

ADF7020、ADF7020-1、ADF7021和ADF7025中的CDR操作

作者: Austin Harney和Philip Quinlan

简介

ADF7020、ADF7020-1、ADF7021及ADF7025中的时钟和数据恢复(CDR)模块通过一个工作于32倍发送数据速率的过采样数字锁相环(DPLL)实现。CDR PLL将接收到的位流重新同步至一个本地时钟，即“Rx clock”。

DPLL中的鉴相器通过比较恢复位流中的位跃迁时间与一个本地时钟上升沿的时间差来测量相位误差。一个数字控制振荡器(NCO)用来产生恢复时钟。当在后解调器的输出端检测到一个位跃迁时，NCO输出端的相位将被调节一位时间的+1/32、0或-1/32。

CDR简化框图如图1所示。

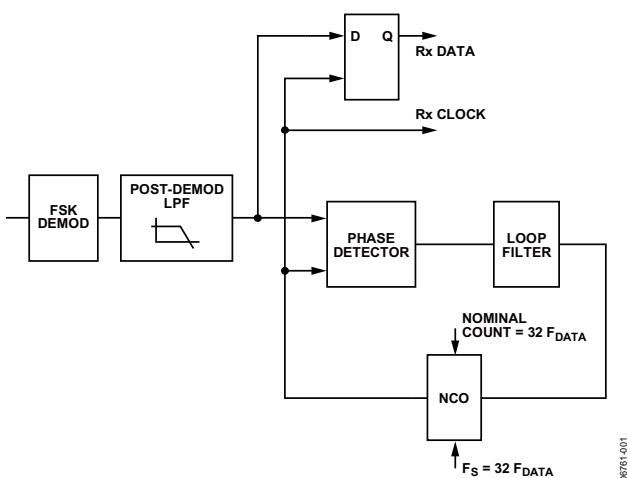


图1 符号定时恢复PLL框图

数据速率容差最大化

CDR数据速率容差取决于发送位流中的位跃迁数。当以一个最大跃迁形式(10101010...)为前同步码时，该容差达到最大值，CDR锁定时间达到最小值。在单个位周期中，允许进行1/32位周期的最大NCO相位调节。因此，当前同步码为101010...时，产生的最大的数据速率容差为 $1/32 \times 100\% = \pm 3.13\%$ 。

但是，该数据速率容差在数据字段中被降低，如图2所示，其中，位跃迁不能保证以规则间隔出现。一般而言，决定CDR的实际数据速率容差的是发送数据字段的游程长度限制(RLL)属性。

编码的RLL属性定义为编码位流中最大的相同邻接位的个数。总体而言，全部编码方案均由一个约束(d, k)定义，其中，d, k分别表示不同字符之间的最小和最大相同字符数。

举例来说，一条(d, k)等于(0, 4)的编码的邻接相同位数不超过4。可见，总体而言，决定最大数据速率容差的是用于数据字段中的编码的k约束。

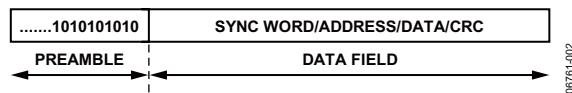


图2 典型消息格式

举个例子，曼彻斯特(Manchester)编码方案中，最大RLL为2位，即是说，曼彻斯特编码序列中的最大相同邻接位数为2位。例如，若曼彻斯特编码器输入码字为0101，其输出码字为01100110，是输入数据速率的两倍。则最大容差为曼彻斯特编码输出速率的 $\pm 3.125\% / 2 = \pm 1.56\%$ 。但实际上，容差大于该值，因为曼彻斯特编码器输出端的平均RLL小于2。仿真表明，给定一个随机输入的二进制位序列，曼彻斯特编码方案可以支持约 $\pm 2\%$ 的数据速率容差。

还有一种较简单的编码方案可用于实现数据速率容差最大化，其方法是按特定时间间隔向发送位流中插入额外的数据位，从而确保获得指定的最大RLL。这种编码方式称为位填充编码。

这种方案的优势在于易于实现，而且不存在曼彻斯特编码方案(其码率为1/2)带来的高码率损失问题。

目录

简介	1
数据速率容差最大化	1
游程长度约束最大化	3
小结	3

表1对此类编码方案的数据速率容差进行了总结。可考虑使用其他高速率编码(如8/9速率编码或8/10速率编码)，但其代价是更高的硬件或软件复杂性。

表1 DPLL频率跟踪范围

数据编码方案	容差
1010101... 前同步码	±3.125%
曼彻斯特编码, 最大RLL = 2,	±2.00%
随机输入数据	
位填充编码, RLL = 3,	±1.60%
随机输入数据	

游程长度约束最大化

在无线流等应用中，或者在采用基于大分组的协议时，须使用特定形式的数据编码方案，以确保数据字段中存在一个有限RLL约束，从而支持CDR正常运行。

然而，当数据速率容差和分组大小很小时，也可容许使用未编码数据字段或使用大RLL编码约束来编码数据字段。这种情况通常存在于基于分组的无线传感器遥测应用中，此类情况下的数据有效载荷通常少于64字节。另外，通常需对数据速率容差进行严格控制，因为这种容差是通过匹配发送晶体和接收晶体来设置的。如果所选数据速率精确地为晶体的约数，这通常可使名义频率误差优于 ± 50 ppm或0.005%。

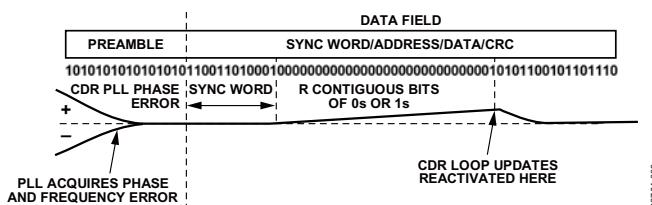


图3 未编码数据字段的相位误差轨迹

在图3所示例子中，采用的前同步码为10101010...，其目的是为了实现快速位同步，并对因Tx与Rx之间的数据速率差导致的频率误差进行跟踪。在前同步码结束时，CDR继续跟踪相位和频率误差，同时恢复同步字。同步字之后的一长串0表示CDR未发现允许跟踪频率误差的任何字符跃迁。在此期间，CDR相位误差呈单调递增(或递减，具体取决于频率误差的正负号)，直到发送位流中出现新的数据跃迁。

这种情况下，CDR仍可恢复数据，条件是最大相位误差处于约 $\pm 10\%$ 位周期之内。CDR相位误差增加的速率与数据速率容差成比例，如果必须支持长RLL值，则应使数据速率容差尽可能小，这一点十分重要。因此，在此类情况下，

如果通过减小数据速率容差和/或分组大小以限制PLL相位累积，则可支持较大RLL约束。

通过以下公式，可确定数据速率容差与RLL约束之间的关系：

$$\text{相位} = 2\pi F t$$

其中， Δt 定义为邻接数据位数R。

如果将相位累积限制为1/10位周期

$$0.1 \equiv \Delta F_{DATA} / R \; Tbit$$

其中 $T_{bit} = 1/F_{DATA}$

则

$$0.1 = \left(\frac{\Delta F_{DATA}}{F_{DATA}} \right) R$$

对于给定数据速率容差($\Delta F_{DATA}/F_{DATA}$)，可容许的相同邻接数据位数R为

$$R = \left(\frac{F_{DATA}}{\Delta F_{DATA}} \right) 0.1$$

表2所列示例为约束长度R与数据速率容差之间的关系。

表2 约束长度R与数据速率容差

数据速率容差 % ($\Delta F_{DATA}/F_{DATA}$)	$F_{DATA}/\Delta F$	R位数
0.1%	1000	100
0.05%	2000	200
0.01%	10,000	1000

以上分析假定，R邻接数据位在分组中出现于相分离的位段。假如存在以下情况，其中，上述位形式被短的随机数据位分开，而这些数据位可能不允许CDR在突发邻接数据之间完全恢复相位误差。这种情况下，最好假定RLL约束为突发数据组合后的总长度，然后再基于这个加大的R值计算数据容差要求(即Tx与Rx Xtal的匹配要求)。

在RLL未知的情况下，或当未编码数据字段中的整个分组都可能存在邻接数据位时，应根据一个等于分组总长度的RLL值计算数据容差要求($R = \text{分组总长度}$)。

总的来说，应将分组加扰作为一种拆分和随机化长邻接数据位形式的技术。

小结

在流应用中，可能需要无线发送数十kB数据，建议采用特

定形式的数据编码方案来最小化RLL。这样，CDR电路就可以成功跟踪输入数据，即使存在长串的连续1或0。

例如，在突发型分组系统中，有效载荷通常少于64字节，如果将CDR_CLK与 $32 \times$ DATARATE之间的名义CDR频率误差降至最低，则CDR可以容许长串的连续1或0。

最好方法是选择一个适当的DATARATE和晶体组合，使

DATARATE精确地为晶体的约数。例如，为了降低9.6 kbps DATARATE的CDR频率误差，11.0592 MHz晶体不失为一种好选择。这种情况下，名义频率误差即为发送晶体与接收晶体之间的频率误差，结果通常可使名义频率误差优于 ± 50 ppm或0.005%。该频率误差允许2000个邻接位。而如果选择12 MHz晶体，其名义频率误差接近2.2%，最大RLL仅为2位。