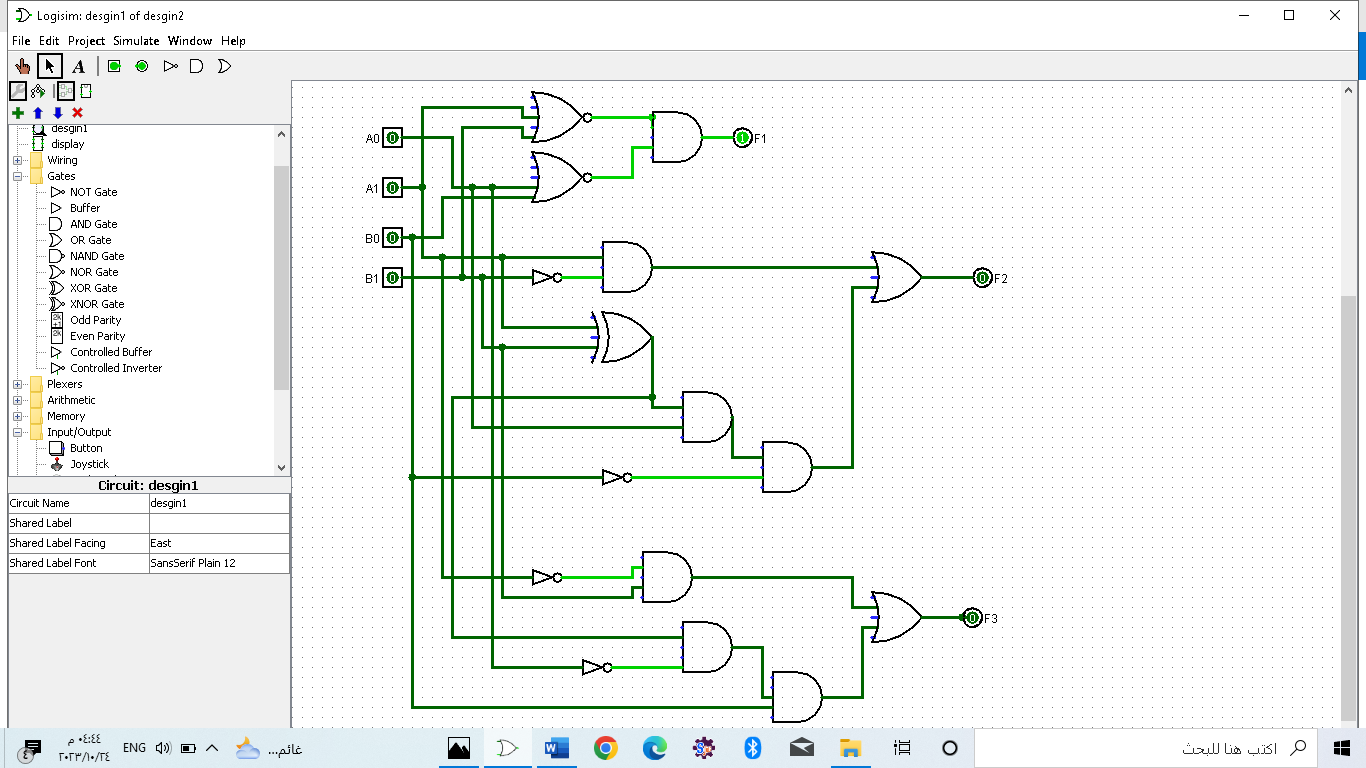
**Truth table**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Input | | | | Output | | | |
| A1 | A0 | B1 | B0 | | F1(A=B) | F2(A>B) | F3(A<B) |
| 0 | 0 | 0 | 0 | | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | | 1 | 0 | 0 |

K-map



circuits implementation on Logisim



**2.**

A 7-segment display consists of an arrangement of LEDs in an )H( form. A truth table is constructed with the combination of inputs for each decimal number. For example, decimal number 1 would command a combination of b and c .

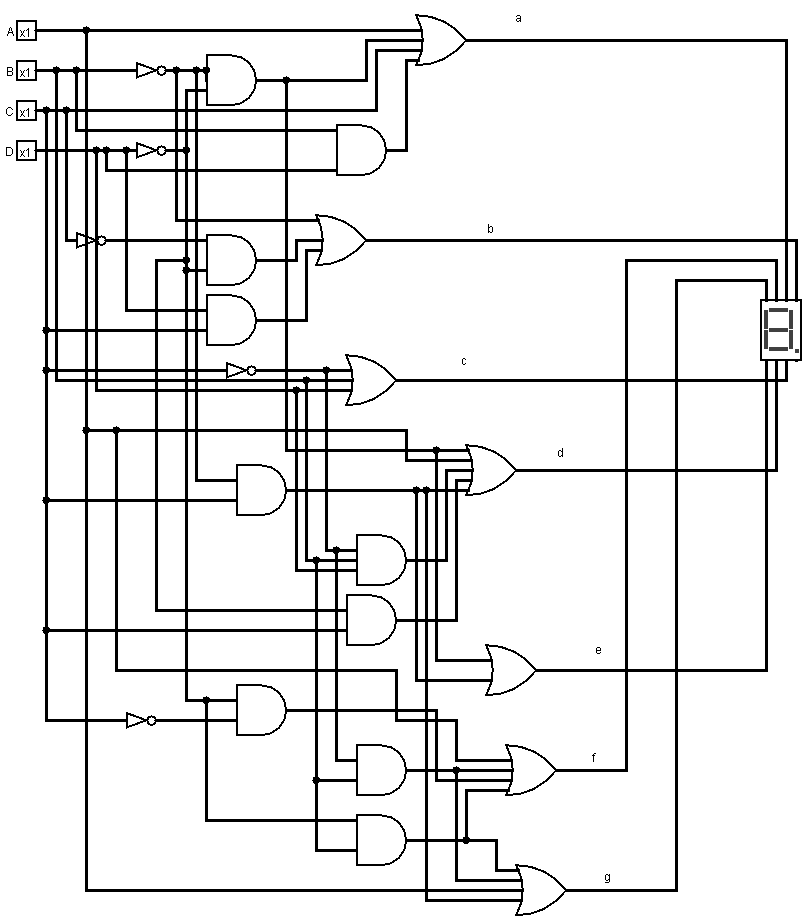
Truth table

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

K-map





circuits implementation on Logisim

