

74HC573 锁存器的功能

74HC573 和 74LS373 原理一样，8 数据锁存器。主要用于数码管、按键等等的控制。74HC573 有 20 个脚，数据的进和出没有逻辑关系，这个芯片主要是看高电压激活还是低电压激活：

1 是低电压激活芯片

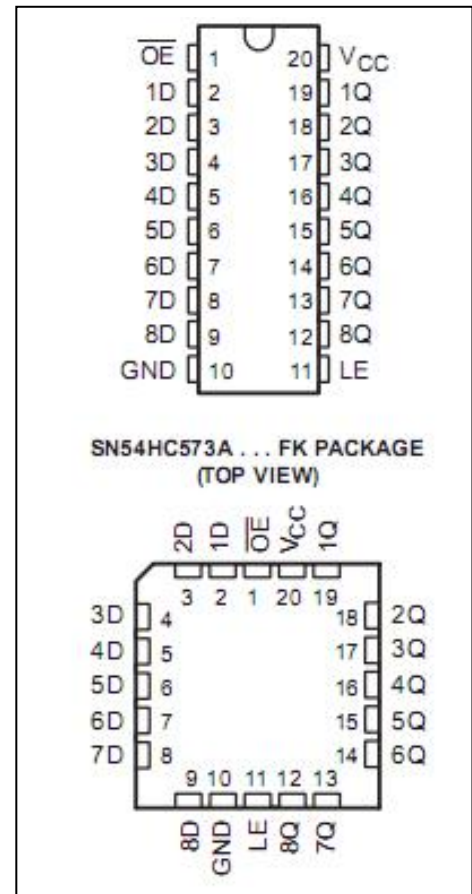
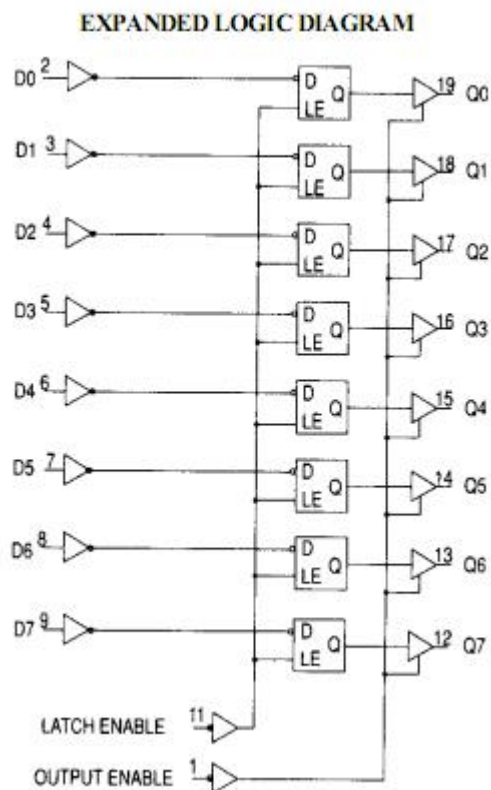
2~9 脚是数据的输入脚从 D0 到 D7

10 脚是接地

11 脚是高电压激活芯片

12~19 脚是数据的输出脚

20 是电源



1.真值表

功能表:

输入			输出
输出使能	锁存使能	D	Q
L	H	H	H
L	H	L	L
L	L	X	不变
H	X	X	Z

X=不用关心

Z=高阻抗

74HC573 真值表,意思如下:

- 第一行/第二行: 当 $OE=0$ 、 $LE=1$ 时, 输出端数据等于输入端数据;
- 第三行: 当 $OE=0$ 、 $LE=0$ 时, 输出端保持不变;
- 第四行: 当 $OE=1$ 是无论 D_n 、 LE 为何, 输出端为高阻态;

2. 高阻态

就是输出既不是高电平, 也不是低电平, 而是高阻抗的状态; 在这种状态下, 可以多个芯片并联输出; 但是, 这些芯片中只能有一个处于非高阻态状态, 否则会将芯片烧毁。高阻态的概念在 RS232 和 RS422 通讯中还可以用到。

3. 数据锁存

当输入的数据消失时, 在芯片的输出端, 数据仍然保持; 这个概念在并行数据扩展中经常使用到。

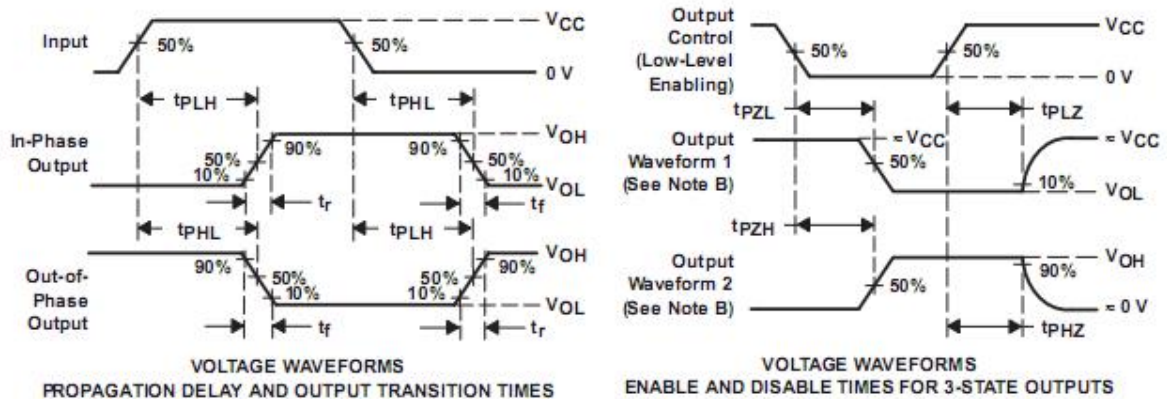
4. 数据缓冲

加强驱动能力: 74LS244/74LS245/74LS373/74LS573 都具备数据缓冲的能力。

- ◆ OE: output_enable, 输出使能;
- ◆ LE: latch_enable, 数据锁存使能, latch 是锁存的意思;
- ◆ D_n : 第 n 路输入数据;
- ◆ O_n : 第 n 路输出数据;

74HC573 波形图,在实际应用的时候是这样做的:

- OE=0;
- 先将数据从单片机的口线上输出到 Dn;
- 再将 LE 从 0->1->0 ;
- 这时, 你所需要输出的数据就锁存在 On 上了, 输入的数据在变化也影响不到输出的数据了; 实际上, 单片机现在在忙着干别的事情, 串行通信、扫描键盘.....单片机的资源有限啊。



在单片机按照 RAM 方式进行并行数据的扩展时,使用 `movx @dptr, A` 这条指令时, 这些时序是由单片机来实现的。

后面的表格中还需要时间的参数, 你不需要去管它, 因为这些参数都是几十 ns 级别的, 对于单片机在 12M 下的每个指令周期最小是 1us 的情况下, 完全可以实现; 如果是你自己来实现这个逻辑, 类似的指令如下:

```
MOV  P0,A ;//将数据输出到并行数据端口
CLR  LE
SETB LE
CLR  LE ;//上面三条指令完成 LE 的波形从 0->1->0 的变化
```

74LS573 跟 74LS373 逻辑上完全一样, 只不过是管脚定义不一样, 数据输入和输出端.