2 Logic design for 4-bit comparator

2.1 logic design procedure

Magnitude comparator is a combinational circuit that compares two numbers and determines their relative magnitude. A comparator is shown as Figure 2.1. The output of comparator is usually 3 binary variables indicating:

A>B
A=B
A<B

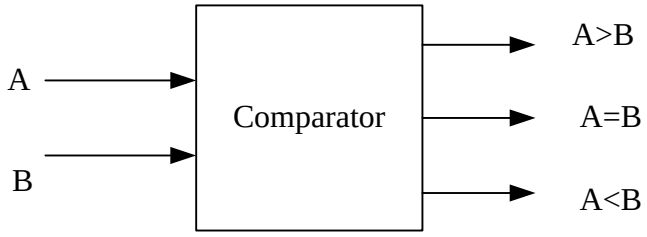


Figure 2.1 1-bit comparator

For a 2-bit comparator (Figure 2.2), we have four inputs A1A0 and B1B0 and three outputs:
E (is 1 if two numbers are equal)
G (is 1 when A > B) and
L (is 1 when A < B)

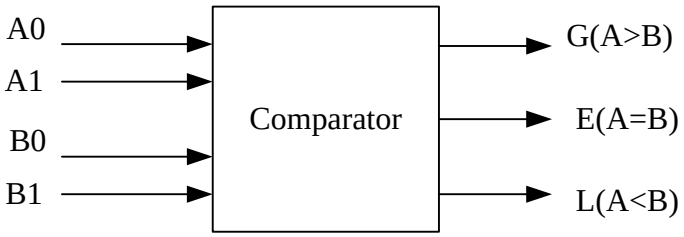


Figure 2.2 2-bit comparator

If we use truth table and K-MAP, the result is
E= A'1A'0B'1B'0 + A'1A0B'1B0 + A1A0B1B0 + A1A'0B1B'0
or E=((A0 ⊕ B0) + (A1 ⊕ B1))'
G = A1B'1 + A0B'1B'0 + A1A0B'0
L= A'1B1 + A'1A'0B0 + A'0B1B0

Here we use simpler method to find E (called X) and G (called Y) and L (called Z)
(1) A=B if all Ai= Bi

Table 2.1

Ai	Bi	Xi
0	0	1
0	1	0
1	0	0
1	1	0

It means X0 = A0B0 + A'0B'0 and
X1= A1B1 + A'1B'1

If X0=1 and X1=1 then A0=B0 and A1=B1
Thus, if A=B then X0X1 = 1 it means
X= (A0B0 + A'0B'0)(A1B1 + A'1B'1)
since $(x \oplus y)' = (xy + x'y')$
$X= (A0 \oplus B0)' (A1 \oplus B1)' = ((A0 \oplus B0) + (A1 \oplus B1))'$
It means for X we can NOR the result of two exclusive-OR gates.

(2) A>B means

Table 2.2

A1	B1	Y1
0	0	0
0	1	0
1	0	1
1	1	0

If A1=B1 (X1=1) then A0 should be 1 and B0 should be 0

Table 2.3

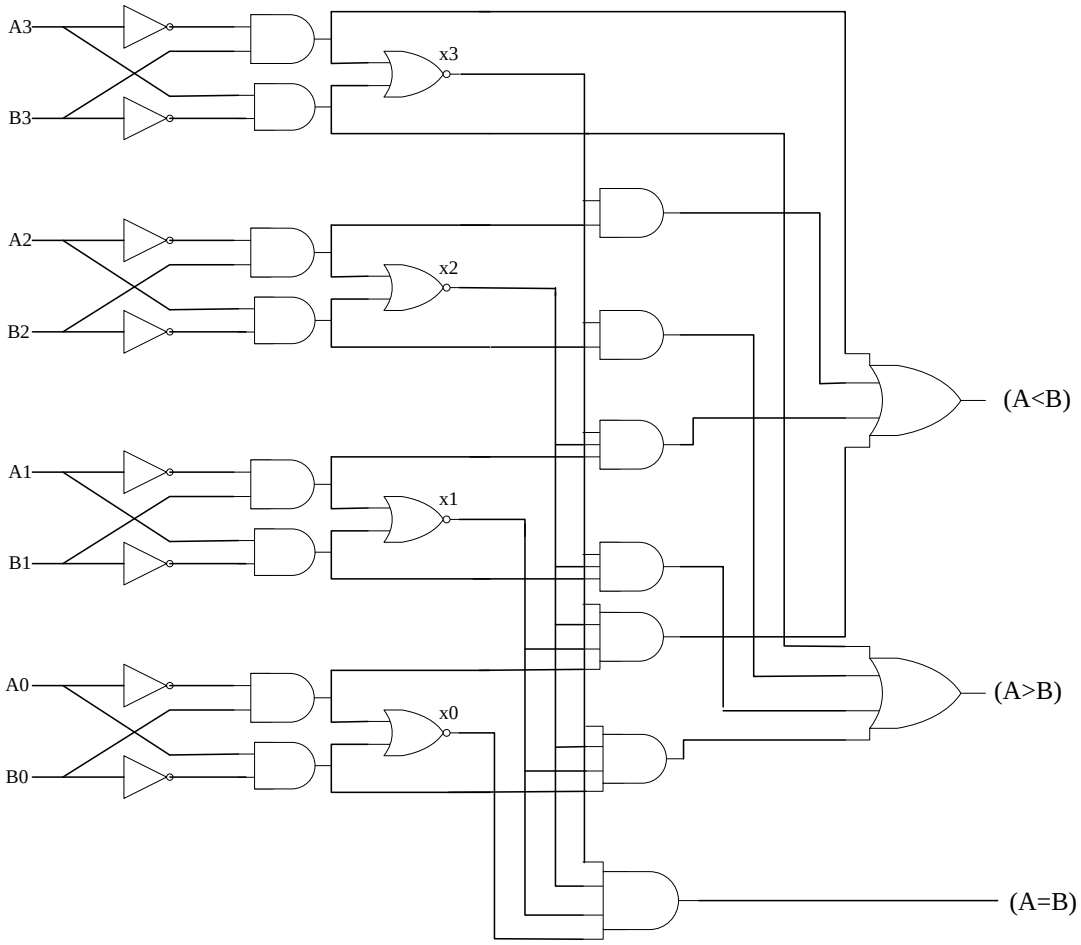
A0	B0	Y0
0	0	1
0	1	0
1	0	0
1	1	0

For A> B: A1 > B1 or
A1 =B1 and A0 > B0
It means Y= A1B'1 + X1A0B'0 should be 1 for A> B.

(3)For B>A: B1 > A1 or
A1=B1 and B0> A0
Z= A'1B1 + X1A'0B0

2.2 4-Bit Comparator

The procedure for binary numbers with more than 2 bits can also be found in the similar way. Figure 2.3 shows the 4-bit magnitude comparator.
Input A=A3A2A1A0;
B=B3B2B1B0



4-Bit Magnitude Comparator

Figure 2. 3 4- bit Magnitude Comparator

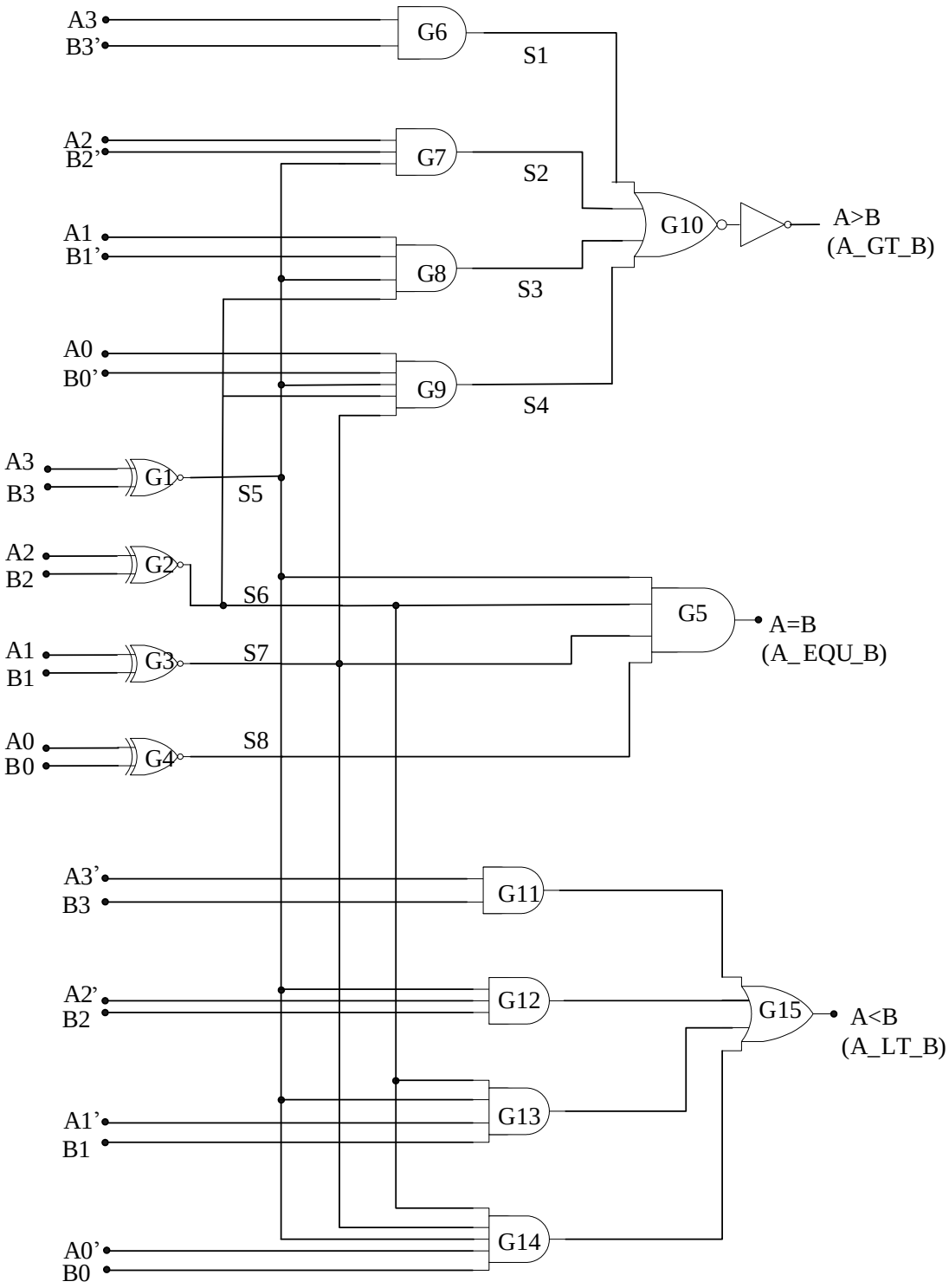
(1)A= B : A3=B3, A2=B2, A1=B1, A0=B0
xi = AiBi + Ai'Bi'
XOR-Invert = (AiBi'+Ai'Bi)'
= (Ai'+Bi)(Ai+Bi')
= Ai'Ai + Ai'Bi' + AiBi + BiBi'
= AiBi + Ai'Bi'
Output: x3x2x1x0

(2)A> B
Output: A3B’3 + x3A2B’2 + x3x2A1B’1+ x3x2x1A0B’0

(3)(A< B)
Output: A’3B3 + x3A’2B2 + x3x2A’1B1+ x3x2x1A’0B0

Table 2.4 Truth table of 4-Bit Comparator

COMPARING INPUTS				OUTPUT		
A3, B3	**A2, B2**	**A1, B1**	**A0, B0**	**A > B**	**A < B**	**A = B**
A3 > B3	X	X	X	H	L	L
A3 < B3	X	X	X	L	H	L
A3 = B3	A2 >B2	X	X	H	L	L
A3 = B3	A2 < B2	X	X	L	H	L
A3 = B3	A2 = B2	A1 > B1	X	H	L	L
A3 = B3	A2 = B2	A1 < B1	X	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 > B0	H	L	L
A3 = B3	A2 = B2	A1 = B1	A0 < B0	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	H	L	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	L	H
H = High Voltage Level, L = Low Voltage, Level, X = Don’t Care						



Logic diagram for a 4-bit comparator

Seeing from the above diagram, we can use 11 gates to implement the 4-Bit comparator beside the inverters. The kind of gates includes XOR, AND, NOR. 4 gates of XOR are the same. 5 gates of AND have different number of inputs, but the principle of layout is the same. So does the NOR gate.