PLS173 as a 10-bit comparator, 74LS460 AN024

DESCRIPTION

The PLS173 is a 24-pin PLA device which has 10 bidirectional outputs and 12 dedicated inputs. The output of the device is the sum of products of the inputs. The polarity of each output may be individually programmed as Active-High or Active-Low. A 10-bit comparator similar to the 74LS460 compares two 10-bit data inputs to establish if EQUIVALENCE or NOT EQUIVALENCE exists. The output has True and Complement comparison status outputs. The logic diagram of the comparator is shown in Figure 1.

The truth table is as shown in Table 1 where vectors a and b are 10-bit inputs to A9 to A0 and B9 to B0. If the input to A9-A0 is bit-to-bit equivalent to the input to B9-B0, the two input vectors are considered EQUIVALENT, and output EQ goes High and NE goes Low. If the two input vectors are not bit-to-bit equivalent, then EQ goes Low and NE goes High. The circuit is implemented with SNAP as shown in Figure 3.

Table 1. Function Table

$A_9 - A_0$	$B_9 - B_0$	EQ	NE
a	a	H	L
b	b	H	L
a	b	L	H
b	a	L	H

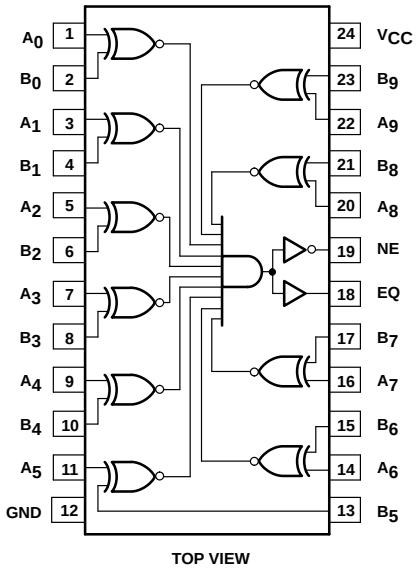

Figure 1. Logical Equivalent Circuit of 10-Bit Comparator

RESOURCES

This design used 20 product terms in the PLS173. As shown in Figure 4, expanded equations, each output needs 20 product terms. Since the product terms are the same, each output shares the 20 product terms with an output polarity fuse determining the proper output level.

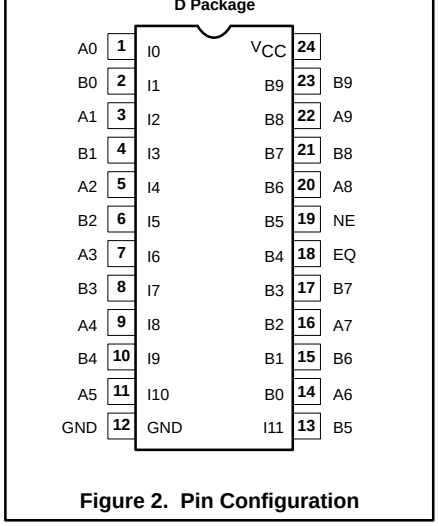

Figure 2. Pin Configuration

PLS173 as a 10-bit comparator, 74LS460 AN024

This circuit compares to two 10-bit inputs. If they are bit-to-bit equivalent, output EQ goes high and NE goes low. If they are not bit-to-bit equivalent, output EQ will be low while NE will be high.

```
@PINLIST
A[0..9]     i;
B[0..9]     i;
EQ          o;
NE          o;

@GROUPS
one         =   a[0..9];
two         =   b[0..9];

@TRUTHTABLE
@LOGIC EQUATIONS

EQ = one == two;
NE = one != two;

@INPUT VECTORS
@OUTPUT VECTORS
@STATE VECTORS
@TRANSITIONS
```

These equations are written for Philips Semiconductors SNAP software.

Figure 3. Boolean Equations of 10-Bit Comparator

```
@LOGIC EQUATIONS
eq          =   / ( ( /b9 * a9 )
            +   ( b9 * a9 )
            +   ( b8 * / a8 )
            +   ( / b8 * a8 )
            +   ( / b7 * a7 )
            +   ( b7 * / a7 )
            +   ( b6 * / a6 )
            +   ( / b6 * a6 )
            =   ( b5 * / a5 )
            +   ( b5 * / a5 )
            +   ( / b5 * a5 )
            +   ( / b4 * a4 )
            +   ( b4 * a4 )
            +   ( / b3 * a3 )
            +   ( b3 * / a3 )
            +   ( / b2 * a2 )
            +   ( b2 * / a2 )
            +   ( / b1 * a1 )
            +   ( b1 * / a1 )
            +   ( b0 * / a0 )
            +   ( / b0 *  a0 ) );


ne          =    ( ( /b9 * a9 )
            +   ( b9 * a9 )
            +   ( b8 * / a8 )
            +   ( / b8 * a8 )
            +   ( / b7 * a7 )
            +   ( b7 * / a7 )
            +   ( b6 * / a6 )
            +   ( / b6 * a6 )
            =   ( b5 * / a5 )
            +   ( b5 * / a5 )
            +   ( / b5 * a5 )
            +   ( / b4 * a4 )
            +   ( b4 * a4 )
            +   ( / b3 * a3 )
            +   ( b3 * / a3 )
            +   ( / b2 * a2 )
            +   ( b2 * / a2 )
            +   ( / b1 * a1 )
            +   ( b1 * / a1 )
            +   ( b0 * / a0 )
            +   ( / b0 *  a0 ) );
```

Figure 4. Expanded Equations