

TI 参考设计

4.4V 至 30V、15A 高性能直流无刷无传感矢量螺旋桨电调参考设计



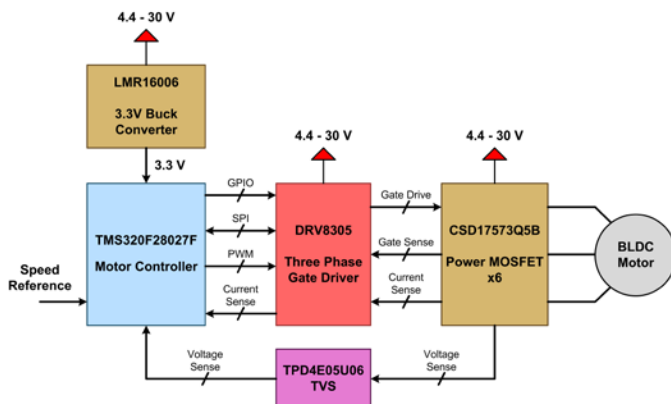
设计概述

TIDA-00643 是一款4.4V 至 30V 直流无刷无传感矢量 电调参考设计，适用于大功率螺旋 桨、风扇以及泵等无传感电机控制应用。该设计采用德州仪器 (TI) 的 DRV8305 直流无刷电机栅极驱动器、CSD17573Q5B 30V NexFET™ 功率 (MOSFET)、TPD4E05U06 瞬态电压抑制器 (TVS)、C2000 电机控制 MCU 以及 LMR16006 3.3V 降压转换器。它采用了 InstaSPIN™-FOC 无传感矢量电机控制算法并通过来自于外部处理器的信号进行调速。这款设计着重于在大功率直流无刷 (BLDC) 电机系统实现无传感矢量控制。

设计资源

TIDA-00643	设计文件夹
DRV8305	产品文件夹
TMS320F28027F	产品文件夹
LMR16006	产品文件夹
TPD4E05U06	产品文件夹
CSD17573Q5B	产品文件夹
InstaSPIN™	产品文件夹
MotorWare™	工具文件夹
BOOSTXL-DRV8305EVM	工具文件夹

框图



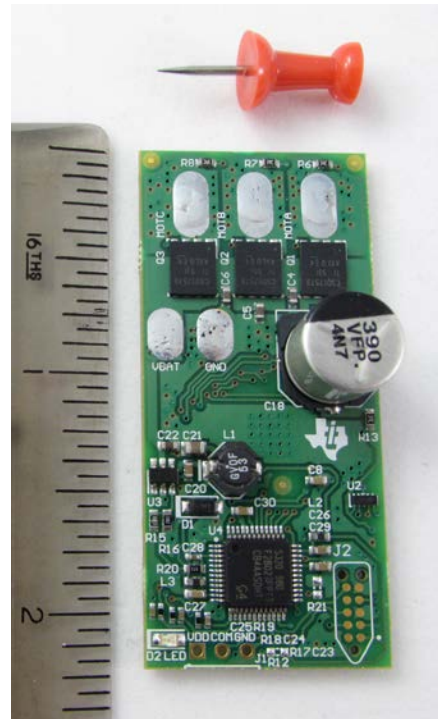
设计特性

- 输入电压范围为 4.4 V 至 30 V
- 15A (RMS)、23A 峰值输出电流
- 小尺寸 (长 x 宽): 2.2" x 1.0"
- 以单个参考信号进行调速
- 3.3V、0.6A 板载降压转换器
- InstaSPIN-FOC™ 无传感矢量 电机控制
- 众多系统保护功能, 包括 MOSFET V_{DS} 过流及欠压保护等

特色应用

- 无人机螺旋桨电子转速控制器
- 无人飞行器
- 带电调的遥控设备
- 大功率风扇和泵

电路板图片



1 系□□明

TIDA-00643 是专为大功率直流无刷 (BLDC) 螺旋桨电调设计的紧凑型无传感矢量电机控制器。它模仿了 RC 和爱好者应用中成熟电子转速控制器 (ESC) 的功能。电机控制器针对中央控制器传输的转速参考信号进行解码并处理为适用于 BLDC 电机的驱动信号。这允许中央控制器卸载与直流电机控制相关的处理器功能而专注于系统的中心功能。此类电机控制器也可以用于大功率风扇和泵应用。

该电机控制器主要由两个组件组成。第一个组件是 MCU，其针对中央控制器传输的转速参考信号进行解码、测量电机的反电动势 (BEMF) 和电流信号并将相应控制信号发送至功率级。第二个组件是功率级，由栅极驱动器和功率 MOSFET 组成。功率级放大从 MCU 传输至电机的控制信号。

电机控制器将无传感矢量控制算法 InstaSPIN-FOC 应用于直流无刷电机。磁场定向控制支持由控制器驱动的电机获得最优效率和噪声性能。InstaSPIN™-FOC 利用电机反电动势和相电流信号判断电机转子的位置，然后发送正确的驱动模式。主电源输入通过一个开关降压转换器为电机控制器供电。

1.1 DRV8305 三相□机□极□□

DRV8305 用作外部 N 沟道功率 MOSFET 的栅极驱动器。MOSFET 与逆变器或三半桥配置相连，以驱动三相 BLDC 电机的各个绕组。栅极驱动器接收微控制器传输的逻辑电平输入并进行放大，然后传送至电机的电池电压轨。DRV8305 是一款高度集成的器件，由单电源供电，所用外部组件最少。该器件针对外部功率 MOSFET 提供精确的转换率控制，以管理系统的开关性能。

DRV8305 通过外部功率 MOSFET 的 V_{DS} 电压监控器提供系统保护。 V_{DS} 监控器可以配置为检测外部 MOSFET 中的过流事件（原因可能是 MOSFET 或者电机短路）。DRV8305 具有电源和栅极电源欠压检测功能，以避免欠驱动及多级温度标志和故障报告。

DRV8305 还提供三个高性能低侧分流放大器监测电机的相电流。分流放大器接收感测电阻两端电压并在放大后传输至微控制器。该放大器具备多种可配置设置，包括增益、消隐时间和基准偏移。

1.2 采用 InstaSPIN-FOC 的 TMS320F28027F Piccolo 微控制器

TMS320F28027F 是电机控制器的大脑。该器件使用电机传输的反馈信号和转速参考信号确定直流无刷电机合适的信号模式。

MCU 支持适用于高效率、高性能应用的无传感矢量算法 InstaSPIN-FOC。无传感器算法无需使用机械电机转子传感器，从而削减系统成本并减重。InstaSPIN™-FOC 使用 TI 的 FAST™（磁通量、角度、转速以及扭矩）软件编码器（无传感器观测器）获取转子位置。

系统通过 MCU 模数转换器（ADC）获取电机反电动势和相电流。高性能增强型脉宽调制器（E PWM）模块为功率级提供 PWM 调制信号，ECAP 模块接收中央控制器传输的转速参考信号。

1.3 LMR16006 降压转换器

LMR16006 将未经稳压的电池输入调节为 3.3V 高效稳压电源。3.3V 电源轨为电机控制器的 C2000 MCU 供电。LMR16006 将 700kHz 开关频率、内部软启动和补偿电路相结合，从而最大限度减少外部引脚数并节省电路板空间。

1.4 CSD17573Q5B NexFET™ N 沟道功率 MOSFET

电机控制器采用 6 个 CSD17573Q5B 构成直流无刷电机逆变器。该功率 MOSFET 是一款 $R_{DS(on)}$ 超低的器件，专为最大限度降低电源转换和开关应用中的损耗而设计。该器件采用紧凑型 8 引脚小外形尺寸无引线（SON）5mm x 6 mm 封装。 $R_{DS(on)}$ 为 0.84m Ω ， V_{GS} 为 10V，从而最大限度降低电路板空间要求并限制散热。

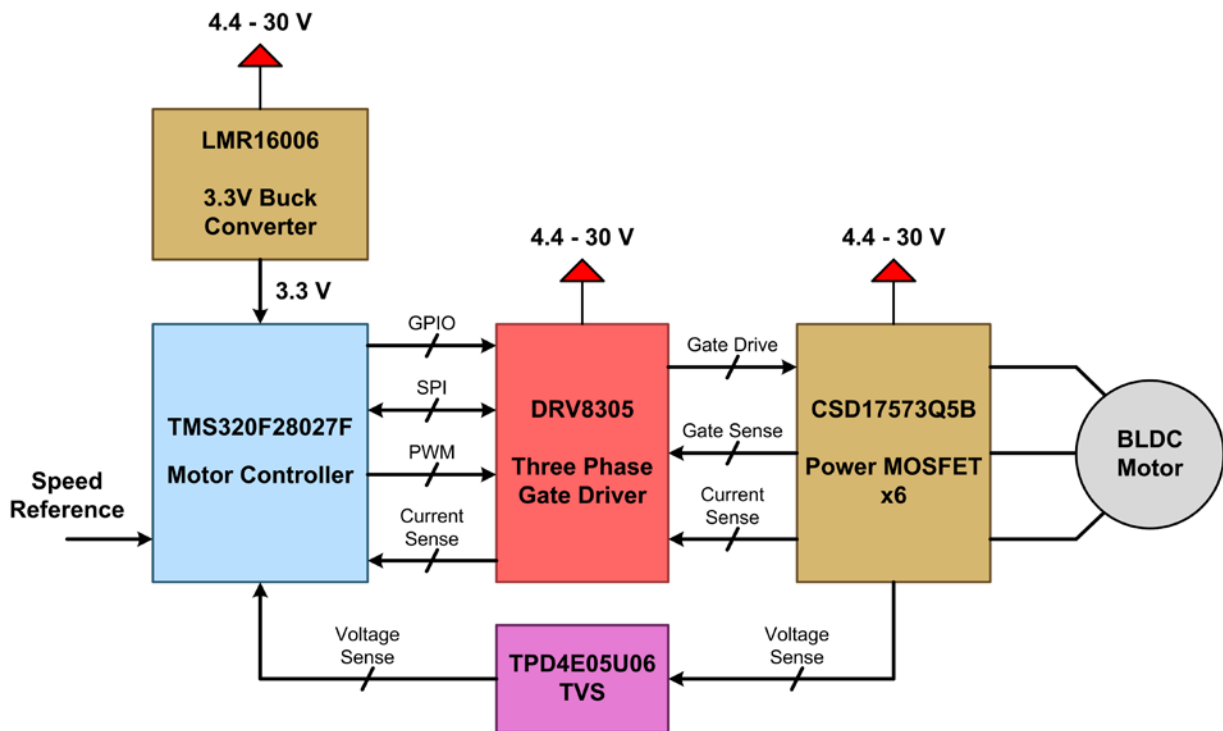
$T_A = 25^\circ\text{C}$		TYPICAL VALUE		UNIT
V_{DS}	Drain-to-Source Voltage	30		V
Q_g	Gate Charge Total (4.5 V)	49		nC
Q_{gd}	Gate Charge Gate-to-Drain	11.9		nC
$R_{DS(on)}$	Drain-to-Source On-Resistance	$V_{GS} = 4.5\text{ V}$	1.19	m Ω
		$V_{GS} = 10\text{ V}$	0.84	m Ω
$V_{GS(th)}$	Threshold Voltage	1.4		V

图 1：CSD17573Q5B 信息

1.5 TPD4E05U06 瞬态电压抑制器

TPD4E05U06 是一款基于单向瞬态电压抑制器（TVS）的静电放电（ESD）保护二极管，具有超低电容。每个器件的 ESD 冲击消散值高于 IEC 61000-4-2 国际标准规定的最高水平。

2 框图



□ 2 : TIDA-00643 框□

2.1 重点□品

本设计的关键器件概述如下。

- [DRV8305](#): 具有分流放大器的三相电机栅极驱动器
- [TMS320F28027F](#): 采用 InstaSPIN-FOC 的 60MHz Piccolo 微控制器

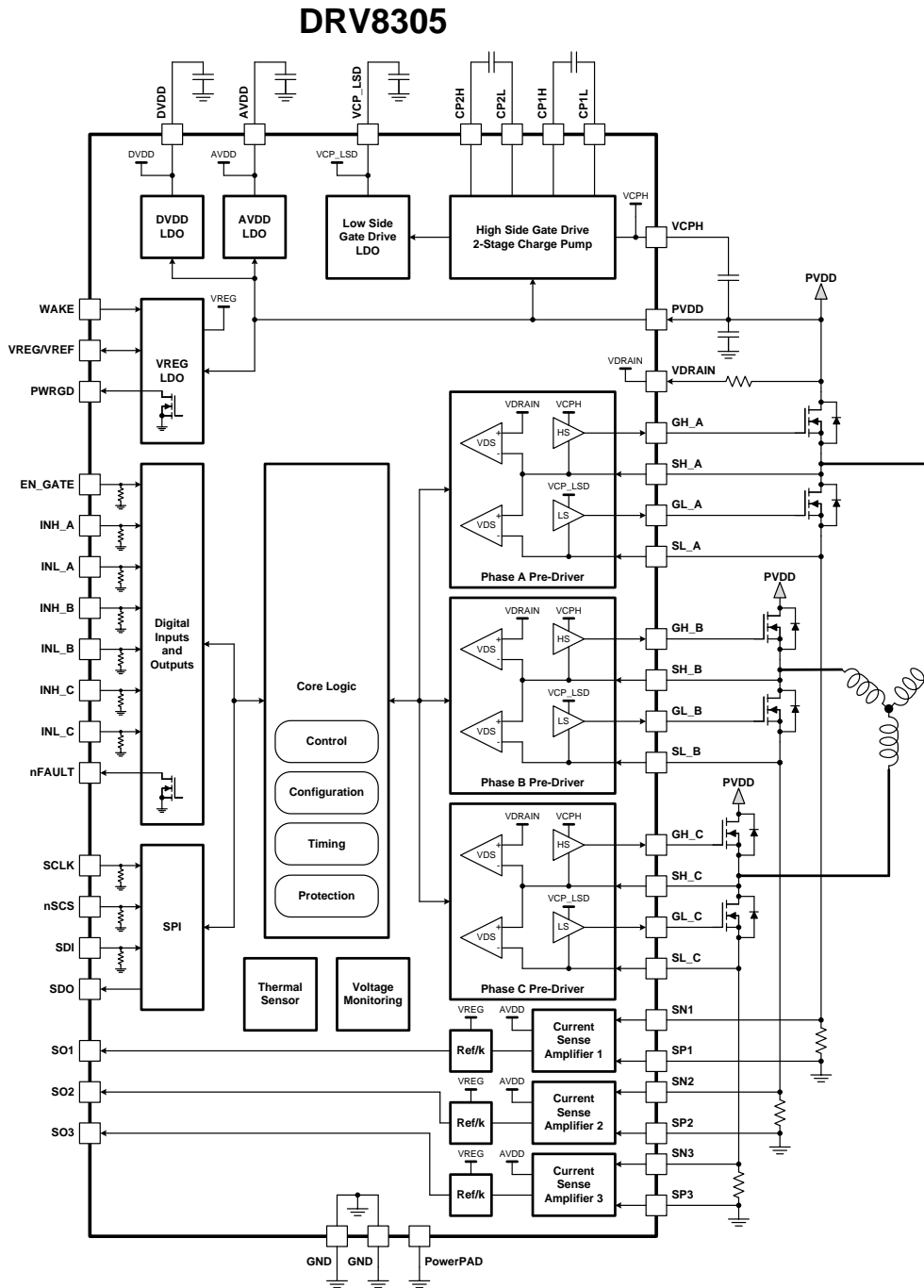
2.1.1 DRV8305

DRV8305 是一款适用于三相电机驱动应用的栅极驱动器 IC。该器件提供三个经高精度调节和温度补偿的半桥驱动器。每个驱动器能够驱动一个高侧和低侧 N 沟道功率 MOSFET。电荷泵驱动器支持占空比为 100% 的低压操作。该栅极驱动器还能够处理最高达 45V 的负载突降脉冲。

栅极驱动器在切换时使用自动握手，以防止电流击穿外部功率 MOSFET。高侧和低侧 MOSFET 使用精确 VDS 感测检测过电流情况。

DRV8305 具有三个双向分流放大器，其增益水平消隐时间可针对精确低侧电流测量进行调节。串行外设接口 (SPI) 和器件寄存器可提供详细错误报告和灵活参数设置，例如分流放大器配置、栅极驱动器转换率控制以及多种保护功能。

栅极驱动器框图如下所示。



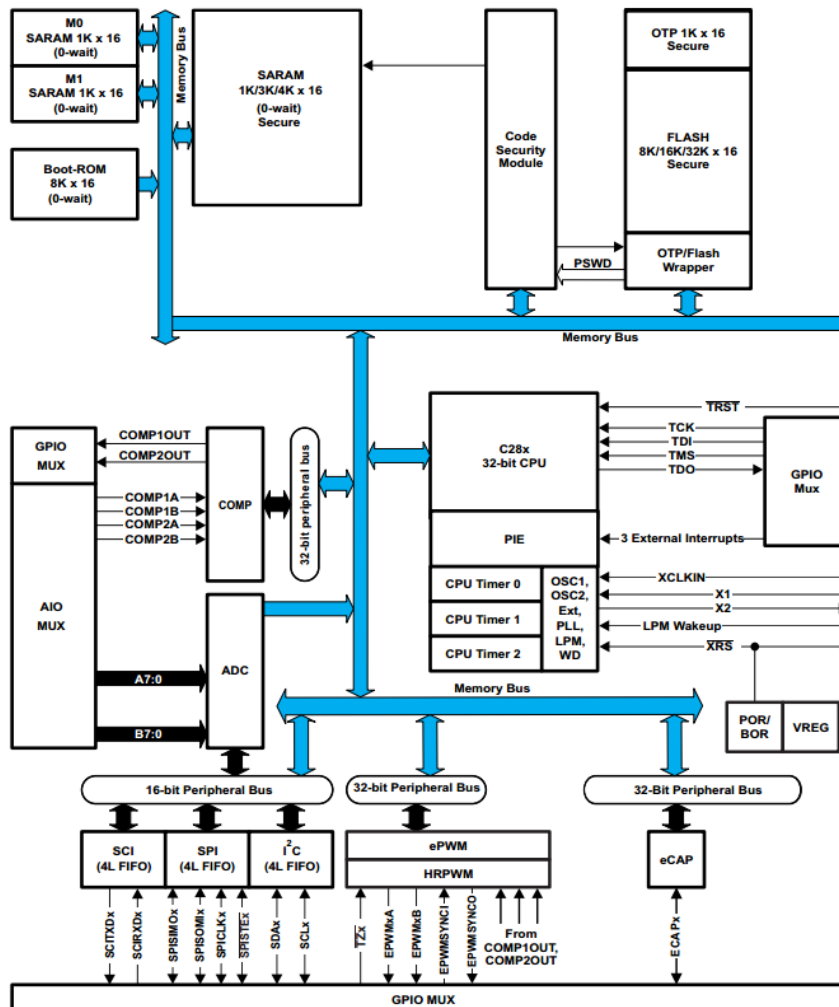
□ 3 : DRV8305 框□

2.1.2 TMS320F28027F

F2802x Piccolo 微控制器系列提供 C28xx 内核的强大功能，同时结合低引脚数器件中高度集成的控制外设。该系列兼容基于 C28x 的原有代码，同时具有较高的模拟集成度。

一个内部稳压器实现了单电源轨运作。对 HRPWM 模块实施了改进，以提供双边缘控制（调频）。增设了具有 10 位内部基准的模拟比较器，可直接进行路由以控制 PWM 输出。ADC 可在 0V 至 3.3V 的固定满量程范围内实施转换，支持 V_{REFH}/V_{REFL} 基准的比例运算。ADC 接口已针对低开销和延迟进行了优化。

这些器件在执行仅 ROM 操作时使用特殊电机控制软件，从而启用 [InstaSPIN-FOC](#) 或 [InstaSPIN-MOTION](#) 解决方案，同时由 [MotorWare](#) 提供系统软件支持。虽然标准的 C2000 [controlSUITE](#) 软件可与这些器件配套使用，但请注意这个特殊只读存储器（ROM）替换了标准 ROM。这意味着 controlSUITE 项目需要在标准 ROM 中实现的某些软件功能需要链接至项目中。更多相关信息，请参见 [《InstaSPIN 用户指南》](#) 的“InstaSPIN-FOC 和 InstaSPIN-MOTION 存储器注意事项”部分或产品数据表的“存储器”部分。



A. Not all peripheral pins are available at the same time due to multiplexing.

□ 4 : TMS430F28027F 框□

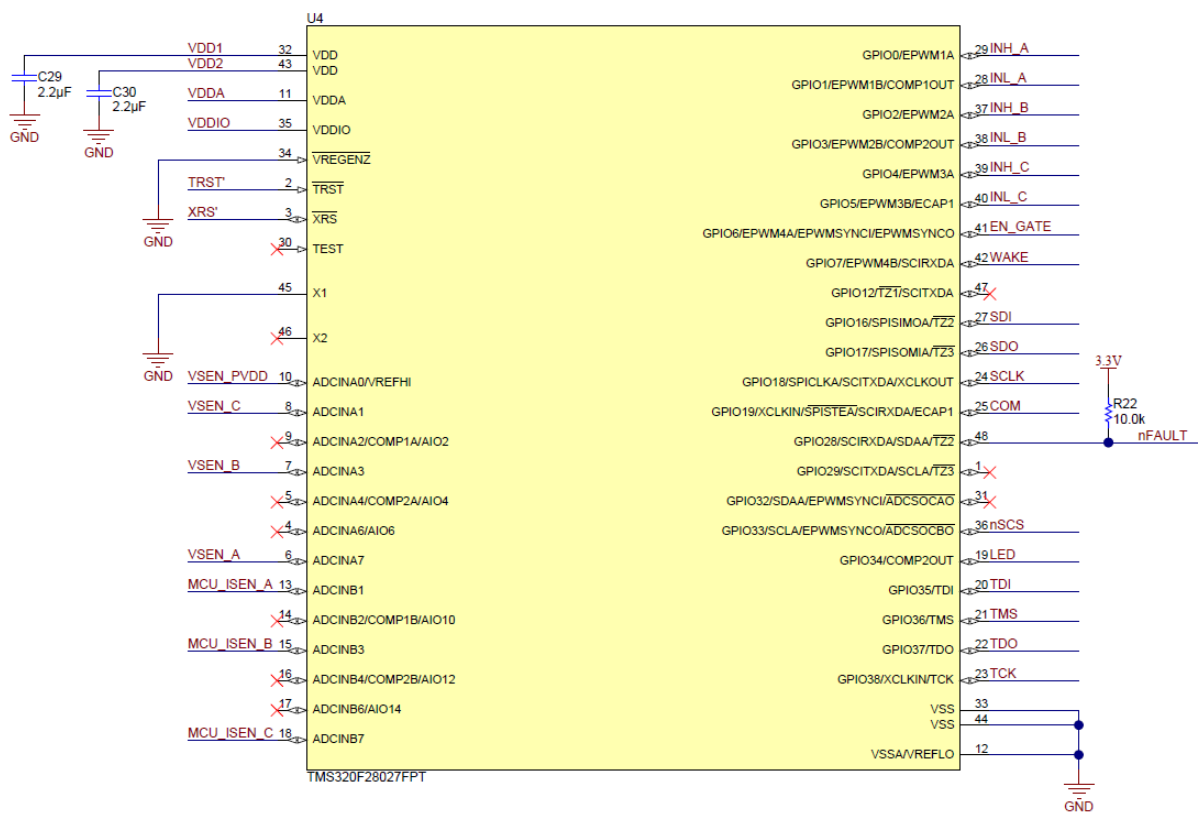
3 系□□□原理

本 4.4V 至 30V、15A 高性能直流无刷螺旋桨控制器参考设计演示了使用 DRV8305 电机栅极驱动器和 TMS320F28027F 微控制器的高性能三相 BLDC 电机控制器。

电机控制器接收馈入 TMS320F28027F MCU ECAP 模块的外部基准电压信号传输的转速命令。MCU 使用电机传输的转速基准和反馈信号，确定发送至由 DRV8305 和 CSD17573Q5B 所组成功率级的信号。

第 4.1 节和第 4.2 节分别介绍了硬件设计原理和参考设计的软件原理。

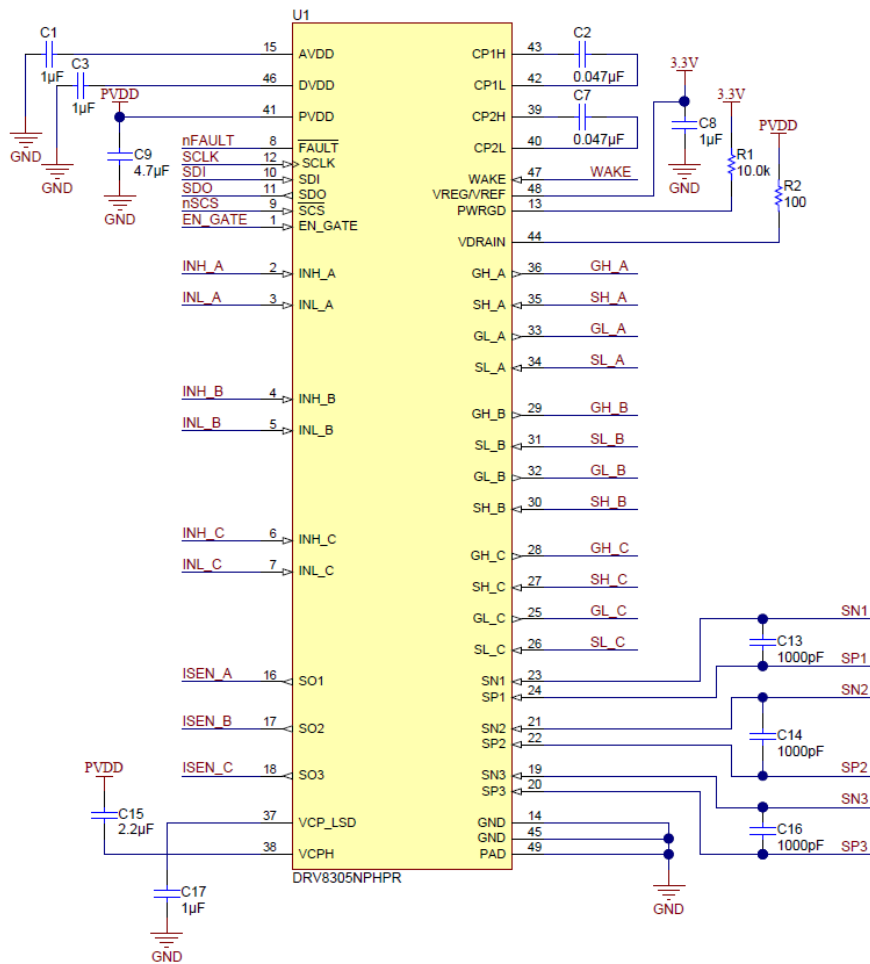
3.1 硬件系□□□原理



□ 5：TMS320F28027F 模□

TMS320F28027F 已配置为使用最少的外部组件并通过内部振荡器生成 60MHz 时钟。使用外部降压稳压器作为 3.3V 电源。

EPWM1、EPWM2 和 EPWM3 增强型 PWM 模块用于为功率级开关生成脉宽信号。使用多条 ADC 通道读取电机反电动势电流和相电流。使用 JTAG 连接器对存储器编程。与 DRV8305 通信的多个通用输入/输出 (GPIO) 和一个 SPI 接口用于设置栅极驱动器模式、相关配置并回读状态信息。

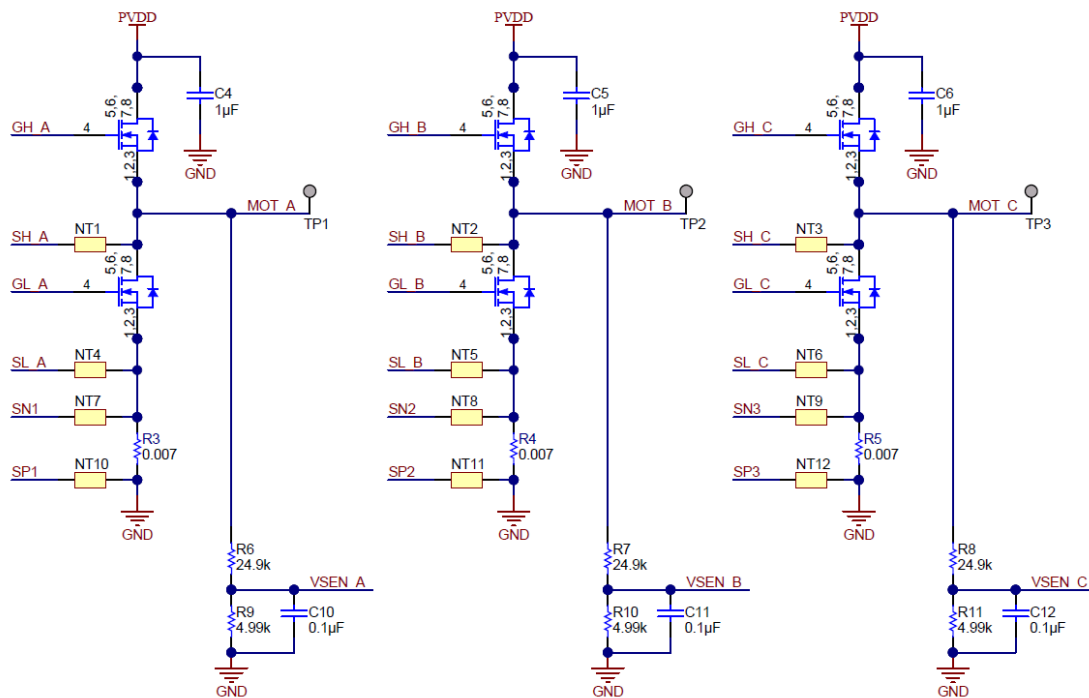


□ 6 : DRV8305 模□

DRV8305 是一款高度集成的三相栅极驱动器，具备专为电机驱动而设计的功能。该器件具备一个通过 4.7 μ F 陶瓷电容进行旁路的单电源 (PVDD)。两个内部稳压器 (AVDD 和 DVDD) 通过 1 μ F 陶瓷电容进行外部旁路。使用三倍频器电荷泵 (VCPH、CP1 和 CP2) 为高侧 N 沟道功率 MOSFET 提供电压输入 (PVDD + 10V)，而线性稳压器 (VCP_LSD) 为低侧 MOSFET 提供 10V 电源。

内部分流放大器以 VREG/VREF 引脚 (由 MCU 3.3V 电源供电) 为基准并且这些设置可通过 SPI 接口进行配置。INH_X 和 INL_X 引脚控制半桥驱动器的状态，同时 GH_X 驱动高侧 MOSFET 栅极，GL_X 驱动低侧 MOSFET 栅极。栅极驱动器放大由 MCU 至电池电源电压 (PVDD) 的逻辑电平输入。VDRAIN 专用引脚用于感测高侧 MOSFET 的漏极电压以便与 V_{DS} 过流监视器搭配使用。WAKE 引脚用于将器件由低功耗休眠模式唤醒。

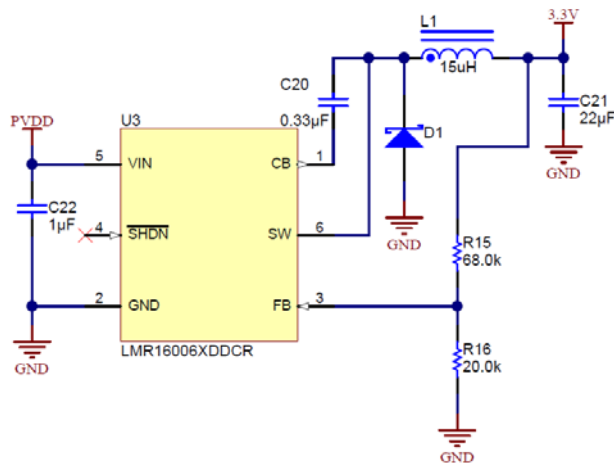
DRV8305 提供管理 MOSFET 转换率和开关性能的配置以及自动切换握手保护，过流保护、欠压保护和过热保护等保护功能。



□ 7：逆口器模口

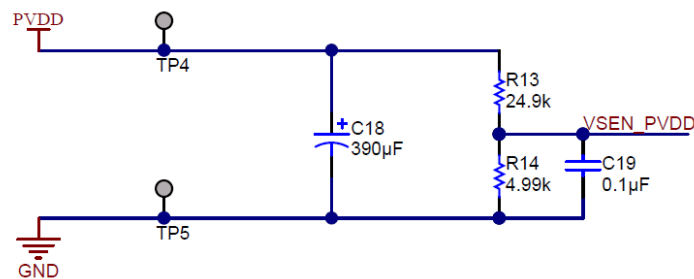
6 个 CSD17573Q5B N 沟道功率 MOSFET 构成直流无刷电机的逆变器。逆变器允许控制器管理电机绕组两端的电压（与电流成正比）。应用正确的电压/电流模式可使电机旋转。为了提高设计效率，MOSFET 由 DRV8305 栅极驱动器驱动。这可以最大限度降低导通和开关损耗。要了解关于电机栅极驱动器和 MOSFET 的更多信息，请参见本应用手册。

除 MOSFET 外，为了测量电机反电动势，每个电机相位均放置了一个分压器。分压器电压直接馈入 MCU ADC。反电动势是电机转动时产生的电压。该电压有助于确定转子位置。为了确定流经每个桥臂的电流，每个半桥的低侧放置了三个分流电阻。分流电阻电压馈入 DRV8305 分流放大器，然后馈入 MCU ADC。这也有助于确定转子位置。



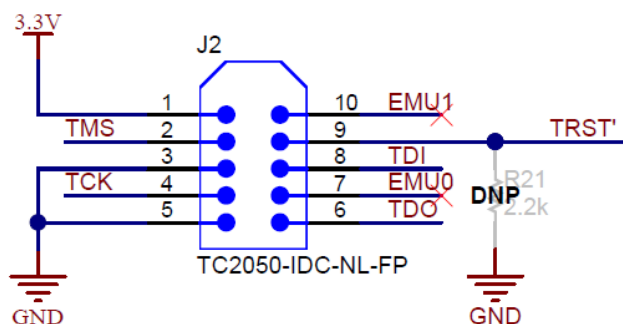
□ 8：3.3V 模□

LMR16006 用作 3.3V 电源的基准。这款经稳压的降压转换器接入电池输入电压并将其降至 MCU 所需的 3.3V。输入电源由 1µF 陶瓷电容旁路，输出则通过一个 15µH 功率电感和一个 22µF 陶瓷电容进行滤波。反馈环路可将转换器输出稳定为 3.3V。



□ 9：□源□入模□

电池电源直接与电路板相连并通过 390µF 大容量电解电容进行滤波。分压器支持 MCU 直接感测电源电压。



□ 10：JTAG 模□

可通过 JTAG XDS100v2 模拟器进行编程和调试。使用紧凑型 Tag-Connect 适配器最大限度缩小连接器的电路板尺寸。

3.2 电机系统原理

电机控制器软件基于 InstaSPIN-FOC 技术和 MotorWare 基础架构。最终系统基于来自 MotorWare 项目的 Lab 5b。Lab 5b 将 InstaSPIN-FOC 和 PI 转速控制器搭配使用。

Lab 5b 使用两个级联比例积分 (PI) 控制器，一个作为电流控制器，另一个作为转速控制器。在该系统中，转速控制器环绕电流控制器。PI 控制器是应用于众多系统的通用控制回路反馈机制。PI 控制器计算一个误差值，作为变量测量值和理想设定值间的差值。然后尝试通过特定响应系统校正该误差。

TIDA-00463 参考设计中的电机采用直流无刷螺旋桨电机，负载惯性来自于螺旋桨。在 TMS320F28027F 上将整个控制系统实例化，功率级则用作控制器与电机之间的接口。

下面的框图显示了整个系统，包含 FAST 观测器、磁场定向控制 (FOC) 算法、ADC 反馈、空间矢量调制 (SVM) 以及 PI 控制器。有关 FOC 控制的更多信息，请观看[此视频](#)或阅读[应用报告](#)。

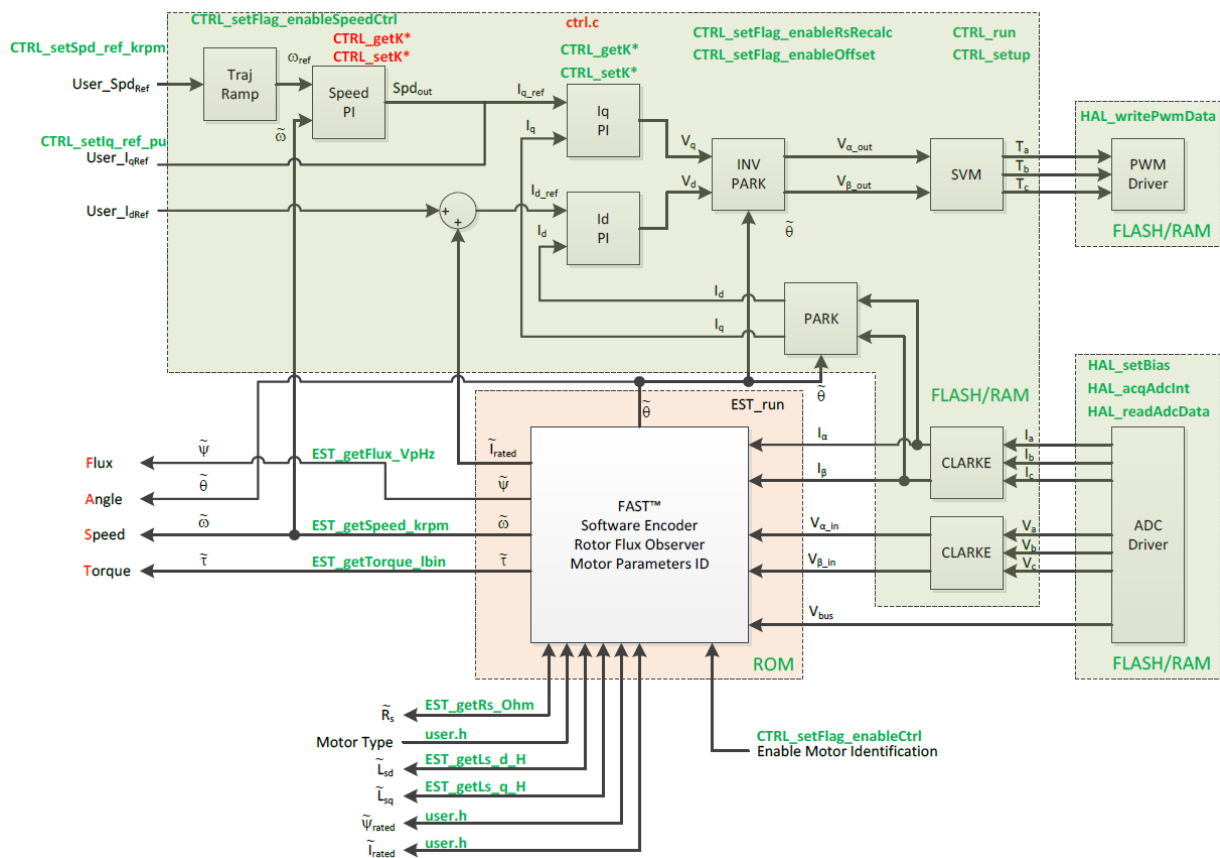
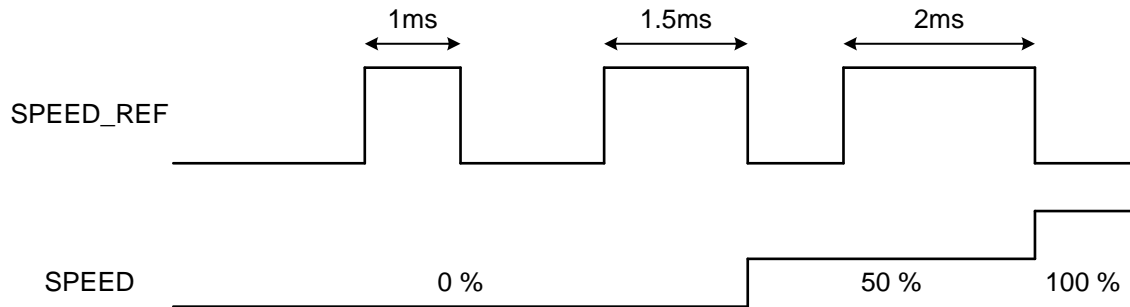


图 11: 电机系统框图

为从中央控制器获得速度参考并将其发送到 PI 控制器中，已将 ECAP 模块配置为接收参考信号。参考信号通过伺服控制方法设置速度命令。它可接受具有 1ms 高周期（对应 0% 速度）和 2ms 周期（对应 100% 速度）的 50Hz 到 500Hz 的频率。两者之间的周期以线性转速曲线运行。



□ 12：速度参考信号

4 硬件入口

4.1 接口

TIDA-00643 参考设计可由兼容电池或 4.4V 到 30V 的电源供电。该电源通过标有 VBAT 和 GND 的两个焊盘连接到电机控制器。可通过固件为大多数直流无刷电机配置电机控制器，但仅面向通常用于无人机和 RC 应用的大功率螺旋桨电机。这三个电机相位可连接到标有 MOTA、MOTB 和 MOTC 的三个焊盘上。

来自中央控制器的速度参考信号通过一个 3 引脚插头与电机控制器相连。此插头标有 VDD（可选 3.3V 电源）、COM（速度参考信号）和 GND。

XDS100v2 JTAG 连接可对 TMS320F28027F 电机控制器进行编程和调试。使用 Tag-Connect 适配器最大限度减小 JTAG 连接器所需的电路板空间。

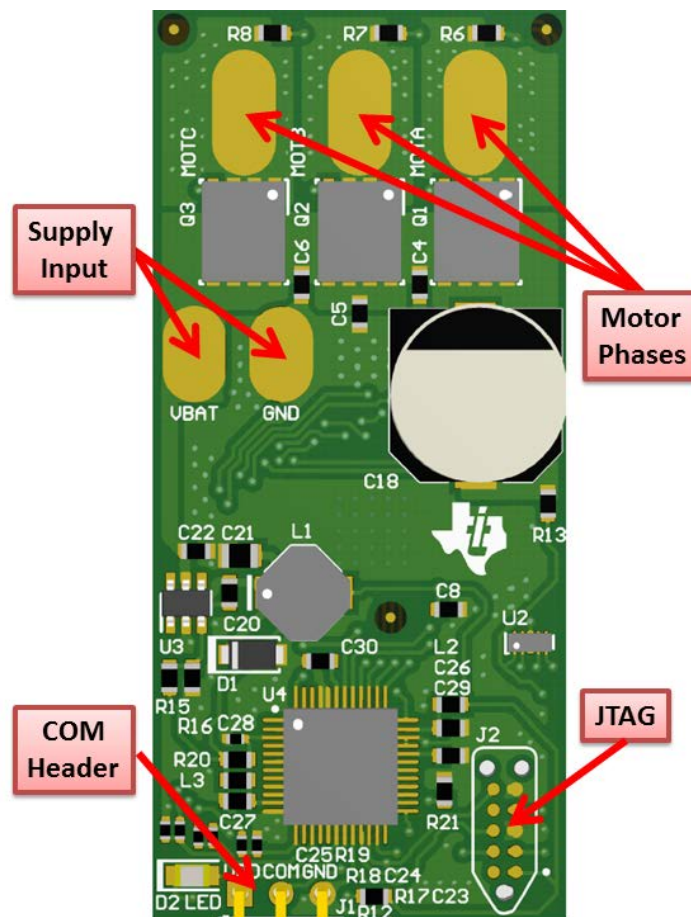


图 13：接口

XDS100v2 JTAG 调试器可从 ti.com 获得，链接为：

- <http://www.ti.com/tool/TMDSEMU100V2U-14T>

Tag-Connect 适配器和电缆可在其[网站](#)上找到，部件号如下。

- TC2050-IDC-NL: “10 引脚，无腿，Tag-Connect 电缆，配有带状连接器”
- TC-C2000-M: “用于 TI C2000 的公头适配器”
- TC2050-CLIP: “与 TC2050-NL 系列搭配使用的固定夹板”

电机控制机通过 MotorWare 调节过程支持多种无刷电机。针对本设计，采用 DJI E300 920 KV 无刷电机。

4.2 步□

请参见以下步骤开始使用参考设计硬件。

1. 通过 VBAT 和 GND 焊盘将电源或电池连接到本设计。
2. 通过 MOTA、MOTB 和 MOC 焊盘将电机相位线连接到本设计。
3. 连接 JTAG 调试器、启用 VBAT 供电并对板载 MCU 编程。如果想要通过 JTAG 连接本设计，此调试器可保持连接。
4. 移除调试器并通过 COM 插头发送适当的控制信号。

5 固件入口

TIDA-00643 为无刷电机控制和系统控制器使用 [MotorWare](#)。要开始使用，请访问 ti.com 并下载最新版本。MotorWare 通过一系列实验为正在使用的特定无刷电机正确识别和调整系统。本文档使用 DJI E300 920 KV 电机演示了此过程，但也可将硬件性能范围内的几乎所有无刷电机重复此步骤。

5.1 向 MotorWare 添加定制硬件

需要将 TIDA-00643 定制硬件添加到 MotorWare 中以开始工作。为正确完成此过程，应先完成几个步骤。

5.1.1 新建的 HAL 目录

首先，需要创建一个新的硬件抽象层（HAL）。

HAL 目录位于：

- MW_INSTALL_DIR\sw\modules\hal

建议根据与定制硬件类似的现有设计创建新的 HAL 目录。本例中将使用 boostxldr8305_revA 硬件。它使用相同的驱动器（DRV8305）和控制器（TMS320F28028F）。创建一个现有设计的副本并重命名为定制硬件。

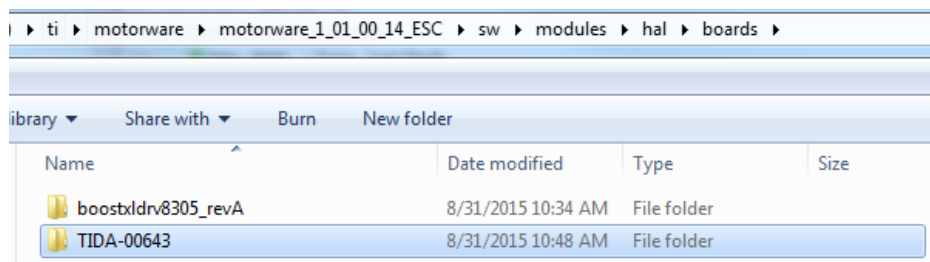


图 14：定制 HAL 目录

本设计中对 HAL 的主要修改为，DRV8305 的 SCS 引脚已移至 GPIO33，并为 GPIO34 添加了一个 LED。

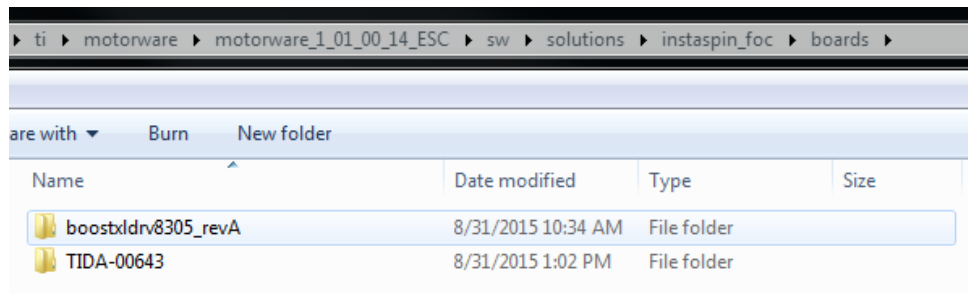
5.1.2 新建的 PROJECT 目录

为了进行一系列独立的实验，需要创建一个新的 PROJECT 目录。

PROJECT 目录位于：

- MW_INSTALL_DIR\sw\solutions\instaspin_foc\boards

建议根据与定制硬件类似的现有设计创建新的 PROJECT 目录。本设计与 boostxldr8305_revA 硬件极为匹配。创建一个现有设计的副本并重命名为定制硬件。

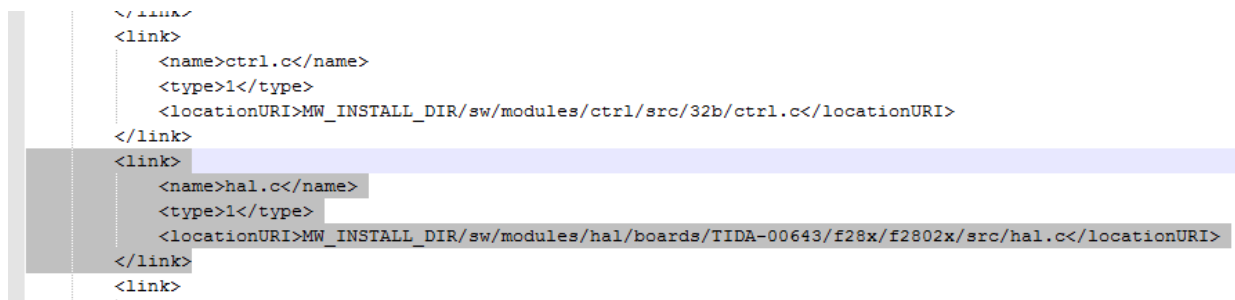


□ 15：定制 PROJECT 目口

为了指向新的 HAL，创建新的 PROJECT 目录后，需要为每个实验项目更新 HAL 链接。

此链接可在每个实验项目下的 .project 文件里进行修改，此文件位于：

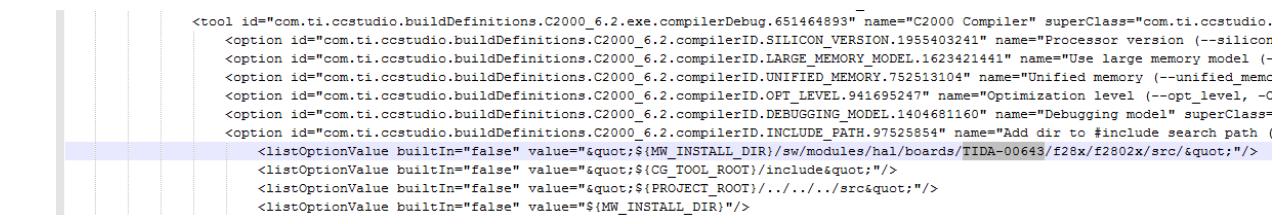
- MW_INSTALL_DIR\sw\solutions\instaspin_foc\boards\TIDA-00643\28x\2802x\projects\ccs5\proj_labXX



□ 16：.project HAL 口接

HAL 目录的包含链接也需要更新，它位于 .cproject 文件中：

- MW_INSTALL_DIR\sw\solutions\instaspin_foc\boards\TIDA-00643\28x\2802x\projects\ccs5\proj_labXX



□ 17：.cproject INCLUDE 口接

此时，您应该能将项目导入 Code Composer Studio 并成功编译。

5.1.3 更新 User.h 文件

验证新项目已在 Code Composer Studio 中成功编译后，下一步是更新具有定制硬件参数的 MotorWare user.h。要更新的参数如下所示。

#define USER_IQ_FULL_SCALE_VOLTAGE_V	(20.0)	// TIDA-00643
#define USER_ADC_FULL_SCALE_VOLTAGE_V	(19.73)	// TIDA-00643


```
#define USER_IQ_FULL_SCALE_CURRENT_A      (23.57) // TIDA-00643
```

```
#define USER_ADC_FULL_SCALE_CURRENT_A     (47.14) // TIDA-00643
```

```
#define USER_VOLTAGE_FILTER_POLE_Hz      (382.64) // TIDA-00643
```

5.2 MotorWare □□

MotorWare 是一套全面的软件和技术资源，旨在最大限度地缩短电机控制系统开发时间。它使用一系列简单的分步实验项目，演示如何建立一个电机控制系统。此项目使用 Lab02c、Lab3a、Lab3b、Lab4、Lab5a 和 Lab5b。接下来将简要介绍每个实验以及它们在 TIDA-00643 无刷电机控制器中发挥的作用。

5.2.1 Lab2c：用于低电感 PMSM 的电机 ID

Lab2 演示了将用于此系统的特定直流无刷电机的参数识别。对于 TIDA-00643 电机控制器，这将为 DJI E300 920 KV 直流无刷电机。由于高速无人机螺旋桨电机的设计，与典型直流无刷电机相比，其绕组电感通常很低。Lab2c 使用一些附加功能来适应这些电机类型。

第一步是在 User.h 文件中创建电机配置文件。DJI_E300 电机具有 7 对电极。为了适应更低的绕组电感，已将 USER_MOTOR_FLUX_EST_PREQ_Hz 增加至 100.0。

```
#define DJI_E300                            119
```

```
#define USER_MOTOR DJI_E300
```

```
#elif (USER_MOTOR == DJI_E300)
#define USER_MOTOR_TYPE                MOTOR_Type_Pm
#define USER_MOTOR_NUM_POLE_PAIRS      (7)
#define USER_MOTOR_Rr                   (NULL)
#define USER_MOTOR_Rs                   (0.08385667)
#define USER_MOTOR_Ls_d                 (1.002232e-05)
#define USER_MOTOR_Ls_q                 (1.002232e-05)
#define USER_MOTOR_RATED_FLUX           (0.005794205)
#define USER_MOTOR_MAGNETIZING_CURRENT (NULL)
#define USER_MOTOR_RES_EST_CURRENT      (1.5)
#define USER_MOTOR_IND_EST_CURRENT      (-1.5)
#define USER_MOTOR_MAX_CURRENT          (15.0)
#define USER_MOTOR_FLUX_EST_FREQ_Hz    (100.0)
```

5.2.2 Lab3a：使用无偏移电机参数

运行 lab2c 后，您应该已经获得电机参数。可使用这些参数更新 user.h 文件以记住此电机。Lab3a 显示了如何在基本控制环路中使用 Lab2c 中的参数运行电机。

在我们的设置中，电机的运行状态实际上很差。这与控制环路使用的默认参数有关。下个实验中将对对此进行优化。

5.2.3 Lab3b：使用有偏移电机参数

Lab3b 显示了如何为使用中的特定硬件保存偏移然后使用这些偏移运行电机。TIDA-00643 电机控制器的偏移如下所示。每个电路板的电压和电流偏移各不相同，因此应针对每套硬件更新最佳结果。

```
//! \brief ADC current offsets for A, B, and C phases
#define I_A_offset (1.019250691) // TIDA-00643
#define I_B_offset (1.022722363) // TIDA-00643
#define I_C_offset (1.021086812) // TIDA-00643

//! \brief ADC voltage offsets for A, B, and C phases
#define V_A_offset (0.2965255976) // TIDA-00643
#define V_B_offset (0.2964715958) // TIDA-00643
#define V_C_offset (0.2963429689) // TIDA-00643
```

5.2.4 Lab4：转矩模式

由于就性质而言，磁场定向控制是一种电流控制方法，Lab4 演示了仅使用一个转矩控制器使电机旋转。

5.2.5 Lab5a：转矩模式和 Id/Iq PI

Lab5a 研究了转矩控制器的调节。由于 InstaSPIN-FOC 使用一个环绕在转矩控制器周围的速度控制器，因此正确调节转矩控制器至关重要。对于此系统，计算的默认值已足够。

5.2.6 Lab5b：速度模式和速度 PI

使用的最后一个实验 Lab5b 研究了速度控制器的调节。本实验也将用于电机控制器控制系统的最终版本。确定正确的 Kp_spd 和 Ki_spd 后，它们可编程为默认加载。

```
// initialize the watch window kp and ki values with pre-calculated values
gMotorVars.Kp_spd = CTRL_getKp(ctrlHandle, CTRL_Type_PID_spd);
gMotorVars.Ki_spd = CTRL_getKi(ctrlHandle, CTRL_Type_PID_spd);

// TIDA-00643 custom values for speed controller
gMotorVars.Kp_spd = _IQ(2.000);
gMotorVars.Ki_spd = _IQ(0.059);
```

由于 InstaSPIN-FOC 是一种正弦波调制技术，应当注意，峰值输出被限制为整个母线电压的 77% 左右。为了实现电机的全速度范围，可在以下 lab10 中使用过调制技术。这不在本设计的范围内。

5.3 eCAP 速度参考接收器

为了使电机控制器接收来自中央控制器的速度参考，需要一个前端接收器。此接收器可使用 TMS320F28027F 的 eCAP 模块实现。eCAP 模块接收来自中央控制器的可变占空比 PWM 并将其转换为 PI 控制器的速度命令。

5.3.1 引脚配置

为了使用 eCAP 模块，必须正确配置引脚。该操作在 hal.c 文件中完成。

```
// ECAP
GPIO_setMode(obj->gpioHandle, GPIO_Number_19, GPIO_19_Mode_ECAPI);
```

还需要更新 SPI SCS 引脚以使用 GPIO 资源。

```
// SPI_SCS
GPIO_setMode(obj->gpioHandle, GPIO_Number_33, GPIO_33_Mode_GeneralPurpose);
GPIO_setDirection(obj->gpioHandle, GPIO_Number_33, GPIO_Direction_Output);
GPIO_setHigh(obj->gpioHandle, GPIO_Number_33);
```

由于 eCAP 正使用来自 C2000 的专用 SPIS CS 资源，因此需要对固件中的现有 SPI 程序稍加修改。这些更改在 drv8305.c 文件中进行。

DRV8305_writeSpi 函数：

```
// Added: SCS High->Low
GPIO_setLow(obj->gpioHandle, GPIO_Number_33);

// write the command
SPI_write(obj->spiHandle, ctrlWord);

// wait for registers to update
for(n=0;n<0xf;n++)
    asm(" NOP");

// Added: SCS Low->High
GPIO_setHigh(obj->gpioHandle, GPIO_Number_33);
```

DRV8305_readSpi 函数:

```
// Added: SCS High->Low
GPIO_setLow(obj->gpioHandle, GPIO_Number_33);

// write the command
SPI_write(obj->spiHandle, ctrlWord);

// wait for the response to populate the RX fifo, else a wait timeout will occur
while((RxFifoCnt < SPI_FifoStatus_1_Word) && (WaitTimeOut < 0xffff))
{
    RxFifoCnt = SPI_getRxFifoStatus(obj->spiHandle);
    if(++WaitTimeOut > 0xffff)
    {
        obj->RxTimeOut = true;
    }
}

// Added: SCS Low->High
GPIO_setHigh(obj->gpioHandle, GPIO_Number_33);
```

5.3.2 eCAP 句柄

在 MotorWare 中，通过硬件抽象层 (HAL) 访问外设。 外设通过其专用句柄访问。 要使用 eCAP，必须也将其添加到 HAL。

第一步是将 eCAP 库添加到 MotorWare 项目中。 可将库的链接添加到 lab5b 项目下的 .project 文件中，此文件位于：

- MW_INSTALL_DIR/sw/solutions/instaspin_foc/boards/TIDA-00643/f28x/f2802xF/projects/ccs5/proj_lab5b

```
.. -----
<link>
  <name>adc.c</name>
  <type>1</type>
  <locationURI>MW_INSTALL_DIR/sw/drivers/adc/src/32b/f28x/f2802x/adc.c</locationURI>
</link>
<link>
  <name>cap.c</name>
  <type>1</type>
  <locationURI>MW_INSTALL_DIR/sw/drivers/cap/src/32b/f28x/f2802x/cap.c</locationURI>
</link>
<link>
```

□ 18 : eCAP □□接

将 cap 库添加到 hal_obj.h 文件。

```
#include "sw/drivers/adc/src/32b/f28x/f2802x/adc.h"  
#include "sw/drivers/cap/src/32b/f28x/f2802x/cap.h"  
#include "sw/drivers/clk/src/32b/f28x/f2802x/clk.h"
```

将 cap 句柄添加到 hal_obj.h 文件中的 _HAL_Obj_。

```
typedef struct _HAL_Obj_  
{  
    ADC_Handle    adcHandle;        //!< the ADC handle  
  
    CAP_Handle    capHandle;        //!< the CAP handle
```

使用其他外设句柄实例化此句柄。

```
// initialize the ADC  
obj->adcHandle = ADC_init((void *)ADC_BASE_ADDR, sizeof(ADC_Obj));  
  
// initialize the eCAP  
obj->capHandle = CAP_init((void *)CAP1_BASE_ADDR, sizeof(CAP_Obj));  
  
// initialize the clock handle  
obj->clkHandle = CLK_init((void *)CLK_BASE_ADDR, sizeof(CLK_Obj));
```

5.3.3 eCAP ISR

创建 eCAP 句柄后，下一步是创建将处理速度参考信号的中断服务程序 (ISR)。首先，为 eCAP ISR 创建一个设置函数。您将向 hal.h 文件添加声明。

```
//! \brief Setup the ECAP
//! \param[in] handle      The hardware abstraction layer (HAL) handle
void HAL_setupeCAP(HAL_Handle handle);
```

然后将定义添加到 hal.c。eCAP 已被设置为测量上升沿与下降沿之间的时间。

```
// setup the ADCs
HAL_setupAdcs(handle);

// setup the ECAP
HAL_setupeCAP(handle);
```

确保 eCAP 时钟已在 hal.c 中设置

```
void HAL_setupPeripheralClks(HAL_Handle handle)
{
    HAL_Obj *obj = (HAL_Obj *)handle;

    CLK_enableAdcClock(obj->clkHandle);

    CLK_enableCompClock(obj->clkHandle, CLK_CompNumber_1);
    CLK_enableCompClock(obj->clkHandle, CLK_CompNumber_2);
    CLK_enableCompClock(obj->clkHandle, CLK_CompNumber_3);

    CLK_enableEcap1Clock(obj->clkHandle);
```

创建设置函数后，必须在 hal.h 中声明 ECAP ISR。

```
// the globals

extern interrupt void mainISR(void);

// ECAP
extern interrupt void ecapISR(void);
```

随后，将其添加至中断向量表，从而正确寻址并指向。pie 是 HAL 句柄对象内的一个句柄，它是支持所有外设中断的 C2000 外设中断扩展 (PIE) 块。

```
static inline void HAL_initIntVectorTable(HAL_Handle handle)
{
    HAL_Obj *obj = (HAL_Obj *)handle;
    PIE_Obj *pie = (PIE_Obj *)obj->pieHandle;

    ENABLE_PROTECTED_REGISTER_WRITE_MODE;

    pie->ADCINT1 = &mainISR;

    // ECAP
    pie->ECAP1_INT = &ecapISR;

    DISABLE_PROTECTED_REGISTER_WRITE_MODE;
}
```

创建 eCAP ISR 后，最后一步是定义实际的 eCAP ISR。这将添加至包含 main() 函数的 proj_lab05b.c 文件中。eCAP ISR 会清除中断，计算速度参考信号处于高电平的时间段，之后计算适当的电机速度。


```

__interrupt void ecapISR(void)
{
    // Clear capture (CAP) interrupt flags
    CAP_clearInt(halHandle->capHandle, CAP_Int_Type_All);

    // Compute the PWM high period (rising edge to falling edge)
    uint32_t PwmDuty = (uint32_t) CAP_getCap2(halHandle->capHandle) - (uint32_t) CAP_get
    Cap1(halHandle->capHandle);

    // Assign the appropriate speed command, combine 0-5% and 95-100%
    // 0-100% speed is proportional to 1-2ms high period
    // 60MHz * 2ms = 120000

    // 0-1%
    if (PwmDuty <= 61000)
    {
        gSpeedRef_Ok = 0;
        gSpeedRef_duty = _IQ(0);
        gMotorVars.Flag_Run_Identify = 0;
    }
    // 1-99%
    if ((PwmDuty > 61000) && (PwmDuty < 119000))
    {
        gSpeedRef_Ok = 0;
        gSpeedRef_duty = _IQdiv(PwmDuty - 60000, 60000);
        gMotorVars.Flag_Run_Identify = 1;
    }
    // 99-100%
    else if (PwmDuty >= 119000)
    {
        gSpeedRef_Ok = 0;
        gSpeedRef_duty = _IQ(1.0);
        gMotorVars.Flag_Run_Identify = 1;
    }
    // Catch all
    else
    {
        gSpeedRef_duty = _IQ(0);
        gMotorVars.Flag_Run_Identify = 0;
    }

    // Clears an interrupt defined by group number
    PIE_clearInt(halHandle->pieHandle, PIE_GroupNumber_4);
} // end of ecapISR() function

```

最后一步是将设定的电机转速送入电机控制器。此操作在 `updateGlobalVariables_motor()` 函数的主 ISR 中完成。

```
void updateGlobalVariables_motor(CTRL_Handle handle)
{
    CTRL_Obj *obj = (CTRL_Obj *)handle;

    // get the speed estimate
    gMotorVars.Speed_krpm = EST_getSpeed_krpm(obj->estHandle);

    // set motor speed command dependent
    gMotorVars.SpeedRef_krpm = gSpeedRef_duty * SPEED_BASE_KRPM;
```

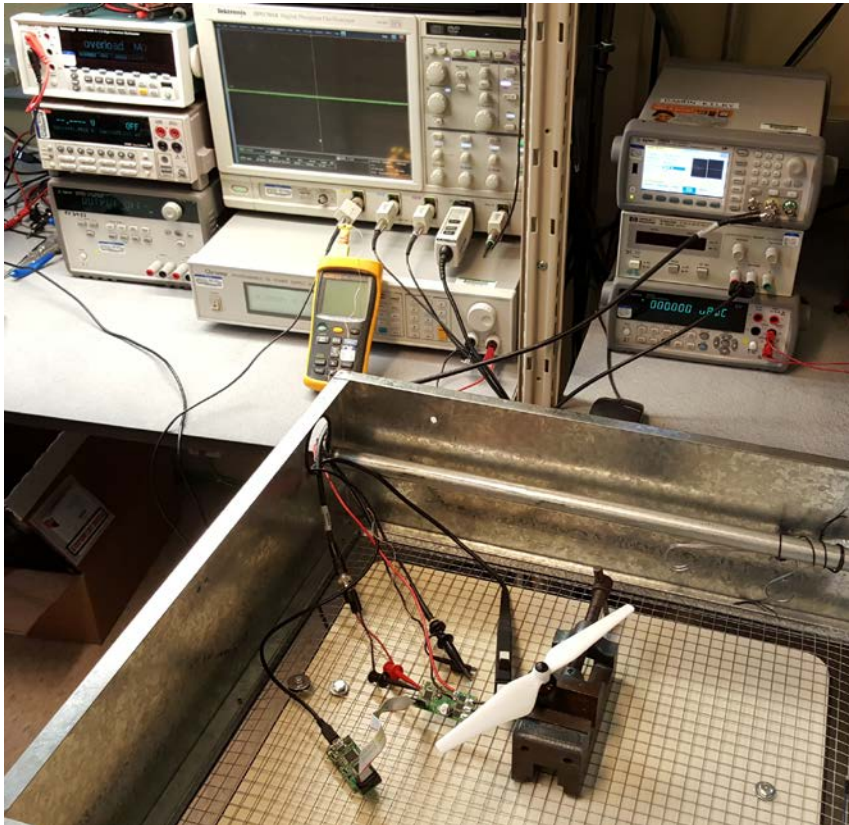
此外，还添加了超时功能，以防速度参考信号丢失后，失去对电机的控制。在用于电机控制的主 ISR 中，已添加速度参考信号的定时器。每次接收到新的速度参考信号时，定时器都会复位。如果定时器超时（信号丢失），则电机禁用。

```
interrupt void mainISR(void)
{
    // toggle status LED
    if(gLEDCnt++ > (uint_least32_t)(USER_ISR_FREQ_Hz / LED_BLINK_FREQ_Hz))
    {
        HAL_toggleLed(halHandle, (GPIO_Number_e)HAL_Gpio_LED2);
        gLEDCnt = 0;
    }

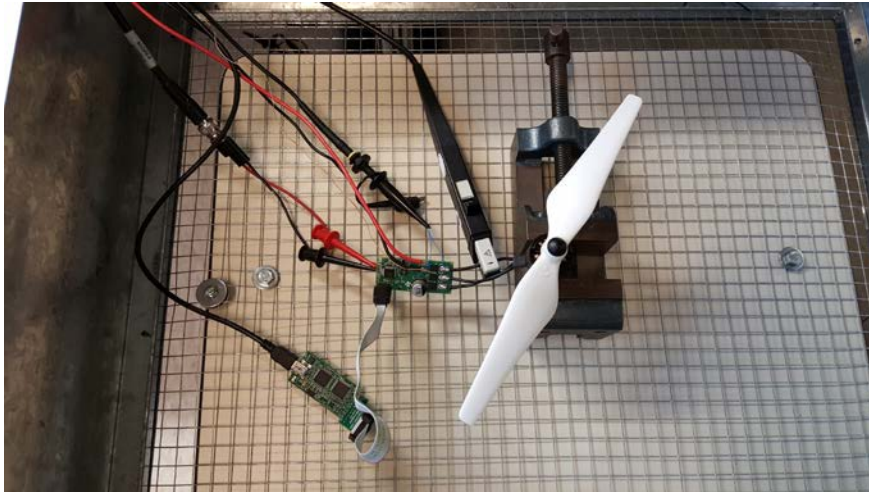
    // Check if speed reference signal is active
    // If more than 2000 service routine cycles pass without signal, disable motor
    if (gSpeedRef_0k++ > 2000)
    {
        gSpeedRef_duty = _IQ(0);
        gMotorVars.Flag_Run_Identify = 0;
        gSpeedRef_0k = 0;
    }
}
```

6 □□装置

设备	名称
直流电源	Chroma 620012P-100-50
万用表	Tektronix DMM 4040
示波器	Tektronix DPO 7054
函数发生器	Agilent 33522A
温度计	Fluke 54



□ 19 : □□装置



□ 20 : □机装置

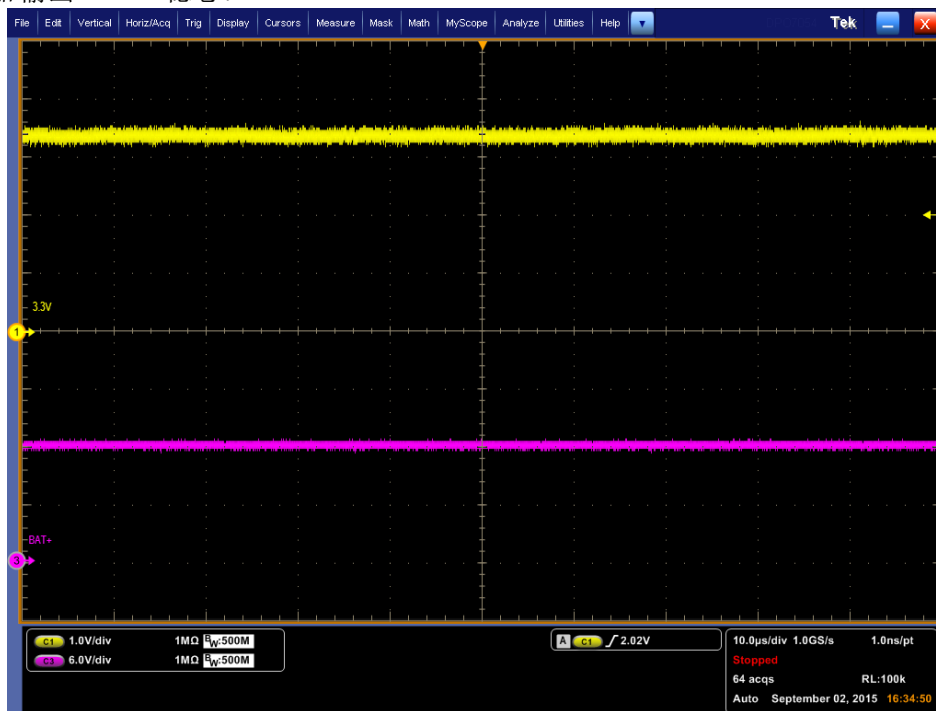
7 □□数据

7.1 功能□□

7.1.1 降□□□器

通过测试检测降压稳压器输入 (VBAT) 与输出 (3.3 V) 之间的关系。

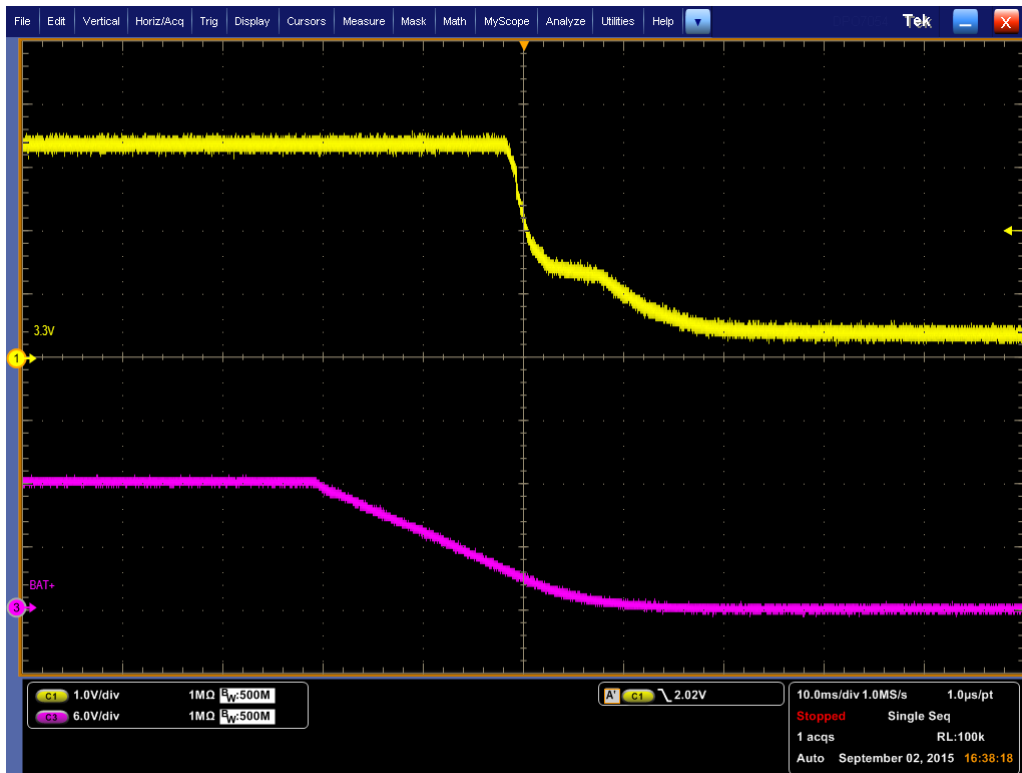
降压稳压器输出 3.3V 稳态:



降压稳压器输出 3.3V 启动:

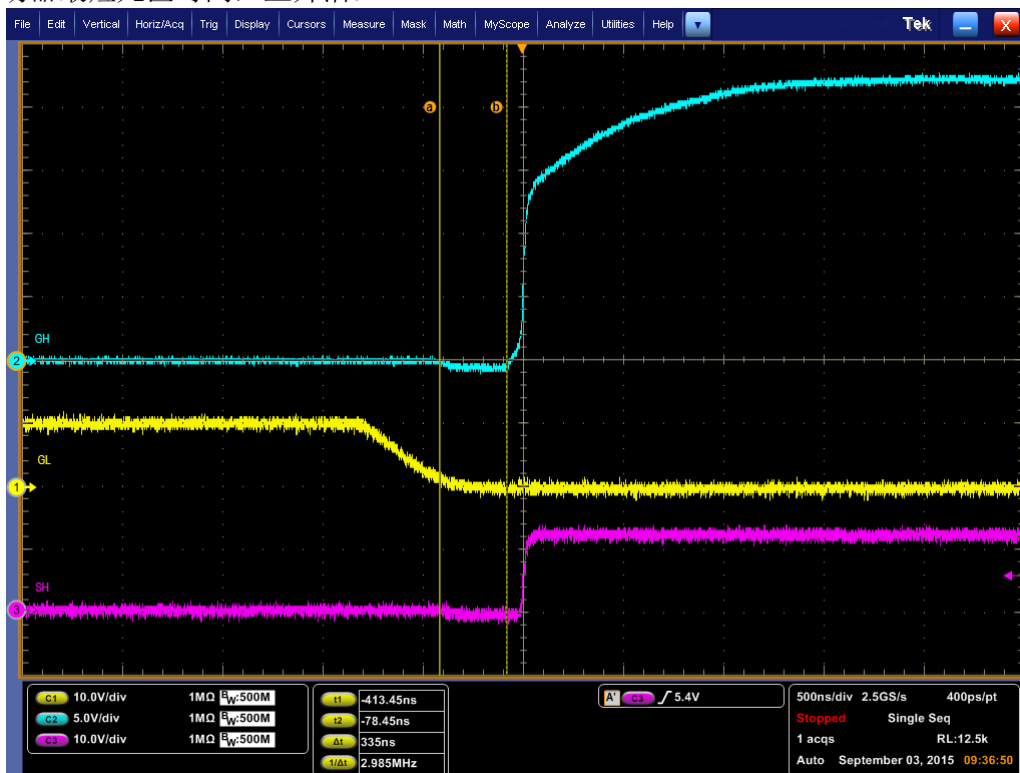


降压稳压器输出 3.3V 关闭:

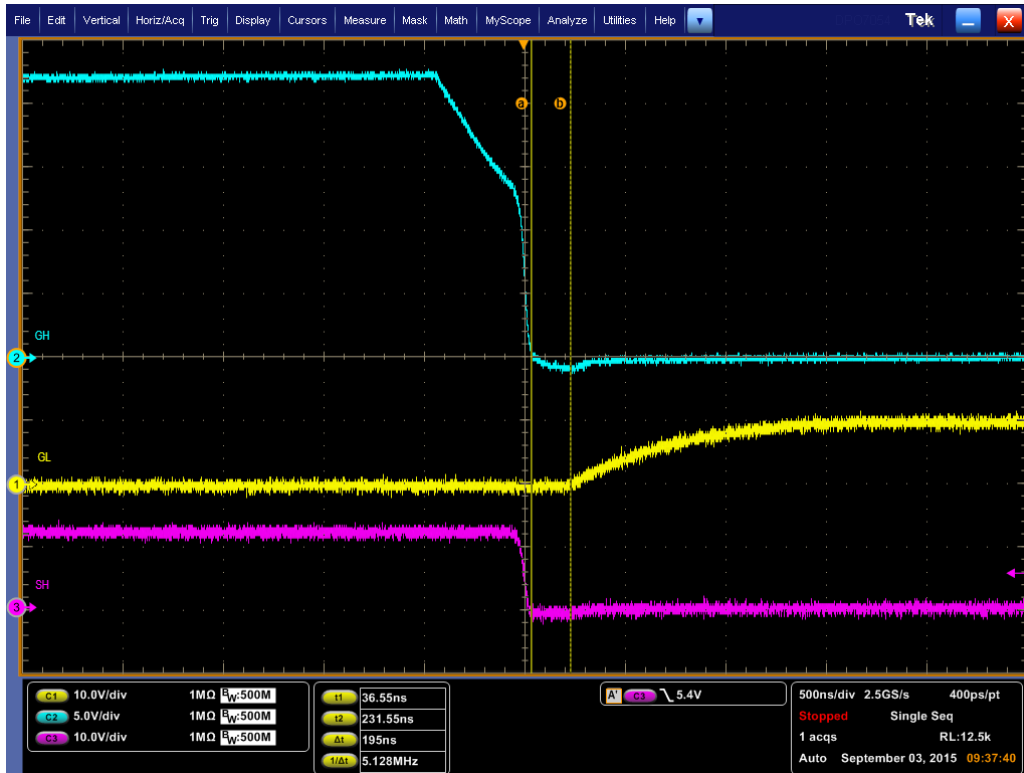


7.1.2 栅极驱动器死区时间

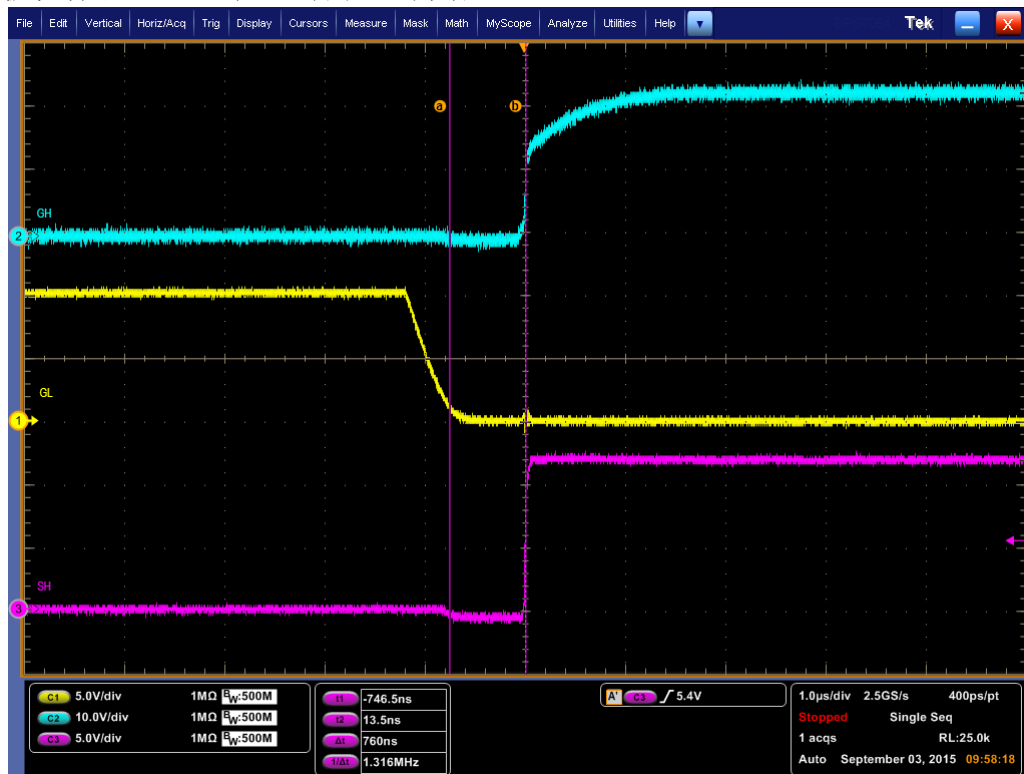
通过测试检测栅极驱动器输出（GL、GH 和 SH）死区时间。死区时间通过 DRV8305 配置。栅极驱动器最短死区时间，上升沿：



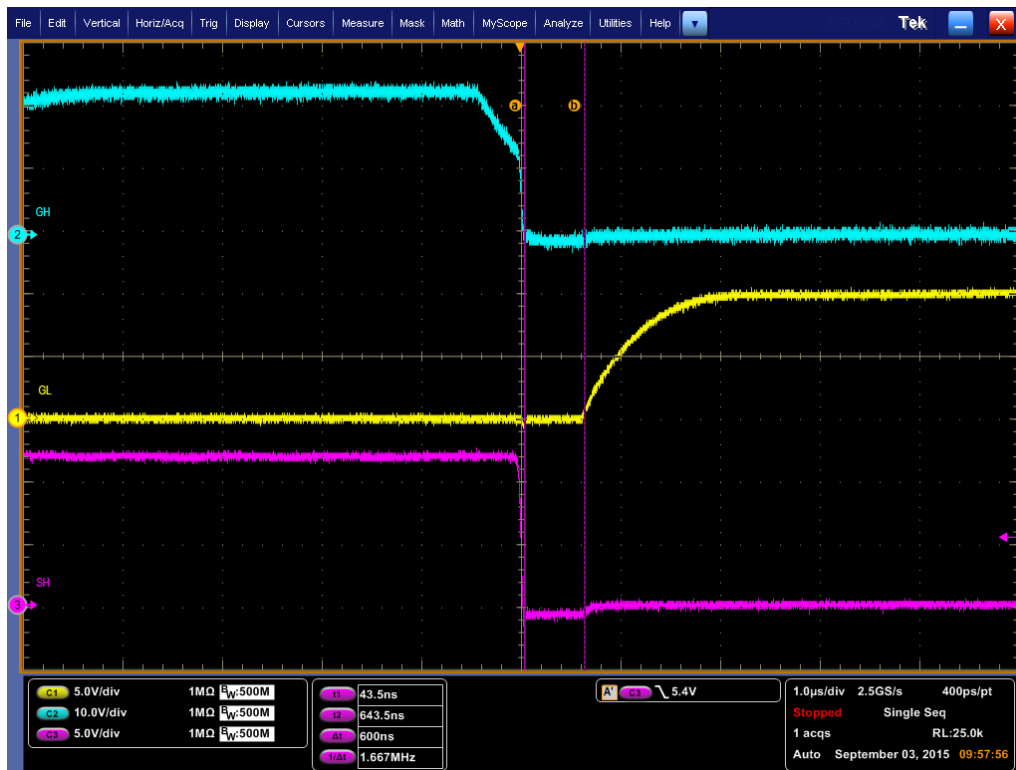
栅极驱动器最短死区时间，下降沿:



插入栅极驱动器 500 ns 死区时间，上升沿:



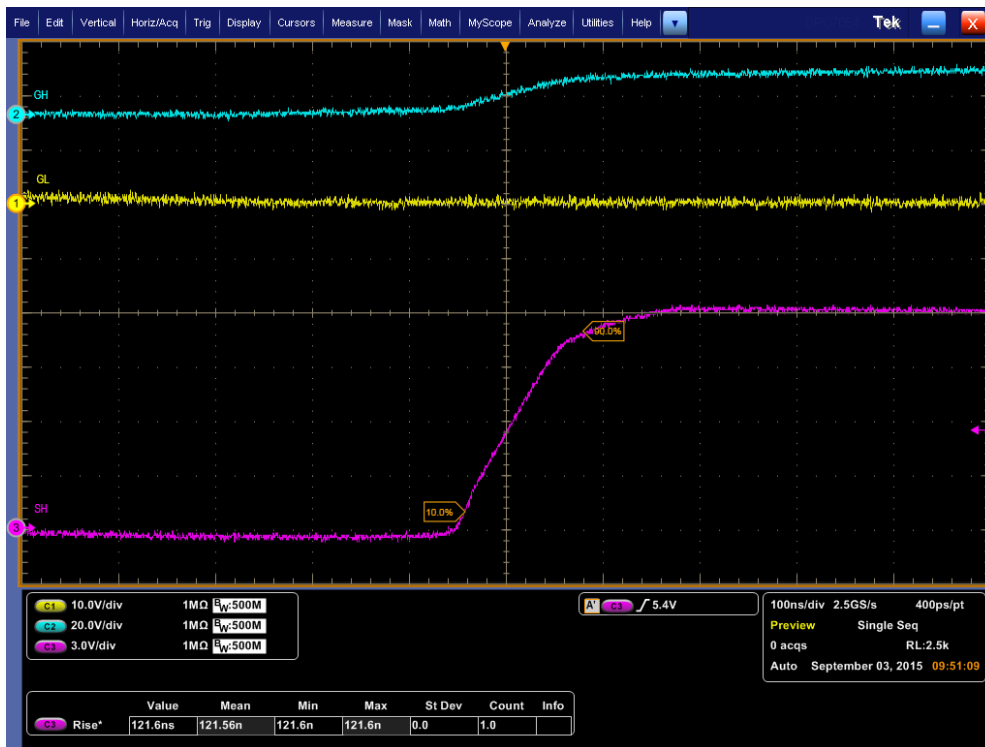
插入栅极驱动器 500 ns 死区时间，下降沿:



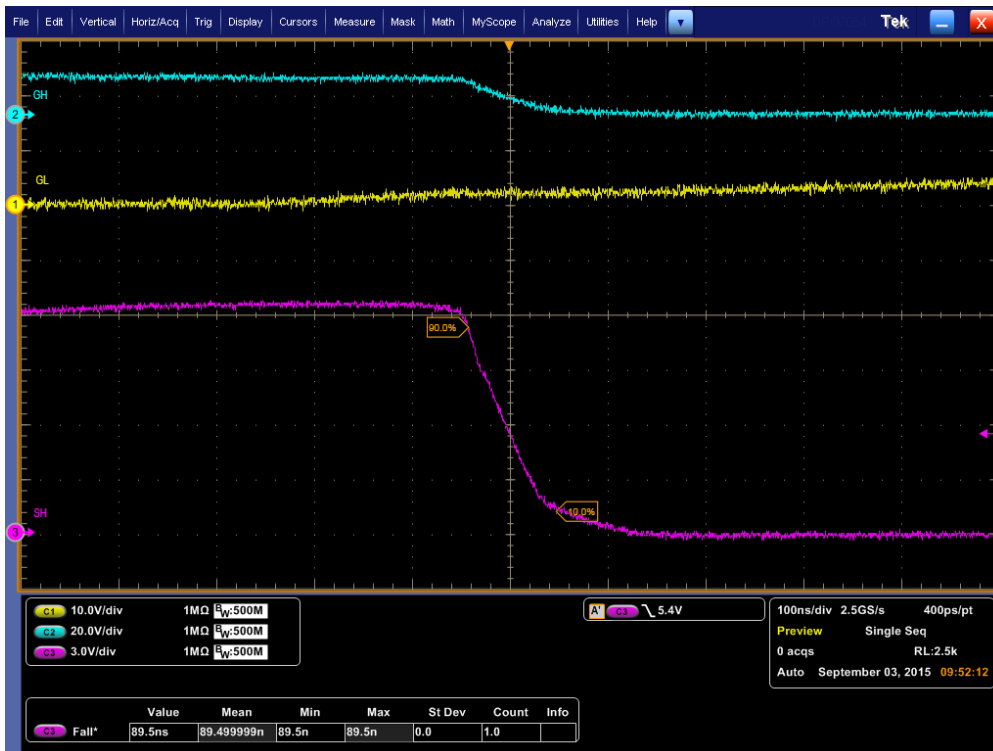
7.1.3 栅极驱动器电流大小

通过测试检测栅极驱动器输出（GL、GH 和 SH）驱动强度。驱动强度通过 DRV8305 配置。

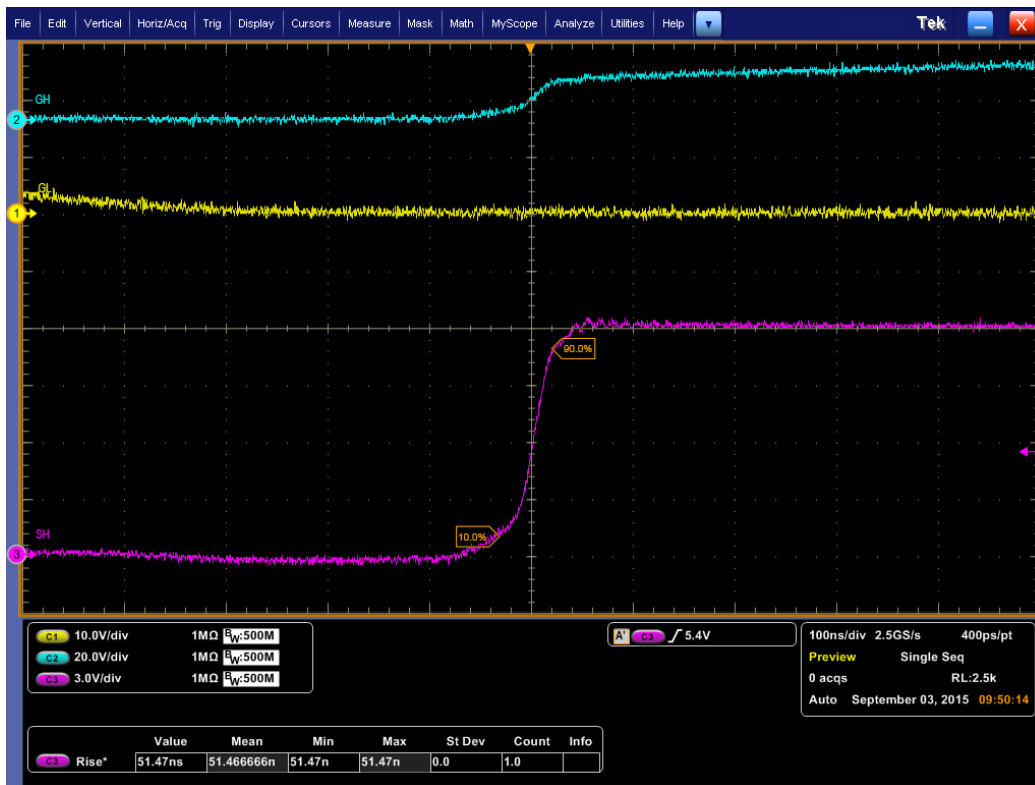
栅极驱动器 50 mA 拉电流，60 mA 灌电流，上升沿：



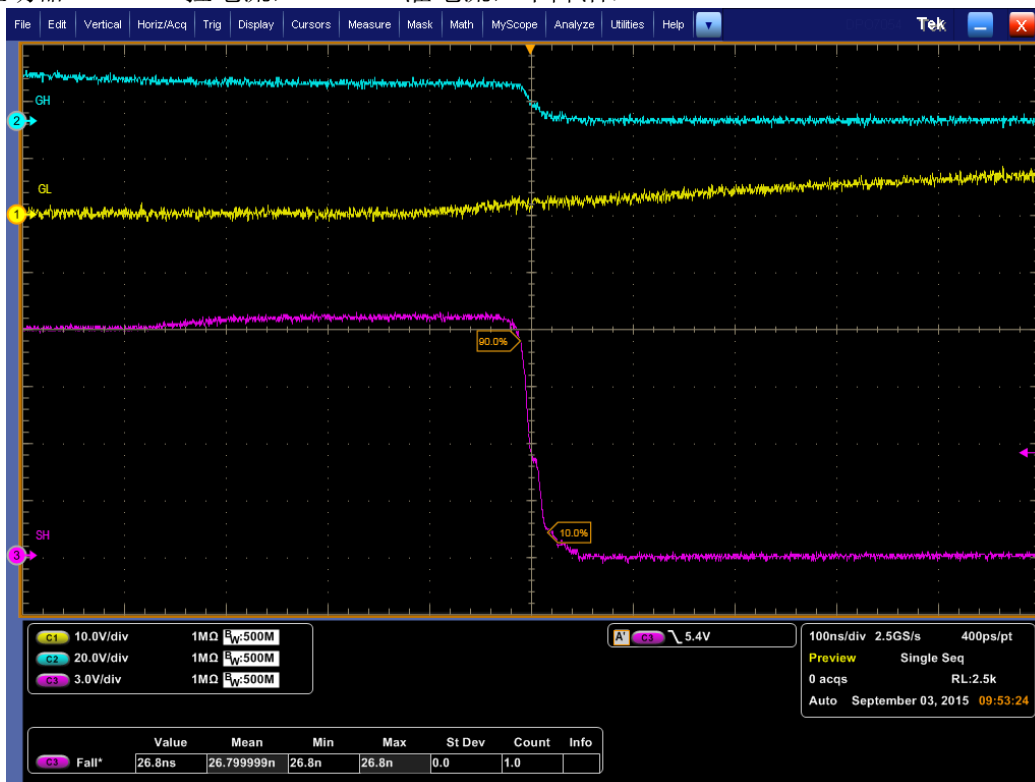
栅极驱动器 50 mA 拉电流, 60 mA 灌电流, 下降沿:



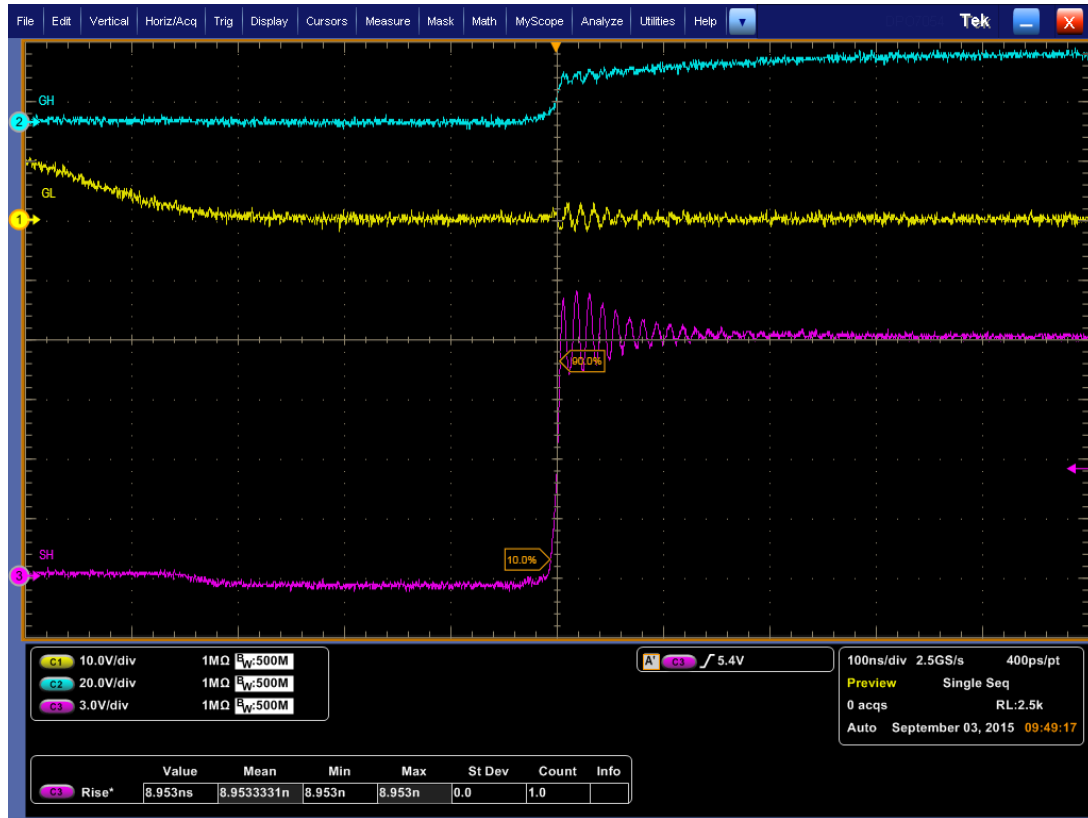
栅极驱动器 125 mA 拉电流, 250 mA 灌电流, 上升沿:



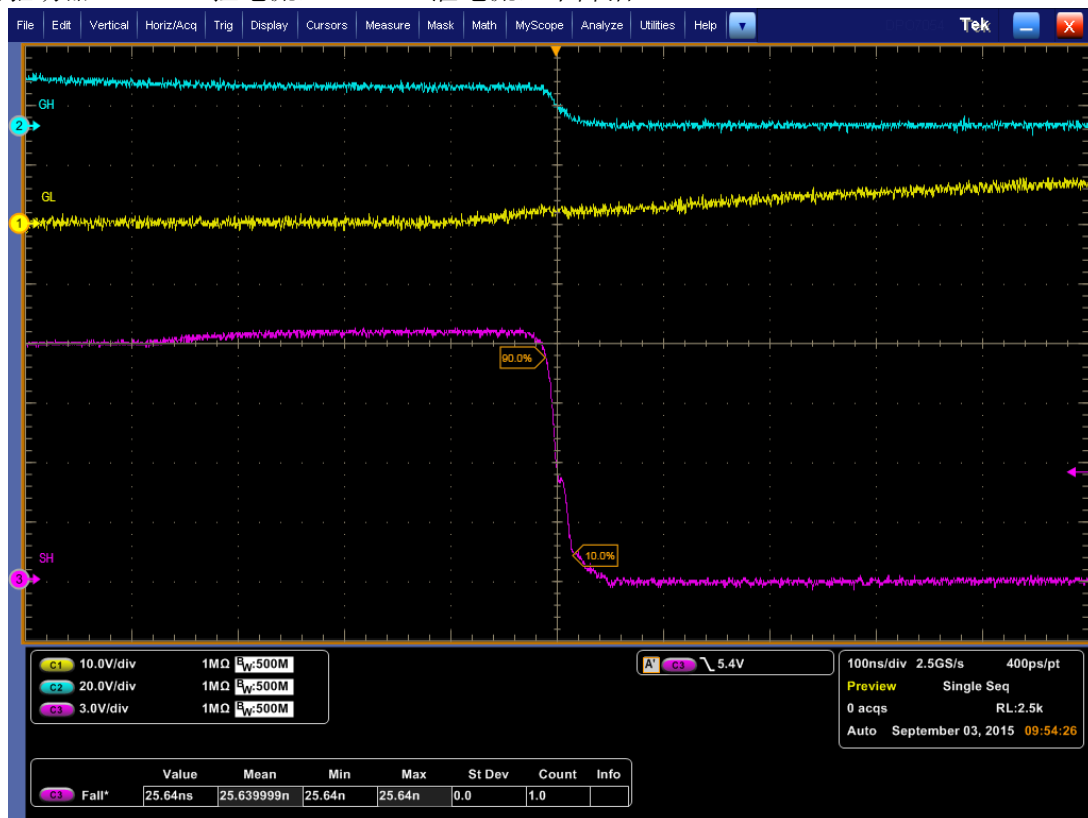
栅极驱动器 125 mA 拉电流, 250 mA 灌电流, 下降沿:



栅极驱动器 500 mA 拉电流, 750 mA 灌电流, 上升沿:



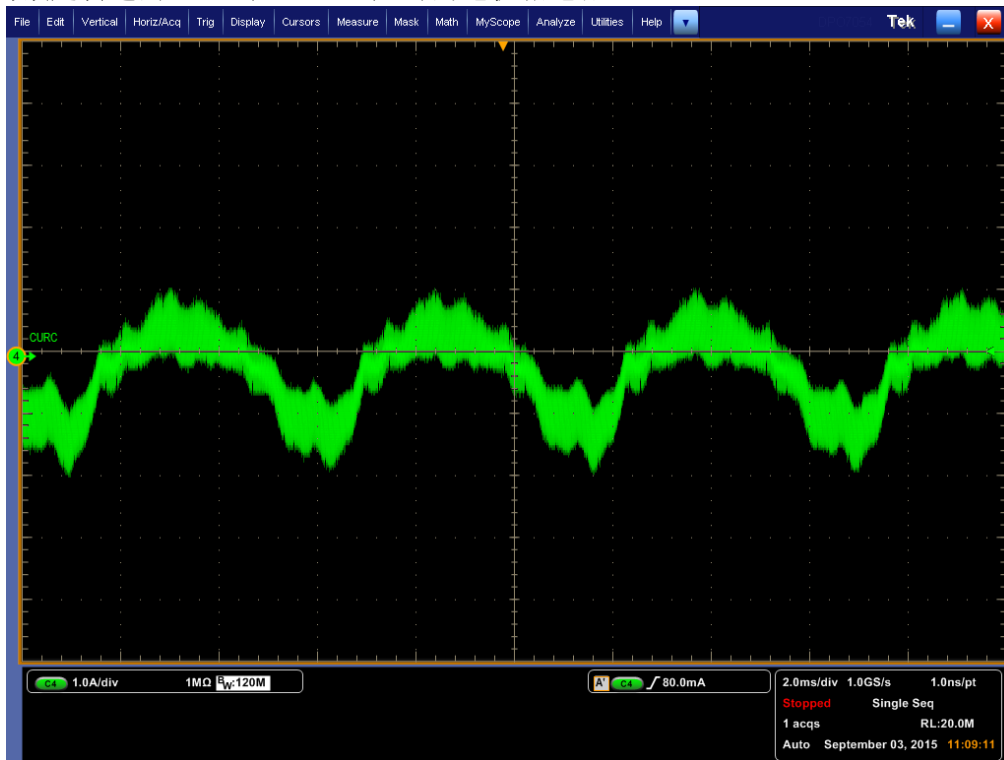
栅极驱动器 500 mA 拉电流, 750 mA 灌电流, 下降沿:



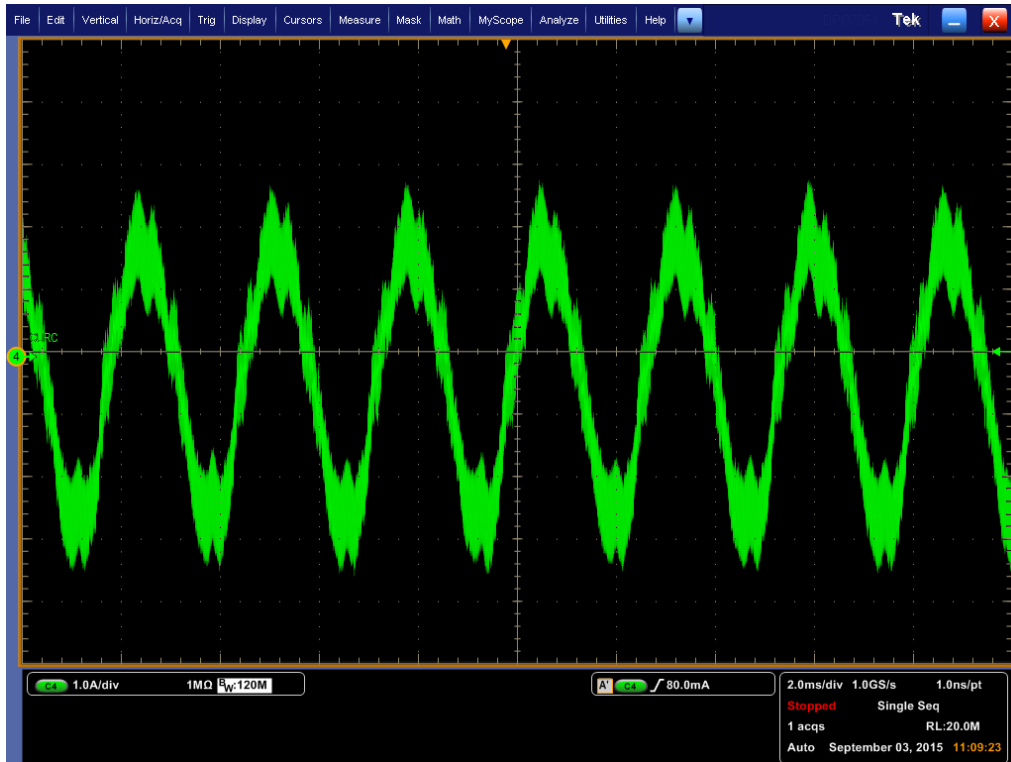
7.2 □□□□

7.2.1 □机相□流

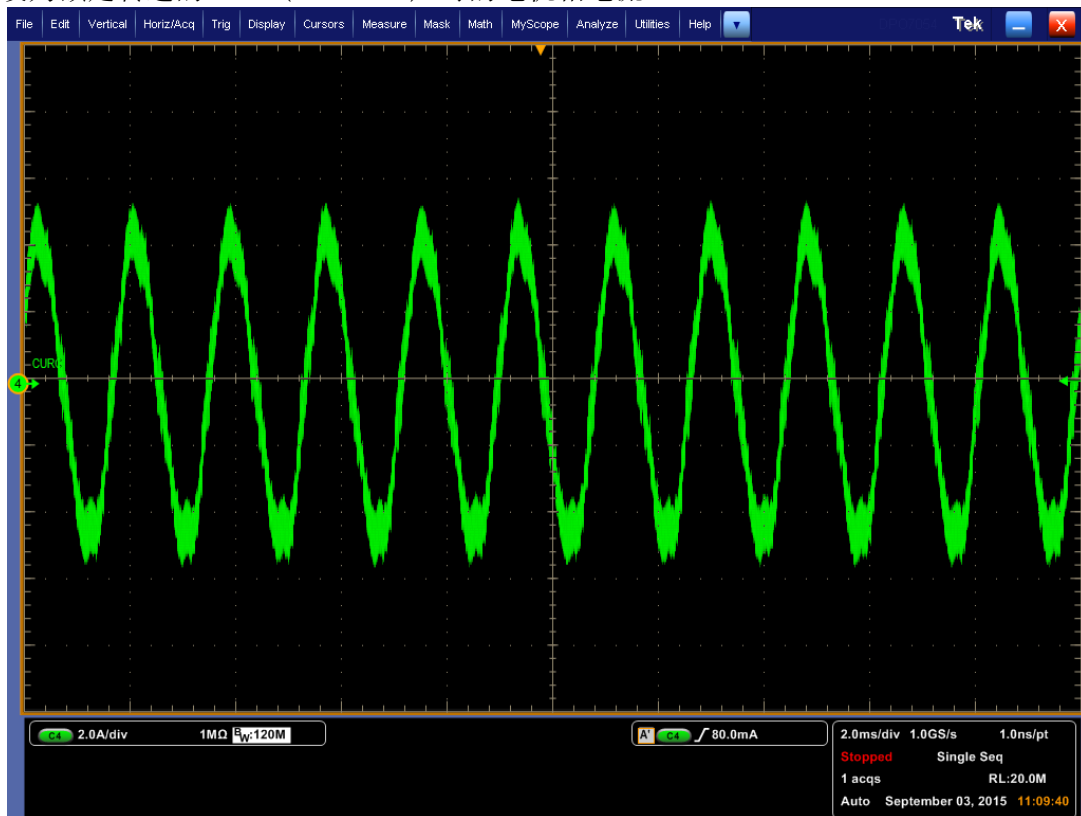
通过测试检测电机相电流与设定电机转速之间的关系。
 转速设为额定转速的 20% (1540 RPM) 时的电机相电流。



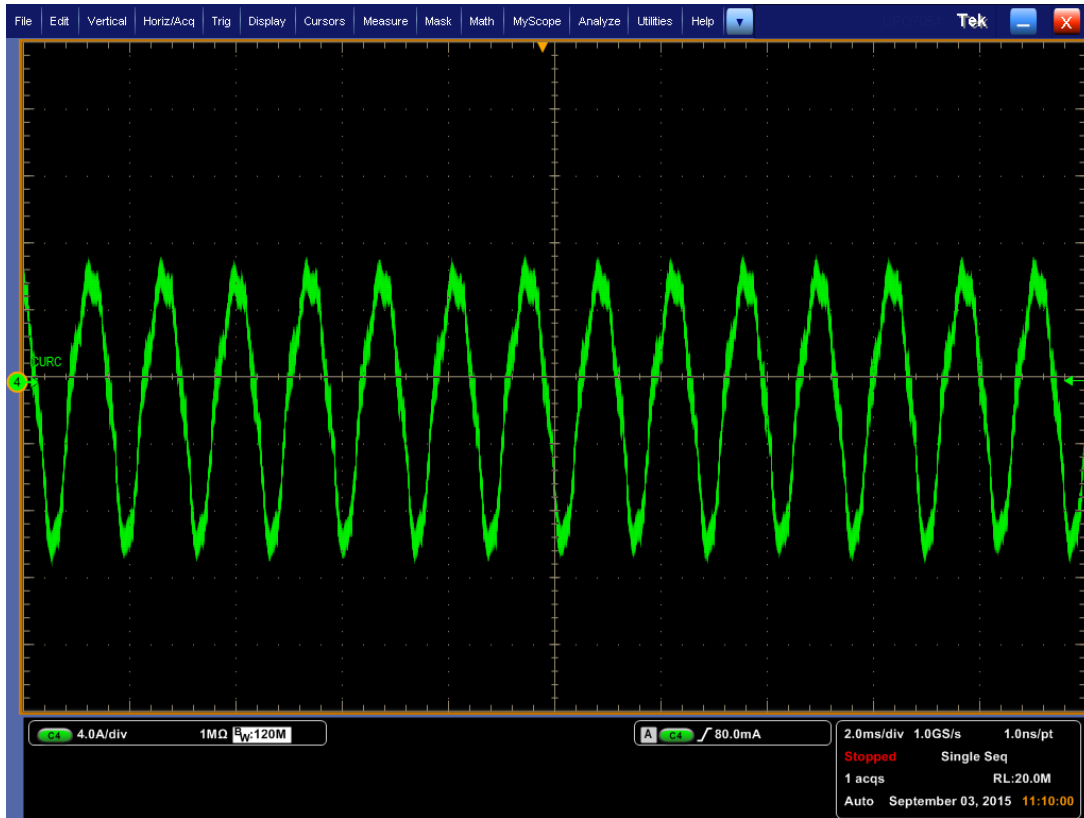
转速设为额定转速的 40% (3080 RPM) 时的电机相电流。



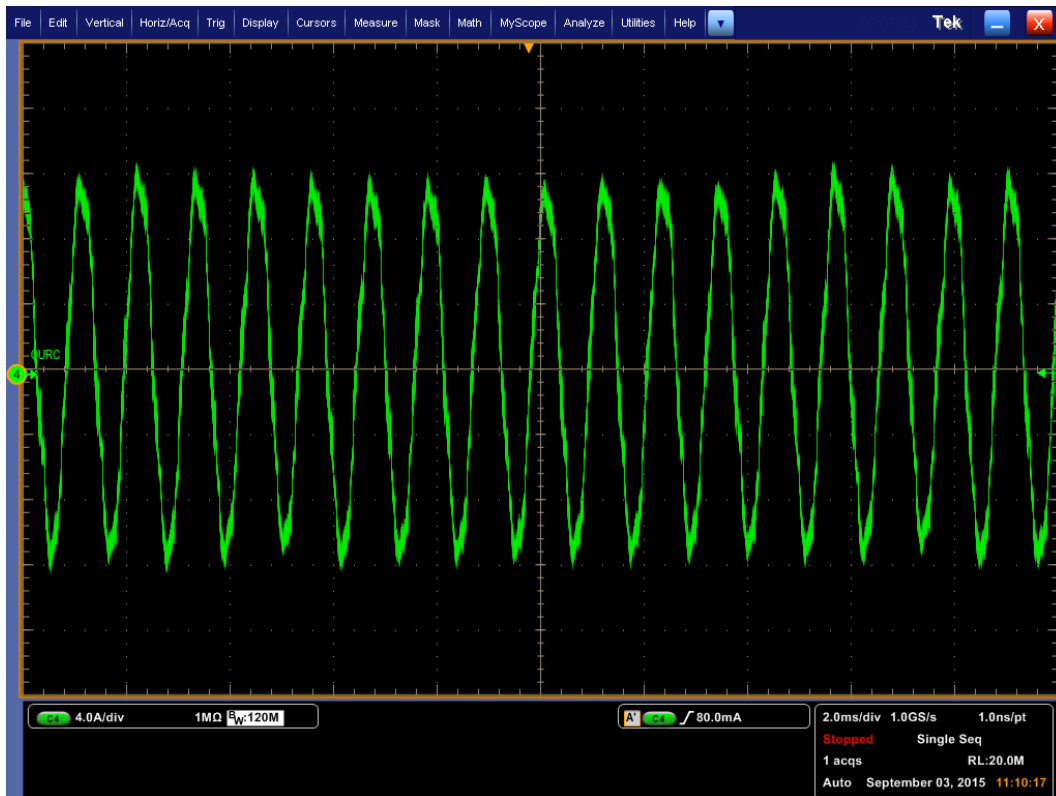
转速设为额定转速的 60% (4620 RPM) 时的电机相电流。



转速设为额定转速的 80% (6160 RPM) 时的电机相电流。

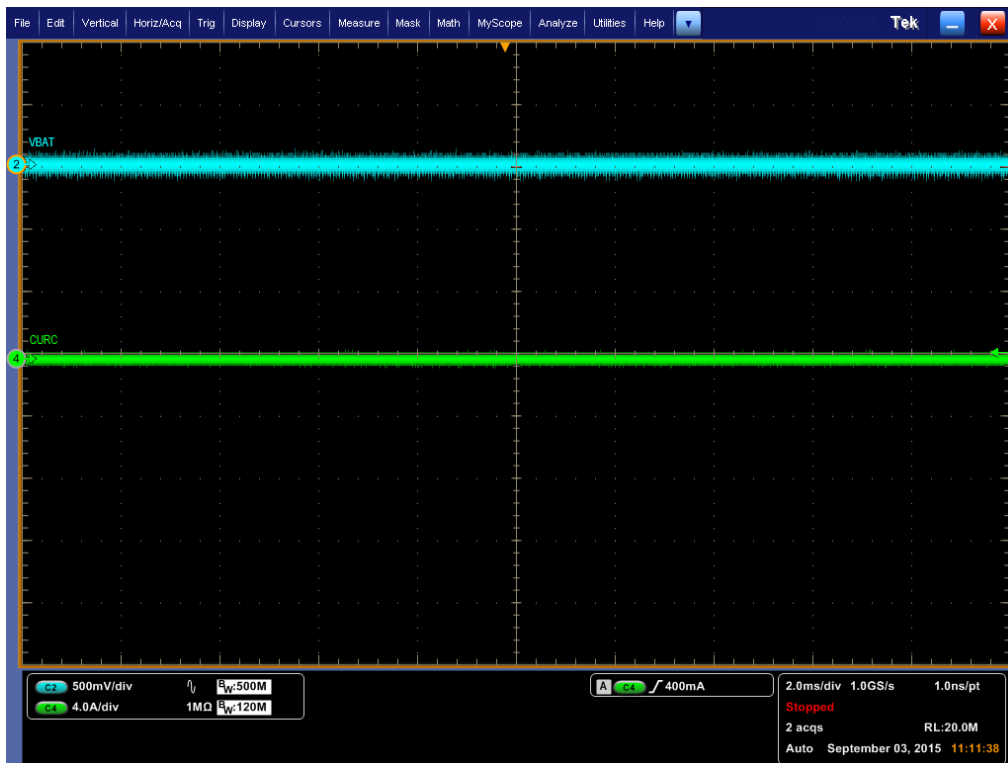


转速设为额定转速的 100% (7700 RPM) 时的电机相电流。

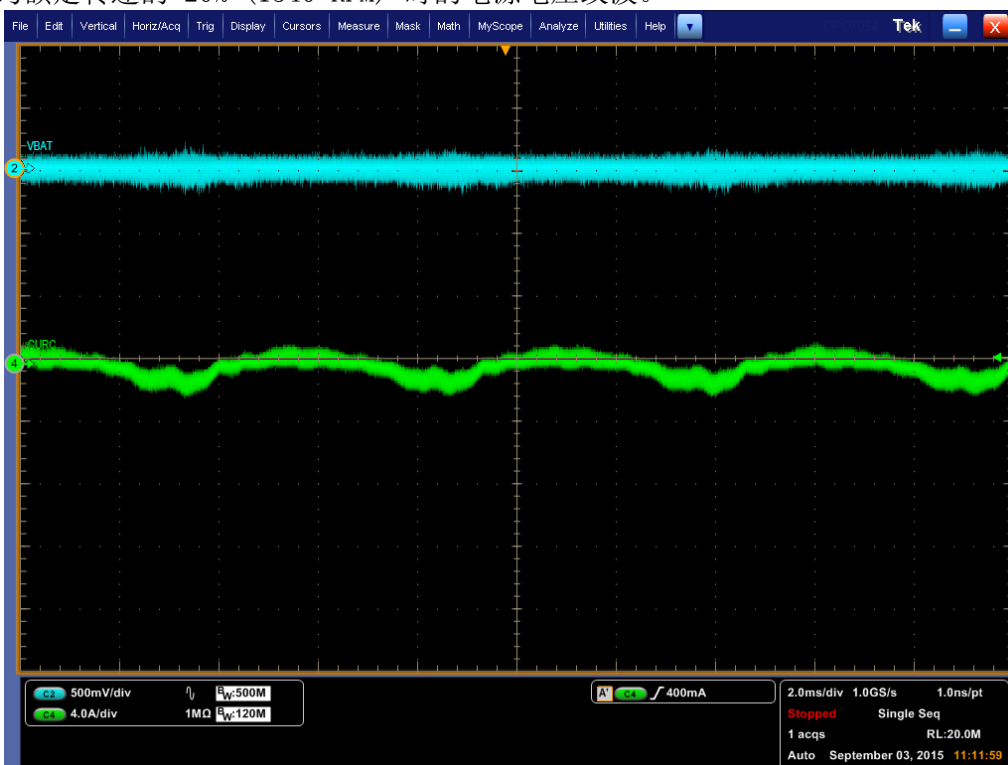


7.2.2 源纹波

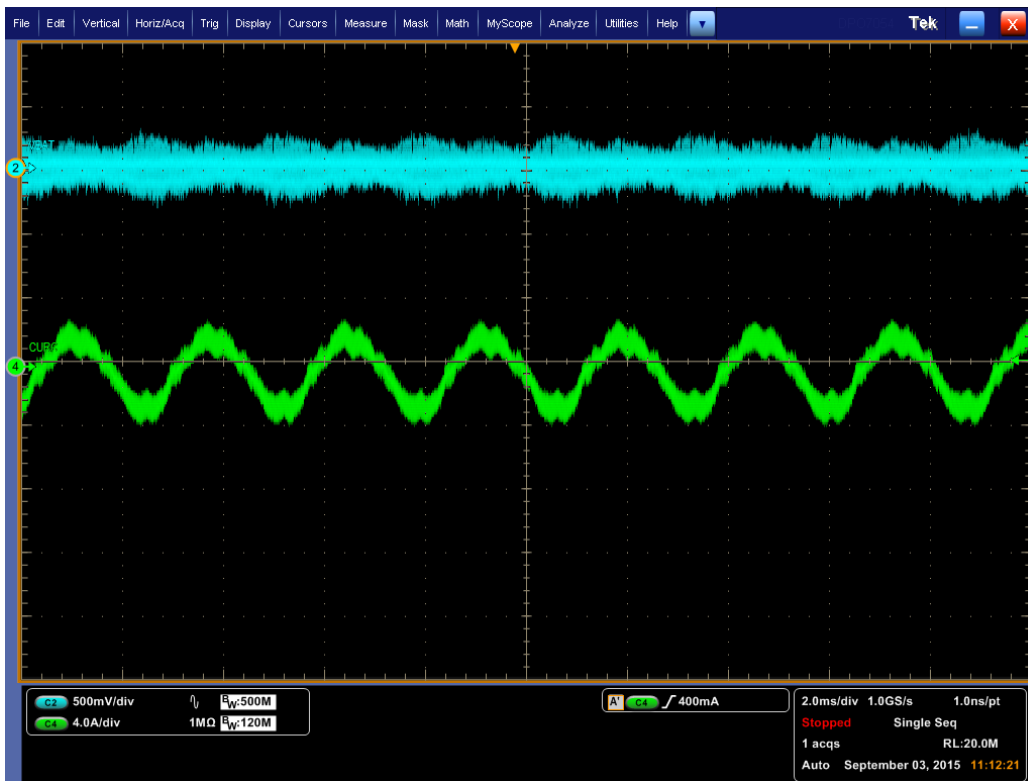
通过测试检测电源电压 (VBAT) 纹波与电机相电流之间的关系。
电机停止 (0 RPM) 时的电源电压纹波。



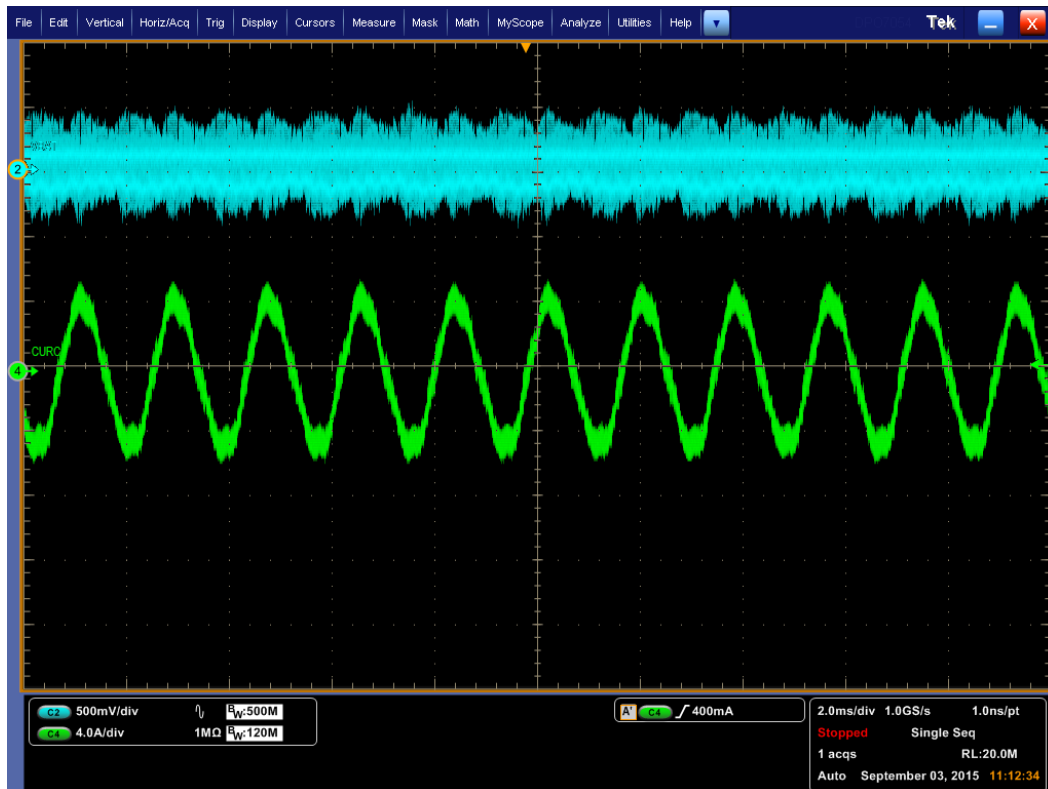
转速设为额定转速的 20% (1540 RPM) 时的电源电压纹波。



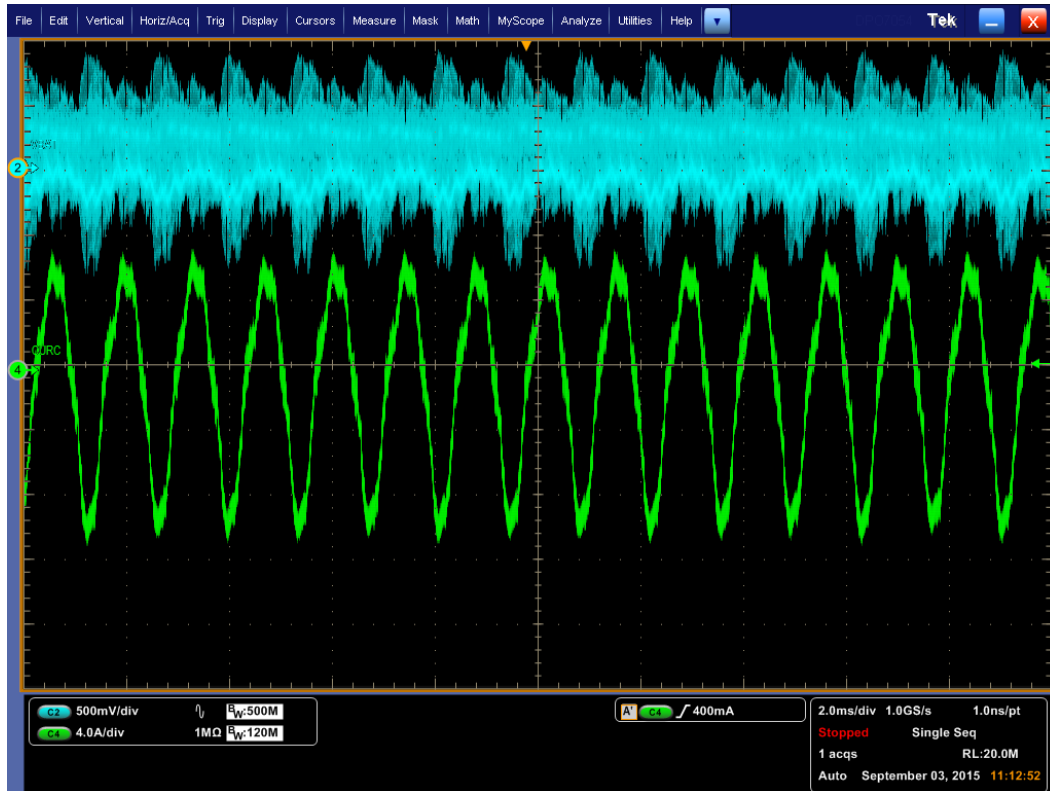
转速设为额定转速的 40% (3080 RPM) 时的电源电压纹波。



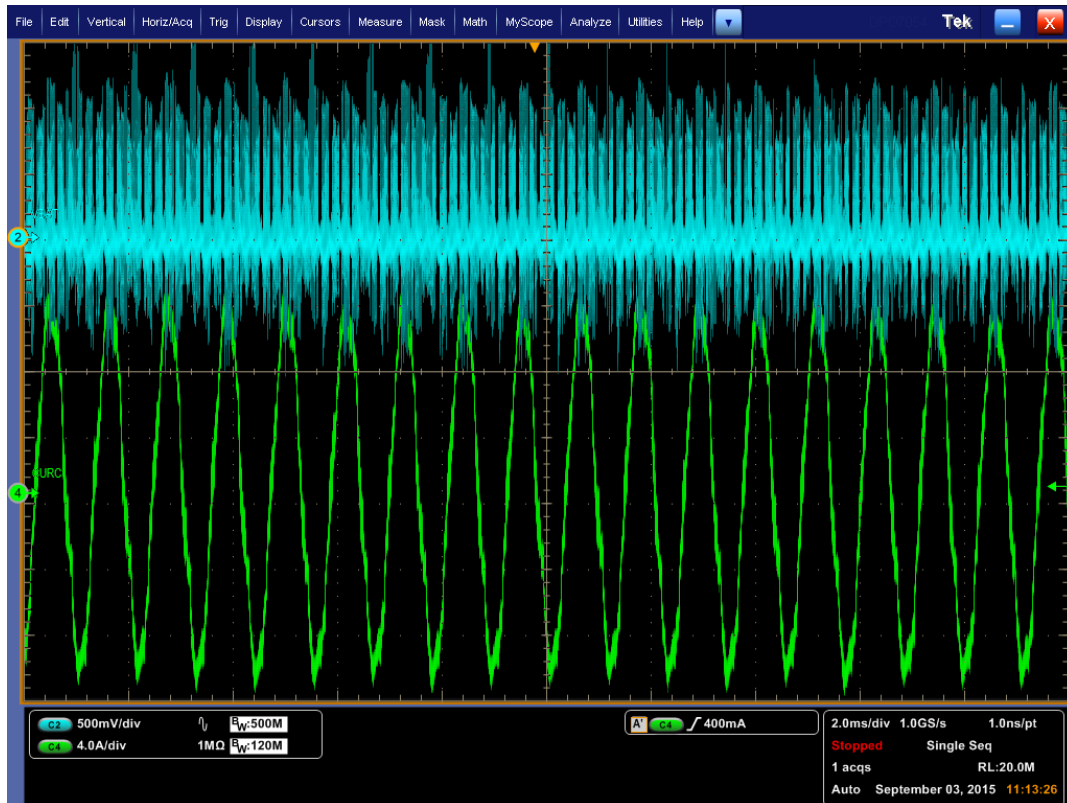
转速设为额定转速的 60% (4620 RPM) 时的电源电压纹波。



转速设为额定转速的 80% (6160 RPM) 时的电源电压纹波。

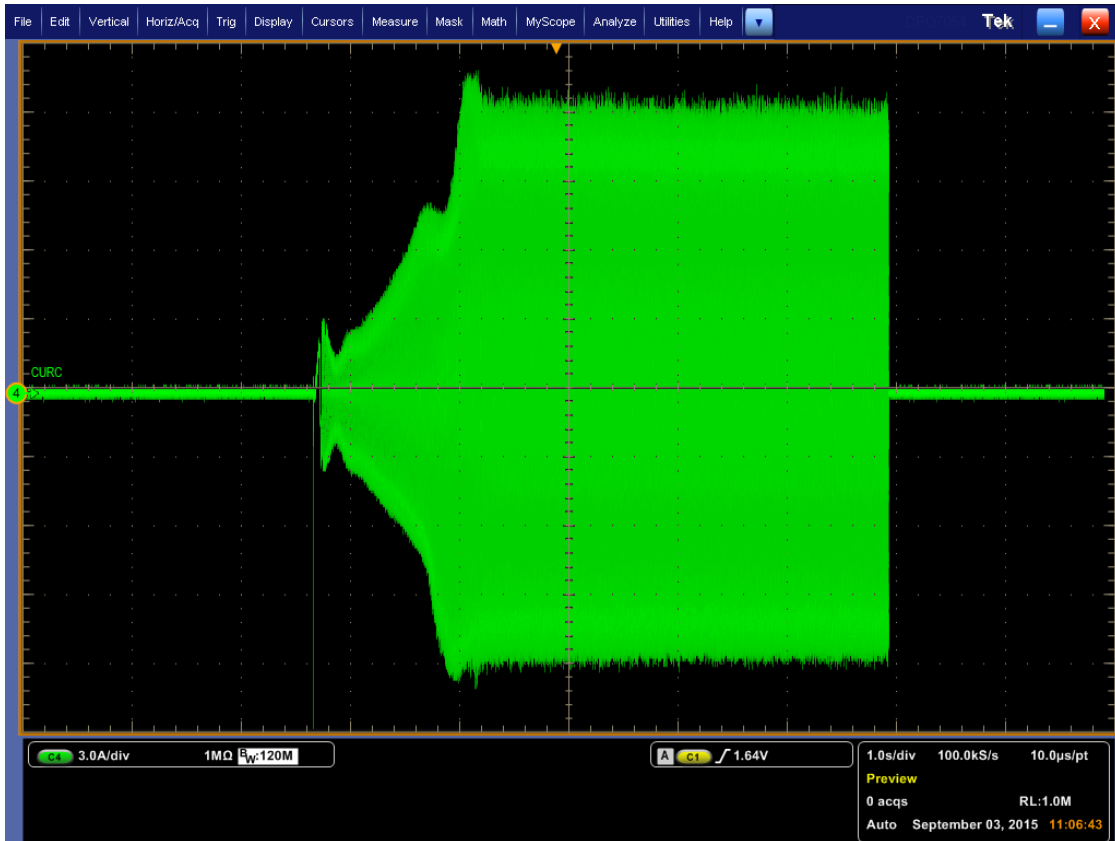


转速设为额定转速的 100% (7700 RPM) 时的电源电压纹波。



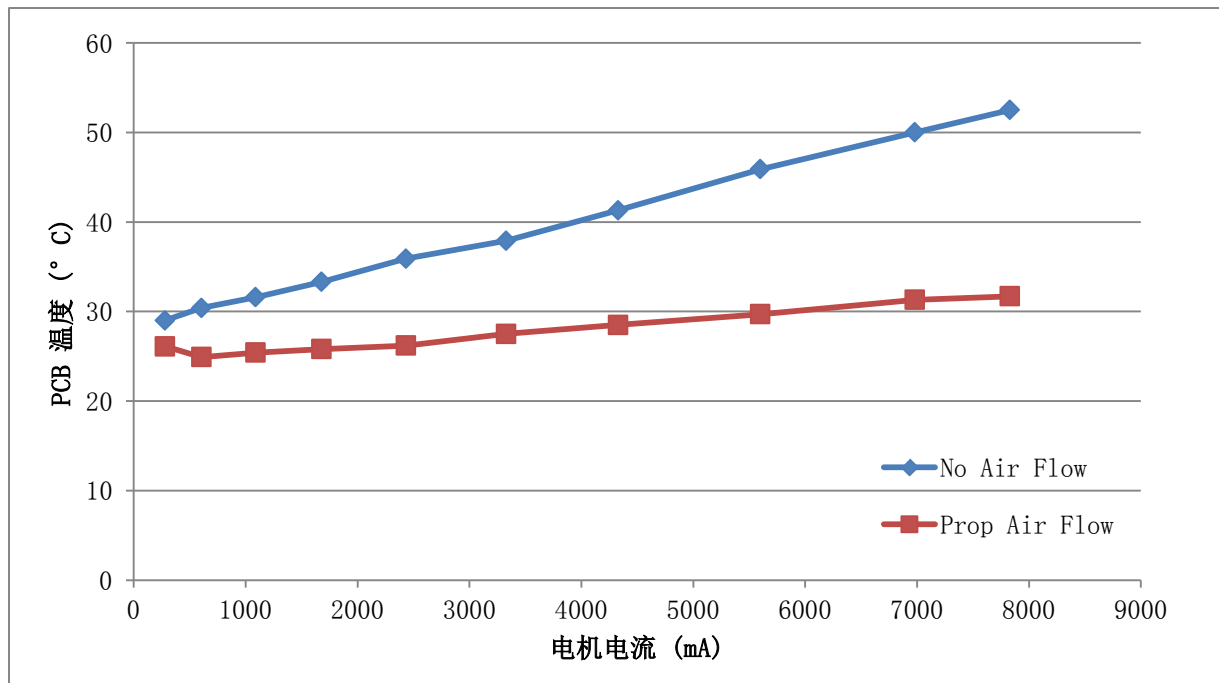
7.2.3 电机斜升

通过测试检测电机从停止 (0 RPM) 斜升至全速 (7700 RPM) 的情况。



7.2.4 PCB 温度

通过测试检测不同电机电流对应的 PCB（印刷电路板）温度。电机相电流通过电流探针测量，并记录 RMS 值。温度通过功率级中的热电偶进行记录。为模拟应用场景，还使用电机控制器（其驱动的电机螺旋桨（10"）的下方）进行测试。该螺旋桨提供主动冷却所需的气流。



8 □□文件

完整的设计文件可通过参考设计主页 [TIDA-00643](#) 下载。

9 □件文件

更新的 MotorWare HAL 和 PROJECT 文件可通过参考设计主页 [TIDA-00643](#) 下载。

10 参考□料

1. 德州仪器 (TI) InstaSPIN 解决方案, www.ti.com/c2x-instaspin-b
2. 德州仪器 (TI) 电机驱动解决方案, <http://www.ti.com/llds/ti/analog/motordrivers/overview.page>
3. 德州仪器 (TI) WEBENCH® 设计中心, <http://www.ti.com/webench>
4. 德州仪器 (TI) E2E 社区, <http://e2e.ti.com/>
5. 德州仪器 (TI) TMS320F28027F C2000 MCU, <http://www.ti.com.cn/product/cn/tms320f28027f>
6. 德州仪器 (TI) DRV8305 BLDC 栅极驱动器, <http://www.ti.com.cn/product/cn/drv8305>
7. 德州仪器 (TI) SIMPLE SWITCHER® 60 V 0.6 A 降压稳压器, <http://www.ti.com.cn/product/cn/lmr16006>
8. 德州仪器 (TI) CSD17573Q5B 30 V N 沟道 NexFET™ 功率 MOSFET, <http://www.ti.com.cn/product/cn/csd17573q5b>
9. 德州仪器 (TI) eCAP 模块参考指南, <http://www.ti.com/lit/ug/sprufz8a/sprufz8a.pdf>
10. 德州仪器 (TI) Code Composer Studio 集成开发环境, <http://www.ti.com.cn/tool/cn/ccstudio>
11. 德州仪器 (TI) MotorWare, <http://www.ti.com.cn/tool/cn/motorware>
12. DRV8305 评估板, <http://www.ti.com.cn/tool/cn/boostxl-drv8305evm>
13. TMSF28027F 评估板, <http://www.ti.com.cn/tool/cn/launchxl-f28027>

11 关于作者

Nicholas Oborny 是德州仪器 (TI) 电机驱动业务的应用工程师, 负责支持 TI 电机驱动产品组合、开发评估工具、产品演示、电机驱动培训和电机驱动 IC 开发。Nicholas 具有多年的硬开关功率级和无刷电机控制系统开发经验。Nicholas 毕业于大学城德克萨斯州农工大学, 拥有计算机工程理学学士学位。

Hector Hernandez Luque 是德州仪器 (TI) 的应用工程师实习生, 负责开发电机驱动业务的参考设计解决方案。Hector 拥有丰富的微处理器接口、软件开发、并行计算、原理图绘制和 PCB 布局设计经验。Hector 目前正在波多黎各大学马亚圭斯校区攻读计算机工程理学学士学位。Hector 是电气电子工程师协会 (IEEE) 的成员。

重要声明

德州仪器(TI) 及其下属子公司有权根据 JESD46 最新标准, 对所提供的产品和服务进行更正、修改、增强、改进或其它更改, 并有权根据 JESD48 最新标准中止提供任何产品和服务。客户在下订单前应获取最新的相关信息, 并验证这些信息是否完整且是最新的。所有产品的销售都遵循在订单确认时所提供的TI 销售条款与条件。

TI 保证其所销售的组件的性能符合产品销售时 TI 半导体产品销售条件与条款的适用规范。仅在 TI 保证的范围内, 且 TI 认为有必要时才会使用测试或其它质量控制技术。除非适用法律做出了硬性规定, 否则没有必要对每种组件的所有参数进行测试。

TI 对应用帮助或客户产品设计不承担任何义务。客户应对其使用 TI 组件的产品和应用自行负责。为尽量减小与客户产品和应用相关的风险, 客户应提供充分的设计与操作安全措施。

TI 不对任何 TI 专利权、版权、屏蔽作品权或其它与使用了 TI 组件或服务的组合设备、机器或流程相关的 TI 知识产权中授予的直接或间接版权限作出任何保证或解释。TI 所发布的与第三方产品或服务有关的信息, 不能构成从 TI 获得使用这些产品或服务的许可、授权、或认可。使用此类信息可能需要获得第三方的专利权或其它知识产权方面的许可, 或是 TI 的专利权或其它知识产权方面的许可。

对于 TI 的产品手册或数据表中 TI 信息的重要部分, 仅在没有对内容进行任何篡改且带有相关授权、条件、限制和声明的情况下才允许进行复制。TI 对此类篡改过的文件不承担任何责任或义务。复制第三方的信息可能需要服从额外的限制条件。

在转售 TI 组件或服务时, 如果对该组件或服务参数的陈述与 TI 标明的参数相比存在差异或虚假成分, 则会失去相关 TI 组件或服务的所有明示或暗示授权, 且这是不正当的、欺诈性商业行为。TI 对任何此类虚假陈述均不承担任何责任或义务。

客户认可并同意, 尽管任何应用相关信息或支持仍可能由 TI 提供, 但他们将独立负责满足与其产品及其应用中使用 TI 产品相关的所有法律、法规和安全相关要求。客户声明并同意, 他们具备制定与实施安全措施所需的全部专业技术和知识, 可预见故障的危险后果、监测故障及其后果、降低有可能造成人身伤害的故障的发生机率并采取适当的补救措施。客户将全额赔偿因在此类安全关键应用中使用任何 TI 组件而对 TI 及其代理造成的任何损失。

在某些场合中, 为了推进安全相关应用有可能对 TI 组件进行特别的促销。TI 的目标是利用此类组件帮助客户设计和创立其特有的可满足适用的功能安全性标准和要求的终端产品解决方案。尽管如此, 此类组件仍然服从这些条款。

TI 组件未获得用于 FDA Class III (或类似的生命攸关医疗设备) 的授权许可, 除非各方授权官员已经达成了专门管控此类使用的特别协议。

只有那些 TI 特别注明属于军用等级或“增强型塑料”的 TI 组件才是设计或专门用于军事/航空应用或环境的。购买者认可并同意, 对并非指定面向军事或航空航天用途的 TI 组件进行军事或航空航天方面的应用, 其风险由客户单独承担, 并且由客户独立负责满足与此类使用相关的所有法律和法规要求。

TI 已明确指定符合 ISO/TS16949 要求的产品, 这些产品主要用于汽车。在任何情况下, 因使用非指定产品而无法达到 ISO/TS16949 要求, TI 不承担任何责任。

	产品		应用
数字音频	www.ti.com.cn/audio	通信与电信	www.ti.com.cn/telecom
放大器和线性器件	www.ti.com.cn/amplifiers	计算机及周边	www.ti.com.cn/computer
数据转换器	www.ti.com.cn/dataconverters	消费电子	www.ti.com.cn/consumer-apps
DLP® 产品	www.dlp.com	能源	www.ti.com.cn/energy
DSP - 数字信号处理器	www.ti.com.cn/dsp	工业应用	www.ti.com.cn/industrial
时钟和计时器	www.ti.com.cn/clockandtimers	医疗电子	www.ti.com.cn/medical
接口	www.ti.com.cn/interface	安防应用	www.ti.com.cn/security
逻辑	www.ti.com.cn/logic	汽车电子	www.ti.com.cn/automotive
电源管理	www.ti.com.cn/power	视频和影像	www.ti.com.cn/video
微控制器 (MCU)	www.ti.com.cn/microcontrollers		
RFID 系统	www.ti.com.cn/rfidsys		
OMAP应用处理器	www.ti.com.cn/omap		
无线连通性	www.ti.com.cn/wirelessconnectivity	德州仪器在线技术支持社区	www.deyisupport.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2016, Texas Instruments Incorporated