



Noel Fung

摘要

在集成 VCO 的射频合成器首次推出之际，典型的单个 VCO 内核合成器便得以实施，因此开发出了大量具有相同 PLL 但不同 VCO 频率内核的器件来支持不同的应用。现代射频合成器集成了多个 VCO 内核和输出分频器，使单个器件支持非常低到非常高的宽输出频率。为确保所有 VCO 内核协调工作，需要进行适当的 VCO 校准。校准对 VCO 频率的选择至关重要，对频率开关时间也很重要。本应用手册阐述了 VCO 校准的必要性、其工作原理以及最大限度地缩短 LMX2594 宽带射频合成器校准时间的方法。本应用手册适用于 LMX2594、LMX2595 和 LMX2615-SP 器件。

内容

1 宽带 VCO 实现方案	3
1.1 分立式 VCO.....	3
1.2 硅基开关电容器 VCO.....	3
1.3 VCO Capcode、VCO 内核和 VCO 幅度.....	4
2 LMX2594 中的 VCO 校准	5
2.1 校准速度.....	5
2.2 初始校准参数设置.....	9
3 减少校准时间	13
3.1 SPI 编程速度.....	13
3.2 部分辅助.....	14
3.3 完全辅助.....	15
4 总结	17
5 参考文献	17
6 修订历史记录	17

插图清单

图 1-1. LC 谐振电路.....	3
图 1-2. 压控振荡器.....	3
图 1-3. 开关电容器谐振电路.....	3
图 1-4. VCO Capcode.....	4
图 2-1. TICS Pro 默认配置.....	5
图 2-2. $f_{SM} = 200\text{MHz}$	6
图 2-3. $f_{SM} = 100\text{MHz}$	6
图 2-4. $f_{SM} = 50\text{MHz}$	6
图 2-5. $f_{SM} = 25\text{MHz}$	6
图 2-6. FCAL_HPFD_ADJ = 0x0 ($f_{PD} \leq 100\text{MHz}$).....	7
图 2-7. FCAL_HPFD_ADJ = 0x1 ($100 < f_{PD} \leq 150\text{MHz}$).....	7
图 2-8. FCAL_HPFD_ADJ = 0x2 ($150 < f_{PD} \leq 200\text{MHz}$).....	7
图 2-9. FCAL_HPFD_ADJ = 0x3 ($f_{PD} > 200\text{MHz}$).....	7
图 2-10. ACAL_CMP_DLY = 10, 上跳.....	8
图 2-11. ACAL_CMP_DLY = 10, 下跳.....	8
图 2-12. ACAL_CMP_DLY = 20, 上跳.....	8
图 2-13. ACAL_CMP_DLY = 20, 下跳.....	8
图 2-14. ACAL_CMP_DLY = 40, 上跳.....	9
图 2-15. ACAL_CMP_DLY = 40, 下跳.....	9

图 2-16. VCO_SEL = 1, 上跳.....	10
图 2-17. VCO_SEL = 1, 下跳.....	10
图 2-18. VCO_SEL = 4, 上跳.....	10
图 2-19. VCO_SEL = 4, 下跳.....	10
图 2-20. VCO_CAPCTRL_STRT = 1, 上跳.....	11
图 2-21. VCO_CAPCTRL_STRT = 1, 下跳.....	11
图 2-22. VCO_CAPCTRL_STRT = 255, 上跳.....	11
图 2-23. VCO_CAPCTRL_STRT = 255, 下跳.....	11
图 2-24. VCO1 跳到 VCO3.....	12
图 2-25. VCO3 跳到 VCO1.....	12
图 2-26. VCO5 跳到 VCO7.....	12
图 2-27. VCO7 跳到 VCO5.....	12
图 3-1. SPI CLK = 25MHz.....	13
图 3-2. SPI CLK = 2.5MHz.....	13
图 3-3. 部分辅助工作流程.....	15
图 3-4. 完全辅助工作流程.....	16
图 3-5. 自动校准, 上跳.....	16
图 3-6. 自动校准, 下跳.....	16
图 3-7. 部分辅助, 上跳.....	17
图 3-8. 部分辅助, 下跳.....	17
图 3-9. 完全辅助, 上跳.....	17
图 3-10. 完全辅助, 下跳.....	17

表格清单

表 3-1. LMX2594 VCO 内核范围.....	14
------------------------------	----

商标

所有商标均为其各自所有者的财产。

1 宽带 VCO 实现方案

电感器与电容器并联形成了一个谐振电路，如图 1-1 所示。

如果用户用变容二极管代替电容器，并在变容二极管上施加不同的电压，就可控制谐振频率。

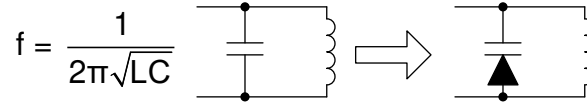


图 1-1. LC 谐振电路

1.1 分立式 VCO

图 1-2 所示为一个典型的 Colpitts 振荡器，它使用变容二极管来实现频率调谐。调谐电压 V_{tune} 用于改变变容二极管的电容，从而改变振荡器的频率。

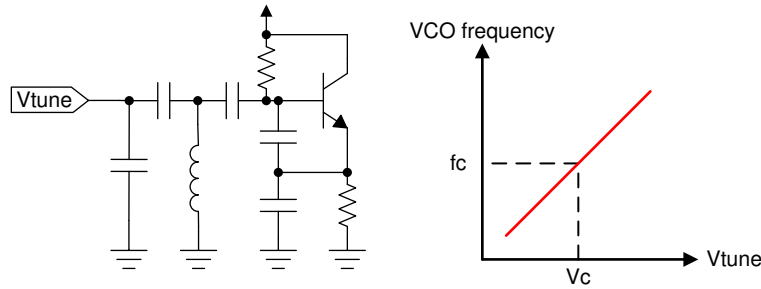


图 1-2. 压控振荡器

假设当 $V_{tune} = V_c$ 时，VCO 的频率为 f_c 。

如果 VCO 增益（也称为 K_{vco} ）为 10MHz/V ，则 $V_c \pm 1\text{V}$ 的相应输出频率为 $f_c \pm 10\text{MHz}$ 。

分立式 VCO 的频率调谐范围不能很宽。通常，调谐范围约为 VCO 中心频率的 10% 至 15%。

1.2 硅基开关电容器 VCO

如果 VCO 内置在芯片中，则可以在应用中切换不同的电容器，从而调整 VCO 的自由运行频率。

换言之，在 V_{tune} 固定的情况下，VCO 的中心频率是可调的。例如，如果将 C_1 接通，则对应的输出频率为 f_1 。因此，VCO 的频率调谐范围已从 $f_c \pm V_{tune}$ 扩展至 $f_1 \pm V_{tune}$ 。

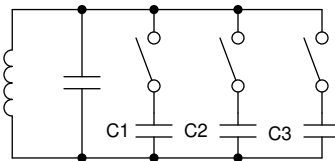


图 1-3. 开关电容器谐振电路

1.3 VCO Capcode、VCO 内核和 VCO 幅度

本文档中的术语“Capcode”表示开关电容器的电容值。在 LMX2594 中，Capcode (寄存器 R19 VCO_CAPCTRL) 介于 0 和 255 之间。Capcode 值越高，VCO 频率越低。为了支持 7.5GHz 至 15GHz 的 VCO 调谐范围，单个开关电容器 VCO 仍然不可行。LMX2594 使用 7 个 VCO 内核 (寄存器 R20 VCO_SEL) 来涵盖此范围。每个 VCO 具有相似的 Capcode 结构，但电感值不同。

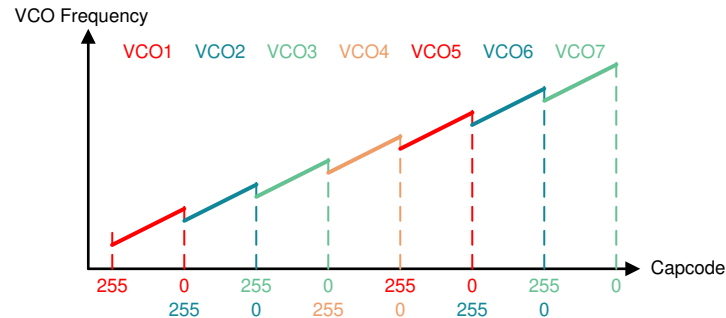


图 1-4. VCO Capcode

NOTE

每个相邻的 VCO 内核都有一些重叠区域。这是为了确保实现 VCO 内核无缝转换和应对工艺-电压-温度 (PVT) 变化。

使用 256 个 Capcode 和 7 个 VCO 内核时，总共有 1792 种组合。对于特定频率，手动选择正确的 VCO 内核和 Capcode 即使并非天方夜谭，也是困难重重。芯片中嵌入了一个 VCO 校准算法，因此 Capcode 和 VCO 内核的选择对用户而言是透明的。除了选择频率之外，校准算法还通过调整所选 VCO (寄存器 R16 VCO_DACISSET) 的偏置来优化相位噪声。

2 LMX2594 中的 VCO 校准

尽管 VCO 校准已尽可能方便用户使用，但请注意确保校准的可靠性并实现出色性能。
在本应用手册的后续部分中，如图 2-1 所示的 TICS Pro 配置将用作演示的默认配置。

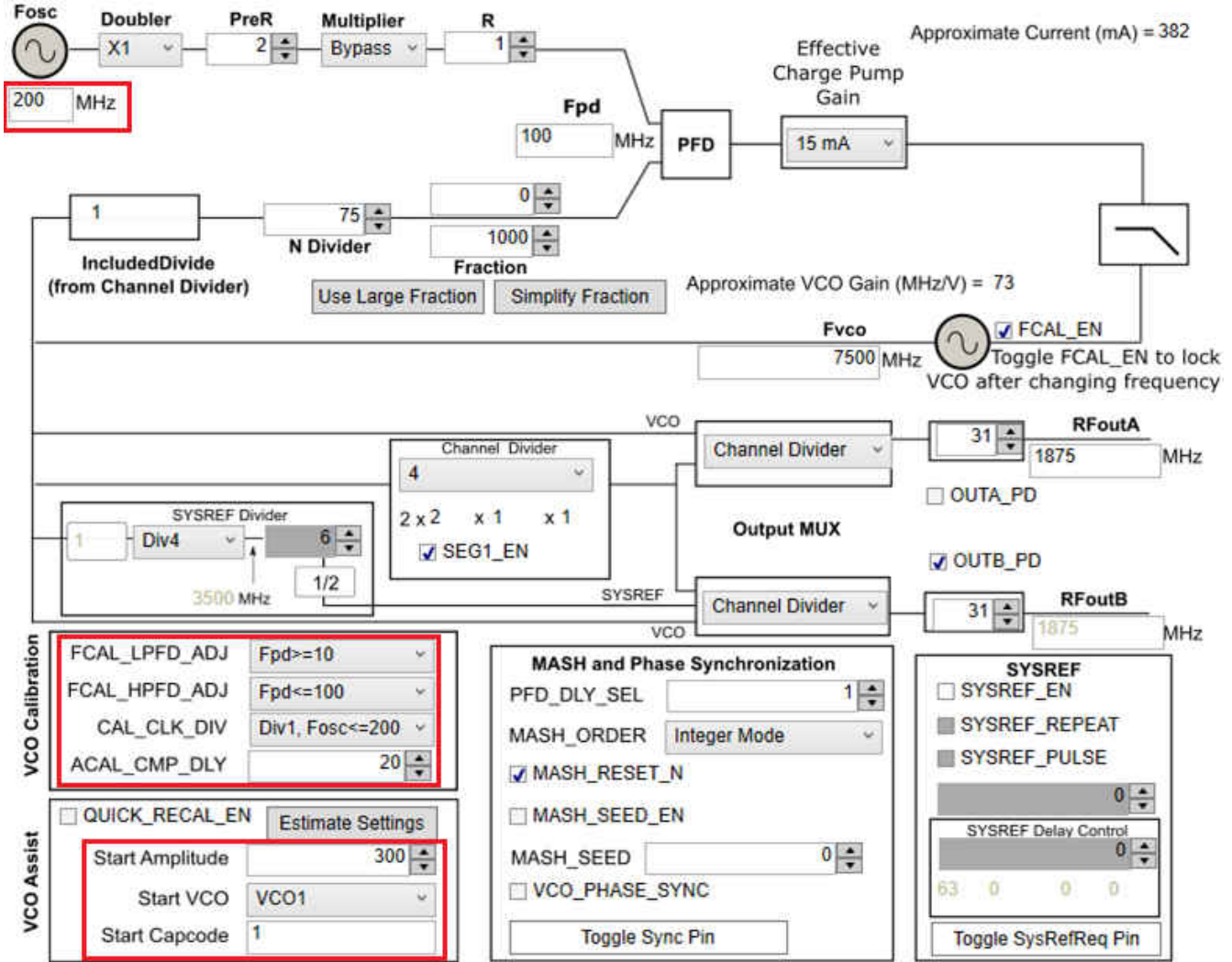


图 2-1. TICS Pro 默认配置

请注意以下有关默认配置的事项：

- 为了实现更好的演示效果，将“Channel Divider”设置为了 4，使频率标度更小，因此更易于读取。
- 为了简化操作，仅使用整数通道进行演示。
- 默认情况下，LMX2594 将按升序扫描 VCO 内核。

2.1 校准速度

校准速度对 VCO 校准的精度影响最大。当然，校准速度也会直接影响校准时间。为了平衡校准时间和精度，应正确设置图 2-1 中突出显示的项目。以下几节将说明这些参数如何影响校准行为。

2.1.1 状态机时钟

LMX2594 的 VCO 校准在状态机时钟频率 f_{SM} 下运行。状态机时钟的时钟来源是参考时钟。最大 f_{SM} 为 200MHz。寄存器 R1 CAL_CLK_DIV 用于设置该频率。

$$f_{SM} = f_{OSC} / 2^{CAL_CLK_DIV} \quad (1)$$

在以下各图中，VCO 使用不同的状态机时钟频率从 7500MHz 跳至 15000MHz。编程顺序为：

1. 将寄存器 R36 PLL_N 从 75 编程为 150 (十进制)。
2. 对寄存器 R0 编程一次以触发 VCO 校准。

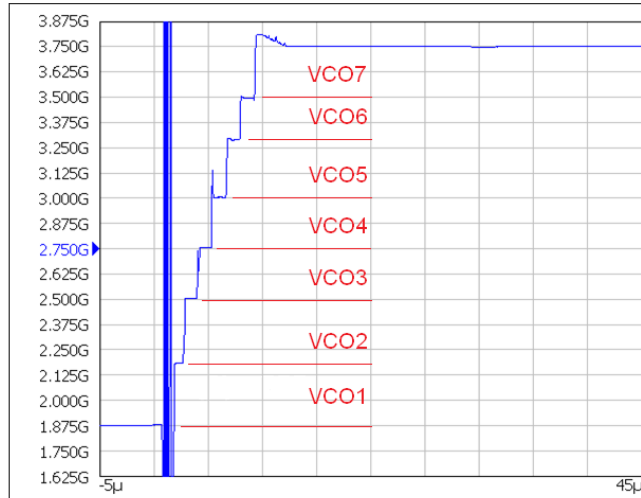


图 2-2. $f_{SM} = 200\text{MHz}$

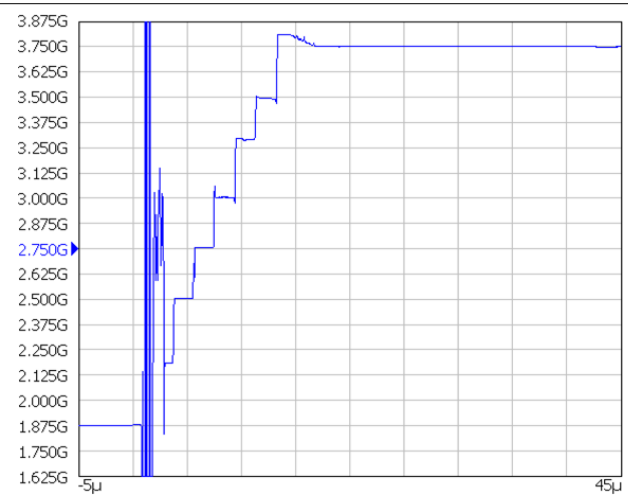


图 2-3. $f_{SM} = 100\text{MHz}$

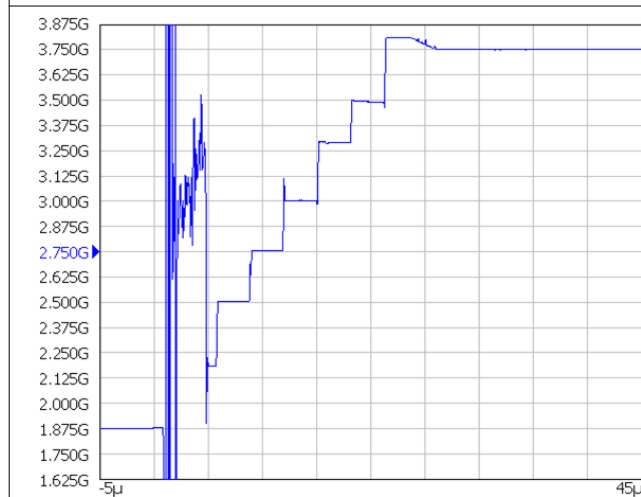


图 2-4. $f_{SM} = 50\text{MHz}$

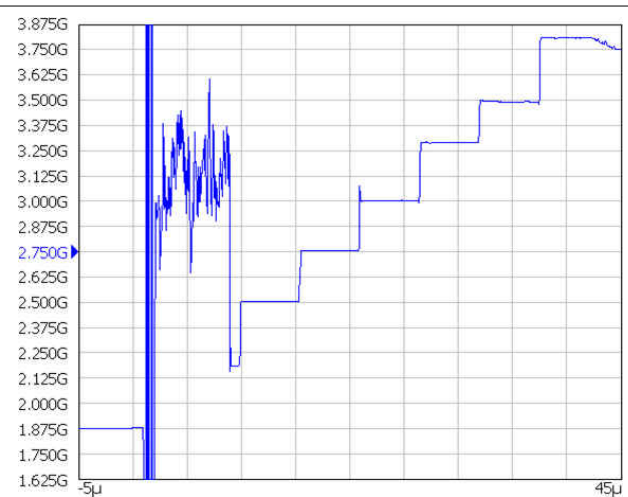


图 2-5. $f_{SM} = 25\text{MHz}$

如上所示， f_{SM} 越高，校准时间越短。如果频率开关时间在某些应用中很重要，则需要更高的参考时钟频率源。

2.1.2 FCAL_xxxx_ADJ

在寄存器 R0 中，FCAL_LPFD_ADJ 和 FCAL_HPFADJ 分别用于在相位检测器频率 f_{PD} 过低或过高时调整校准速度。

在大多数用例中， f_{PD} 不会低于 10MHz，因此 FCAL_LPFD_ADJ 基本用不上。

相反，如果 f_{PD} 过高，则应当适当设置 FCAL_HPFADJ 以减慢校准速度。

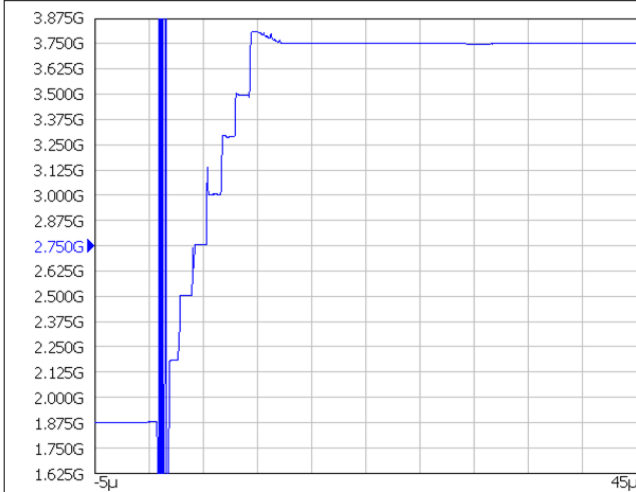


图 2-6. FCAL_HPFADJ = 0x0 ($f_{PD} \leq 100\text{MHz}$)

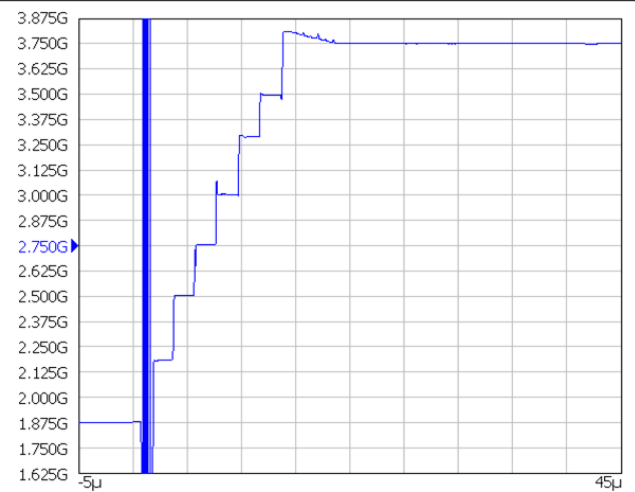


图 2-7. FCAL_HPFADJ = 0x1 ($100 < f_{PD} \leq 150\text{MHz}$)

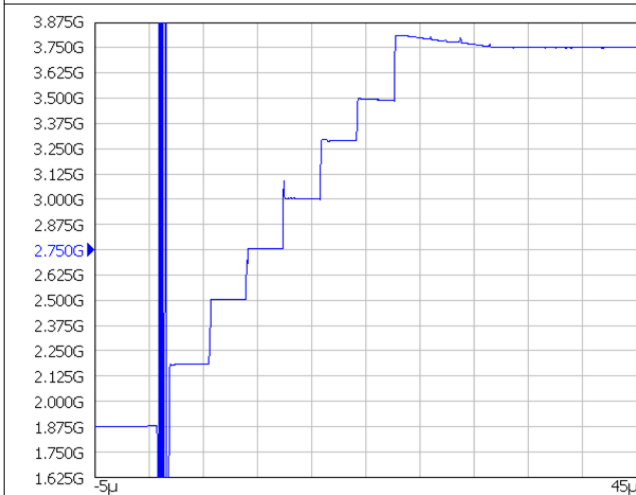


图 2-8. FCAL_HPFADJ = 0x2 ($150 < f_{PD} \leq 200\text{MHz}$)

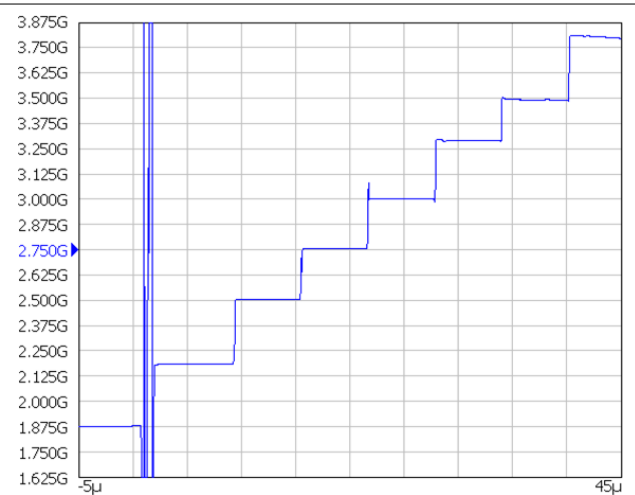


图 2-9. FCAL_HPFADJ = 0x3 ($f_{PD} > 200\text{MHz}$)

状态机时钟频率在此处保持不变，但随着 FCAL_HPFADJ 值的升高，校准速度会降低。

2.1.3 ACAL_CMP_DLY

在寄存器 R4 中，ACAL_CMP_DLY 是为校准 VCO 幅度而添加的延迟。在幅度校准期间，将逐步调整 VCO 的偏置电压，直到获得理想的相位噪声。该偏置电压来自内部 LDO。LDO 电压稳定下来需要时间，因此每个阶跃电压变化都需要延迟。降低该值可以加快 VCO 校准速度，但降低太多可能会使 VCO 相位噪声性能下降。为了让 VCO 在所有场景下校准到正确频率，该字段的最小允许值为 10。若要产生理想和可重复性较高的 VCO 相位噪声，应根据方程式 2 找到 ACAL_CMP_DLY 值。

$$ACAL_CMP_DLY > f_{SM} / 10MHz \quad (2)$$

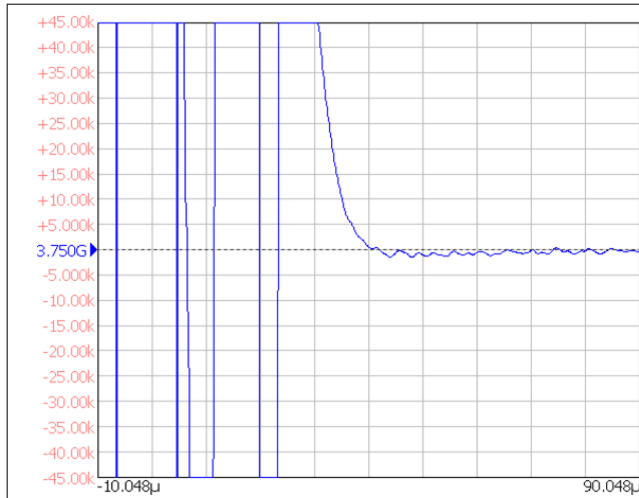


图 2-10. ACAL_CMP_DLY = 10，上跳

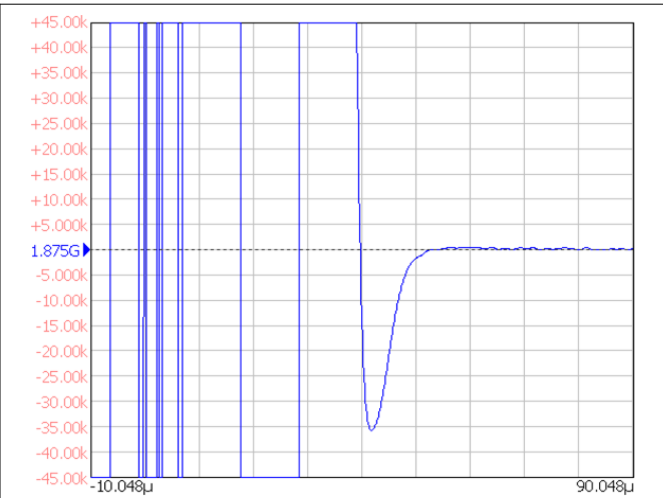


图 2-11. ACAL_CMP_DLY = 10，下跳

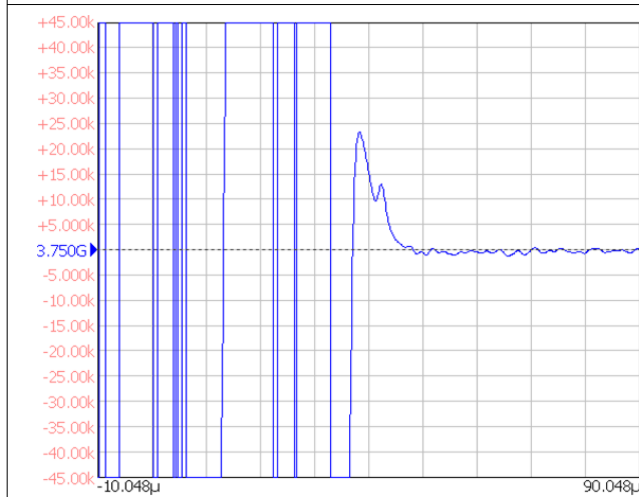


图 2-12. ACAL_CMP_DLY = 20，上跳

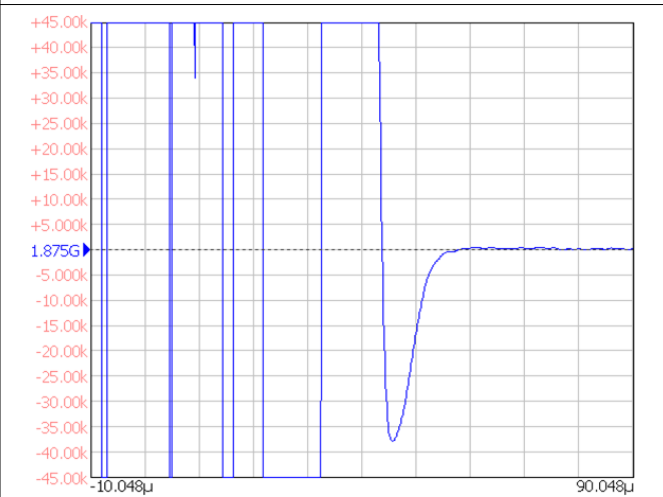


图 2-13. ACAL_CMP_DLY = 20，下跳

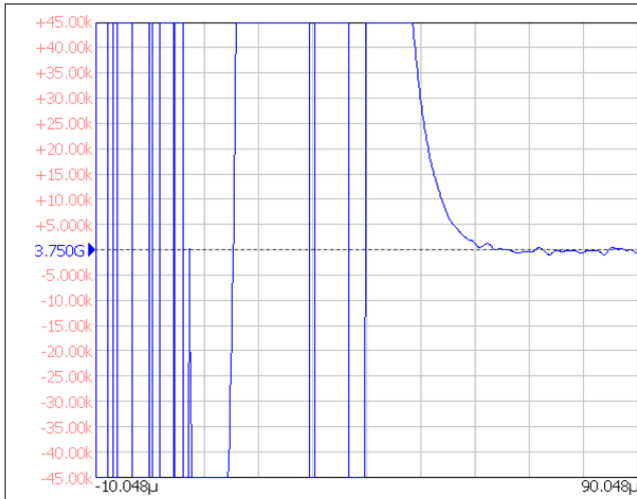


图 2-14. ACAL_CMP_DLY = 40 , 上跳

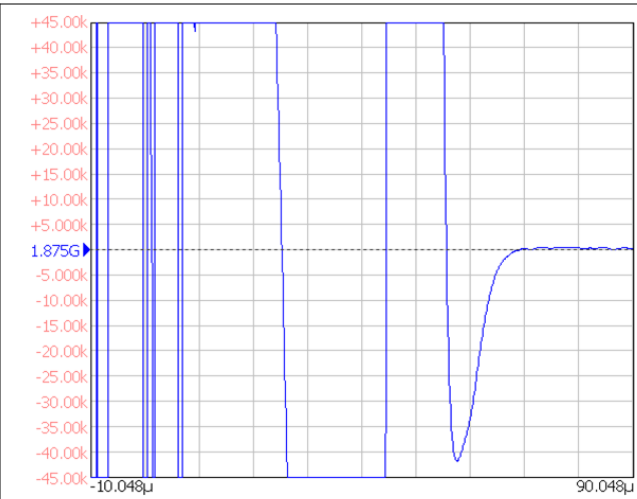


图 2-15. ACAL_CMP_DLY = 40 , 下跳

在以上各图的左栏中，VCO 从 7500MHz 跃升至 15000MHz。在右栏中，VCO 频率被翻转。

这些示例中使用的状态机时钟为 200MHz。根据方程式 2 可知晓 ACAL_CMP_DLY = 20。从图中可以看出，与最小 ACAL_CMP_CAL 要求“10”相比，总锁定时间增加了约 10μs。

2.2 初始校准参数设置

无论当前设置如何，VCO 校准将始终从预定义的 VCO 内核、VCO Capcode 和幅度设置开始。这些设置将会影响校准时间。

2.2.1 VCO_SEL

在寄存器 R20 中，VCO_SEL 定义了 VCO 校准开始时使用的 VCO 内核。以下各示例比较了不同 VCO_SEL 设置下 (VCO 上跳和下跳) 的校准时间。

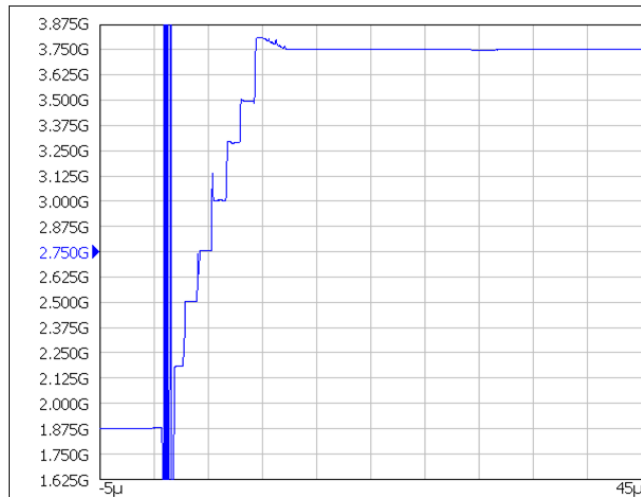


图 2-16. VCO_SEL = 1, 上跳

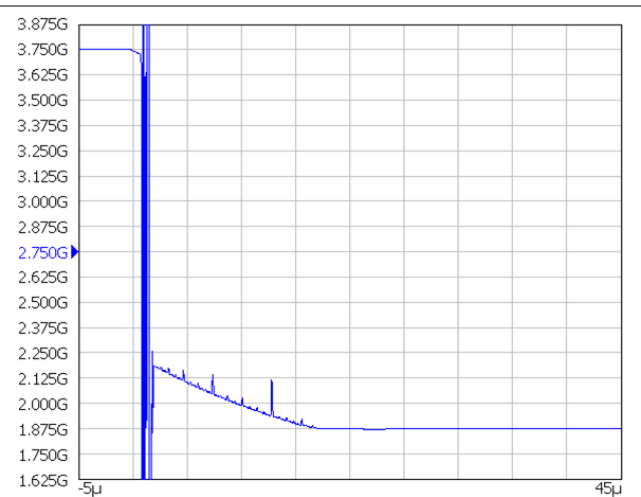


图 2-17. VCO_SEL = 1, 下跳

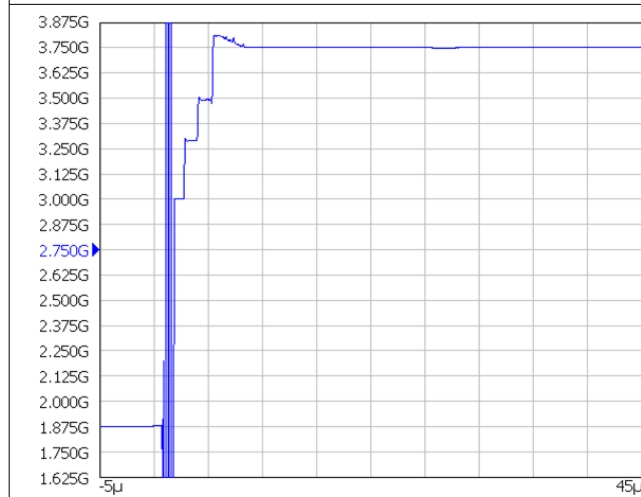


图 2-18. VCO_SEL = 4, 上跳

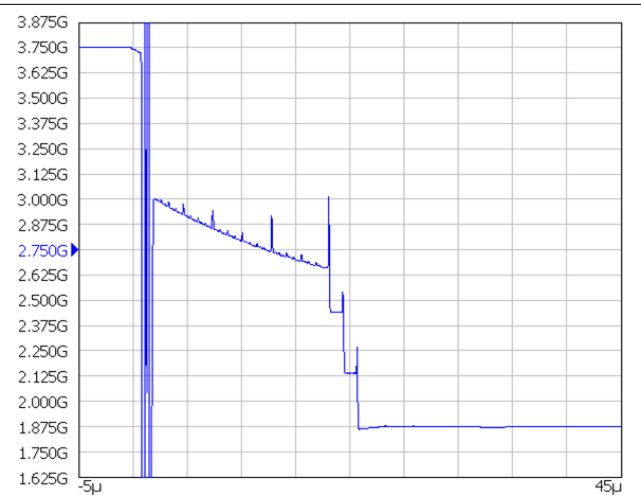


图 2-19. VCO_SEL = 4, 下跳

VCO_SEL 对上跳和下跳的校准时间都有影响。在上面的示例中，如果我们使用 VCO4 作为起始 VCO，上跳校准时间会更短。但是，下跳校准时间比使用 VCO1 作为起始 VCO 时更长。在某些用例中，将此设置为 VCO1 或 VCO7 可能很有益，一种均衡的方法应使其从整个 VCO 覆盖范围的中间开始。

2.2.2 VCO_CAPCTRL_STRT

寄存器 R78 VCO_CAPCTRL_STRT 设置用于 VCO 校准的初始 Capcode 值。Capcode 值介于 0 和 255 之间。0 代表最高 VCO 频率设置，而 255 将 VCO 驱动到最低频率边界。

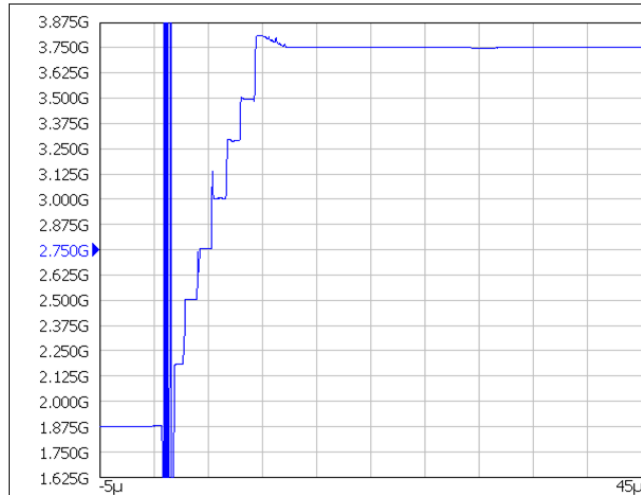


图 2-20. VCO_CAPCTRL_STRT = 1，上跳

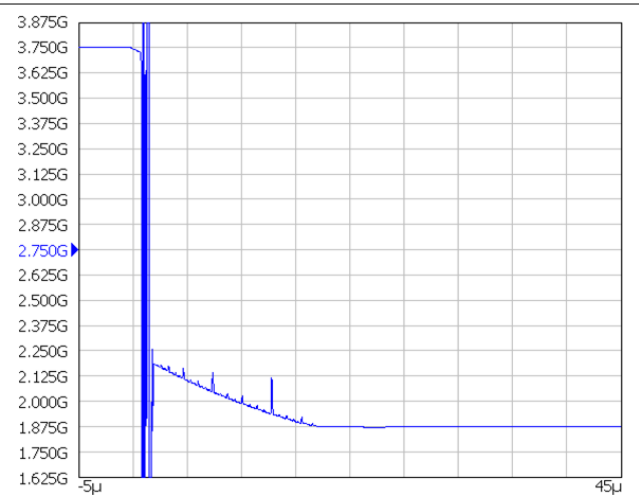


图 2-21. VCO_CAPCTRL_STRT = 1，下跳

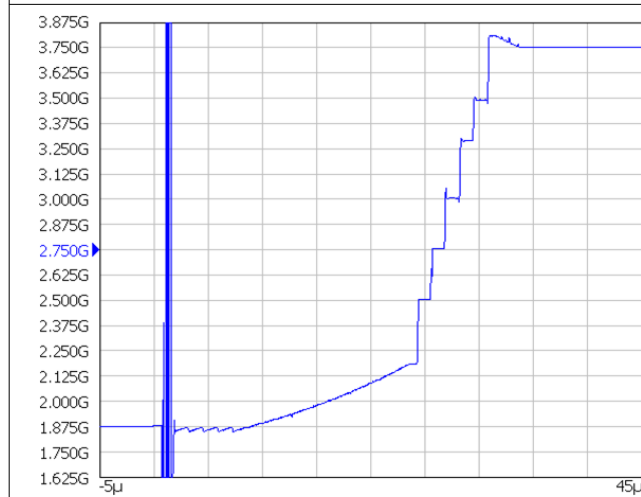


图 2-22. VCO_CAPCTRL_STRT = 255，上跳

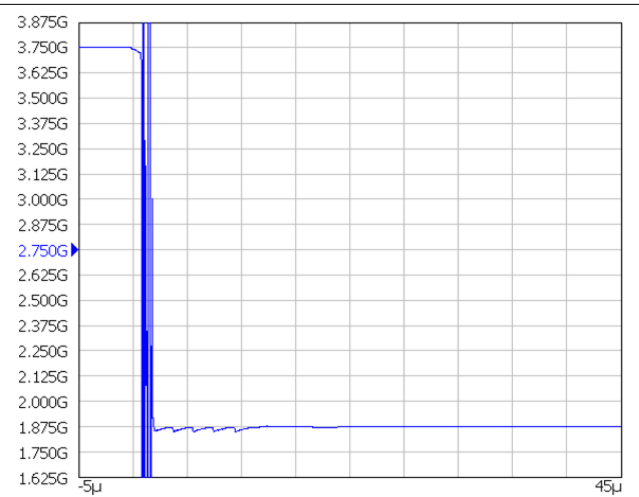


图 2-23. VCO_CAPCTRL_STRT = 255，下跳

由于 VCO 校准算法的行为，如果校准从 Capcode = 255 开始，校准时间通常会更长。

2.2.3 VCO_DACISSET_STRT

寄存器 R17 VCO_DACISSET_START 设置在 VCO 校准开始时使用的 VCO 幅度值。该寄存器对校准时间的影响相对较小。

2.2.4 建议的初始校准参数设置

为了在上跳和下跳行为上实现折衷，建议使用以下初始校准参数：

- VCO_DACISSET_STRT = 300
- VCO_SEL = 4
- VCO_CAPCTRL_STRT = 1

以下是 VCO 如何上下跳跃的一些示例（使用了建议的初始校准参数）。以下各图中 VCO 内核的频率为：

- VCO1 = 7500MHz。
- VCO3 = 10GHz。
- VCO5 = 12.4GHz。

- VCO7 = 15GHz。

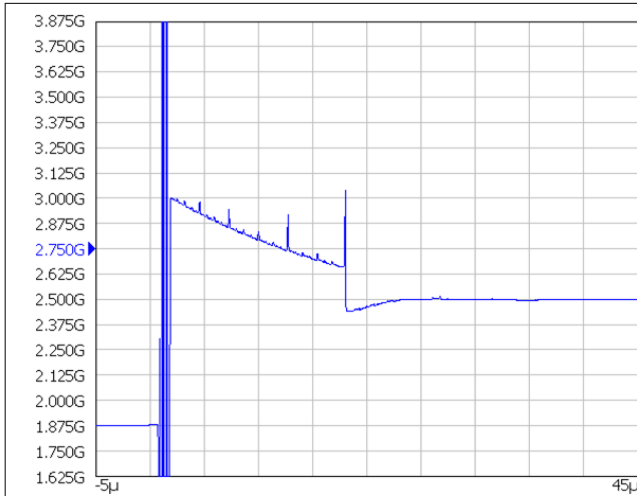


图 2-24. VCO1 跳到 VCO3

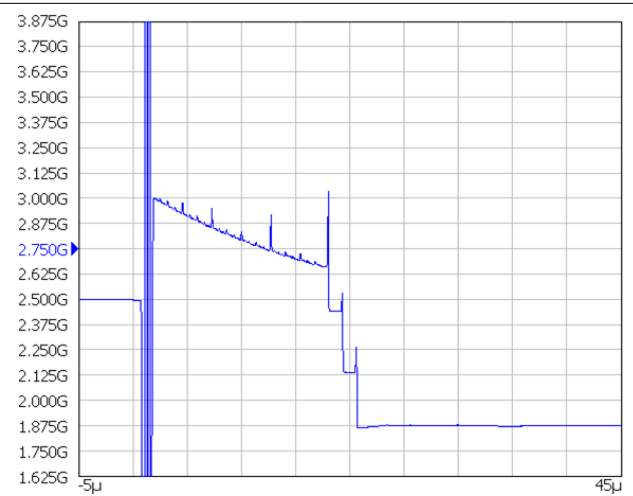


图 2-25. VCO3 跳到 VCO1

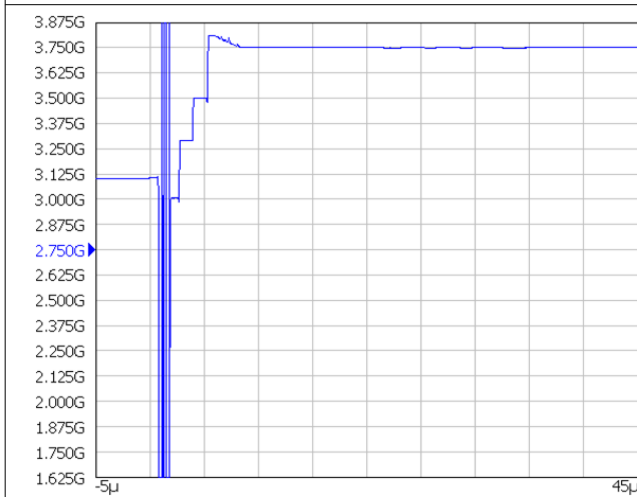


图 2-26. VCO5 跳到 VCO7

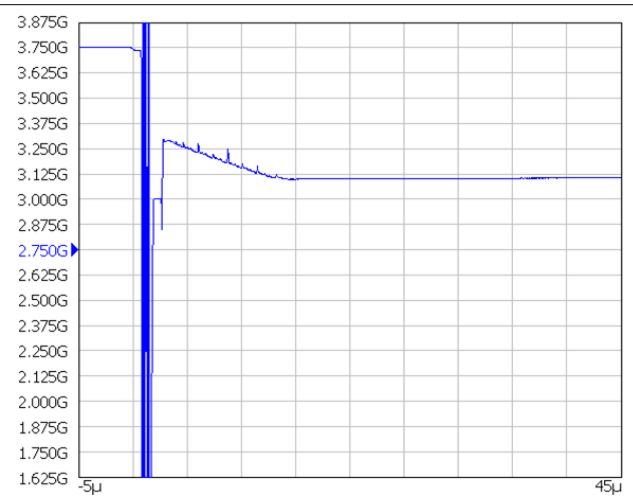


图 2-27. VCO7 跳到 VCO5

3 减少校准时间

使用建议的初始校准参数和正确的校准速度设置，VCO 算法将根据正确的 Capcode 和幅度设置选择正确的 VCO 内核。但是，必须为 VCO 校准例程保留足够的完成时间。若要控制校准时间或大幅缩短校准时间，请在 VCO 校准中使用部分辅助或完全辅助模式。VCO 校准时间越短，总的 VCO 频率开关时间就越短，因此适用于跳频或扫频系统。

3.1 SPI 编程速度

在部分辅助或完全辅助模式下，除了必须更新的频率相关寄存器（如 N 分频器和分数分子）外，还有几个额外的寄存器需要编程。总之，至少必须对五个寄存器进行编程。一个寄存器的长度为 24 位 (DATA) + 1 位 (CSB)，因此，在 1MHz SPI 时钟速率下写入寄存器需要 25 μ s。此外，与寄存器关联的功能或值在写入寄存器后会立即生效。例如，如果更新 N 分频器，则 VCO 频率会立即漂移，并且 PLL 环路会在用户写入 R0 寄存器时解锁。所有这些变化都发生在 VCO 开始校准之前。

PLL 环路仍在闭合，因此电荷泵电压会进入电源轨。这可能会在 VCO 校准完成后环路再次闭合之时影响 PLL 锁定时间，因为环路滤波器中的电容器可能需要更多时间来进行充电或放电。

以下各图显示了使用更高 SPI 时钟速率的好处。VCO 从 VCO5 (12.4GHz) 切换到了 VCO7 (14.8GHz)，且通道分频器 CHDIV = 8。首先更新了 N 分频器，然后是寄存器 R0，从而触发 VCO 校准。如图 3-2 所示，即使 SPI 时钟速率为 2.5MHz，寄存器写入也需要 10 μ s。与此同时，VCO 已经漂移了约 80MHz。

TI 建议尽可能使用更高的 SPI 时钟速率。

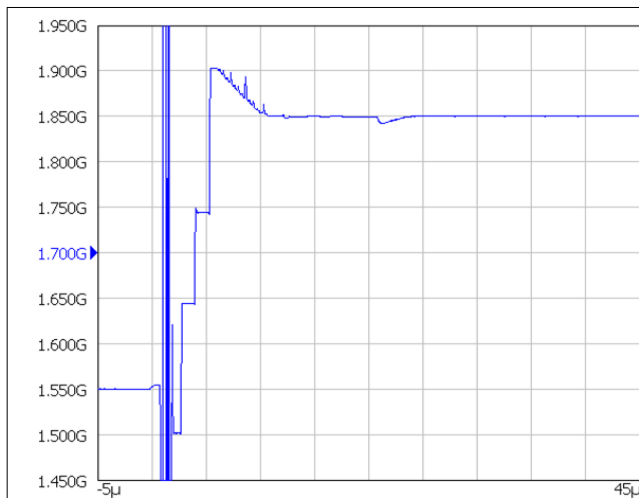


图 3-1. SPI CLK = 25MHz

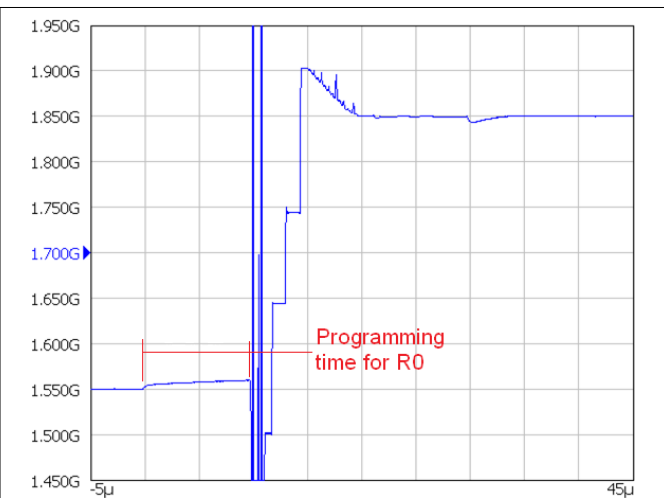


图 3-2. SPI CLK = 2.5MHz

3.2 部分辅助

在此模式下，用户不使用节 2.2.4 中建议的初始校准参数，而是在写入 R0 以触发 VCO 校准之前，用户根据表 3-1 为每次频率变化提供 VCO 内核 (VCO_SEL)、Capcode (VCO_CAPCTRL_STRT) 和幅度 (VCO_DACISSET_STRT) 的初始起点。若要执行部分辅助，请按以下过程操作：

1. 确定 VCO 内核。
 - 找到 VCO 频率符合要求的 VCO 内核。如果位于两个内核之间的边界，请根据相位噪声或性能选择一个内核。
2. 根据方程式 3 计算 VCO Capcode。

$$\text{VCO_CAPCTRL_STRT} = \text{round} [C_{\text{CoreMin}} - (C_{\text{CoreMin}} - C_{\text{CoreMax}}) \times (f_{\text{VCO}} - f_{\text{CoreMin}}) / (f_{\text{CoreMax}} - f_{\text{CoreMin}})] \quad (3)$$

3. 根据方程式 4 计算 VCO 幅度设置。

$$\text{VCO_DACISSET_STRT} = \text{round} [A_{\text{CoreMin}} - (A_{\text{CoreMin}} - A_{\text{CoreMax}}) \times (f_{\text{VCO}} - f_{\text{CoreMin}}) / (f_{\text{CoreMax}} - f_{\text{CoreMin}})] \quad (4)$$

表 3-1. LMX2594 VCO 内核范围

VCO 内核	f _{CoreMin}	f _{CoreMax}	C _{CoreMin}	C _{CoreMax}	A _{CoreMin}	A _{CoreMax}
VCO1	7500	8600	164	12	299	240
VCO2	8600	9800	165	16	356	247
VCO3	9800	10800	158	19	324	224
VCO4	10800	12000	140	0	383	244
VCO5	12000	12900	183	36	205	146
VCO6	12900	13900	155	6	242	163
VCO7	13900	15000	175	19	323	244

下面我们举例说明。

若要从 VCO1 (7500MHz) 跳到 VCO3 (10GHz)，请对以下寄存器进行编程：

- R36 = 0x240064 (PLL_N = 100)
- R20 = 0x14D848 (VCO_SEL = 3)
- R78 = 0x4E0105 (VCO_CAPCTRL_STRT = 130)
- R17 = 0x110130 (VCO_DACISSET_STRT = 304)
- R0 = 0x00241C (用于触发校准)

若要从 VCO3 (10GHz) 跳到 VCO1 (7500MHz)，请对以下寄存器进行编程：

- R36 = 0x24004B (PLL_N = 75)
- R20 = 0x14C848 (VCO_SEL = 1)
- R78 = 0x4E0149 (VCO_CAPCTRL_STRT = 164)
- R17 = 0x11012B (VCO_DACISSET_STRT = 299)
- R0 = 0x00241C (用于触发校准)

测试结果如图 3-7 和图 3-8 所示。

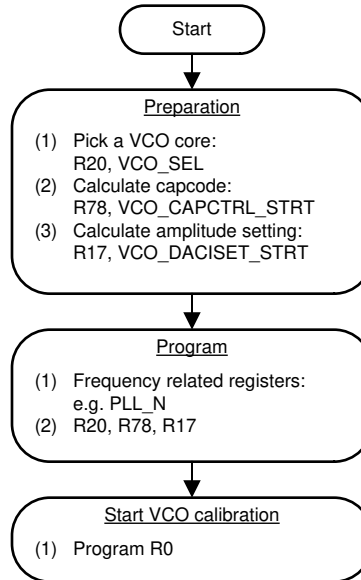


图 3-3. 部分辅助工作流程

3.3 完全辅助

在此模式下会完全绕过 VCO 校准。用户强制启用 VCO 内核 (VCO_SEL)、幅度设置 (寄存器 R16 VCO_DACISSET) 和 Capcode (寄存器 R19 VCO_CAPCTRL)，并手动设置相应的值。要针对特定的 VCO 频率获取这些值，用户必须提前运行自动校准。用户为此频率设置自动校准，然后在寄存器 R110 rb_VCO_SEL、R111 rb_VCO_CAPCTRL 和 R112 rb_VCO_DACISSET 中读回这些值。要使用这些值，用户必须启用寄存器 R20 VCO_SEL_FORCE 以及 R8 VCO_CAPCTRL_FORCE 和 VCO_DACISSET_FORCE 位。

此处有一个示例，假设 VCO_SEL_FORCE、VCO_CAPCTRL_FORCE 和 VCO_DACISSET_FORCE 都等于 1。

要从 VCO1 (7500MHz) 跳到 VCO3 (10GHz)，请对以下寄存器进行编程：

- R36 = 0x240064 (N = 100)
- R20 = 0x14DC48 (VCO_SEL = 3)
- R19 = 0x132787 (VCO_CAPCTRL = 135)
- R16 = 0x10012E (VCO_DACISSET = 302)

要从 VCO3 (10GHz) 跳到 VCO1 (7500MHz)，请对以下寄存器进行编程：

- R36 = 0x24004B (N = 75)
- R20 = 0x14CC48 (VCO_SEL = 1)
- R19 = 0x1327AB (VCO_CAPCTRL = 171)
- R16 = 0x10012C (VCO_DACISSET = 300)

测试结果如图 3-9 和图 3-10 所示。

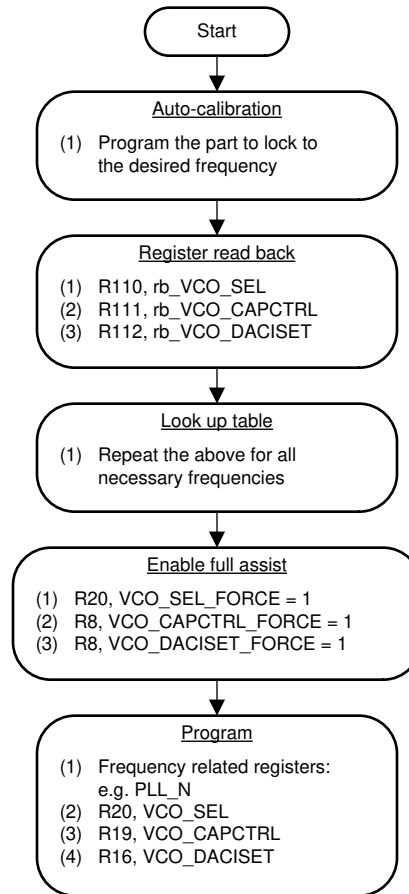


图 3-4. 完全辅助工作流程

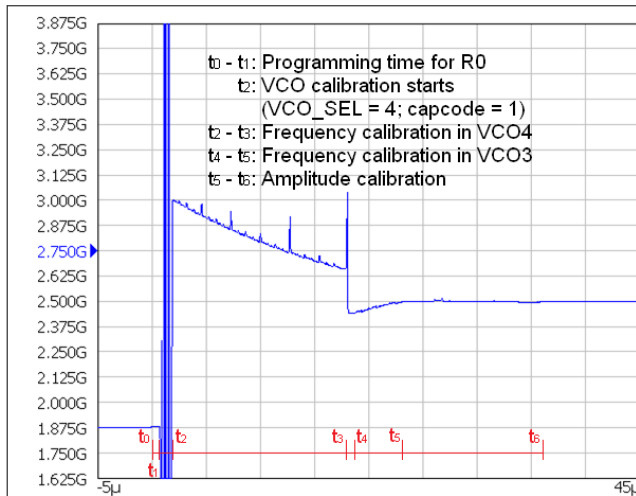


图 3-5. 自动校准，上跳

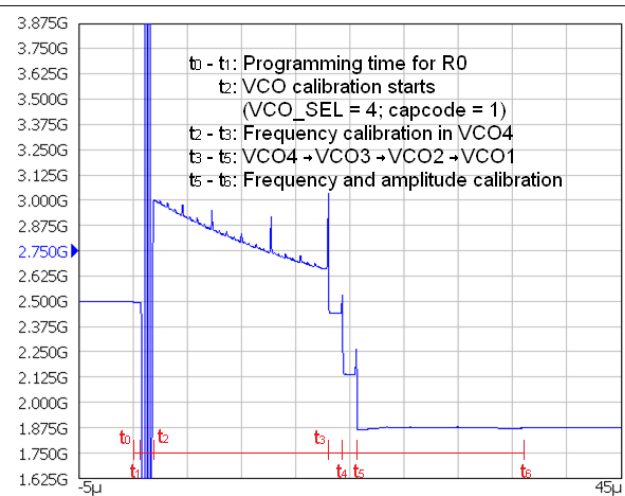


图 3-6. 自动校准，下跳

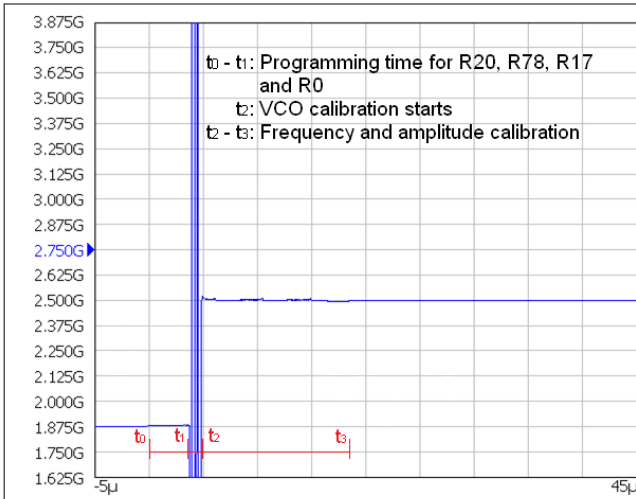


图 3-7. 部分辅助，上跳

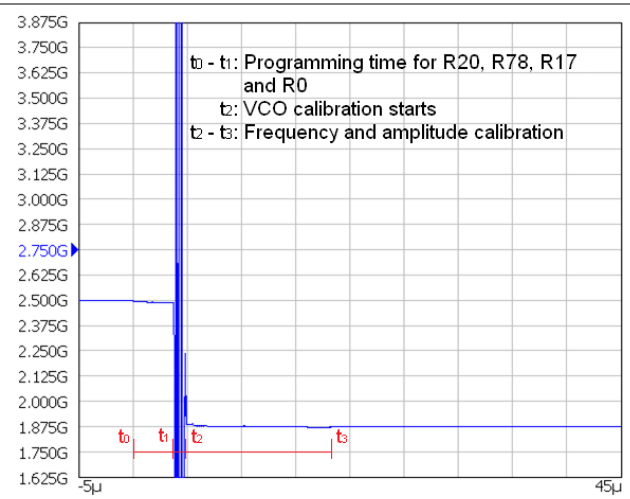


图 3-8. 部分辅助，下跳

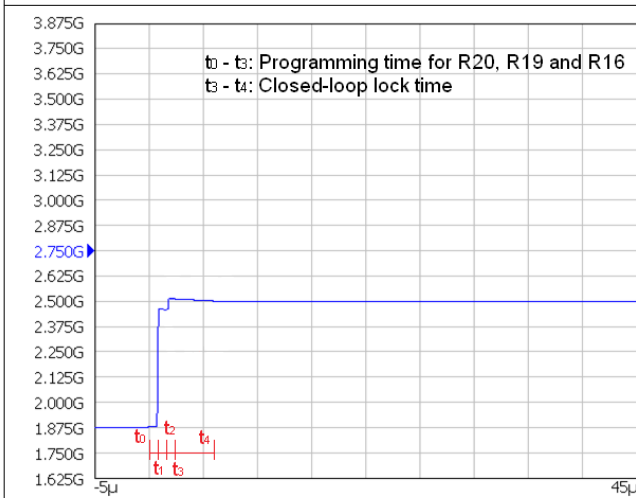


图 3-9. 完全辅助，上跳

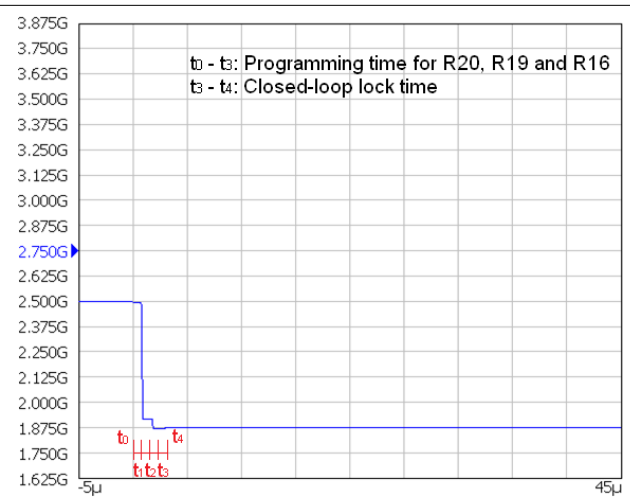


图 3-10. 完全辅助，下跳

4 总结

使用自动校准功能时，校准时间会有所不同，但会毫不费力。为了缩短校准时间，用户可使用本文档中列出的方法来提供特定的校准参数。SPI 编程速度也很重要；为了缩短在校准开始前 VCO 漂移之时的解锁时间，应将 SPI 编程速度设置得足够快。

5 参考文献

以下链接提供了相关产品信息和工具：

- [LMX2594](#)
- [LMX2595](#)
- [LMX2615-SP](#)
- [TICS Pro 软件](#)
- [PLLatinum 仿真器工具 \(PLL Sim\)](#)

6 修订历史记录

注：以前版本的页码可能与当前版本的页码不同

Changes from Revision * (January 2020) to Revision A (July 2021)

Page

- 更新了整个文档中的表格、图和交叉参考的编号格式。..... 3

重要声明和免责声明

TI 提供技术和可靠性数据 (包括数据表)、设计资源 (包括参考设计)、应用或其他设计建议、网络工具、安全信息和其他资源, 不保证没有瑕疵且不做任何明示或暗示的担保, 包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任: (1) 针对您的应用选择合适的 TI 产品, (2) 设计、验证并测试您的应用, (3) 确保您的应用满足相应标准以及任何其他安全、安保或其他要求。这些资源如有变更, 恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务, TI 对此概不负责。

TI 提供的产品受 TI 的销售条款 (<https://www.ti.com/legal/termsofsale.html>) 或 [ti.com](https://www.ti.com) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

邮寄地址: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2021, 德州仪器 (TI) 公司

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2022，德州仪器 (TI) 公司