

Design Guide: TIDM-02013

采用 GaN 的 7.4kW 电动汽车/混合动力电动汽车双向车载充电器参考设计



说明

PMP22650 参考设计是一款 7.4kW 的双向车载充电器。该设计采用两相图腾柱 PFC 和带有有源同步整流功能的全桥 CLLLC 转换器。CLLLC 使用频率和相位调制在宽电压范围内调节输出。该设计使用单个 TMS320F280039C 微控制器来控制 PFC 和直流/直流级。TMS320F28P65x 微控制器也支持该设计。通过使用高速 GaN 开关 (LMG3522-Q1) 实现了高密度。峰值系统效率为 96.5%，该数值在 3.8kW/L 开放式框架功率密度下实现。

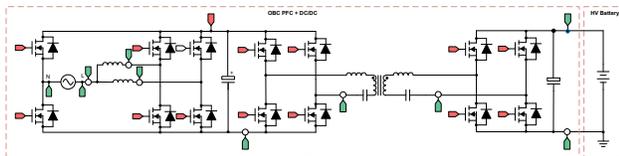
该设计演示了如何在闭合电压和闭合电流环路模式中使用单个 C2000™ MCU 控制此电源拓扑。采用此设计的硬件和软件可帮助您缩短产品上市时间。

资源

TIDM-02013、PMP22650	设计文件夹
TMS320F280039C、TMS320F28P650DK	产品文件夹
AMC3330-Q1、AMC3302-Q1、UCC21222-Q1	产品文件夹
C2000WARE-DIGITAL-POWERSDK	软件文件夹
TMDSCNCD280039C、TMDSCNCD28P65X	工具文件夹



请咨询我司 TI E2E™ 支持专家



特性

- 最大功率：7.4kW，96.5% 峰值效率
- VAC 90 - 264V 交流：240V 交流典型值
- Vprim：400V 直流典型值；Vsec：250 - 450V 直流
- 具有 500kHz 标称值 PWM 开关 (范围为 200kHz - 800kHz) 的 CLLLC 谐振回路可以实现更高的功率密度
- 软开关在初级侧使用零电压开关 (ZVS)；在次级侧使用零电流开关 (ZCS) 和 ZVS，可实现更高的效率
- 使用 Rogowski 线圈传感器实现有源同步整流方案，可实现更高的效率
- 控制律加速器 (CLA) 为 TMS320F28003x 器件提供软件支持，可以实现通过单一 C2000 MCU 控制交流/直流和直流/直流的集成式 OBC 设计。
- 为 TMS320F28P65x 器件提供软件支持，其中一个 CPU (CPU1) 控制交流/直流和直流/直流级。
- 利用 TMS320F28P65x 的全新硬件过采样特性减少了 CPU 开销

应用

- 混合动力、电动和动力总成系统
- 直流快速充电站
- 电源转换系统 (PCS)



1 CLLLC 系统说明

车载充电器 (OBC) 是电动汽车 (EV) 和混合动力电动汽车 (HEV) 必不可少的组成部分。OBC 通常由交流/直流 [功率因数校正 (PFC) 整流器级] 和隔离式直流/直流转换器组成, 如图 1-1 所示。C2000 MCU 旨在实现汽车应用所需的高级数字电源控制; 有关更多信息, 请参阅 C2000 数字电源和 C2000 电动汽车。



图 1-1. 典型的 OBC 架构

对于大多数电动汽车 1 级和 2 级充电器而言, 强烈需要能够在夜间将电池充满电。随着电池容量的增加, OBC 需要设计为具有更高的功率。随着 OBC 功率容量的增加, 由于汽车的空间和冷却能力有限, 功率密度和效率等规格变得更加重要。

CLLLC (电容器-电感器-电感器-电感器-电容器) 具有对称谐振回路、软开关特性以及以更高的频率进行开关的能力, 是这些应用的不错选择。在该设计中展示了 CLLLC 拓扑的控制和实现, 如图 1-2 所示。

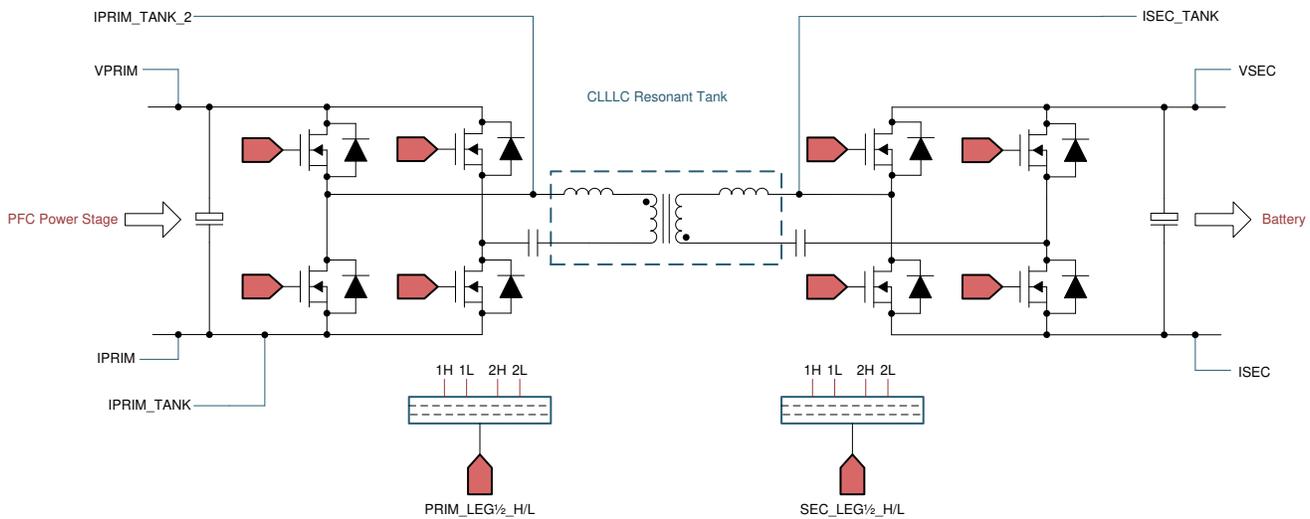


图 1-2. 适用于隔离式直流/直流转换器的 CLLLC 拓扑

图 1-2 的命名规则如下:

VPRIM	初级侧电压 (通常来自 PFC 转换器)
IPRIM	初级侧的返回电流, 可用于保护和监测。
IPRIM_TANK、IPRIM_TANK_2	初级侧的谐振回路电流, 有两种检测方法, 一种是使用分流电流检测, 另一种是使用罗氏线圈。仅需一个, 用于实现反向同步整流 (例如次级到初级)。也用于提供保护。
VSEC	次级侧电压 (通常为电池)
ISEC	次级侧的返回电流, 用于实现电池电流控制环路。
ISEC_TANK	次级侧的谐振回路电流, 用于实现正向功率流的同步整流 (例如初级到次级)。
PRIM_LEG1/2_H/L	初级侧全桥的 PWM
SEC_LEG1/2_H/L	次级侧全桥的 PWM

1.1 主要系统规格

表 1-1 列出了 CLLLC 参考设计电源规格。

表 1-1. 主要系统规格

参数	规格
初级电压 (V _{prim})	400V - 450V 直流 (平均值)
次级电压 (V _{sec})	250V - 450V 直流 最大值
正向额定功率	7.4kW
输出电流 (I _{out})	20 A 最大值
效率 (CLLLC)	峰值 98%
PWM 开关频率	500kHz 标称值 (200kHz - 800kHz 范围)



警告

TI 建议, 该参考设计仅可在 **实验室环境中运行, 不可作为成品** 供一般消费者使用。该设计旨在环境室温下运行, 未在其他环境温度下进行运行测试。

TI 建议, 该参考设计仅可由 **熟悉处理高压电子和机械部件、系统及子系统所存在相关风险的合格工程师和技术人员** 使用。

电路板中存在可接触到的高电压。如电路板的电压和电流处理不当或施加不正确, 则可能导致电击、火灾或伤害事故。使用该设备时应特别小心, 并采取相应的保护措施, 以避免伤害自己或损坏财产。



小心

请勿在无人照看的情况下使该设计通电。



警告

高电压! 电路板中存在可接触到的高电压。可能发生电击。如电路板的电压和电流处理不当, 则可能导致电击、火灾或伤害事故。使用该设备时应特别小心, 并采取相应的保护措施, 以避免伤害自己或损坏财产。为安全起见, 强烈建议使用具有过压和过流保护功能的隔离式测试设备。

TI 认为在对电路板通电或进行仿真之前, 用户有责任确认其已明确并理解电压和隔离要求。通电后, 请勿触摸该设计或与该设计相连的元件。



警告

表面高温! 接触可致烫伤。请勿触摸!

电路板上电后, 某些元件可能会达到 55°C 以上的高温。由于存在高温, 在运行过程中或运行刚结束时, 用户不得触摸电路板。

2 CLLLC 系统概述

2.1 方框图

图 2-1 展示了 CLLLC 拓扑的方框图。

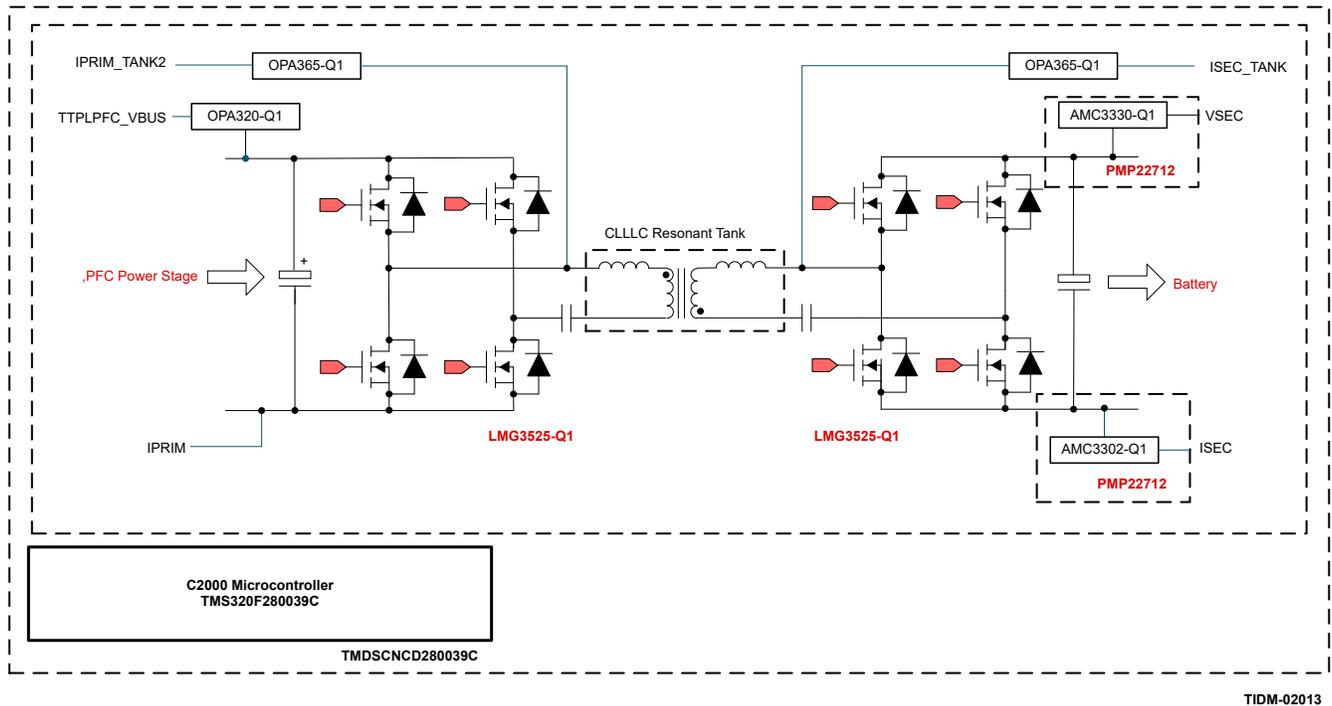


图 2-1. CLLLC 方框图

该参考设计使用以下 EVM 和 PMP 设计来实现本指南中所述的运行：

1. OBC 基板：PMP22650 (带 PMP22712 + PMP22773 子卡)
2. F280039C controlCARD 评估模块：TMDSCNCD280039C

以下各节讨论硬件、软件和系统设计详细信息。

2.2 设计注意事项和系统设计原理

LLC 转换器由于能够在初级侧实现 ZVS 和在次级侧实现 ZCS 而广受欢迎。图 2-2 展示了一个典型的 LLC 串联谐振转换器 (SRC)。该转换器的初级侧为半桥；因此，从伏秒角度来看，变压器利用率为一半。此外，与使用全桥结构时相比，开关的额定电流是所需额定电流的两倍。

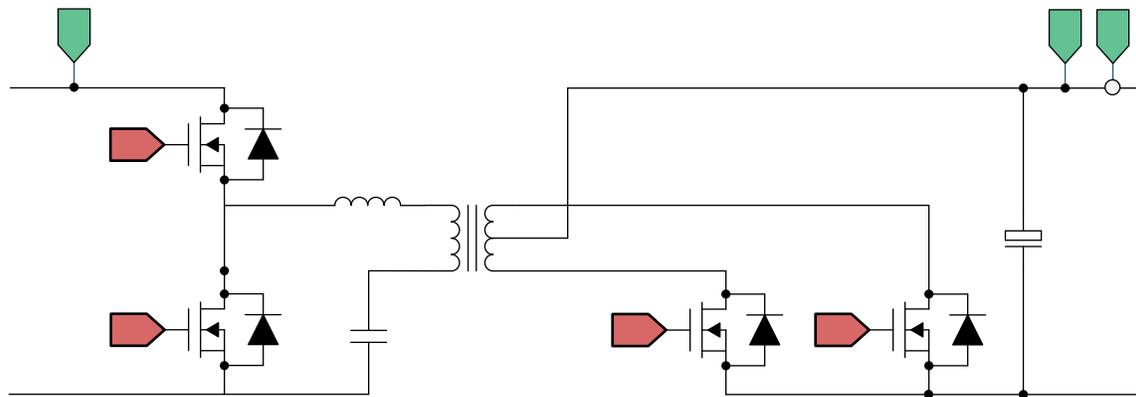


图 2-2. LLC 半桥 SRC

尽管出于成本原因，半桥 LLC SRC 在较低功耗方面很有吸引力，但高功率和高密度应用仍需要使用全桥 LLC SRC，原因如下：

1. 全桥 LLC 转换器可以更好地利用变压器次级侧和初级侧的磁芯；因此，该转换器能够提供更高的功率密度。
2. 全桥 LLC 转换器降低了额定电流，从而降低了导线中铜的成本。利用该转换器，还可以使用相同的铜线实现更高的功率（与半桥 SRC 相比）。

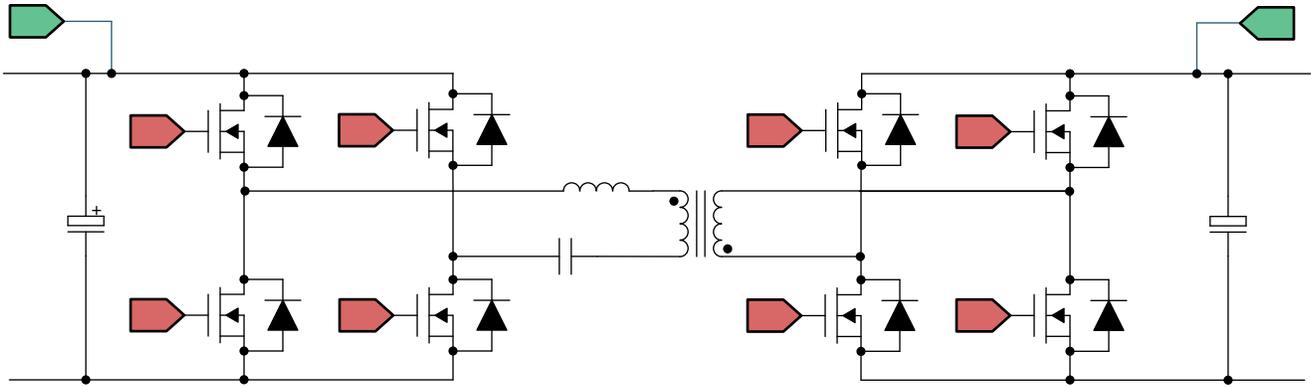


图 2-3. 全桥 LLC 转换器

如图 2-3 所示，全桥 LLC 转换器属于广泛的双有源电桥 (DAB) 转换器类别。在 DAB 转换器下，转换器可根据型号或操作进行分类：

1. 相移 DAB 转换器是最流行的转换器之一。
2. 谐振 DAB 转换器在谐振回路上具有不同的变体 (LC、LLC、CLLC、CLLLC 等)。

谐振 DAB 转换器之所以备受关注，是因为此类转换器可实现高效率、高功率和高密度。CLLLC 具有对称谐振回路，能够双向运行。使用 LLC 结构进行双向使用的问题在于，当在反向功率流模式下运行时，开关频率取决于变压器绕组电容和漏电感。这对功率级增益和开关频率几乎没有控制，或者完全没有控制。因此，优先考虑使用 CLLLC 类型的结构，因为该结构可以更好地控制开关频率并为增益提供更高的自由度。

2.2.1 谐振回路设计

本节根据所需的电压增益、软开关特性讨论 CLLLC 的谐振回路参数选择，并根据 CLLLC 为充电器选择适当的功率分布。

有关其他计算和信息，请参阅软件安装包中位于 C2000Ware_DigitalPower_SDK_<ver>/solution/tidm_02013/hardware/ 的以下文件

2.2.1.1 电压增益

为了了解谐振回路设计，首先必须使用一次谐波近似计算通过一次谐波分析 (FHA) 来分析电池充电模式和反向功率流模式的增益。图 2-4 提供了谐振回路的简化图。

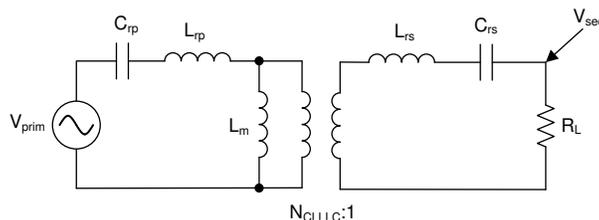


图 2-4. 电池充电模式 (BCM) 期间 CLLLC 谐振回路的 FHA 模型

图 2-4 的命名规则如下：

V_{prim} (TTPLPFC_VBUS)	初级侧的电压输入
L_{rp}	初级侧谐振电感器
C_{rp}	初级侧谐振电容器
N_{CLLLC}	变压器的匝数比
L_{m}	磁化电感器
V_{sec}	次级侧的电压输出
L_{rs}	次级侧谐振电感器
C_{rs}	次级侧谐振电容器
R_{L}	在次级输出端使用 FHA 时看到的有效负载

请注意，此处的有效 R_{L} 计算为 $R_{\text{L}} = \left(\frac{8}{\pi^2}\right) R_{\text{L_dc}}$ 其中 $R_{\text{L_dc}}$ 是输出端的直流阻性负载。

初级侧以次级侧量为基准，

- L_{rs}' 等于 $L_{\text{rs}} * N_{\text{CLLLC}} * N_{\text{CLLLC}}$
- C_{rs}' 等于 $C_{\text{rs}} / (N_{\text{CLLLC}} * N_{\text{CLLLC}})$
- R_{L}' 等于 $R_{\text{L}} * (N_{\text{CLLLC}} * N_{\text{CLLLC}})$
- V_{rs}' 等于 $V_{\text{rs}} * N_{\text{CLLLC}}$

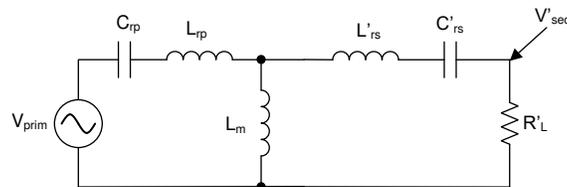


图 2-5. FHA CLLLC，量以 BCM 中的初级侧为基准

使用 KCL 和 KVL，增益公式可以写为方程式 1。

$$\frac{V_{\text{sec}}}{V_{\text{prim}}} = \frac{\left[Z_{\text{m}} \parallel \left(Z_{\text{rs}}' + R_{\text{L}}' \right) \right] R_{\text{L}}'}{\left(Z_{\text{rp}} + \left[Z_{\text{m}} \parallel \left(Z_{\text{rs}}' + R_{\text{L}}' \right) \right] \right) \left(Z_{\text{rs}}' + R_{\text{L}}' \right) N_{\text{CLLLC}}} \quad (1)$$

同样，对于反向功率流，电路可以进行简化，如图 2-6 所示，而增益可以写为方程式 2。

$$\frac{V_{\text{prim}}}{V_{\text{sec}}} = \frac{N_{\text{CLLLC}} \left[Z_{\text{m}} \parallel \left(Z_{\text{rp}} + R_{\text{L}} \right) \right] R_{\text{L}}}{\left(Z_{\text{rs}}' + \left[Z_{\text{m}} \parallel \left(Z_{\text{rp}} + R_{\text{L}} \right) \right] \right) \left(Z_{\text{rp}} + R_{\text{L}} \right)} \quad (2)$$

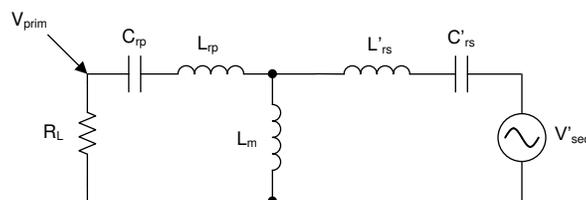


图 2-6. RCM 中用于计算增益的 FHA 模型

在下一节中，使用 [方程式 1](#) 和 [方程式 2](#) 根据为设计选择的参数来研究电压增益。

2.2.1.2 变压器增益比设计 (N_{CLLLC})

在以谐振频率或接近谐振频率运行时，谐振转换器通常具有最高的效率。由于这是双向电池充电器，因此该设计需要涵盖一定范围的输出电压。这意味着在选择 n 时应使转换器在尽可能低的电流下运行，从而帮助降低 I^2R 损耗。根据该设计概念，在需要提供全功率的最低输出电压下将出现最高输出电流。此时我们将设置 n ，使转换器以尽可能接近谐振的方式运行。在此设计中，可以计算出其匝数比为 1.1:1。这样可以实现最低的损耗，同时仍支持宽输出电压范围。

2.2.1.3 磁化电感选择 (L_m)

为了确保初级侧 FET 的 ZVS 运行，我们需要确保谐振回路中存储的能量大于 FET 输出电容器中存储的能量。我们可以使用 [方程式 3](#) 来确定全桥 LLC SRC 所需的 L_m 。

$$L_m \leq \frac{T_{\text{dead}}}{16 * C_{\text{OSS}}} \quad (3)$$

其中转换器的预期开关频率为 500kHz，因此 $T = 1/(500 * 10^3)$ 并基于功率器件。也可以在功率器件数据表中找到选择的参数（例如 t_{dead} 和 C_{OSS} ）。通常，必须使用曲线拟合来计算有效 C_{OSS} 。在该设计中，根据讨论的设计参数， L_m 必须小于 20 μH 。除了上述计算中考虑的因素之外，实际变压器中还存在绕组间电容，需要通过谐振回路电流对其进行放电。因此，使用仿真选择了 14 μH 值来确保转换器工作范围内的 ZVS；该值用于后续的选择过程。

2.2.1.4 谐振电感器和电容器选择 (L_{rp} 和 C_{rp})

在选择 L_{rp} 时， L_m 与 L_{rp} 之比被广泛用作设计参数，

$$L_n = \frac{L_m}{L_{rp}} \quad (4)$$

在选择 L_n 值时，该值应确保谐振回路中的电压增益在转换器的工作范围内是足够的。在该设计中，由于输入电压来自 PFC 级并且估计具有 10% 的纹波，因此需要至少 10% 的增益变化。考虑到这一标准以及 L_n 应保持较高以降低电感器感值（从而降低损耗）的事实，根据 L_n 随负载变化的 FHA 图，该设计选择 L_n 等于 14（请参阅 [图 2-7](#)）。

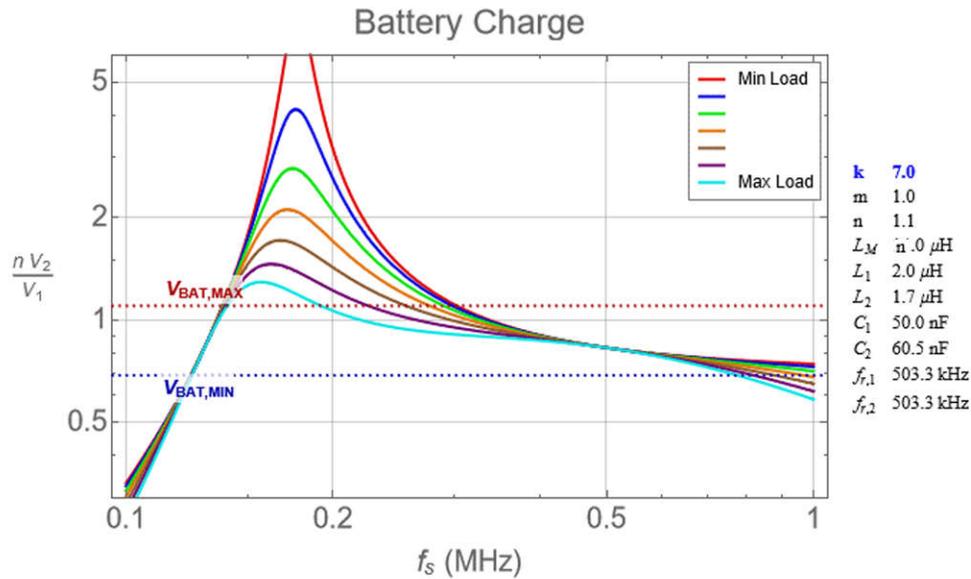


图 2-7. CLLLC 谐振回路增益随 L_n 变化的变化情况

现在已经选择了 L_n ，可以使用方程式 4 来计算 L_{rp} 。 L_{rp} 和 C_{rp} 决定了转换器的串联谐振频率，其关系如方程式 5 所示。

$$f_{\text{res}} = \frac{1}{2\pi\sqrt{L_{rp}C_{rp}}} \quad (5)$$

然后可以使用方程式 5 来计算设计所需的 C_{rp} 。但是、由于元件可用性，设计中使用了下一个最接近的 C_{rp} 值。在使用这些元件值的情况下，BCM 增益如图 2-7 所示。

在图 2-7 中，随着负载增加（即 R_{L_dc} 变小），增益曲线在串联谐振频率以下的区域中变得非单调。这可能导致初级 FET 上的 ZVS 丧失，更严重的是失去控制。因此，假设在标称 V_{out} 下具有最大负载，则负载被限制或钳位到 $R_{L_dc} = 30 \Omega$ ，其增益是单调的（请参阅图 2-7）。

此外，图 2-7 表明，在 BCM 中，我们在 200kHz 至 800kHz 的工作频率范围内有足够的增益来覆盖所有工作条件。最后，值得注意的是，如果 PFC 纹波能够降低，则总体预期输入范围也会减小。这会导致所需的增益范围减小，并最终帮助减少支持所有负载条件所需的频率变化。

2.2.2 电流和电压检测

以下各节讨论该设计中不同电流和电压的检测方案。设计中实现了多种方案，用户可以根据其应用需求选择合适的方案。

2.2.2.1 VPRIM 电压检测

C2000 MCU 在初级侧进行偏置；因此，通过连接到电路板接地端的电阻分压器来检测初级电压。由于使用了过采样，电压跟随器布局中的运算放大器用于缓冲 ADC 的信号，如图 2-8 所示。缓冲器有助于降低 ADC 看到的阻抗，因此可以使用更快的采样速率。否则，采样将受到电阻分压器电阻时间常数的限制，该时间常数通常很高，因此只能进行慢速采样。

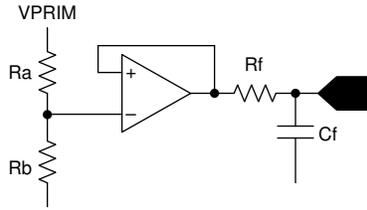


图 2-8. VPRIM 电压检测电路

2.2.2.2 VSEC 电压检测

使用 AMC3330-Q1 以隔离方式检测次级侧电流，如图 2-9 所示。

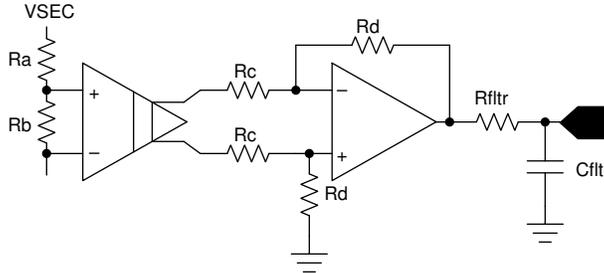


图 2-9. VSEC 电压检测电路

2.2.2.3 ISEC 电流检测

还使用 AMC3302-Q1 以隔离方式检测次级侧输出电流，如图 2-10 所示。

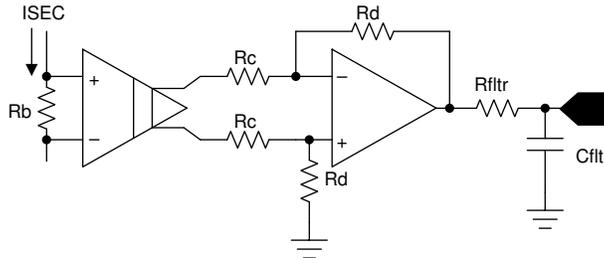


图 2-10. ISEC 电流检测电路

2.2.2.4 ISEC 谐振回路和 IPRIM 谐振回路

我们选择了基于罗氏线圈的检测机制，以隔离方式检测初级侧和次级侧谐振回路中的高频电流，如图 2-11 所示。ADC 引脚在内部连接到比较器子系统 (CMPSS)，该子系统可以生成正确的脉冲，这些脉冲通过 X-Bar 到达 PWM，以获得同步整流所需的操作。

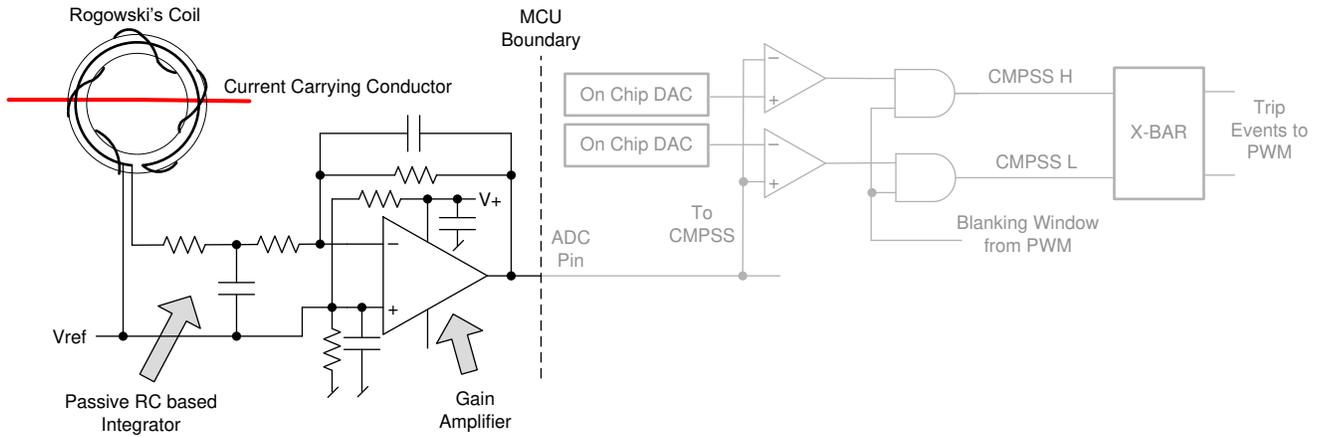


图 2-11. 使用罗氏线圈的 ISEC 谐振回路电流检测

2.2.2.5 IPRIM 电流检测

使用 LMV796-Q1 检测初级侧电流 IPRIM。(请参阅图 2-12)。

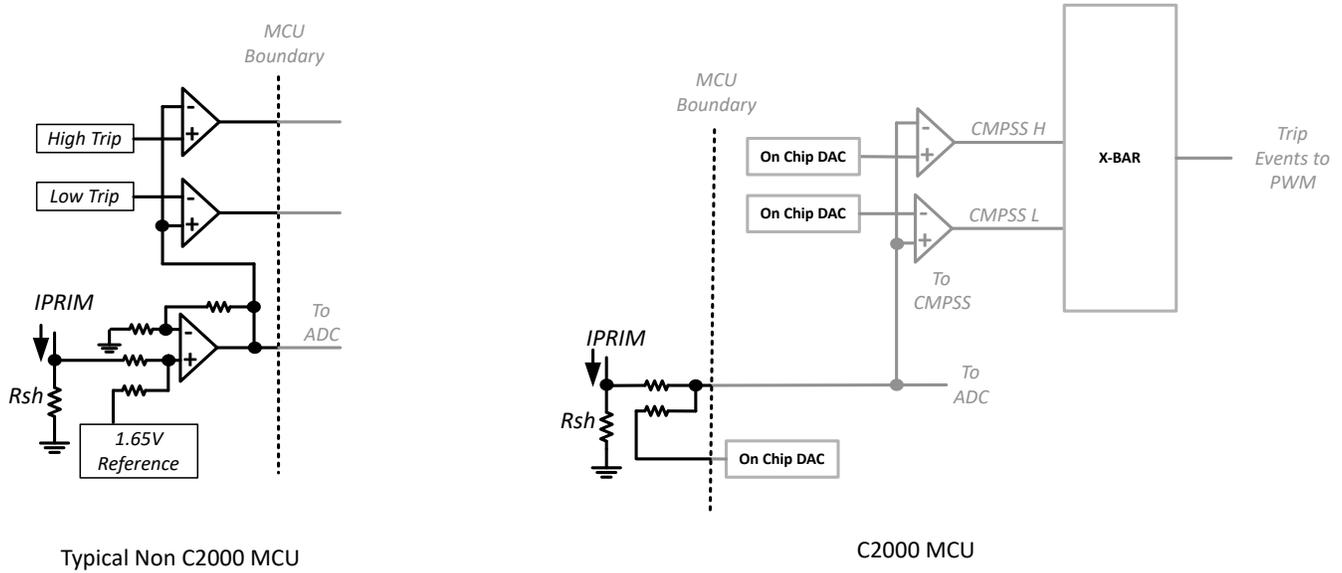


图 2-12. IPRIM 电流检测电路，典型 MCU 与 C2000 MCU 的比较

2.2.2.6 保护 (CMPSS 和 X-Bar)

大多数电力电子转换器需要过流事件保护。对于该设计，需要多个比较器，并且需要为跳闸点生成基准。通过使用在内部连接到 PWM 模块的 C2000 MCU (例如 TMS320F280039，该器件具有作为比较器子系统 (CMPSS) 一部分的片上窗口比较器以及用于跳闸设定点的 12 位 DAC)，无需外部硬件即可实现 PWM 的快速跳闸。这样可以在最终应用中节省布板空间，因为可以通过使用片上资源 (例如 DAC、比较器和 ADC) 避免额外的元件。所有这些资源都可以同时使用，无需任何额外的外部连接。此外，CMPSS 生成的信号进入 X-Bar，在此处这些信号能够以不同且独特的方式组合，以标记来自多个来源的独特跳闸事件。

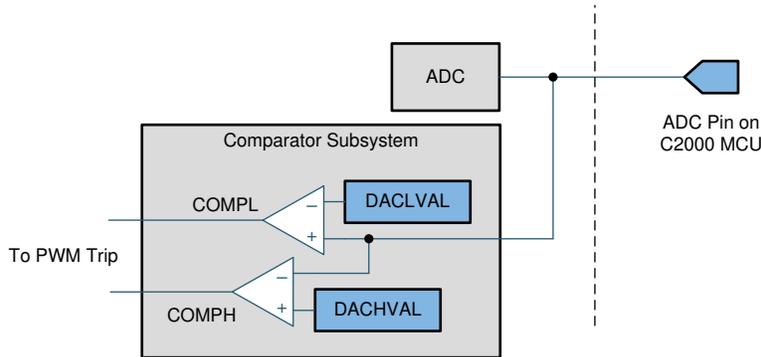


图 2-13. 用于过流保护的比较器子系统 (CMPSS)

2.2.3 PWM 调制

图 2-14 展示了该设计中使用的 PWM 波形配置。

为初级桥臂和次级桥臂使用了高分辨率 PWM。使用了向上/向下计数模式来生成 PWM。为了使用高分辨率 PWM，PRIM_LEG1_H PWM 脉冲以周期事件为中心，并且时基配置为上-下计数模式。然后，为互补开关生成具有高分辨率死区时间的互补脉冲。在桥臂 1 与桥臂 2 之间，存在 180 度的相移用于全桥运行。这是通过使用 PWM 模块上的特性交换 xA 和 xB 输出来实现的。（或者，也可以实现相移，但该设计不需要。）

发送到次级侧的 PWM 脉冲通过隔离器，这会增加额外的传播延迟。为了解决该传播延迟，需要稍微提前 PWM。这是以相对于初级有效 PWM 脉冲下降沿的相移延迟形式实现的。次级侧的相移是隔离器所需的周期和延迟的组合，如图 2-14 所示。由于使用了有源同步整流方案，上升沿由初级侧 PWM 开关时序控制。由于开关事件可能有很大的噪声，因此使用了消隐窗口。次级谐振回路中的电流可能是不连续的，具体取决于工作频率和负载。因此，下降沿由次级电流一达到零就触发的跳闸操作控制。然后，跳闸一直被锁存到下一个零或周期事件，以避免由于噪声而导致次级侧开关出现任何虚假导通。消隐脉冲由 PWM 时基生成，但跳闸锁存和消隐操作作为 CMPSS 的一部分发生。根据是谐振回路电流的正半部分还是负半部分，会生成两个不同的跳闸信号并通过 X-Bar 将其发送到 PWM 模块。C2000 MCU 上的 4 类 PWM 可以唯一地使用这些事件在向上计数期间触发 xA 脉冲，在向下计数期间触发 xB 脉冲。有关详细信息，请参阅函数 CLLLC_HAL_setupSynchronousRectificationAction() 中的代码，该函数是该解决方案的 HAL 文件，请参阅节 5.1.2。

Type-4 PWM 上的全局链接机制用于减少更新寄存器并启用高频运行所需的周期数。例如，CLLLC_HAL_setupPWM() 函数的以下代码会链接所有 PWM 桥臂的 TBPRD 寄存器。借助该链接，对 PRIM_LEG1 TBPRD 寄存器的单次写入会将相应的值写入 PRIM_LEF2、SEC_LEG1 和 SEC_LEG2 中。

```
EPWM_setupEPWMLinks(CLLLC_PRIM_LEG2_PWM_BASE,
                    EPWM_LINK_WITH_EPWM_1,
                    EPWM_LINK_TBPRD);
```

```
EPWM_setupEPWMLinks(CLLLC_SEC_LEG1_PWM_BASE,
                    EPWM_LINK_WITH_EPWM_1,
                    EPWM_LINK_TBPRD);
```

```
EPWM_setupEPWMLinks(CLLLC_SEC_LEG2_PWM_BASE,
                    EPWM_LINK_WITH_EPWM_1,
                    EPWM_LINK_TBPRD);
```

高分辨率 PWM 依赖于将上一个周期的余数计算结果结转到下一个周期；因此，不应在初级侧和次级侧 PWM 之间使用周期性同步来维持相位关系。每当检测到频率变化或占空比变化时，就会使用快速中断服务例程发 (ISR1，请参阅节 5.1.2.2) 出一次性同步。

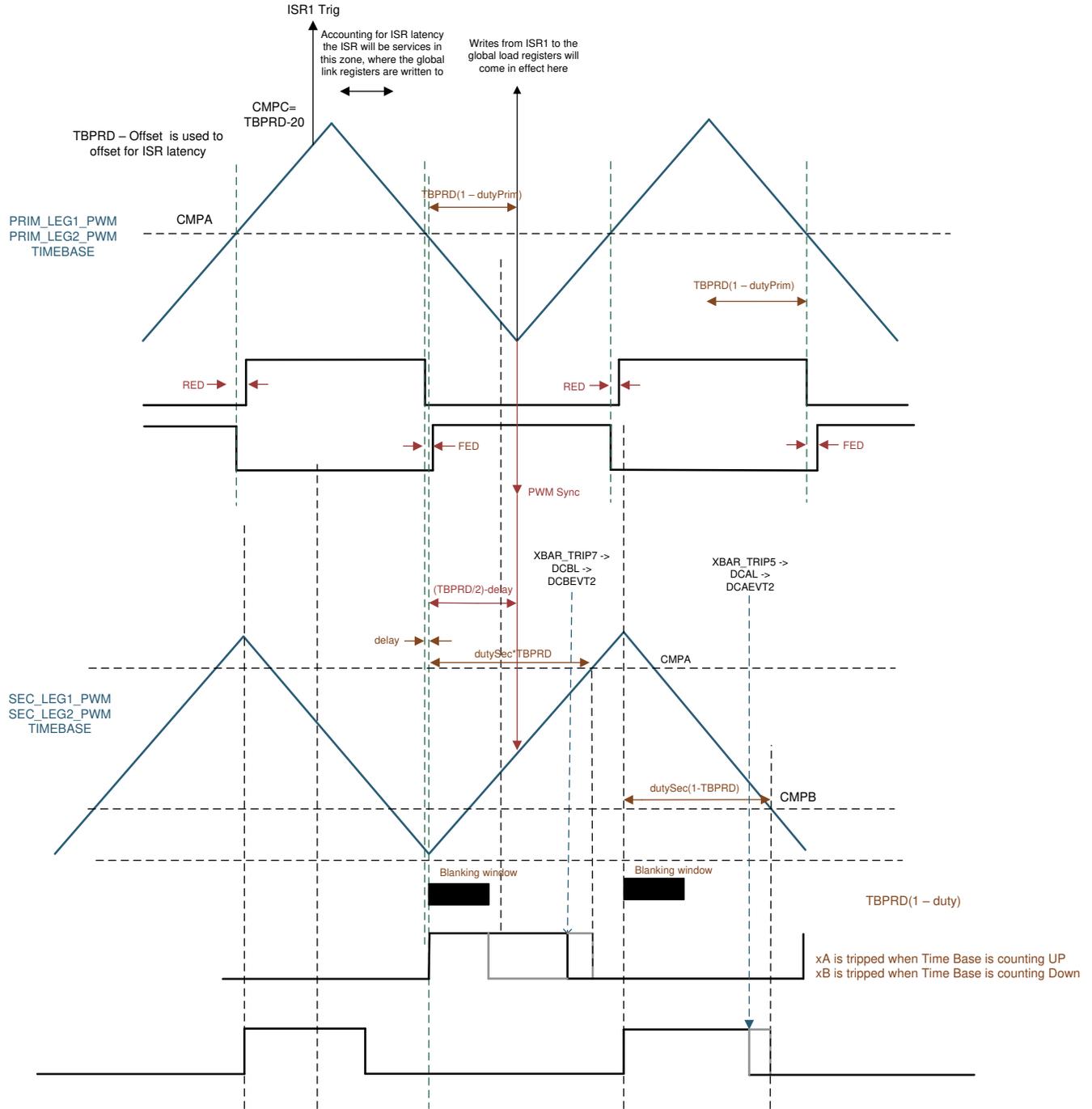


图 2-14. CLLLC 设计中使用的 PWM 方案，具有初级到次级功率流的有源同步整流

同样，对于反向功率流方向，使用的 PWM 配置如图 2-15 所示

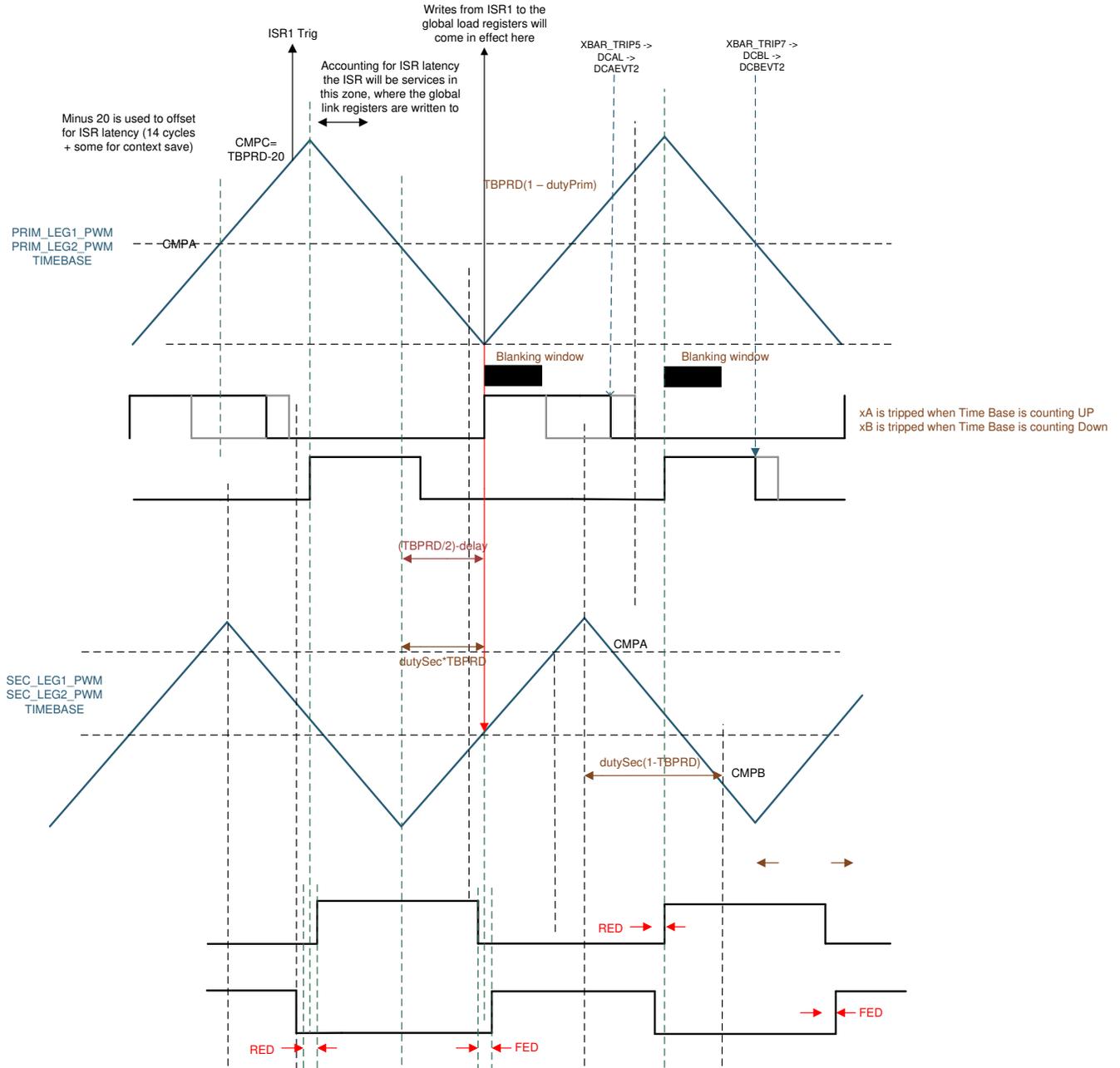


图 2-15. CLLLC 设计中使用的 PWM 方案，具有次级到初级功率流的有源同步整流

3 图腾柱 PFC 系统说明

警告



请勿在无人照看的情况下使该器件通电。



高电压！电路板中存在可接触到的高电压。可能发生电击。如电路板的电压和电流处理不当，则可能导致电击、火灾或伤害事故。使用该设备时应特别小心，并采取相应的保护措施，以避免受伤或损坏财产。为安全起见，强烈建议使用具有过压和过流保护功能的隔离式设备。TI 认为在对电路板通电或进行仿真之前，用户有责任确认其已明确并理解电压和隔离要求。通电后，请勿触摸该设计或与该设计相连的元件。



表面高温！接触可致烫伤。请勿触摸！电路板上电后，某些元件可能会达到 55°C 以上的高温。由于存在高温，在运行过程中或运行刚结束时，用户不得触摸电路板。

3.1 图腾柱无桥 PFC 的优势

所有插电式混合动力电动汽车 (PHEV) 需要在电网和车辆内部的高压电池包之间采用车载充电器 (OBC)。必须实施功率因数校正 (PFC) 转换器才能直接连接到电网进行交流/直流电源转换并更大限度地提高流向下游直流/直流转换器的有功功率。

传统的 PFC 转换器实现了无源二极管电桥以进行整流，该技术现在称为无源 PFC 技术。此类方案的优点为：设计简单，可靠性高，系统控制环路速度慢以及成本低。但缺点也很明显：无源器件很重，功率因数低，并且会产生显著的功率损耗，从而导致散热器体积庞大以及散热量大。通过进一步调查发现，在宽电源应用的低压线路上，输入电桥大约消耗输入功率的 2%。如果设计人员可以抑制串联二极管之一，则可以节省输入功率的 1%，从而使效率从 94% 上升至 95% (Turchi ; Dalal ; Wang ; Lenck 2014)。由于上述缺点，桥式传统 PFC 的额定功率被限制在数百瓦以下，尤其是在混合动力电动汽车 (HEV) 或电动汽车 (EV) 中，其中小空间和小重量是关键设计参数。

因此，无桥架构趋势日益明显，这种架构消除了传统的二极管电桥。OBC 基于硅功率器件，存在低效率、低功率密度和高重量等限制。凭借 SiC MOSFET 的优势，设计人员可以利用快速开关、低反向恢复电荷和低 $R_{DS(ON)}$ 的卓越性能，极大地改善这些限制。

图 3-1 展示了图腾柱无桥 PFC 升压整流器的基本结构。该元件包含一个升压电感器、两个高频升压 GaN 或 SiC 开关 (在下图中标记为 SiC_1 和 SiC_2) 以及两个用于在工频下传导电流的元件。工频元件可以是两个慢速二极管，如图 3-1 所示。(A) 侧显示了两个硅二极管 (D_1 和 D_2)。(B) 侧显示使用 Si_1 和 Si_2 可以进一步提高效率。

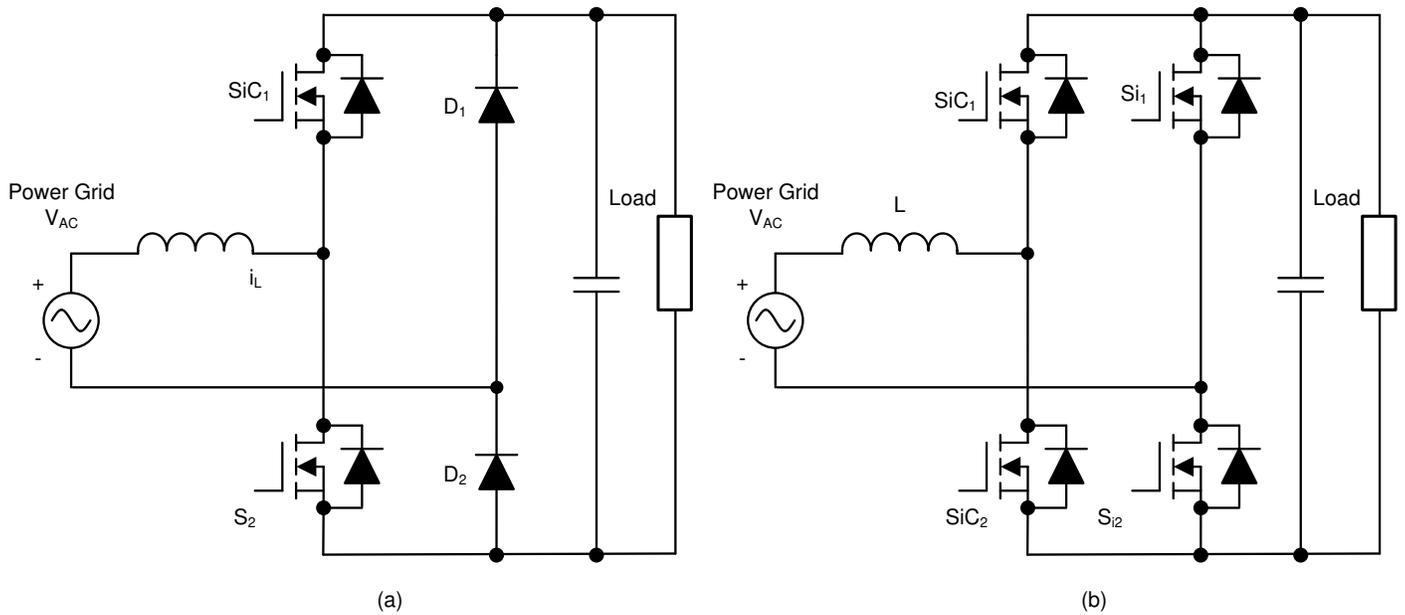


图 3-1. 图腾柱无桥 PFC 升压转换器拓扑：(A) 用于线路整流的二极管 (B) 用于线路整流的 MOSFET

图腾柱 PFC 中的固有问题是交流电压过零处的运行模式转换。当交流输入在过零处从正半线变为负半线时，低侧高频开关 SiC₂ 的占空比从 100% 变为 0%，SiC₁ 的占空比从 0% 变为 100%。由于高侧二极管（或 MOSFET 的体二极管）的反向恢复速度较慢，D₂ 的阴极电压无法立即从接地跳变为直流正电压（这会导致较大的电流尖峰）。由于该问题，设计人员无法在连续导通模式 (CCM) 图腾柱 PFC 中使用 Si MOSFET。因此，SiC₁ 和 SiC₂ 必须是氮化镓 (GaN) 或 SiC MOSFET 场效应晶体管 (FET)，具有低反向恢复，对于 TIDM-02013，我们选择了 GaN FET。

图腾柱 PFC 的最大优势是导电路径中的功率损耗较低。表 3-1 展示了传统 PFC 和图腾柱 PFC 之间的器件比较。

表 3-1. 传统桥式 PFC 和图腾柱无桥 PFC 的器件比较

参数	低频二极管	高频二极管	高频开关	导电路径
传统桥式 PFC	四	一个	一个	两个低速二极管 + 一个开关或 (两个低速二极管 + 一个高速二极管)
图腾柱无桥 PFC	两个	零	两个	一个高速 GaN 开关 + 一个低速 Si (或 SiC) MOSFET

以下列表总结了图腾柱 PFC 的优点：

- 虽然传统 PFC 升压转换器是最常用的拓扑，但其效率受到前端二极管桥式整流器传导损耗的影响，并且无法双向运行。图腾柱 PFC 具有固有的双向运行功能。
- 无桥 PFC 升压转换器极大地减少了二极管数量且提高了功率密度和效率。
- 该 PFC 具有以下优点：高效率、小共模噪声、小交流电流纹波、小反向恢复电流和更少的元件。
- GaN 体二极管的低反向恢复电荷和 GaN FET 的低导通电阻使该转换器成为双向车载充电器的高效且具有成本效益的解决方案。

3.2 图腾柱无桥 PFC 运行

图腾柱 PFC 分别在交流电源输入的正周期和负周期运行，并根据高频 GaN MOSFET 的开关方式确定电流（请分别参阅图 3-2 和图 3-3）。

高频 GaN MOSFET 与电感器一起构成了一个同步模式升压转换器。在正半周期内， S_2 是由占空比 D 驱动的升压开关， S_1 由互补脉宽调制 (PWM) 信号 $(1-D)$ 驱动。图 3-2 (A) 展示了电流的流动方向。同样，在 S_2 以 $1-D$ 开关期间， S_1 以 D 开关；图 3-2 (B) 展示了电流的流动方向。请注意，在该周期内， S_{D2} 连续导通。

在负半周期内，除了高侧和低侧高频开关的作用交换外，运行方式相似。图 3-3 展示了电流的流动方向。请注意，在该周期内， S_{D1} 连续导通。

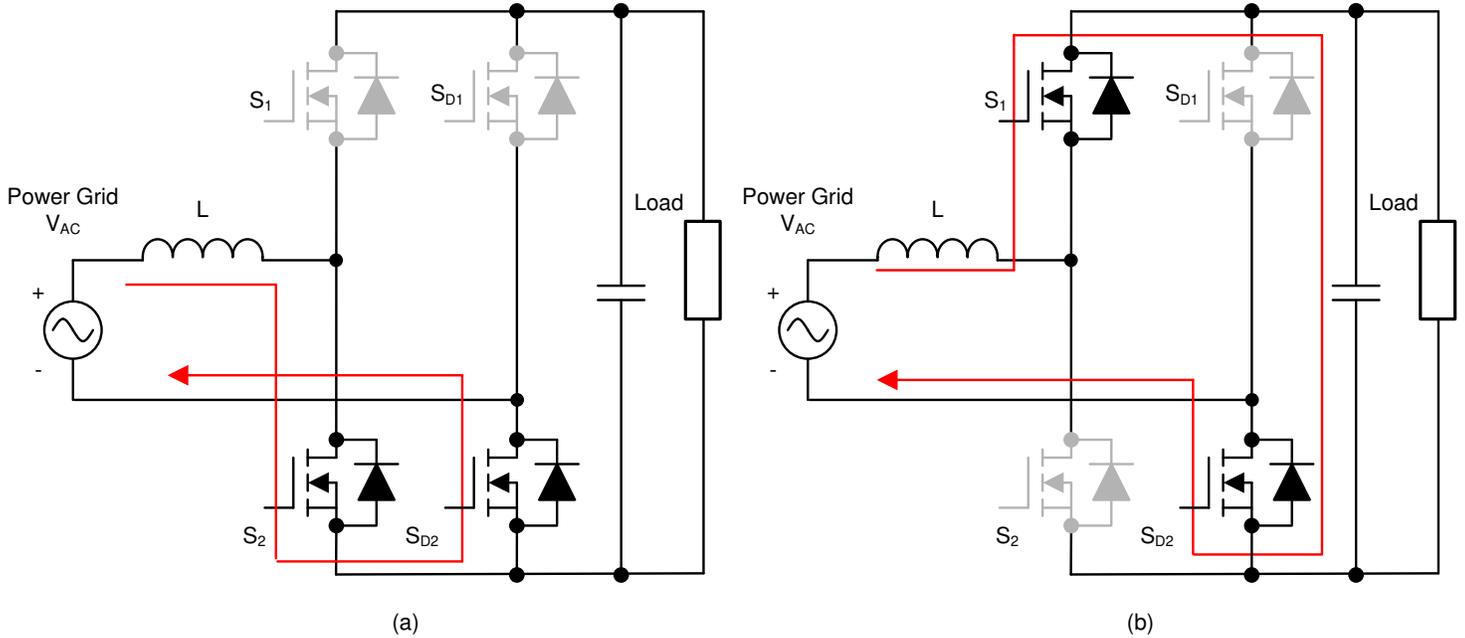


图 3-2. 正半周期内的图腾柱无桥 PFC 运行：(A) S_2 开启时 (B) S_2 关断时

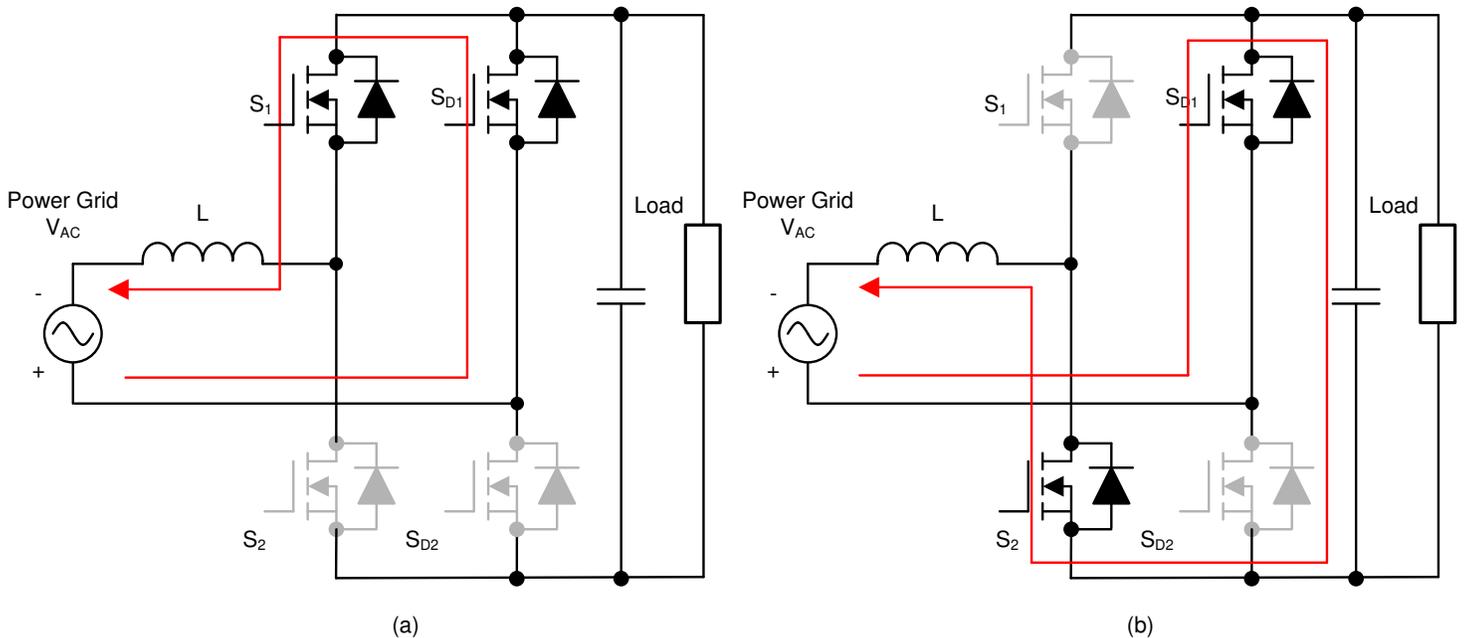


图 3-3. 负半周期内的图腾柱无桥 PFC 运行：(A) S_1 开启时 (B) S_1 关断时

该参考设计使用 GaN FET (LMG3522R030-Q1) 和 TI 的 C2000™ Piccolo™ (TMS320F280039C) 高性能 MCU。高频 GaN FET 在 120kHz 的开关频率下运行，一对 Si MOSFET 在工频 (大约 45Hz 至 60Hz) 下运行。因此，导通路径包括一个 GaN 开关和一个低频 Si 开关，导通损耗显著降低。使用双通道交错技术以降低导通损耗和输入电流纹波。测试结果表明效率高达 98.5% 以上。

3.3 主要系统规格

表 3-2 列出了该设计的主要系统规格。

表 3-2. TIDA-02013 PFC 主要系统规格

参数	规格
输入	<ul style="list-style-type: none"> • 单相 • 电压：大约为 90V AC_{RMS} 至 264V AC_{RMS} • 交流线路频率范围：50Hz 至 60Hz • 输入电流：240V 时为 32A_{RMS_MAX}，120V 时为 32A_{RMS_MAX} • 功率因数：≥ 0.99
输出	<ul style="list-style-type: none"> • PFC 输出：大约为 400V 典型值 • 最大输出功率：大约 400V 时为 7.4kW • 峰值效率：98.5%
性能	<ul style="list-style-type: none"> • 适用于高压锂离子电池 OBC 的 PFC 级 • 开关频率：120kHz • 隔离：增强型 • 输入交流检测 • PFC 输出电压检测
保护	<ul style="list-style-type: none"> • 过热保护 • 短路保护 • 过流保护 • 欠压保护 • 过压保护

3.4 系统概述

3.4.1 方框图

图 3-4 展示了 TIDM-02013 参考设计的系统方框图，其中包含以下元件。

- 电源开关 G1-G4 是高频 GaN MOSFET，每个半桥桥臂之间存在 180° 相移。G5 和 G6 形成低频 (40 至 60Hz) 同步整流器桥，几乎没有开关损耗；这两种器件需要具有低导通损耗特性。
- TMS320F280039C C2000 实时微控制器充当控制器，具有所有电压和电流传感器输入，为 G1-G6 生成正确的 PWM 信号。该控制器还从栅极驱动器板读取任何故障信号，并在发生故障时关闭系统。在启动期间或故障清除时，会使用复位功能。
- 霍尔传感器用于检测总输入电流和每个通道的电流。分压器用于检测输入线电压和中性点电压以及输出直流总线电压。

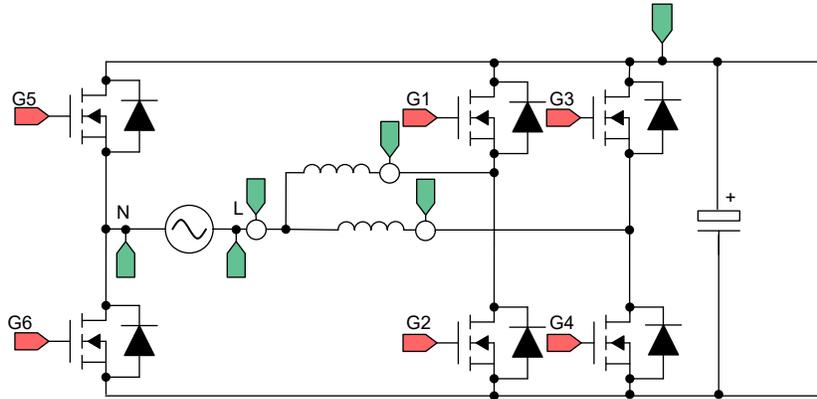


图 3-4. TIDM-02013 PFC 方框图

3.5 系统设计原理

3.5.1 PWM

图 3-5 所示为交错式 TTPL PFC 拓扑的简单单相图。要控制该整流器，需要对占空比进行控制，以直接调节电压。如果将软件变量 Duty 或 D 设置为当其等于 1 时 Q3 始终导通，并且该设置使电压 V_{xiN} 等于 V_{bus} 电压，则可以进行此调节。当 Duty 被设置为 0 时，Q3 从不开启，并且 Q4 始终连接到直流总线负极，从而使电压变为 0。

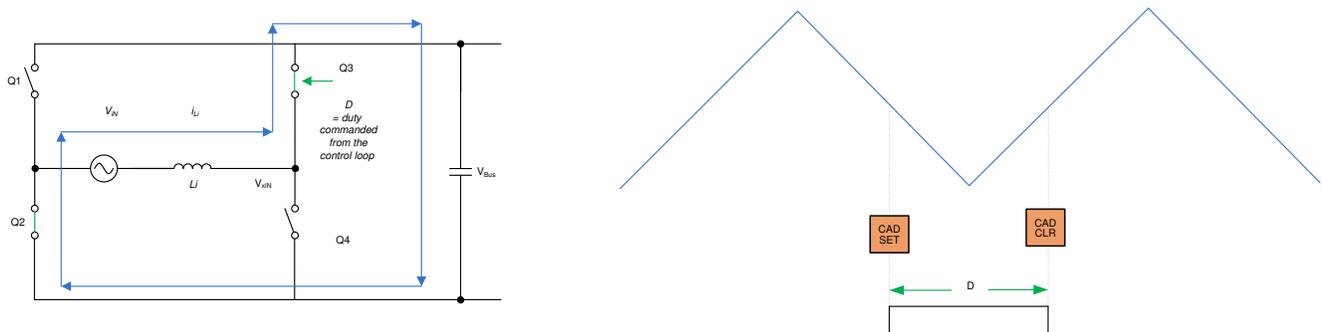


图 3-5. TTPL PFC 的单相图

3.5.2 电流环路模型

要了解电流环路模型，应首先仔细查看电感器电流。在图 3-5 中，为连接到开关 Q3 和 Q4 的 PWM 调制器提供了占空比 (D)。在这里，方程式 6 表达为：

$$V_{xiN} = D \times V_{bus} \quad (6)$$

备注

当 D 设置为 1 时，Q3 始终开启，而当 D 设置为 0 时，Q3 始终关断。

要调制流经电感器的电流，应使用 Q3 和 Q4 开关的占空比控制调节电压 V_{xiN} 。假设电流的方向沿从交流线路到整流器的方向为正并使用直流母线前馈和输入交流电压前馈，同时假设电网的阻抗相当小。图 3-6 所示为简化的电流环路，电流环路受控体模型表达为 方程式 7。

$$H_{p_i} = \frac{i_{Li}^*}{D} = \frac{1}{K_v_gain} \times K_{i_gain} \times G_d \times \frac{1}{Z_i} \quad (7)$$

其中，

- K_v_gain 是检测到的最大总线电压的倒数 $\frac{1}{V_{busMaxSense}}$
- K_{i_gain} 为感测到的最大交流电压的倒数， $\frac{1}{I_{AC_MaxSense}}$
- K_{i_fltr} 为从电流传感器连接到 ADC 引脚的 RC 滤波器的响应
- G_d 为与 PWM 更新相关的数字延迟，数字控制是当前命令
- i_{Li}^* 为当前命令

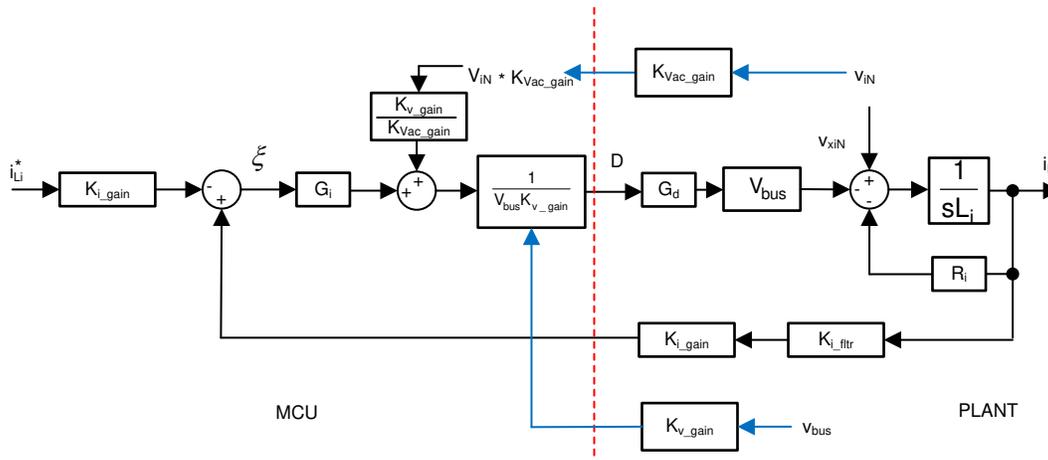


图 3-6. 电流环路控制模型

备注

由于电流环路被视为对电压 V_{xiN} 进行调节，因此在基准上使用了负号。若要增大电流，必须降低 V_{xiN} ，因此在图 3-6 中，对基准电压和电压反馈标记了“+”。

该电流环路模型用于设计电流补偿器。该电流环路使用了一个简单比例积分控制器。

对于两个交错相位，为每个桥臂提供相同的占空比，因此电流仅增加了两倍。因此，受控体模型表达为 方程式 8。

$$H_{p_i} = \frac{i_{Li}^*}{D} = 2 \times \frac{1}{K_v_gain} \times K_{i_gain} \times K_{i_fltr} \times G_d \times \frac{1}{Z_i} \quad (8)$$

3.5.3 直流母线调节环路

假设直流母线调节环路提供基准电源。然后将基准电源除以线电压 RMS 的平方，可得出电导率，再乘以线电压，以给出瞬时电流命令。

通过围绕运行点对方程式 9 进行线性化来形成直流母线调节环路的小信号模型。

$$\hat{i}_{DC} V_{bus} = \eta \bar{V}_{Nrms} \hat{i}_{Nrms} \rightarrow \hat{i}_{DC} = \eta \frac{\bar{V}_{Nrms}}{V_{bus}} \hat{i}_{Li} \quad (9)$$

对于阻性负载，总线电压与电流相关，如方程式 10 所示：

$$\hat{V}_{bus} = \frac{R_L}{1 + sR_L C_o} \hat{i}_{DC} \quad (10)$$

可以画出直流电压调节环路控制模型，如图 3-7 所示。施加了额外的 V_{bus} 前馈，以使控制环路独立于总线电压。因此，总线控制的受控体模型可以写成方程式 11：

$$H_{p_bus} = H_{load} * \eta * \frac{1}{K_{i_gain}} * K_{v_gain} * K_{v_fltr} * \left(\frac{K_{v_gain}}{K_{vac_gain}} \right) \quad (11)$$

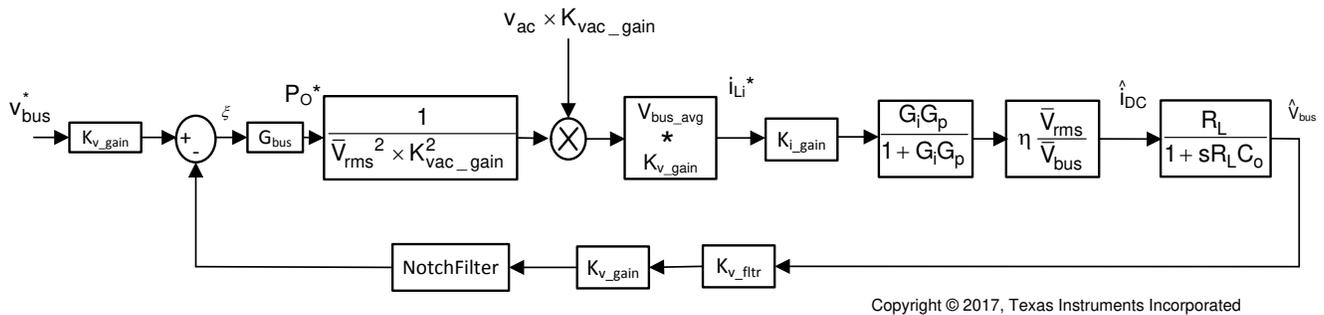


图 3-7. 直流电压环路控制模型

借助图 3-7，为该电压环路设计了一个比例积分器 (PI) 补偿器。该环路的带宽保持在较低水平，因为它在稳态下与 THD 相冲突。

3.5.4 过零附近的软启动可消除或减少电流尖峰

对于 TTPL PFC 拓扑，过零电流尖峰是一个具有挑战性的问题。通过实施软启动方案，状态机按照特定的序列开启和关断开关，可以解决该问题。

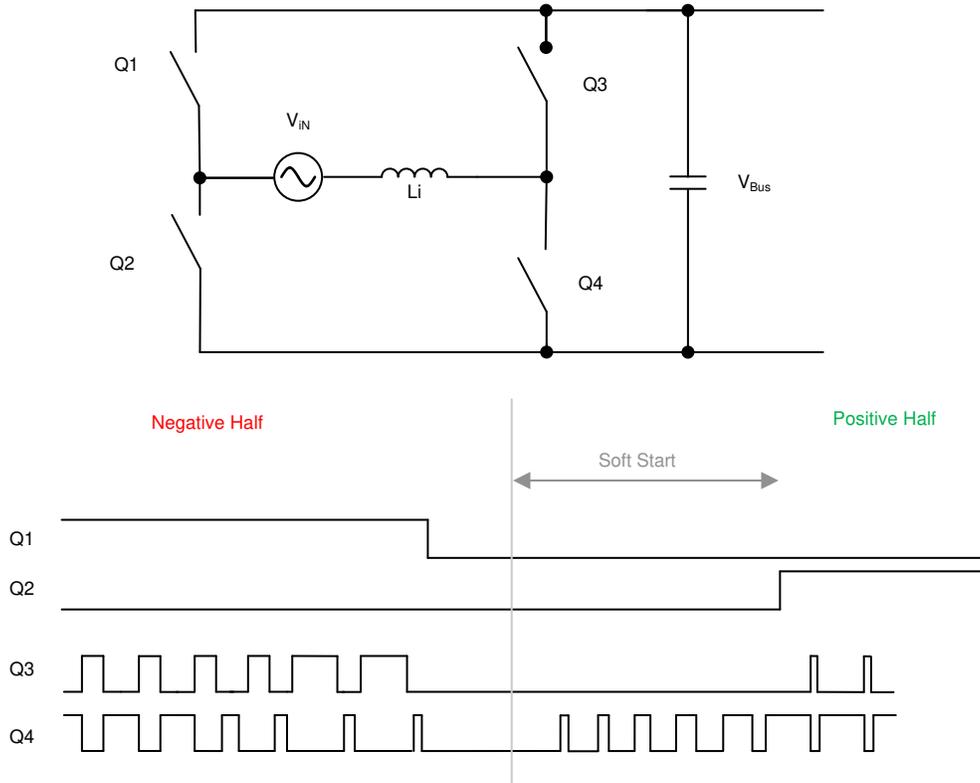


图 3-8. 软启动 PWM 序列，用于减少过零时的电流尖峰

图 3-8 所示为交流波从负到正时的开关序列。在负半周期，Q1 开启，Q3 是有源 FET，Q4 是同步 FET。在此期间，Q2 两端的电压是直流总线电压。当交流周期发生变化时，Q2 必须处于 100% 或接近 100%。如果 Q2 立即开启，则会产生明显的正尖峰。因此，使用软启动序列导通 Q4，如图 3-8 所示。该软启动的调优取决于电感值和其他功率级参数，例如器件 C_{OSS} 。

过零附近出现负电流尖峰的另一个原因是过零附近的交流电压相对较低。当 Q3 导通时，虽然占空比很低，但会施加高电压差，并会导致高负电流尖峰。因此，在 Q3 再次开始切换回之前应用了足够的延迟。

类似地，在软启动开始后，Q2 会延迟一段时间开启。

3.5.5 电流计算

根据最大输入电流选择输入保险丝、滤波器电流额定值，计算方法如方程式 12 所示：

$$I_{inrms} = \frac{P_{out_最大值}}{\eta \cdot V_{inrms} \cdot PF} = 28.2 A \quad (12)$$

其中，

- P_{OUT_MAX} 是最大输出功率 6.6kW
- η 是效率 (假设为 98.6%)
- V_{IN_RMS} 是输入电压 RMS 值 (240V)
- PF 是功率因数 (假设为 0.99)

3.5.6 电感器计算

电感器在影响系统效率、电流纹波和整体尺寸方面发挥着重要作用。电感器始终在效率和功率密度之间保持平衡。根据输入电压、输出电压和最坏情况下的纹波来计算电感值。

可以使用以下公式来计算占空比：

$$D = 1 - \frac{V_{in}}{V_{out}} \quad (13)$$

电感器电流纹波的计算可分为三个阶段：

$$I_{ripple} = \left(\frac{V_{in}}{L} - 2 \times \frac{V_{out} - V_{in}}{L} \right) \times D \times T_s \leftarrow \text{For } D \leq 1/3 \quad (14)$$

$$I_{ripple} = \left(\frac{2 \times V_{in}}{L} - \frac{V_{out} - V_{in}}{L} \right) \times \left(D - \frac{1}{3} \right) \times T_s \leftarrow \text{For } 1/3 < D < 2/3 \quad (15)$$

$$I_{ripple} = \left(\frac{3 \times V_{in}}{L} \right) \times \left(D - \frac{2}{3} \right) \times T_s \leftarrow \text{For } D \geq 2/3 \quad (16)$$

在最坏的情况下，公式变为：

$$I_{ripple} = \frac{V_{out} \times T_s}{12 \times L} \quad (17)$$

该设计的目标是在最大输入功率和最大交流电流下实现 10% 的电流纹波：

$$I_{ripple} < 10\% \times \frac{\sqrt{2} \times P_{out_max}}{V_{in_max} \times \eta} \quad (18)$$

其中，

- P_{out_max} 是最大输出功率
- η 是效率
- V_{in_max} 是最大输入电压

因此，在 12A RMS 电流下，电感的计算结果为 126 μ H。

3.5.7 输出电容器计算

由于直流链路电容器上存在输入双倍工频纹波，因此其电容主要由输出电压纹波决定，其计算方法如方程式 19 所示：

$$C_{out(min)} \geq \frac{P_{out}/V_{out}}{4 \cdot \pi \cdot f_{line_min} \cdot V_{ripple_max}} = 860\mu F \quad (19)$$

其中，

- P_{OUT} 是输出功率
- V_{OUT} 是输出电压
- f_{LINE_MIN} 是最小工频
- V_{RIPPLE} 是输出纹波

实际使用的电容器为 1410 μ F (3 x 470 μ F)。

3.5.8 电流和电压感应

霍尔效应传感器 ACS733KLATR-40AB-T 用于总输入电流检测，如图 3-10 所示。基于 OPA320 的放大器电路将传感器的低输出电压调整到更高的电平，并将该电压发送到控制器 ADC 引脚。ACS733KLATR-40AB-T 器件检测每个交错相的电流，从而实现相电流平衡。

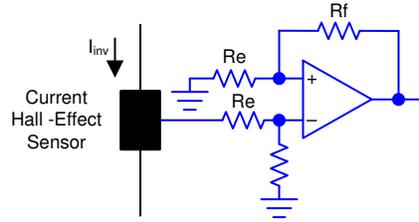


图 3-9. 霍尔效应传感器信号调节电路

信号调节电路的输出电压使用图 3-9 中所示的电路进行调节，以匹配 ADC 范围。电压的计算公式为：

$$I_{out} = \frac{R_f}{R_e} \left(I_{inv} \times \frac{V_{nominal}}{I_{nominal_最大值}} + V_{offset} \right) \quad (20)$$

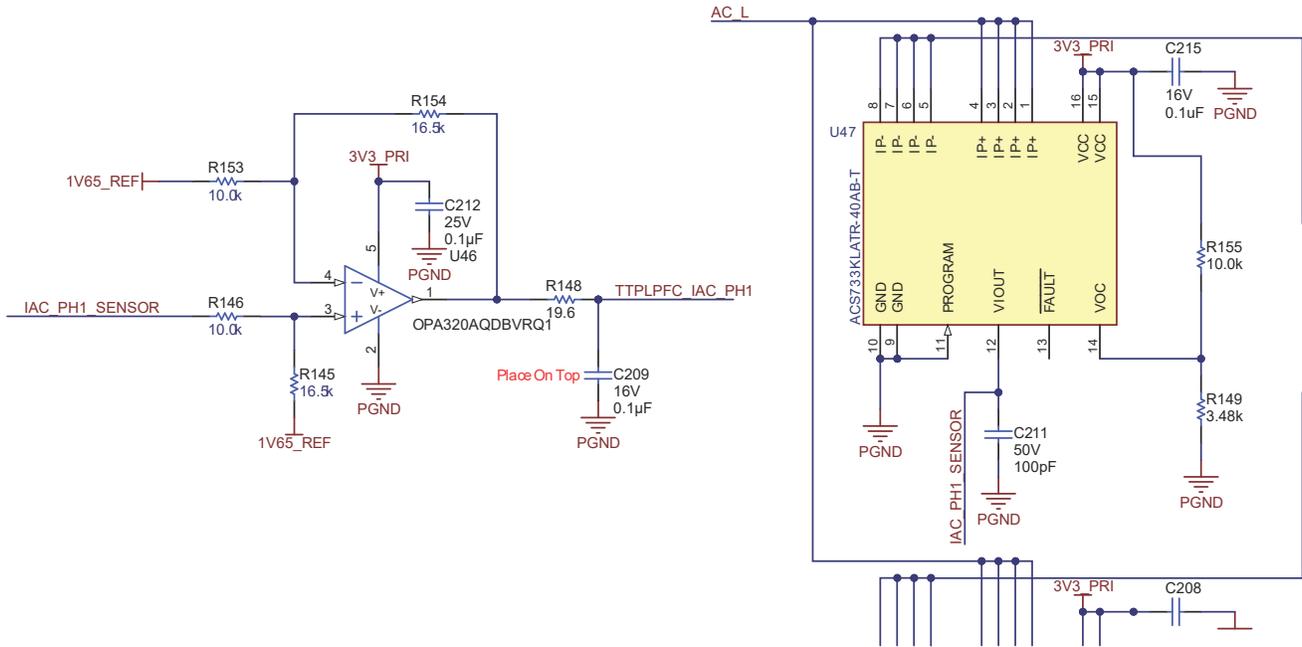


图 3-10. 电流检测原理图

通过检测线路以差分方式检测输入交流电压，中性输入分别通过两个分压器以控制接地为基准，如图 3-11 所示。控制接地是直流链路负极端子；因此，可以使用单个分压器来检测直流总线电压。在连接到控制器之前，使用 RC 滤波器对信号进行滤波。该设计的所有检测信号共用一个 RC 滤波器。

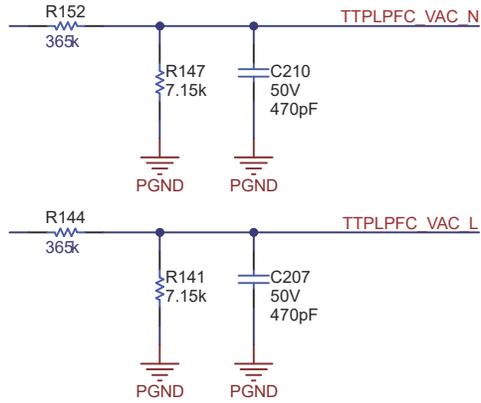


图 3-11. 用于交流输入电压检测的分压器原理图

4 重点产品

4.1 C2000 MCU TMS320F28003x

C2000™ 32 位微控制器针对处理、感应和驱动进行了优化，可提高实时控制应用（如工业电机驱动器、光伏逆变器和数字电源、电动汽车和运输、电机控制以及感应和信号处理）的闭环性能。

TMS320F28003x (F28003x) 是一个功能强大的 32 位浮点微控制器单元 (MCU)，可让设计人员在单个器件上集成关键的控制外设、差分模拟和非易失性存储器。

CLA 能够将大量的常见任务从主 C28x CPU 上卸下。CLA 是一款与 CPU 并行执行的独立 32 位浮点数学加速器。此外，CLA 自带专用存储资源，它可以直接访问典型控制系统中所需的关键外设。与硬件断点和硬件任务切换等主要特性一样，ANSI C 子集支持是标准配置。

F28003x MCU 上集成了高性能模拟块，以进一步支持系统整合。三个独立的 12 位 ADC 可准确、高效地管理多个模拟信号，从而最终提高系统吞吐量。四个模拟比较器模块可以针对跳闸情况对输入电压电平进行持续监控。

TMS320C2000™ 器件包含出色的控制外设，具有与频率无关的增强型脉宽调制器/高分辨率脉宽调制器 (ePWM/HRPWM) 和增强型捕获 (eCAP) 模块，可以对系统进行出色的控制。内置的 Σ - Δ 滤波器模块 (SDFM) 允许在隔离层上无缝集成过采样 Σ - Δ 调制器。

通过各种业界通用通信端口 [例如串行外设接口 (SPI)、串行通信接口 (SCI)、集成电路总线 (I2C) 和控制器局域网 (CAN)] 支持连接，并提供了多个多路复用选项，可在各种应用中实现出色的信号布局。C2000™ 平台新增了完全符合标准的电源管理总线 (PMBus)。此外，快速串行接口 (FSI) 率先在业内实现了高速可靠的通信，补充了嵌入该器件的各种外设的功能。

专门实现的器件型号 TMS320F28003xC 允许访问可配置逻辑块 (CLB) 以支持额外连接功能。有关更多信息，请参阅 [TMS320F28003x 微控制器数据手册](#) 中的“器件比较”表。

嵌入式实时分析和诊断 (ERAD) 模块通过提供用于分析的附加硬件断点和计数器来增强器件的调试和系统分析功能。

以下是该设计突显的 C2000 MCU 功能子集，用于实现高频 CLLLC 拓扑控制：

- 高分辨率 PWM**：C2000 MCU 上的 ePWM 模块可实现皮秒级分辨率，从而能够准确生成高频 PWM。利用 4 类 PWM 高分辨率周期控制，可以实现高分辨率占空比控制、高分辨率死区控制以及高分辨率相移控制。这支持为谐振回路激励生成平衡脉冲，是高频电源转换器的使能特性。
- 用于有源同步整流、具有 ePWM 的**比较器子系统 (CMPSS)**：有源同步整流可实现更高的效率，对于在谐振点以下和以上运行的高频谐振转换器而言，这是该拓扑的必要功能。C2000 MCU 的集成 CMPSS 通过使用集成比较器和集成数模转换器 (DAC) 来实现有源同步整流脉冲的生成。（请参阅节 2.2.2.4。）
- 消隐窗口**：由于开关转换器中的噪声是不可避免的，因此使用消隐窗口功能来在高噪声开关事件期间抑制 CMPSS 输出。该消隐窗口由 ePWM 时基提供，可以在 PWM 周期的不同时间应用。（请参阅节 2.2.3。）
- X-Bar**：电源转换器中可能有多个跳闸源。X-bar 能够组合来自不同 CMPSS 或 GPIO 的不同跳闸操作，以在无需外部逻辑的情况下生成 ePWM 中所需的跳闸行为。
- 控制律加速器 (CLA)** 支持在单个控制器上集成对多个拓扑的控制。该设计随附的软件提供了在 CLA 或 C28x 上运行控制环路的选项。
- PWM 模块中的全局链接功能**支持通过单次写入更新多个 PWM，从而减轻 CPU 负担并轻松控制频率更高的转换器。

4.2 LMG352xR30-Q1

LMG352xR30-Q1 是一款符合汽车标准的 650V 30mΩ GaN FET，具有集成驱动器、保护和温度报告功能。集成驱动器可实现高达 150V/ns 的开关速度。与分立式外部栅极驱动器相比，TI 的集成式精密栅极偏置可实现更高的开关 SOA。这种集成与低电感封装相结合，可在硬开关电源拓扑中提供干净的开关和超小的振铃。其他特性（包括用于 EMI 控制的可调栅极驱动强度、过热保护、稳健过流保护和故障指示）可提供优化的 BOM 成本、布板尺寸和外形尺寸。高级电源管理功能包括数字温度报告；GaN FET 的温度通过可变占空比 PWM 输出进行报告，从而使系统能够以最佳方式管理负载。

4.3 UCC21222-Q1

UCC21222 器件是具有可编程死区时间的隔离式双通道栅极驱动器。该器件采用 4A 峰值拉电流和 6A 峰值灌电流来驱动功率 MOSFET、IGBT 和 GaN 晶体管。

UCC21222 器件可配置为两个低侧驱动器、两个高侧驱动器或一个半桥驱动器。该器件的 5ns 延迟匹配性能允许并联两个输出，能够在重负载条件下将驱动强度提高一倍，而无内部击穿风险。

输入侧通过一个 3.0kV_{RMS} 隔离栅与两个输出驱动器隔离，共模瞬态抗扰度 (CMTI) 的最小值为 100V/ns。

可通过电阻器进行编程的死区时间帮助您调整系统限制的死区时间，从而提高效率并防止输出重叠。其他保护特性包括：当 DIS 设置为高电平时，通过禁用功能同时关闭两路输出；集成的抗尖峰滤波器可抑制短于 5ns 的输入瞬变；以及在输入和输出引脚上对高达 -2V 的尖峰进行 200ns 的负电压处理。所有电源都有 UVLO 保护。

4.4 AMC3330-Q1

AMC3330 是一款具有完全集成的隔离式直流/直流转换器的精密隔离式放大器，能够实现器件低侧的单电源运行。该增强型电容隔离层通过了 VDE V 0884-11 和 UL1577 标准认证，将以不同共模电压电平运行的系统各部分隔开，并保护低压域免受损坏。AMC3330 的输入经优化，可直接连接高阻抗电压信号源（例如电阻分压器网络），用于检测高压信号。集成式隔离直流/直流转换器可测量不以地为基准的信号，并使该器件成为充满噪音的空间受限型应用的独特设计。

4.5 AMC3302-Q1

AMC3302 是一款隔离式精密放大器，针对基于分流器的电流测量进行了优化。完全集成的隔离式 DC/DC 转换器允许从器件的低侧进行单电源操作，从而使该器件成为空间受限应用的独特解决方案。增强型电容式隔离栅已通过 VDE V 0884-11 和 UL1577 认证，并支持高达 1.2kV_{RMS} 的工作电压。该隔离栅将运行在不同共模电压电平上的系统部件隔开，并保护低压侧免受危险电压冲击和损坏。AMC3302 的输入经过优化，可直接连接到低阻抗分流电阻器或另一具有低信号电平的低阻抗电压源。出色的直流精度和低温漂支持在 -40°C 至 +125°C 的扩展工业温度范围内进行精确的电流测量。

5 硬件、软件、测试要求和测试结果

5.1 所需的硬件和软件

本部分将详细介绍硬件，并说明电路板上的不同区域，以及如何为本设计指南所述的实验设置这些区域。

5.1.1 硬件设置

该设计遵循高速边缘卡 (HSEC) 控制卡理念，而且可能会用到从 C2000 MCU 产品系列中提供 HSEC 控制卡的任何器件。表 5-1 中列出了微控制器上用于控制功率级的重要资源。图 5-1 展示了参考设计上的主要功率级和连接器。表 5-3 列出了主要连接器及其功能。

1. 确保没有电源连接到电路板。
2. 在 J25 插槽中插入控制卡。
3. 在 J15 上连接 12V 辅助电源 (+12V, 2A) 的电源 (但不要上电)，如图 5-1 所示。
4. 现在，开启辅助电源的电源。控制卡上的绿色 LED 将亮起。这表明 C2000 MCU 器件已通电。注意：微控制器的偏置与功率级是分离的；这样就能够按照这组指令安全地启动系统。
5. 要连接 JTAG，请使用 USB 电缆连接控制卡和主机。
6. 为了使 TTPLPFC 级正常运行，交流输入必须连接到 J33 (90V - 264V)。为了进行测试，使用了功率大于 10kW 的电源，但在仅进行低功率测试的情况下，可以使用干净、稳定且额定值较低的电源。
7. 为了使 PFC 级独立运行，可以将负载连接到 J37 和 J38，也可以使用 CLLLC 来加载 PFC 级。
8. 为了使 CLLLC 级独立运行，可以将直流电源 (400V) 连接到 J15 处的 VBUS。如果执行该操作，则不应在软件中启动 TTPLPFC，并且不应连接上面的第 6 步中介绍的交流电源。
9. 使用时，应将负载连接到 CLLLC 转换器的次级侧。可以使用 J7 和 J10 连接此类负载。
10. 在操作 PFC 和直流/直流级时，如上面的第 6 步所示连接一个交流电源，并如上面的第 9 步所示连接一个负载。不需要连接 VBUS，但电流泄放电阻器可能有助于确保在执行 OBC 后快速释放过高的电压。
11. 可以连接电流和电压探针来观察初级和次级的谐振回路电流。或者，也可以连接功率计来测量效率。

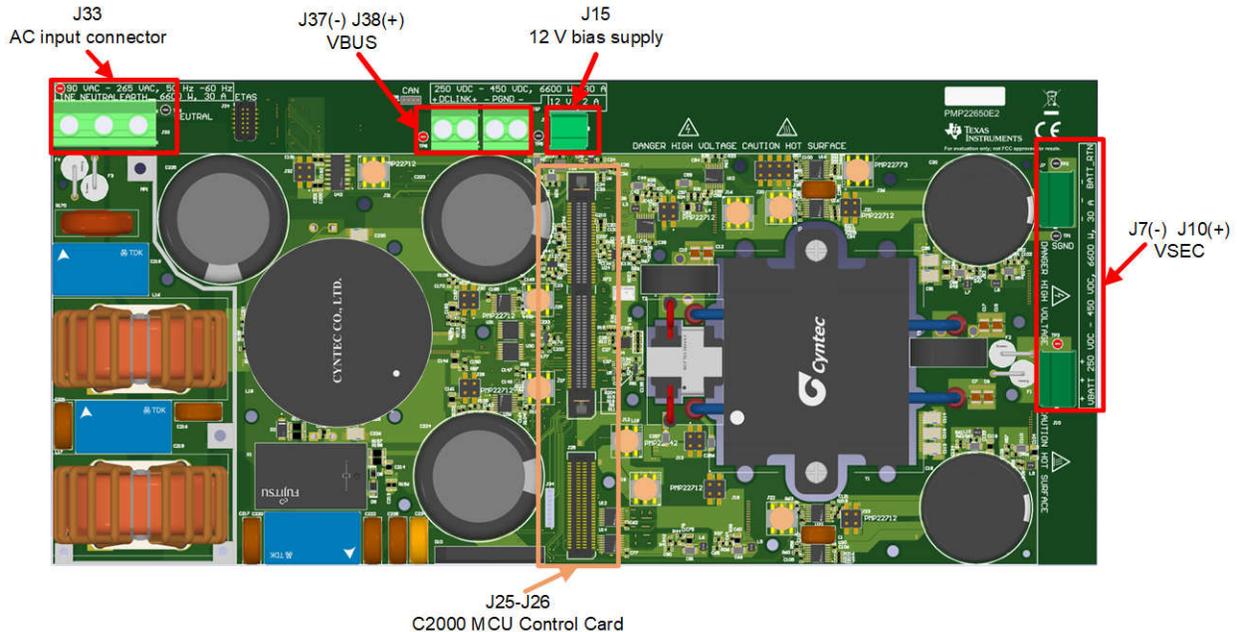


图 5-1. 电路板概述

需要 7 个辅助电源子卡 (以红色显示)

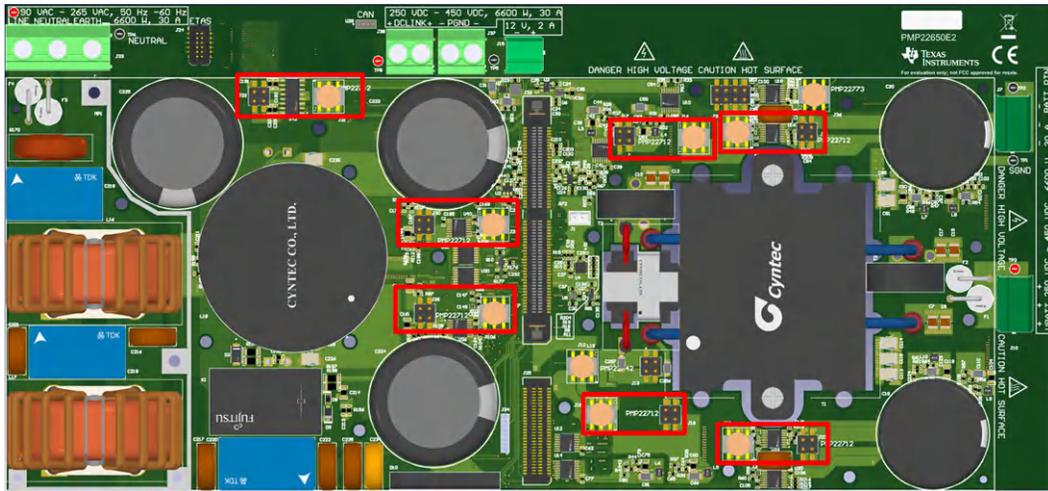


图 5-2. PMP22712 - 辅助电源

需要一个反馈隔离子卡 PMP22773 (以红色表示)

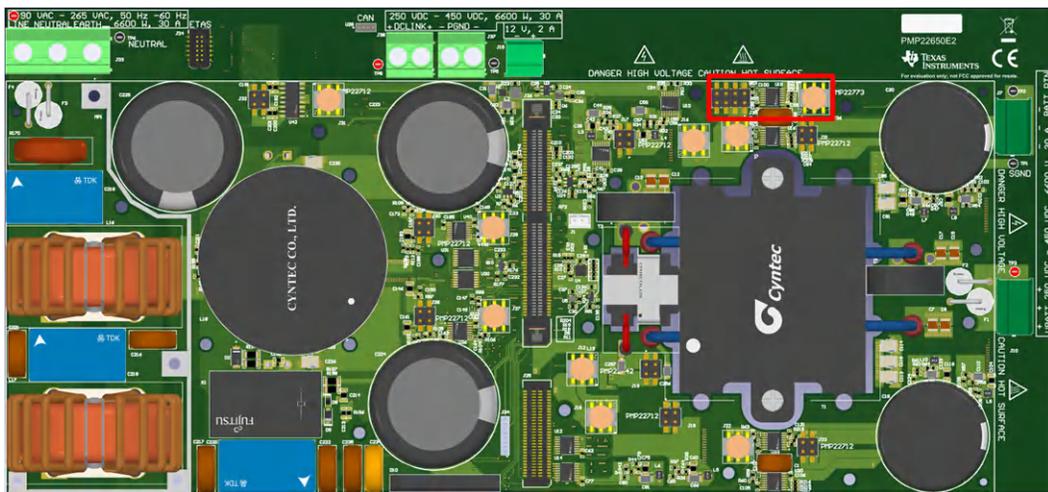


图 5-3. PMP22773 - 反馈隔离子卡

表 5-1. 关键数字引脚分配

信号名称	HSEC 引脚编号	F28003x 外设
SYSTEM ISR Trigger	-	ECAP1
CLLLC_CONTROL_OUTPUT_DAC_PIN	14	DACA
CLLLC_PRIM_LEG1_H/L	49/51	EPWM1 (A/B)
CLLLC_PRIM_LEG2_H/L	53/55	EPWM2 (A/B)
CLLLC_SEC_LEG1_H/L	50/52	EPWM3 (A/B)
CLLLC_SEC_LEG2_H/L	54/56	EPWM4 (A/B)
CLLLC_FAULTn	74	GPIO-23 → INPUTXBAR2
CLLLC_LC_CHANGE	62	GPIO-14
CLLLC_SEC_SIDE_DIAG	80	GPIO-30
TTPLPFC_LOW_FREQ_H/L	57/59	EPWM5 (A/B)
TTPLPFC_HIGH_FREQ_PH1_H/L	61/63	EPWM6 (A/B)
TTPLPFC_HIGH_FREQ_PH2_H/L	58/60	EPWM7 (A/B)
TTPLPFC_FAULTn	72	GPIO-22 → INPUTXBAR1
TTPLPFC_INRUSH_RELAY_CTRL	64	GPIO-15

表 5-1. 关键数字引脚分配 (续)

信号名称	HSEC 引脚编号	F28003x 外设
ERRORSTSn	102	GPIO55
SYSTEM_WATCHDOG_OUT	75	GPIO24
SYSTEM_WATCHDOG_DISABLE	77	GPIO25 (电阻器选项)
SYSTEM_PMIC_SPI (保留)	79	GPIO26 (电阻器选项)
SYSTEM_PMIC_SPI (保留)	81	GPIO27 (电阻器选项)
SYSTEM_DISABLE_FET_SUPPLY	85	GPIO32
SYSTEM_TEMP_MUX_OUT1	91	GPIO41 -> ECAP2 → INPUTXBAR3
SYSTEM_TEMP_MUX_OUT2	96	GPIO60 -> ECAP3 → INPUTXBAR4
SYSTEM_TEMP_MUX_SEL_1-3	93 94 95	GPIO47 GPIO58 GPIO59
SYSTEM_PROFILING1-3	89 92 101	GPIO40 GPIO44 GPIO49
FSI_TX_D0	101	GPIO-49/FSITXA_D0
FSI_TX_D1	103	GPIO-50/FSITXA_D1
FSI_TX_CLK	105	GPIO-51/FSITXA_CLK
LED1	82	GPIO-31 → LED1
LED2	86	GPIO-34 → LED2 (SFRA)

该表介绍了参考设计的采样方案。顶部的每一列代表一个独立的 ADC。每个 ADC 的工作完全独立于其他 ADC。每个信号被分配一个或多个转换开始 (SOC)。每个 SOC 代表该通道的一次独立读取，例如在 ADCA 内为 TTPLPFC_IAC_PH1 分配了 SOC0 和 SOC1。这意味着该信号在每个周期中会被采样两次，一次由 ePWM6_SOCA 触发，一次由 ePWM6_SOCB 触发。由于该触发器以 120kHz 的频率运行，因此在每个 120kHz 采样周期中会有效地对信号进行两倍过采样。类似地，会对 CLLLC_ISEC 进行 11 倍过采样，不对 CLLLC_IPRIM 进行过采样。该表中还给出了几个低频采样信号，可以看出这些信号使用了不同的 SOC 信号。最后，由于使用循环计数器按数字顺序处理 SOC，因此该表按照采样顺序从上到下读取为时间线。

表 5-2. 关键模拟信号

	ADC-A	ADC-B	ADC-C
具有最高优先级的信号 (120kHz)	TTPLPFC_IAC_PH1 (A2 , CMPSS1) SOC0 → ADC_TRIGGER_EPWM6_SOCA SOC1 → ADC_TRIGGER_EPWM6_SOCB	TTPLPFC_IAC_PH2 (B12 , CMPSS3) SOC0 → ADC_TRIGGER_EPWM6_SOCA SOC1 → ADC_TRIGGER_EPWM6_SOCB	TTPLPFC_VAC (C7) SOC0 → ADC_TRIGGER_EPWM6_SOCA SOC1 → ADC_TRIGGER_EPWM6_SOCB
	CLLLC_ISEC (A5 , CMPSS2) SOC2 → ADC_TRIGGER_EPWM6_SOCA SOC3 → ADC_TRIGGER_EPWM6_SOCA SOC4 → ADC_TRIGGER_EPWM6_SOCA SOC5 → ADC_TRIGGER_EPWM6_SOCA SOC6 → ADC_TRIGGER_EPWM6_SOCA SOC7 → ADC_TRIGGER_EPWM6_SOCA SOC8 → ADC_TRIGGER_EPWM6_SOCB SOC9 → ADC_TRIGGER_EPWM6_SOCB SOC10 → ADC_TRIGGER_EPWM6_SOCB SOC11 → ADC_TRIGGER_EPWM6_SOCB SOC12 → ADC_TRIGGER_EPWM6_SOCB	TTPLPFC_VBUS/CLLLC_VBUS (B4) SOC2 → ADC_TRIGGER_EPWM6_SOCA SOC3 → ADC_TRIGGER_EPWM6_SOCB SOC4 → ADC_TRIGGER_EPWM7_SOCA SOC5 → ADC_TRIGGER_EPWM7_SOCB	CLLLC_VSEC (C11 , CMPSS2) SOC2 → ADC_TRIGGER_EPWM6_SOCA SOC3 → ADC_TRIGGER_EPWM6_SOCA SOC4 → ADC_TRIGGER_EPWM6_SOCA SOC5 → ADC_TRIGGER_EPWM6_SOCA SOC6 → ADC_TRIGGER_EPWM6_SOCA SOC7 → ADC_TRIGGER_EPWM6_SOCA SOC8 → ADC_TRIGGER_EPWM6_SOCA SOC9 → ADC_TRIGGER_EPWM6_SOCA SOC10 → ADC_TRIGGER_EPWM6_SOCA SOC11 → ADC_TRIGGER_EPWM6_SOCA SOC12 → ADC_TRIGGER_EPWM6_SOCA
	CLLLC_IPRIM (A9 , CMPSS2) SOC13 → ADC_TRIGGER_EPWM1_SOCA		
低频采样信号 (10kHz)	TTPLPFC_VAC_L (A4) SOC14 → ADC_TRIGGER_CPU1_TINT2	TTPLPFC_VAC_N (B2) SOC10 → ADC_TRIGGER_CPU1_TINT2	TTPLPFC_VBUS2 (C10 , CMPSS2) SOC14 → ADC_TRIGGER_CPU1_TINT2
	SYSTEM_TEMP_1 (A11) SOC15 → ADC_TRIGGER_CPU1_TINT2	SYSTEM_VREF_1_65 (B5) SOC11 → ADC_TRIGGER_CPU1_TINT2	CLLLC_VSEC (C11 , CMPSS2) VSEC13 → SOC15 → ADC_TRIGGER_CPU1_TINT2
未采样，仅限 CMPSS		CLLLC_IPRIM_TANK (A12/C5 , CMPSS2)	CLLLC_ISEC_TANK (C1 , CMPSS4)

表 5-3. 主要连接器及其功能

连接器名称	功能
J33	交流输入
J37/J38	VBUS 连接；PFC 输出、直流/直流 VPRIM
J7/J10	直流/直流输出连接；直流/直流 VSEC
J15	12V 2A 电源
J25/J26	HSEC 控制卡连接器槽

5.1.1.1 控制卡设置

需要对器件控制卡进行某些设置，以通过 JTAG 进行通信并使用隔离式 UART 端口。用户还必须提供正确的 ADC 基准电压。下面是需要对 F280039C 控制卡修订版 A 进行的设置。用户还可以参考位于 C2000Ware 内 `\c2000ware\boards\controlcards\TMDSCNCD280039C` 下的信息表，也可以从 [F280039 controlCARD 信息指南](#) 中获取该信息表。

1. 必须在两端将控制卡上的 S1:A 设置到 *开启 (左侧)* 位置，以实现 JTAG 到器件的连接以及 SFRA GUI 的 UART 连接。如果该开关被设置为 *关闭 (右侧)*，则用户无法使用控制卡上的内置隔离式 JTAG，SFRA GUI 也无法与器件进行通信。
2. J1:A 是适用于 USB 电缆的连接器的，该电缆用于实现运行 Code Composer Studio™ 集成开发环境 (IDE) 的主机 PC 与器件之间的通信。
3. 需要为该设计的控制环路调优使用 3.3V 基准。使用 F28003x 的内部基准，为此，必须将 S3 开关移至顶部位置 (即指向 INT)。
4. 控制卡上隔离式接地端之间连接了一个电容器 C7:A。在连接高压电源之前必须将其拆下。

为了实现最佳性能，需要在多个 ADC 通道中添加 RC 滤波电容器。丝印中清楚地标记了每个元件标识符，可以在适用于 TMDSCNCD280039C 的 C2000Ware 中找到装配图。

`\c2000ware\boards\controlcards\TMDSCNCD280039C`

信号	电容器元件标识符	电容值
TTPLPFC_VBUS	C46	1μF 0603
CLLLC_VSEC	C43	1μF 0603
TTPLPFC_VAC	C32	1μF 0603
TTPLPFC_IAC_PH1	C31	1μF 0603
TTPLPFC_IAC_PH2	C49	1μF 0603
CLLLC_ISEC_TANK	C50	560pF 0603
CLLLC_ISEC	C34	1μF 0603

5.1.2 软件

适用于 C2000 MCU 的 [DigitalPower 软件开发套件 \(SDK\)](#) (C2000WARE-DIGITALPOWER-SDK) 中提供了适用于该设计的软件。

5.1.2.1 在 Code Composer Studio 中打开工程

首先：

1. 从 Code Composer Studio (CCS) 集成开发环境 (IDE) 工具文件夹安装 Code Composer Studio。建议使用版本 12.0 或更高版本。
2. 通过以下两种方式之一安装 C2000WARE-DIGITAL-POWER-SDK：
 - 通过 C2000Ware Digital Power SDK 工具文件夹
 - 转至 CCS 的“View”→“Resource Explorer”下。在 TI Resource Explorer 下，转至 C2000WARE-DIGITAL-POWER-SDK，然后点击安装按钮。
3. 安装完成后，关闭 CCS，然后打开一个新的工作区。CCS 自动检测 powerSUITE。可能需要重新启动 CCS 才能使更改生效。

备注

powerSUITE 默认随 SDK 一起安装。

现在可以按如下方式导入固件工程：

用户还可以执行以下操作来直接导入工程：进入 CCS 内部以点击“Project”→“Import CCS Projects”，并浏览到位于 `<SDK>/solutions/tidm_02013/f28003x/ccs` 中的解决方案文件夹。

此时将显示工程规范，点击该规范将创建一个包含该工程的所有依赖项的自包含文件夹。

5.1.2.2 工程结构

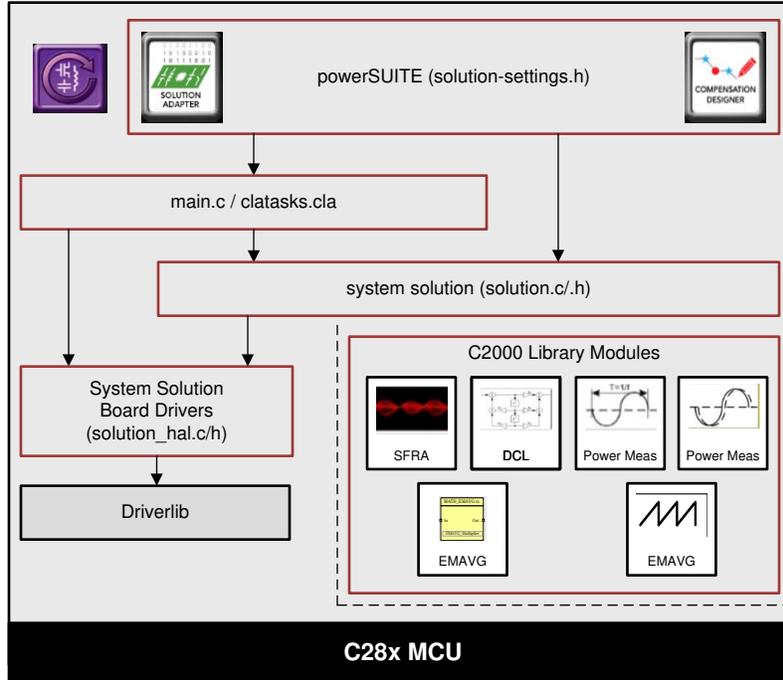


图 5-4. 工程结构概览

图 5-4 展示了工程的总体结构。导入工程后，CCS 中将显示 Project Explorer，如图 5-5 所示。

`<solution>.c/h` 中包含特定于解决方案并且独立于器件的文件，其中含有核心算法代码。例如 `TTPLPFC.c` 或 `CLLLC.h`。

`<solution>_hal.c/h` 中包含特定于电路板和特定于器件的文件。该文件由特定于器件的驱动程序组成，以用于运行解决方案。如果用户想要使用不同的调制方案或不同的器件，除了更改工程中的器件支持文件，用户只需要对这些文件进行更改。

`<solution>-main.c` 文件包含工程的主要框架。该文件由对电路板和解决方案文件的调用（有助于构建系统框架）以及中断服务例程 (ISR) 和慢速后台任务组成。

对于该设计，有三个 `<solution>` 名字对象，即 `obc_7_4kw`、`clllc` 和 `ttplpfc`。请注意，为了保持最大的灵活性，我们已选择使 `CLLLC` 代码库和 `TTPLPFC` 代码库尽可能保持独立，而在需要包含独立于 `TTPLPFC` 和 `CLLLC` 的设置时添加了 `obc_7_4kw` 文件。这使最终用户能够独立操作每个级，并根据需要在其最终设计中为 PFC 或直流/直流级轻松整合不同的拓扑。

`<solution>_settings.h` 文件包含代码配置设置，例如要构建哪个实验。而 `<solution>_user_settings.h` 包含板级配置，例如 ADC 映射、GPIO 等的 `#define`。

`solution.js` 文件包含一个脚本文件，该文件可以在执行每个实验期间帮助将相关变量填充到观测器中。要使用这些脚本，请打开脚本控制台（“View” - “Scripting Console”）。将 `solution.js` 文件的内容粘贴到脚本控制台中，然后按 `Enter`。这将填充表达式窗口以供稍后调试。

解决方案名称也用作解决方案中使用的所有变量和定义的模块名称。因此，所有变量和函数调用都以 `CLLLC` 名称作为前缀（例如 `CLLLC_vSecSensed_pu`）。这种命名规则使用户组合不同的解决方案，同时避免命名冲突。

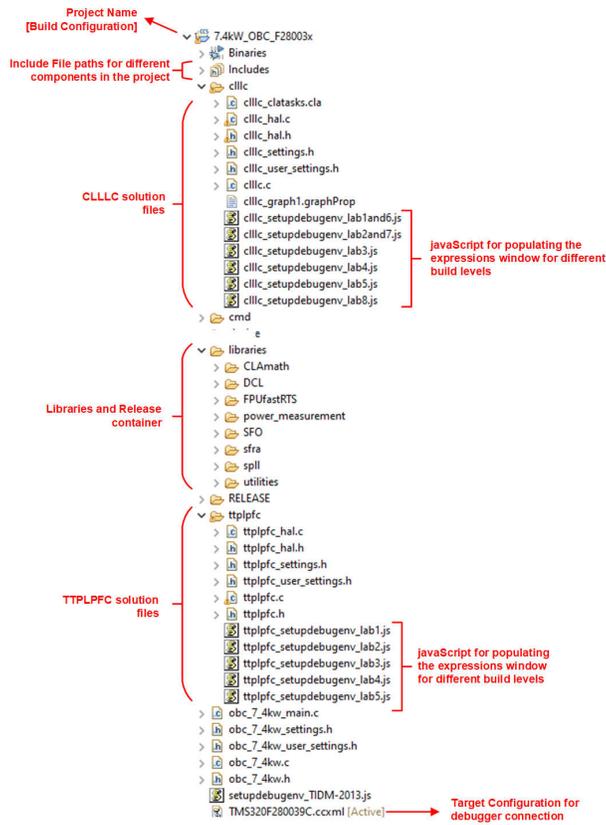


图 5-5. CLLLC 工程的 Project Explorer 视图

OBC 工程由在 2 个内核 (C28x 内核和 CLA 内核) 上运行的三个 ISR (ISR1、ISR2 和 ISR3) 组成。通过对 ISR1 触发器使用 ePWM，对 ISR2 触发器使用 eCAP，对 ISR3 触发器使用 ADC，可以完全通过硬件来控制 ISR 优先级。表 5-4 展示了每个 ISR 的分区方式及其任务。

表 5-4. ISR 分区和任务

ISR	触发源	C28x	CLA
ISR1 (120kHz)	ePWM	不适用	更新 CLLLC PWM 值
ISR2 (120kHz)	eCAP	PFC 电流环路	CLLLC 控制代码，启用 ISR1
ISR3 (10kHz)	ADC	PFC 电压环路，仪表	不适用

ISR1 是为 PWM 更新保留的速度最快且不可嵌套的 ISR，完全在 CLA 上运行。ISR1 由 PRIM_LEG1_PWM_BASE → EPWM_INT_TBCTR_U_CMPC 事件触发。通常，可以通过向 CMPC 写入一个值来禁用该中断，该值大于 PRIM_LEG1_PWM_BASE 的所有可能的 TBPRD 寄存器值。这是通过 CLLLC_HAL_setupISR1Trigger 函数完成的。以下是与该 ISR 相关的定义。

```
#define CLLLC_ISR1_PERIPHERAL_TRIG_BASE CLLLC_PRIM_LEG1_PWM_BASE
#define CLLLC_ISR1_TRIG INT_EPWM1
#define CLLLC_ISR1_PIE_GROUP INTERRUPT_ACK_GROUP3
```

```
#define CLLLC_ISR1_TRIG_CLA CLA_TRIGGER_EPWM1INT
```

ISR2 被拆分到两个内核上。这样便可以对 TTPLPFC 和 CLLLC 代码进行简单的模块化代码隔离。在 C28x 上运行的 ISR2 和在 CLA 上运行的 ISR2 均由同一个源触发并协同运行。C28x 内核运行与 TTPLPFC 相关的任务，而 CLA 内核运行与 CLLLC 执行相关的任务。

ISR2 负责在需要 ISR1 时通过对 CMPC 进行写入以写入有效值来触发 ISR1。(注意：为了实现这一点，CMPC 未绑定到全局加载机制。此外，CMPC 的影子加载被禁用。) 可以调整 CMPC 值以从 ISR1 获取所需的时序。每次启用 ISR1 时，都会将其触发两次。在第一个 ISR1 中，PWM 寄存器被更新，同步被启用。在第二个 ISR1 中，PWM 同步被禁用，CMPC 被设置为一个适当的值，使得 ISR1 不会再次触发。为简单起见，软件图和结构仅显示首次触发的 ISR1。

ISR2 以 ISR2_FREQUENCY 定期触发。一个备用 CAP 模块用于生成时基并触发中断。一个备用 ePWM 模块也配置有相同的时基，用于触发 ADC 转换。ISR2 负责运行控制律和计算 PWM 所需的时钟周期。影子寄存器写入完成后，ISR2 通过向 CMPC 寄存器写入一个有效值 (即一个小于当前 TBPRD 寄存器的值) 来启用 ISR1 触发器。ISR2 有 ISR2_primToSecPowerFlow 和 ISR2_secToPrimPowerFlow 两个变体，一个用于从初级侧到次级侧的功率流，另一个用于次级侧到初级侧的功率流。这样做是为了在控制不同方向的功率流时优化 CPU 周期。为了方便说明，这两者均在各自的实验中被称为 ISR2。根据时序，ISR1 可能会嵌套 ISR2 以进行更新写入，这些写入是时序关键型操作。以下是与该 ISR 相关的定义。

```
#define CLLLC_ISR2_ECAP_BASE ECAP1_BASE
#define CLLLC_ISR2_PWM_BASE EPWM5_BASE
#define CLLLC_ISR2_TRIG INT_ECAP1
#define CLLLC_ISR2_PIE_GROUP INTERRUPT_ACK_GROUP4
```

```
#define CLLLC_ISR2_TRIG_CLA CLA_TRIGGER_ADCA2
```

ISR3 完全在 C28x 内核上运行并由 ADCINT2 触发。ADCINT2 由使用 CPU 计时器启动的转换启动。它用于运行 TTPLPFC 的电压环路以及通用功能，例如对电流和电压信号执行连续均值计算以消除噪声。甚至为命令基准运行压摆率功能。

```
#define CLLLC_ISR3_TIMEBASE CLLLC_TASKC_CPUTIMER_BASE
#define CLLLC_ISR3_PERIPHERAL_TRIG_BASE ADCC_BASE
#define CLLLC_ISR3_TRIG INT_ADCC2
```

```
#define CLLLC_ISR3_PIE_GROUP INTERRUPT_ACK_GROUP10
```

这样就可以轻松地嵌套中断。图 5-6 展示了发生的三个中断的嵌套。该图像是通过监视窗口在周期变化启动时在开环中获取的，因此仅观察到一次 ISR1 触发。对于闭环系统，一个控制 ISR 周期到另一个控制 ISR 周期的时长变化很小，因此 ISR1 将被重复触发。

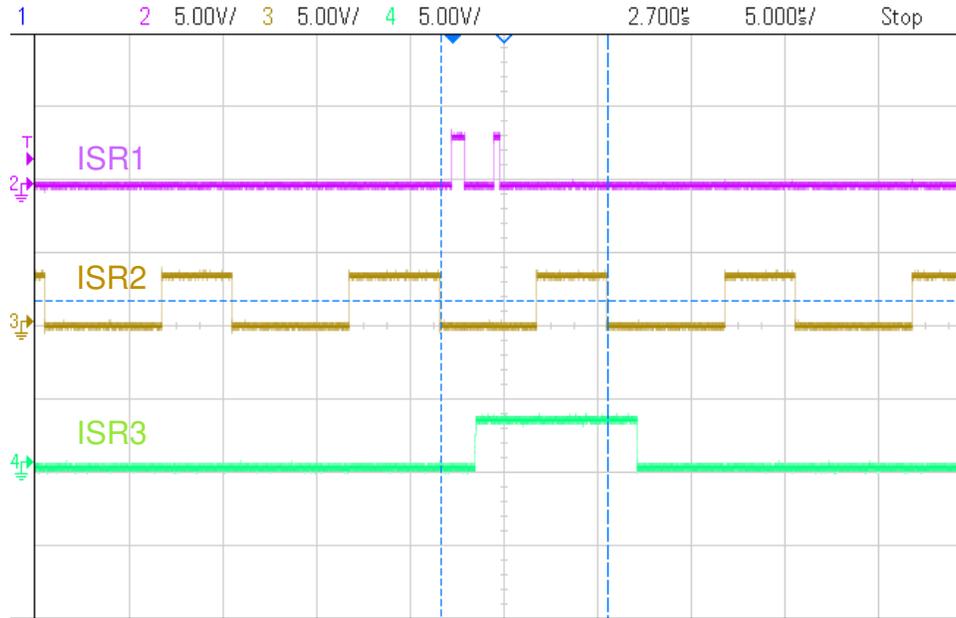


图 5-6. 三级嵌套 ISR

另外，CPU 计时器用于触发慢速后台任务（这些任务均不由中断驱动，而是采用轮询方式）。

“A”任务在 TASKA_FREQ（即 100Hz）时触发。必须以此速率调用 SFRA GUI。以这一速率执行一个任务 A1。

“B”任务在 TASKB_FREQ（即 10Hz）时触发。这些任务用于一些基本的 LED 切换和不是时间关键型的状态机项目。这为三个任务（B1、B2、B3）提供服务；因此，每个任务的执行速率为 3.33Hz。

```
#define TASKA_FREQ 100
#define TASKB_FREQ 10
```

该参考设计的软件被组织到按解决方案分隔的实验中，每个实验具有增量构建 (INCR_BUILD)。这些测试简化了系统启动和设计。

CLLLC 实验

实验 1：初级到次级功率流，开环检查 PWM 驱动器，未向电路板施加高功率。请参阅节 5.2.2.1

实验 2：初级到次级功率流，开环检查 PWM 驱动器和 ADC，具有保护功能，次级连接阻性负载。请参阅节 5.2.2.2

实验 3：初级到次级功率流，闭合电压环路检查，次级连接阻性负载。请参阅节 5.2.2.3

实验 4：初级到次级功率流，闭合电流环路检查，次级连接阻性负载。请参阅节 5.2.2.4

实验 5：初级到次级功率流，闭合电流环路检查，次级连接与电压源并联的阻性负载，以模拟次级侧的电池连接。请参阅节 5.2.2.5

这些定义位于“settings.h”文件中，可以直接在该文件中更改这些定义。

5.2 测试和结果

5.2.1 测试设置 (初始)

图 5-7 展示了用于开始测试该设计的硬件测试设置。节 5.1.1 介绍了详细的硬件设置。

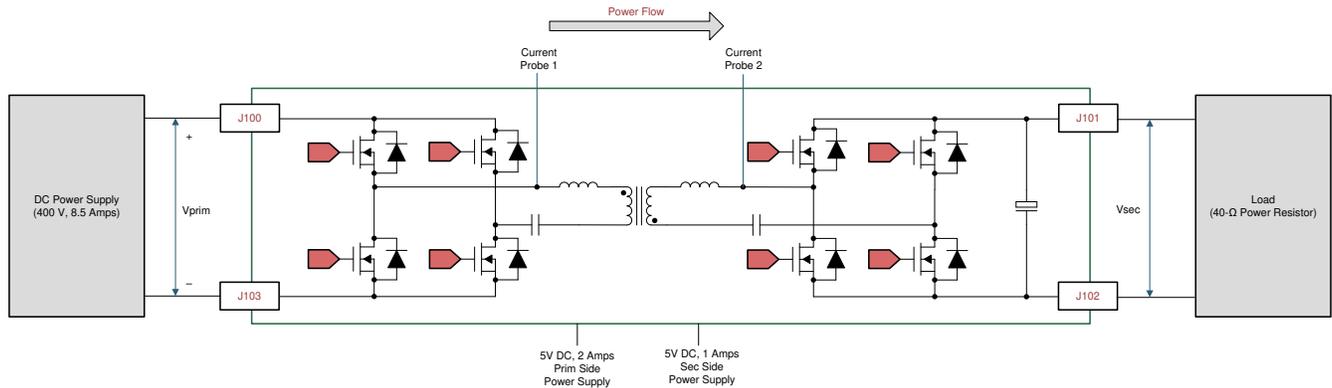


图 5-7. 用于运行软件的硬件设置

5.2.2 CLLC 测试程序

5.2.2.1 实验 1. 初级到次级功率流，开环检查 PWM 驱动器

从软件的角度来看，该实验选项主要作为针对 PWM 的重点测试提供，以便其可以独立于参考设计的硬件连接运行。在该实验中，用户可以在 C2000 控制卡或 LaunchPad 上运行代码来观察 PWM 波形。

用于加载和运行的步骤与节 5.2.2.2 类似。

如果预计 PWM 驱动器不会发生任何变化，则可以轻松跳过该实验，用户可以直接转到实验 2。因此，未记录该实验过程，因为该实验过程主要用于 PWM 驱动器开发和调试。

5.2.2.2 实验 2. 初级到次级功率流，开环检查 PWM 驱动器和 ADC，具有保护功能，次级连接阻性负载

在该构建中，电路板以开环方式和指定的频率受到激励，可通过监视窗口更改该频率。该频率由 CLLC_pwmPeriodRef_pu 变量控制。

该构建验证来自功率级的反馈值检测以及 PWM 栅极驱动器的运行，并确保没有硬件问题。此外，可以在该生成中执行输入和输出电压感应校准。图 5-8 展示了该构建的软件结构。

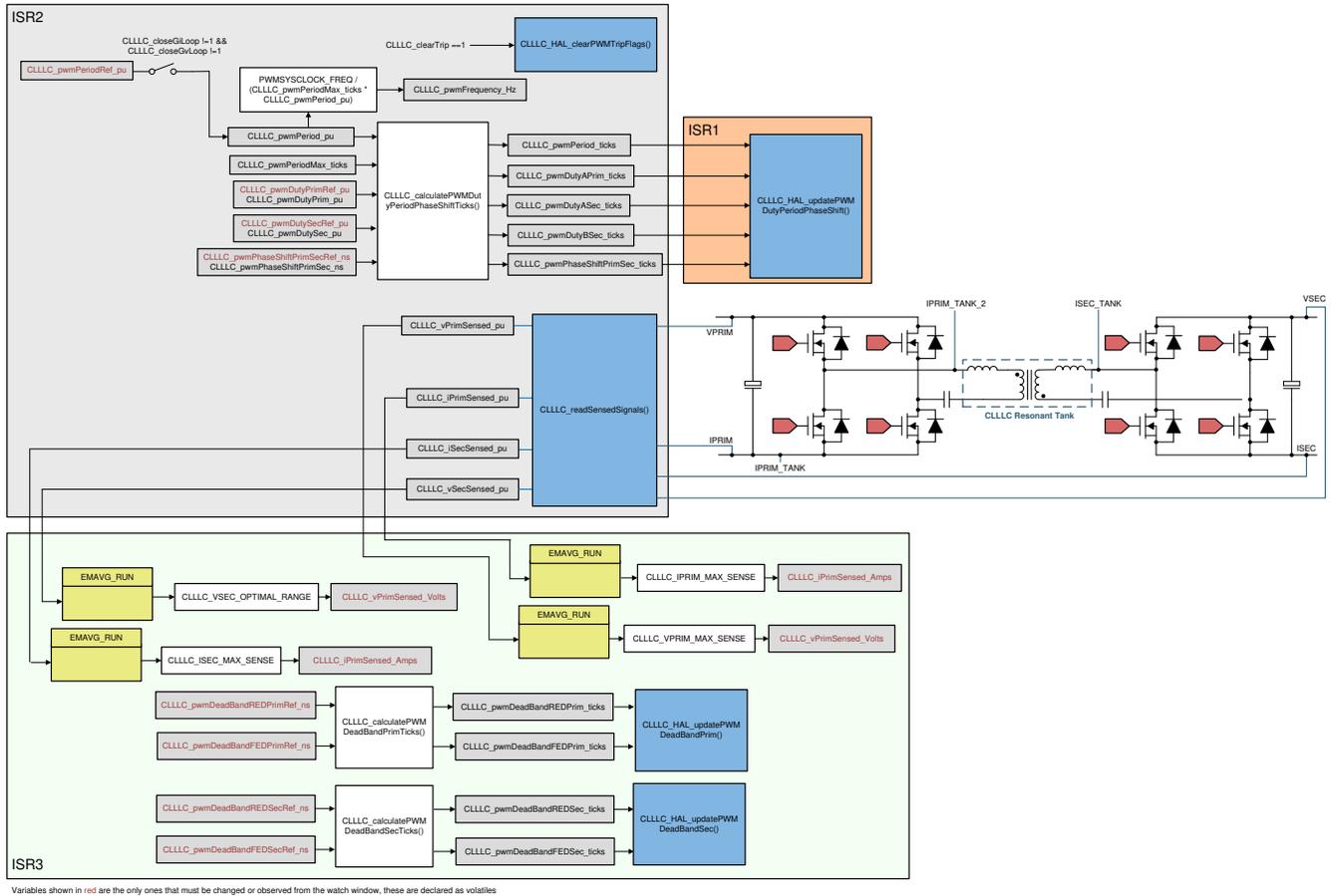


图 5-8. 实验 1 和 2 软件图

5.2.2.2.1 设置实验 2 的软件选项

1. 要开始进行设置，请按照节 5.1.2.1 中所述打开 CCS 工程。
2. 在构建过程中，在 `settings.h` 文件中设置了以下定义。

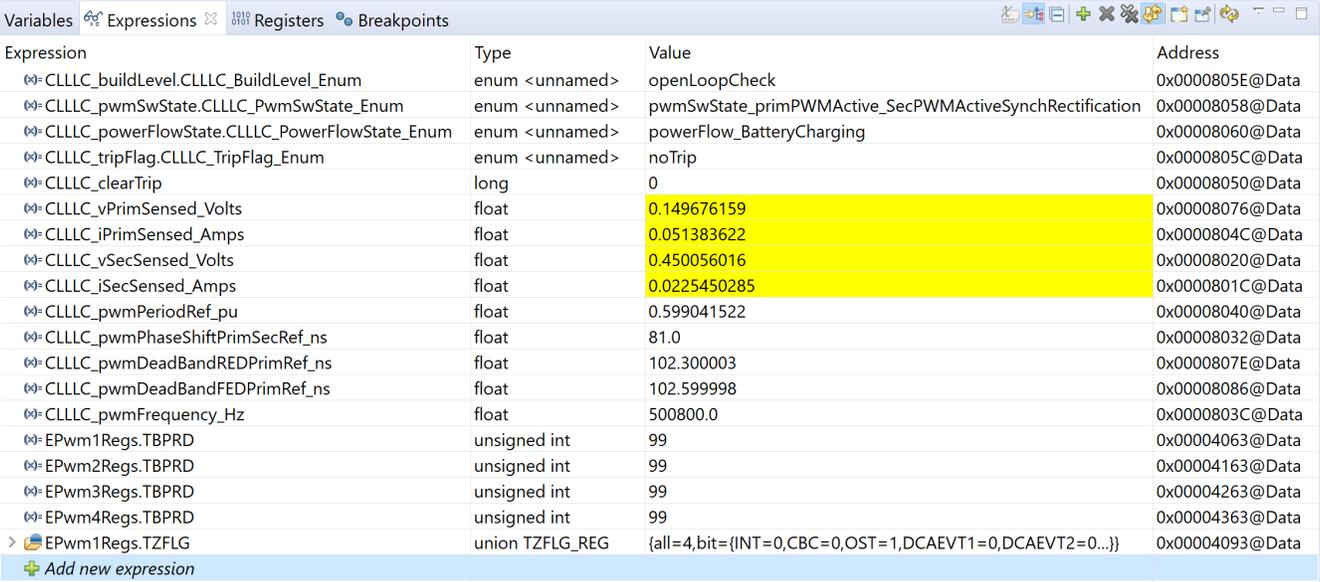
```
#if CLLLC_LAB == 2
#define CLLLC_CONTROL_RUNNING_ON CLA_CORE
#define CLLLC_POWER_FLOW CLLLC_POWER_FLOW_PRIM_SEC
#define CLLLC_INCR_BUILD CLLLC_OPEN_LOOP_BUILD
#define CLLLC_TEST_SETUP CLLLC_TEST_SETUP_RES_LOAD
#define CLLLC_PROTECTION CLLLC_PROTECTION_ENABLED

#if CLLLC_SFRA_ALLOWED == 1
#define CLLLC_SFRA_TYPE CLLLC_SFRA_VOLTAGE
#else
#define CLLLC_SFRA_TYPE CLLLC_SFRA_DISABLED
#endif

#define CLLLC_SFRA_AMPLITUDE (float32_t)CLLLC_SFRA_INJECTION_AMPLITUDE_LEVEL2
#endif
```

5.2.2.2.2 生成和加载工程以及设置调试环境

1. 右键点击工程名称，然后点击 **Rebuild Project**。
2. 将成功生成项目。
3. 在 **Project Explorer** 中，确保在 **targetConfigs** 项下将正确的目标配置文件设置为“Active”（请参阅图 5-5）。
4. 然后，点击 **Run → Debug**（运行 → 调试）以启动调试会话。对于双 CPU 器件，可能会显示一个窗口，用户可以在其中选择要对哪个 CPU 执行调试。在本例中，选择 **CPU1**。
5. 然后工程将加载到器件上，并且 **CCS** 调试视图将变为激活状态。代码将在主例程的开始处停止。
6. 要在监视/表达式窗口中添加变量，请点击 **View → Scripting Console** 以打开脚本控制台对话框。在该控制台的右上角，点击 **open**，然后浏览到位于工程文件夹内的 **setupdebugenv_lab2and7.js** 脚本文件。该文件将采用调试系统所需的相应变量填充监视窗口。
7. 点击监视窗口中的 **Continuous Refresh** 按钮 ，以持续更新控制器中的值。监视窗口如图 5-9 所示。



Expression	Type	Value	Address
CLLLC_buildLevel.CLLLC_BuildLevel_Enum	enum <unnamed>	openLoopCheck	0x0000805E@Data
CLLLC_pwmSwState.CLLLC_PwmSwState_Enum	enum <unnamed>	pwmSwState_primPWMActive_SecPWMActiveSynchRectification	0x00008058@Data
CLLLC_powerFlowState.CLLLC_PowerFlowState_Enum	enum <unnamed>	powerFlow_BatteryCharging	0x00008060@Data
CLLLC_tripFlag.CLLLC_TripFlag_Enum	enum <unnamed>	noTrip	0x0000805C@Data
CLLLC_clearTrip	long	0	0x00008050@Data
CLLLC_vPrimSensed_Volts	float	0.149676159	0x00008076@Data
CLLLC_iPrimSensed_Amps	float	0.051383622	0x0000804C@Data
CLLLC_vSecSensed_Volts	float	0.450056016	0x00008020@Data
CLLLC_iSecSensed_Amps	float	0.0225450285	0x0000801C@Data
CLLLC_pwmPeriodRef_pu	float	0.599041522	0x00008040@Data
CLLLC_pwmPhaseShiftPrimSecRef_ns	float	81.0	0x00008032@Data
CLLLC_pwmDeadBandREDPrimRef_ns	float	102.300003	0x0000807E@Data
CLLLC_pwmDeadBandFEDPrimRef_ns	float	102.599998	0x00008086@Data
CLLLC_pwmFrequency_Hz	float	500800.0	0x0000803C@Data
EPwm1Regs.TBPRD	unsigned int	99	0x00004063@Data
EPwm2Regs.TBPRD	unsigned int	99	0x00004163@Data
EPwm3Regs.TBPRD	unsigned int	99	0x00004263@Data
EPwm4Regs.TBPRD	unsigned int	99	0x00004363@Data
EPwm1Regs.TZFLG	union TZFLG_REG	{all=4,bit={INT=0,CBC=0,OST=1,DCAEVT1=0,DCAEVT2=0...}}	0x00004093@Data

图 5-9. 实验 2 表达式窗口

5.2.2.2.3 使用实时仿真

实时仿真是一项特殊仿真功能，允许在 **MCU 运行时** 更新 **Code Composer Studio** 内的窗口。该功能不但可实现图形和监视视图更新，而且使用户能够改变监视或存储器窗口中的值，并且无需停止处理器即可查看这些更改对系统的影响。

1. 通过将鼠标悬停在水平工具栏中的按钮上并点击  按钮来启用实时模式。

Enable Silicon Real-time Mode (service critical interrupts when halted, allow debugger accesses while running)
2. 可能会显示一个消息框。若是如此，选择 **YES** 来启用调试事件。这会将状态寄存器 1 (ST1) 的位 1 (DGBM 位) 设置为 0。DGBM 是调试使能屏蔽位。当 DGBM 位设置为 0 时，内存和寄存器值将传递到主机处理器以更新调试器窗口。

5.2.2.2.4 运行代码

1. 点击  以运行工程。
2. 现在，通过向 `CLLLC_clearTrip` 变量写入 1 来清除跳闸。
3. 在监视视图中，检查 `CLLLC_vPrimSensed_Volts`、`CLLLC_iPrimSensed_Amps`、`CLLLC_vSecSensed_Volts` 和 `CLLLC_iSecSensed_Amps` 变量是否定期更新。（注意：由于现在未通电，因此这些变量将接近于零。）
4. 现在，缓慢地将 `VPRIM` 直流输入电压从 0V 增加至 400V。确保 `CLLLC_vPrimSensed_Volts` 显示正确的值。
5. 默认情况下，`CLLLC_pwmPeriodRef_pu` 变量被设置为 0.599（如图 5-10 所示），即 500.8kHz。这接近转换器的串联谐振频率；然而，由于实际硬件上的元件变化，该值可能低于或高于串联谐振频率。例如，在图 5-11 中，我们看到频率略低于串联谐振频率。
6. 根据设计的谐振回路增益，`VSEC` 变量将显示接近 300V 的电压。验证 `CLLLC_vSecSensed_Volts` 是否显示正确的电压。这就实现了对电路板上电压检测的验证。

Expression	Type	Value	Address
CLLLC_buildLevel.CLLLC_BuildLevel_Enum	enum <unnamed>	openLoopCheck	0x0000805E@Data
CLLLC_pwmSwState.CLLLC_PwmSwState_Enum	enum <unnamed>	pwmSwState_primPWMActive_SecPWMActiveSynchRectification	0x00008058@Data
CLLLC_powerFlowState.CLLLC_PowerFlowState_Enum	enum <unnamed>	powerFlow_BatteryCharging	0x00008060@Data
CLLLC_tripFlag.CLLLC_TripFlag_Enum	enum <unnamed>	noTrip	0x0000805C@Data
CLLLC_clearTrip	long	0	0x00008050@Data
CLLLC_vPrimSensed_Volts	float	403.606567	0x00008076@Data
CLLLC_iPrimSensed_Amps	float	4.82742071	0x0000804C@Data
CLLLC_vSecSensed_Volts	float	296.258301	0x00008020@Data
CLLLC_iSecSensed_Amps	float	6.27206373	0x0000801C@Data
CLLLC_pwmPeriodRef_pu	float	0.599041522	0x00008040@Data
CLLLC_pwmPhaseShiftPrimSecRef_ns	float	81.0	0x00008032@Data
CLLLC_pwmDeadBandREDPrimRef_ns	float	102.300003	0x0000807E@Data
CLLLC_pwmDeadBandFEDPrimRef_ns	float	102.599998	0x00008086@Data
CLLLC_pwmFrequency_Hz	float	500800.0	0x0000803C@Data
EPwm1Regs.TBPRD	unsigned int	99	0x00004063@Data
EPwm2Regs.TBPRD	unsigned int	99	0x00004163@Data
EPwm3Regs.TBPRD	unsigned int	99	0x00004263@Data
EPwm4Regs.TBPRD	unsigned int	99	0x00004363@Data
EPwm1Regs.TZFLG	union TZFLG_REG	{all=0,bit={INT=0,CBC=0,OST=0,DCAEVT1=0,DCAEVT2=0...}}	0x00004093@Data

图 5-10. 实验 2 表达式窗口，谐振时

1. 在测试条件中指定负载的情况下，对于 `CLLLC_iPrimSensed_Amps`，来自 PRIM 和 SEC 侧的电流将接近 4.8A，对于 `CLLLC_iSecSensed_Amps`，来自 PRIM 和 SEC 侧的电流将接近 6.8A。

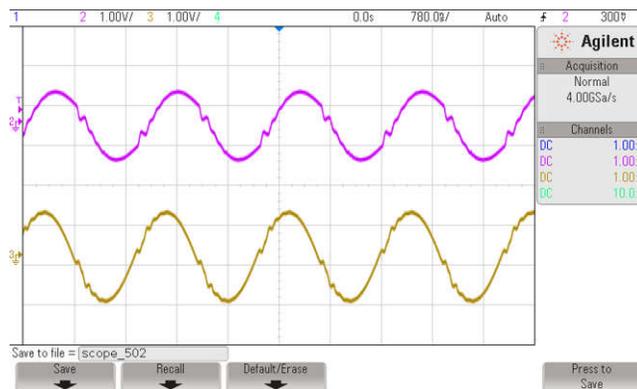


图 5-11. 实验 2，谐振时的初级 (ch2) 和次级 (ch3) 电流

- 接下来，要查看不同频率（即高于谐振和低于谐振）下的运行情况，请将 `CLLLC_pwmPeriodRef_pu` 变量更改为 `0.47`，这将对应于 `639kHz` 的频率。图 5-12 展示了该条件下的波形。

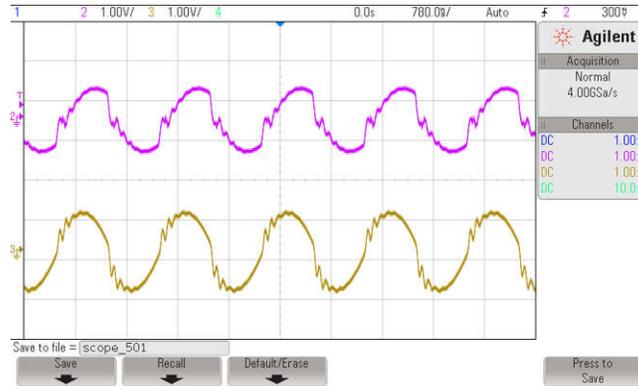


图 5-12. 实验 2，高于串联谐振频率时的初级 (ch2) 和次级 (ch3) 电流

- 接下来，通过输入 `0.8` 作为 `CLLLC_pwmPeriodRef_pu`（这将使生成的频率为 `374kHz`）来测试低于串联谐振频率时的行为。在这种情况下，初级电流将变得不连续，次级侧占空比将调制以实现二极管仿真，如图 5-13 所示。
- 这在基本层面上验证 PWM 驱动器和硬件连接。

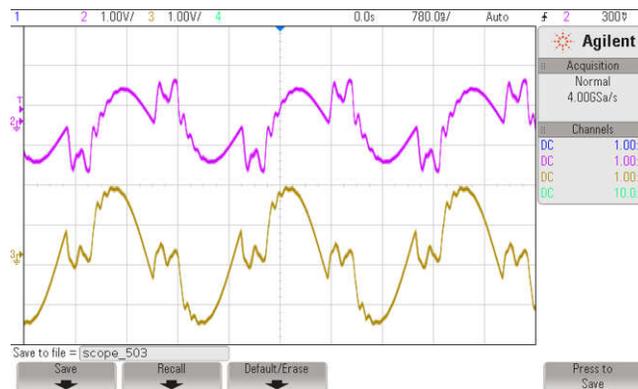


图 5-13. 实验 2，低于串联谐振频率时的初级 (ch2) 和次级 (ch3) 电流

5.2.2.2.5 测量电压环路的 SFRA 装置

- SFRA 集成在该构建的软件中，用于测量受控体响应，然后可以利用该响应来设计补偿器。要运行 SFRA，请使工程保持运行状态，然后导航至 `<Install directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\SFRA_GUI.exe`
- 在 SFRA GUI 上选择器件对应的各个选项；例如，对于 F280039，请选择浮点。点击“Setup Connection”。在弹出窗口中，取消选择 `boot-on-connect` 选项，然后选择合适的 COM 端口。点击 `OK`。返回到 SFRA GUI，然后点击 `Connect`。
- SFRA GUI 将连接到器件。现在可以通过点击 `Start Sweep` 启动 SFRA 扫描。完整的 SFRA 扫描需要几分钟才能完成。可以通过查看 SFRA GUI 上的进度条以及检查控制卡背面用于指示 UART 活动的蓝色 LED 闪烁情况来监视活动。完成之后，将显示包含测量值的图，如图 5-14 所示。（请注意，在实验中进行开环测量是无效的，因为环路不是闭合的。用户只能参考受控体测量值。）

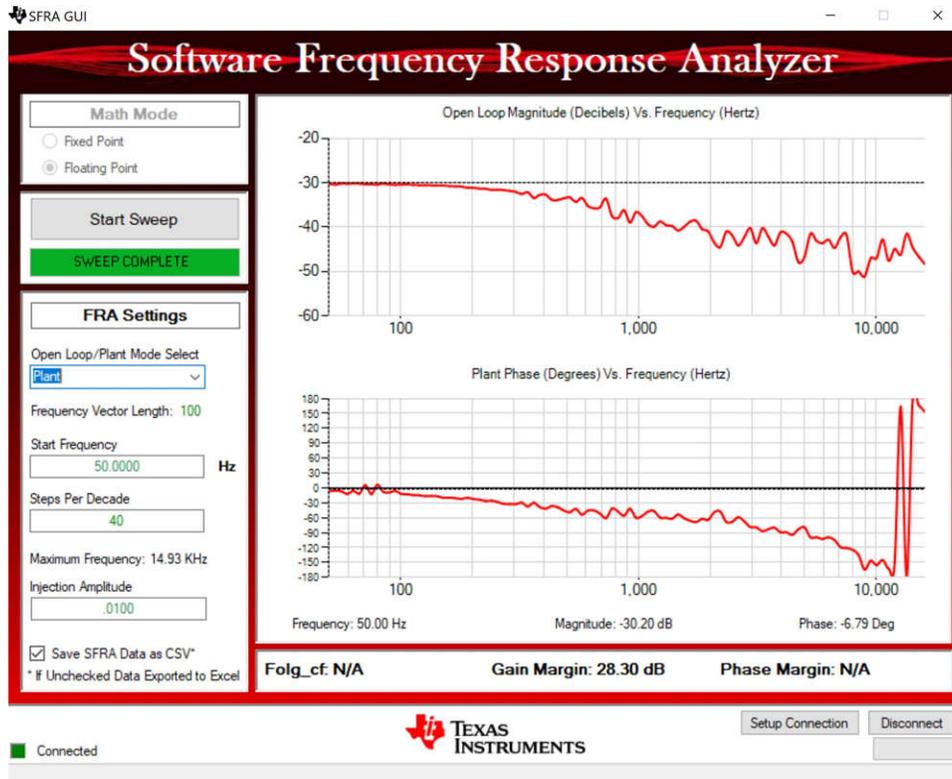


图 5-14. 闭合电压环路的 SFRA 开环图 (V_{prim} 400V , V_{sec} 300V , 功率 1.972kW , F_{sw} 500kHz)

频率响应数据也保存在工程文件夹的 SFRA 数据文件夹下，并根据 SFRA 运行的时间添加时间戳。SFRA 可以在不同的频率设定条件下运行来涵盖系统的工作范围。在下一个实验中将使用这些测量图来设计补偿器；因此，请记住该时间戳，或者将 *SFRA.csv* 文件重命名一个合适的名称以便于识别。

在不同的频率点重复分析，在不同的频率点受控体增益将会不同，有关在 333kHz 测量的增益，请参阅图 5-15，有关在 680kHz 测量的增益，请参阅图 5-16。因此，需要选择一个在转换器的整个频率范围内都保持稳定的补偿器。所有运行都将保存在 CSV 文件中，然后可以导入到补偿设计器中以检查整个工作范围内的稳定性。

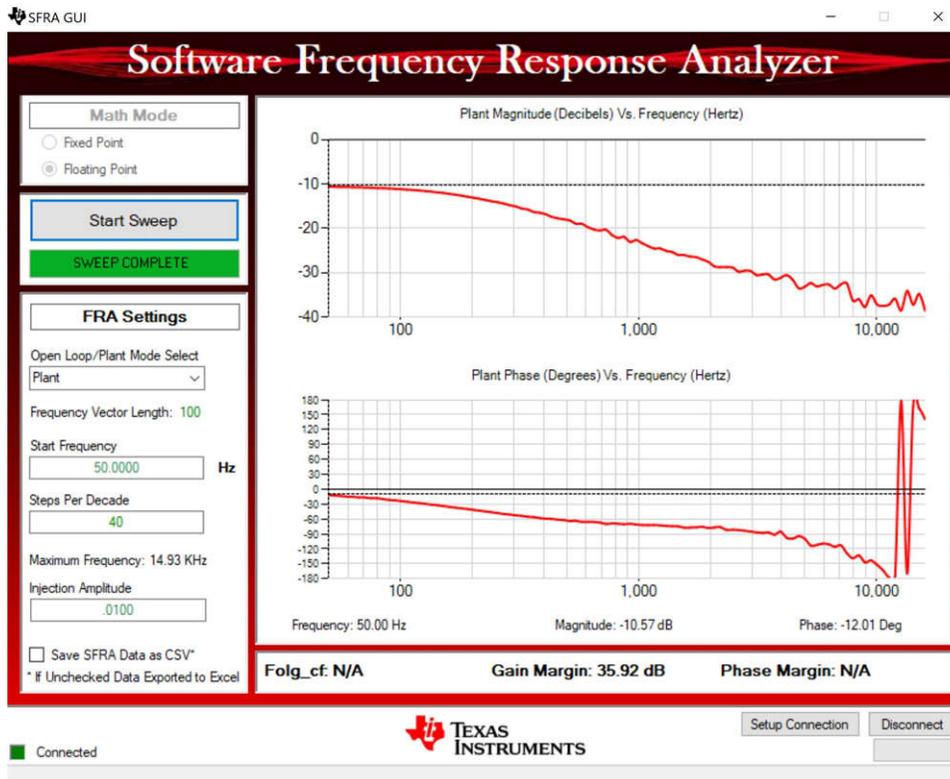


图 5-15. 闭合电压环路的 SFRA 开环图 (V_{prim} 400V , V_{sec} 320V , 功率 2.174kW , F_{sw} 333kHz)

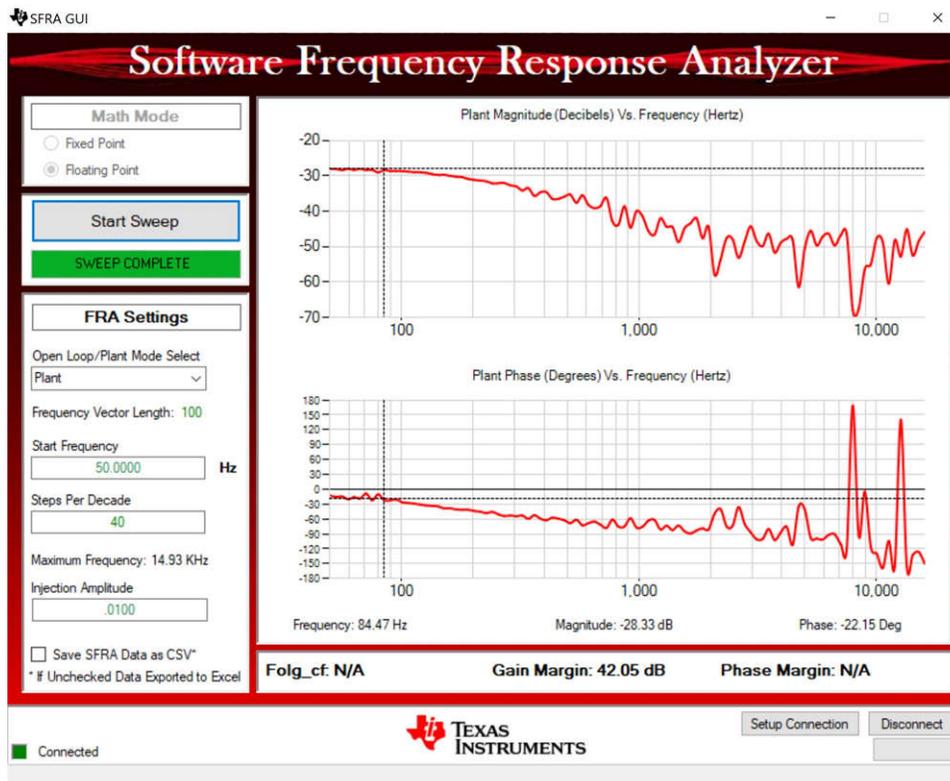


图 5-16. 闭合电压环路的 SFRA 开环图 (V_{prim} 400V , V_{sec} 293V , 功率 1.828kW , F_{sw} 680kHz)

5.2.2.2.6 验证有源同步整流

- 或者，为了验证有源同步整流，用户也可以探测 PWM 信号并查看占空比的变化。要为其连接探头，用户必须首先按如下所述停止功率级，并切断所有被测电路的电源。

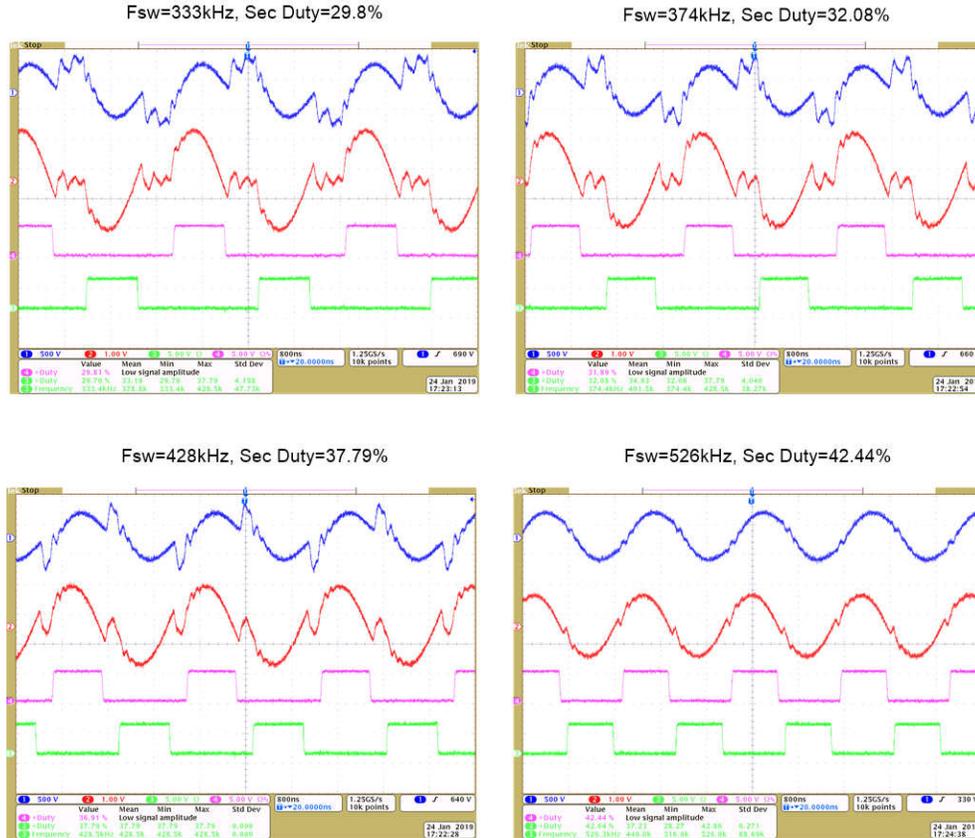


图 5-17. 有源同步整流检查, ch1 → IPRIM_TANK, ch2 → ISEC_TANK, ch3/ch4 → SEC_LEG1_PWMH/L

- 完成后，将输入电压 VPRIM 降至零。观察监视窗口中的电压是否降至零。
- 在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的 *Halt* 按钮 () 或使用 *Target* → *Halt* 来暂停处理器。然后单击 ，使 MCU 退出实时模式。最后，将 MCU 复位 ()。
- 单击 *Terminate Debug Session* () (“Target” → “Terminate all”)，关闭 CCS 调试会话。

5.2.2.2.7 测量电流环路的 SFRA 装置

- 现在，返回到 SYSCFG 页面，并在 SFRA 选项中选择电流来测量电流环路受控体。
- 重新构建工程并重新加载工程。重复 2 (在节 5.2.2.2.1 中) 至节 5.2.2.2.7 (在节 5.2.2.2.5 中)。这次 SFRA 扫描将测量电流环路的受控体。保存该 CSV 文件以便稍后在实验 3 中使用。用户可以在多个点测量受控体，以确保涵盖所有运行条件，图 5-18 展示了 500kHz 下的电流环路受控体测量。

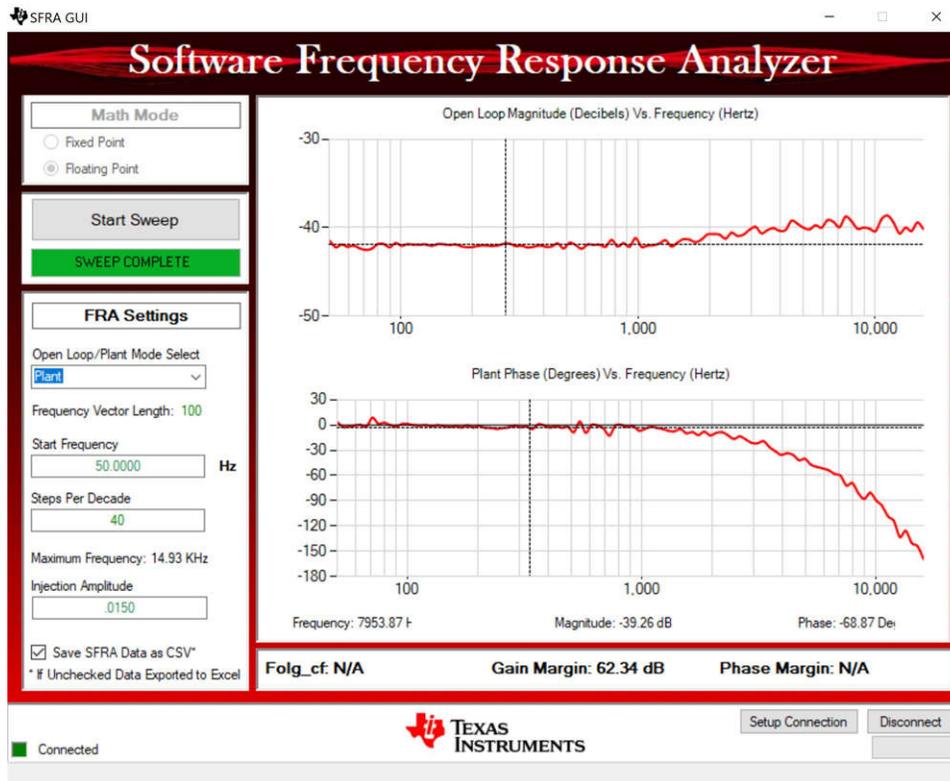


图 5-18. Vprim 400V、VSEC 295V、开关频率 500kHz 和 1887W 下电流环路的 SFRA 受控体测量

3. 至此结束对该构建的检查，成功完成该构建后，请验证以下事项：

- 电压和电流检测以及缩放是否正确
- ISR1、ISR2 和 ISR3 中构建 1 代码的中断生成和执行
- PWM 驱动器和开关
- 电流和电压环路的受控体测量

如果观察到任何问题，则可能需要仔细检查硬件，以消除所有构建问题。

4. 现在可以停止控制器，并终止调试连接。

5. 在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的“Halt”按钮  或使用“Target” → “Halt”来暂停处理器。然后点击 ，使 MCU 退出实时模式。最后，将 MCU 复位 ()。

6. 点击“Terminate Debug Session”  (“Target” → “Terminate all”)，关闭 CCS 调试会话。

5.2.2.3 实验 3. 初级到次级功率流，闭合电压环路检查，次级连接阻性负载

在该实验中，电压环路 G_v 闭合，输出端具有阻性负载。图 5-19 展示了该构建的完整软件图。假设硬件按图 5-7 所示进行设置。

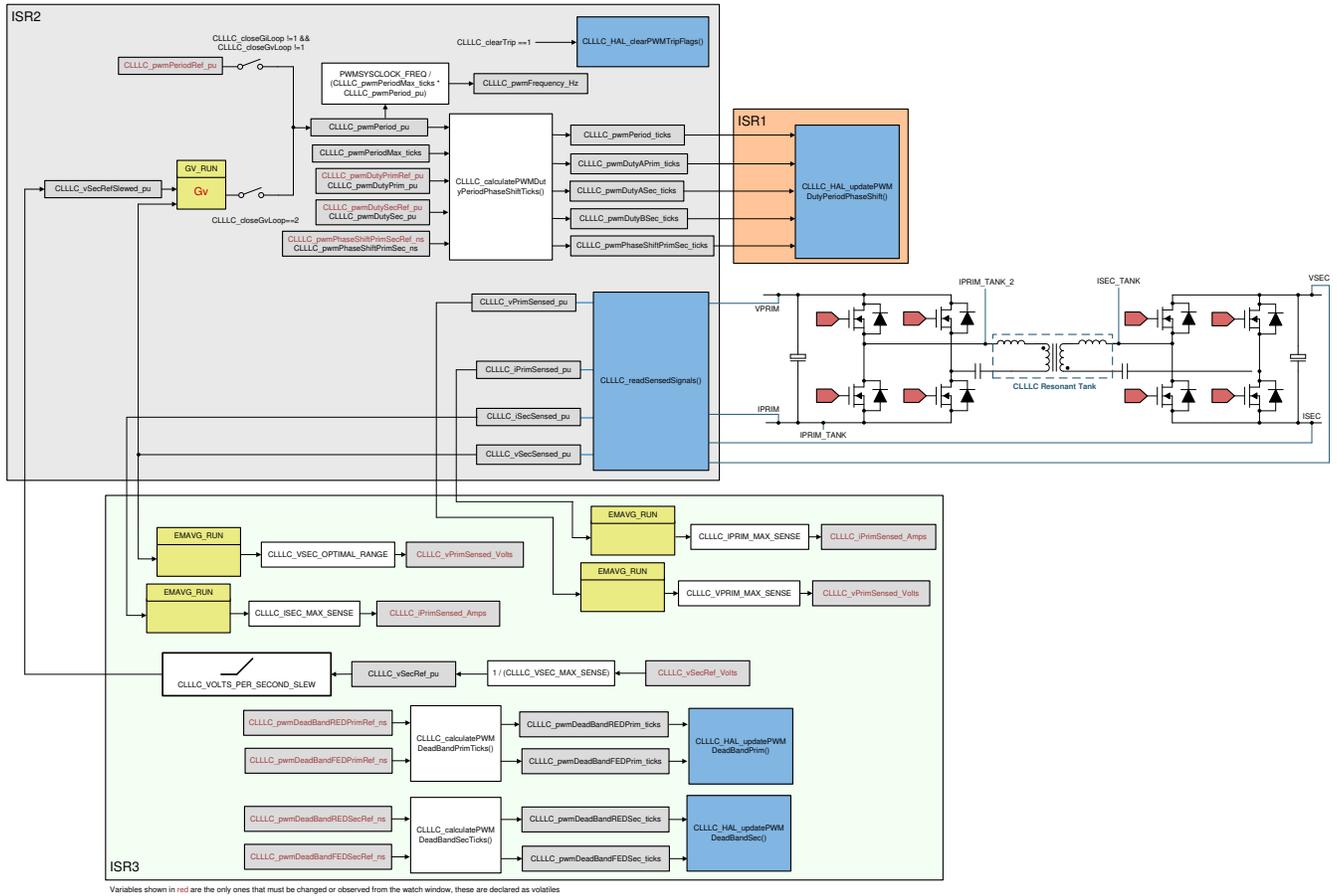


图 5-19. 软件图：闭合电压环路初级到次级功率流

5.2.2.3.1 设置实验 3 的软件选项

1. 要运行该实验，请确保按前面部分（即图 5-7）所述设置硬件。请勿向电路板提供任何高压电源。
2. 打开 “<install Directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\CompDesigner.exe”
3. 然后，补偿设计器将启动并提示用户选择有效的 SFRA 数据文件。将实验 1 中运行的 SFRA 数据导入补偿设计器来设计双极点、双零点补偿器。在该设计迭代过程中最好保持更大的裕度，以确保环路闭合时系统是稳定的。可以检查来自 SFRA 不同运行的受控体数据，以在所有条件下实现稳定的系统，例如，图 5-20 和图 5-21 显示使用设计的补偿器在 500kHz 和 300kHz 下进行的两次运行是稳定的。

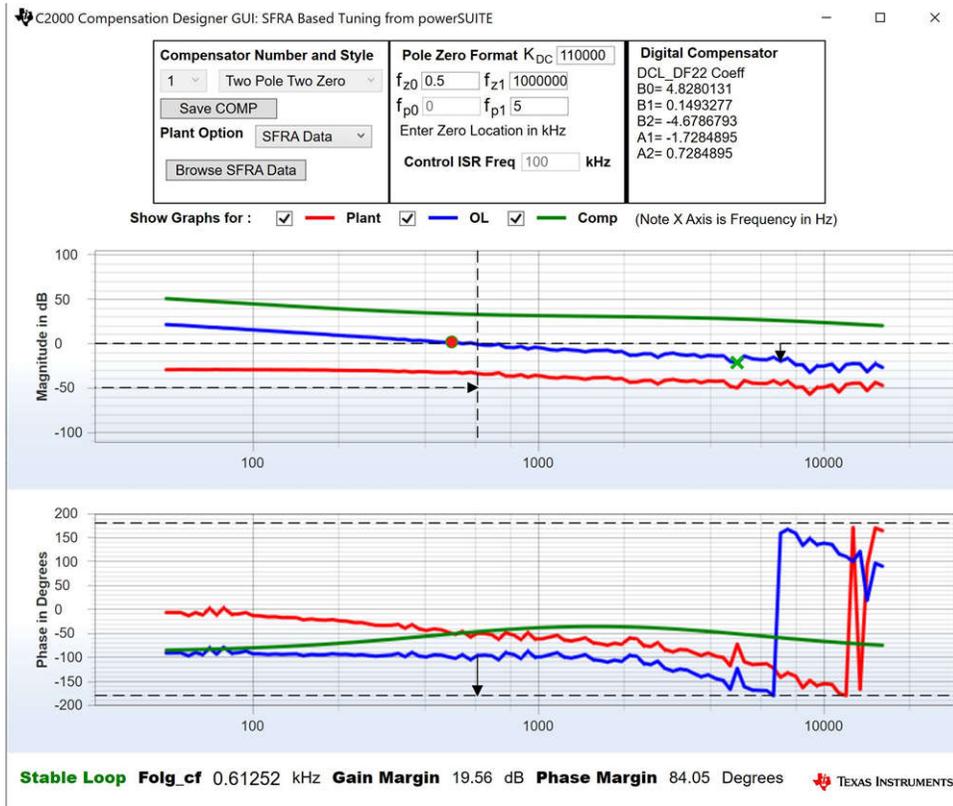


图 5-20. 输出连接阻性负载时电压环路的补偿器设计 (采用基于 SFRA 的受控体测量), 含 500kHz 下的测量数据

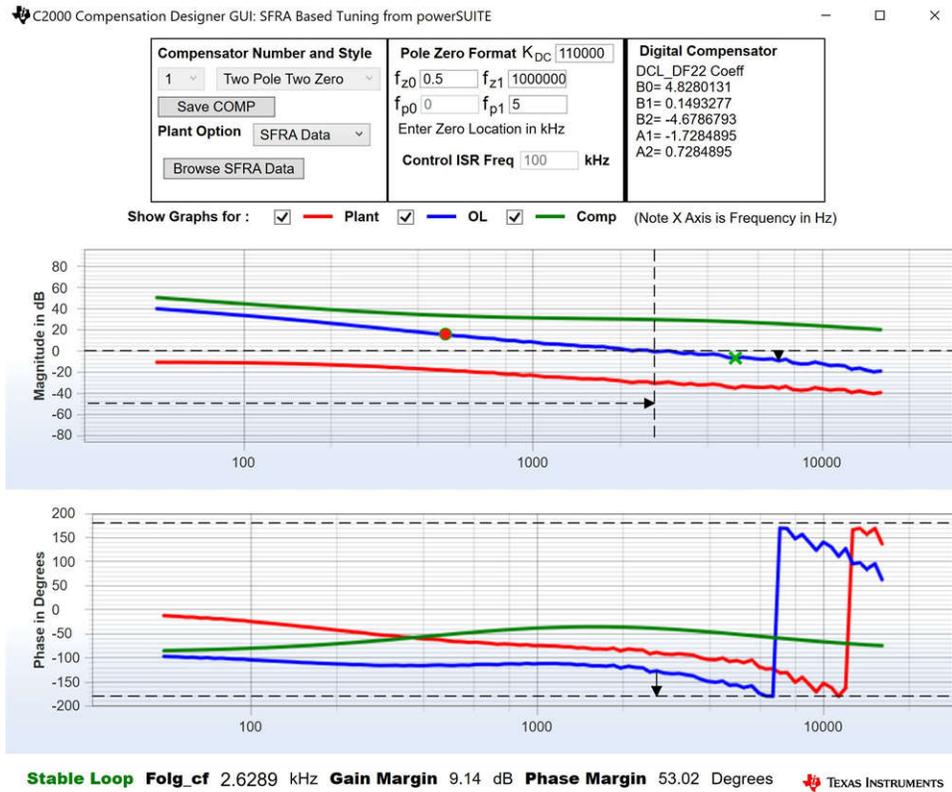


图 5-21. 输出连接阻性负载时电压环路的补偿器设计 (采用基于 SFRA 的受控体测量), 含 333kHz 下的测量数据

备注

调优以 DF22 方式执行；不过，我们在软件中运行 DF13。之所以这么做，是因为 DF13 软启动更容易，而 DF22 结构则无法实现软启动。两种情况下的系数是相同的。在编写本文档时，DCL 中不提供 DF12 结构。

4. 对补偿器设计感到满意后，可以在 `CLLLC_settings.h` 文件中更新补偿器值。
5. 关闭补偿设计器
6. 在构建过程中，在 `settings.h` 文件中设置了以下定义。

```
#if CLLLC_LAB == 3 #define CLLLC_CONTROL_RUNNING_ON CLA_CORE #define CLLLC_POWER_FLOW
CLLLC_POWER_FLOW_PRIM_SEC #define CLLLC_INCR_BUILD CLLLC_CLOSED_LOOP_BUILD #define
CLLLC_CONTROL_MODE CLLLC_VOLTAGE_MODE #define CLLLC_TEST_SETUP CLLLC_TEST_SETUP_RES_LOAD
#define CLLLC_PROTECTION CLLLC_PROTECTION_ENABLED #if CLLLC_SFRA_ALLOWED == 1 #define
CLLLC_SFRA_TYPE CLLLC_SFRA_VOLTAGE #else #define CLLLC_SFRA_TYPE CLLLC_SFRA_DISABLED #endif
#define CLLLC_SFRA_AMPLITUDE (float32_t)CLLLC_SFRA_INJECTION_AMPLITUDE_LEVEL1 #endif
```

5.2.2.3.2 生成和加载工程以及设置调试环境

1. 现在，右键点击工程名称，然后点击 *Rebuild Project*。
2. 将成功生成项目。
3. 点击 *Run* → *Debug* 以启动调试会话。对于双 CPU 器件，可能会显示一个窗口，用户可以在其中选择要对哪个 CPU 执行调试。在本例中，选择“CPU1”。
4. 然后工程将加载到器件上，并且 CCS 调试视图将变为激活状态。代码将在主例程的开始处停止。
5. 要在监视/表达式窗口中添加变量，请点击“View” → “Scripting Console”以打开脚本控制台对话框。在该控制台的右上角，点击 *open* 浏览到位于工程文件夹内的 `setupdebugenv_lab3.js` 脚本文件。该文件将采用调试系统所需的相应变量填充监视窗口。
6. 点击监视窗口中的“Continuous Refresh”按钮 ，以持续更新控制器中的值。

7. 通过将鼠标悬停在水平工具栏中的按钮上并点击



Enable Silicon Real-time Mode (service critical interrupts when halted, allow debugger accesses while running)

按钮来启用实时模式。

5.2.2.3.3 运行代码

1. 点击  以运行工程
2. 现在，通过向 `CLLLC_clearTrip` 变量写入 `1` 来清除跳闸。由于 `CLLLC_closeGvLoop` 变量尚未被设置为 `0`，因此转换器将在开环模式下运行。由于固件中未实现软启动，因此首先手动软启动初级侧和次级侧的电压。
3. 在监视视图中，检查 `CLLLC_vPrimSensed_Volts`、`CLLLC_iPrimSensed_Amps`、`CLLLC_vSecSensed_Volts` 和 `CLLLC_iSecSensed_Amps` 变量是否定期更新。（注意：由于现在未通电，因此这些变量将接近于零。）
4. 现在，缓慢地将输入 `PRIM` 直流电压从 `0V` 增加至 `400V`，以软启动转换器。确保 `CLLLC_vPrimSensed_Volts` 显示正确的 `VPRIM` 值（即接近 `400V`）。
5. 默认情况下，`CLLLC_pwmPeriodRef_pu` 变量被设置为 `0.599`，即 `500.8kHz`。这接近转换器的串联谐振频率；然而，由于实际硬件上的元件变化，该值可能低于或高于串联谐振频率。
6. 对于 `400V` 初级输入，匝数比为 `1.33`，`CLLLC_vSecSensed_Volts` 变量将接近 `300V`。将 `CLLLC_vSecRef_Volts` 变量设置为 `300V`。
7. 现在，将 `CLLLC_closeGvLoop` 变量设置为 `1`。这将使电压环路闭合，控制器现在将尝试调节电压。
8. 通过将 `CLLLC_vSecRef_Volts` 从 `295V` 改变为 `320V` 来测试闭环运行。用户将观察到 `CLLLC_vSecSensed_Volts` 会跟踪此命令基准。转换器将在低于串联谐振时、谐振时和高于谐振时运行。现在，将电压更改回至 `300V` 以运行 `SFRA`。

5.2.2.3.4 测量闭合电压环路的 SFRA

1. `SFRA` 集成在该构建的软件中，通过在硬件上进行测量来验证设计的补偿器是否可以提供足够的增益裕度和相位裕度。要运行 `SFRA`，请使工程保持运行状态，然后导航至 `<Install directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfragui\SFRA_GUI.exe`
2. 在 `SFRA GUI` 上选择器件对应的各个选项；例如，对于 `F280039`，请选择浮点。点击“`Setup Connection`”。在弹出窗口中，取消选择“`boot-on-connect`”选项，然后选择合适的 `COM` 端口。点击“`Ok`”。返回到 `SFRA GUI`，然后点击“`Connect`”。
3. `SFRA GUI` 将连接到器件。现在可以通过点击 `Start Sweep` 启动 `SFRA` 扫描。完整的 `SFRA` 扫描需要几分钟才能完成。可以通过查看 `SFRA GUI` 上的进度条以及检查控制卡背面用于指示 `UART` 活动的蓝色 `LED` 闪烁情况来监视活动。完成之后，会显示一个包含开环图的图，如图 [5-22](#) 所示。

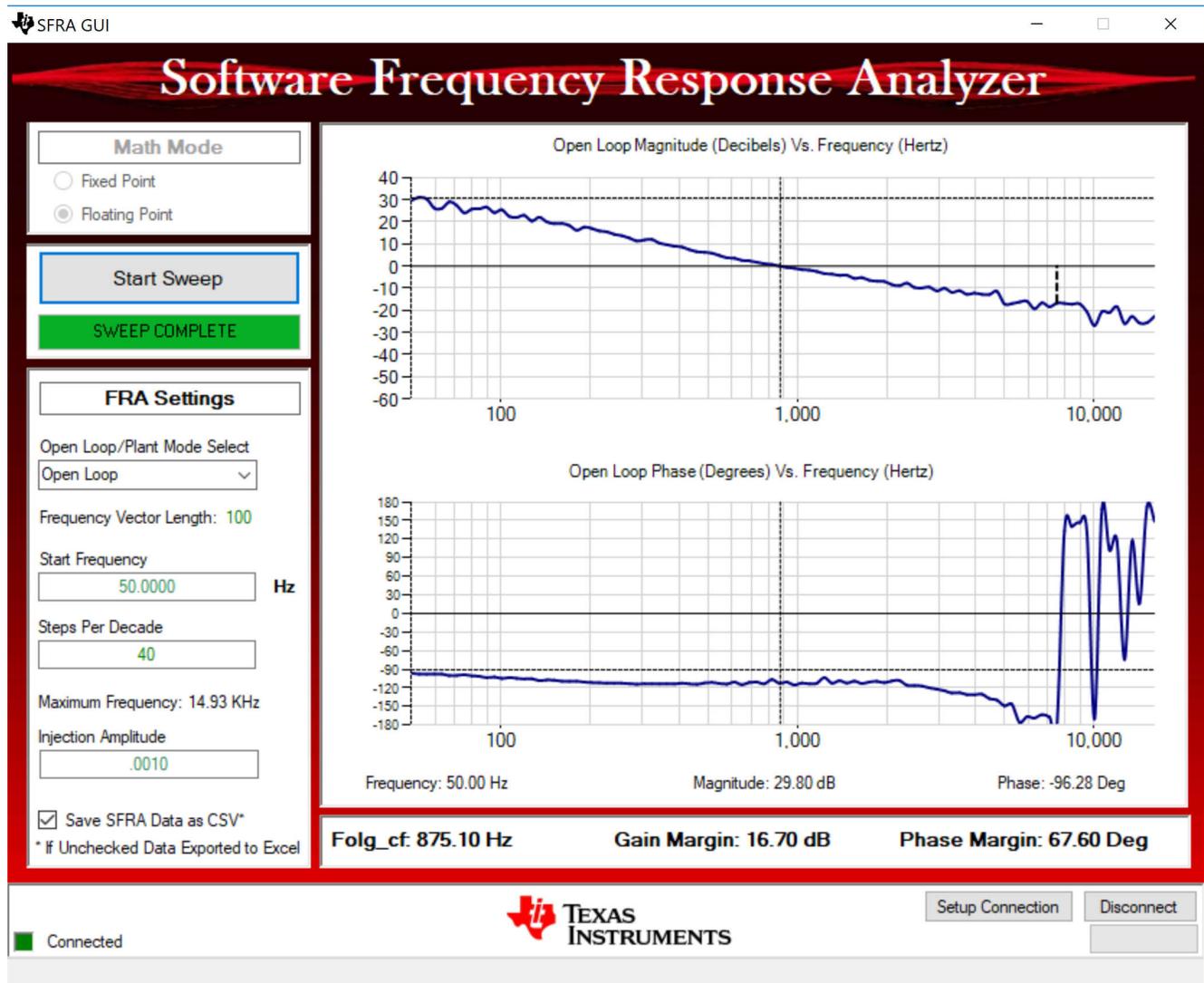


图 5-22. 闭合电压环路的 SFRA 开环图 (V_{prim} 400V , V_{sec} 300V , 功率 1.972kW , 输出端具有阻性负载)

频率响应数据也保存在工程文件夹的 SFRA 数据文件夹下，并根据 SFRA 运行的时间添加时间戳。

数据与设计的补偿器非常匹配，但预期偏差是合理的，因为开环测量很容易出现误差，这是由于小信号注入可能会导致转换器的直流点漂移。

在不同电压下测试 SFRA，以验证系统在可运行范围内是否稳定。

4. 这可以验证电压环路设计。
5. 要使系统安全停止，请将输入 V_{PRIM} 电压降至零。观察监视窗口中的电压和电流是否降至零。
6. 在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的“Halt”按钮  或使用“Target” → “Halt”来暂停处理器。然后点击 ，使 MCU 退出实时模式。最后，将 MCU 复位 ()。
7. 点击“Terminate Debug Session”  (“Target” → “Terminate all”)，关闭 CCS 调试会话。

5.2.2.4 实验 4. 初级到次级功率流，闭合电流环路检查，次级连接阻性负载

在该实验中，输出电流控制环路是闭合的。图 5-23 展示了该构建的完整软件图。假设硬件按图 5-7 所示进行设置。

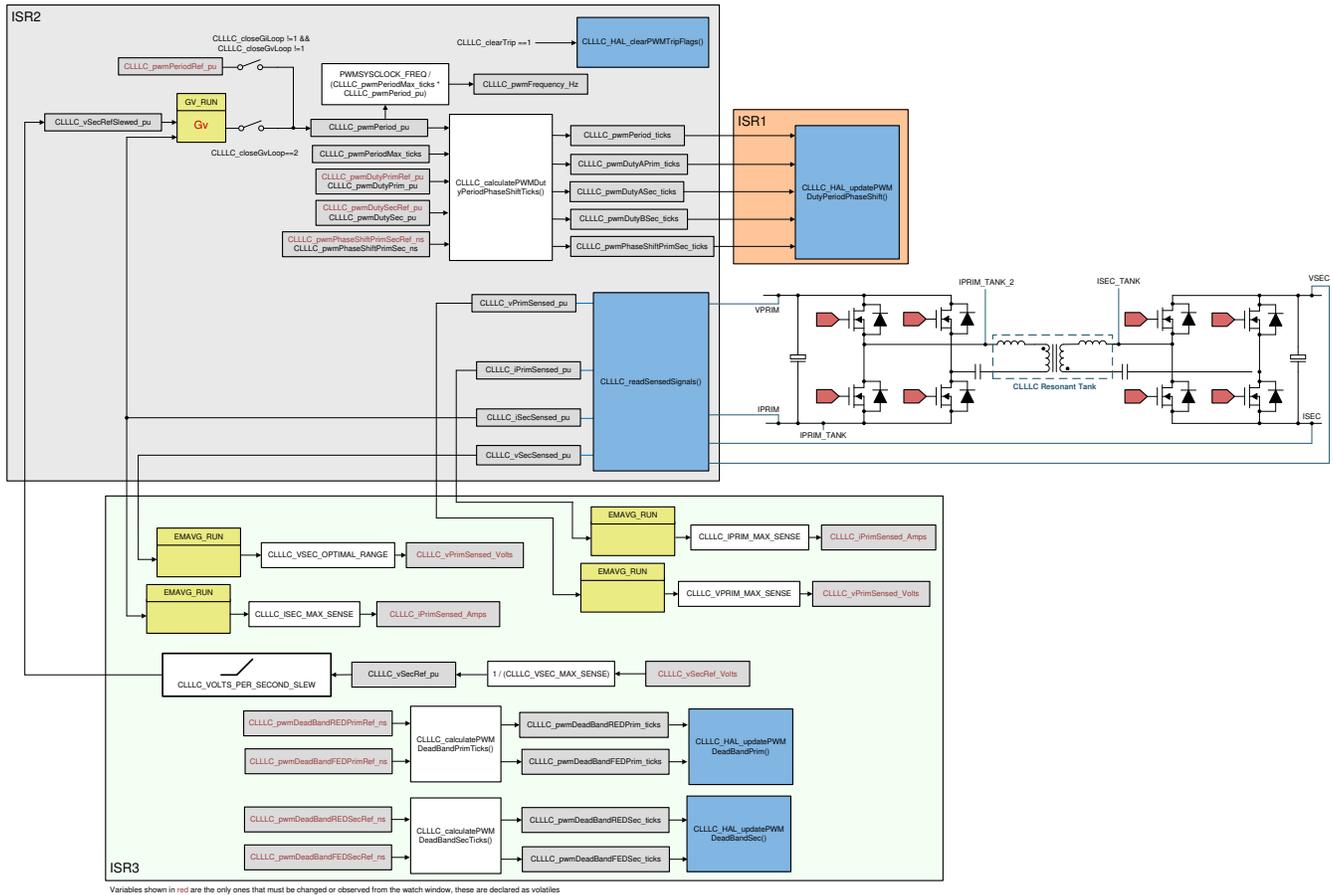


图 5-23. 实验 4 控制软件图：闭合电流环路

5.2.2.4.1 设置实验 4 的软件选项

1. 打开 `<install Directory>\IC2000Ware_DigitalPower_SDK_<version>\libraries\sfr\gui\CompDesigner.exe`
2. 然后，补偿设计器将启动并提示用户选择有效的 SFRA 数据文件。将实验 1 中运行的电流环路 SFRA 数据导入补偿设计器来设计双极点、双零点补偿器。在该设计迭代过程中最好保持更大的裕度，以确保环路闭合时系统是稳定的。可以检查来自 SFRA 不同运行的受控体数据，以在所有条件下实现稳定的系统。

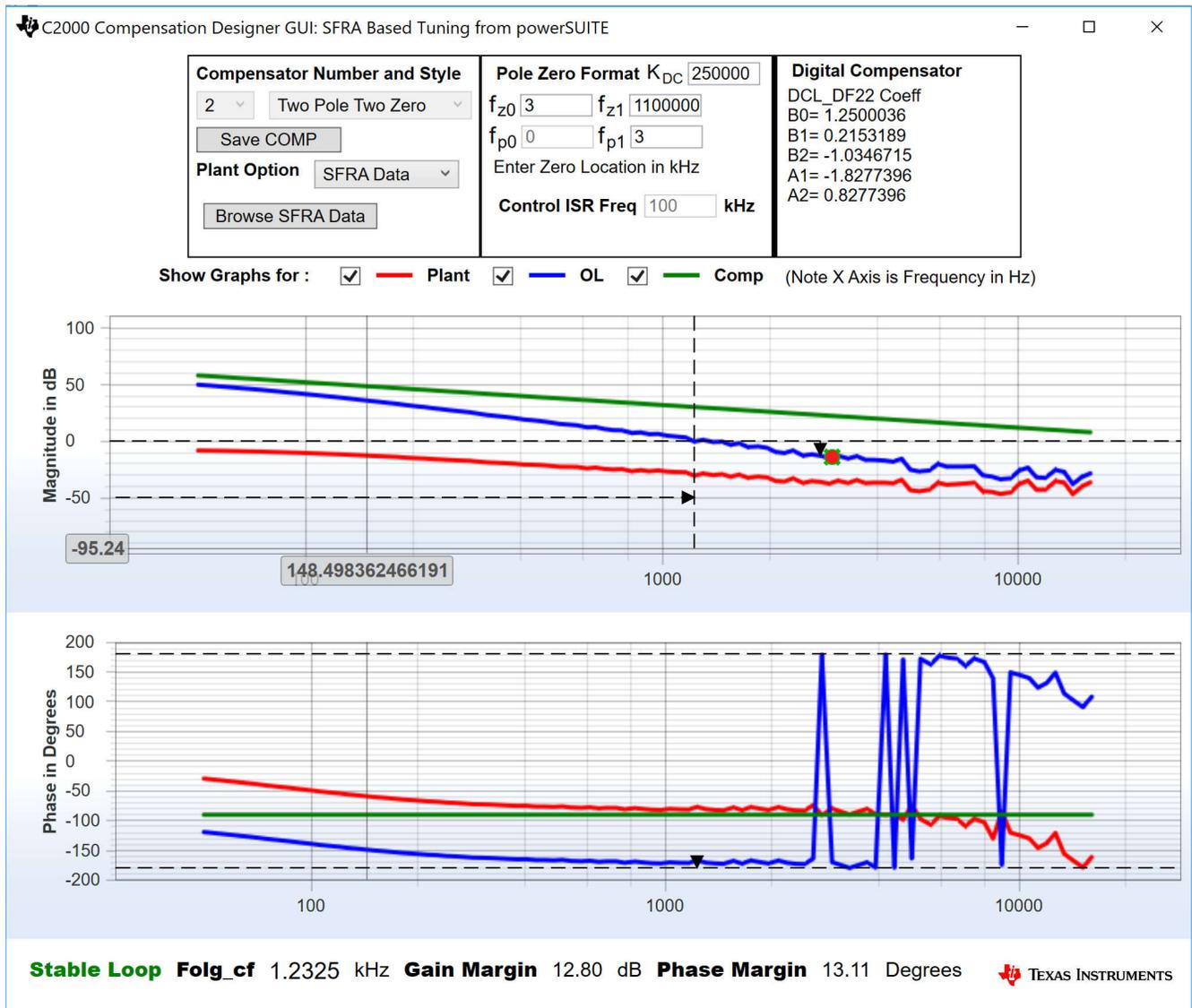


图 5-24. 电流环路的补偿器设计 (采用基于 SFRA 的受控体测量), 实验 4

备注

调优以 DF22 方式执行；不过，我们在软件中运行 DF13。之所以这么做，是因为 DF13 软启动更容易，而 DF22 结构则无法实现软启动。两种情况下的系数是相同的。在编写本文档时，DCL 中不提供 DF12 结构。

3. 对补偿器设计感到满意后，可以在 CLLLC_settings.h 文件中更新补偿器值。
4. 关闭补偿设计器

- 在构建过程中，在 `settings.h` 文件中设置了以下定义。

```
#if CLLLC_LAB == 4 #define CLLLC_CONTROL_RUNNING_ON CLA_CORE #define CLLLC_POWER_FLOW
CLLLC_POWER_FLOW_PRIM_SEC #define CLLLC_INCR_BUILD CLLLC_CLOSED_LOOP_BUILD #define
CLLLC_CONTROL_MODE CLLLC_CURRENT_MODE #define CLLLC_TEST_SETUP CLLLC_TEST_SETUP_RES_LOAD
#define CLLLC_PROTECTION CLLLC_PROTECTION_ENABLED #if CLLLC_SFRA_ALLOWED == 1 #define
CLLLC_SFRA_TYPE CLLLC_SFRA_CURRENT #else #define CLLLC_SFRA_TYPE CLLLC_SFRA_DISABLED #endif
#define CLLLC_SFRA_AMPLITUDE (float32_t)CLLLC_SFRA_INJECTION_AMPLITUDE_LEVEL1 #endif
```

5.2.2.4.2 生成和加载项目以及设置调试

- 现在，右键单击工程名称，然后点击 **Rebuild Project**。
- 将成功生成项目。
- 点击 **Run** → **Debug** 以启动调试会话。对于双 CPU 器件，可能会显示一个窗口，用户可以在其中选择要对哪个 CPU 执行调试。在本例中，选择“CPU1”。
- 然后工程将加载到器件上，并且 CCS 调试视图将变为激活状态。代码将在主例程的开始处停止。
- 要在监视/表达式窗口中添加变量，请点击 **View** → **Scripting Console** 以打开脚本控制台对话框。在该控制台的右上角，点击 **open** 浏览到位于工程文件夹内的 `setupdebugenv_build4c.js` 脚本文件。该文件将采用调试系统所需的相应变量填充监视窗口。
- 点击监视窗口中的 **Continuous Refresh** 按钮 ，以持续更新控制器中的值。
- 通过将鼠标悬停在水平工具栏中的按钮上并点击  按钮来启用实时模式。

Enable Silicon Real-time Mode (service critical interrupts when halted, allow debugger accesses while running)

按钮来启用实时模式。

5.2.2.4.3 运行代码

- 点击  以运行工程
- 通过向 `CLLLC_clearTrip` 变量写入 1 来清除跳闸。由于 `CLLLC_closeGvLoop` 变量尚未被设置为“0”，因此转换器将在开环模式下运行。由于固件中未实现软启动，因此首先手动软启动初级侧和次级侧的电压。
- 在监视视图中，检查 `CLLLC_vPrimSensed_Volts`、`CLLLC_iPrimSensed_Amps`、`CLLLC_vSecSensed_Volts` 和 `CLLLC_iSecSensed_Amps` 变量是否定期更新。（注意：由于现在未通电，因此这些变量将接近于零。）
- 现在，缓慢地将输入 PRIM 直流电压从 0V 增加至 400V，以软启动转换器。确保 `CLLLC_vPrimSensed_Volts` 显示正确的 `VPRIM` 值（即接近 400V）。
- 默认情况下，`CLLLC_pwmPeriodRef_pu` 变量被设置为 0.6，即 500.8kHz。这接近转换器的串联谐振频率；然而，由于实际硬件上的元件变化，该值可能低于或高于串联谐振频率。
- 对于 400V 初级输入，匝数比为 1.33，`CLLLC_vSecSensed_Volts` 变量将接近 300V。此外，对于测试条件中指定的负载，负载将接近 6.5A。将 `CLLLC_iSecRef_Amps` 变量设置为 6.5A。如果由于某种原因测量的电流为 6.5A 以外的值，请将 `Ref` 设置为接近测量值。由于软件中没有软启动，因此保持该基准接近工作点至至关重要。
- 现在，将 `CLLLC_closeGiLoop` 变量设置为 1。这将使电流环路闭合，控制器现在将尝试调节电流。
- 通过将 `CLLLC_iSecRef_Amps` 从 6.3A 改变为 6.8A 来测试闭环运行。用户不能将电流改变太多，因为输出端连接了一个阻性负载，其电压随电流的变化比电池的变化大得多。电压的这种快速增加可以快速将转换器置于超出固定 `VPRIM` 可控范围的范围之内。在该小范围内，用户可以看到电流跟踪。

5.2.2.4.4 测量闭合电流环路的 SFRA

- SFRA 集成在该构建的软件中，通过在硬件上进行测量来验证设计的补偿器是否可以提供足够的增益裕度和相位裕度。要运行 SFRA，请使工程保持运行状态，然后导航至 `<Install directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\SFRA_GUI.exe`。SFRA GUI 将弹出。
- 在 SFRA GUI 上选择器件对应的各个选项；例如，对于 F280039，请选择浮点。点击“Setup Connection”。在弹出窗口中，取消选择“boot-on-connect”选项，选择合适的 COM 端口，然后点击“Ok”。返回到 SFRA GUI，然后点击“Connect”。
- SFRA GUI 将连接到器件。现在可以通过点击 **Start Sweep** 启动 SFRA 扫描。完整的 SFRA 扫描需要几分钟才能完成。可以通过查看 SFRA GUI 上的进度条以及检查控制卡背面用于指示 UART 活动的蓝色 LED 闪烁情况来监视活动。完成之后，会显示一个包含开环图的图，如图 5-25 所示。

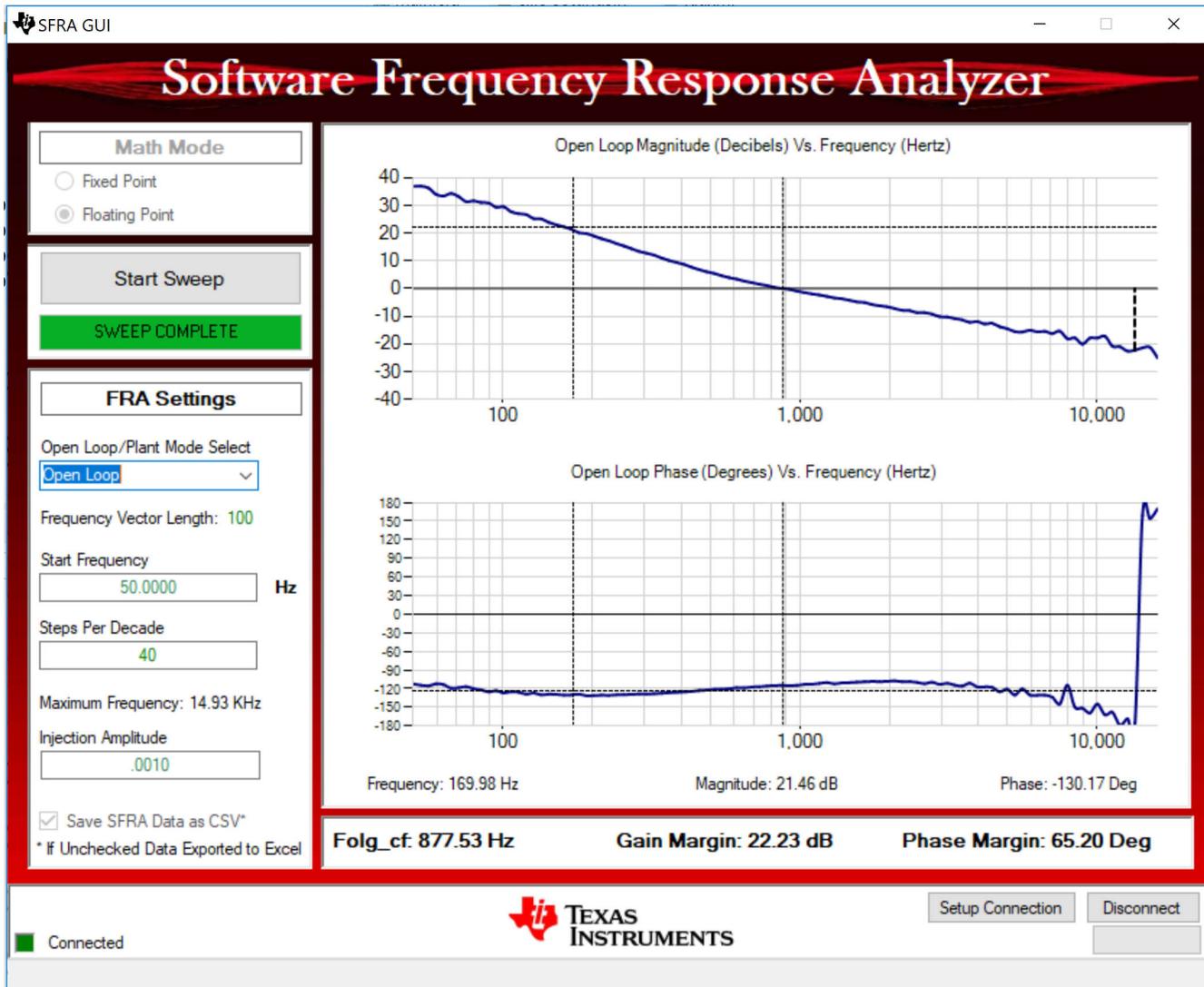


图 5-25. 闭合电流环路的 SFRA 开环图 (V_{prim} 400V , V_{sec} 300V , 功率 1.972kW , 实验 4)

频率响应数据也保存在工程文件夹的 SFRA 数据文件夹下，并根据 SFRA 运行的时间添加时间戳。

数据与设计的补偿器非常匹配，但预期偏差是合理的，因为开环测量很容易出现误差，这是由于小信号注入可能会导致转换器的直流点漂移。

在不同的电流设定点测试 SFRA，确保周期未被钳制，以验证系统在可运行范围内是否稳定。

4. 这可以验证实验 4 电流环路设计。
5. 要使系统安全停止，请将输入 V_{PRIM} 电压降至零。观察监视窗口中的电压和电流是否降至零。
6. 在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的“Halt”按钮  或使用 *Target* → *Halt* 来暂停处理器。然后点击 ，使 MCU 退出实时模式。最后，将 MCU 复位 ()。
7. 点击“Terminate Debug Session”  (“Target” → “Terminate all”)，关闭 CCS 调试会话。

5.2.2.5 实验 5. 初级到次级功率流，闭合电流环路检查，次级连接与电压源并联的阻性负载，以模拟次级侧的电池连接

在该实验中，输出电流控制环路是闭合的，次级连接与电压源并联的阻性负载，以模拟电池连接。假设硬件按图 5-26 所示进行设置。图 5-27 展示了该构建的完整软件块图。

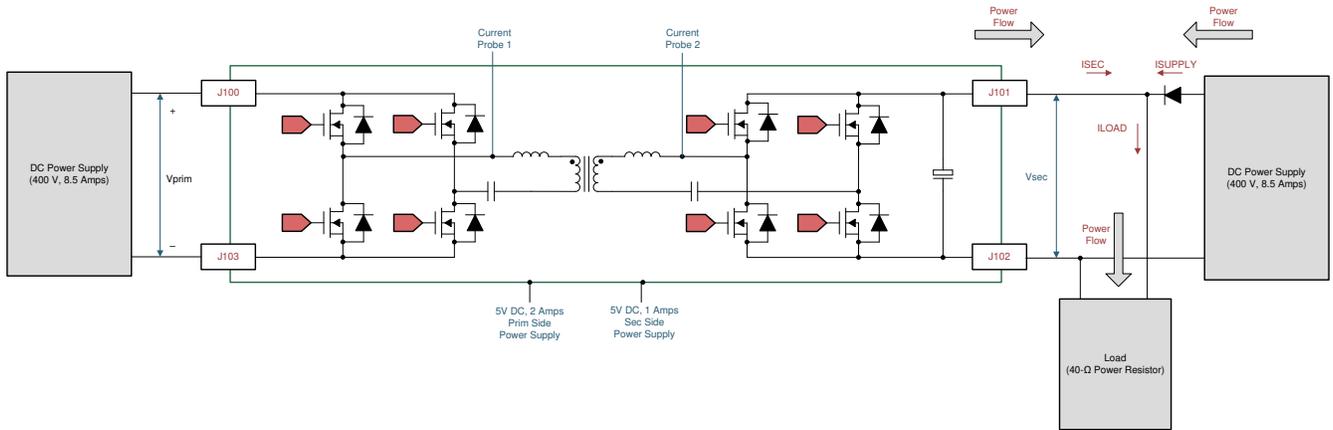


图 5-26. 实验 5 的硬件设置

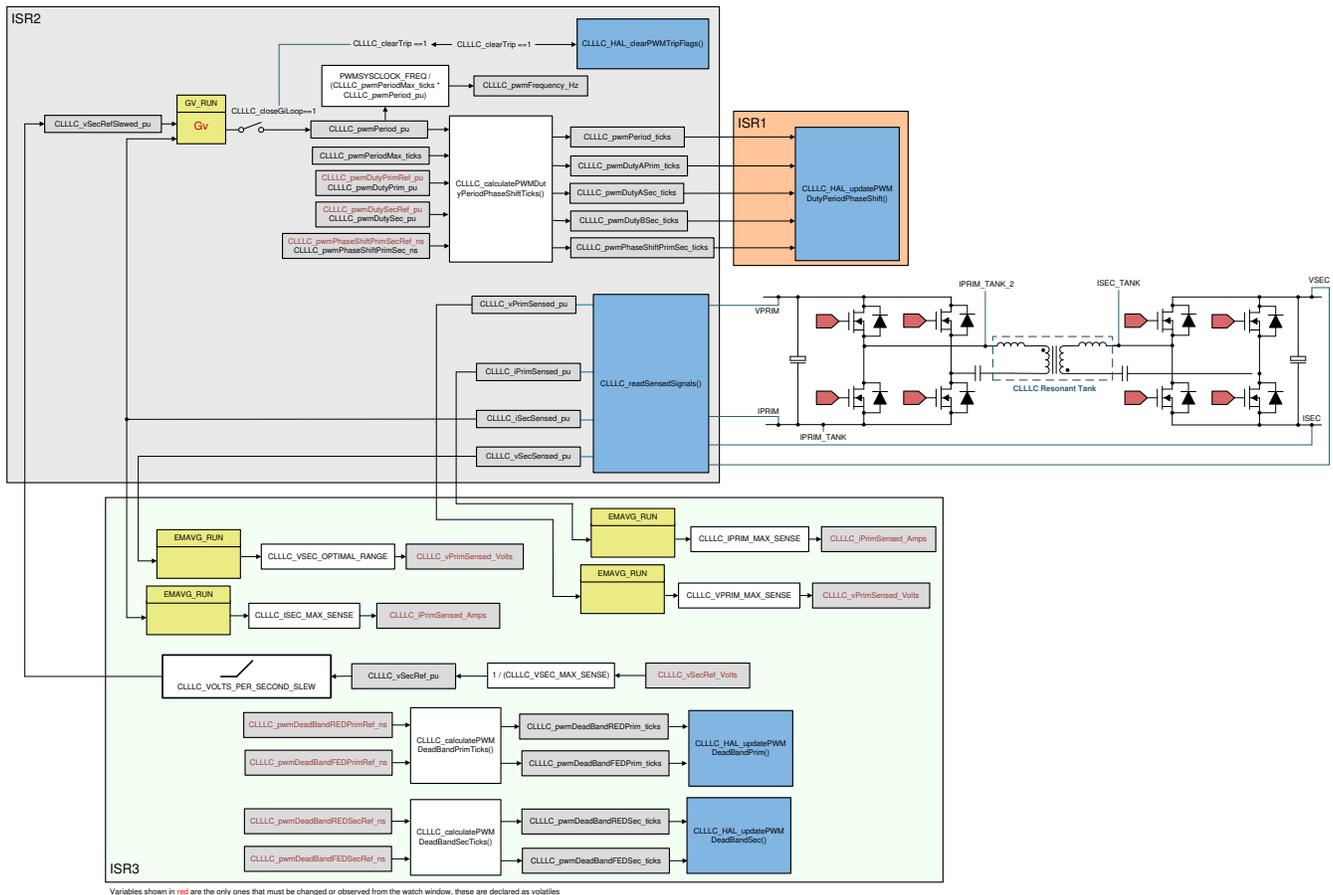


图 5-27. 实验 5 软件图

5.2.2.5.1 设置实验 5 的软件选项

1. 打开 <install Directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\CompDesigner.exe。

5.2.2.5.2 设计电流环路补偿器

1. 然后补偿设计器将启动。目前尚无数学模型；因此，通过使用在该板上完成的调整，设计了以下补偿。仿真电池模式下的受控体将具有更大的增益，因此需要降低系数以适应这种情况。图 5-28 展示了此设计中针对该实验使用的系数。

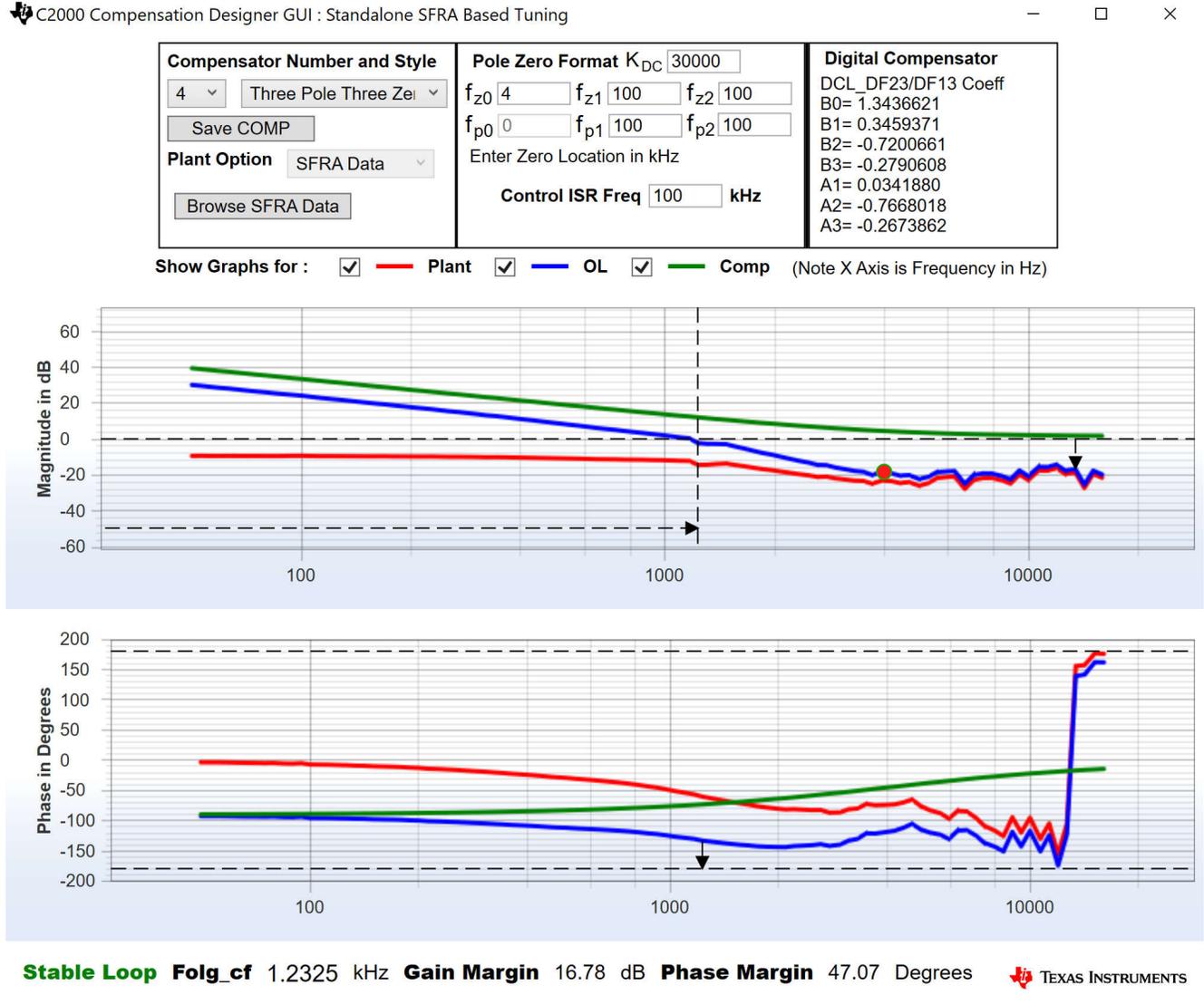


图 5-28. 实验 5，补偿设计器

- 对补偿器设计感到满意后，可以在 CLLLC_settings.h 文件中更新补偿器值。最好采用保守的系数，使其远低于实验 3 中使用的系数。
- 关闭补偿设计器
- 在构建过程中，在 settings.h 文件中设置了以下定义。

```
#if CLLLC_LAB == 5 #define CLLLC_CONTROL_RUNNING_ON 1 #define CLLLC_POWER_FLOW
CLLLC_POWER_FLOW_PRIM_SEC #define CLLLC_INCR_BUILD CLLLC_CLOSED_LOOP_BUILD #define
CLLLC_CONTROL_MODE CLLLC_CURRENT_MODE #define CLLLC_TEST_SETUP
CLLLC_TEST_SETUP_EMULATED_BATTERY #define CLLLC_PROTECTION CLLLC_PROTECTION_ENABLED #if
CLLLC_SFRA_ALLOWED == 1 #define CLLLC_SFRA_TYPE CLLLC_SFRA_CURRENT #else #define
CLLLC_SFRA_TYPE CLLLC_SFRA_DISABLED #endif #define CLLLC_SFRA_AMPLITUDE
(float32_t)CLLLC_SFRA_INJECTION_AMPLITUDE_LEVEL1 #endif
```

5.2.2.5.3 生成和加载项目以及设置调试

1. 现在，右键点击工程名称，然后点击 **Rebuild Project**。
2. 将成功生成项目。
3. 点击 **Run** → **Debug** 以启动调试会话。对于双 CPU 器件，可能会显示一个窗口，用户可以在其中选择要对哪个 CPU 执行调试。在本例中，选择“CPU1”。
4. 然后工程将加载到器件上，并且 CCS 调试视图将变为激活状态。代码将在主例程的开始处停止。
5. 要在监视/表达式窗口中添加变量，请点击 **View** → **Scripting Console** 以打开脚本控制台对话框。在该控制台的右上角，点击 **open** 浏览到位于工程文件夹内的 `setupdebugenv_build4.js` 脚本文件。该文件将采用调试系统所需的相应变量填充监视窗口。
6. 点击监视窗口中的“Continuous Refresh”按钮 ，以持续更新控制器中的值。
7. 通过将鼠标悬停在水平工具栏中的按钮上并点击  按钮来启用实时模式。

Enable Silicon Real-time Mode (service critical interrupts when halted, allow debugger accesses while running)

按钮来启用实时模式。

5.2.2.5.4 运行代码

1. 点击  以运行工程。
2. 现在，缓慢地将输入 PRIM 直流电压从 0V 增加到 400V。确保 `CLLLC_vPrimSensed_Volts` 显示正确的 VPRIM 值（即接近 400V）。此时，PWM 被触发；因此，不会从初级侧汲取电流。
3. 接下来，将 `VSEC` 增加到 300V。负载将从次级连接的电源汲取所有电流，该电流将接近 6.5A。
4. 现在，将 `CLLLC_iSecRef_Amps` 变量设置为 0.1A。
5. 通过向 `CLLLC_clearTrip` 变量写入 1 来清除跳闸。该实验中的软件会自动将 `CLLLC_closeGiLoop` 变量设置为 1。
6. 由于固定初级侧电压的电压范围较窄，因此转换器达到最高频率并饱和，并且 ISEC 处消耗的电流将高于 0.1A。用户可以通过监视 `CLLLC_pwmFrequency_Hz` 变量来观察该现象，该变量将在饱和上限期间接近 800kHz，在饱和和下限期间接近 200kHz。
7. 缓慢地将电流提高至 2 - 3A。现在，该电流将由次级连接的电压源和被测设计 (DUT) 共享。

5.2.2.5.5 在电池仿真模式下测量闭合电流环路的 SFRA

1. SFRA 集成在该构建的软件中，通过在硬件上进行测量来验证设计的补偿器是否可以提供足够的增益裕度和相位裕度。要运行 SFRA，请使工程保持运行状态，然后导航至 `<Install directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\SFRA_GUI.exe`。SFRA GUI 将弹出。
2. 在 SFRA GUI 上选择器件对应的各个选项；例如，对于 F280039，请选择浮点。点击“Setup Connection”。在弹出窗口中，取消选择“boot-on-connect”选项，选择合适的 COM 端口，然后点击 OK。返回到 SFRA GUI，然后点击 **Connect**。
3. SFRA GUI 将连接到器件。现在可以通过点击 **Start Sweep** 启动 SFRA 扫描。完整的 SFRA 扫描需要几分钟才能完成。可以通过查看 SFRA GUI 上的进度条以及检查控制卡背面用于指示 UART 活动的蓝色 LED 闪烁情况来监视活动。完成之后，会显示一个包含开环图的图，如图 5-29 所示。

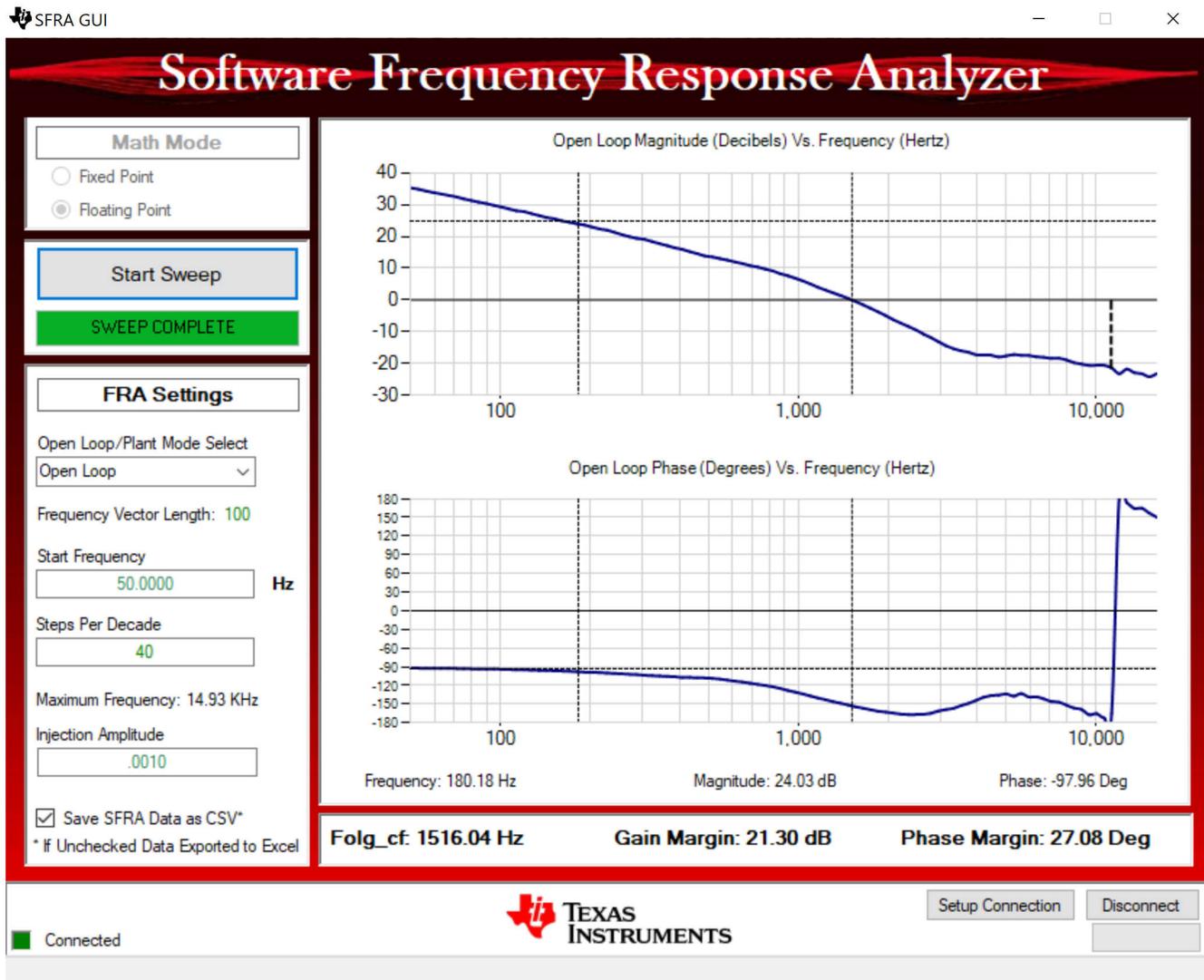


图 5-29. 仿真电池连接的闭合电流环路的 SFRA 开环图 (V_{prim} 400V , V_{sec} 300V , 功率 1.972kW , 实验 5)

频率响应数据也保存在工程文件夹的 SFRA 数据文件夹下，并根据 SFRA 运行的时间添加时间戳。

在不同的电流设定点测试 SFRA，确保周期未被钳制，以验证系统在可运行范围内是否稳定。

4. 这可以验证实验 5 电流环路设计。
5. 要使系统安全停止，请将输入 VPRIM 电压降至零。观察监视窗口中的电压和电流是否降至零。
6. 在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的“Halt”按钮  或使用“Target” → “Halt”来暂停处理器。然后点击 ，使 MCU 退出实时模式。最后，将 MCU 复位 ()。
7. 点击 *Terminate Debug Session*  (“Target” → “Terminate all”)，关闭 CCS 调试会话。

5.2.3 TTPLPFC 测试程序

5.2.3.1 实验 1：开环，直流

在该构建中，电路板以开环方式受到激励（采用固定占空比）。占空比通过 `dutyPU_DC` 变量进行控制。该构建验证来自功率级的反馈值检测以及 PWM 栅极驱动器的运行，并确保没有硬件问题。此外，可以在该生成中执行输入和输出电压感应校准。图 5-30 展示了该生成的软件结构。系统中有两个 ISR：用于电流环路的快速 ISR，以及用于运行电压环路和仪表功能的较慢 ISR。图 5-30 展示了每个 ISR 中运行的模块（请注意，TIDM-02013 是两相交错式 TTPLPFC）。

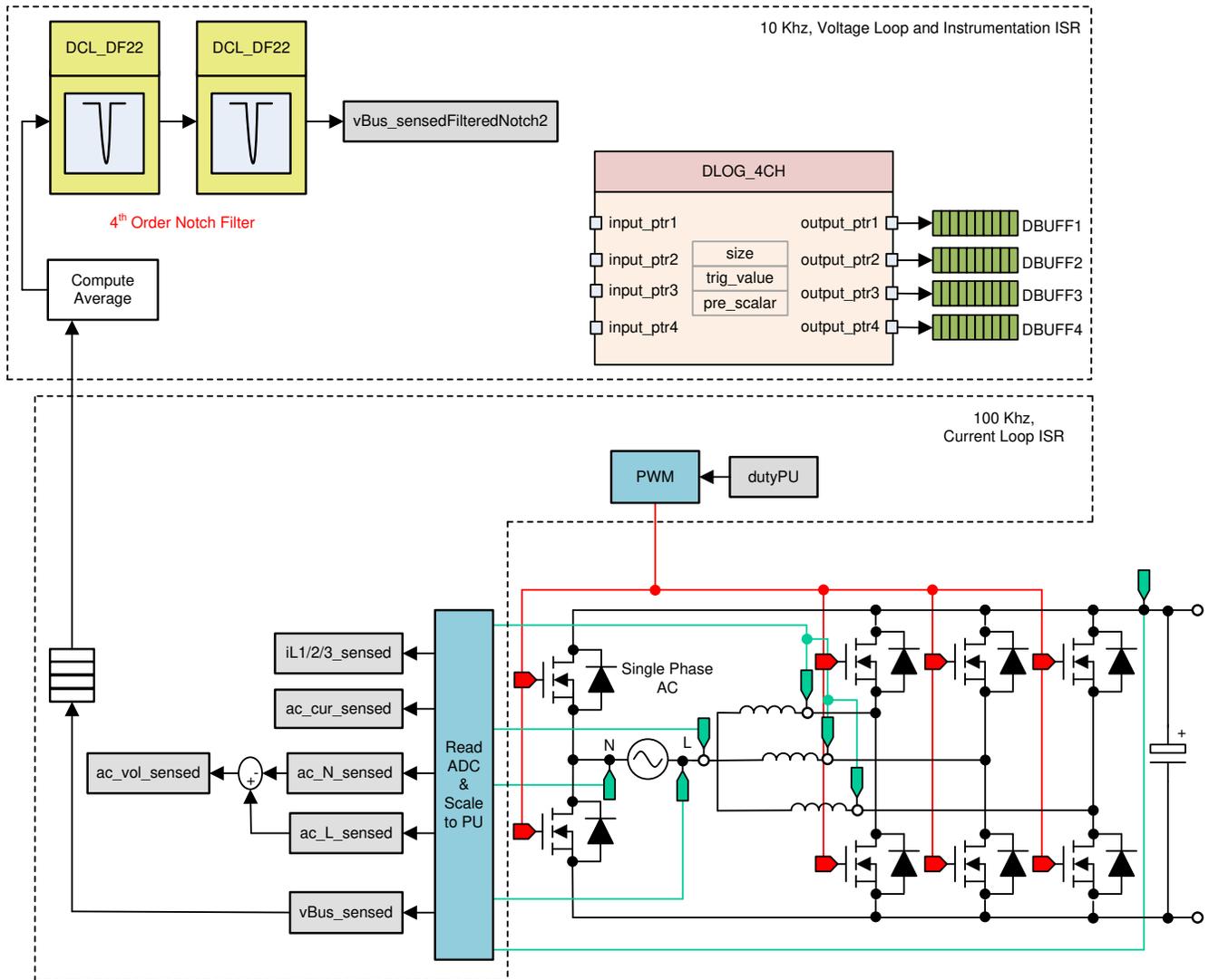


图 5-30. 构建级别 1 控制软件图：开环工程

5.2.3.1.1 设置 BUILD 1 的软件选项

打开 `TTPLPFC_settings.h` 并启用实验 1

```
#define TTPLPFC_LAB 1
```

5.2.3.1.2 构建和加载工程

右键点击工程名称，然后点击 *Rebuild Project*。

随后将成功构建工程。

在 *Project Explorer* 中，确保在 `targetConfigs` 项下将正确的目标配置文件设置为“Active”。

点击 **Run** → **Debug**。启动调试会话。在双 CPU 设备的情况下，可能会显示一个窗口来选择调试必须执行的 CPU。在本例中，选择“CPU1”。

然后项目加载到器件上，同时 CCS 调试视图变为有效状态。代码在主例程开始时停止。

5.2.3.1.3 设置调试环境窗口

要在监视和表达式窗口中添加变量，请点击 **View** → **Scripting Console** 以打开脚本控制台对话框。在该控制台的右上角，点击“open”。浏览到工程文件夹内的 `setupdebugenv_lab1.js` 脚本文件。该脚本文件会采用调试系统所需的相应变量填充监视窗口。点击监视窗口中的“Continuous Refresh”按钮，以持续更新控制器中的值。监视窗口如图 5-31 所示。

Expression	Type	Value	Address
buildInfo	enum enum_BuildLevel	BuildLevel1_OpenLoop_DC	0x0000A817@Data
guiVbus	float	-0.000226858858	0x0000A87E@Data
guiVin	float	-0.714660585	0x0000A876@Data
guiIi	float	0.0891165435	0x0000A874@Data
ac_cur_sensed	float	0.00343942642	0x0000A8A4@Data
clearTrip	int	0	0x0000A827@Data
EPwm1Regs.TZFLG	Register	0x0000	
EPwm2Regs.TZFLG	Register	0x0000	
dutyPU	float	0.5	0x0000A858@Data
dutyPU_DC	float	0.5	0x0000A85E@Data
vBus_sensed	float	0.0	0x0000A8C0@Data
iL1_sensed	float	-0.00341796875	0x0000A8C4@Data
iL2_sensed	float	-0.00830078125	0x0000A8C6@Data
iL3_sensed	float	-0.00732421875	0x0000A8AA@Data
+ Add new expression			

图 5-31. 版本级别 1 表达式视图

点击  来运行工程。

使用工具栏上的 **Halt** 按钮 () 来暂停处理器。

5.2.3.1.4 使用实时仿真

实时仿真是一项特殊仿真功能，允许在 MCU 运行时更新 CCS 内的窗口。该功能不但可实现图形和监视视图更新，而且允许改变监视或存储器窗口中的值，并且无需停止处理器即可查看这些更改对系统的影响。

通过将鼠标悬停在水平工具栏中的按钮上并点击  按钮来启用实时模式。

Enable Silicon Real-time Mode (service critical interrupts when halted, allow debugger accesses while running)

可能会显示一个消息框。若是如此，选择 **YES** 来启用调试事件。此操作会将状态寄存器 1 (ST1) 的位 1 (DGBM 位) 设置为 0。DGBM 是调试使能屏蔽位。当 DGBM 位设置为 0 时，内存和寄存器值将传递到主机处理器以更新调试器窗口。

5.2.3.1.5 运行代码

点击  再次来运行工程。

几秒钟后，浪涌继电器发出咔嚓声。软件在具有直流的构建级别中被编程为执行此操作。跳闸清除，并施加 0.5 的占空比。

在监视视图中，定期查看是否更新了 `guiVin`、`guiVbus`、`guiIi` 变量。由于未通电，因此该值接近于零。

缓慢增大输入直流电压 (从 0 到 120V)。输出电压显示升高的电压，因为作为默认设置施加了 0.5 PU 的稳定占空比。如果消耗了大电流，请验证电压端子是否交换。如果属实，请先将电压降至零并纠正问题，再恢复测试。

通过确保 *TTPLPFC_vBusAvg_pu* 显示正确的值来验证电压检测。对于 120V 直流输入，这在某种程度上验证了电路板的电压感测。

可以更改 *dutyPU_DC* 变量以查看在各种升压条件下的运行情况。这在基本层面上验证 PWM 驱动器和硬件连接完成后，将输入电压降至零，并观察总线电压是否降至零。

对该构建的检查到此结束。成功完成该构建后，会验证以下项目：

- 电压和电流检测以及缩放是否精确
- 电流环路 ISR 和电压环路仪表 ISR 中构建 1 代码的中断生成和执行
- PWM 驱动器和开关

如果观察到任何问题，则可能需要仔细检查硬件，以消除所有生成问题，等等。

现在可以停止控制器，并终止调试连接。

在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的 *Halt* 按钮 () 或使用 *Target* → *Halt* 来暂停处理器。然后单击 ，使 MCU 退出实时模式。最后，单击  以对 MCU 进行复位。

单击 *Terminate Debug Session* (终止调试会话) (*Target* (目标) → *Terminate all* (终止全部)) 关闭 CCS 调试会话。



5.2.3.2 实验 2：闭合电流环路，直流

在构建 2 中，内侧电流环路是闭合的，也就是说，使用电流补偿器 *Gi* 控制电感器电流。对该电流补偿器的输出应用了直流总线和输出电压前馈，以生成逆变器的占空比。该操作可以简化电流补偿器的受控体，并且可以使用比例 (P) 控制器来调节内侧电流的环路。我们得出了电流环路的模型。图 5-32 说明了该构建的完整软件图 (请注意，TIDM-02013 是两相交错式 TTPLPFC)。

$$duty1PU = \frac{(ac_cur_meas - ac_cur_ref_inst) \times Gi + ac_vol_sensed}{vBus_sensed} \quad (21)$$

图 5-32 说明了该构建的完整软件图。

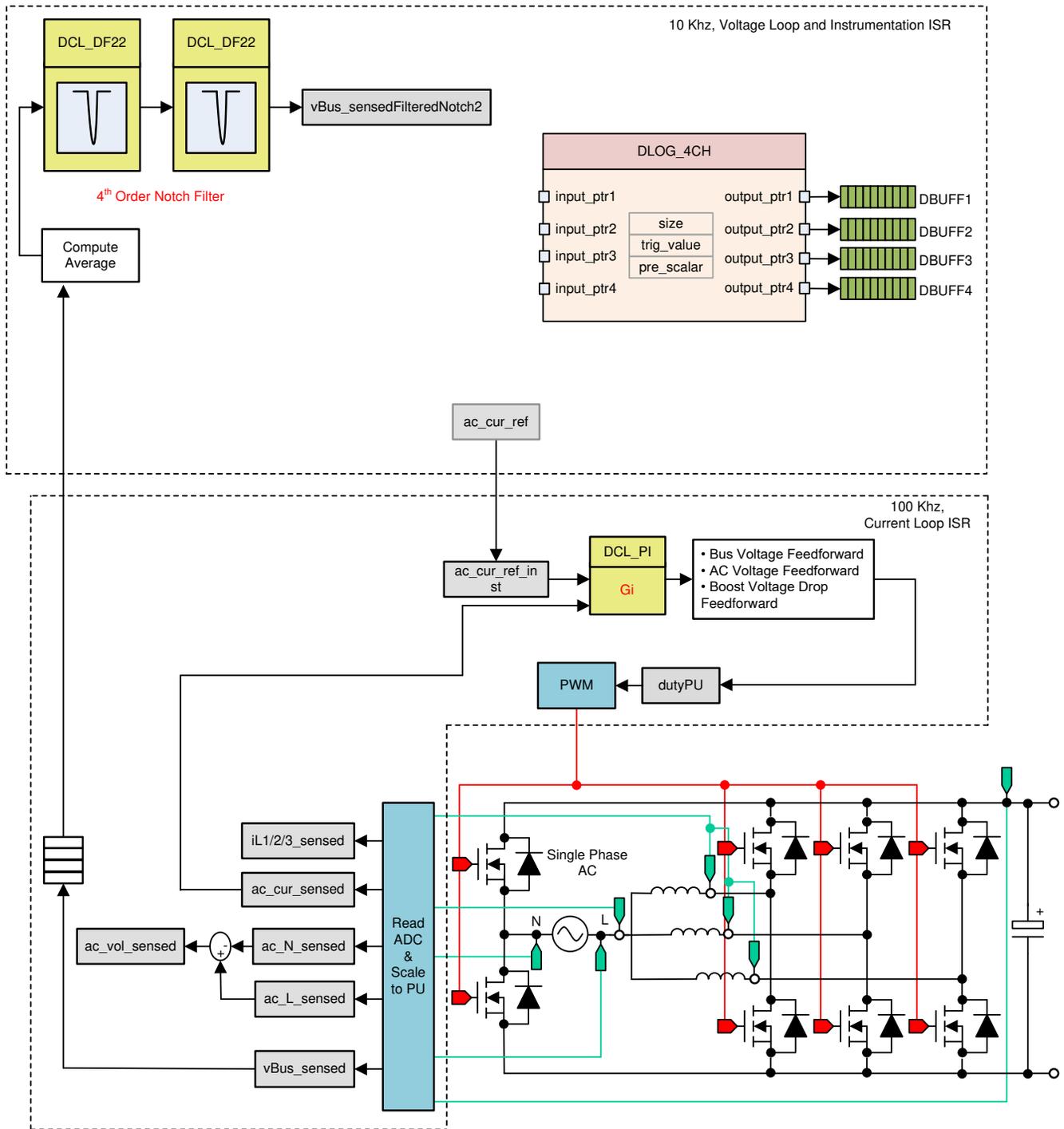


图 5-32. 构建级别 2 控制软件图：闭合电流环路

5.2.3.2.1 设置 BUILD 2 的软件选项

确保按照节 5.1.1 中所述为独立 PFC 运行设置硬件。先不要向电路板提供任何高压 (HV) 电源。

打开 TTPLPFC_settings.h 并启用实验 2

```
#define TTPLPFC_LAB 2
```

确保所有其他选项与前面节 5.2.3.2 中指定的选项相同。

1. 打开补偿设计器 <install
Directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\CompDesigner.exe

5.2.3.2.2 设计电流环路补偿器

补偿设计器启动。可以零极点角度调整基于 PI 的控制器，以确保稳定的闭环运行。可以通过观察补偿设计器中开环传递函数图上的增益裕量和相位裕度来验证使用所设计的补偿器时的系统稳定性，如图 5-33 所示。

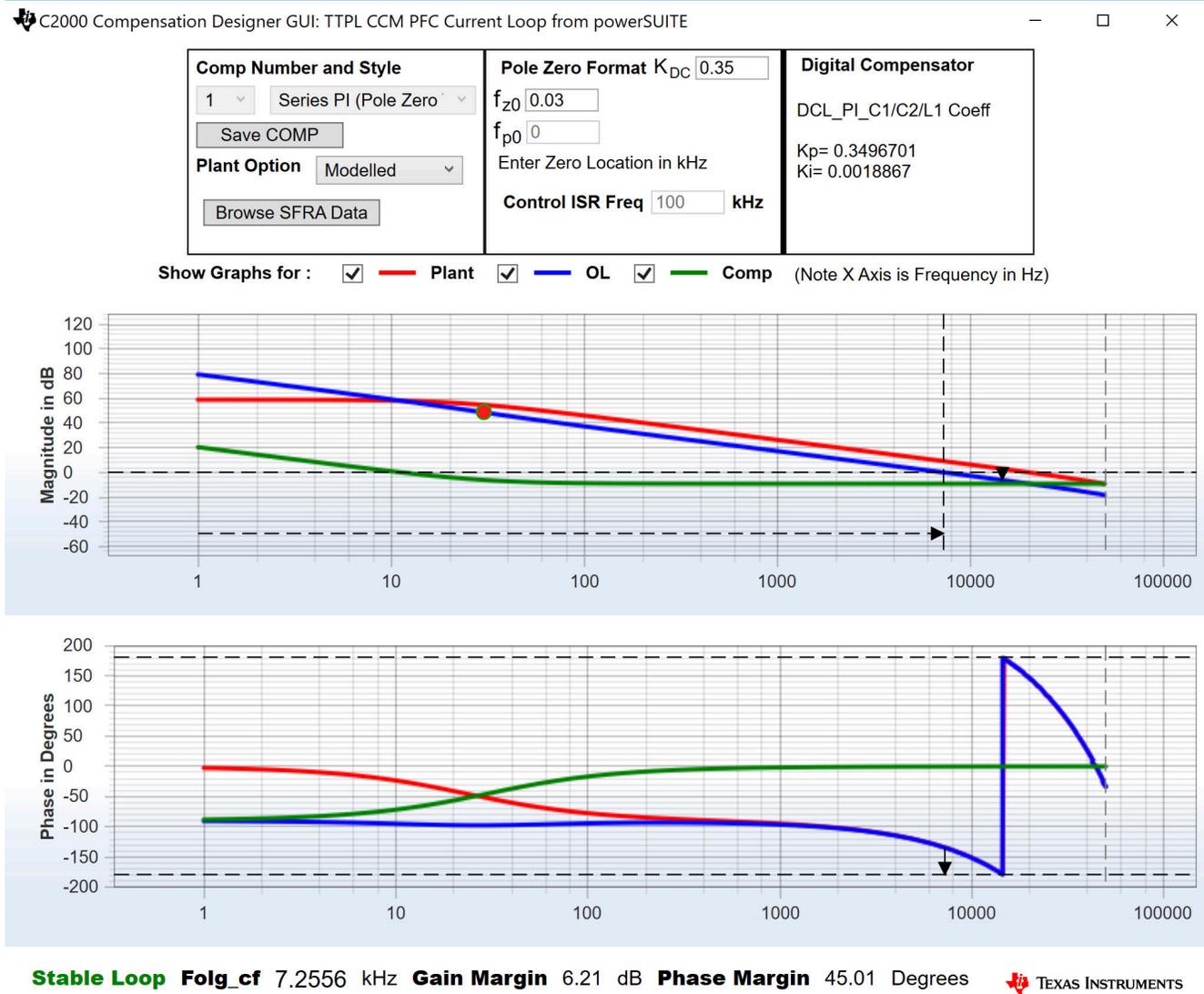


图 5-33. 使用补偿设计器的电流环路设计

对开环增益感到满意后，可以在 `ttplpfc_settings.h` 文件中更新补偿器值。

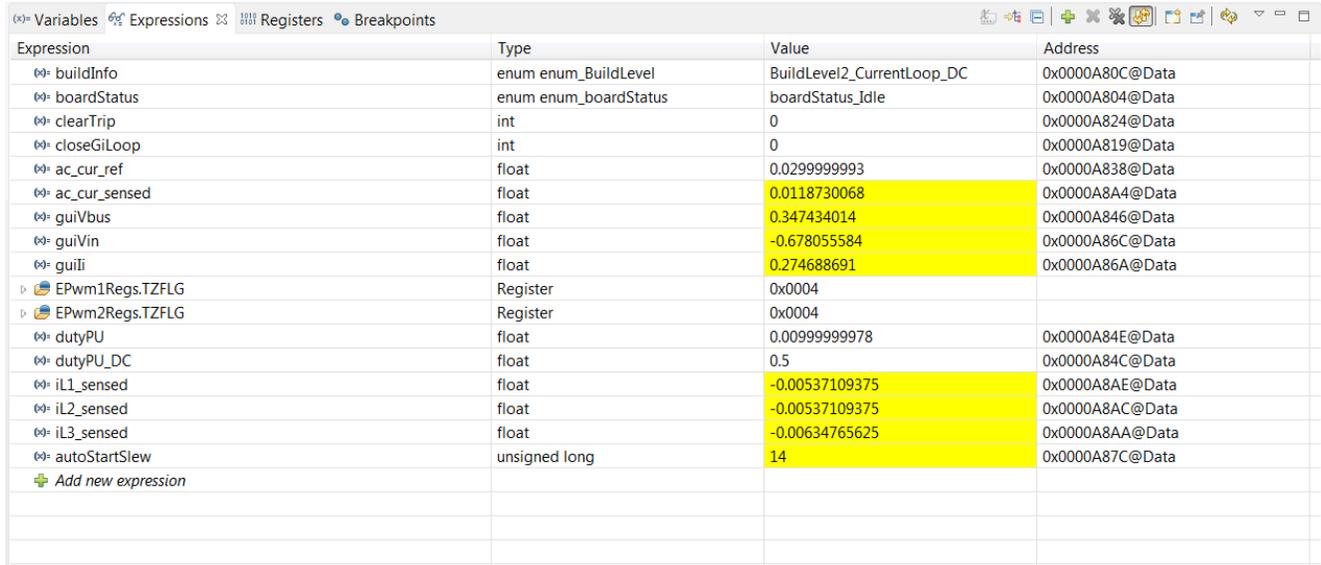
关闭补偿设计器

5.2.3.2.3 构建和加载工程以及设置调试

右键单击工程名称，然后单击 **Rebuild Project**。随后将成功构建工程。单击 **Run** (运行) → **Debug** (调试)，这将启动调试会话。对于双 CPU 器件，可能会显示一个窗口，可以在其中选择必须执行调试的 CPU。在本例中，选择“CPU1”。然后项目加载到器件上，同时 CCS 调试视图变为有效状态。代码在主例程开始时停止。

要在监视和表达式窗口中添加变量，请点击 **View** → **Scripting Console** 以打开脚本控制台对话框。在该控制台的右上角，点击 **Open** 浏览到位于工程文件夹内的 `setupdebugenv_lab2.js` 脚本文件。该文件使用调试系统所需的合

适变量来填充监视窗口。点击监视窗口中的 **Continuous Refresh** 按钮 ()，以持续更新控制器中的值。监视窗口将如图 5-34 所示。



Expression	Type	Value	Address
buildInfo	enum enum_BuildLevel	BuildLevel2_CurrentLoop_DC	0x0000A80C@Data
boardStatus	enum enum_boardStatus	boardStatus_Idle	0x0000A804@Data
clearTrip	int	0	0x0000A824@Data
closeGilLoop	int	0	0x0000A819@Data
ac_cur_ref	float	0.0299999993	0x0000A838@Data
ac_cur_sensed	float	0.0118730068	0x0000A8A4@Data
guiVbus	float	0.347434014	0x0000A824@Data
guiVin	float	-0.678055584	0x0000A86C@Data
guiIi	float	0.274688691	0x0000A86A@Data
EPwm1Regs.TZFLG	Register	0x0004	
EPwm2Regs.TZFLG	Register	0x0004	
dutyPU	float	0.00999999978	0x0000A84E@Data
dutyPU_DC	float	0.5	0x0000A84C@Data
iL1_sensed	float	-0.00537109375	0x0000A8AE@Data
iL2_sensed	float	-0.00537109375	0x0000A8AC@Data
iL3_sensed	float	-0.00634765625	0x0000A8AA@Data
autoStartSlew	unsigned long	14	0x0000A87C@Data
Add new expression			

图 5-34. 构建级别 2：闭合电流环路表达式视图

通过将鼠标悬停在水平工具栏中的按钮上并点击  按钮来启用实时模式。

点击  来运行工程。

使用工具栏上的 **Halt** 按钮  来暂停处理器。

5.2.3.2.4 运行代码

编程该项目以驱动浪涌继电器，并在设定的时间后清除跳闸，即 `autoStartSlew==100`。软件在具有直流的构建级别中被编程为执行此操作。必须在运行后和此自动压摆计数器达到 100 之前施加输入电压。如果计数器达到 100，在输入端施加电压之前，必须将代码复位。为此，控制器必须退出实时模式，进行重置并重新启动。通过将鼠标悬停在水平工具栏上并点击  按钮，重复执行节 5.2.3.2.3 中启用实时模式的步骤。

点击  以运行工程。

在 `autoStartSlew` 达到 100 之前施加约 50V 的输入电压。一旦 `autoStartSlew` 达到 100，浪涌继电器即触发，PWM 跳闸被清除，同时关闭电流环路标志。

Expression	Type	Value	Address
buildInfo	enum enum_BuildLevel	BuildLevel2_CurrentLoop_DC	0x0000A80C@Data
boardStatus	enum enum_boardStatus	boardStatus_Idle	0x0000A804@Data
clearTrip	int	1	0x0000A824@Data
closeGiLoop	int	1	0x0000A819@Data
ac_cur_ref	float	0.0299999993	0x0000A838@Data
ac_cur_sensed	float	0.0300658941	0x0000A8A4@Data
guiVbus	float	127.377548	0x0000A846@Data
guiVin	float	48.3203316	0x0000A86C@Data
guiIi	float	0.707000256	0x0000A86A@Data
EPwm1Regs.TZFLG	Register	0x0000	
EPwm2Regs.TZFLG	Register	0x0000	
dutyPU	float	0.386497527	0x0000A84E@Data
dutyPU_DC	float	0.5	0x0000A84C@Data
iL1_sensed	float	0.0107421875	0x0000A8AE@Data
iL2_sensed	float	0.0087890625	0x0000A8AC@Data
iL3_sensed	float	0.009765625	0x0000A8AA@Data
autoStartSlew	unsigned long	101	0x0000A87C@Data

图 5-35. 监视表达式，构建实验 2，闭合电流环路后的直流操作开始

输入电流调节在 1.5A 左右，输出电压升至约 193V。

将 ac_cur_ref 缓慢增大至 0.045，即 2.4A 输入。

缓慢增大 $V_{in} = 120V$ ，输出电压将大于 370V。

Expression	Type	Value	Address
buildInfo	enum enum_BuildLevel	BuildLevel2_CurrentLoop_DC	0x0000A80C@Data
boardStatus	enum enum_boardStatus	boardStatus_Idle	0x0000A804@Data
clearTrip	int	1	0x0000A824@Data
closeGiLoop	int	1	0x0000A819@Data
ac_cur_ref	float	0.100000001	0x0000A838@Data
ac_cur_sensed	float	0.0993705988	0x0000A8A4@Data
guiVbus	float	380.123596	0x0000A846@Data
guiVin	float	117.478661	0x0000A86C@Data
guiIi	float	2.46380639	0x0000A86A@Data
EPwm1Regs.TZFLG	Register	0x0000	
EPwm2Regs.TZFLG	Register	0x0000	
dutyPU	float	0.308701962	0x0000A84E@Data
dutyPU_DC	float	0.5	0x0000A84C@Data
iL1_sensed	float	0.0493164063	0x0000A8AE@Data
iL2_sensed	float	0.052734375	0x0000A8AC@Data
iL3_sensed	float	0.0458984375	0x0000A8AA@Data
autoStartSlew	unsigned long	101	0x0000A87C@Data

图 5-36. 监视表达式，构建实验 2，闭合电流环路后的直流操作以全电压开始

SFRA 集成在该构建的软件中，通过在硬件上进行测量来验证设计的补偿器是否可以提供足够的增益裕度和相位裕度。要运行 SFRA，请使工程保持运行状态，然后导航至 `<Install directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\SFRA_GUI.exe`。然后出现 SFRA GUI。

在 SFRA GUI 上选择器件的相应选项。例如，对于 F28003x，选择“Floating Point”。点击 **Setup Connection**。在弹出窗口中取消选中“boot on connect”选项，然后选择合适的 COM 端口。确保已取消选择 **Boot on Connect**（连接时启动）。点击 **OK**。返回到 SFRA GUI，然后点击 **Connect**。

SFRA GUI 将连接到器件。现在可以通过点击 **Start Sweep** 启动 SFRA 扫描。完整的 SFRA 扫描需要几分钟才能完成。可以通过查看 SFRA GUI 上的进度条以及检查控制卡背面用于指示 UART 活动的蓝色 LED 的闪烁情况来监视活动。完成之后，将立即显示一个包含开环图的图。频率响应数据也保存在工程文件夹的 SFRA 数据文件夹下，并根据 SFRA 运行的时间添加时间戳。

此外，测量的受控体频率响应可用于通过补偿设计器设计电流补偿器。<install
Directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\CompDesigner.exe。

针对 GUI 上的受控体选项选择 **SFRA Data**。这项操作使用测量的受控体信息来设计补偿器。该选项可用于微调补偿。默认情况下，补偿设计器指向最近运行的 SFRA。如果必须使用运行的前一个 SFRA 受控体信息，则选择 SFRADData.csv 文件（通过点击 **Browse SFRA Data** 浏览到该文件）。该操作验证了电流补偿器设计。

通过将输入直流电压降至零使系统安全停止。确保 guiVbus 也降至零。

在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的 **Halt** 按钮 () 或使用 **Target → Halt** 来暂停处理器。然后点击 ，使 MCU 退出实时模式。最后，将 MCU 复位 ()。

点击 **Terminate Debug Session (Target → Terminate all)**，关闭 CCS 调试会话。



5.2.3.3 实验 3：闭合电流环路，交流

在实验 3 中，内侧电流环路是闭合的，也就是说，使用电流补偿器 G_i 控制电感器电流。对该电流补偿器的输出应用了直流总线和输出电压前馈，以生成逆变器的占空比，并在过零附近对 PWM 进行软启动。

图 5-37 说明了该构建的完整软件图（请注意，TIDM-02013 是两相交错式 TTPLPFC）。

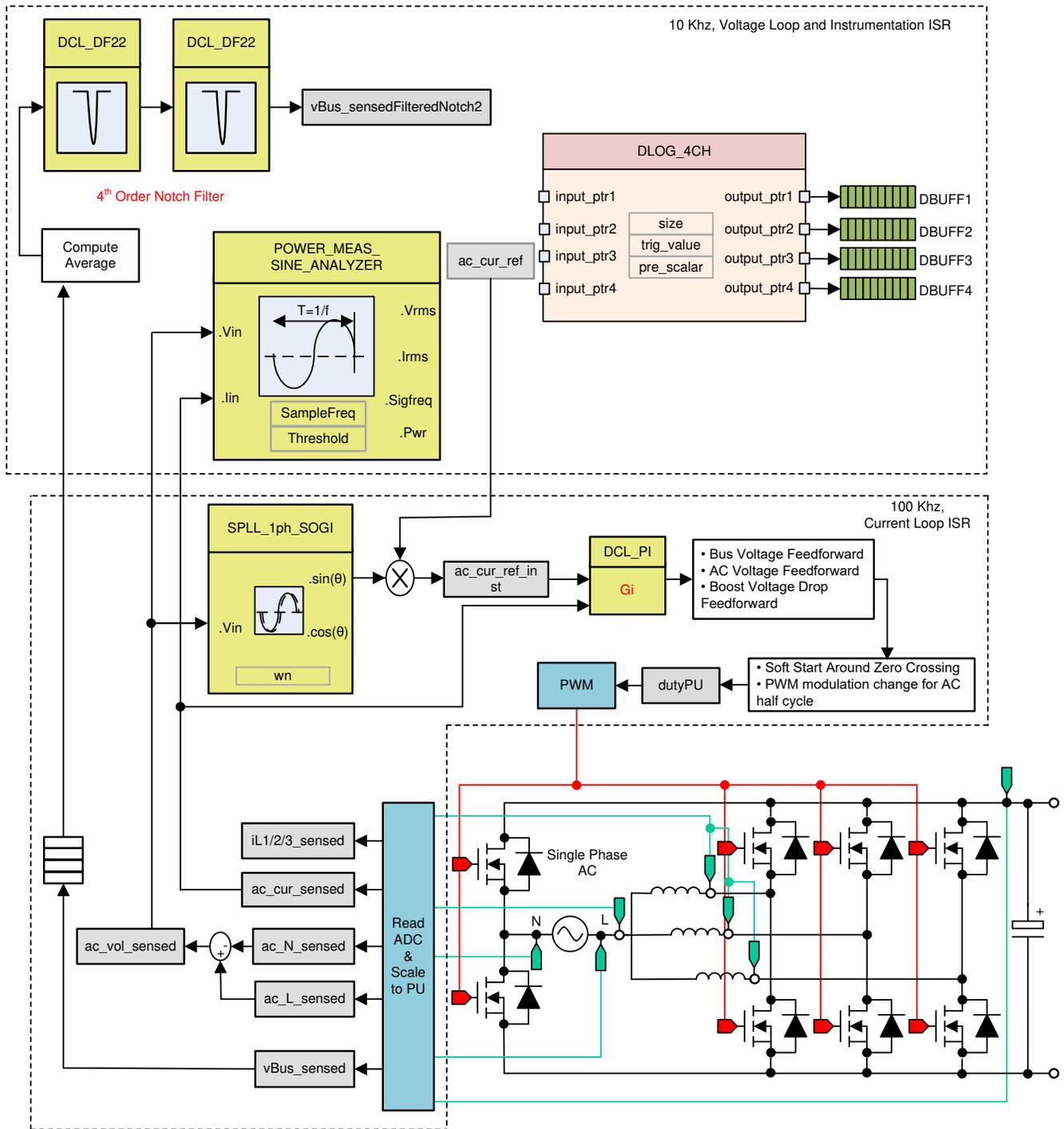


图 5-37. 构建实验 3 控制软件图：闭合电流环路，交流

5.2.3.3.1 设置实验 3 的软件选项

打开 TTPLPFC_settings.h 并启用实验 3

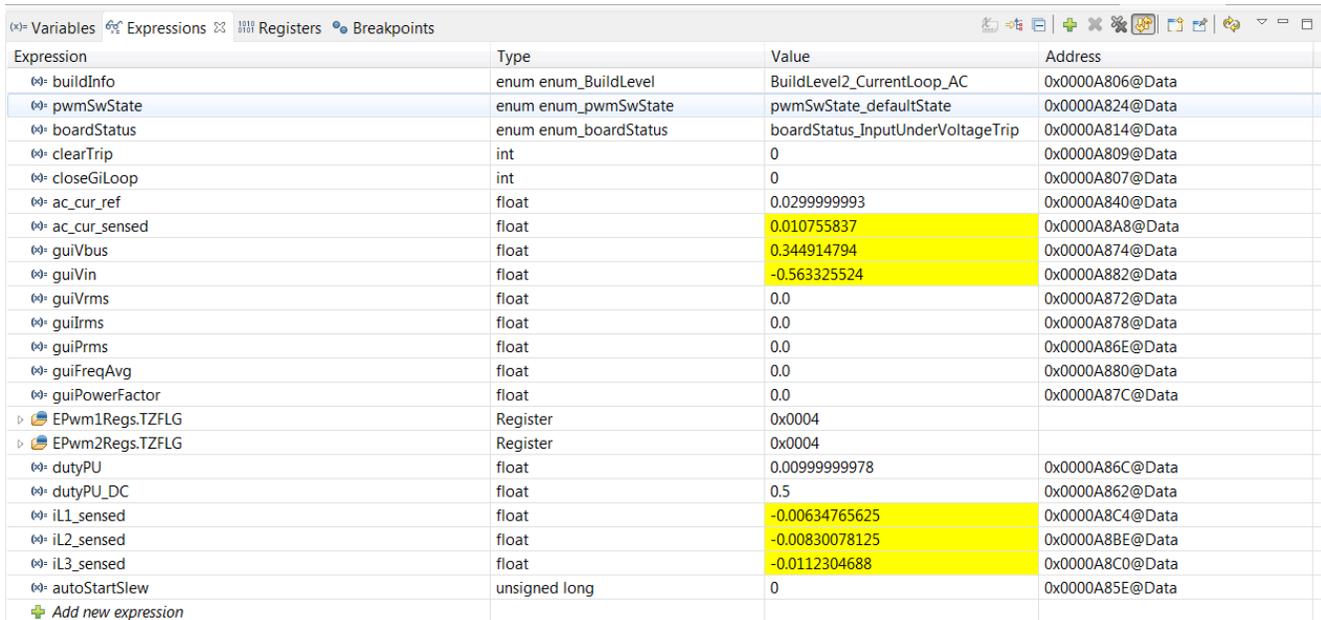
```
#define TTPLPFC_LAB 3
```

该构建中重新使用了先前构建中的电流补偿器，因此不需要额外的步骤来调整该构建级别中的电流环路。

5.2.3.3.2 构建和加载工程以及设置调试

右键点击工程名称，然后点击 **Rebuild Project**。随后将成功构建工程。点击 **Run (运行)** → **Debug (调试)**，这将启动调试会话。对于双 CPU 器件，可能会显示一个窗口，可以在其中选择必须执行调试的 CPU。在本例中，选择“CPU1”。然后项目加载到器件上，同时 CCS 调试视图变为有效状态。代码在主例程开始时停止。

要在监视和表达式窗口中添加变量，请点击 **View → Scripting Console** 以打开脚本控制台对话框。在该控制台的右上角，点击 **Open** 浏览到位于工程文件夹内的 `setupdebugenv_Lab3.js` 脚本文件。该文件使用调试系统所需的合适变量来填充监视窗口。点击监视窗口中的 **Continuous Refresh** 按钮 ()，以持续更新控制器中的值。监视窗口将如图 5-38 所示。



Expression	Type	Value	Address
buildInfo	enum enum_BuildLevel	BuildLevel2_CurrentLoop_AC	0x0000A806@Data
pwmSwState	enum enum_pwmSwState	pwmSwState_defaultState	0x0000A824@Data
boardStatus	enum enum_boardStatus	boardStatus_InputUnderVoltageTrip	0x0000A814@Data
clearTrip	int	0	0x0000A809@Data
closeGiLoop	int	0	0x0000A807@Data
ac_cur_ref	float	0.02999999993	0x0000A840@Data
ac_cur_sensed	float	0.010755837	0x0000A8A8@Data
guiVbus	float	0.344914794	0x0000A874@Data
guiVin	float	-0.563325524	0x0000A882@Data
guiVrms	float	0.0	0x0000A872@Data
guiirms	float	0.0	0x0000A878@Data
guiPrms	float	0.0	0x0000A86E@Data
guiFreqAvg	float	0.0	0x0000A880@Data
guiPowerFactor	float	0.0	0x0000A87C@Data
EPwm1Regs.TZFLG	Register	0x0004	
EPwm2Regs.TZFLG	Register	0x0004	
dutyPU	float	0.009999999978	0x0000A86C@Data
dutyPU_DC	float	0.5	0x0000A862@Data
iL1_sensed	float	-0.00634765625	0x0000A8C4@Data
iL2_sensed	float	-0.00830078125	0x0000A8BE@Data
iL3_sensed	float	-0.0112304688	0x0000A8C0@Data
autoStartSlew	unsigned long	0	0x0000A85E@Data

图 5-38. 实验 3 交流：闭合电流环路表达式视图

通过将鼠标悬停在水平工具栏中的按钮上并点击  按钮来启用实时模式。

5.2.3.3.3 运行代码

该工程被编程为等待输入电压超过大约 $70V_{rms}$ 以驱动浪涌继电器，并清除跳闸。

点击  以运行工程。

施加大约 120V 的输入电压，电路板就会脱离欠压状态，并且驱动浪涌继电器。跳闸清除，并消耗大约 1.3A RMS 的少量电流。监视窗口与图 5-39 类似。总线电压接近 270V。

Expression	Type	Value	Address
buildInfo	enum enum_BuildLevel	BuildLevel2_CurrentLoop_AC	0x0000A806@Data
pwmSwState	enum enum_pwmSwState	pwmSwState_positiveHalf	0x0000A824@Data
boardStatus	enum enum_boardStatus	boardStatus_NoFault	0x0000A814@Data
clearTrip	int	1	0x0000A809@Data
closeGiLoop	int	1	0x0000A807@Data
ac_cur_ref	float	0.0299999993	0x0000A840@Data
ac_cur_sensed	float	-0.00663924217	0x0000A8A8@Data
guiVbus	float	180.061981	0x0000A874@Data
guiVin	float	-49.6501122	0x0000A882@Data
guiVrms	float	117.459831	0x0000A872@Data
guiIrms	float	0.551513135	0x0000A878@Data
guiPrms	float	64.2371902	0x0000A86E@Data
guiFreqAvg	float	59.8999023	0x0000A880@Data
guiPowerFactor	float	0.978407621	0x0000A87C@Data
EPwm1Regs.TZFLG	Register	0x0000	
EPwm2Regs.TZFLG	Register	0x0000	
dutyPU	float	-0.880984187	0x0000A86C@Data
dutyPU_DC	float	0.5	0x0000A862@Data
iL1_sensed	float	0.0180664063	0x0000A8C4@Data
iL2_sensed	float	-0.0048828125	0x0000A8BE@Data
iL3_sensed	float	-0.0283203125	0x0000A8C0@Data
autoStartSlew	unsigned long	5	0x0000A85E@Data

图 5-39. 监视表达式，实验 2，闭合电流环路后的交流操作开始

将 ac_cur_ref 缓慢增大至 0.078，即 2.4A 输入，总线电压升至 400V。电压和电流波形如图 5-40 所示。

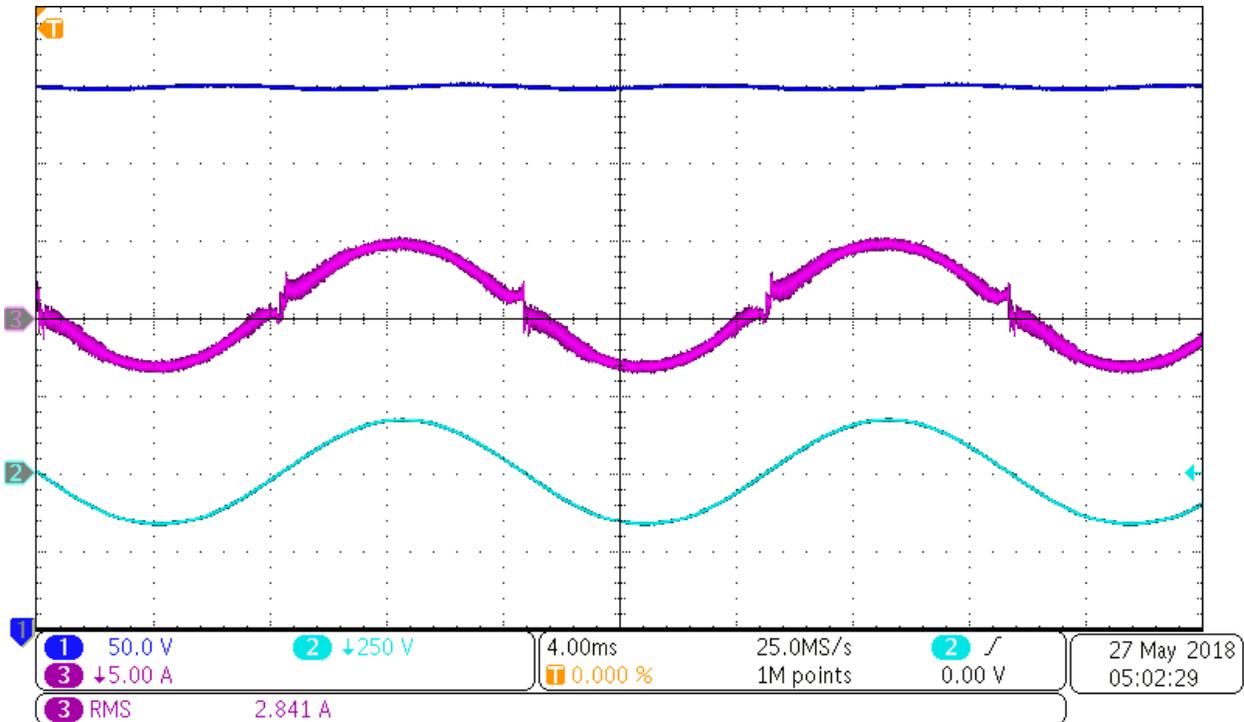


图 5-40. 输入交流电流和输出直流电压波形

本构建将 SFRA 集成在软件中，通过在硬件上进行测量来验证设计的补偿器是否能够提供足够的增益裕量和相位裕度。要运行 SFRA，请使工程保持运行状态，然后导航至 `<Install directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\SFRA_GUI.exe`。然后出现 SFRA GUI。

在 SFRA GUI 上选择器件的相应选项。例如，对于 F280039，选择“Floating Point”。点击 *Setup Connection*。在弹出窗口中，取消选择“boot-on-connect”选项，然后选择合适的 COM 端口。点击 *OK*。返回到 SFRA GUI，然后点击 *Connect*。

SFRA GUI 将连接到器件。现在可以通过点击 *Start Sweep* 启动 SFRA 扫描。完整的 SFRA 扫描需要几分钟才能完成。可以通过查看 SFRA GUI 上的进度条以及检查控制卡背面用于指示 UART 活动的蓝色 LED 闪烁情况来监视活动。完成之后，将立即显示一个包含开环图的图。这类似于直流条件下看到的图；但交流谐波频率接近测量频率，因此可以看到一些额外的噪声。BW、PM 和 GM 数字与直流情况非常相似。

要使系统安全停止，请关闭交流电源的输出，从而使输入交流电压降至零。确保 *guiVbus* 也降至零。

在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的 *Halt* 按钮 () 或使用 *Target → Halt* 来暂停处理器。然后点击 ，使 MCU 退出实时模式。最后，将 MCU 复位 ()。

点击 *Terminate Debug Session (Target → Terminate all)*，关闭 CCS 调试会话。



5.2.3.4 实验 4：闭合电压和电流环路

在该构建中，在内侧电流环路闭合的情况下，外侧电压环路是闭合的。我们在图 5-41 中得出了外侧电压环路的模型（请注意，TIDM-02013 是两相交错式 TTPLPFC）。对于外侧电压环路，使用基于 PI 的补偿器，并通过补偿设计器对其进行调谐。

图 5-41 展示了该生成的软件图。

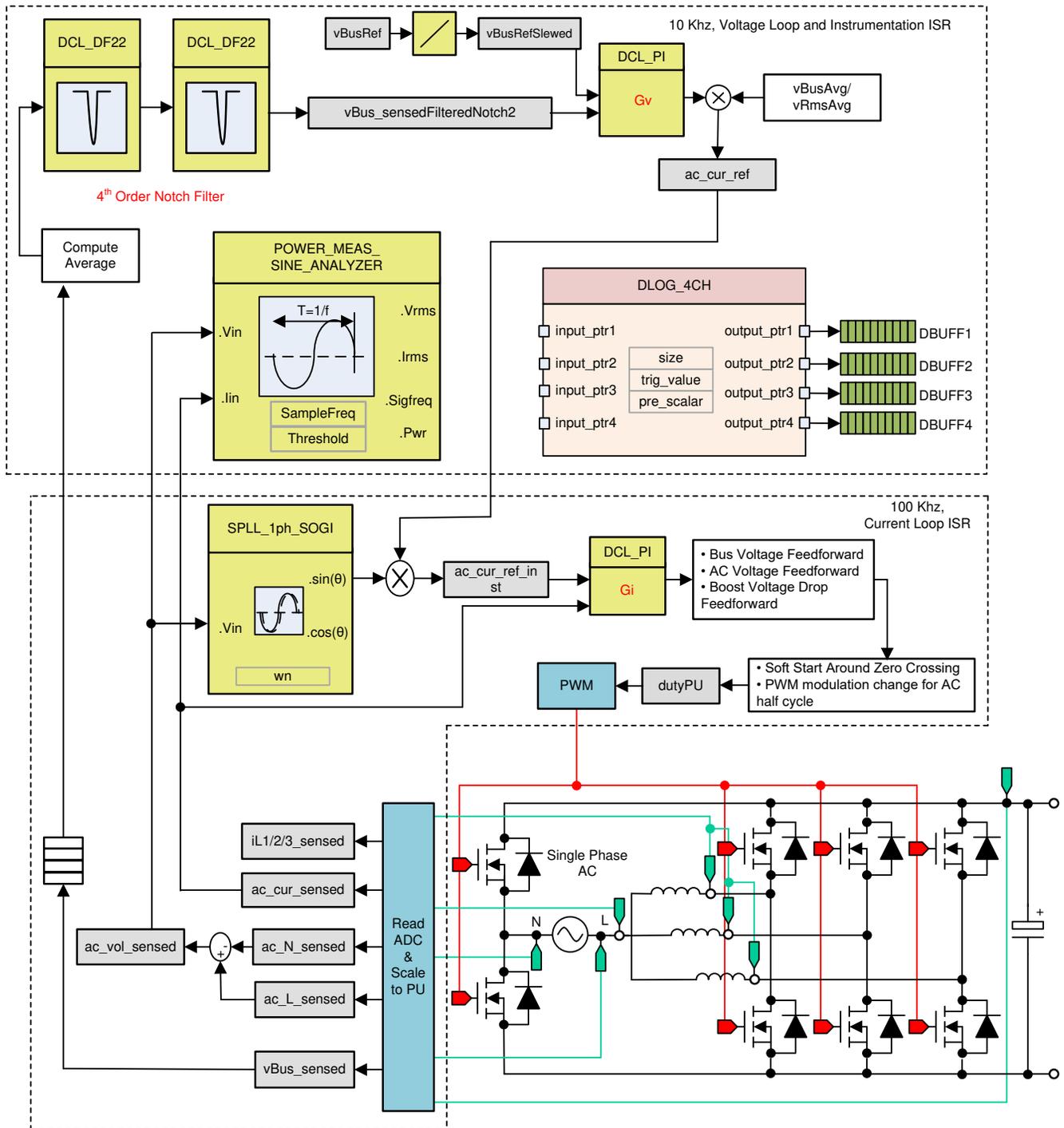


图 5-41. 构建级别 4 控制图：使用内侧电流环路进行输出电压控制

5.2.3.4.1 设置 BUILD 4 的软件选项

确保按照节 5.1.1 中所述为独立 PFC 运行设置硬件。先不要向电路板提供任何高压 (HV) 电源。

打开 TTPLPFC_settings.h 并启用实验 4

```
#define TTPLPFC_LAB 4
```

确保所有其他选项与前面图 5-41 中指定的选项相同。

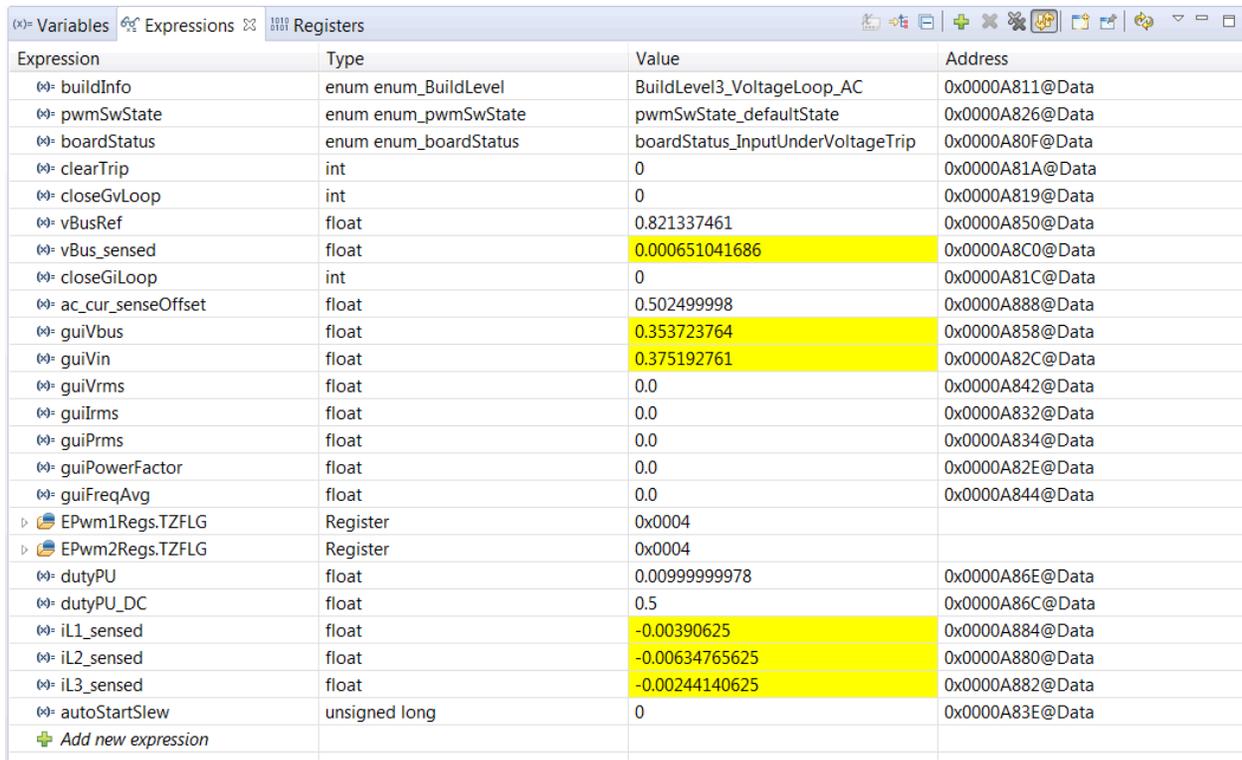
1. 打开补偿设计器 <install

Directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\CompDesigner.exe

5.2.3.4.2 构建和加载工程以及设置调试

右键单击工程名称，然后单击 **Rebuild Project**。随后将成功构建工程。单击 **Run (运行)** → **Debug (调试)**，这将启动调试会话。对于双 CPU 器件，可能会显示一个窗口，可以在其中选择必须执行调试的 CPU。在本例中，选择“CPU1”。然后项目加载到器件上，同时 CCS 调试视图变为有效状态。代码在主例程开始时停止。

要在监视和表达式窗口中添加变量，请点击 **View → Scripting Console** 以打开脚本控制台对话框。在该控制台的右上角，点击 **Open** 浏览到位于项目文件夹内的 **setupdebugenv_lab4.js** 脚本文件。该文件使用调试系统所需的合适变量来填充监视窗口。点击监视窗口中的 **Continuous Refresh** 按钮 ()，以持续更新控制器中的值。监视窗口如图 5-42 所示。



Expression	Type	Value	Address
buildInfo	enum enum_BuildLevel	BuildLevel3_VoltageLoop_AC	0x0000A811@Data
pwmSwState	enum enum_pwmSwState	pwmSwState_defaultState	0x0000A826@Data
boardStatus	enum enum_boardStatus	boardStatus_InputUnderVoltageTrip	0x0000A80F@Data
clearTrip	int	0	0x0000A81A@Data
closeGvLoop	int	0	0x0000A819@Data
vBusRef	float	0.821337461	0x0000A850@Data
vBus_sensed	float	0.000651041686	0x0000A8C0@Data
closeGilLoop	int	0	0x0000A81C@Data
ac_cur_senseOffset	float	0.502499998	0x0000A888@Data
guiVbus	float	0.353723764	0x0000A858@Data
guiVin	float	0.375192761	0x0000A82C@Data
guiVrms	float	0.0	0x0000A842@Data
guiIrms	float	0.0	0x0000A832@Data
guiPrms	float	0.0	0x0000A834@Data
guiPowerFactor	float	0.0	0x0000A82E@Data
guiFreqAvg	float	0.0	0x0000A844@Data
EPwm1Regs.TZFLG	Register	0x0004	
EPwm2Regs.TZFLG	Register	0x0004	
dutyPU	float	0.00999999978	0x0000A86E@Data
dutyPU_DC	float	0.5	0x0000A86C@Data
il1_sensed	float	-0.00390625	0x0000A884@Data
il2_sensed	float	-0.00634765625	0x0000A880@Data
il3_sensed	float	-0.00244140625	0x0000A882@Data
autoStartSlew	unsigned long	0	0x0000A83E@Data
+ Add new expression			

图 5-42. 构建实验 4：表达视图

通过将鼠标悬停在水平工具栏中的按钮上并单击  按钮来启用实时模式。

单击  来运行工程。

使用工具栏上的 **Halt** 按钮 () 来暂停处理器。

5.2.3.4.3 运行代码

该工程被编程为等待输入电压超过大约 70V_{rms} 以驱动浪涌继电器，并清除跳闸。

单击  以运行工程。

施加大约 120V 的输入电压，电路板就会脱离欠压状态，并且驱动浪涌继电器。跳闸清除，输出端升至 380V 直流电压。从交流输入端获取正弦电流。图 5-43 展示了程序在该阶段运行时的监视窗口。

Expression	Type	Value	Address
buildInfo	enum enum_BuildLevel	BuildLevel3_VoltageLoop_AC	0x0000A811@Data
pwmSwState	enum enum_pwmSwState	pwmSwState_negativeHalf	0x0000A826@Data
boardStatus	enum enum_boardStatus	boardStatus_NoFault	0x0000A80F@Data
clearTrip	int	1	0x0000A81A@Data
closeGvLoop	int	1	0x0000A819@Data
vBusRef	float	0.821337461	0x0000A850@Data
vBus_sensed	float	0.822998047	0x0000A8C0@Data
closeGiLoop	int	1	0x0000A81C@Data
ac_cur_senseOffset	float	0.502499998	0x0000A888@Data
guiVbus	float	380.081421	0x0000A858@Data
guiVin	float	-152.073486	0x0000A82C@Data
guiVrms	float	120.093376	0x0000A842@Data
guiIrms	float	2.40836215	0x0000A832@Data
guiPrms	float	277.007263	0x0000A834@Data
guiPowerFactor	float	0.990778685	0x0000A82E@Data
guiFreqAvg	float	60.0219727	0x0000A844@Data
EPwm1Regs.TZFLG	Register	0x0000	
EPwm2Regs.TZFLG	Register	0x0000	
dutyPU	float	-0.4262546	0x0000A86E@Data
dutyPU_DC	float	0.5	0x0000A86C@Data
il1_sensed	float	0.0561523438	0x0000A884@Data
il2_sensed	float	-0.0673828125	0x0000A880@Data
il3_sensed	float	-0.0434570313	0x0000A882@Data
autoStartSlew	unsigned long	5	0x0000A83E@Data
+ Add new expression			

图 5-43. 构建实验 4：施加交流电压后的表达式视图

本构建将 SFRA 集成在软件中，通过在硬件上进行测量来验证设计的补偿器是否能够提供足够的增益裕量和相位裕度。要运行 SFRA，请使工程保持运行状态，然后导航至 `<Install directory>\C2000Ware_DigitalPower_SDK_<version>\libraries\sfra\gui\SFRA_GUI.exe`。然后出现 SFRA GUI。

在 SFRA GUI 上选择器件的相应选项。例如，对于 F28003x，选择“Floating Point”。点击 **Setup Connection**，然后在弹出窗口中取消选择“boot on connect”选项，然后选择相应的 COM 端口。点击 **OK**。返回到 SFRA GUI，然后点击 **Connect**。

SFRA GUI 将连接到器件。现在可以通过点击 **Start Sweep** 启动 SFRA 扫描。完整的 SFRA 扫描需要几分钟才能完成。可以通过查看 SFRA GUI 上的进度条以及检查用于指示 UART 活动的控制卡背面蓝色 LED 的闪烁情况来监视活动。完成之后，将立即显示一个包含开环图的图，如图 5-44 中所示。该操作证明设计的补偿器确实是稳定的。

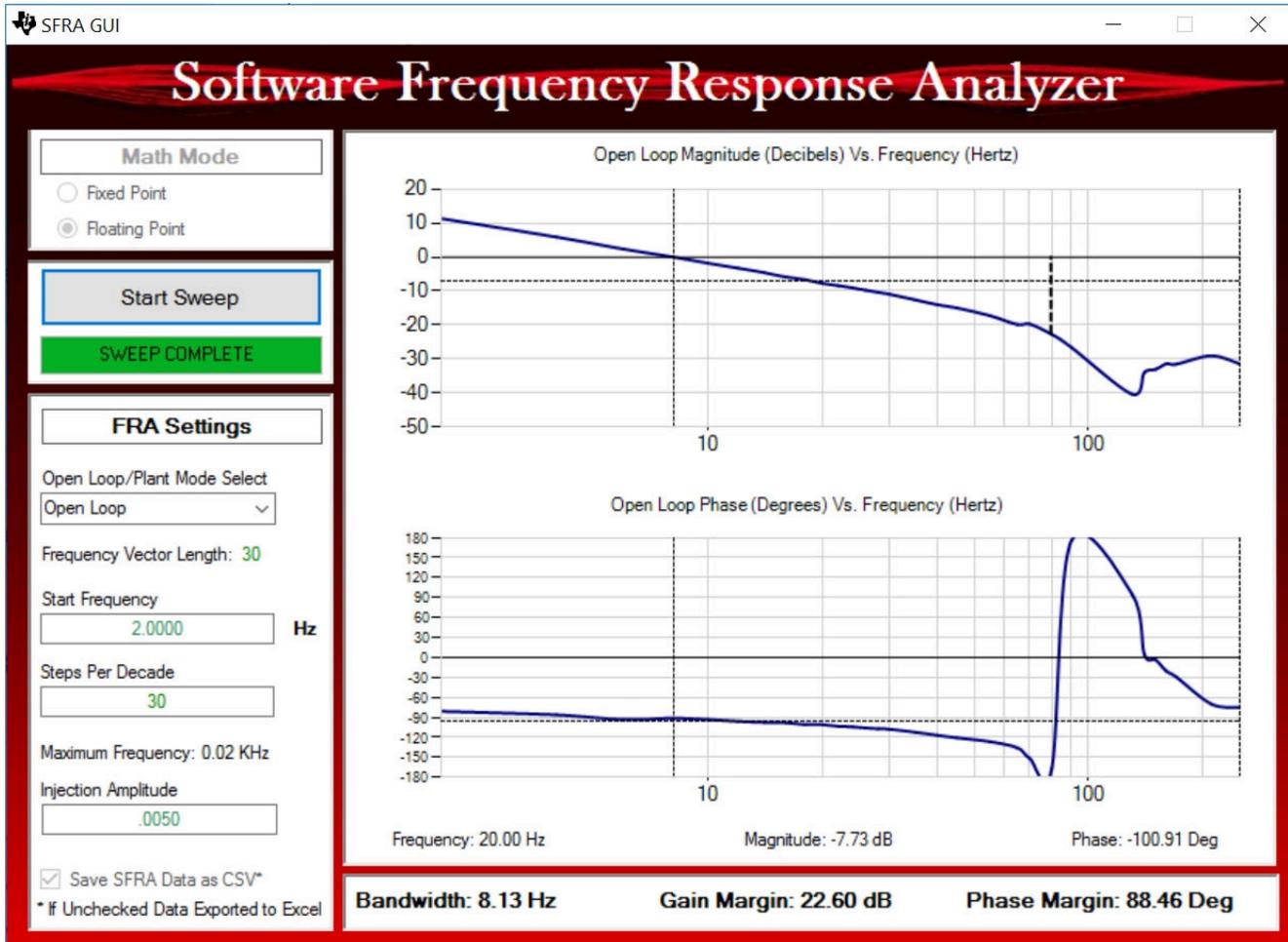


图 5-44. 在闭合电压环路上运行的 SFRA

或者，重新打开补偿设计器，然后针对 GUI 上的受控体选项选择 *SFRA Data*。该选项使用测量受控体信息来设计补偿器，并且可用于微调补偿。默认情况下，补偿设计器指向最近运行的 SFRA。如果必须使用运行的前一个 SFRA 受控体信息，则选择 *SFRADData.csv* 文件（通过点击 *Browse SFRA Data* 浏览到该文件）。关闭补偿设计器，这将验证电压补偿器设计。

若要使系统安全停止，请将输入交流电压降至零。确保 *guiVBus* 也降至零。

在实时模式下，完全停止 MCU 需要执行两个步骤。首先，使用工具栏上的 *Halt* 按钮 () 或使用 *Target* → *Halt* 来暂停处理器。然后点击  ，使 MCU 退出实时模式。最后，将 MCU 复位 ()。

点击 *Terminate Debug Session* (终止调试会话) (*Target* (目标) → *Terminate all* (终止全部)) 关闭 CCS 调试会话。



5.2.4 测试结果

该设计实现的功率密度为 3.8kW/L (62.5W/in³)。总系统效率为 96.5%。PFC 的效率为 98.5%，CLLLC 的效率为 98%。

5.2.4.1 效率

下面的图提供了使用和不使用 12V 辅助电源时的效率数据。辅助电源为控制、隔离器和栅极驱动器供电。图 5-45 中的图是在以下条件下获取的：

- $V_{IN,RMS} = 240V$
- $V_{OUT} = 400V$
- 冷却液温度：20° C

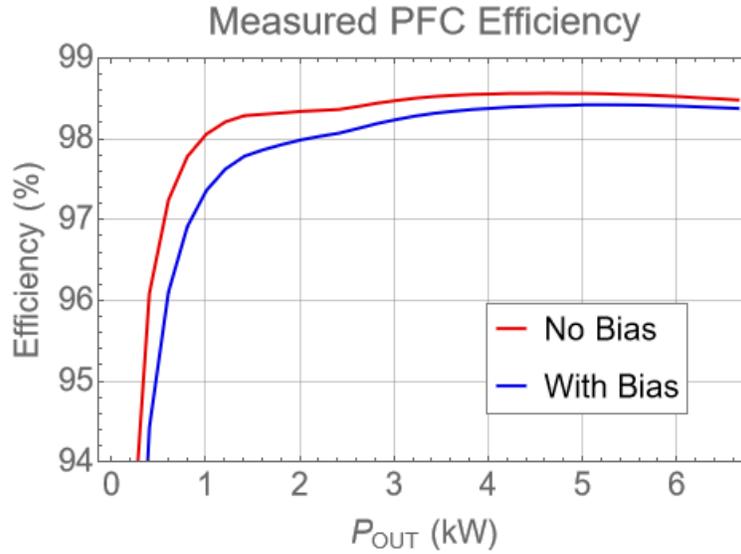


图 5-45. PFC 效率

图 5-46 中的图是在以下条件下获取的：

- $V_{IN} = 400V$
- $V_{OUT} = 350V$
- 冷却液温度：20° C

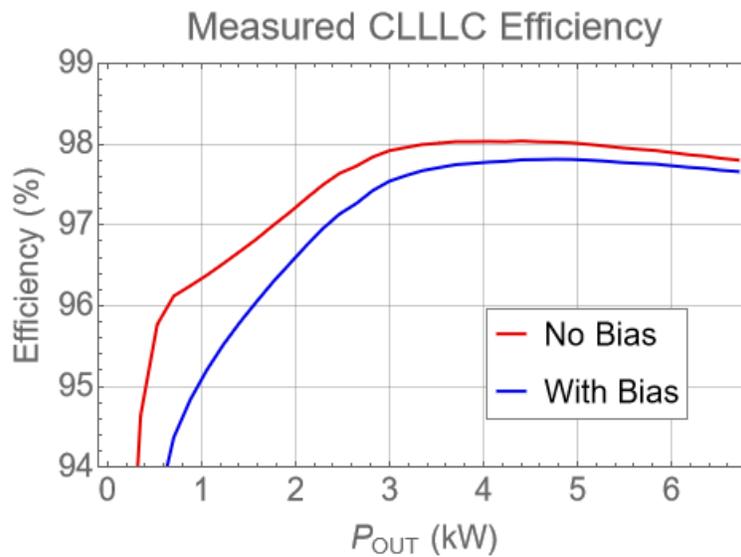


图 5-46. CLLLC 效率

图 5-47 中的图是在以下条件下获取的：

- $V_{IN,RMS} = 240V$
- $V_{OUT} = 350V$
- 冷却液温度：20° C

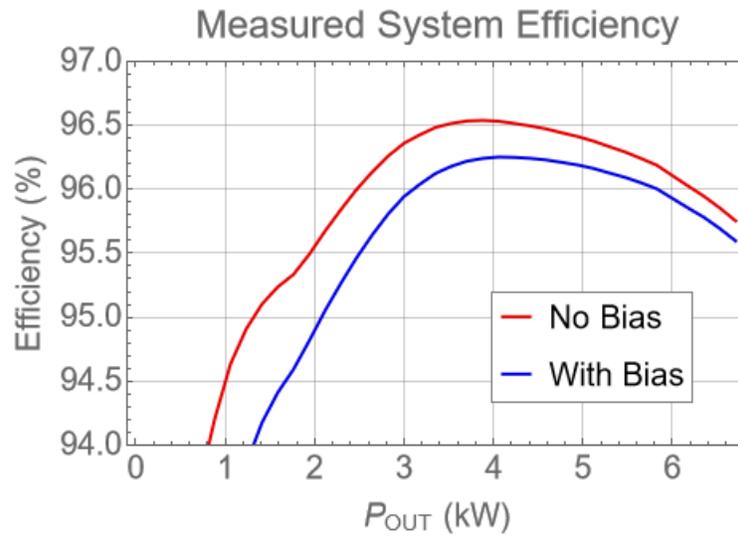


图 5-47. 系统效率

5.2.4.2 系统性能

下面的图汇总了总系统效率、系统损耗、总谐波失真 (THD) 和归一化输出电压调节精度。

该设计实现的功率密度为 3.8kW/L (62.5W/in³)。这产生的总系统效率为 96.5%。高于 1.5kW 的负载的 THD 小于 5%，输出电压的调节精度大于处于 ±0.06% 以内。

图 5-48 中的图使用以下条件：

- $V_{IN,RMS} = 240V$
- $V_{OUT} = 350V$
- 冷却液温度：20° C

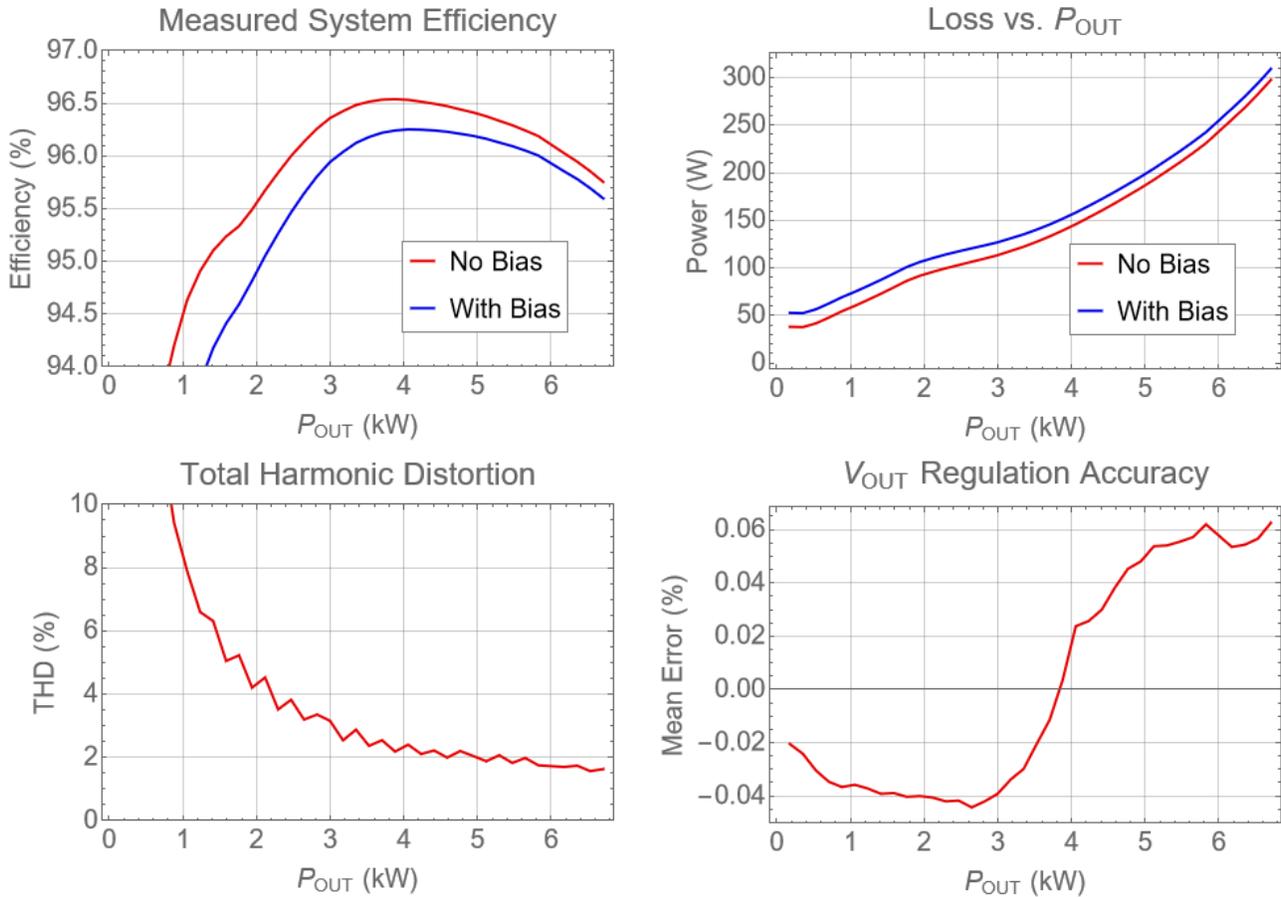


图 5-48. 系统性能

5.2.4.3 波特图

以下波特图是使用 TMS320F28388D 微控制器内部的板载软件频率响应分析仪获取的。测试中使用的负载被配置为恒流阱。微控制器被配置为调节恒定的输出电压。带宽约为 1kHz 至 2.5kHz，相位裕度超过 45°。

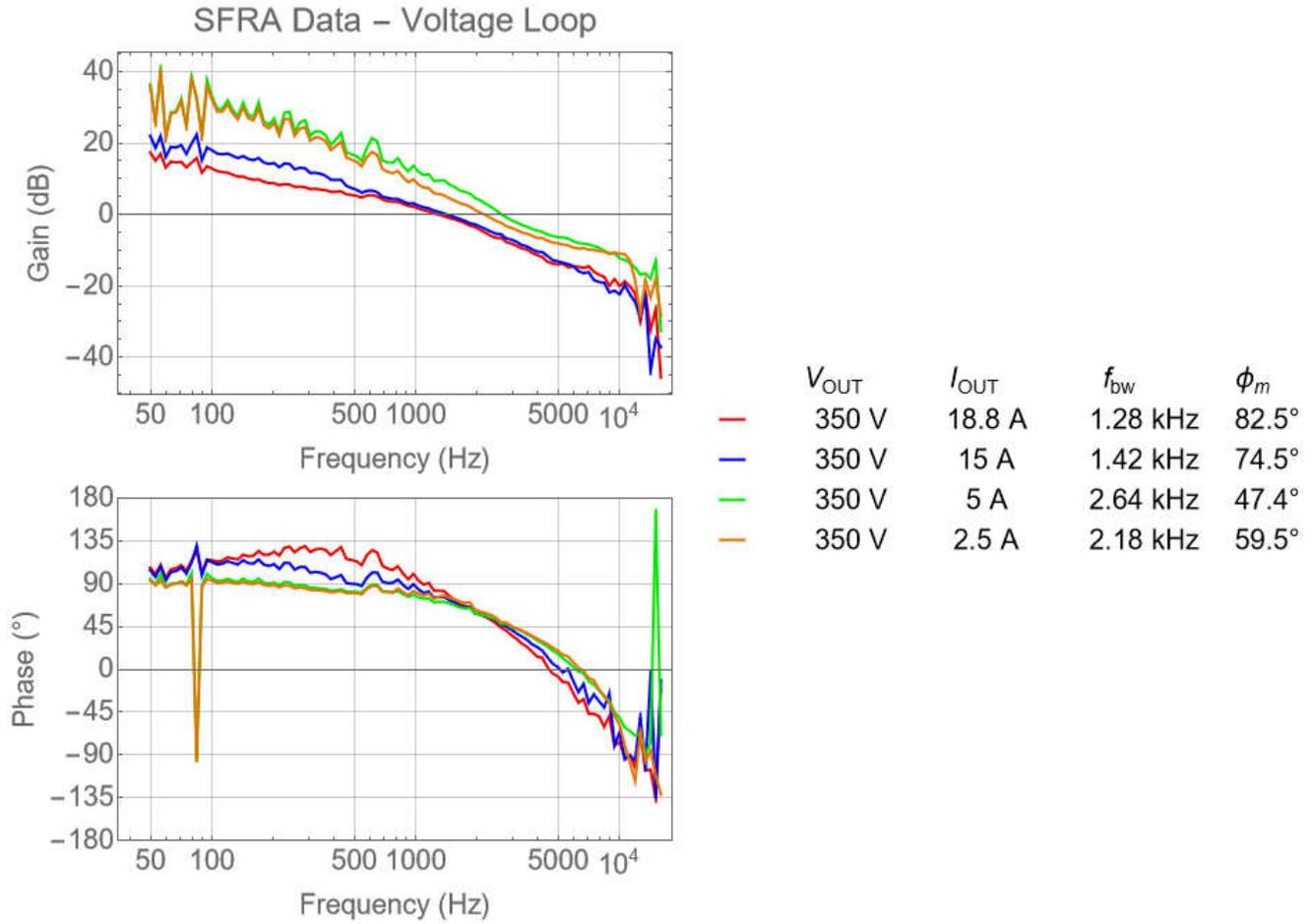


图 5-49. 控制环路波特图

以下波特图是使用 TMS320F28388D 微控制器内部的板载软件频率响应分析仪获取的。测试中使用的负载被配置为恒压。微控制器被配置为调节恒定的输出电流。带宽约为 1kHz 至 2.5kHz，相位裕度超过 60°。

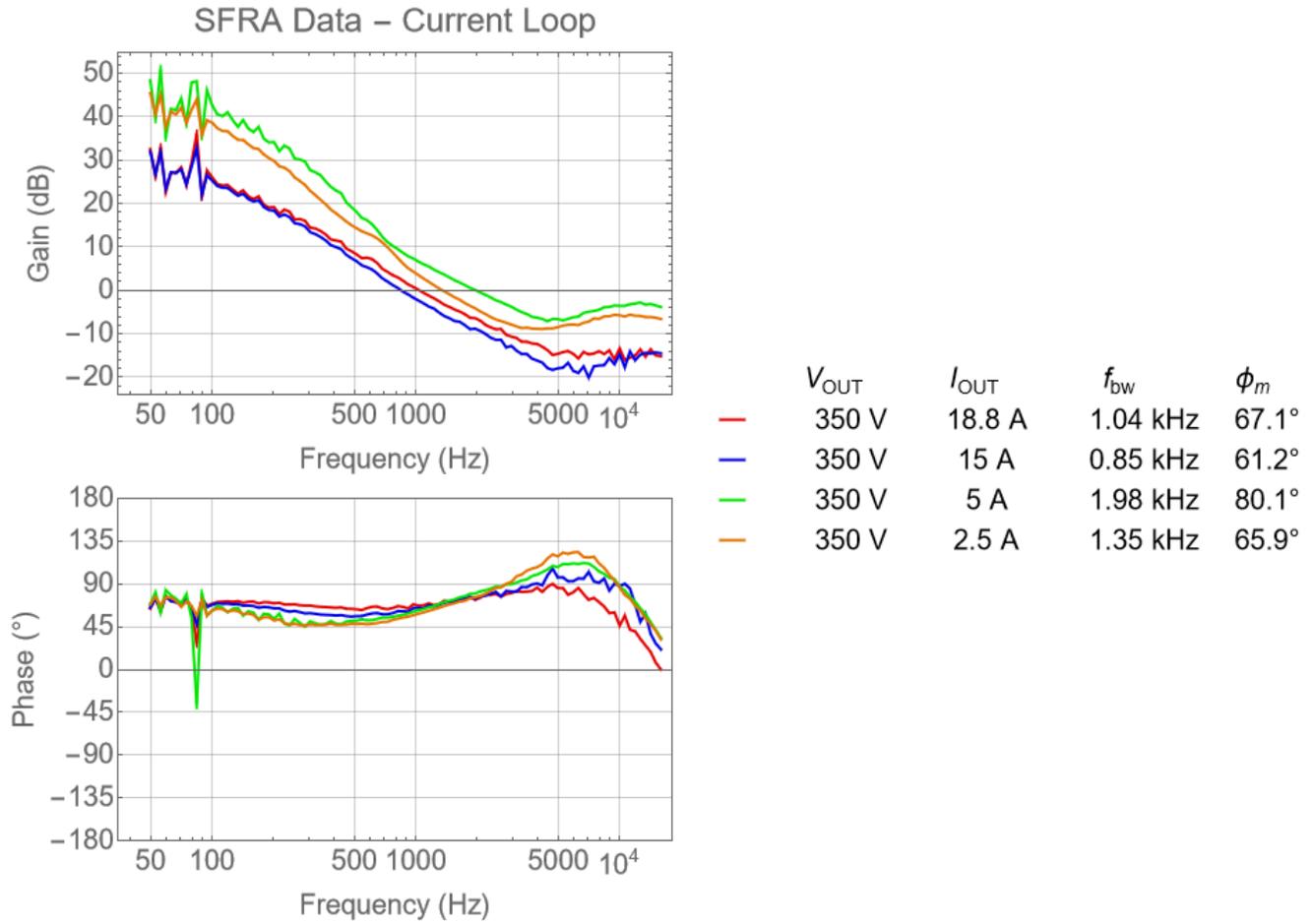


图 5-50. 电压环路波特图 (恒压负载)

5.2.4.4 效率和调节数据

下表显示了效率和调节数据。

V _{OUT} (V)	I _{OUT} (A)	P _{OUT} (W)	V _{IN} (V)	I _{IN} (A)	P _{IN} (W)	V _{BIAS} (V)	I _{BIAS} (A)	P _{BIAS} (W)	无偏置时的效率 (%)	有偏置时的效率 (%)
352.76	0.5	177.35	240.12	1.17	215.36	11.92	1.24	14.73	82.35	77.08
352.74	1	354.12	240.05	1.79	391.84	11.92	1.23	14.67	90.37	87.11
352.72	1.5	530.65	239.98	2.49	572.25	11.92	1.24	14.8	92.73	90.39
352.71	2	706.27	239.91	3.23	753.97	11.92	1.24	14.78	93.67	91.87
352.7	2.5	882.52	239.84	3.97	936.83	11.92	1.24	14.76	94.2	92.74
352.7	3	1058.9	239.76	4.72	1118.95	11.92	1.23	14.71	94.63	93.41
352.7	3.5	1235.3	239.69	5.48	1301.58	11.92	1.23	14.65	94.91	93.85
352.69	4	1411.74	239.62	6.23	1484.44	11.92	1.23	14.61	95.1	94.18
352.69	4.5	1587.99	239.55	6.99	1667.43	11.92	1.22	14.57	95.24	94.41
352.69	5	1763.62	239.47	7.75	1849.94	11.92	1.22	14.51	95.33	94.59
352.69	5.5	1939.96	239.4	8.51	2031.56	11.93	1.21	14.48	95.49	94.82
352.69	6	2116.85	239.32	9.27	2212.59	11.93	1.21	14.48	95.67	95.05
352.68	6.5	2293.18	239.25	10.02	2392.72	11.93	1.21	14.46	95.84	95.26
352.68	7	2469.51	239.17	10.78	2572.43	11.93	1.21	14.4	96	95.46
352.67	7.5	2645.17	239.1	11.53	2751.5	11.93	1.19	14.16	96.14	95.64
352.68	8	2821.8	239.02	12.29	2931.48	11.94	1.16	13.85	96.26	95.81
352.69	8.5	2998.24	238.95	13.04	3111.5	11.94	1.14	13.56	96.36	95.94
352.71	9	3174.78	238.87	13.8	3292.5	11.95	1.11	13.26	96.43	96.04
352.72	9.5	3350.46	238.79	14.56	3472.6	11.95	1.09	12.99	96.48	96.12
352.76	10	3527.33	238.71	15.33	3654.7	11.95	1.07	12.76	96.51	96.18
352.79	10.5	3704.19	238.63	16.09	3837.2	11.96	1.05	12.52	96.53	96.22
352.84	11	3881.08	238.54	16.87	4020.3	11.96	1.03	12.37	96.54	96.24
352.91	11.5	4058.4	238.46	17.64	4204.2	11.96	1.02	12.24	96.53	96.25
352.92	12	4235	238.37	18.42	4387.9	11.96	1.02	12.16	96.51	96.25
352.93	12.5	4411.3	238.28	19.2	4571.5	11.96	1.01	12.08	96.5	96.24
352.96	13	4588.2	238.18	19.98	4756	11.96	1	12	96.47	96.23
352.99	13.5	4765.1	238.09	20.77	4940.9	11.96	1	11.92	96.44	96.21
353	14	4941.9	237.99	21.55	5125.8	11.97	0.99	11.86	96.41	96.19
353.02	14.5	5118.9	237.89	22.34	5311.4	11.97	0.99	11.79	96.38	96.16
353.02	15	5294.8	237.79	23.13	5496.4	11.97	0.98	11.73	96.33	96.13
353.02	15.5	5471.6	237.68	23.92	5682.4	11.97	0.98	11.69	96.29	96.09
353.03	16	5648.3	237.62	24.71	5868.9	11.97	0.97	11.64	96.24	96.05
353.05	16.5	5825.2	237.51	25.51	6056.1	11.97	0.97	11.59	96.19	96
353.02	17.5	6176.6	237.26	27.13	6432.3	11.97	0.96	11.52	96.03	95.85
353.02	18	6353.2	237.14	27.94	6621.6	11.97	0.96	11.49	95.95	95.78
353.03	18.5	6529.8	237.01	28.76	6812.2	11.97	0.96	11.47	95.85	95.69

V_{OUT} (V)	I_{OUT} (A)	P_{OUT} (W)	V_{IN} (V)	I_{IN} (A)	P_{IN} (W)	V_{BIAS} (V)	I_{BIAS} (A)	P_{BIAS} (W)	无偏置时的效率 (%)	有偏置时的效率 (%)
353.05	19	6706.8	236.87	29.59	7004.3	11.97	0.96	11.44	95.75	95.6

5.2.4.5 散热数据

图 5-51 是在满负载运行的情况下获取的。所有重要的发热元件都连接到电路板底部的冷板。该图中显示的最热元件来自 EMI 滤波器中的共模电感器。这些器件无法接触冷板，其所有冷却效果都来自周围空气。

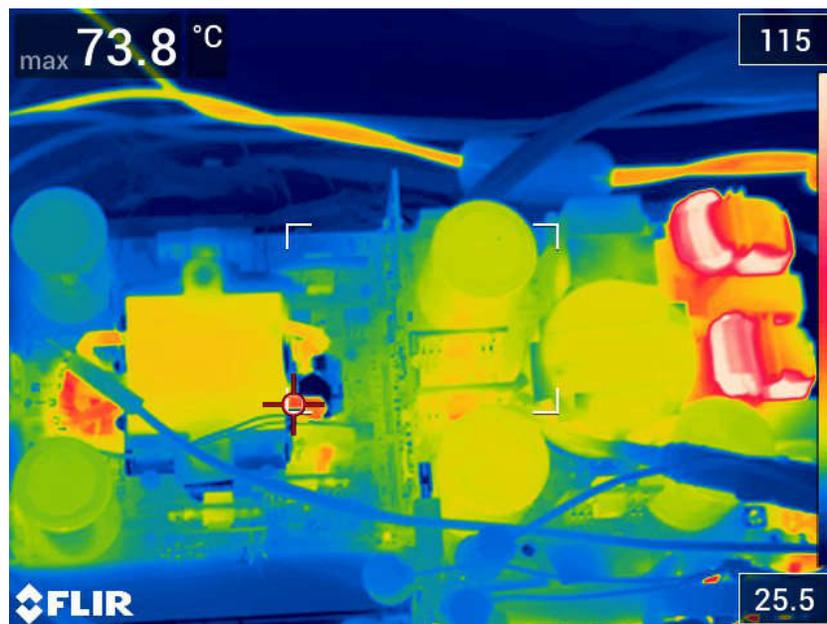


图 5-51. 顶部热图像

GaN FET 温度通过 LMG3522 器件内部的板载温度传感器提供。在满载条件下，所有 FET 温度都低于 75°C。

表 5-5 列出了以下条件下的 GaN FET 温度测量值：

- $V_{IN,AC}$: 240V
- $V_{DC,LINK}$: 400V
- 冷却液温度 : 33°C

表 5-5. GaN FET 温度测量

GaN FET	温度 (°C)
PFC	66.8
CLLLC 初级 (350V/19A)	58.1
CLLLC 次级 (350V/19A)	59.5
CLLLC 初级 (300V/19A)	61.0
CLLLC 次级 (300V/19A)	74.0

图 5-52 展示了以下条件下的变压器临界温度：

- 冷却液温度：33°C
- 变压器温度测量位置
 - 初级 1 - 在初级绕组的内表面测量
 - 初级 2 - 在初级绕组的外表面测量
 - 次级 1 - 在次级绕组的内表面测量
 - 次级 2 - 在次级绕组的外表面测量
 - 磁芯 1 - 在磁芯中心支腿顶部测量
 - 磁芯 2 - 在磁芯中心支腿底部测量
 - 磁芯 3 - 在磁芯侧面测量
 - 磁芯 4 - 在磁芯顶部测量

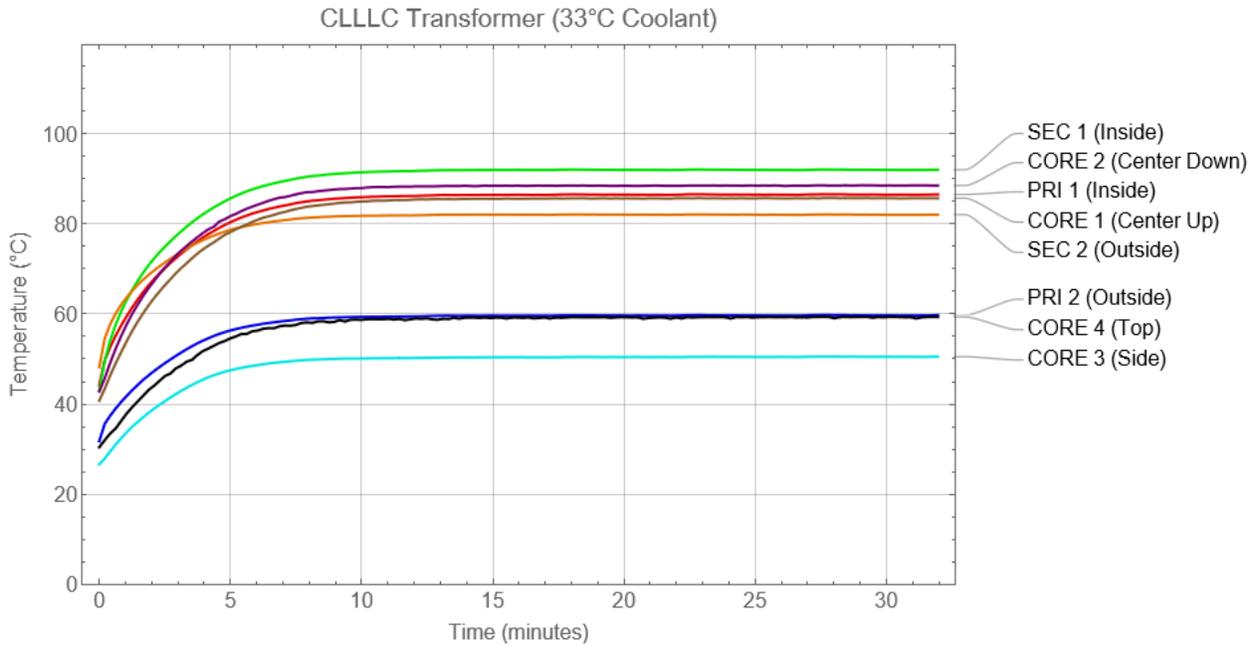


图 5-52. CLLLC 变压器温度

5.2.4.6 PFC 波形

图 5-53 展示了在以下参数下测量的 PFC 输入电压和输入电流波形：

- 迹线
 - C2 : V_{IN}
 - C4 : I_{IN}
- 工作条件
 - $V_{IN} = 208V$
 - $V_{OUT} = 400V$
 - $R_{OUT} = 43\Omega$

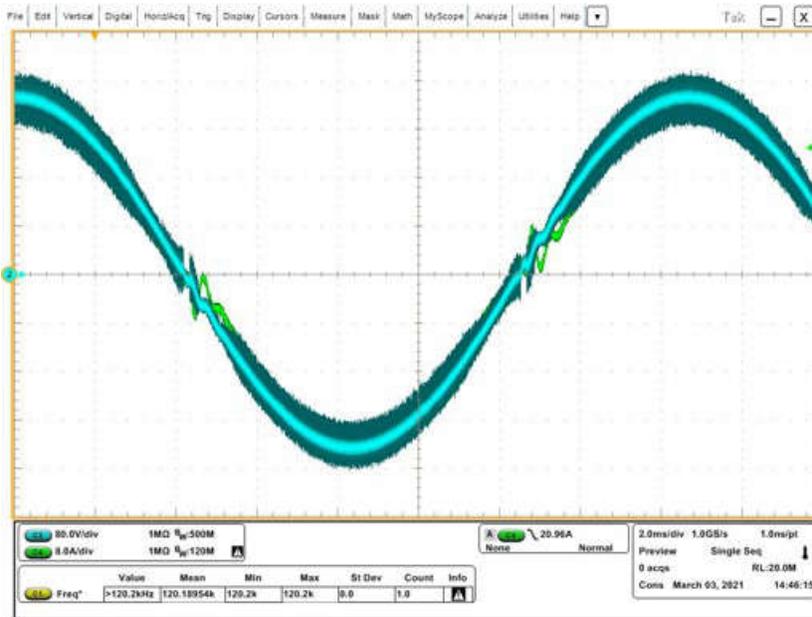


图 5-53. PFC 输入电压和输入电流

图 5-54 展示了在以下参数下测量的 PFC GaN 漏极电压波形：

- 迹线
 - C1 : GaN 开关节点漏极电压
 - C2 : V_{IN}
 - C4 : I_{IN}
- 工作条件
 - $V_{IN} = 208V$
 - $V_{OUT} = 400V$
 - $R_{OUT} = 43\Omega$

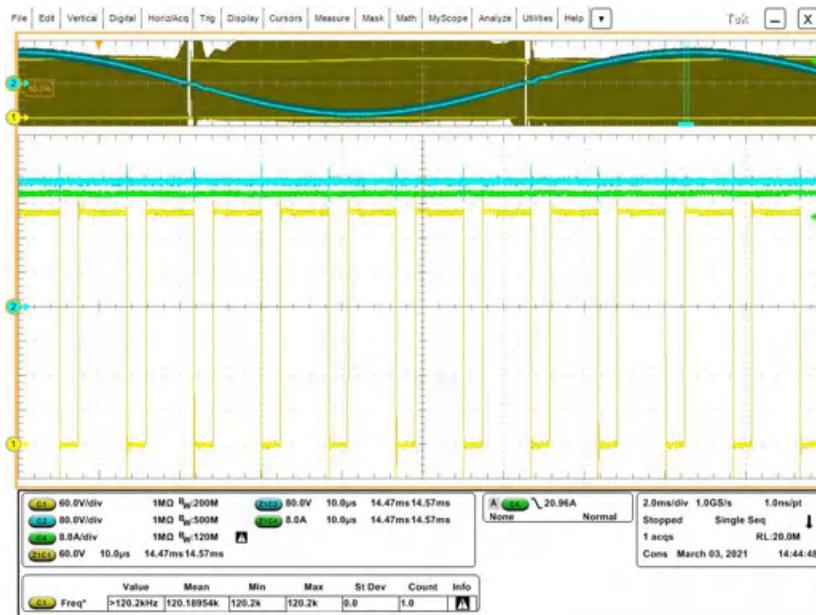


图 5-54. PFC GaN 漏极电压

图 5-55 展示了约 20ns 的 GaN 开关漏源电压转换放大图。该快速转换源于 LMG3522 的低 C_{OSS} 。

图 5-55 中的波形是使用以下参数测量的：

- 迹线
 - C1 : GaN 开关节点漏极电压
 - C2 : V_{IN}
 - C4 : I_{IN}
- 工作条件
 - $V_{IN} = 208V$
 - $V_{OUT} = 400V$
 - $R_{OUT} = 43\Omega$

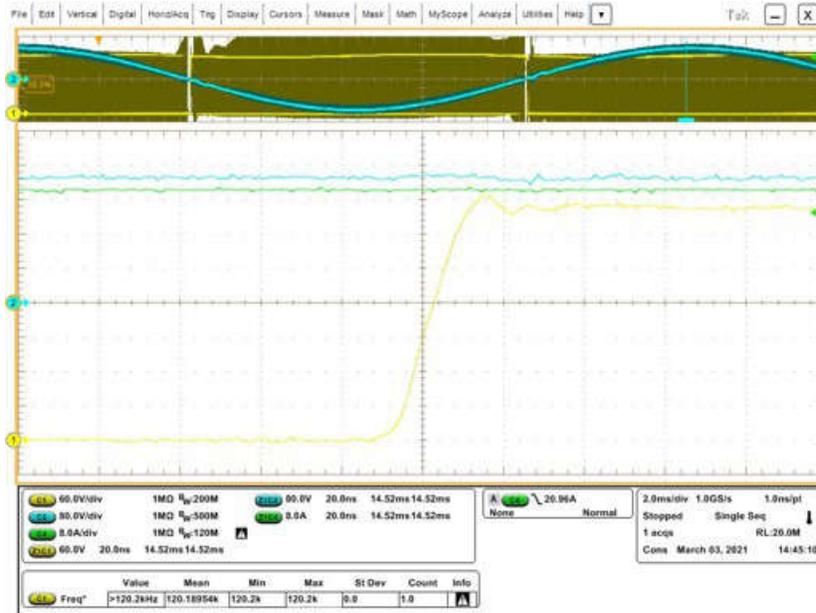
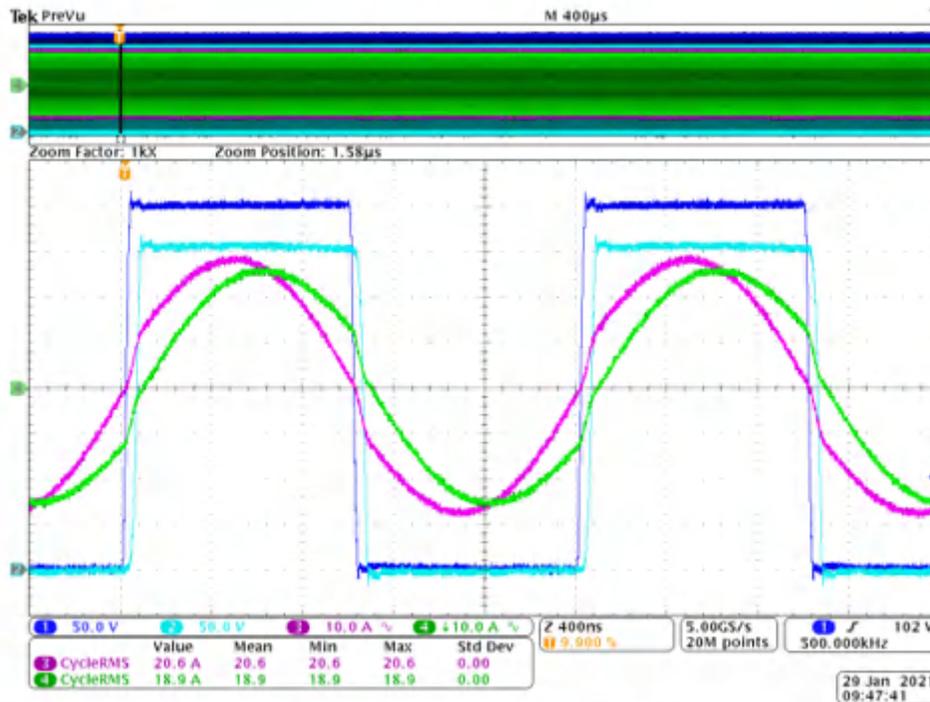


图 5-55. PFC GaN 漏极电压 - 转换

5.2.4.7 CLLLC 波形

图 5-56 展示了在 19A (6.6kW) 以及以下参数下的 CLLLC 运行情况：

- 迹线
 - C1 : GaN 初级开关节点漏极电压
 - C2 : GaN 次级开关节点漏极电压
 - C3 : 变压器初级电流
 - C4 : 变压器次级电流
- 工作条件
 - $V_{IN} = 400V$
 - $V_{OUT} = 350V$
 - $I_{OUT} = 19A$



DPO4104B - 8:52:04 AM 1/29/2021

图 5-56. CLLLC 运行 19A (6.6kW)

图 5-57 展示了在 10A 及以下参数下的 CLLLC 运行情况：

- 迹线
 - C1 : GaN 开关节点漏极电压桥臂 1
 - C2 : GaN 开关节点漏极电压桥臂 2
 - C3 : 变压器初级电流
 - C4 : 变压器次级电流
- 工作条件
 - $V_{IN} = 400V$
 - $V_{OUT} = 350V$
 - $I_{OUT} = 10A$

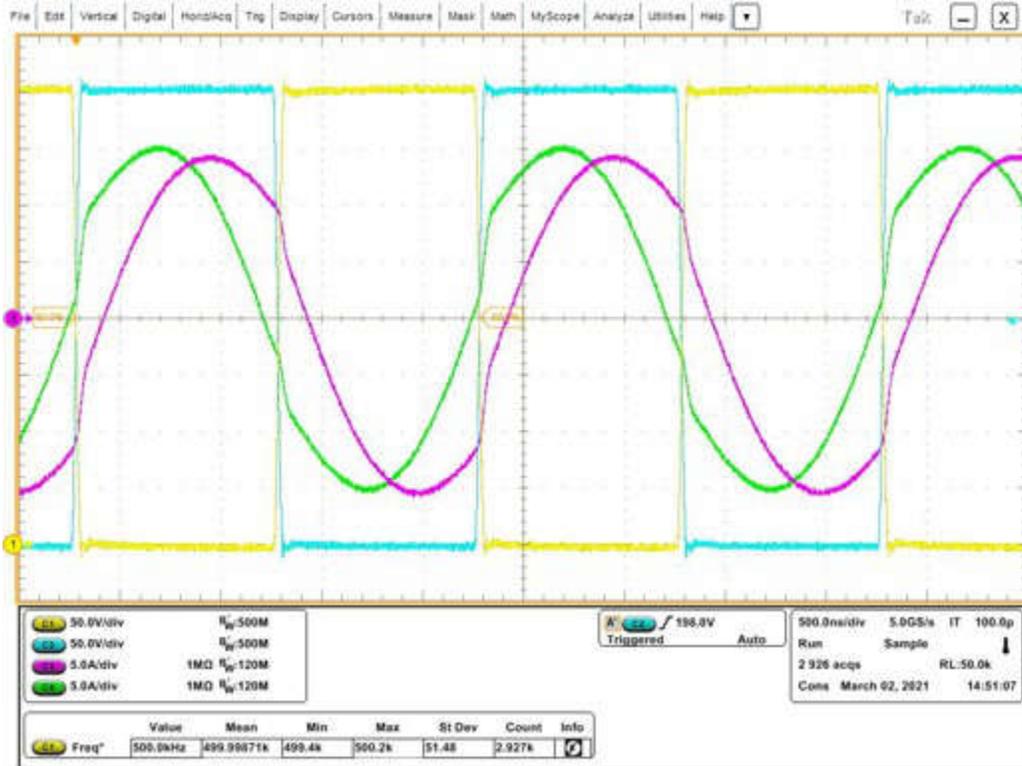


图 5-57. CLLLC 运行 10A

图 5-58 展示了约 40ns 的 GaN 开关漏源电压转换放大图。该快速转换源于 LMG3522 的低 C_{OSS} 。

图 5-58 中的波形是使用以下参数测量的：

- 迹线
 - C1 : GaN 开关节点漏极电压桥臂 1
 - C2 : GaN 开关节点漏极电压桥臂 2
 - C3 : 变压器初级电流
 - C4 : 变压器次级电流
- 工作条件
 - $V_{IN} = 400V$
 - $V_{OUT} = 350V$
 - $I_{OUT} = 10A$

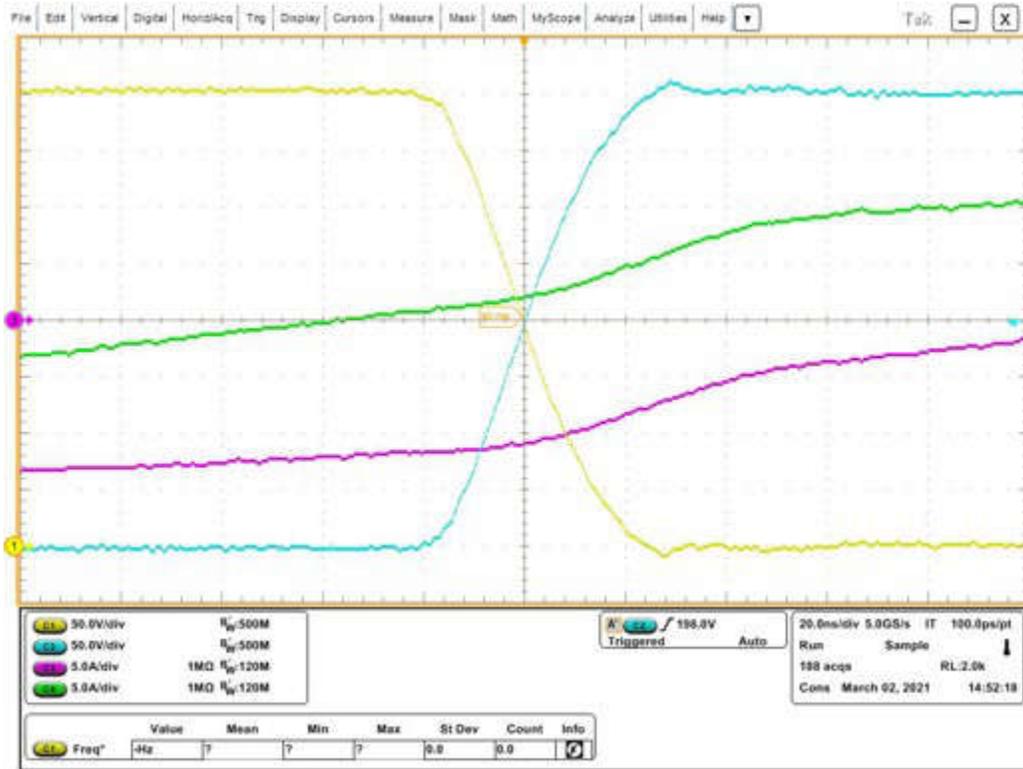


图 5-58. CLLLC 运行 10A - GaN FET 转换

图 5-59 中的波形是使用以下参数测量的：

- 迹线
 - C1 : GaN 开关节点漏极电压桥臂 1
 - C2 : GaN 开关节点漏极电压桥臂 2
 - C3 : 变压器初级电流
 - C4 : 变压器次级电流
- 工作条件
 - $V_{IN} = 400V$
 - $V_{OUT} = 350V$
 - $I_{OUT} = 2A$

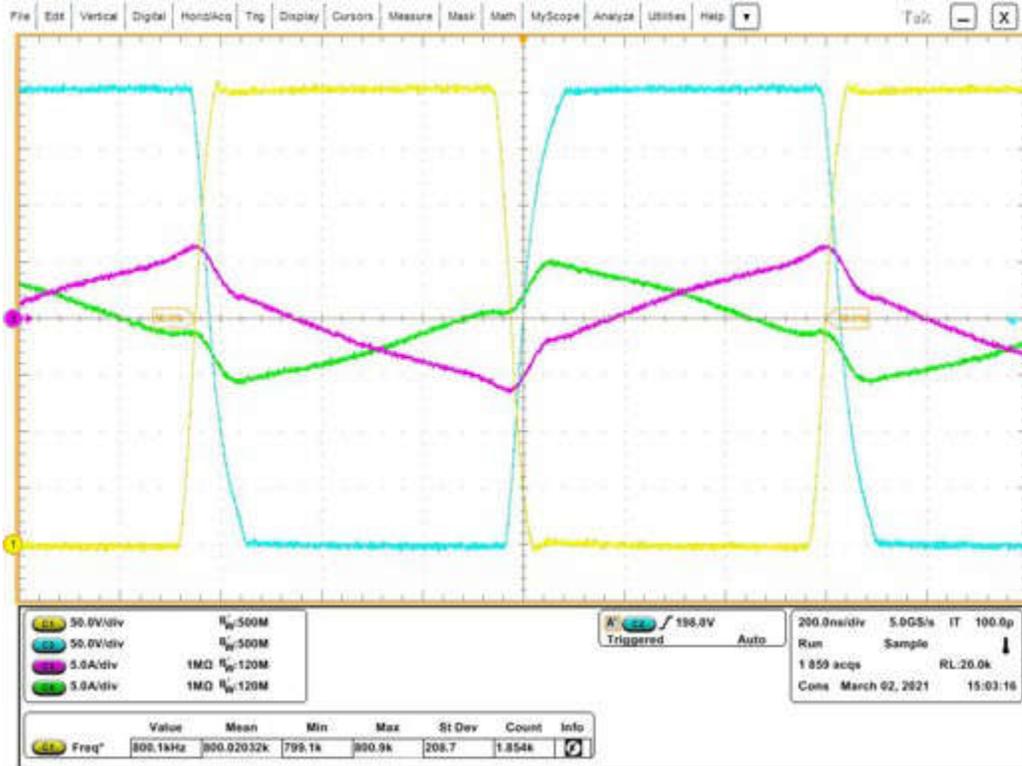


图 5-59. CLLLC 运行 2A

图 5-60 展示了约 75ns 的 GaN 开关漏源电压转换放大图。该快速转换源于 LMG3522 的低 C_{OSS} 。该图中稍长的转换时间源于较轻的负载条件和由此产生的较小电流。

图 5-60 中的波形是使用以下参数测量的：

- 迹线
 - C1 : GaN 开关节点漏极电压桥臂 1
 - C2 : GaN 开关节点漏极电压桥臂 2
 - C3 : 变压器初级电流
 - C4 : 变压器次级电流
- 工作条件
 - $V_{IN} = 400V$
 - $V_{OUT} = 350V$
 - $I_{OUT} = 2A$

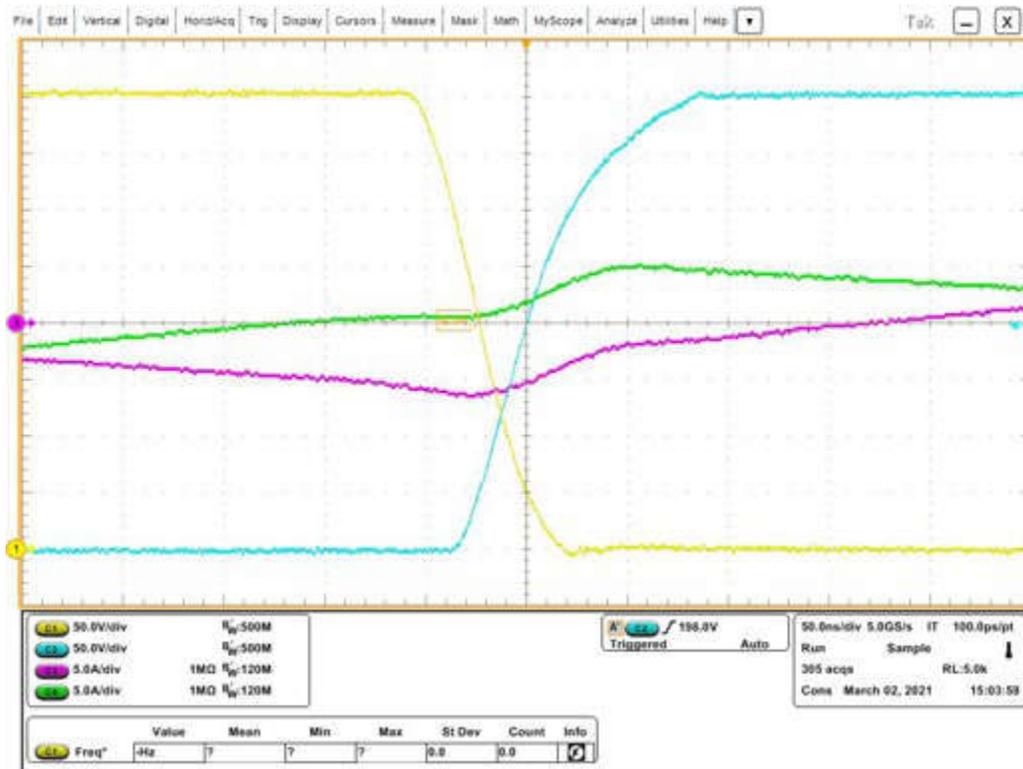


图 5-60. CLLLC 运行 2A - GaN FET 转换

6 设计文件

6.1 原理图

要下载原理图，请参阅 [TIDM-02013](#) 中的设计文件。

6.2 物料清单

要下载物料清单 (BOM)，请参阅 [TIDM-02013](#) 中的设计文件。

6.3 Altium 工程

要下载 Altium Designer® 工程文件，请参阅 [TIDM-02013](#) 中的设计文件。

6.4 Gerber 文件

要下载光绘文件，请参阅 [TIDM-02013](#) 中的设计文件。

7 软件文件

要下载适用于该参考设计的软件，请转至[适用于 C2000 MCU 的 DigitalPower 软件开发套件 \(SDK\)](#) 网站。

8 相关文档

1. 德州仪器 (TI)，[TMS320F28003x 实时微控制器](#)，数据表。
2. 德州仪器 (TI)，[C2000™ 软件频率响应分析器 \(SFRA\) 库和 SDK 框架中的补偿设计器](#)，用户指南。
3. Zaka Ullah Zahid, Zakariya M. Dalala, Rui Chen, Baifeng Chen, and Jih-Sheng Lai, *Design of Bidirectional DC - DC Resonant Converter for Vehicle-to-Grid (V2G) Applications*, *IEEE Transactions on Transportation Electrification*, Vol. 1, No. 3, October 2015, pp. 232 - 244.
4. Zakariya M. Dalala, Zaka Ullah Zahid, Osama S. Saadeh, Jih-Sheng Lai, *Modeling and Controller Design of a Bidirectional Resonant Converter Battery Charger*, *IEEE Access*, Vol. 6, April 2018, pp. 23338 - 23350.
5. Biao Zhao, Qiang Song, Wenhua Liu, and Yandong Sun, *Overview of Dual-Active-Bridge Isolated Bidirectional DC - DC Converter for High-Frequency-Link Power-Conversion System*, *IEEE Transactions on Power Electronics*, Vol. 29, No. 8, August 2014, pp. 4091 - 4106.
6. Joel Turchi, J.T., Dhaval Dalal, D.D., Patrick Wang, P.T., and Laurent Lenck, L.L.(2014).[Power Factor Correction \(PFC\) Handbook: Choosing the Right Power Factor Controller Solution](#). Revision 5, <http://www.onsemi.com/pub/Collateral/HBD853-D.PDF>
7. 德州仪器 (TI)，[图腾柱 PFC 中的控制挑战](#)，模拟应用期刊。

8.1 商标

C2000™, TI E2E™, Piccolo™, TMS320C2000™, and Code Composer Studio™ are trademarks of Texas Instruments.

Altium Designer® is a registered trademark of Altium LLC or its affiliated companies.

所有商标均为其各自所有者的财产。

9 术语

首字母缩写词	定义
BCM	电池充电模式
BW	带宽
CAN	控制器局域网
CCM	连续传导模式
CCS	Code Composer Studio
CLA	控制律加速器
CLB	可配置逻辑块
CLLLC	电容器电感器电感器电感器电容器
CMPSS	比较器子系统
CMTI	共模瞬态抗扰度
DAB	双有源电桥
DAC	数模转换器
DCM	不连续导通模式[Discontinuous Conduction Mode]
DLOG	数据记录器
DT	死区时间
DUT	被测设计
eCAP	增强型捕捉
ePWM	增强型脉宽调制器
ERAD	嵌入式实时分析和诊断
EV	电动汽车
FET	场效应晶体管
FHA	一次谐波分析
FSI	快速串行接口
GaN	氮化镓
HEV	混合动力电动汽车
HRPWM	高分辨率脉宽调制器
HSEC	高速边缘卡
I2C	集成电路总线
IDE	集成开发环境
IGBT	绝缘栅双极晶体管
ISR	中断处理例程
KCL	基尔霍夫电流定律
KVL	基尔霍夫电压定律
LIN	本地互连网络
MOSFET	金属氧化物半导体场效应晶体管
OBC	车载充电器
PFC	功率因数校正
PGA	可编程增益放大器
PMBus	电源管理总线
SCI	串行通信接口
SDFM	Σ - Δ 滤波器模块
SFRA	软件频率响应分析器
SiC	碳化硅
SPI	串行外设接口
SRC	串联谐振转换器
ZCS	零电流开关
ZVS	零电压开关

10 作者简介

Cody Watkins 是德州仪器 (TI) C2000 微控制器团队的一名应用工程师。Cody 于 2015 年获得辛辛那提大学电气工程学位。Cody 对替代能源、可持续发展和能源自给自足有浓厚的兴趣。

11 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 * (October 2022) to Revision A (February 2024)	Page
• 该设计中添加了对 TMS320F28P65x 微控制器的软件支持。.....	1

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 © 2024，德州仪器 (TI) 公司