

简介

作者: Alexandru Zîrnea 和 Leona Pop, Microchip Technology Inc.

本应用笔记介绍了一种无传感器无刷直流 (Brushless Direct Current, BLDC) 电机控制方法, 该方法使用基于 AVR® DA 和 AVR DB 系列 AVR128DA48 或 AVR128DB48 单片机的接插模块 (Plug-In Module, PIM)。

文档的第一部分重点介绍了软件实现所基于的电机控制理论, 其中包括有传感器和无传感器控制方法、关于梯形换向的详细信息以及其他关于如何驱动电机的重要信息。

第二部分专门介绍系统的软件和硬件实现。

示例应用使用连接到 dsPICDEM™ MCLV-2 开发板 (电机控制低电压) (MCLV-2) 的基于 AVR128DA48 或 AVR128DB48 的 PIM。MCLV-2 开发板提供了一个电位器用于调节旋转速度, 而输入保护和反电动势 (Back Electromotive Force, BEMF) 电路则位于 PIM 上。

该系统能够实现再生制动, 这是所使用驱动方法的固有特性。

电机的驱动波形独立于内核产生, 利用单片机外设的硬件功能。

此外, 应用还采用了多种保护机制, 如失速检测、错误输入检测以及双向过流保护。

该系统可以通过适当调整固件来驱动各种 BLDC 电机。文档末尾概述了针对不同类型电机所需参数的调整指南。

BLDC 电机广泛应用于无线电动工具、计算机、硬盘驱动器、多媒体设备、小型电子设备中的冷却风扇、电动车辆等领域, 是未来电气化发展中增长迅速的市场。所提出的解决方案通过使用 8 位单片机创建了一个极具成本效益的电机驱动器, 而传统上创建此类设备只使用 16 位或更多位数的单片机。



View the Motor Control PIM Code on GitHub

[Click to browse repository](#)

目录

简介..... 1

1. 概述.....3

2. 相关器件..... 4

3. 电机控制理论.....5

4. 固件实现..... 15

5. 接插模块（PIM）板说明..... 25

6. 调整..... 34

7. 结论..... 37

8. 参考资料.....38

9. 版本历史.....39

Microchip信息.....40

1. 概述

本应用笔记介绍了驱动 BLDC 电机所需的电机控制理论，包含有传感器和无传感器电机控制的比较、功率级的典型实现、调制技术以及对定制 PIM 的详细说明。

此外，文档还详细介绍了双极切换，这是本示例的重点。

为获得转子位置，文中介绍了零交叉点检测方法，该方法使用带有由电机虚拟中性点构成的可变参考电压的比较器。

电机在完成对准程序后从初始位置启动。这样做是为了使电机在启动时能够跟随生成的开环波形。

在编译时，用户可以选择电机必须运行的模式：开环或闭环。

MCLV-2 开发板上的电位器提供了一个由 ADC 读取的值，该值用于计算波形定时器的参考占空比，此读取过程以及电流测量是与软件的电机逻辑驱动异步进行的。这部分在主循环中运行，而电机控制则在中断时运行，以获得尽可能最佳的响应时间。

在本文档的末尾，用户可以找到关于调整固件参数以获得所需结果的简要介绍。

此应用示例使用以下外设：

- 模拟比较器（Analog Comparator, AC）
- 模数转换器（Analog-to-Digital Converter, ADC）
- I/O 引脚控制器（PORT）
- A 型定时器/计数器（TCA）
- B 型定时器/计数器（TCB）
- D 型定时器/计数器（TCD）
- 定制可配置逻辑（Custom Configurable Logic, CCL）
- 参考电压（ V_{REF} ）
- 通用同步/异步收发器（Universal Synchronous Asynchronous Receiver Transmitter, USART）

准备工作：

1. 软件：

- Microchip Studio 7 集成开发环境（Integrated Development Environment, IDE）和 AVR DX 器件系列包（DFP 1.7.85 或更新版本）

2. 硬件：

- AVR128DA48 电机控制 PIM 或带内部运放配置板的 AVR128DB48 电机控制 PIM
- dsPICDEM™ MCLV-2 开发板（电机控制低电压）
- AC300020——24V 三相无刷直流电机
- AC002013——24V 电源
- MPLAB® PICkit™ 4 在线调试器

MCLV-2 开发板旨在控制无刷电机或永磁同步电机，同时支持有传感器和无传感器两种工作模式。在本演示中，它将与 AVR128DA48 或 AVR128DB48 电机控制 PIM 一起使用，以无传感器工作模式驱动无刷电机。该开发板提供三相驱动器、测量反馈信号的方式以及板上运放。AVR128DB48 PIM 使用附加配置板的原因是内部运算放大器信号的连接方式不同。

本演示将使用 MCLV-2 开发板的其他一些功能（如编程接口和串行接口），以接收关于电机工作参数的信息。此外，该开发板上还有 LED、按钮和电位器，可提供用户接口以显示哪些 PWM 输出处于有效状态、设置电机方向并启动和停止电机，以及更改电机速度。

2. 相关器件

本章列出了本文档的相关器件。下图给出了不同系列的器件之间的关系，并注明了不同的引脚数与存储器大小：

- 从下到上迁移无需修改代码，因为这些器件的引脚彼此兼容，后者可提供相同甚至更多的功能
- 水平向左移植会减少引脚数，进而减少可用的功能
- 具有不同闪存大小的器件通常也具有不同的 SRAM 和 EEPROM

图 2-1. AVR[®] DA 系列概览

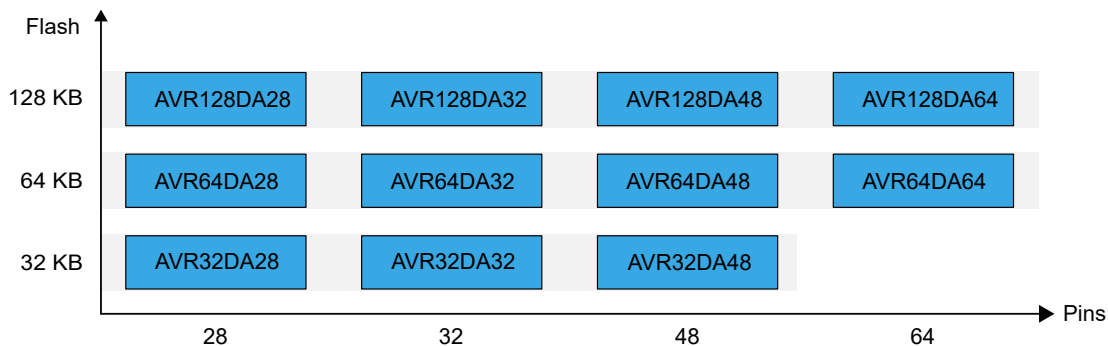
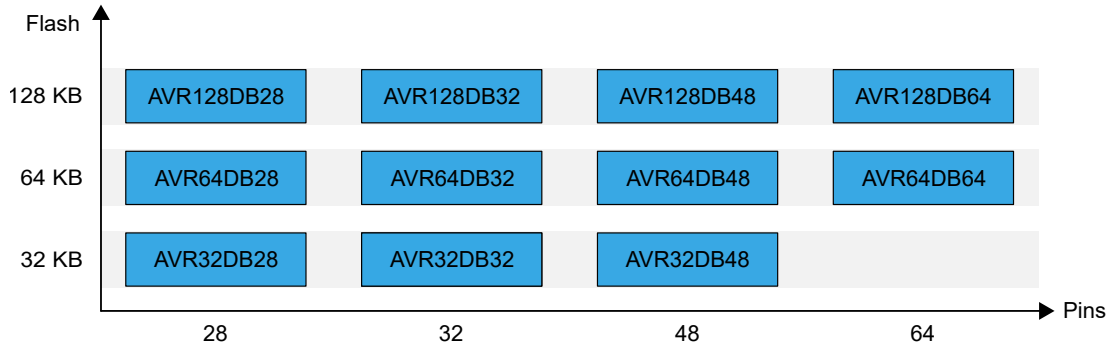


图 2-2. AVR[®] DB 系列概览



3. 电机控制理论

本节介绍电机控制概念及其在软件和硬件实现中的应用。

3.1. 有传感器控制与无传感器控制对比

BLDC 电机是一种尺寸小巧的同步电机，可提供超高效率且无需任何维护。相比直流有刷电机，此类电机拥有很大优势，非常适合要求非常严格的消费类和工业应用。

直流有刷电机采用机械换向。它们具有电刷和换向器，根据特定的转子位置向线圈施加电压。

与直流有刷电机不同，BLDC 电机的换向取决于转子的位置，此位置必须是已知的。

要确定转子的位置，可以使用有传感器或无传感器算法。

3.1.1. 有传感器控制

一种方法是在定子外部使用传感器，定子会相应地对转子上的磁铁作出反应，并给出可用于确定位置的信号。通常，三个霍尔效应传感器将在两次转换之间提供 60 度的差值。

使用传感器的优势在于，通过了解每个霍尔效应传感器的状态，即使在零速下，系统上电时也能获知转子的位置。

使用传感器的主要缺点在于，它们会增加电机的成本和复杂性，同时还需要辅助电源。

3.1.2. 无传感器控制

另一种方法是使用无传感器控制，它只需要控制板上的几个无源元件。

这种方法的原理是检测转子中磁铁运动所产生的悬空相上的电压，称为反电动势（BEMF）。BEMF 为单片机提供关于转子当前位置的数据。

由于一个线圈中产生的电压与转子速度成正比，因此在极低的速度下，BEMF 的幅值非常小且存在难以滤除的噪声。因此，存在一种开环启动过程，其中转子异步跟随由单片机产生的磁场。速度持续增加，直到 BEMF 的幅值达到可检测的程度。

无传感器控制的优点在于，任何有传感器电机都可以在无传感器模式下驱动，并且在可行的情况下可作为故障安全控制方法。此外，这种控制方法也更加可靠，因为除了电机的固有 BEMF 以外，系统不依赖于其他外部输入。

与有传感器控制不同，霍尔效应传感器提供逻辑信号，没有任何噪声，而无传感器控制需要信号滤波和调理，这是由于电机结构、电源噪声和其他因素造成的。

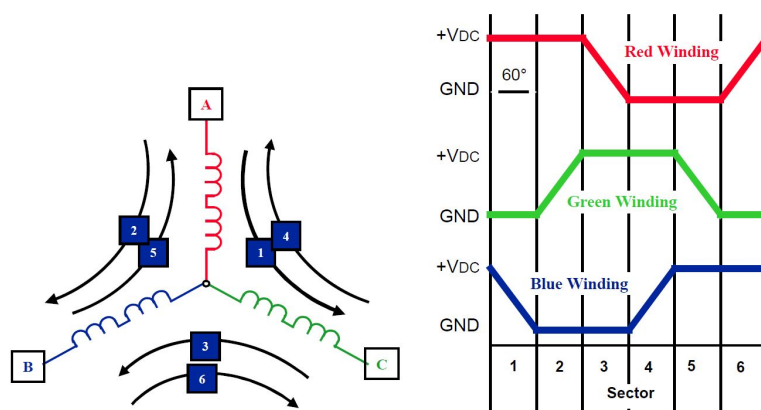
由于 BEMF 滤波，电机速度的突然变化无法被精确检测到，并可能导致系统发生失锁。有传感器控制不受此问题影响，因为无论电机工作条件如何，霍尔传感器都能提供信号。

3.2. 六步（梯形）BLDC 换向

梯形控制由三相（在这种情况下为 A、B 和 C）组成，其中两相被驱动，第三相保持悬空，从而能够检测到由转动转子产生的 BEMF 的过零。

每个步骤或扇区表示 360 度或一次完整电旋转中的 60 度。

图 3-1. 理想梯形波形



六步换相按如下方式完成：

步骤 1:	步骤 4:
• 红色绕组被正向驱动 • 绿色绕组被负向驱动 • 蓝色绕组未被驱动	• 绿色绕组被正向驱动 • 红色绕组被负向驱动 • 蓝色绕组未被驱动
步骤 2:	步骤 5:
• 红色绕组被正向驱动 • 蓝色绕组被负向驱动 • 绿色绕组未被驱动	• 蓝色绕组被正向驱动 • 红色绕组被负向驱动 • 绿色绕组未被驱动
步骤 3:	步骤 6:
• 绿色绕组被正向驱动 • 蓝色绕组被负向驱动 • 红色绕组未被驱动	• 蓝色绕组被正向驱动 • 绿色绕组被负向驱动 • 红色绕组未被驱动

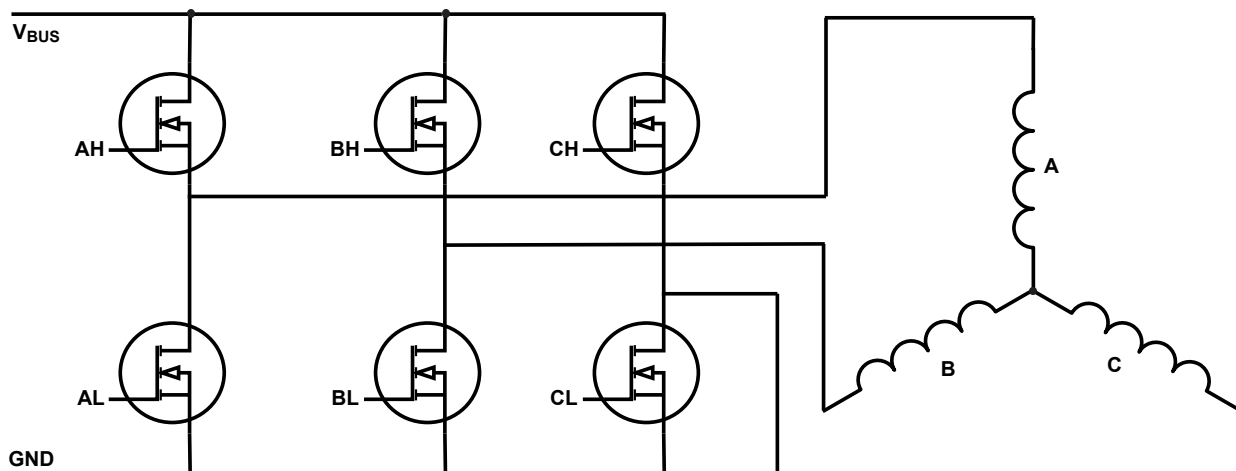
在恒速或小范围速度变化的情况下，两次换向之间的周期是相等的。这是闭环控制电机时的估算方法。

这种实现方式使用梯形换向，原因在于它非常简单且可以利用 8 位单片机轻松达成，因为它只需极少的存储器资源和处理能力。

3.3. 功率级

功率级由一个逆变器组成，该逆变器包含三个可以将每相连接到电源电压或地的半桥。开关元件通常选用 MOSFET（用于低电压应用）或 IGBT（用于高电压应用）。

图 3-2. 典型功率级原理图



AH、BH 和 CH 代表来自单片机的上桥臂命令信号，而 AL、BL 和 CL 代表下桥臂信号。

除了功率晶体管外，还使用 MOSFET/IGBT 驱动器来确保良好的上升和下降时间，并提供上桥臂和下桥臂晶体管所需的栅极电压。

组合逻辑电路通过在上桥臂和下桥臂命令信号之间添加死区来避免发生直通。

该死区确保同一分支上的 MOSFET 或 IGBT 完全关断，不会同时导通电流。

3.4. 调制技术

要控制 BLDC 电机的速度，需要调整施加到电机各相的电压。这可以像直流有刷电机那样轻松实现，即通过脉宽调制（Pulse-Width Modulation, PWM）来调节电压。因此，电机速度与 PWM 信号的占空比直接成正比。

以下章节介绍了两种开关方法及其优缺点。

3.4.1. 单极开关

单极开关需要在一个梯形步中导通两个晶体管。

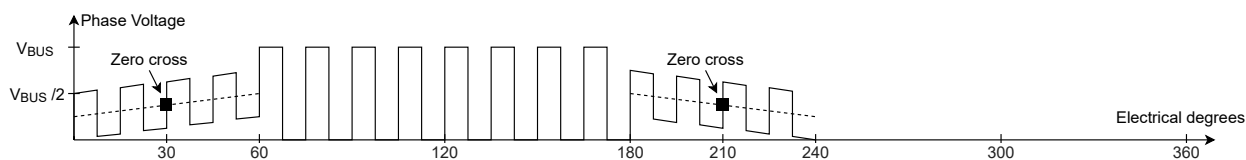
下桥臂晶体管一直导通，而上桥臂晶体管接收 PWM 信号。在大多数应用中，上桥臂晶体管的控制由 MOSFET 驱动器处理。MOSFET 驱动器有一个自举电容，在下桥臂导通时充电。如果使用的方法在下桥臂上应用 PWM，自举电容的充电将取决于占空比，在低占空比时会导致充电不足，最终使晶体管的 $R_{DS(on)}$ 增大，降低整体效率，或导致驱动器 IC 芯片发生欠压锁定。

在无传感器控制中，这种方法存在如下几个挑战：

- 在 BEMF 暴露的时间内，由于所有线圈之间的磁耦合，波形会被斩波，因为悬空相中会感应出 PWM。因此，需要同步方法在特定时间对 BEMF 进行“采样”。理想的采样时间是在高电平脉冲的中间点，此时 BEMF 稳定且不发生变化。
- 任何覆盖波形的噪声都难以滤除，并且会在错误的时间触发过零，从而导致效率降低并增加声学 and 电气噪声。
在这种方法中，传统的 RC 滤波方法被证实低速下效率极低，因为它需要较大的值，并且在高速时会引入大量延时。

图 3-3 中给出了使用单极开关时的典型波形。

图 3-3. 单极开关相电压（50%占空比）



这是大多数简单设计中使用的经典场景。

3.4.2. 双极开关

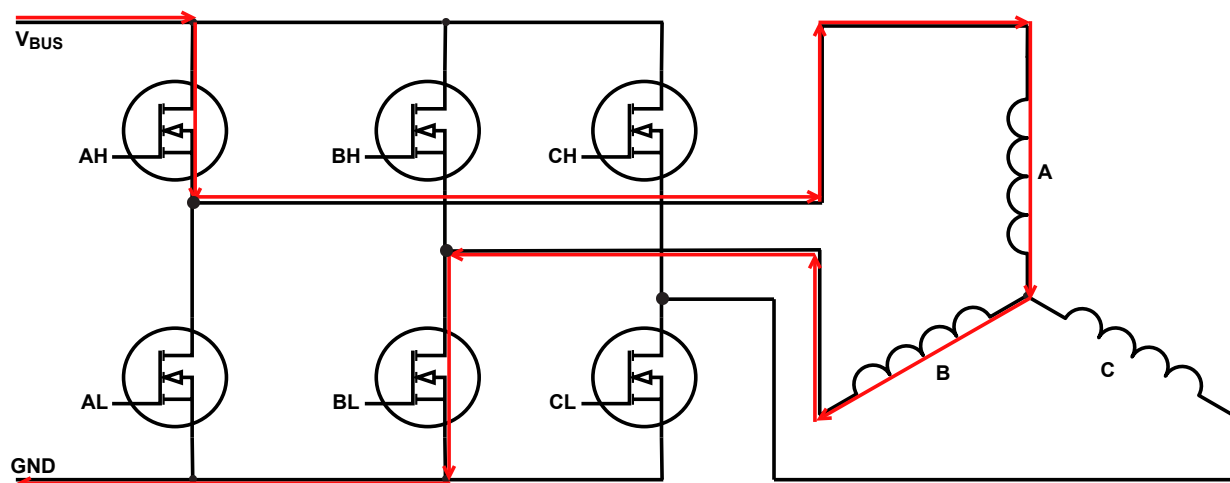
与单极开关不同，双极开关需要在一个梯形步中导通四个晶体管。

与单极开关相比，在这种情况下，要导通的另外两个晶体管与每个步特有的晶体管互补。

以下情况下会表现出这种行为：

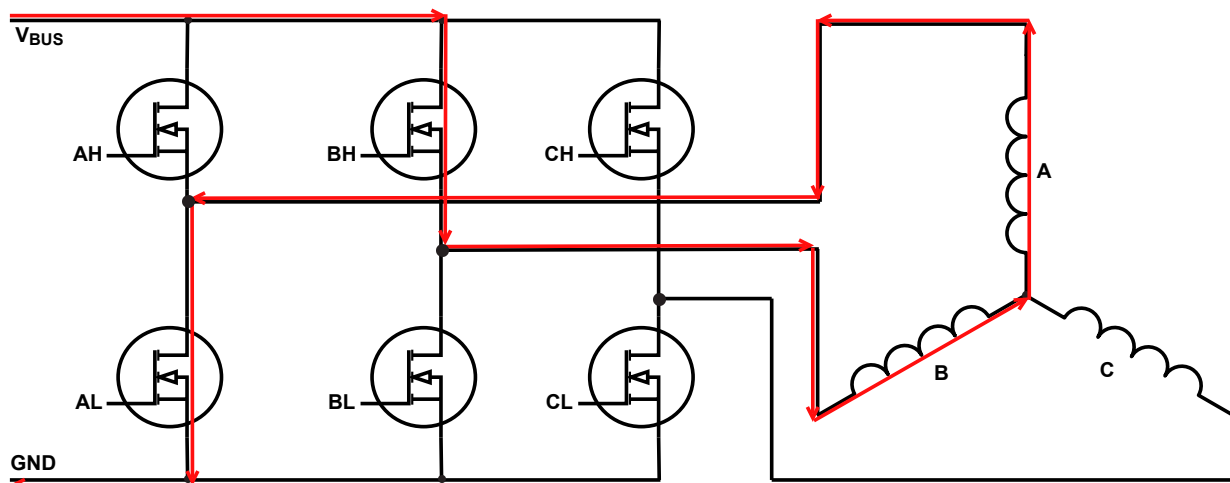
特征状态为 AH-BL，这意味着 A 相连接到高电平，B 相连接到低电平。这是通过导通 A 相半桥的上桥臂晶体管 and B 相的下桥臂晶体管实现的。

图 3-4. AH-BL 状态下的电流流向



AH-BL 的互补状态为 AL-BH，电流将以这种方式流动。请参见下面的图 3-5。

图 3-5. AL-BH 状态下的电流流向



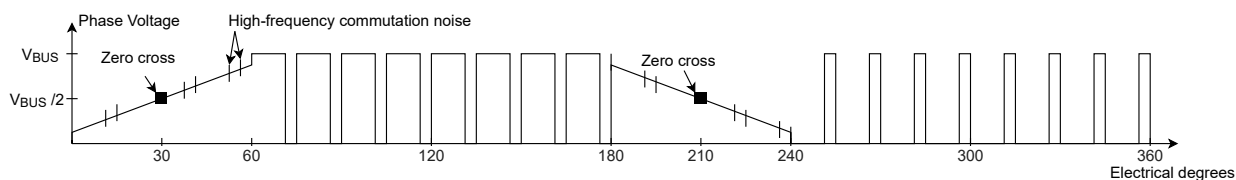
对于单极驱动，如果零速对应于 0% 占空比，则此情况下零速对应于 50% 占空比，此时互补状态时间等于特征状态时间。

这种方法的优点在于能产生非常干净的 BEMF，因为悬空相中感应的所有电压都被互补状态抵消，从而获得只包含高频尖峰的波形，这些尖峰可使用具有极小时间常数的 RC 滤波器进行滤除。

这种方法的主要缺点在于，电机产生更多的电磁和声学噪声，而且由于需要另一对晶体管进行开关，导致整个系统的损耗增加。

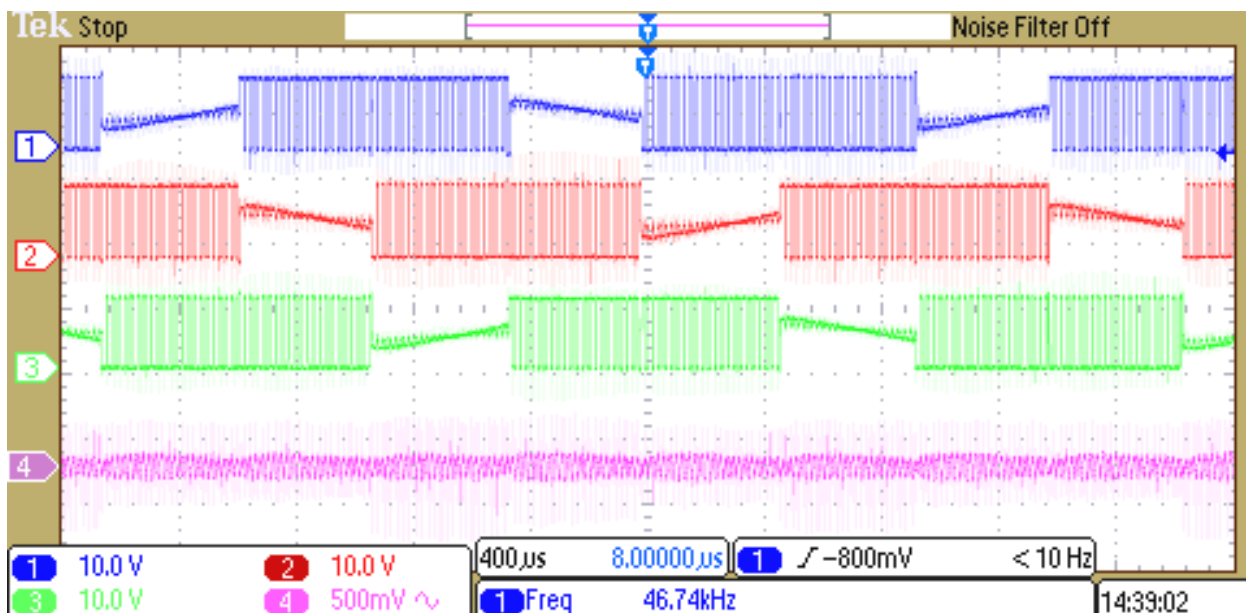
图 3-6 中给出了一个相上的电压。

图 3-6. 双极开关相电压（75% 占空比）



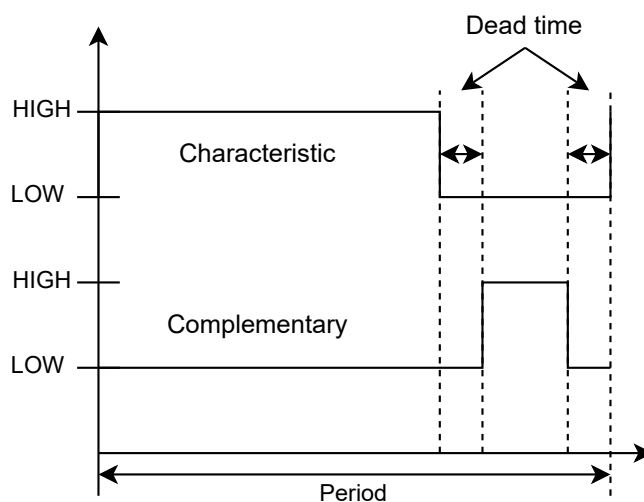
高频尖峰出现在特征状态和互补状态之间的转换处。这些可以在低电感电机中观察到，在换向区域附近，尖峰的幅值会变得更

图 3-7. 低电感电机上的 BEMF 噪声



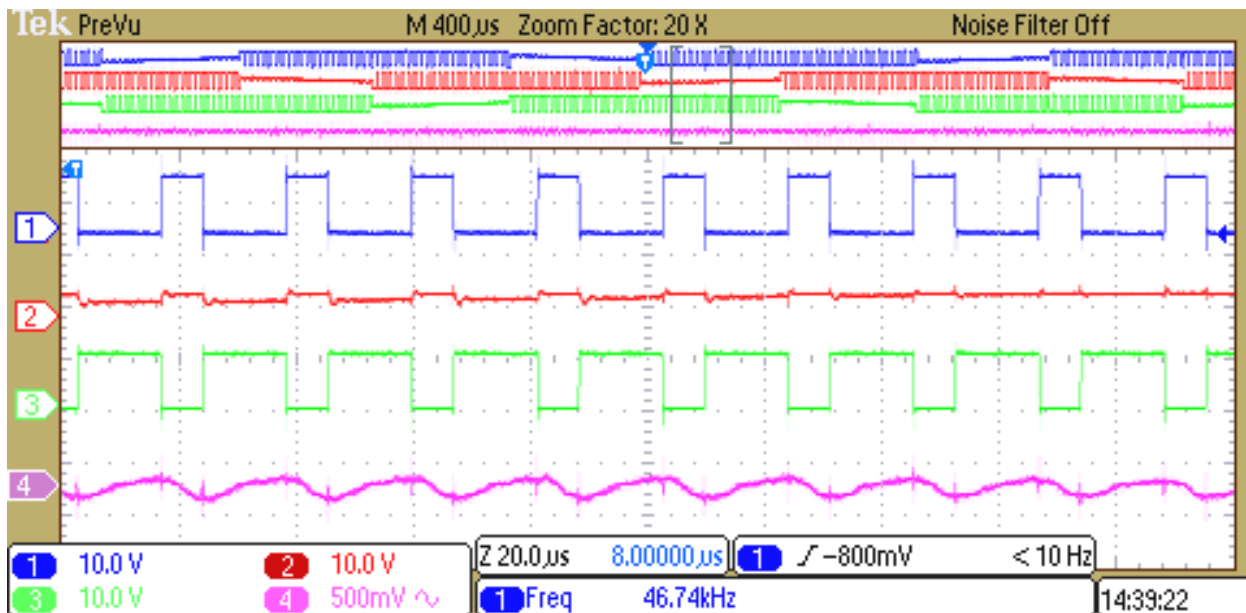
在图 3-7 中，前三个波形代表电机三相的 PWM 信号，而第四个波形代表通过晶体管检测到的电流。为避免发生直通，将在驱动器的命令信号之间添加死区。

图 3-8. 死区插入



添加了大约 42 ns 的死区（PWM 定时器的一个时钟周期），图 3-9 中给出了结果。不过，如果功率 MOSFET/IGBT 的栅极电容较高，则建议使用更长的死区。

图 3-9. 电机波形特写



在图 3-9 中，前三个波形代表三相的 PWM 信号，而第四个波形是通过晶体管检测到的电流。电流波形中的小尖峰对应于晶体管的开关，代表少量的直通。如果没有添加死区，电流会显著上升，晶体管会发热并烧毁。

开关噪声

与单极开关类似，双极开关也存在一个特定问题：在换向期间，当电机承受一定的负载时，将产生一个消磁序列，这会影响过零的检测并可能提供错误的输入，进而导致电机失去同步。

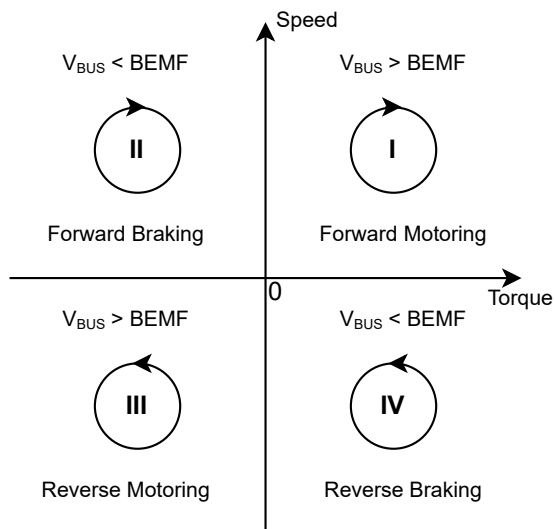
这极具挑战性，因为消磁序列几乎可以达到 30 电角度，这反过来会限制可以检测到过零的区域，甚至可能抑制它。

解决这种行为的方案是添加消隐时间，在此期间完全忽略过零。此参数将在控制算法中动态设置，并在整个 RPM 范围内自适应。

再生制动

无刷电机可以在四个不同的区域运行。这些区域称为“象限”，如图 3-10 中所述。

图 3-10. 四象限操作



在本应用笔记中，我们只关注第一象限和第二象限，电机可在其中任一象限运行，而无需控制算法的特定干预。

在第一象限，电机从总线汲取电流，处于自由运行状态或已承受负载。转子的测量速度等于或小于设定速度。

另一方面，在第二象限，由于系统惯性或其他外力，导致电机的设定速度低于测量速度。

双象限操作与占空比直接相关，占空比将施加特定的 RPM，如下面的公式所示。

$RPM = K_v \times U$ ，其中 K_v 是电机的一个参数， U 是施加到线圈的电压。

鉴于以下概念： RPM_{SET} 和 $RPM_{MEASURED}$ 提供关于电机当前 RPM 和任意给定时间的设定 RPM 的信息。

因此，有必要用以下公式表示 RPM 的差异：

$$\Delta RPM = RPM_{SET} - RPM_{MEASURED}$$

如果 ΔRPM 大于零，则电机处于第一象限，并从电源汲取电流。

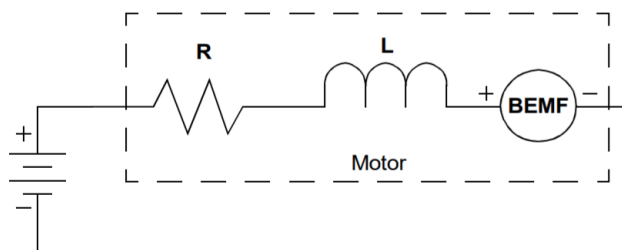
否则，如果 ΔRPM 小于零，则电机处于第二象限，并向电源输送电流。

这是电机行为的一个粗略近似表示，并未考虑摩擦损耗或负载等变量。

在第二象限，系统表现得像一个升压转换器，能够产生的最大电流取决于电机的参数，以及前面提及的 ΔRPM 和电池的 ESR。

电机的等效电路如下所示：

图 3-11. 电机的等效电路

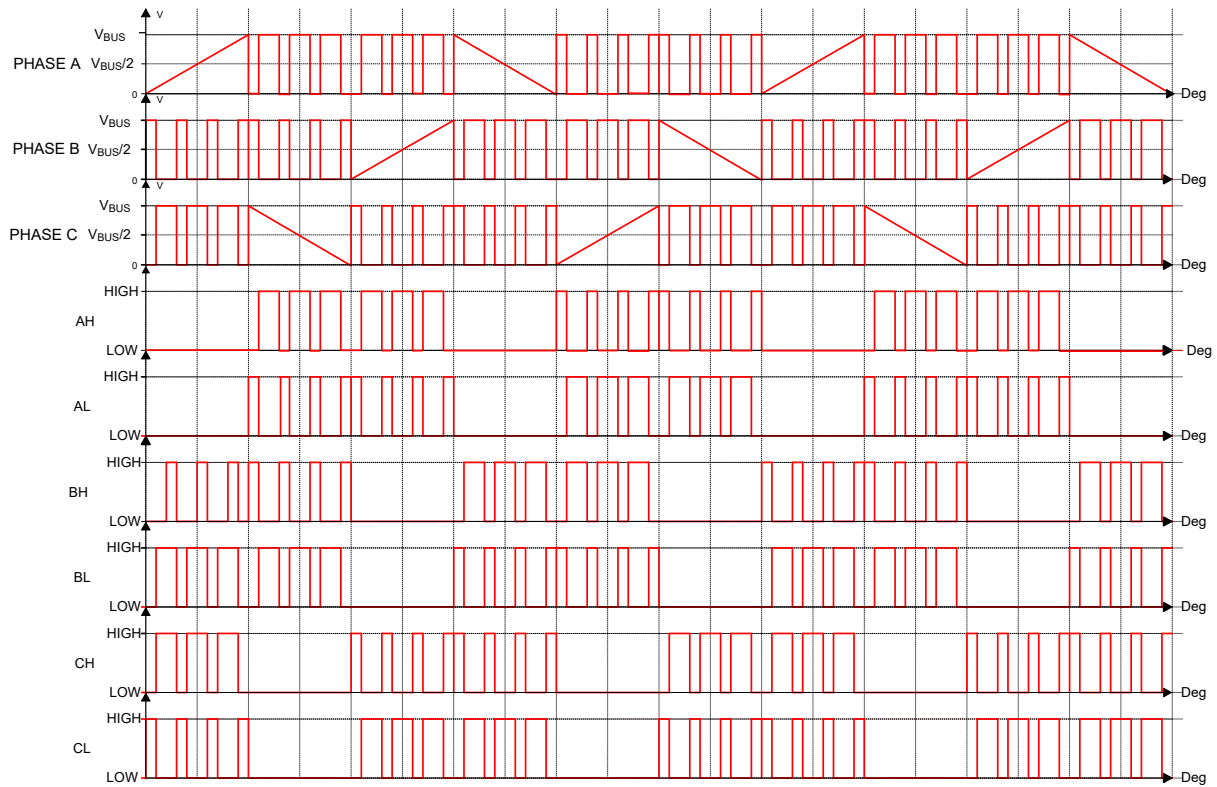


在整个 RPM 范围内，线圈电阻和电感保持不变，但 BEMF 与 RPM 直接成正比： $BEMF = RPM/K_v$ 。

BEMF 幅值等于电源电压乘以 PWM 占空比（对于双极操作，50% 占空比意味着 0V）。如果电机没有任何负载，BEMF 幅值将在满占空比时接近电源电压。

为了保护系统和电源受过压状况的影响，需要额外的电路来检测总线上的电压。

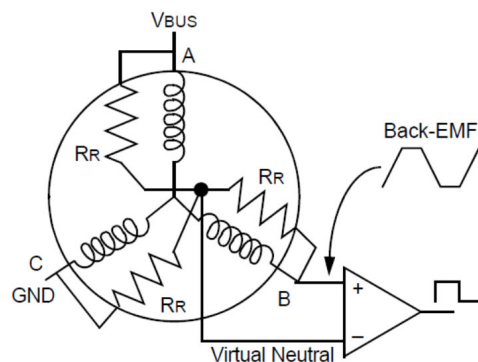
图 3-12. 双极开关的相电压和命令信号



3.5. 过零点检测

在此实现中，测量过零点的方法是使用比较器，其可变参考电压由所有相电压之和组成。

图 3-13. 使用比较器进行过零点检测

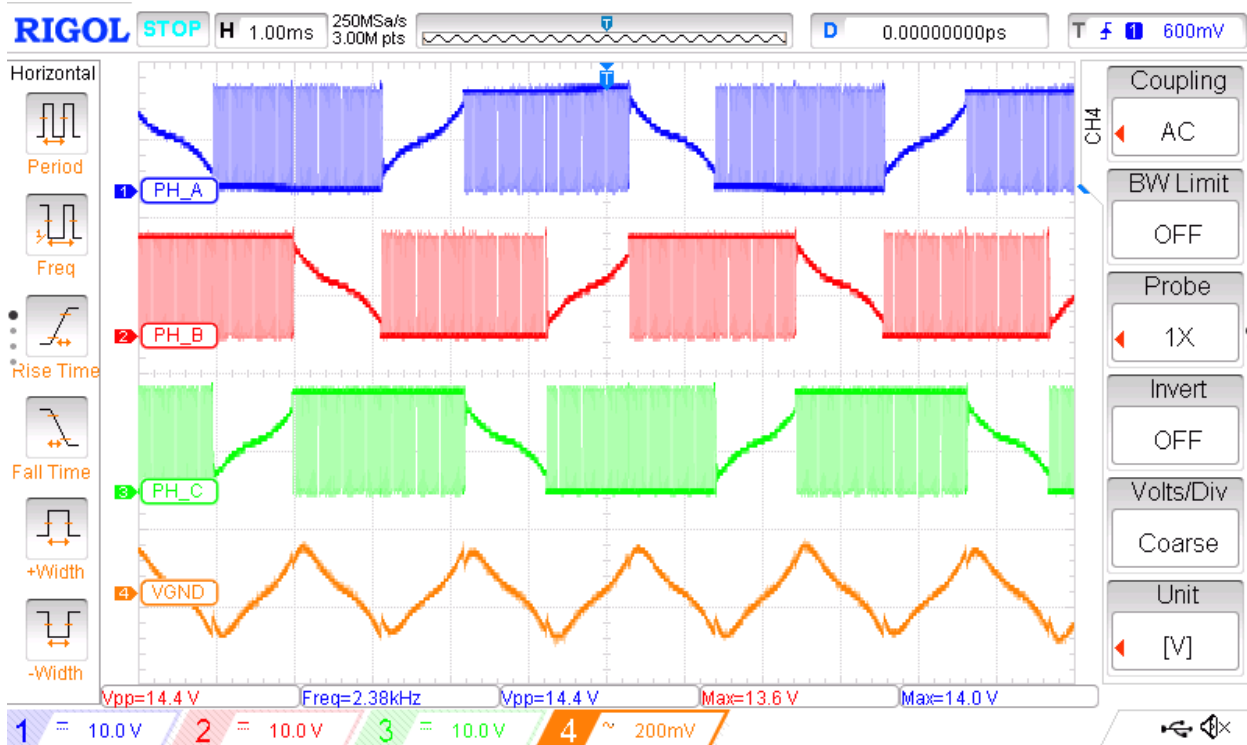


中性点电压可以表示为 $V_n = \frac{BEMF A + BEMF B + BEMF C}{3}$ ，其中 V_n 是虚拟中性点电压， $BEMF A$ 是 A 相的 BEMF 电压， $BEMF B$ 是 B 相的 BEMF， $BEMF C$ 是 C 相的电压。

由于线圈阻抗不对称，虚拟中性点信号不会保持恒定，而是会根据驱动周期变化。为了避免换向点发生任何相移，不能对虚拟中性点进行过度滤波。只需使用低 RC 时间常数的低通滤波器抑制高频噪声。

图 3-14 中给出了虚拟地的不规则性，其中前三个波形代表三相各自的 PWM 信号，第四个波形是虚拟地。

图 3-14. 虚拟地波形



4. 固件实现

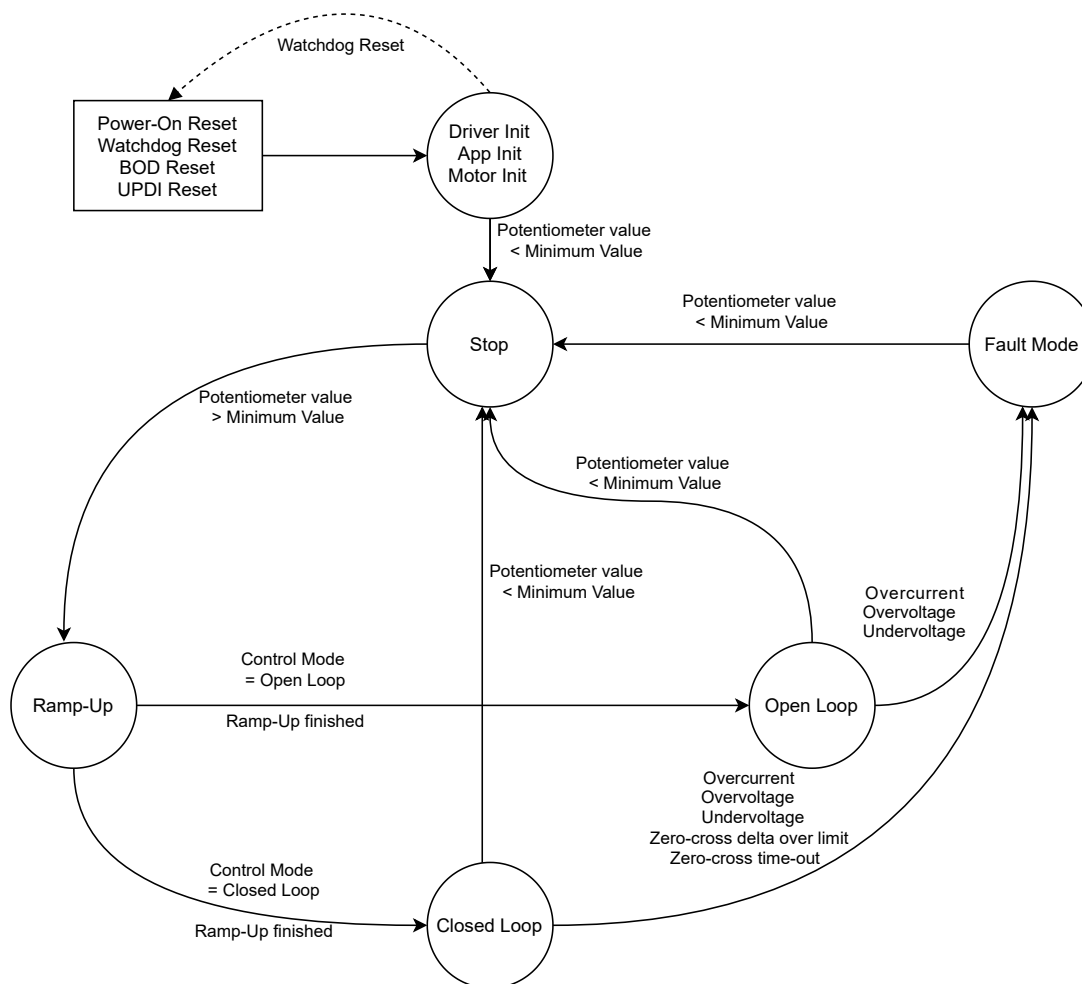
本节介绍了如何使用内部 AVR 外设来实现特定任务，并描述了固件如何用作电机控制系统。

4.1. 系统概述

系统状态机

固件基于控制系统行为的状态机。它提供故障处理和复位信息，并通过 UART 发送的错误消息使能调试功能。

图 4-1. 系统控制状态机



系统上电时，系统控制状态会设置为驱动程序初始化、应用程序初始化状态。这是系统在任何复位后的默认状态。为了确定复位原因，系统会输出消息以通知用户关于特定事件的信息。事先检查电位器值以确保系统不会立即启动（例如，如果系统在电机全速运行时发生外部复位，它不会立即启动，从而保护电机和功率级）。如果条件不满足，看门狗将触发并复位单片机。

之后，电机进入停止状态。如果电位器值增至高于设定阈值，则转子将被对齐，系统状态将设置为斜升。在这里，电机将加速到软件中定义的速度，而无需了解转子的位置。

完成斜升序列后，系统进入定义控制状态，可以是开环或闭环。

在开环状态下，电机运行时单片机没有关于其转子位置的任何信息。可以通过电位器调节电机速度。如果其值小于设定阈值，电机停止，系统进入停止状态。如果检测到过流，系统会进入错误处理状态。

在闭环状态下，电机的运行基于单片机接收到的关于转子位置的反馈。电机的速度保持恒定，只能通过电位器调节占空比。如果其值小于设定阈值，电机停止，系统进入停止状态。

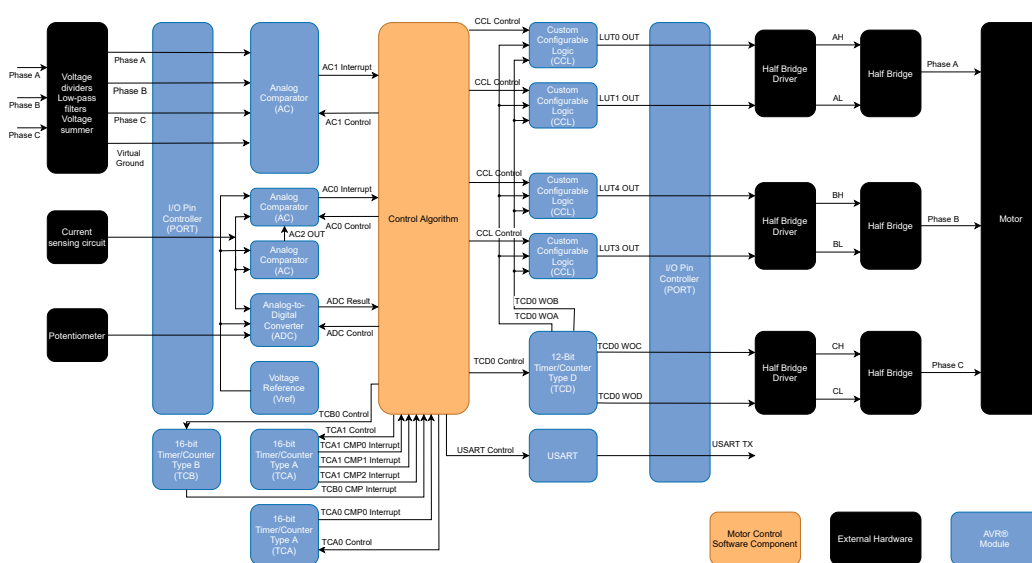
除过流以外，过零增量超限、过零超时、欠压和过压也会导致系统进入错误处理状态。

如果系统进入错误状态，则惟一的复位方法是将电位器值降至设定阈值以下。随后系统会返回到初始停止状态。

已使用器件模块和信号概述

图 4-2 给出了硬件、单片机和软件模块之间的关系。术语“控制”用于表示对外设寄存器的任何修改。

图 4-2. 系统模块概述



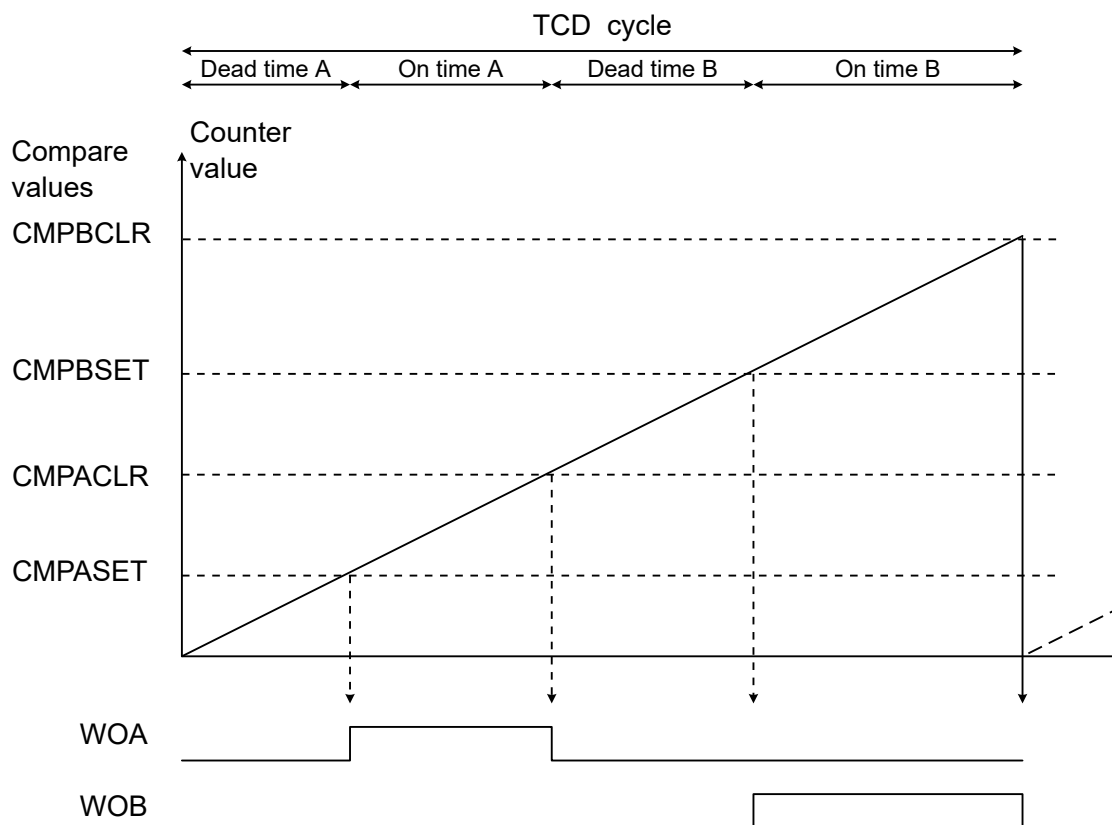
此实现利用上述双极开关，并通过使用内部比较器检测过零。

4.2. PWM 发生

D 型定时器/计数器 (TCD) 外设用于生成控制半桥驱动器输入的 PWM 信号。

定时器配置为单斜坡模式，图 4-3 中给出了其功能。

图 4-3. TCD 单斜坡模式波形



在单斜坡模式下，TCD 周期为：
$$T_{TCD_cycle} = \frac{(CMPBCLR + 1)}{f_{CLK_TCD_CNT}}$$

例如，使用 24 MHz 的输入时钟，1:1 的预分频比，以及 512 的 CMPBCLR 值，将得到 21.375 μ s 的周期，或大约 46.783 kHz 的频率。

在定时器占空比的外部更改期间通过软件设置死区。

对于此应用，工作区域在 50% 占空比以上以在第一象限和第二象限中运行。

PWM 信号布线

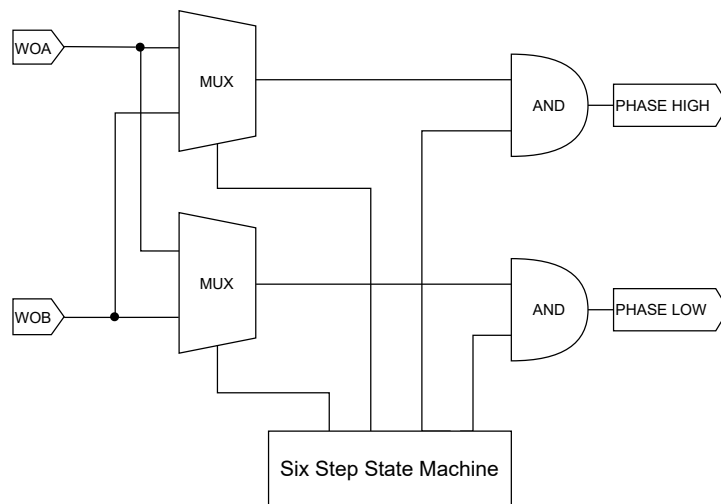
对于六步梯形控制，三个半桥驱动器必须由单片机控制。TCD 只有四个波形输出，只允许直接控制两个半桥驱动器。可借助定制可配置逻辑（Custom Configurable Logic, CCL）创建另外两相。

TCD 外设的 WOC 和 WOD 输出可用于单片机上的物理引脚，并用于驱动 C 相。

TCD 外设的 WOA 和 WOB 输出用于控制另外两相。借助 CCL 逻辑，WOA 和 WOB 信号可用于两对不同的引脚。

根据系统的行为，通过 TRUTH 寄存器配置一个相位的等效逻辑：

图 4-4. 等效逻辑布线原理图



CCL 逻辑实现了 PWM 信号到半桥驱动器的智能布线，仅使用一对互补 PWM（WOA 和 WOB）就能控制两个电机相位，如下面的用例所述。

如果相位的特征状态为低电平，则 WOA 连接到下桥臂晶体管，而 WOB 连接到上桥臂晶体管。

如果相位的特征状态为高电平，则 WOA 连接到上桥臂晶体管，而 WOB 连接到下桥臂晶体管。

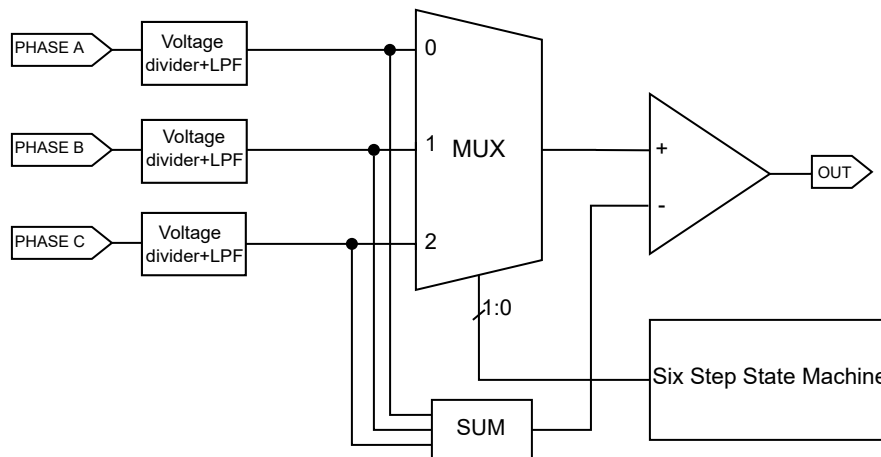
4.3. 过零检测

必须以高精度检测过零点以确保开关点精确。每相上的电压幅值通过分压器降低，然后通过低通 RC 滤波器进行滤波。

使用一个内部比较器（AC1）来比较滤波后的电压与电机的虚拟中性点，该中性点由所有相电压之和提供。

此处使用的 AC1 比较器具有多个反相和同相输入，可借助内置多路开关进行选择。此功能通过状态机切换 AC1 输入功能，可使用一个交流电源为全部三相供电，从而减少所需的外设数量。图 4-5 给出了用于检测过零点的等效电路。

图 4-5. 过零检测等效电路



4.4. 过零滤波

在某些情况下，由于噪声或其他因素（如电机的特殊性），过零点的检测时间可能早于或晚于理想情况。

可以使用一种滤波方法来避免导致电机失锁的异常行为。

选择该滤波器时，需要遵循几个简单规则：

- 滤波器的值不能过于激进，以免在速度变化适中时影响电机的响应时间，也不能过于宽松，否则会违背其设计初衷
- 该滤波器应避免大量计算，否则可能导致获得预期结果耗时过长，进而使换向点延迟
- 需要最短的直通延时才能获得最佳结果

基于上述要求，选择采用移动平均低通滤波器形式的一阶 IIR 滤波器，其计算公式为

$$y(n) = \frac{y(n-1) \times (a-1)}{a} + \frac{x(n)}{a}, \text{ 其中 } y \text{ 是滤波器的输出, } x \text{ 是输入, } a \text{ 是滤波器系数。}$$

```
timerValue = (uint32_t)((uint32_t)previous_zero_cross_time * (MOTOR_DIVISION_FACTOR - 1) +
(uint32_t)current_zero_cross_time) / MOTOR_DIVISION_FACTOR;
```

理想情况下，为了提高计算速度，滤波器系数（MOTOR_DIVISION_FACTOR）必须为 2 的幂，因为它可以使用特定内核指令的移位来实现。

输入和输出值均为无符号 16 位值，因此，为了防止在 $y(n-1)$ 的乘法以及总和计算过程中发生溢出，所有运算均通过强制转换为 32 位无符号值来执行。

在本应用范围内，可认为滤波器系数值 4 最为合适。

4.5. 30 度定时

30 度等待时间定义为换向点与下一个过零点之间的时间。在梯形控制中，当找到过零时，系统必须等待 30 度电角度才能进行换向。

任何对该时间的更改称为超前或滞后，即换向操作早于或晚于预定时间发生。

超前的目的是在加速期间获得更大的转矩，同样，在滞后期间制动力会增加。

可通过从新计算的 30 度时间中减去该值来实现固定超前值，以此补偿任何直通延时，并在换相点处增加超前时间。

这可通过以下代码行实现：`tempValue = timerValue - MOTOR_ADVANCE_TIME;`

4.6. BEMF 换向噪声消隐

在梯形控制中，当系统移动到下一步时，新步特定悬空相的 BEMF 会受到换向噪声的影响。因此，必须避免捕获该序列，因为它可能触发错误的过零点。

这称为消隐时间，必须根据电机的速度进行设置。例如，在高速场景中，如果消隐时间适用于低速情况，则该消隐时间可能过长，导致无法捕捉过零点。该值使用 30 度定时中采用的前一个值的一半计算得出。

4.7. 处理时间相关事件

电机的驱动高度依赖时间。该系统中使用 TCA 外设的两个实例和 TCB 外设的一个实例来处理时间相关事件。

TCA 外设的第一个实例（TCA0）在系统处于斜升/开环模式时控制电机的速度。它按照 CMP0 比较寄存器中的值确定的间隔触发周期性中断。

通过减少 CMP0 寄存器中的值，中断周期减少，电机加速。如果寄存器值停止减少，电机将以固定周期换向，控制模式切换到开环模式。

当达到中断时，算法移动到下一个梯形步，然后触发它。

这可通过以下代码行实现：

```
tmr_ramp_up_compare_value -= MOTOR_TIMER_RAMP_UP_DECREMENT;  
TCA0_setCMP0Value(tmr_ramp_up_compare_value);
```

```
#define MOTOR_TIMER_RAMP_UP_DECREMENT 800
```

TCA 外设的第二个实例（TCA1）用于在每个步中驱动/处理时间相关的电机事件。

在一个梯形步中，计数器有四个用途：

- 处理消隐时间
- 对从换向点到过零点的时间进行计数
- 对从过零点到新换向点的新计算时间进行计数
- 处理失速检测（过零超时）

计数器有三个比较通道，每个通道都配置为触发特定的中断。

操作方法如下所述：

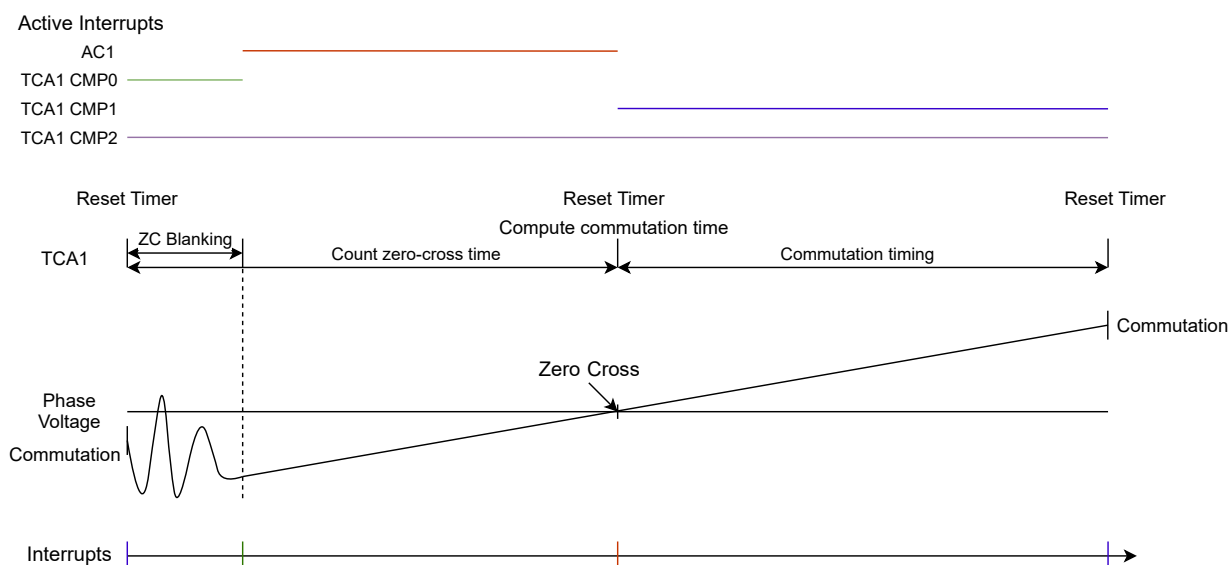
在新的换向后，CNT 寄存器值清零，并允许 CMP0 中断。

当达到 CMP0 中断时，允许 AC1 中断，并禁止 CMP0 中断。

如果在 CMP2 时间间隔内找到过零点，则定时器停止，CNT 寄存器值经过过零滤波器处理后设置到 CMP1 寄存器中。计数器寄存器再次清零，并使能 CMP1 中断。

当定时器达到 CMP1 值时，它会触发中断，换向至下一步，禁止 CMP1 中断，然后重复循环。图 4-6 中给出了此模块功能的详细时域表示。

图 4-6. TCA1 时域表示



TCB0 定时器/计数器作为一个非常基本的调度程序来运行非时间关键型任务。它为比较寄存器使用一个固定值，该值触发中断并将主循环中检查的 `SCHEDULER_mainLoopFlag` 置 1。

4.8. 过流保护

为了保护电机、功率级和电源，需要一种过流保护方法。这种保护通过电流检测电阻实现，它会发出一个具有 $V_{DD}/2$ 直流偏移的模拟信号，使电压可以在两个方向上摆动。因此，在 $V_{DD}/2$ 值处，系统消耗的电流为零。

为提高响应速度，使用两个窗口模式的内部模拟比较器来检测过流。

窗口的上限和下限是使用两个 DAC 产生的，每个比较器一个。由此电流窗口产生的中断将强制系统进入故障状态。

4.9. 停转检测

此实现采用三种失速检测方法：

- 使用 TCA1 CMP2 中断的过零超时
- 使用过零增量限值。这意味着，如果电机在检测到过零的时间上有任何突然变化，基于前一次检测，则认为电机已失速。
- 最可能情况是，在失速期间，通过绕组的电流将大幅上升并超过过流限值

失速检测最敏感的部分是在启动时。当电机在开环模式下加速并切换为闭环时，系统可能会得到过零有效的虚假响应，尽管 BEMF 是由驱动和非驱动线圈之间的磁耦合产生的，而不是由移动磁铁在线圈中感应的电压。

4.10. RPM 测量

RPM 测量使用过零的滤波时间完成。这意味着 RPM 仅使用梯形步的一半计算（从电机换向到找到过零点的时间点）。因此，电机 RPM 检测的精度降低。

随后使用过零定时器值计算 RPM，将其乘以二得到梯形步的持续时间，再乘以六得到电气周期。在此之后，只需将值代入公式 $RPM = 120 / (T_e \times P)$ ，其中 T_e 是电频率， P 是电机特有的极对数。

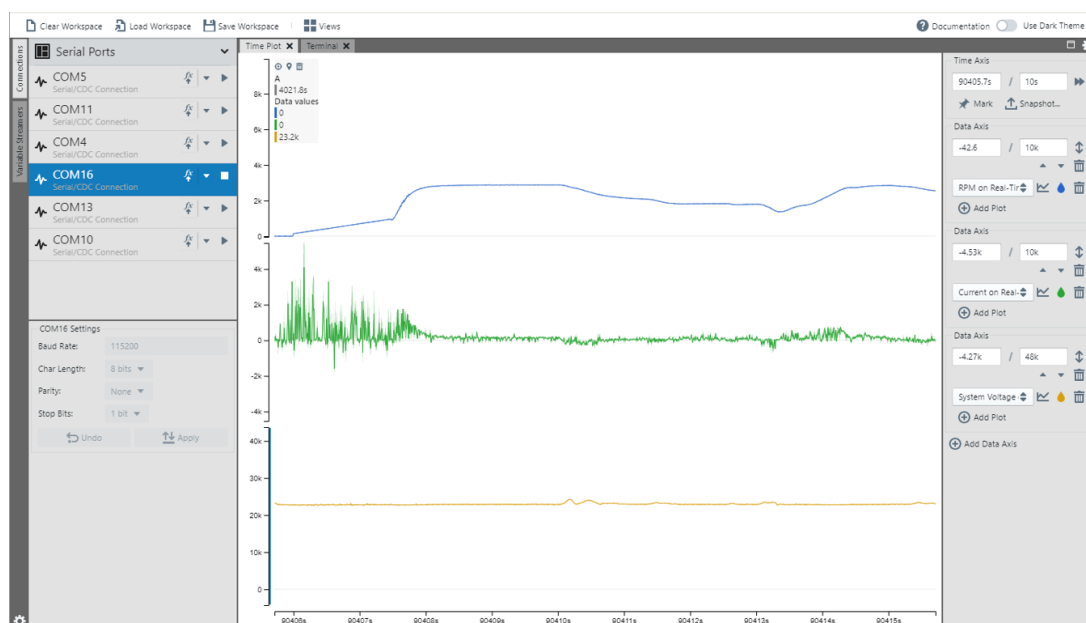
为补偿此效应，对 TCA1 定时器/计数器预分频器进行了调整，使其在低速时不会溢出，同时获得尽可能高的值。

通过输出过零定时器值并更改预分频比以在最小转速下获得可能的最大值（即换向和过零信号之间的最大时间间隔），可完成确定最佳预分频比的过程。

4.11. 调试

MPLAB®数据可视化器用于查看实时数据和调试目的。

图 4-7. MPLAB®数据可视化器示例



将显示以下变量：

- 电机速度，每分钟转数（RPM）
- 电机电流（mA）
- 系统电压（mV）

此外，可以在 MPLAB 数据可视化器的终端部分中使能和查看显示任何故障或状态变化的调试消息。以下代码行包含在项目应用区域的 app.c 文件中。强烈建议每次仅使能单个功能，即调试消息或数据可视化器。调试消息默认禁止。要使能这些消息，请注释掉第 31 行和第 33 行，并取消注释第 30 行。对代码的任何更改都需要重新编译并重新上传项目。

图 4-8. 调试选项

```

30  // #define COMM_debugPrint(x) printf(x) // comment to disable debug messages
31  #define COMM_debugPrint(x) // uncomment to disable debug messages
32
33  #define ENABLE_DATA_VISUALIZER () // comment to disable data visualizer

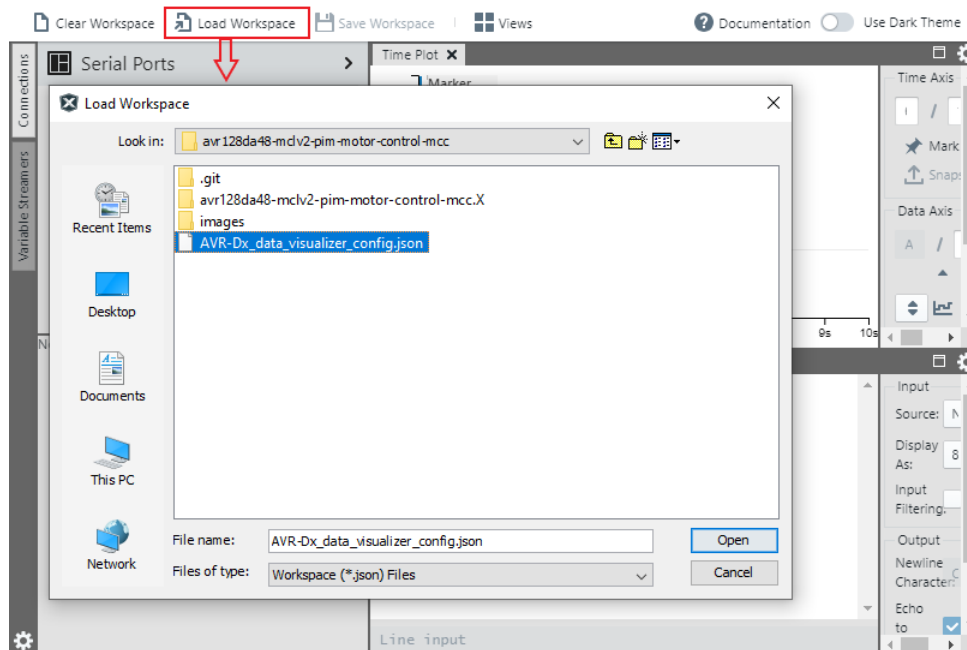
```

配置

1. 使用 Mini-USB 线缆将 MCLV-2 开发板连接到 PC。
2. 确保已安装 MPLAB 数据可视化器插件。

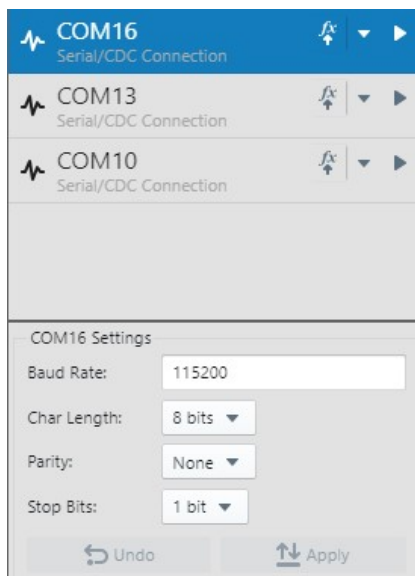
- a. 转到 **Tools → Plug-ins → Available**（工具 → 插件 → 可用）插件并搜索 MPLAB 数据可视化器。
3. 启动 MPLAB 数据可视化器，然后从顶部栏单击 **Load Workspace**（加载工作区）。选择资源库根目录中的 **.json** 文件。

图 4-9. 加载工作区



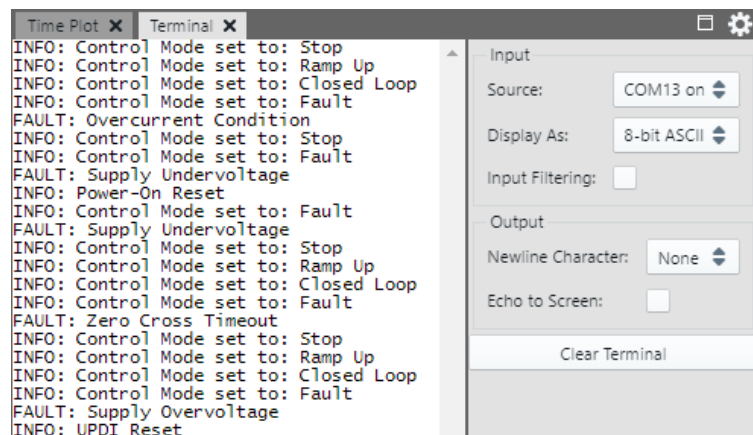
4. 在左侧的 **Connections**（连接）选项卡上，选择 **COM port**（COM 端口）并将波特率设置为 115200。按图 4-10 所示保持其余配置：

图 4-10. COM 端口配置



5. 按下所选 COM 端口的 **Start streaming**（开始传输）按钮。要在 Time Plot（时间曲线图）上查看轨迹，需要激活 **Show Live Data**（显示实时数据）模式。
 - 如果已使能调试消息（请参见 app.c 源文件宏），则可在 **Terminal**（终端）选项卡中查看这些消息，其中的 **Display As**（显示为）模式设置为 8-bit ASCII（8 位 ASCII）

图 4-11. 各种状态和故障的调试消息



5. 接插模块（PIM）板说明

AVR DX PIM 有两种版本：

- 内部运算放大器版本：基于包含内部运放的 AVR128DB48 单片机
- 外部运算放大器版本：基于 AVR128DA48，它代表普通版本并使用外部 MCLV-2 运放

图 5-1. 带内部运放的 AVR® DB PIM

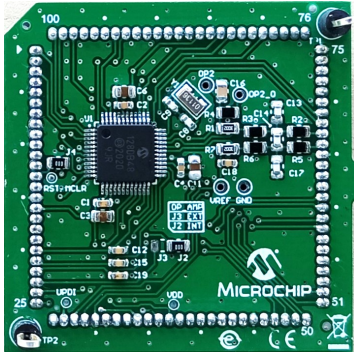
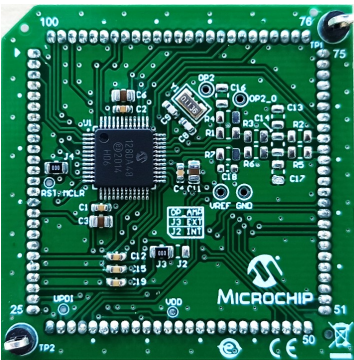


图 5-2. 带外部运放的 AVR® DA PIM



内部运放的使用主要依赖于包含运放模块的单片机（如 AVR DB 系列），无需外部运放。放大后的电流随后送入一个具有软件可调限制的比较器窗口，通过更改 PIM 上的单个选择跳线，让用户能够在内部与外部参考之间进行选择。

需要一个特殊的配置板才能获取来自电流检测电阻的信号和电机的反电动势。它与 PIM 采用相同的封装。只有在使用内部运放电流参考时才需要该配置板。不过，如果 PIM 配置为使用外部运算放大器运行，它也可与提供的 MCLV-2 外部运放配置板一起工作。

5.1. 引脚映射

表 5-1 提供 100 引脚 PIM 与 48 引脚器件之间的映射。

表 5-1. 引脚映射表（按 PIM 引脚编号排序）

PIM 引脚编号	信号名称	引脚分配说明	AVR® DX 引脚
1	DBG_LED2	调试 LED 2	PA5
2	V _{DD}	数字电源	VDD
3	PWM1H3	PWM 输出——3H	PA6
13	确保复位的 MCLR	器件主复位	PF6
15	V _{SS}	数字电源	GND

表 5-1. 引脚映射表（按 PIM 引脚编号排序）（续）

PIM 引脚编号	信号名称	引脚分配说明	AVR® DX 引脚
16	V _{DD}	数字电源	VDD
18	FAULT	直流总线电流故障（逻辑低电平有效）	PC2
19	TX	UART 发送	N/A
20	PIM_V_M3	电压反馈信号	PD4
21	PIM_V_M2	电压反馈信号	PD6
22	PIM_V_M1	电压反馈信号	PD3
23	PIM_IMOTOR_SUM	直流总线电流信号	PD2
24	PIM_IMOTOR2	参考电流信号	PE0
25	PIM_IMOTOR1	参考电流信号	PF2
26	PGC	器件编程时钟线	N/A
27	PGD	器件编程数据线	UPDI
28	V _{REF}	参考电压（AV _{DD} 电压的一半）	PD7
29	PIM_REC_NEUTR	重构的电机中性点电压	PD5
30	AV _{DD}	模拟电源	AVDD
31	AV _{SS}	模拟电源	GND
32	PIM_POT	电位器信号	PD0
34	PIM_GEN2	通用 I/O	PA4
35	PIM_VBUS	直流总线电压（已按比例缩小）	PD1
36	V _{SS}	数字电源	GND
37	V _{DD}	数字电源	VDD
41	PIM_MONITOR_1	霍尔传感器/电流检测/电压反馈信号	N/A
42	PIM_MONITOR_2	霍尔传感器/电流检测/电压反馈信号	N/A
43	PIM_MONITOR_3	霍尔传感器/电流检测/电压反馈信号	N/A
45	V _{SS}	数字电源	GND
46	V _{DD}	数字电源	VDD
47	HALLB	霍尔传感器/QEI 输入	PB5
48	HALLC	霍尔传感器/QEI 输入	PB4
49	RX	UART 接收	PA0
50	TX	UART 发送	PA1
51	USB_TX	UART 发送（直接连接到 U7）	PB1
52	USB_RX	UART 接收（直接连接到 U7）	PB0
58	PIM_FLT_OUT2	通用 I/O	PC6
59	PIM_FLT_OUT1	通用 I/O	PC5
60	DBG_LED1	调试 LED 1	PC0
61	HOME	QEI 的归位信号	PC7
62	V _{DD}	数字电源	VDD
63	OSC1/CLKO	晶振输入	N/A
64	OSC2/CLKI	晶振输出	N/A
65	V _{SS}	数字电源	GND
66	PIM_IBUS+	总线电流检测信号	N/A
67	PIM_IBUS-	总线电流检测信号	N/A
68	LIN_CS	LIN 片选信号	PA2
69	LIN_FAULT	LIN 故障信号	PC1
70	RX	UART 接收	N/A
72	USB_RX	UART 接收（直接连接到 U7）	N/A

表 5-1. 引脚映射表（按 PIM 引脚编号排序）（续）

PIM 引脚编号	信号名称	引脚分配说明	AVR® DX 引脚
73	PIM_IB+	IMOTOR1 电流检测信号	N/A
74	PIM_IA+	IMOTOR2 电流检测信号	N/A
75	V _{SS}	数字电源	GND
76	USB_TX	UART 发送（直接连接到 U7）	N/A
77	CAN_TX	CAN 发送	N/A
78	CAN_RX	CAN 接收	N/A
80	HALLA	霍尔传感器/QEI 输入	PB2
82	PIM_GEN1	通用 I/O	N/A
83	BTN_1	按钮 S2 输入	PF5
84	BTN_2	按钮 S3 输入	PF4
86	V _{DD}	数字电源	VDD
87	CAN_RX	CAN 接收	N/A
88	CAN_TX	CAN 发送	N/A
93	PWM1L1	PWM 输出——1L	PC3
94	PWM1H1	PWM 输出——1H	PA3
98	PWM1L2	PWM 输出——2L	PF3
99	PWM1H2	PWM 输出——2H	PB3
100	PWM1L3	PWM 输出——3L	PA7

注：

1. 数字电源（V_{DD}）引脚连接到 PIM。
2. 数字地（V_{SS}）和模拟地（AV_{SS}）引脚连接到 PIM。

表 5-2 提供 48 引脚器件与 100 引脚 PIM 之间的映射。

表 5-2. 引脚分配表（按 AVR® DX 引脚名称排序）

PIM 引脚编号	引脚名称	引脚分配说明	AVR® DX 引脚
30	AV _{DD}	模拟电源	AVDD
31	AV _{SS}	模拟电源	GND
15	V _{SS}	N/A	GND
36	V _{SS}	N/A	GND
45	V _{SS}	N/A	GND
65	V _{SS}	N/A	GND
75	V _{SS}	N/A	GND
19	TX	UART 发送	N/A
26	PGC	器件编程时钟线	N/A
41	PIM_MONITOR_1	霍尔传感器/电流检测/电压反馈信号	N/A
42	PIM_MONITOR_2	霍尔传感器/电流检测/电压反馈信号	N/A
43	PIM_MONITOR_3	霍尔传感器/电流检测/电压反馈信号	N/A
63	OSC1/CLKO	晶振输入	N/A
64	OSC2/CLKI	晶振输出	N/A
66	PIM_IBUS+	总线电流检测信号	N/A
67	PIM_IBUS-	总线电流检测信号	N/A
70	RX	UART 接收	N/A
72	USB_RX	UART 接收（直接连接到 U7）	N/A
73	PIM_IB+	IMOTOR1 电流检测信号	N/A

表 5-2. 引脚分配表（按 AVR® DX 引脚名称排序）（续）

PIM 引脚编号	引脚名称	引脚分配说明	AVR® DX 引脚
74	PIM_IA+	IMOTOR2 电流检测信号	N/A
76	USB_TX	UART 发送（直接连接到 U7）	N/A
77	CAN_TX	CAN 发送	N/A
78	CAN_RX	CAN 接收	N/A
82	PIM_GEN1	通用 I/O	N/A
87	CAN_RX	CAN 接收	N/A
88	CAN_TX	CAN 发送	N/A
49	RX	UART 接收	PA0
50	TX	UART 发送	PA1
68	LIN_CS	LIN 片选信号	PA2
94	PWM1H1	PWM 输出——1H	PA3
34	PIM_GEN2	通用 I/O	PA4
1	DBG_LED2	调试 LED 2	PA5
3	PWM1H3	PWM 输出——3H	PA6
100	PWM1L3	PWM 输出——3L	PA7
52	USB_RX	UART 接收（直接连接到 U7）	PB0
51	USB_TX	UART 发送（直接连接到 U7）	PB1
80	HALLA	霍尔传感器/QEI 输入	PB2
99	PWM1H2	PWM 输出——2H	PB3
48	HALLC	霍尔传感器/QEI 输入	PB4
47	HALLB	霍尔传感器/QEI 输入	PB5
60	DBG_LED1	调试 LED 1	PC0
69	LIN_FAULT	LIN 故障信号	PC1
18	FAULT	直流总线电流故障（逻辑低电平有效）	PC2
93	PWM1L1	PWM 输出——1L	PC3
59	PIM_FLT_OUT1	通用 I/O	PC5
58	PIM_FLT_OUT2	通用 I/O	PC6
61	HOME	QEI 的归位信号	PC7
32	PIM_POT	电位器信号	PD0
35	PIM_VBUS	直流总线电压（已按比例缩小）	PD1
23	PIM_IMOTOR_SUM	直流总线电流信号	PD2
22	PIM_V_M1	电压反馈信号	PD3
20	PIM_V_M3	电压反馈信号	PD4
29	PIM_REC_NEUTR	重构的电机中性点电压	PD5
21	PIM_V_M2	电压反馈信号	PD6
28	V _{REF}	参考电压（AV _{DD} 电压的一半）	PD7
24	PIM_IMOTOR2	参考电流信号	PE0
25	PIM_IMOTOR1	参考电流信号	PF2
98	PWM1L2	PWM 输出——2L	PF3
84	BTN_2	按钮 S3 输入	PF4
83	BTN_1	按钮 S2 输入	PF5
13	确保复位的 MCLR	器件主复位	PF6
27	PGD	器件编程数据线	UPDI
2	V _{DD}	N/A	VDD
16	V _{DD}	N/A	VDD

表 5-2. 引脚分配表（按 AVR® DX 引脚名称排序）（续）

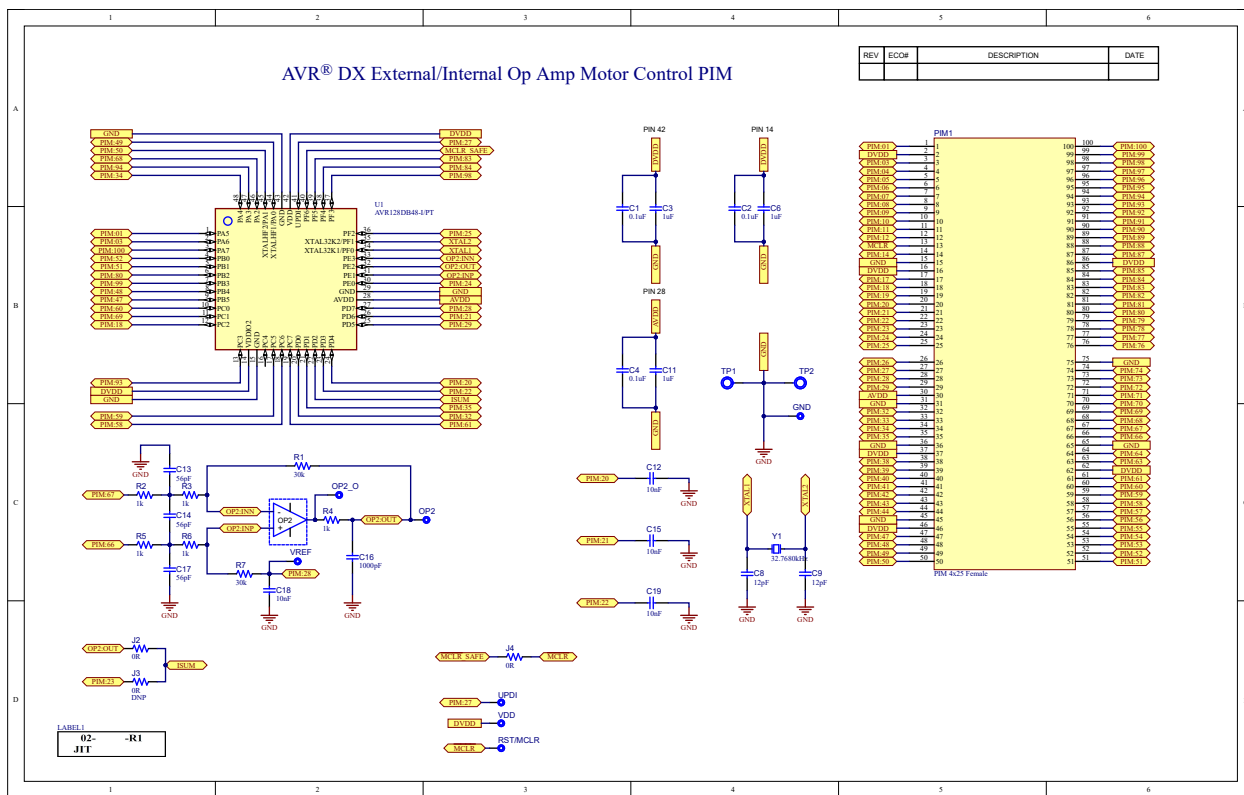
PIM 引脚编号	引脚名称	引脚分配说明	AVR® DX 引脚
37	V _{DD}	N/A	VDD
46	V _{DD}	N/A	VDD
62	V _{DD}	N/A	VDD
86	V _{DD}	N/A	VDD

注:

1. 数字电源 (V_{DD}) 引脚连接到 PIM。
2. 数字地 (V_{SS}) 和模拟地 (AV_{SS}) 引脚连接到 PIM。

5.2. PIM 原理图

图 5-3. PIM 原理图

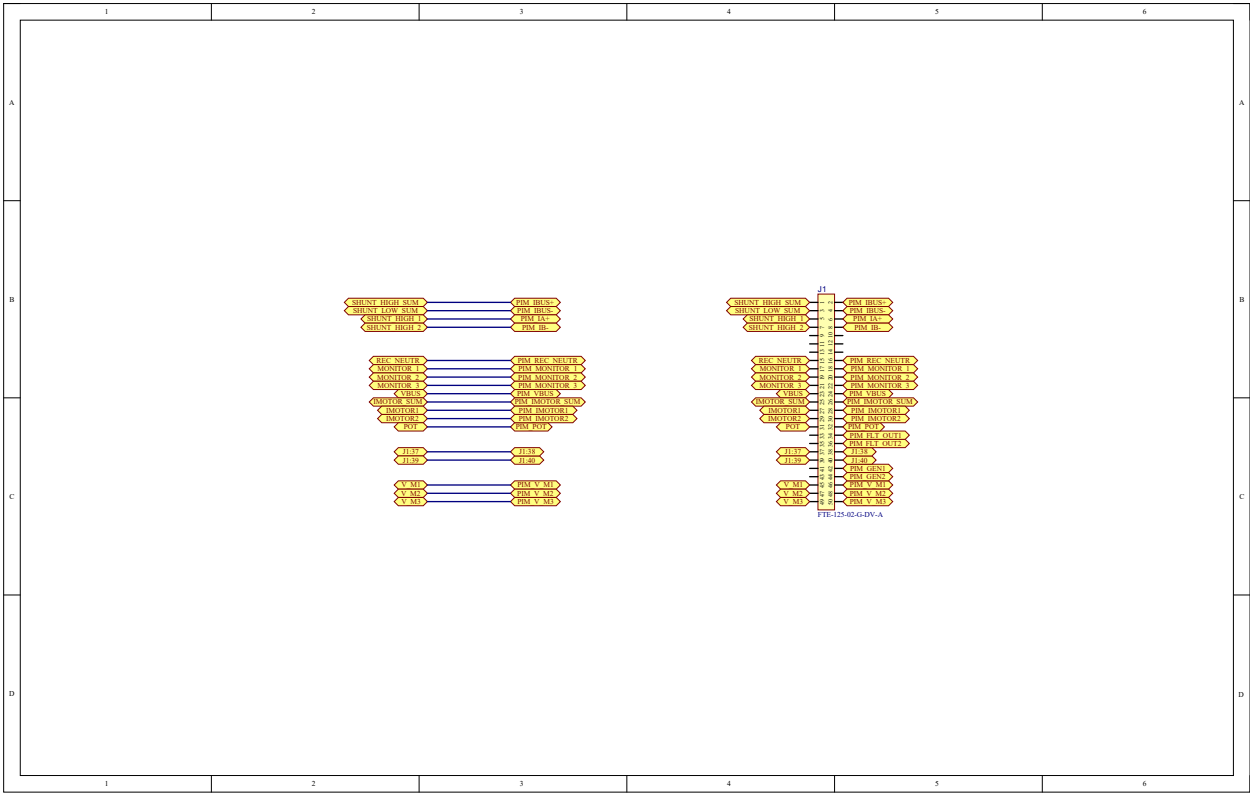


注:

1. 跳线 J2 和 J3 用于在外部和内部运放之间切换。
2. 跳线 J4 可以替换为电阻，以保护单片机免受意外 HV 编程模式的影响。

5.3. 配置板原理图

图 5-4. PIM 原理图

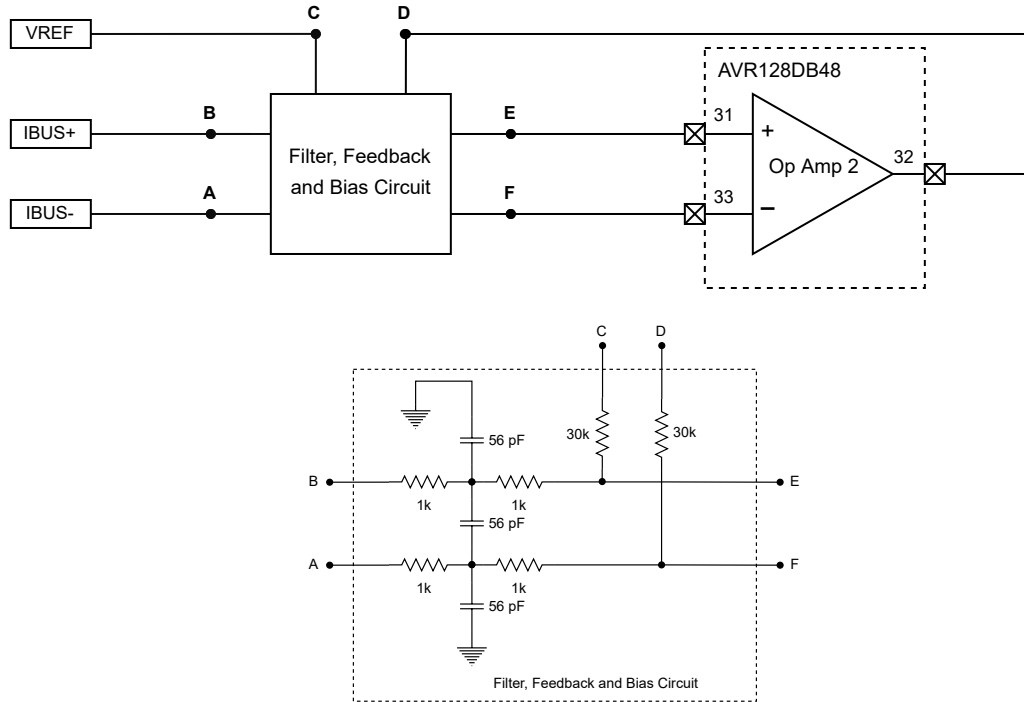


5.4. 模拟功能

表 5-3. 内部运放参数

运放编号	模拟功能	无源组件	设计公式
2	低通滤波器	R2、R3、R5、R6、C13、C14 和 C17	$R2 = R3 = R5 = R6 = R$
	参考电压偏置	R1 和 R7	$C13 = C17 = C$
	差分放大器输入	R2、R3、R5 和 R6	$R1 = R7$
	差分放大器反馈	R1	$\text{Common - mode } f_{-3\text{ dB}} \cong \frac{1}{2\pi RC}$ $\text{Differential - mode } f_{-3\text{ dB}} \cong \frac{1}{2\pi(2R)\left(\frac{C}{2} + C14\right)}$ $\text{Differential amplifier gain} \cong \frac{R7}{2R}$

图 5-5. 内部运放配置

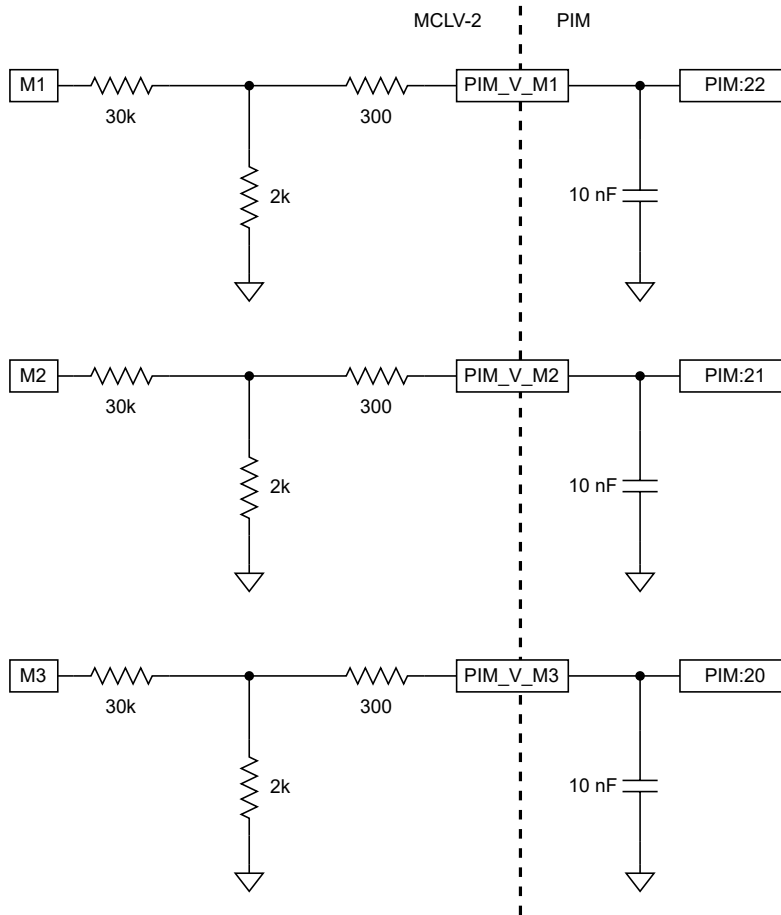


$$\text{Differential amplifier gain} = \frac{30k\Omega}{2 * 1k\Omega} = 15$$

$$\text{Differential mode } f_{-3\text{ dB}} \cong \frac{1}{2\pi(2 * 1k\Omega)(\frac{56\text{ pF}}{2} + 56\text{ pF})} \cong 948\text{ KHz}$$

$$\text{Common - mode } f_{-3\text{ dB}} \cong \frac{1}{2\pi(1k\Omega)(56\text{ pF})} \cong 2.8\text{ MHz}$$

图 5-6. 用于相反电动势的分压器和低通滤波器



$$\text{Cutoff frequency}_{-3\text{ dB}} = \frac{1}{2\pi(30k \parallel 2k + 300) 10\text{ nF}} \cong 7321\text{ Hz}$$

$$30\text{ degree phase shift frequency} = \frac{\tan(30)}{2\pi(30k \parallel 2k + 300) 10\text{ nF}} = 4226\text{ Hz}$$

因此，电机的理论最大速度将为：

$$4226\text{ Hz} * 60 \cong 253560\text{ eRPM}$$

$$\text{Mechanical RPM} = \text{eRPM} / \text{Pole pairs}$$

5.5. 用户接口

表 5-4 给出了使用 AVR DX PIM 的 MCLV-2 开发板的用户接口。

表 5-4. 用户接口说明

接口	类型	功能	说明
S1	输入	复位	按下此按钮将复位单片机并使系统进入驱动程序初始化、应用程序初始化状态
S2	输入	启动/停止	启动或停止电机。如果处于错误处理状态，按下此按钮将使系统进入停止状态（制动 PWM LED D10、D12 和 D14 亮起）。
S3	输入	更改方向	更改电机的旋转方向。此操作仅在系统处于停止状态时有效。更改方向也会更改方向 LED。
POT1	输入	设置占空比	直接设置 PWM 信号的占空比
D2 和 D17	输出	电机方向指示	指示电机的方向（顺时针或逆时针，取决于电机接线）
D10-D15	输出	PWM 输出	PWM 输出处于有效状态的视觉指示。没有 LED 亮起表示系统要么已关闭，要么处于错误处理状态。

6. 调整

6.1. 首次启动

如果提供 24V 电源电压并使用默认值，电机将正常启动。

不过，所用电机的极对数可能与默认配置中使用的不同。如果是这种情况，系统将表现为正常行为，但 MPLAB 数据可视化器中显示的 RPM 值将不正确。

要解决此问题，需要修改 AVR_Dx_tuning_params.xslm Excel 文件中的两个字段：

- 电机极对数：根据所用电机进行更改
- 目标 RPM：根据公式更改：
 - $NeWRPM = NeWPolePairs * OldRPM / OldPolePairs$
 - 将数值四舍五入到最接近的整数

如果结果正常且电机响应正确，则无需进行调整。

6.2. Excel 文件参数调整

可能出现电机无法启动或性能不佳的情况。

图 6-1. Excel 文件可编辑参数

	A	B	C	D
	Parameter	Value	Description	Range
1	Startup PWM Duty [%]	25	Startup duty cycle in percents	1 - 100
2	Rotor align time [ms]	250	Rotor align time in which the duty is raised to the startup PWM duty above	1 - 14000
3	Target RPM [RPM]	800	Final RPM after startup. This becomes the minimum RPM(also see tolerance below)	
4	Initial startup period [ms]	300	From where to start the ramp. Never start from zero.	1 - 1000
5	Ramp duration [ms]	2000	Total ramp time without sustain	100 - 5000
6	Ramp sustain duration [ms]	1	Time in which the motor is kept at a constant speed after acceleration	1 - 5000
7	Holdoff steps	1	How many trapezoidal steps have the PWM disabled, after the sustain zone	1-250
8	Motor pole pairs	4	The number of motor pole pairs	1 - 255
9	Minimum PWM duty [%]	20	Minimum PWM duty value in percents - Indirectly limits the minimum speed	0 - 100
10	Maximum PWM duty[%]	100	Minimum PWM duty value in percents - Indirectly limits the maximum speed	0 - 100
11	Braking PWM duty [%]	90	PWM duty cycle for the non-regenerative braking	0 - 100
12	Motoring current limit[mA]	4420	Maximum current allowed to draw from the supply. Triggers a fault condition	0 - 4420
13	Braking current limit[mA]	-4420	Maximum current allowed to push to the supply. Triggers a fault condition	-4420 - 0
14	Undervoltage protection[mV]	11000	Maximum supply voltage. Triggers a fault condition	11000-48000
15	Overvoltage protection[mV]	25000	Minimum supply voltage. Triggers a fault condition	11000-48000
16	Moving average filter division factor	8	Higher values eliminates glitches but has slower response. Set as power of 2	1 - 128
17	Advance angle[Deg]	0	Advance angle for the next sector timing	0 - 30
18	Passthrough delay compensation[us]	200	Fixed compensation time for any hardware or software delays	1-1000
19	Delta division factor	1	Maximum variation of the current zero cross time expressed as (previous time/factor)	0-8(disabled for 0)
20	Minimum RPM tolerance[%]	40	Minimum RPM for closed loop mode. Subtracts this percentage from the startup RPM	0-100

以下段落将重点探讨调整参数及其对系统行为的影响：

1. **Start-up PWM Duty [%]**（启动 PWM 占空比[%]）——表示启动占空比的百分比。值过低可能导致斜升序列无法正常进行，因为绕组中的电流过小，产生的磁场强度不足以使转子跟随。
值过高将触发过流保护或导致绕组过热。启动占空比也主要取决于电机启动时的负载。更高的负载需要更高的占空比。
此参数仅适用于斜升序列。
2. **Rotor align time [ms]**（转子对准时间[ms]）——转子的对准时间，单位为毫秒。占空比通过线性斜升从 0%逐渐增加到启动 PWM 占空比值，从而避免产生机械冲击。这种对准操作的目的是让电机从已知位置启动。
此参数仅适用于斜升序列。
3. **Target RPM [RPM]**（目标 RPM [RPM]）——这是斜升序列后的最终 RPM。如果设置的值过高，则需要更高的占空比，以便电机能够在高 RPM 下跟随磁场。
此参数仅适用于斜升序列，但闭环控制也依赖于它。

4. **Initial start-up period [ms]** (初始启动周期[ms]) ——斜坡的起始点。通常无需调整此参数。
此参数仅适用于斜升序列。
5. **Ramp duration [ms]** (斜坡持续时间[ms]) ——设置斜坡持续的时长。要更快启动, 请减小此值。请注意, 强烈的加速同样需要更高的占空比。
此参数仅适用于斜升序列。
6. **Ramp sustain duration [ms]** (斜坡维持持续时间 [ms]) ——在电机加速的斜升序列之后, 会进入一个维持阶段, 期间电机以**目标 RPM** 持续旋转。除非使用 PI 控制器 (在控制器训练过程中可能带来益处), 否则建议将此参数保持在最小值。
此参数仅适用于斜升序列。
7. **Hold-off steps** (推迟步) ——在斜升序列结束后, 立即禁止 PWM 并使能比较器。这样做是为了在没有任何相位噪声的情况下检测转子的确切位置。除非电机的相电压变化非常缓慢, 否则此参数需要至少保持为 1。
此参数仅适用于斜升序列。
8. **Motor pole pairs** (电机极对数) ——电机的极对数。
9. **Minimum PWM duty [%]** (最小 PWM 占空比[%]) ——最小 PWM 占空比。使用此参数限制电机的最小速度。
此参数适用于闭环和开环控制。
10. **Maximum PWM duty [%]** (最大 PWM 占空比[%]) ——最大 PWM 占空比。使用此参数限制电机的最大速度。
此参数适用于闭环和开环控制。
11. **Braking PWM duty [%]** (制动 PWM 占空比[%]) ——电机在停止模式下的占空比。系统受到保护, 可防止因高速制动导致的过流。在这种情况下, 固件将进入故障状态并保护晶体管。
此参数仅适用于停止状态。
12. **Motoring current limit [mA]** (电机电流限制[mA]) ——从总线获取的最大电流值, 超过此值将触发故障条件。最大值由 MCLV-2 限制为 4420 mA。
此参数适用于除错误处理状态外的所有系统状态。
13. **Braking current limit [mA]** (制动电流限制[mA]) ——可馈送到总线的最大电流值, 超过此值将触发故障条件。最大值由 MCLV-2 限制为-4420 mA。
此参数适用于除错误处理状态外的所有系统状态。
14. **Undervoltage protection [mV]** (欠压保护[mV]) ——MCLV-2 总线上的最小电压。不要设置低于 11V 的值。
此参数适用于除错误处理状态外的所有系统状态。
15. **Overvoltage protection [mV]** (过压保护[mV]) ——系统的最大电压。如果使用单个电源, 请勿设置超过 24V 的值; 如果使用双电源, 请勿设置超过 48V 的值 (请参见 [MCLV-2 user guide](#))。
此参数适用于除错误处理状态外的所有系统状态。
16. **移动平均滤波器分频系数** (Moving average filter division factor) ——过零时间的移动平均滤波器分频系数需要设置为 2 的幂, 因为它可以通过编译器使用右移进行优化。建议使用较高的值, 因为系统对任何毛刺有更高的抗扰性, 但会限制两个过零事件之间的最大变化, 从而限制电机的最大加速度。此参数仅适用于闭环控制。
17. **Advance angle [Deg]** (提前角[度]) ——电机的提前角。增大此值将使波形的产生相对于电机反电动势过零点提前特定角度。同时, 它也会影响电机产生的转矩。
此参数仅适用于闭环控制。
18. **Pass-through delay compensation [us]** (直通延时补偿[us]) ——下一扇区定时的固定提前时间。此值直接从下一扇区定时中减去, 以补偿任何软件或硬件延时。
此参数仅适用于闭环控制。

19. **Delta division factor**（增量分频系数）——此参数控制附加的失速和毛刺检测机制。在过零点检测过早或过晚的情况下，如果前一次过零时间与当前测量时间之间的绝对差值大于前一次过零时间/增量分频系数，则会触发故障条件。为提高代码效率，增量分频系数最好设置为 2 的幂。过高的值会导致系统变得不稳定，并会通过触发故障条件来限制最大加速度。过低的值可能无法很好地检测毛刺，但在启动时不会遇到任何问题。

此参数仅适用于闭环控制。

20. **Minimum RPM tolerance [%]**（最小 RPM 容差[%]）——此参数控制定时器的超时和最小 RPM。启动后，系统可能会比预期稍晚检测到过零点，如果容差极低，则会被视为超时事件。为避免这种情况，需要以最小 RPM 百分比的形式给定一个容差。因此，最小 RPM 变为 **目标 RPM - 目标 RPM * 最小 RPM 容差**。

此参数仅适用于闭环控制。

修改参数后，按下 **Patch File**（补丁文件）按钮，然后重新编译并用固件对 AVR DX 进行编程。

6.3. 更改控制模式

斜升后，系统将以闭环或开环模式运行。用户可以通过配置单行代码自由选择每一种模式。

图 6-2. 控制模式代码行

```
35 | /* Control mode after ramp-up */
36 | #define MOTOR_CONTROL_MODE MOTOR_CLOSED_LOOP_MODE
```

有效选项如下：MOTOR_CLOSED_LOOP_MODE 或 MOTOR_OPEN_LOOP_MODE。

这两种模式的行为如下所述：

1. 闭环模式：在默认控制模式下，换向根据从内部比较器接收的过零事件进行。主动保护措施如下：
 - 过流保护
 - 过压/欠压保护
 - 过零增量超限
 - 过零超时
2. 开环模式：在备用控制模式下，换向借助负责斜升序列的定时器进行。电机按 Excel 文件中设置的 RPM 旋转。在测试斜升序列是否正常工作时，这尤为有用，可确保转子在斜升结束阶段不会因磁场变化而停止跟随。在此模式下，电位器用于设置占空比，但电机速度将保持恒定。在此模式下，以下保护措施有效：
 - 过流保护
 - 过压/欠压保护

7. 结论

本应用笔记介绍了使用双极开关驱动 BLDC 电机的方法，内容涵盖波形生成以及将相应信号连接到所需引脚的整个过程。

无需外部比较器或其他逻辑电路，惟一例外是用于调理连接至单片机的输入信号的少数无源元件。

运行频率为 24 MHz 的 AVR 内核可提供高达 24 MIPS 的性能，是一种极其强大的系统设计选择。它既能满足多种功能需求，又能结合独立于内核的外设（AC、ADC、TCA、TCB、CCL 和 USART）来驱动电机，从而构成完整的控制系统。

双极驱动可确保干净的 BEMF，有助于精确检测过零点，无需与 PWM 输出信号同步。

为提升性能，内置的移动平均滤波器可在外部干扰情况下提供额外的保护措施，确保电机在整个 RPM 范围内平稳运行。

8. 参考资料

1. [AN857, Brushless DC Motor Control Made Easy \(DS00857\)](#)
2. [AN1160, 用择多函数实现反电动势滤波的无传感器 BLDC 控制 \(DS01160B_CN\)](#)
3. [AVR444: Sensorless control of 3-phase brushless DC motors](#)
4. [AN2522, Core Independent Brushless DC Fan Control Using CCL AVR® microcontrollers \(DS00002522\)](#)

9. 版本历史

文档版本	日期	备注
B	2021 年 6 月	更新了简介部分，添加了 GitHub 链接。
A	2021 年 5 月	文档初始版本。

Microchip信息

商标

“Microchip”的名称和徽标组合、“M”徽标及其他名称、徽标和品牌均为 Microchip Technology Incorporated 或其关联公司和/或子公司在美国和/或其他国家或地区的注册商标或商标（“Microchip 商标”）。有关 Microchip 商标的信息，可访问 <https://www.microchip.com/en-us/about/legal-information/microchip-trademarks>。

ISBN: 979-8-3371-2417-9

法律声明

提供本文档的中文版本仅为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物及其提供的信息仅适用于 Microchip 产品，包括设计、测试以及将 Microchip 产品集成到您的应用中。以其他方式使用这些信息都将被视为违反条款。本出版物中的器件应用信息仅为您提供便利，将来可能会发生更新。您须自行确保应用符合您的规范。如需额外的支持，请联系当地的 Microchip 销售办事处，或访问 www.microchip.com/en-us/support/design-help/client-support-services。

Microchip “按原样”提供这些信息。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对非侵权性、适销性和特定用途的适用性的暗示担保，或针对其使用情况、质量或性能的担保。

在任何情况下，对于因这些信息或使用这些信息而产生的任何间接的、特殊的、惩罚性的、偶然的或附带的损失、损害或任何类型的开销，Microchip 概不承担任何责任，即使 Microchip 已被告知可能发生损害或损害可以预见。在法律允许的最大范围内，对于因这些信息或使用这些信息而产生的所有索赔，Microchip 在任何情况下所承担的全部责任均不出您为获得这些信息向 Microchip 直接支付的金额（如有）。如果将 Microchip 器件用于生命维持和/或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切损害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任。除非另外声明，在 Microchip 知识产权保护下，不得暗中以其他方式转让任何许可证。

Microchip 器件代码保护功能

请注意以下有关 Microchip 产品代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术规范。
- Microchip 确信：在按照操作规范正常使用的情况下，Microchip 系列产品非常安全。
- Microchip 重视并积极保护其知识产权。任何试图破坏 Microchip 产品代码保护功能的行为均可视为违反了《数字器件千年版权法案（Digital Millennium Copyright Act）》并予以严禁。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。代码保护功能处于持续发展中。Microchip 承诺将不断改进产品的代码保护功能。