

关于 ASK/OOK 调制的无线信号解码经验

1 这种常规的 315、433、868 的无线调制信号，发射端一般用 2262/1527/PT2240/HCS301，如果选择 2262，接收端则直接搭配好 2272，匹配好振荡电阻和烙好 A0-A7 的脚就可以直接配套使用了，那么以下步骤你可以忽略了。

2 发射用 1527.PT2240.hcs301 等芯片，这些大多都是各应用者自己编写解码程序（2262 也可以用单片机来解，只是如果用途单一，用 2272 更易上手且价更低），单片机解码，要不断的处理 100us--20ms 的脉冲识别，还要干别的任务，比如按键 LED 数码扫描，GSM 拨号等等，所以此处与开发者经验息息相关，通过阅读本文，迅速提升你的这方面的无线解码经验，也是本文的目的之所在。

3 如果单片机的任务单一，可以在 main 中，去循环查询端口的方式来不断的判断脉冲宽度，一般工程师都是用这种方式入手的，这种方式仅限 EMC153 这类低档 CPU，如果是 51 或 STM，我绝不推荐，原因是，你也要心痛下 CPU，不断地要去判断，它累一点到还是次要的，最主要的是单片机要有其它任务，你就会丢掉无线接收，比如要去拨打 GSM 电话，或传输一次 WIFI 数据到云端，这些耗时较长，会直接造成无线接收的丢码。一句话，这种解码方式，入门尝可，仅限入门用，此方法因不太推荐，所以至此也就不再具体介绍，反正就是来回看引脚波形的时间，在不在解码有效时间范围内，不再深入。

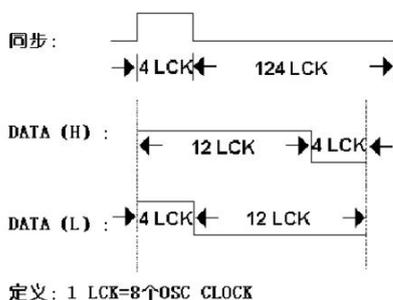
4 我推荐的解码方法是一个中断引脚加一个定时器，在中断里完成这部分的无线解码，虽然模块在静态时，它有着 30us—1000us 不等杂波，占用 CPU 太多的时间，我凭经验告诉你，完全可以忽视，写好你的中断代码就好了，中断以外的时间处理其它任务完全没有问题。比如串口收发，SPI 通讯等，处理串口中断，和其它定时中断也基本不影响，如果串口波特率较高，实测有误码时，可以将串口中断的优先级别高于无线接收。

5 下面讨论具体解码方法，一般选择单片机具有上升沿/下降沿,双边沿中断的引脚来直接连接无线接收模块（老式 51 单片机仅有下降沿方式，后面讲此改进方法），注意不要灌电给接收模块，就是单片机 5V，模块选的 3V，这样会改变接收模块的电压，必做处理，一般此脚设为上拉输入，但更建议将引脚设为高阻输入，这样减小接收模块的负载，有利其减小电压纹波。

6 下面重点来了，如何解码？我们选观察下以下编码图，以 1527 编码为例说明



上面中间部分是一帧完整的数据，最有特征的就是最宽的部分，俗称同步脉冲，两个同步脉冲间的小脉冲是我们解的编码。你可以数一下，两个同步头间的高电平是 25bit，其中最后 1bit 是下一同步头的。我们只解其中的 24bit 即可



上面的 CLK 是指振荡电阻的振荡周期，我们可以忽视，直接用示波器量下 4CLK 的宽度就好了，不用对照 DATASHEET 去找这个 CLK 到底是多少。你仔细观察上面的图，它的 H 和 L 周期是一致的，只是颠倒了位置，所以为了提高解码效率，我们虽是双边沿中断触发，但只检查低电平的持续时间是多久就行了，高电平直接无视，请看下流程图。上升沿时判断定时器的脉冲宽度，**值得提醒，1527 这类编码必须要接收正确的两次信号才可以向外输出**，证明数据收到了，这是许多新入门者常犯的错误，因为无线的乱码有可能会形成这种巧合的接收，请务必校验两次，否则你的产品批量出货后会有大麻烦。HCS301 的编码只需接收一次就行了，它一包中 64bit 位中自带校验码，不需要我们这种重复两次的方法来校验。Chkbuf 中的数据最好也加个时间限制，超时后，校验位为清 0，下次最收到码时，又成为了第一包数据了。数据收到后，交由主程序中查询去处理，有可能还要有 E2 中的数据再次比较，证明是否有无对码等的操作。

