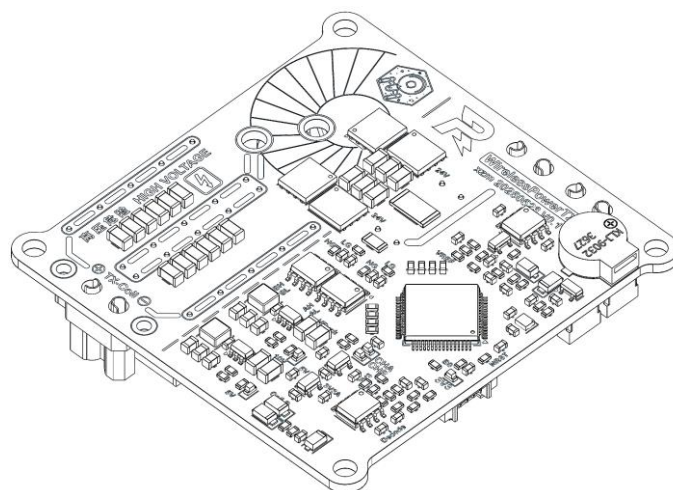
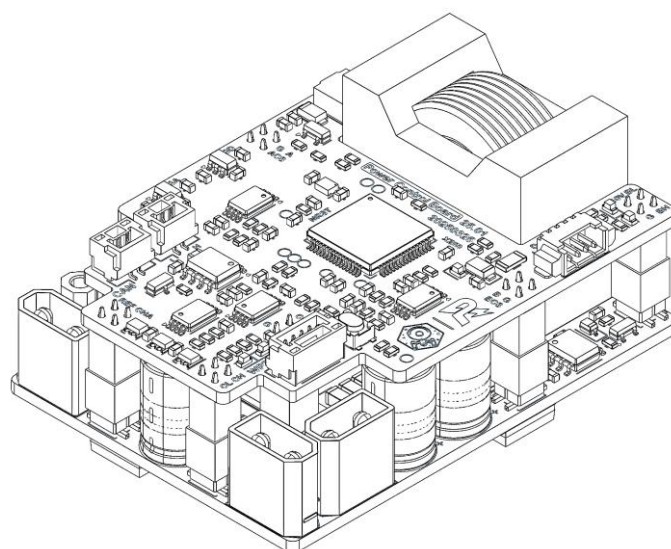


超级电容与无线充电系统技术方案

RM2025 香港科技大学 ENTERPRIZE 战队

2025/08/08 V1.0



目录

1. 技术背景概述.....	4
2. 整体硬件架构.....	5
2.1 超级电容系统硬件拓扑.....	5
2.2 无线充电系统硬件架构.....	5
3. 超级电容系统.....	6
3.1 方案总览与主要创新点.....	6
3.2 硬件设计.....	7
3.2.1 硬件设计总览.....	7
3.2.2 功率级.....	8
3.2.3 驱动级.....	11
3.2.4 上管常开实现方式.....	12
3.2.5 采样与反馈.....	14
3.2.6 MCU 与控制.....	19
3.2.7 辅助供电.....	20
3.2.8 通讯与交互.....	23
3.2.9 成本控制.....	23
3.3 环路与控制.....	24
3.3.1 整体控制框图.....	24
3.3.2 控制系统整体需求分析.....	24
3.3.3 纯数字控制模式分析.....	27
3.3.4 峰值电流控制模式介绍.....	30
3.3.5 峰值电流 Buck-Boost 硬件架构.....	33
3.3.6 时序图.....	35
3.3.7 MCU 外设配置.....	39
3.3.8 升降压模式切换.....	47
3.3.9 数字外环设计与实现.....	51
3.3.10 充电限制和纯电容模式.....	58
3.4 保护.....	59
3.4.1 过压与过流保护.....	59
3.4.2 短路与功率级保护.....	63
3.4.3 通讯断连保护.....	64
3.4.4 电容组漏电/断连保护.....	64
3.4.5 错误分级处理与其他保护.....	65
3.5 通讯与交互.....	66

ROBOMASTER

3.5.1 CAN 通讯.....	66
3.5.2 交互	67
3.6 超级电容组设计.....	68
3.6.1 需求分析	68
3.6.2 电容选型	68
3.6.3 原理图与 Layout	69
3.6.4 保护外壳	71
4. 无线充电系统.....	73
4.1 发射端	73
4.1.1 功率级及驱动	74
4.1.2 谐振电路	75
4.1.3 采样与反馈.....	77
4.1.4 辅助供电	78
4.1.5 ASK 解码电路	79
4.1.6 功率级硬件配置	81
4.2 接收端	83
4.2.1 接收端线圈板	83
4.2.2 功率控制板无线充电部分	85
4.3 控制与通讯算法.....	86
4.3.1 FSK 通讯	86
4.3.2 ASK 通讯	86
4.3.3 发射端状态机	89
5 迭代/测试记录.....	90
6 未来改进方向.....	93
7 参考资料.....	95
8 致谢	96

The Hong Kong University of
Science and Technology

1. 技术背景概述

在 RM2025 的规则中，对于整车硬件要求大为提高。首先，总能量的限制的引入对电容控制器的效率、响应速度等提出了更高的要求；其次，无线充电系统的加入使得底盘及超级电容系统的架构变得更加复杂。

RM2025 赛季的一大变化是“能量限制”，与之前单纯的功率限制相比，能量限制对整车设计要求再一次提高，从“开源”与“节流”两个角度分别来分析针对“能量限制”加入的优化方法：

开源（增大可用能量）

- [硬件] 进行无线充电（在充电站以 120W 功率向电容组充入额外能量），详见“4. 无线充电系统”
- [硬件] 提升比赛开始时的电容容量：例如比赛开始时电容组实际存入 1600J，实际可用能量为 $23000\text{J} + 1600\text{J} = 24600\text{J}$ ，假设电容组的实际容量可以提升为 1900J，那么实际可用能量也会相应提升，详见“3.6 超级电容组设计”

节流（减小能量损耗）

- [硬件] 进行动能回收，将刹车时的能量存入电容，详见“3.3.9 数字外环设计与实现”
- [硬件] 减小硬件损耗，如电容控制器、电容组、线损等等，在下文中将会重点讨论。详见“3.2 硬件设计”

本文分为三部分探讨解决方案：整体硬件架构、超级电容系统与无线充电系统。

The Hong Kong University of
Science and Technology

·2. 整体硬件架构

2.1 超级电容系统硬件拓扑

超级电容系统拓扑继续沿用上代电容控制器的**双向 Buck-Boost 升降压架构**，使得电容组电压可以高于 等于或低于底盘电压，电容组最大电压值可以达到规则限制的 30V。由于电容组电流受限于裁判系统超级电容管理模块 CM01 的 15A，所以电容组电压越高，理论最大放电功率越大。以本设计中的 29V 电容组为例，其满电时理论最高可输出 435W 功率。

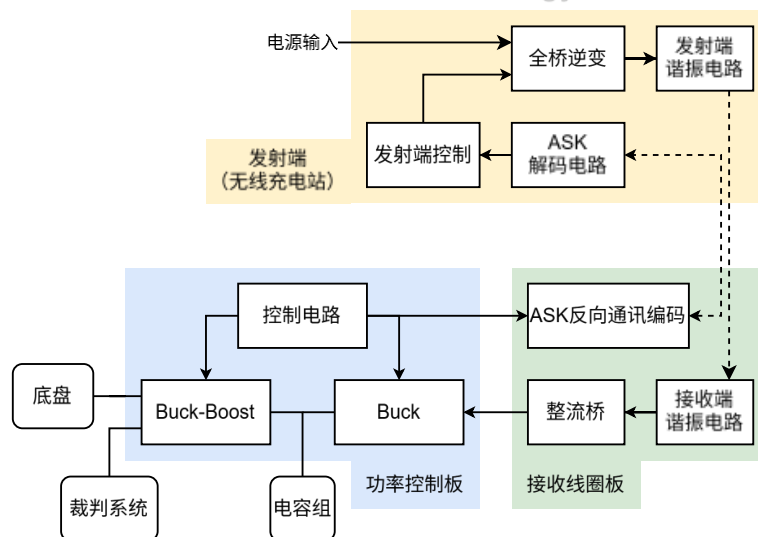
2.2 无线充电系统硬件架构

最大化无线充电系统的效率需要通过配谐来最小化无功功率，在 RM 场景下对无线充电系统有以下要求：

- 无线充电系统的发射端体积可以较大，但接收端体积限制较为严格
- 在未对准时需要有较强的可靠性
- 发射、接收端间需要进行握手后进行充电
- 接收端需要能够处理 120W 的输入功率，并将其充入电容组中

在以上需求之外，为了简化硬件设计和接线难度，将无线充电接收端的相关电路与超级电容系统融合，控制在同一块 MCU 上实现。

在考量上述因素并且查阅相关资料后，最终决定在功率传输部分使用 LCC-S 拓扑，激励源选择全桥逆变，接收端的功率变换器选择 Buck 架构，将整流桥输出降压输入到超级电容中。无线充电握手部分，选择通过模仿较为成熟的 Qi 协议的方法，通过 FSK 和 ASK 实现。



3. 超级电容系统

3.1 方案总览与主要创新点

峰值电流内环

25 赛季超级电容控制器的最大改变是实现了峰值电流内环控制，由模拟电路实现的峰值电流内环理论上可以实现电感电流单周期达到设定峰值，配合简单的软件外环，其电流阶跃响应可以达到 20us 级别。同时，纯模拟内环大大降低了 MCU 的运算负载，使得数字环路的频率可以进一步提高，以实现更快的响应速度与更灵敏的保护。

峰值电流采样与平均电流采样共用放大器，最小化斜坡补偿

与峰值电流常见的在低位采样的设计不同，本设计在高位 MOS 处加入采样电阻进行采样，将滤波电容放置于采样电阻后以采集电感电流。电流采样放大器输出直接连接到比较器的输入进行峰值电流控制，同时也经过二阶低通滤波获得平均电流，并使用 ADC 采样用于软件计算。这样的设计无需增加额外的采样电路，同时这样可以得到更多开关阶段的电感电流，可以通过选择不同阶段的电流进行控制来最小化斜坡补偿的幅度。

充分利用外设简化电路

在具体实现上，得益于 STM32G474 的强大外设，硬件电路极为简化。峰值电流控制环路的实现仅在电流感应放大器部分使用了 INA240A1 外置元件，剩余部分的环路，包括比较器和 DAC 都在 MCU 内部实现；在电压检测中，通过使用 MCU 内置的运放的 Follower（跟随器）模式，省去了外置运放，节约成本的同时简化了 PCB 布局。

充分利用外设减小软件运算量

在硬件层面外，25 赛季的超级电容系统充分利用 MCU 内部的不同外设间的互联矩阵，进一步通过硬件外设替代软件计算。例如 OVP（过压保护）和 OCP（过流保护）通过 ADC 的 AWDG（模拟看门狗）实现触发保护，在进一步减小 MCU 的运算量的同时也加快了保护的响应速度。在实际测试中，25 赛季超级电容系统的保护响应速度快于 24 年的设计，在短路保护测试中尤其明显，短路产生的火花几乎不可见，且多次触发也不会损坏功率级电路，证明了其优异的响应速度和稳定性。

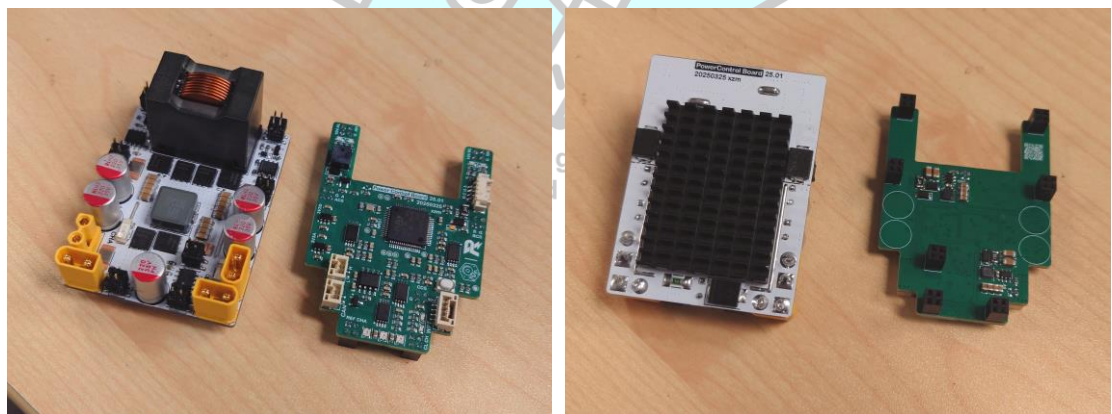
无隔离电源上管常开

四开关 Buck-Boost 实现上管常开有助于提高转换效率。与使用隔离电源模块的传统做法不同，25 赛季超级电容控制器的功率级利用拓扑本身的特性，使用了类似电荷泵的结构实现上管常开，同时无需增加额外的控制芯片。无隔离电源的设计有助于减小静态功耗，使用隔离电源模块时，静态功耗约为 2W，使用类似电荷泵的方式静态功耗仅有 0.8-1.2W 左右。

3.2 硬件设计

3.2.1 硬件设计总览

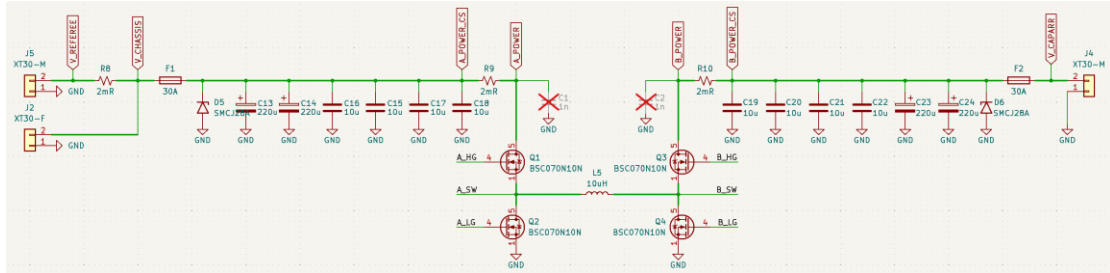
硬件上整体分为上下两块 PCB 板，上板为四层板，包含所有控制电路，下板为双层板，包含超级电容和无线充电接收端的功率级，两板间使用 2.0mm 排针连接。



由于测试中发现下板功率级发热不大，故未采用铝基板，使得硬件设计更为自由（可以使用直插电容、XT30 等通孔元件），并在板后粘贴一块 40x30x5mm 的散热片来辅助散热。将发热网络（例如 MOS 管的 D 极）通过密集过孔导热至 PCB 背面。

3.2.2 功率级

功率级为双向四开关 Buck-Boost 架构，分为 AB 两侧。本设计中将采样电阻放置于滤波电容和高侧 MOSFET 间以对电感电流进行采样，当高边 MOSFET 打开时，采样电阻上的电流等于电感电流，具体的峰值电流控制方式将在后文“3.3 环路与控制”部分详细介绍。



开关频率与电感选型

选择电感时，要在感值和饱和电流足够的情况下保证体积不会过大、且内阻尽量小。本设计中电流双向流动，变换器的功率级一直处于 CCM 模式，控制上有四种工作模式（“3.3.6 时序图”中详细说明），分别为 Buck、Buck-Boost($V_{in} < V_{out}$)、Buck-Boost($V_{in} > V_{out}$)、Boost。定义 A 侧为输入端，由于 A 侧连接电管，电压范围即为电池电压范围，以使用 TB48S 为例，电压最小为 22.2V，最大为 26.1V。B 侧为输出端，连接电容，电容电压范围为 0 - 30V。选择开关频率为 250kHz，此频率下可以保证开关损耗不会过高、电感体积不会过大且不会因为频率过高提升峰值电流控制的难度。适当大小的电流纹波会使得峰值电流控制更容易，所以选取目标峰值电流为 4A 以内。

$$\text{CCM Buck 模式下, 电感电流纹波为: } \Delta I_L = \frac{V_{out}}{L} (1 - D_A) \frac{1}{f_s} = \frac{V_{in}}{L f_s} (1 - D_A) D_A$$

当 $D_A = 0.5$ 时, ΔI_L 取最大值, V_{in} 最大为 TB48S 电池的 26.1V, 因此:

$$\Delta I_{L,max} = \frac{26.1V}{L \times 250kHz} \times 0.25, \text{ 令 } \Delta I_{L,max} < 4A, L > 6.525\mu H$$

$$\text{CCM Boost 模式下, 电感电流纹波为: } \Delta I_L = \frac{V_{in}}{L} (1 - D_B) \frac{1}{f_s} = \frac{V_{in}}{L} \left(1 - \frac{V_{in}}{V_{out}}\right) \frac{1}{f_s}$$

当 $V_{out} = 30V, V_{in} = 22.2V$ 时取得最大值:

$$\Delta I_{L,max} = \frac{22V}{L \times 250kHz} \times \left(1 - \frac{22V}{30V}\right), \text{ 令 } \Delta I_{L,max} < 4A, L > 6.96\mu H$$

CCM Buck - Boost 模式下, 设计一侧占空比为 0.84, 另一侧小于 0.84 (后文会详细展开), 此时 $V_{out} \approx V_{in} \approx 24V$, 估算电感电流纹波为: $\Delta I_L = \frac{24V}{L} (1 - 0.84) \frac{1}{f_s}$

$$\text{令 } \Delta I_L < 4A, L > 3.84\mu H$$

B 侧由于输出电流会受到裁判系统电容管理模块 CM01 的 16A 电流限制，实际最大电感电流为 Boost 状态下 $V_{out} = 30V, V_{in} = 22V$ 时取得，此时：

$$I_{L,peak} = 16A \times V_{out}/V_{in} + 0.5\Delta I_L < 22.5A$$

考虑留有一定余量，选择电感饱和电流至少为 25A

最终，选择 10uH VER2923 电感，饱和电流 30 - 37A，DCR 约为 2.5mΩ，损耗较小，最大电流纹波小于 3A，满足设计要求。

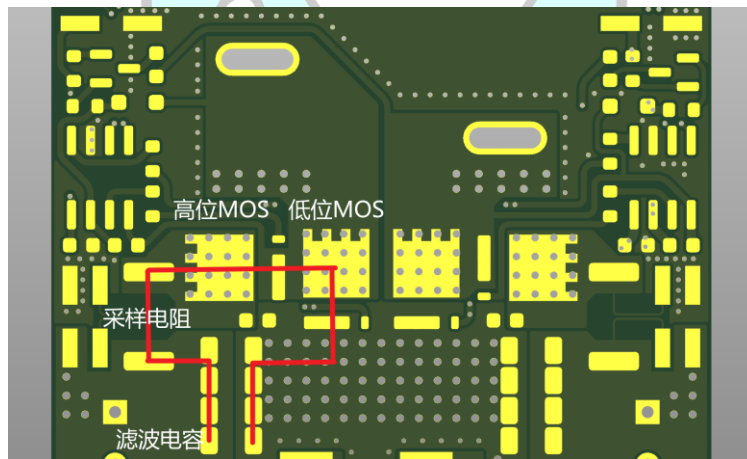
电容选型

选择电容时，假设最大电流纹波为 3A，由于 Buck-Boost 模式下电感电流波形近似为梯形，Boost 或 Buck 模式下近似为三角波，所以在电流纹波相同时最大电压纹波在 Buck-Boost 时取得（积分电感电流与平均电流的差值最大），可以估算输出纹波一定小于：

$$\Delta V_{out} < \frac{T}{2} \times \frac{I_{pp}}{2} \times \frac{1}{C_{out}}$$

在采样电阻后每侧使用 4 个 10μF 50V MLCC 电容和 2 个 220μF 35V 固态电容 分别进行低频和高频滤波，此时输出纹波理论最大不超过 50mV，符合设计要求。

在 PCB Layout 时将高频功率环路（MLCC 电容 - 采样电阻 - 半桥 - 地平面）面积最小化，并在保证过流能力足够的情况下最小化两侧 SW 的面积来减小干扰。



采样电阻选型

采样电阻选用 2mΩ 2512 的合金电阻 进行电感电流，选型时主要有以下考量：

- 考虑损耗时，采样电阻不能太大。2512 电阻的最大功率为 3W，以最大电感电流 25A 计算， $R_{max} = P_{max}/I_{max}^2 = 4.8m\Omega$ ，且相对 NMOS 内阻 (5.5mΩ) 不能太大
- 考虑采样时，采样电阻不能太小。采样电感电流需要较高带宽的电流感应放大器，而一般放大倍数越大的放大器带宽越低、噪声越大，限制了放大倍数不能很大。由于 ADC 的精度限制，采样电阻阻值太小会导致分辨率过低。结合使用的电流感应放大器

INA240A1 (选型在后文“3.2.5 采样与反馈”会提到) 的 20V/V 放大倍率, 最终选择 2mΩ的采样电阻。

最终 ADC 采样值理论计算公式为 (ADC 分辨率 12bit) :

$$\text{ADC Value} = I \times 2\text{m}\Omega \times 20\text{V/V} \times (4096/V_{REF})$$

理论分辨率和量程为 (VREF 取 2.9V) :

$$I_{res} = \frac{2.9\text{V} \times 1}{2\text{m}\Omega \times 20\text{V/V} \times 4096} = 0.0177\text{A}$$

$$I_{range} = \pm \frac{2.9\text{V}/2}{2\text{m}\Omega \times 20\text{V/V}} = \pm 36.25\text{A}$$

功率 MOSFET 的选型

对于功率 MOSFET, 要在保证 VDS 耐压余量足够的情况下, 取得栅极电荷量、导通内阻间的平衡。这里沿用 2024 赛季队内开源的选型 BSC066N06NS: 其耐压为 60V, 高于两倍最大电压, 防止电压过冲损坏 MOS; 栅极电荷小于 20nC, 保证较低的开关损耗并缩短死区时间, 防止在 VDS 较高时超过 SOA; Vgs = 10V 时 Rds(on) 为 5.5mΩ, 损耗较小; 持续电流 64A、峰值 256A, 远大于正常运行需求, 保证了足够的安全余量。

其他设计

两侧滤波电容后各放置有一对 TVS+保险丝的保护电路, 此电路可以在电流失控、输入输出反接或过压等情况时熔断保险丝。保险丝选型为 1808 封装的 20A 快断保险丝, 内阻约为 1.5mΩ。

A 侧在保险丝后有另一个 2mΩ 的采样电阻, 用来采集裁判系统端的输入电流, 此处虽然电流更小 (最大约为满级功率时 5A 左右), 但是考虑到简化 BOM (防止错焊、减少备料种类等) 且分辨率足够的情况下保持相同的采样电阻和电流感应放大器的配置。

将采样电阻放置于保险丝后的设计可以保证即使在超级电容系统发生严重故障、保险丝断开时, 底盘也可以直接被裁判系统供电, 且整个超级电容系统完全断开。

3.2.3 驱动级

栅极驱动芯片选型

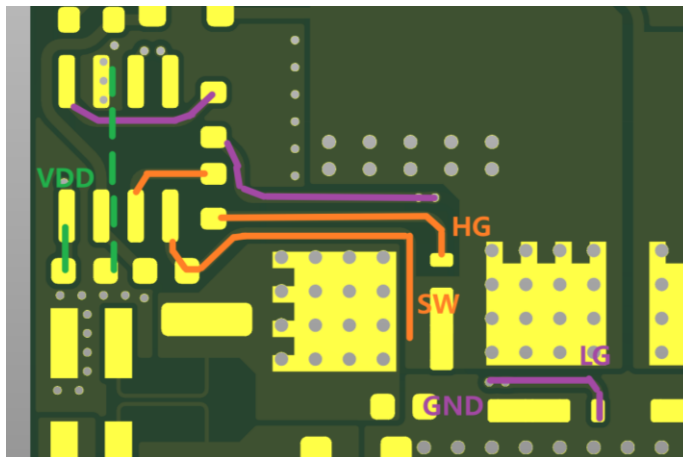
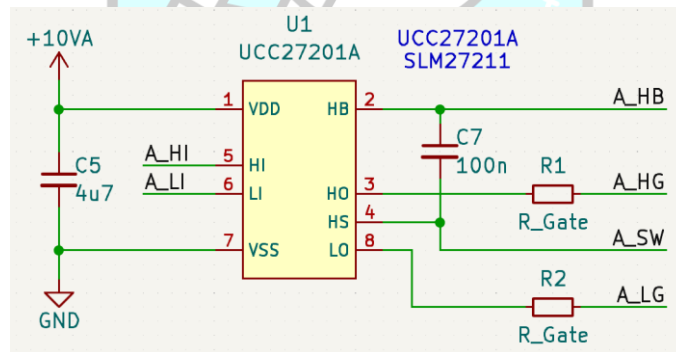
栅极驱动芯片选用 UCC27201A 或者其平替 SLM27211。虽然 UCC27201A 性能稍弱于 UCC27211，但后者非常容易买到假货，且 UCC27201A 3A 的推挽能力在 250kHz、功率 MOSFET 栅极电荷不到 20nC 的已经足够，所以选用 UCC27201A 作为栅极驱动。后来经过测试发现 SLM27211 稳定性足够且价格更便宜，于是在后续版本换用 SLM27211。

栅极驱动电路

栅极驱动电路以 UCC27201A 为核心，贴近 VDD 引脚放置一个 4.7μF 50V 电容，保证在开关瞬间不会因为寄生电阻过大导致 VDD 电压突降而使芯片进入 UVLO 保护。在输出引脚 HO 和 LO 各串联一各栅极电阻，防止过快的开关导致电压过冲，此电阻需要通过测试反复调整为合适的值，这里取 3.3Ω。自举电容按数据手册选用 100nF 0603 50V 电容。

布局中要尽量缩短高 di/dt 回路，尤其是栅极的走线要尽可能粗且短，输入电容要贴近芯片的 VDD 引脚并保证地的连通性较好。

为了保证 MOSFET 的 R_{dson} 较小，设置驱动电压为 10.8V。过高的电压会使栅极电荷增大，过小则会加大 R_{dson} ，考虑到数据手册推荐电压为 10V 和 高频开关中寄生参数的损失（上管自举电容充电二极管的压降等），设计驱动电压为 10.8V



3.2.4 上管常开实现方式

在输出与输入电压有一定差距后，Buck 或 Boost 模式较比 Buck-Boost 更为高效，但如果一侧上管直接常开，由于 G 与 S 级间漏电等原因，上管会渐渐转为不导通状态，所以要实现上管常开必须采用一定方法保持上管栅极电压不变。

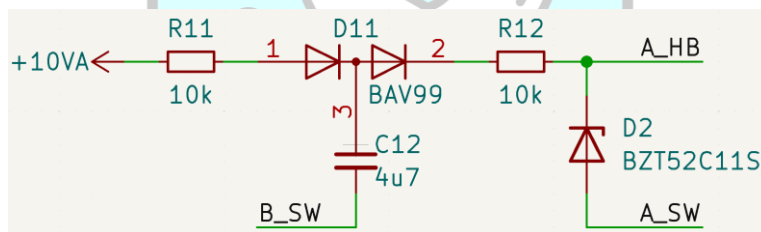
现有方案参考

队内 2023 和 2024 赛季的解决方案为引入两个隔离电源模块，输入为 10.8V 和 GND，输出负连接到 SW，正连接到 HB（自举电容两侧），这个方案的优点是设计较为简单，但是即使隔离电源输出电流很小，也会有较大的静态功耗（在本赛季这个问题会被放大）。

另一种解决方式是在上管常开一侧以较低的频率进行开关充电，例如对侧开关 500 次，常开侧进行一次占空比为 0.95 的开关，并配合增大的自举电容实现上管近似常开。这种方案的优点是没有引入额外的静态功耗，且无需额外硬件上的改变，但是软件上实现较为复杂。

本系统使用的新方案

在本方案种则使用了一种类似电荷泵原理的方式实现上管常开。如图，其中 BAV99 为 100V 耐压、SOT-23 封装的开关二极管（其实就是两个二极管合在一起），BZT52C11S 为齐纳电压为 $11V \pm 0.6V$ 的稳压二极管，两个 10kΩ 电阻为限流电阻。输出连接到自举电容两端。



此电路工作原理主要依赖一个特性：常开侧对侧的电路一定在进行开关，例如 Buck 工作时 A 侧进行 250kHz 的开关，B 侧常开，Boost 同理。

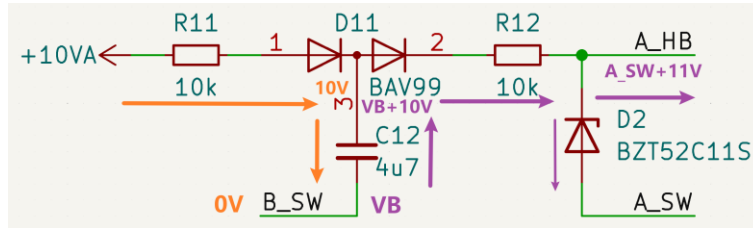
原理分析

以如图 Boost 工作、输入电压 24V、输出电压 24V（理论上最不理想的电压）需要 A 侧上管常开时为例，工作分为两阶段：

B 侧下管打开，此时 B_SW 为 0V，电容 C12 通过 R11 电阻与 D11 左侧二极管充电，此时 D11 右侧的二极管截止。在稳态运行中，C12 的电压为 $10.8V - 0.8V = 10V$ 。

B 侧上管打开，此时 B_SW 为 24V，A_SW 为 24V，C12 上端电压约为 $24V + 10V = 34V$ ，此时如果自举电容电压较低，则 C12 通过 D11 右侧的二极管与 R12 电阻向自举电容充电，同时通过稳压二极管限制自举电容两端电压为 11V 左右，稳态运行中自举电容可以至少可以被充电至 9.2V 左右，即高侧开关 Vgs 至少会保持在 9.2V 左右。

不考虑漏电，实际运行中因为常开侧电压一定小于开关侧，且差距一般至少为 2V（否则进入 Buck-Boost 模式），常开的上管的 Vgs 一定可以维持到稳压二极管的电压（11V 左右）。



漏电与充电理论计算

上管常开时，栅极漏电的来源有 (1)MOSFET 本身 (2)稳压二极管 (3)自举电容 (4)驱动芯片，经查阅数据手册，其最大值分别为 (1)100nA (2)0.1μA (3)~5nA (4)1μA，加和为 μA 级别，且驱动芯片正常漏电为 0.0005μA，与最大值相差较多，实际漏电电流远小于估算值。正常运行时，假设 C12 两端电压为 9.9V，自举电容两端电压为 9.0V，开关二极管压降为 0.8V (10mA)，则单次开关的充电能力为（两阶段一致）：

$$Q = \Delta V \left(1 - e^{-\frac{t}{RC}}\right) C = (10.8V - 0.8V - 9.9V) \left(1 - e^{-\frac{2\mu s}{4.7\mu F \times 10k\Omega}}\right) \times 4.7\mu F \approx 0.02nC$$

此时平均充电电流 $I \approx \frac{0.02nC}{4\mu s} \approx 5\mu A$ 大于漏电流最大值，由于估算条件极为保守，实际充电电流会显著大于估算值。

静态功耗分析

估算静态功耗时，假设自举电容稳定在 10V，充电电流忽略不计，功耗近似为：

$$P_{DC} = |V_A - V_B| \times (|V_A - V_B| - 11V) / 20k\Omega$$

可以看出最大静态功耗为 A 侧 24V B 侧 0V，电容刚刚开始充电的场景，此时功耗约为 0.016W 且只有单侧上管常开电路有功耗，在正常运行电压范围（B 侧 10-30V）时静态功耗极低，仅有 mW 级别，可以忽略不计。

结论与实际测试

综上所述，本电路理论上可以十分稳定且低功耗地实现上管常开，实际测试中栅极电压可以稳定在 10V 左右。Buck 和 Boost 状态下都可以稳定运行。对比隔离电源模块的方案，25 赛季采用的新方案使得整体静态功耗从 2W 降低到 0.8W-1.2W 左右，并缩小了元件所占体积。

3.2.5 采样与反馈

总体需求分析

超级电容系统中，需要采集 5 个物理量作为控制依据，分别为：两侧电压 VA、VB，两侧电流 IA、IB，裁判系统端电流 IR。其中两侧电流需要特殊处理，峰值电流需要电感电流的波形，而 ADC 需要采样其平均电流。

为了提高响应速度，软件外环的运行频率为 62.5kHz (开关频率/4)，每开关周期一采样，所以需要采样电路滤波有较高截止频率。数控电源滤波频率一般取开关频率的 1/10 到 1/20，考虑到提升响应的需求，适当选择较高频率，接近 25kHz 左右。

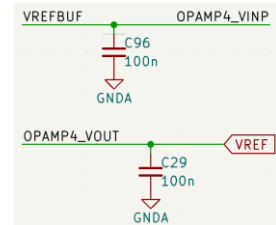
由于 STM32 的 ADC 采样时近似于给一个小电容充电，所以如果条件允许则在 ADC 引脚前放置一个小电容，防止采样时间过短产生误差，或使用运放进行跟随来减小阻抗。

参考电压

采样电路与 ADC 需要精确的参考电压以保证结果的稳定与一致。如果直接使用 LDO，即使在进行校准后也无法避免温度和输出电流等因素导致的电压漂移。所以在本设计中使用了 STM32G4 系列内置的电压参考，省去了独立的参考电压芯片。

STM32G4 系列内置高精度电压参考，有 2.048V 2.5V 2.9V 三个电压挡位，本设计中选用 2.9V 挡位，其精度标称为 ±4mV，理论精度 0.14%，意味着测量功率时由参考电压造成的误差最大为 0.276%。

为了给电路感应放大器提供参考电压，将 MCU VREF+ 引脚输出的参考电压经过 MCU 内部运放跟随后的输出作为参考电压输出，同时在 VREF+ 引脚和最终输出的 VREF 网络都使用电容进行滤波和稳压。



The Hong Kong University of Science and Technology

电压采样电路

电压采样首先使用 33k 和 2.2k 电阻分压，与 3.3nF 电容组成一阶滤波，设计参数如下：

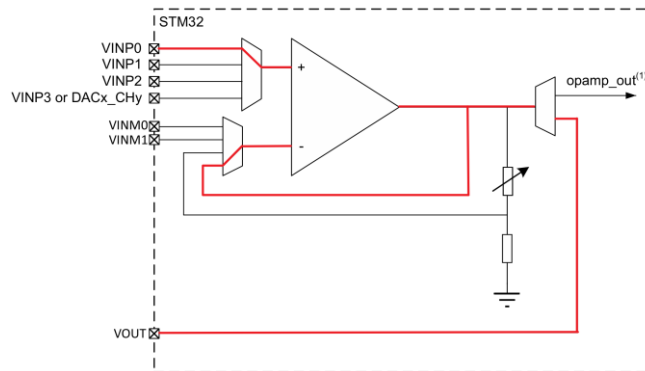
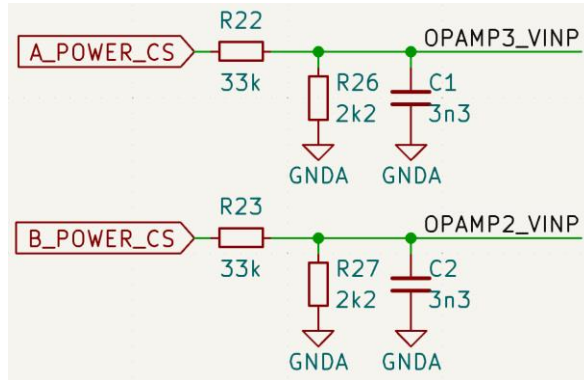
$$\text{直流缩放倍率: } \text{Gain}_{\text{DC}} = \frac{2.2\text{k}\Omega}{2.2\text{k}\Omega + 33\text{k}\Omega} = \frac{1}{16} \text{ V/V}$$

$$\text{最大量程: } V_{\text{max}} = V_{\text{Reference}} / \text{Gain}_{\text{DC}} = 2.9\text{V} \times 16 = 46.4\text{V}$$

$$\text{一阶滤波截止频率: } f_{\text{cutoff}} = \frac{1}{2\pi R_{\text{eq}} C} = \frac{1}{2\pi(33\text{k}\Omega \parallel 2.2\text{k}\Omega) \times 3.3\text{nF}} \approx 23.38\text{kHz}$$

$$\text{ADC 理论分辨率: } V_{\text{max}} / 4096 = 11.328\text{mV}$$

滤波后的信号进入 STM32G4 内部运放的输入，内部运放配置为 Follower 模式，并将运放输出连接到 ADC 输入。内置运放的 Power Mode 配置为 High Speed，可以提供 45 V/μs 的压摆率，更适合处理高频模拟信号。

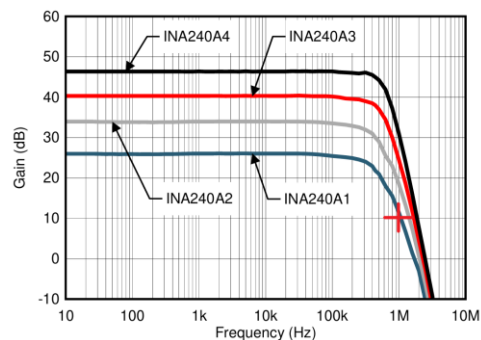


电流感应放大器的选择

峰值电流控制需要获得波形较为精确的电感电流，这就对采样带宽提出了很高要求，而其电流绝对精度要求不高，所以选择放大器时基本仅需重点关注带宽和压摆率（例如 LM51770 的电感电流采样放大器带宽为 1MHz），配合低位采样即可较好地获得相对精确的电感电流波形。而 ADC 采样的平均电流对带宽要求不高（一般为开关频率 1/10 到 1/20 左右即足够），但却对绝对精度要求较高，选择放大器时需要重点关注误差、漂移和噪声抑制等参数。同时，由于电流双向流动，需要选择支持双向电流采样或带有 VREF 引脚的电流感应放大器。

综上所述，本方案的做法虽然电路简单，但是对电流感应放大器要求较高，需要同时具备高带宽、高共模电压、低误差、低漂移和噪声抑制等特性。由于符合要求且带宽 800kHz 以上的电路感应放大器价格较高（例如带宽 1.1MHz 的 INA241A1 价格接近 20 元/片），所以在测试 INA240A1 后发现其带宽足以稳定实现峰值电流控制后，选用其作为电流感应放大器。

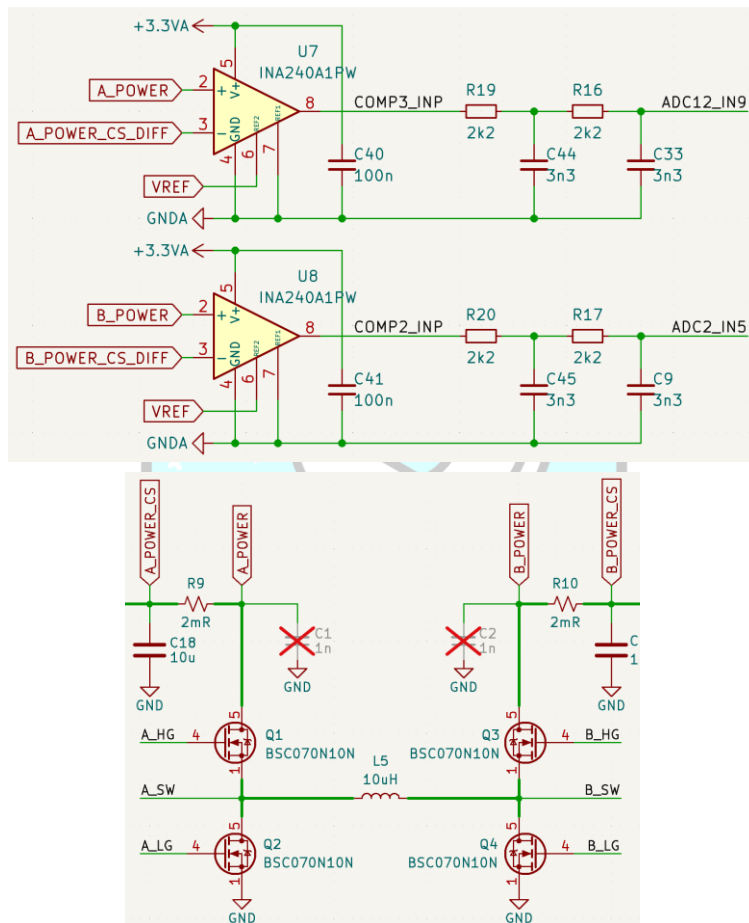
INA240 系列共有 A1-A4 四种型号，为了减小噪声，选择增益为 20V/V、标称带宽为 400kHz 的 A1 型号。通过数据手册中的增益-频率图可以看出在 1MHz 频率下 INA240A1 仍可以保持 10dB 的增益，由于峰值电流控制并不需要增益很准确的高频信号，故其可以稳定实现峰值电流控制，与测试结果相符。



电流采样电路

峰值电流控制模式需要对电感电流进行采样，使用上文提及的 $2\text{m}\Omega$ 电阻，在上板使用 INA240 电流感应放大器将采样电阻两端差分电压转化为单端电压输出。电感电流信号连接比较器进行峰值电流控制，同时经过二阶 RC 滤波得到平均电流并通过 ADC 采样以进行软件环路控制。

与将电感电流采样与平均电流采样分离的设计不同（如 LT8705 等），本方案的设计可以通过单一电流感应放大器同时获得电感电流与平均电流。这种做法很大程度上简化了电路，核心部分仅需两个电流感应放大器，并且无需引入额外的采样电阻，减小了损耗。



通过第一版硬件的测试发现一阶 RC 滤波的 ADC 采样结果噪音较大，所以在第二版中使用了二阶 RC 滤波，可以实现在截止频率后 -40dB/dec 的衰减。参考电压源分别连接到 2.9V 的参考电压和模拟地，实现 $1/2$ 参考电压的输出偏置以实现双向电流检测。具体理论设计参数如下：

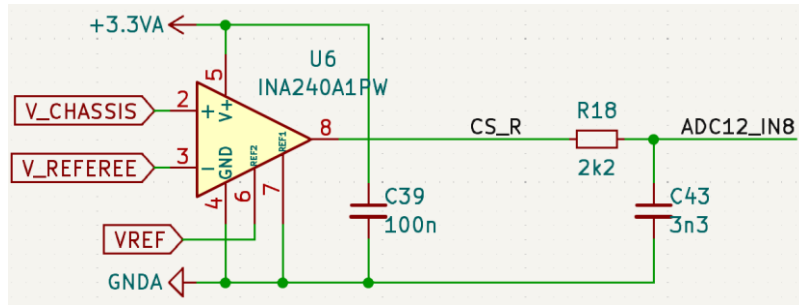
$$\text{直流缩放倍率: } \text{Gain}_{\text{DC}} = 2\text{m}\Omega \times 20\text{V/V} = 0.04\text{V/A}$$

$$\text{单向最大量程: } V_{\text{max}} = 0.5V_{\text{Reference}}/\text{Gain}_{\text{DC}} = 0.5 \times 2.9\text{V}/(0.04\text{V/A}) = \pm 36.25\text{V}$$

$$\text{二阶滤波截止频率: } f_{\text{cutoff}} \approx \frac{1}{2\pi RC} = \frac{1}{2\pi \times 2.2\text{k}\Omega \times 3.3\text{nF}} \approx 21.92\text{kHz}$$

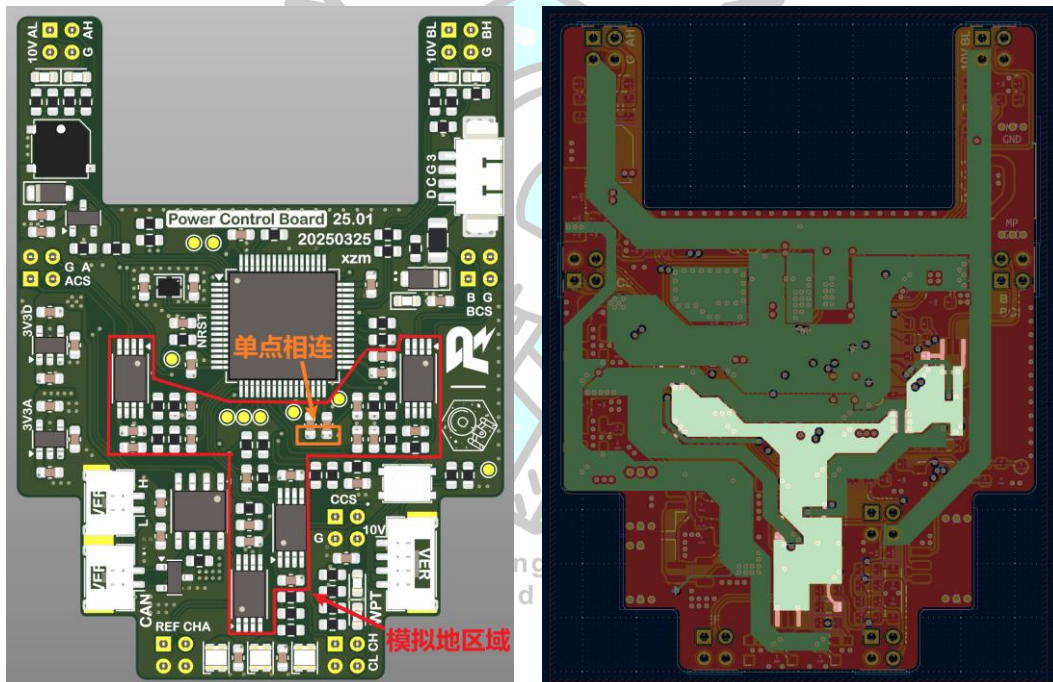
$$\text{ADC 理论分辨率: } 2V_{\text{max}}/4096 = 17.70\text{mA}$$

对于裁判系统电流采样，由于其采样直接为平均电流（采样电阻两侧均有滤波电容），噪声较小，故使用一阶 RC 滤波即可得到噪声较小的数据，其余参数与上文一致。

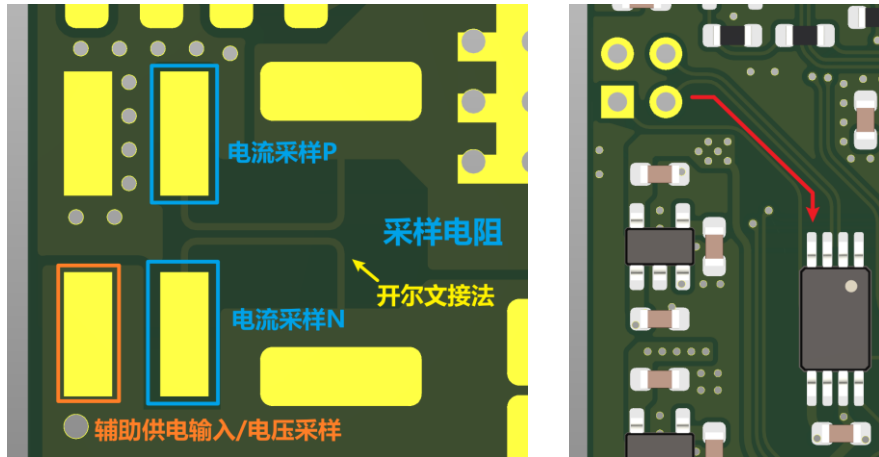


模拟电路 PCB 布局

由于超级电容的 PCB 上同时由敏感的模拟采样电路、数字电路和大功率的功率开关电路，对 PCB 布局由较高要求。首先电容控制器分为上下两板，将大功率地和数字地进行了分割；在上板上对模拟部分的地做了分割，并在 MCU 的 VDDA/VSSA 滤波电容处单点相连。



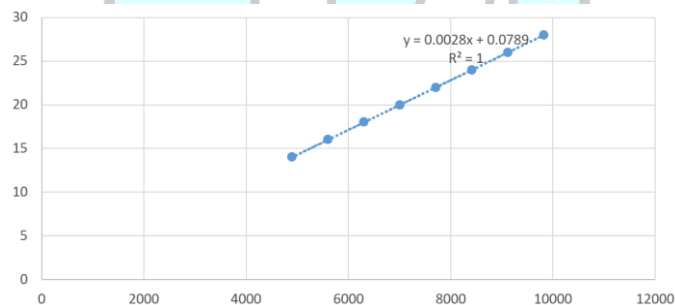
对于电流采样，差分信号线从采样电阻开始使用开尔文接法，通过排针来到上板并继续保证等长差分，同时保证其不与相同网络其他处有连接，保持由采样电阻到放大器完全独立。



对于电压采样，由于排针体积 Pin 数的限制，在板对板连接处辅助供电输入端共用同一引脚，在上板排针焊盘处分离，分别走向电压采样电路和辅助电压输入。

校准

尽管前文提到的各种设计已经尽量消除误差，但由于各种因素（例如 ADC 精度、温度、放大器误差等）导致的误差仍然不可忽略，所以使用一次函数对 ADC 采样结果与真实值对比进行拟合，实测可以极大提升采样精度。例如下图为采样电压的校准拟合：

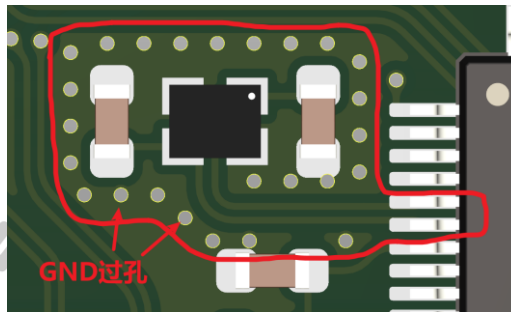


实际使用中对每个焊接的电容控制板的五个测量量分别进行校准，校准结果统一存储在 Calibration.hpp 文件中，并根据 MCU 内部的 96 位 UID 进行编号，保证任意电容控制板只需要进行一次校准且不会因为烧错代码使用错误的校准信息。

3.2.6 MCU 与控制

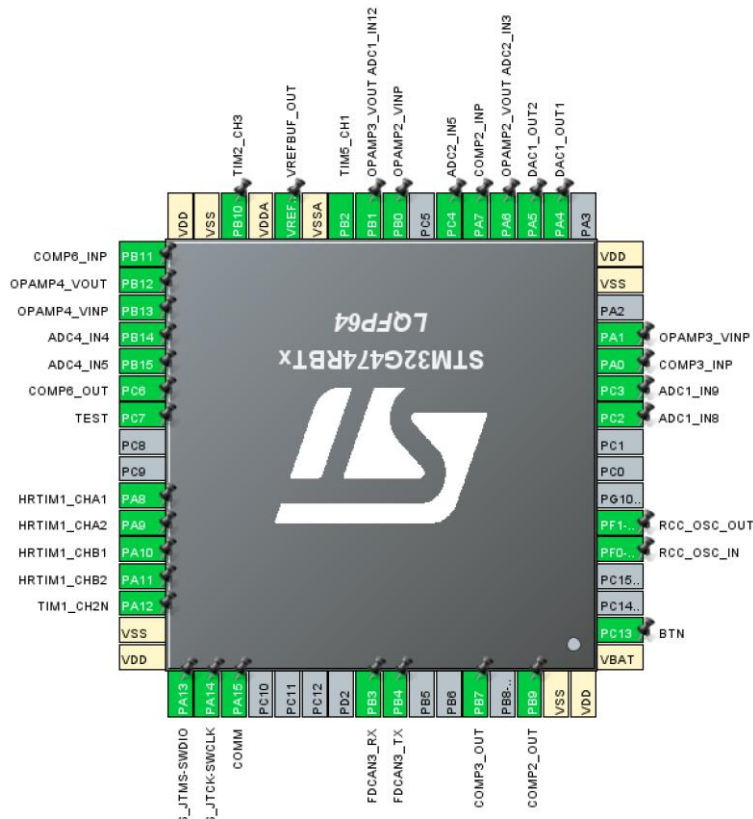
MCU 选用 STM32G474RBT3。STM32G474 提供了功能极为丰富的高精度定时器 (HRTIM V2)，使得峰值电流控制成为可能；170Mhz 的主频也使得环路计算可以更快；丰富的模拟外设 (电压参考、内置运放、比较器、DAC 等) 使得外围电路得到极大简化；同时，选用 RBT3 版本使得其耐温达到 -40°C 到 125°C，提高稳定性。

MCU 采用 LDO 供电，并在每个供电引脚配备 100nF 电容。晶振选用 2012 封装的 25Mhz 晶振以减小体积和干扰，PCB 上对晶振本身和其走线进行了包地处理、并在所有层禁止其他走线来防止干扰。



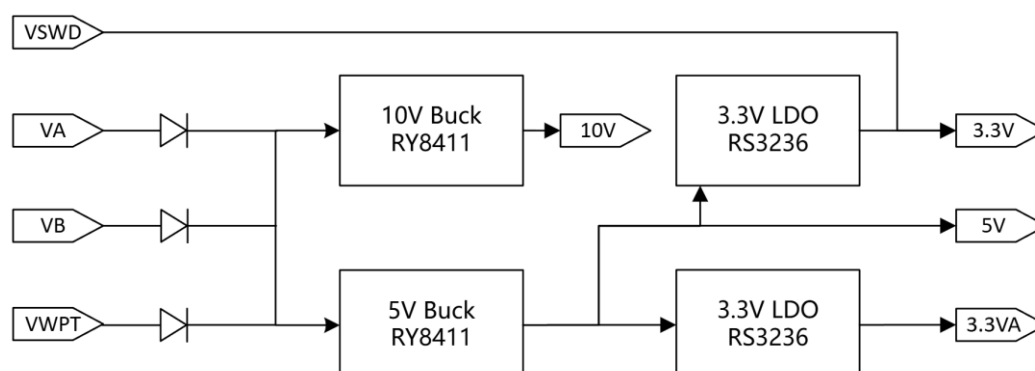
由于实际使用中对于 BOOT0 和 NRST 按钮需求极小，故移除独立按钮，仅保留下拉与上电复位电路。保留一个连接到 GPIO 的按钮进行交互。

STM32CubeMX 中的部分引脚配置如下 (不包含无线充电部分)：



3.2.7 辅助供电

电源框图



超级电容及无线充电系统共有三个电压源，分别为底盘端电压、电容组端电压、无线充电输入端电压。为了保证任意电压源有输入时系统都能工作，使用三个二极管分别从三个电压源指向辅助供电 Buck 的输入，因为辅助电源功率很小，使用普通肖特基二极管即可。同时为了烧录及调试方便，将 SWD 的 3.3V 输入通过保险丝和 TVS 保护后通入 3.3V 网络。

辅助电源电源网络

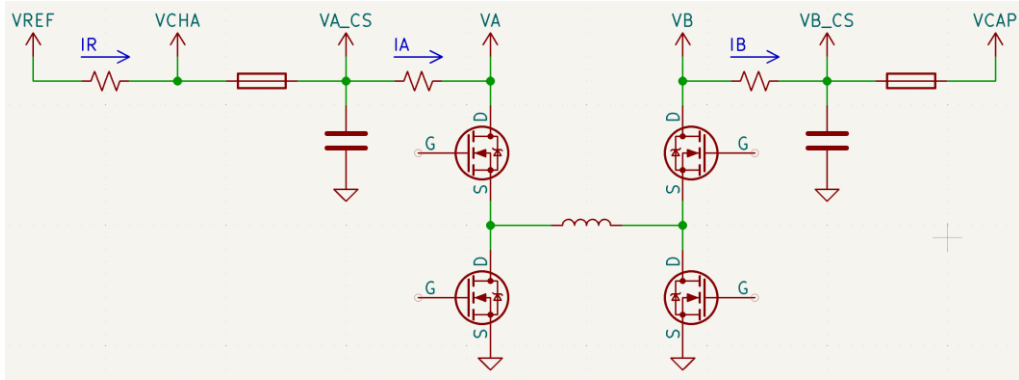
电源网络名称	描述	用电器
10V	MOS 栅极驱动电源	MOS 栅极驱动芯片
5V	5V 主供电	CAN 收发器、蜂鸣器、LDO 等
3.3V	3.3V 主供电	MCU、CAN 收发器
3.3VA	模拟 3.3V 供电	MCU VDDA、电流感应放大器

取电位置

The Hong Kong University of
Science and Technology

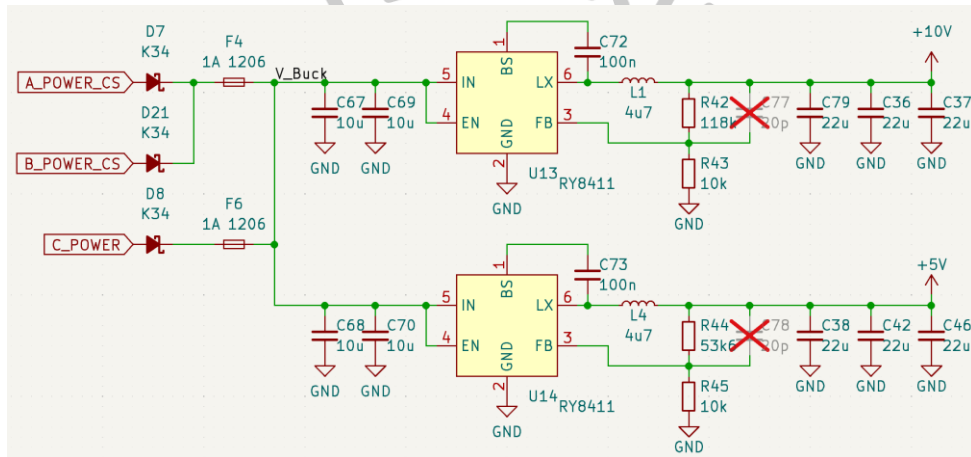
虽然辅助电源的功耗较小，但其电流仍然不可忽视，所以需要选取合适的取电位置来避免对功率级电流测量的干扰。如图为超级电容系统功率电电路的框图：VREF 为裁判系统电源管理模块的 Chassis 接口输出，通过裁判系统电流检测电阻后输出到底盘；底盘网络通过保险丝进入 Buck-Boost 电路的 A 侧，在通过两个电流检测电阻、功率级后再经过保险丝进入电容组。

实际辅助电源的 VA、VB 输入分别接入如同所示中的 VA_CS 和 VB_CS 网络。对于 A 侧，在 VA_CS 处取电可以避免辅助供电输入绕过裁判系统电流检测电路，使得测得的裁判系统功率不准。不在 VA 和 VB 处（采样电阻后）取电可以避免随功率级的电流采样造成干扰。

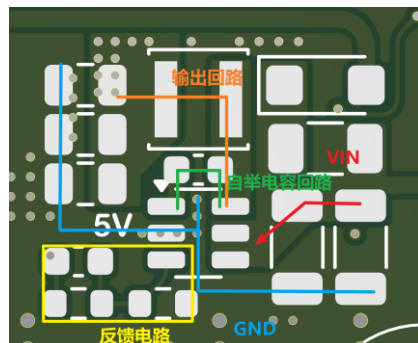
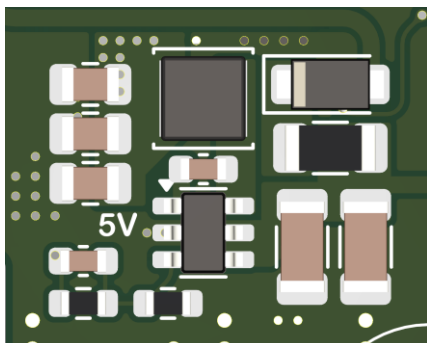


Buck 部分电路

Buck 芯片选用 RY8411。输入电压最高 42V，可以保证在功率级电压不超过 30V 的情况下有充足的余量；输出电流最大 1.2A，远超辅助供电的需求；同步整流可以提供较高的效率并且无需续流二极管、减小了电路面积。



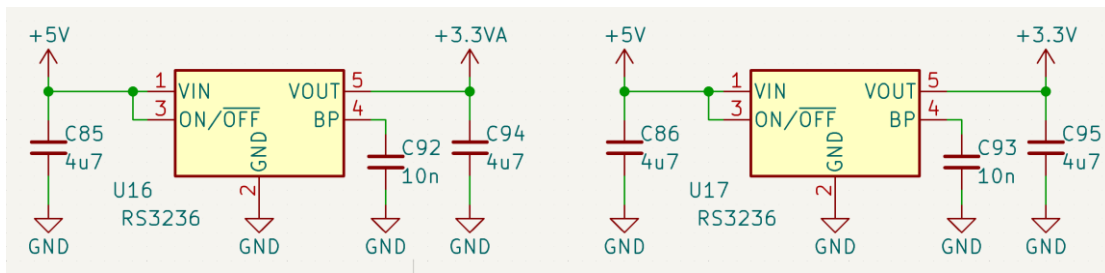
输入部分采用三个耐压 40V，最大电流 3A 的肖特基二极管 (K34)，并且在经过 1A 的保险丝后进入 Buck 芯片的输入；其中无线充电输入端 (C_POWER) 独立使用一个保险丝是为了布局方便。每个 Buck 芯片输入各使用两个 1206 封装耐压 50V 的 10uF 电容，输出各采用三个 0805 封装耐压 25V 的 22uF 电容，由于 X7R/X5R 材质的电容实际容值随着两端直流电压的升高而降低，所以选用高耐压电容来减小容量衰减。电感选用 3015 封装的 4.7uH 电感，虽然其 DCR 较大 (约为 100mΩ 级别) 且饱和电流较小 (约为 1.1A)，但对于辅助供电来说已经足够。



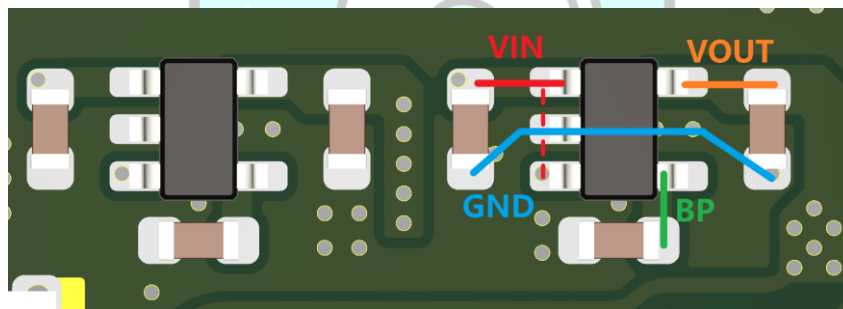
布局上将全部功率回路在顶层布局完成。保证输入电容 - 芯片 - 输出电容的 GND 回路阻抗较小，且输入电容距离芯片的 VIN 和 GND 尽可能近；自举电容高频回路紧挨芯片的 SW 和 BST 引脚减小干扰，电感布局保证 SW 的网络面积尽量小防止干扰；反馈回路在输出电容后通过内层独立走线连接至采样电阻处，且保证 FB 网络面积尽可能小避免干扰。

LDO 部分电路

LDO 选用 RS3236，输入电压最大 7.5V，输出电流最大 500mA，都有较大余量。电路基本参考数据手册，在 BP 引脚连接 10nF 电容来减小输出噪声。

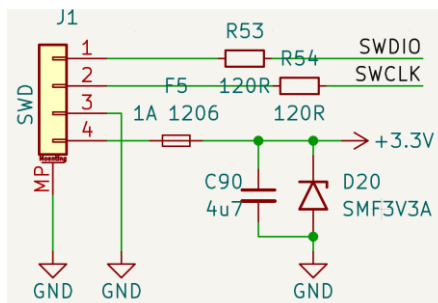


PCB 布局上依然在同一层完成了功率回路的走线，保证输入和输出电容的两端离芯片尽量近，为了防止 EN 引脚走线切割 GND，将其从内层接入芯片。



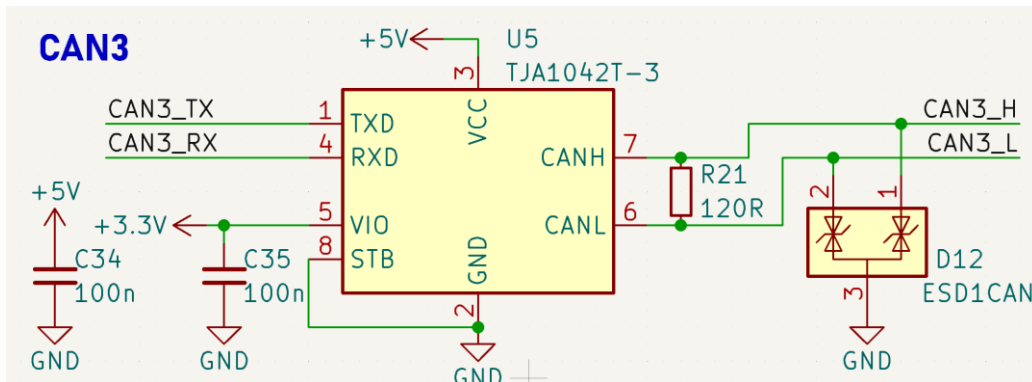
SWD 电源输入

由于电容控制器烧录及调试代码频率较低，对于 SWD 供电的处理和保护较为简单，通过 1A 保险丝和 3.3V TVS 进行防护，并使用一个 4.7uF 电容进行滤波。

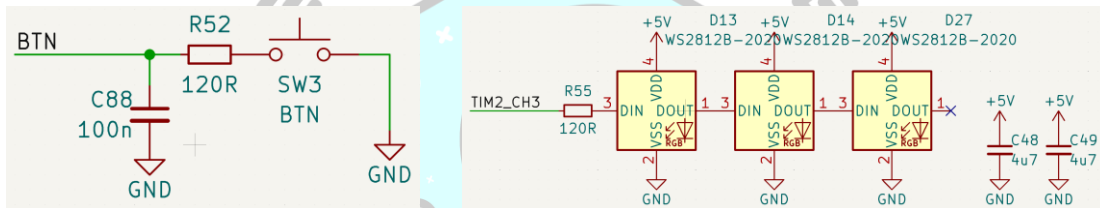


3.2.8 通讯与交互

通讯部分采用 CAN 协议，直接接入底盘的 CAN 网络并与主控板进行通讯。收发芯片选用 SIT1042T/3，输出采用 ESD1CAN 进行防护。



交互上使用一个按钮进行重启和清除故障等操作，通过 WS2812B 进行状态指示。按钮使用一个 RC 滤波电路来消抖，引脚使用 GPIO 内部上拉。WS2812B 选用 2020 封装减小体积。



3.2.9 成本控制

The Hong Kong University of Science and Technology

为了节约经费，超级电容控制器在设计上进行了最大程度的精简，例如前文提到的用 MCU 内部运放代替独立运放芯片等。同时，对于电阻或电容取值较宽松的地方进行了取值的统一，在不同电流采样处统一采样电阻和电流感应放大器的选型，最大程度简化 BOM。

本赛季考虑全阵容+备车+备件的需求大致需要 8 套电容控制器，使用嘉立创经济型 SMT 贴片核心板的电阻电容，其他元件和下板全部手焊，可以控制整套控制器成本在 100 元左右。

限制因素

对于裁判系统 R 端 (0V 或电池电压) :

- 比赛规则限制此接口输出的功率上限, 功率计算方式为 R 端的电压 \times 电流, 其中电压为电池电压, 此功率上限一定不能超过, 否则会导致判罚。
- 由于裁判系统电源管理模块无法检测负功率 (不会返还反向流入裁判系统的能量), 所以理想情况下要保持此接口功率一直为正, 即保证此接口只会输出能量。
- 虽然有功率限制的存在, 但理想情况下控制器需要有完全吃满裁判系统功率限制的能力以保证性能。

总结: R 端一定要保证不超过功率限制, 最好保证功率不倒灌, 最好可以吃满功率, 电压输入范围为电池电压

对于底盘端 (与 R 端电压一致) :

- 此端口功率受上位机控制的电机决定。
- 此端口的瞬时功率较大, 例如电机急停时会有巨大的反冲功率灌回系统中, 系统必须反冲有较强的处理能力, 否则其将不受控制地流回电池, 造成浪费。

总结: 底盘端会承受较大瞬时功率, 且系统无法控制接口的输出输入能量

对于电容组端 (0V-电容组最大电压) :

- 此端口电压绝对不能超过电容组的最大电压, 需要严格限制并保护。
- 由于此接口与裁判系统电容管理模块 CM01 直接串联, 此端口的电流不能超过 CM01 的电流限制 ($\pm 15A$) 。
- 此接口电流不能超过电容组最大充放电电流, 并且在电容电压较低时需要进一步限制电容组的充放电电流。由于电容组的最大电流一般很大 (30A 级别), 远大于 CM01 的限制, 所以这里只需要考虑低电压限制电流的情况

总结: 电容端需要有严格的电压电流限制, 并且在电容组电压低时要进一步限制电流

对于功率级本身, 任意时刻电感电流不能超过饱和电流 (约 30A 级别)

工况分析

	底盘需求功率低	底盘需求功率高
电容能量低	快速充电	提供尽量多的功率, 但优先保护系统本身不会损坏
电容能量高	保持电容能量	提供尽可能高的爆发功率

除了一般工况, 还有一些特殊情况需要考虑:

- 正常运行时电机可能会有反向功率 (如刹车或下坡时), 此时系统理想情况下可以回收这部分功率, 并且电容要留有余量来接收这部分能量。
- 即将无线充电时, 为了充入尽量多的能量, 需要将电容能量放电至较低水平, 具体做法为令底盘优先消耗电容组中的能量 (闭环裁判系统功率为一个较低的值)。

控制方式实现

假设 Buck-Boost 效率较高并可以忽略其他损耗, 那么三端口能量守恒, 定义 R 端输入为正, Chassis 端输出为正, Cap 端输出为正:

$$P_R = P_{Chassis} + P_{Cap}$$

由于 R 端和 Chassis 端电压一致 (忽略采样电阻压降), 且都为电池电压:

$$V_{BAT}I_R = V_{BAT}I_{Chassis} + V_{Cap}I_{Cap} \rightarrow V_{BAT}(I_R - I_{Chassis}) = V_{Cap}I_{Cap}$$

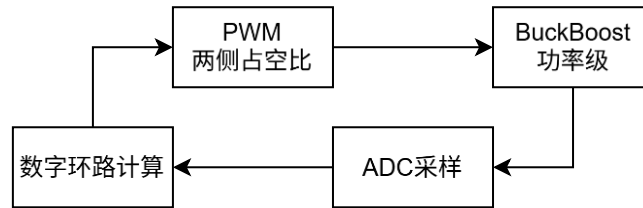
其中电池电压 V_{BAT} 由电池电量决定 (22V-26V 左右), 不可控; 电容组电压 V_{Cap} 取决于电容组的能量, 也不可控; $I_{Chassis}$ 由电机需求决定, 不受电容控制器控制。所以在底盘功率需求波动时 (即 $I_{Chassis}$ 波动时), 需要根据需求控制 I_{Cap} 来保证 I_R 不超过 P_{Limit}/V_{BAT} , 且最好一直为正。同时, 控制器也需要保证电容组电压 V_{Cap} 不超过一个限定值。

3.3.3 纯数字控制模式分析

本部分主要参考队内 2024 赛季的开源：

【RM2024 超级电容控制器硬件+软件完全开源】（史上最详细的开源！）香港科技大学 ENTERPRIZE 战队 <https://bbs.robomaster.com/article/54127?source=4>

控制逻辑框图



功率级分析

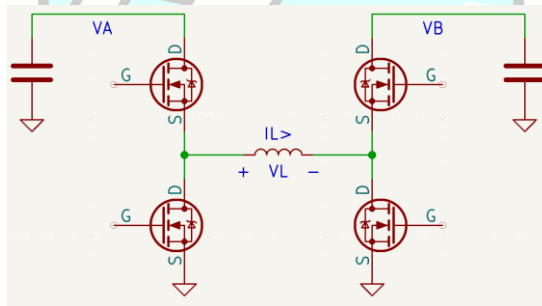
对于同步四开关 Buck-Boost，两侧的占空比直接作用于两侧电压的比值：

假设两侧占空比为 D_A 和 D_B ，电压为 V_A 和 V_B ，下管不会同时导通，工作在 CCM 模式下，电路共有三个工作状态，根据伏秒平衡：

$$(V_A - V_B)(D_A + D_B - 1)T + V_A(1 - D_B)T + (-V_B)(1 - D_A)T = 0V \cdot s$$

$$\rightarrow V_A D_A = V_B D_B$$

其中 D_A 和 D_B 取值可以为 1（分别对应 Boost 和 Buck 模式）



根据不同的占空比分配方式，例如持续处于 Buck-Boost 模式或根据占空比需求在 Buck Buck-Boost 和 Boost 模式间进行切换，可以直接通过一个输入量控制电压比值

$$D = V_B / V_A = D_A / D_B$$

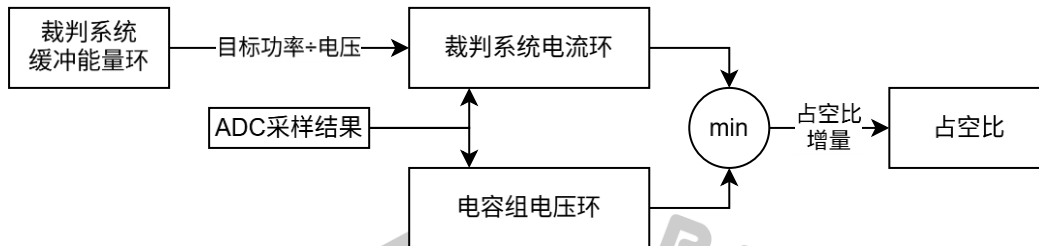
当需要控制电压时，可以直接通过增量式 PID 控制目标占空比。

当需要控制电流时，由于电路各处寄生电阻的存在，较小幅度地控制占空比即可控制电流。

控制逻辑简述

纯数控超级电容控制器的核心逻辑是裁判系统功率（电流）环与电容组电压环进行竞争，并进行增量式控制。

具体实现上，两个控制环路使用增量式 PID 实现，裁判系统电流环的目标值为“裁判系统功率限制÷底盘电压”，结果在“功率级 A 侧电流”的目标值上累加，电容组电压环 1 目标值为“电容组最大电压”，当前值为“电容组当前电压”，在分别进行计算后各自累加，并取最小值进行输出，并将结果累加到当前占空比上。



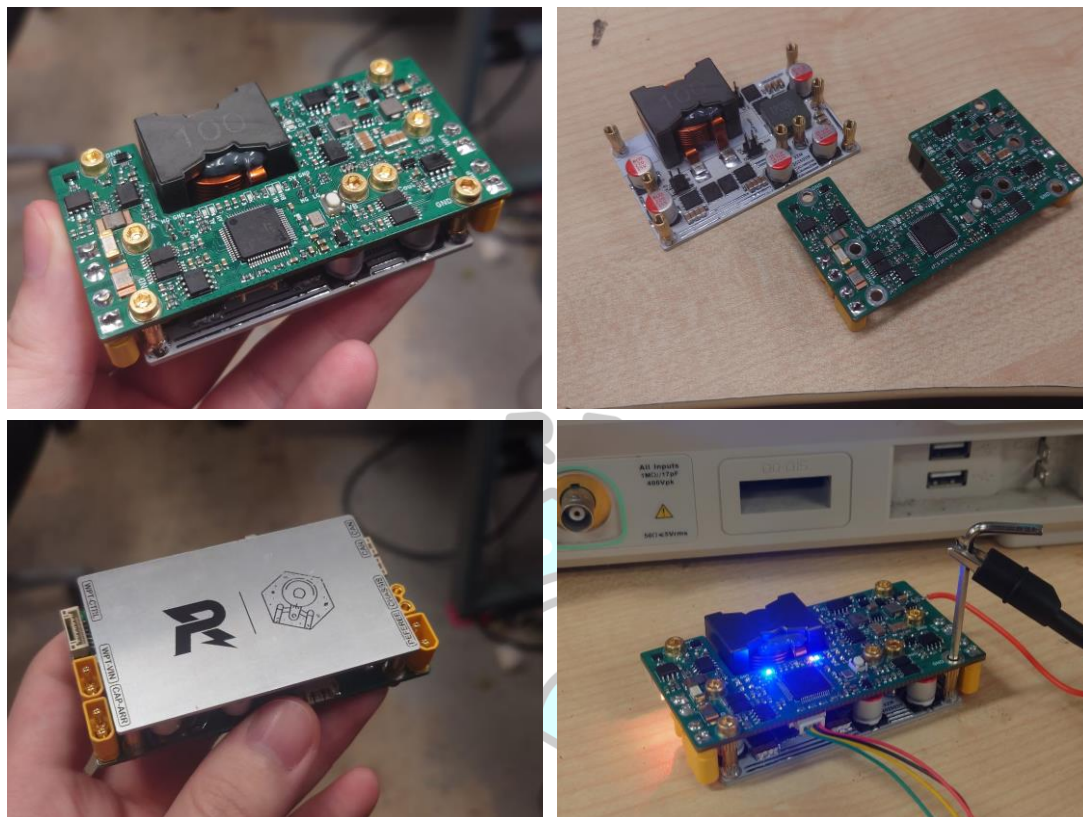
为了保证裁判系统功率的准确，最大化系统输入功率，引入了裁判系统缓冲能量环来时“校准”目标功率值。

优势与不足

数控超级电容系统的主要优势是软件、硬件设计都较为简洁，控制逻辑较为简单，并且通过做好保护很容易实现很高的稳定性。例如队内 24 赛季开源的超级电容控制器在场上从未出现故障。并且纯数控方案调试比较容易，仅需做好软硬件的配置，并调整软件环路参数即可达到较好的控制效果。

然而，纯数控方案在现有基础上提升响应速度基本只能靠提升环路速度来实现，而加快环路速度极大地受到 MCU 运算速度的影响，例如在本赛季初经过测试，在保留最小程度的保护功能的前提下，STM32G474 可以实现开关频率 1/2 周期的环路运算速度（约 136kHz），但此时中断负载极高，使得其它功能的代码的运行受到极大干扰。数控方案提升环路速度同样受到采样带宽的限制，更快的环路运算速度需要提升采样带宽，由于需要采样的量为平均电流，提升采样带宽也会导致噪声和误差的增大。

综上所述，新赛季的规则间接对超级电容系统的响应速度提出了较高要求，在本赛季初进行了一系列尝试后，发现数控方案提升响应速度较难，例如下图展示了本人在赛季初设计的第二版数控电容的测试版（包括无线充电部分电路）



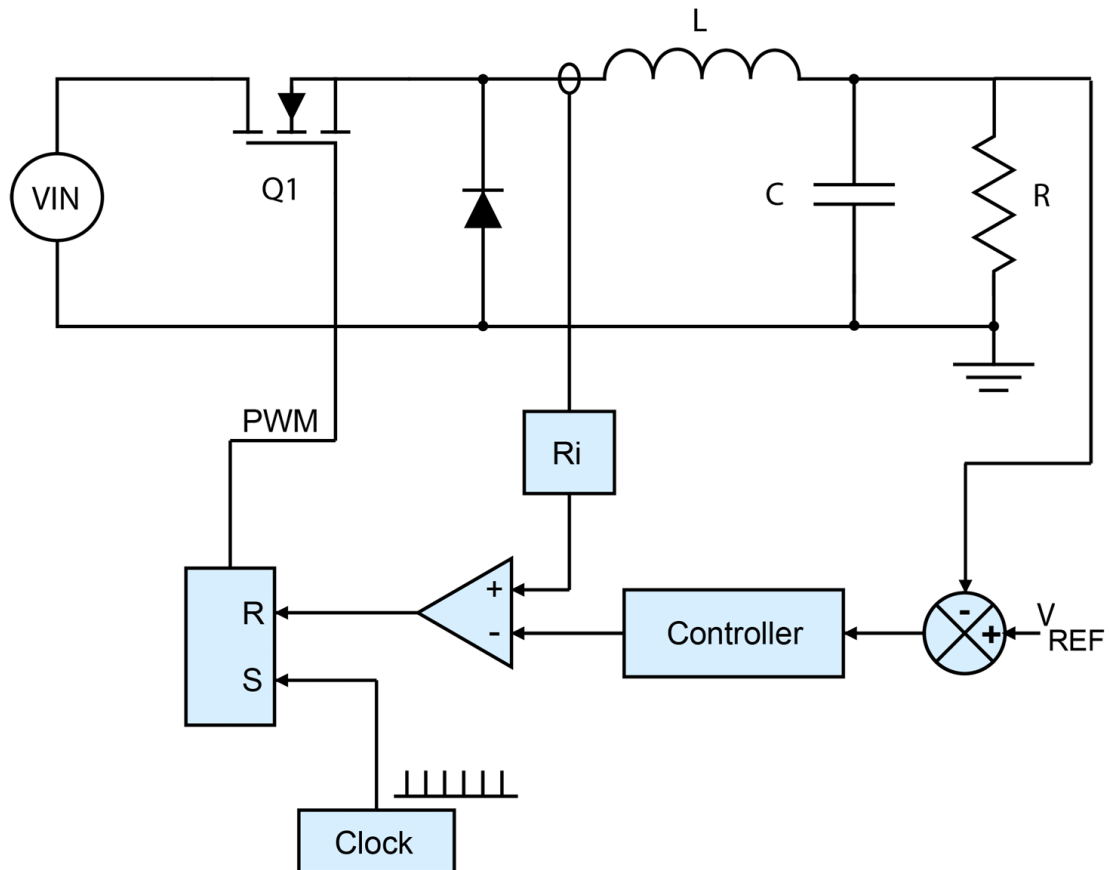
所以，在尝试了解更多 DC-DC 控制模式相关知识后，决定对峰值电流模式控制进行探索与尝试，通过底层控制方式的改变来达到提升响应速度的目标。

3.3.4 峰值电流控制模式介绍

峰值电流模式 (Peak Current Mode Control) 的核心是直接将电流波形用作 PWM 生成比较器中。本部分将以 Buck 变换器为例，简要分析峰值电流控制模式。

本节主要参考了 STM32 的应用手册 AN5497: Introduction to the buck current mode with the B-G474E-DPOW1 Discovery kit

典型电路框图 (截取自 AN5497)



控制方式

如上图所示，控制器最终目的为输出恒定电压，并由电压外环控制峰值电流内环，由于在 RoboMaster 竞赛超级电容系统的应用场景下，大多数时间都在控制电流，所以本部分将主要关注电流内环的实现方式。

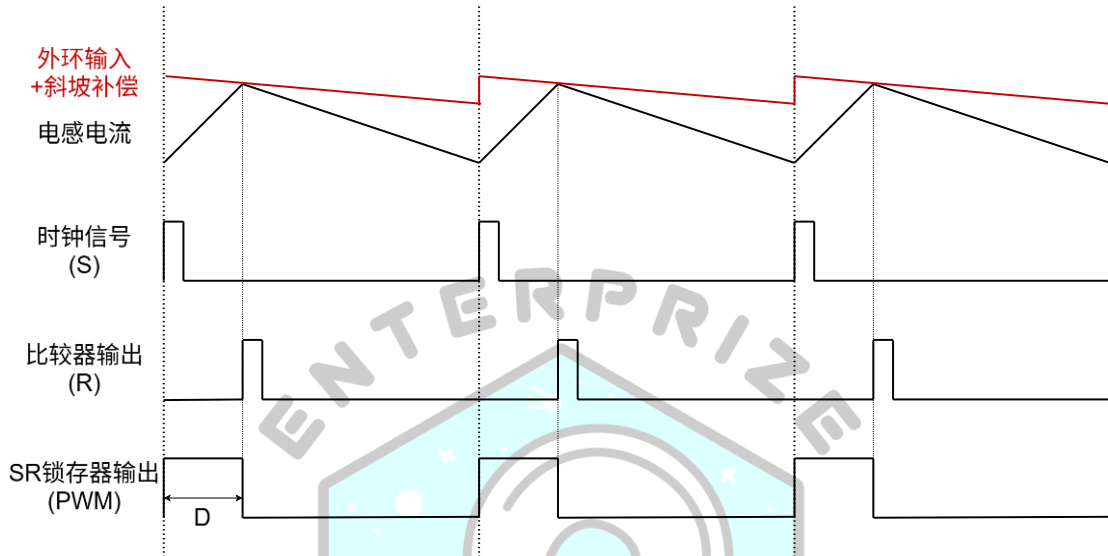
峰值电流控制硬件上最大的不同是加入了电感电流采样。电流采样值输入到比较器的一个输入，另一个输入由外环决定；比较器的结果输出到 SR 锁存器的一个输入，另一个输入为固定频率的时钟源。

具体运行方式如下，参考时序图：

- 周期开始，时钟信号触发 SR 锁存器的 Set，SR 锁存器输出为高，Buck 功率级的上管

打开，电感电流开始上升。此时电流斜率为 $(V_{in} - V_{out})/L$

- 当电感电流上升到目标峰值（由外环提供）比较器输出为高电平
- 比较器输出为高电平，触发 SR 锁存器的 Reset，此时 SR 锁存器输出为低，上管关闭
- 上管关闭后，下管打开（或续流二极管导通），电感电流开始下降，此时电感电流下降斜率为 V_{out}/L ，直到周期结束



主要优势

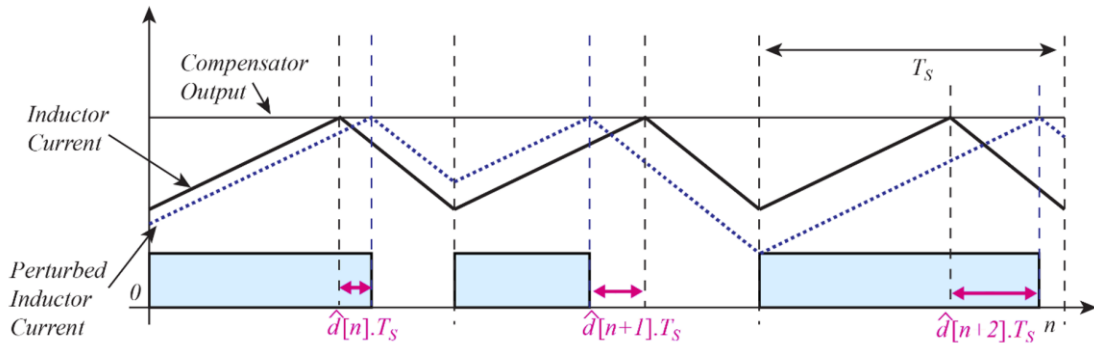
峰值电流模式最大的优势是电流响应速度极快。由于占空比不是由控制器决定，而且直接由导通时间内电感电流的上升决定的，只要阶跃值小于电流纹波，电感峰值电流可以在单周期内达到目标峰值。而根据电感大小、开关频率和占空比可以计算电感电流的峰峰值 I_{pp} ，通过 $I_{peak} = I_{average} + 0.5 \times I_{pp}$ ，可以通过目标平均电感电流计算出对应需要设定的峰值电流目标，也就是说外环可以直接通过设定比较器负输入的电压来直接控制电感电流。

峰值电流模式的另一优势是可以严格限制峰值电流，从而最大程度减小过流风险。通过上段相同的方法，由于 I_{pp} 一般不会剧烈变化且可估算，所以可以通过此方法限制正向或反向电流，适用于电流双向流动的场景。

由于峰值电流环的运行只依靠硬件，无需任何软件的参与，这可以节省非常多计算资源，这使得超级电容控制器拥有了一个无需很多计算资源(仅需进行模式切换)且响应极快的内环。

次谐波震荡

虽然峰值电流控制带来了极快的响应，但也为控制带来了更多挑战。其中最显著的问题是次谐波震荡。如下图所示(截取自 AN5497)，当上升端斜率 < 下降段斜率时，电感电流波形会发散，并使得占空比抖动，最终呈现为“大小波”的现象，即一个周期占空比极大，下一个周期占空比极小。对于上文所示架构的 Buck 电路，理论上此现象会在占空比大于或等于 50%的情况下。



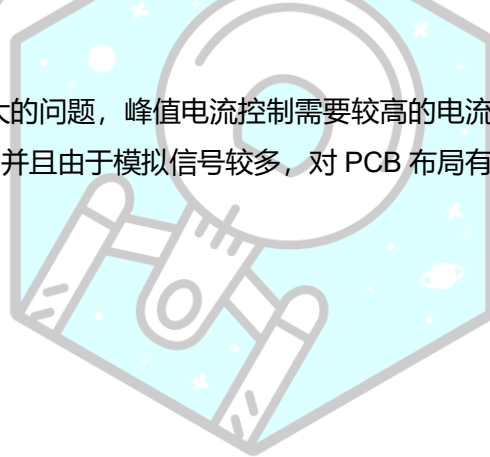
为了解决此问题，需要引入“斜坡补偿”，即通过给比较器输入增加斜率来补偿上升斜率过小的问题，且占空比越大，需要的斜坡斜率越大。

电流准确度

虽然如上文提到，峰值电流与平均电流的关系可以通过理论计算得到，但是叠加斜坡补偿斜率和噪音的影响，使得计算出的值与实际相差较大。对于一般的 DC-DC 变换器，峰值电流限制不需要很准确，但是在超级电容的场景下需要较为准确地闭环电流，所以需要引入数字外环来通过 ADC 采样出的平均电流对内环进行控制（或校准）。

其他问题

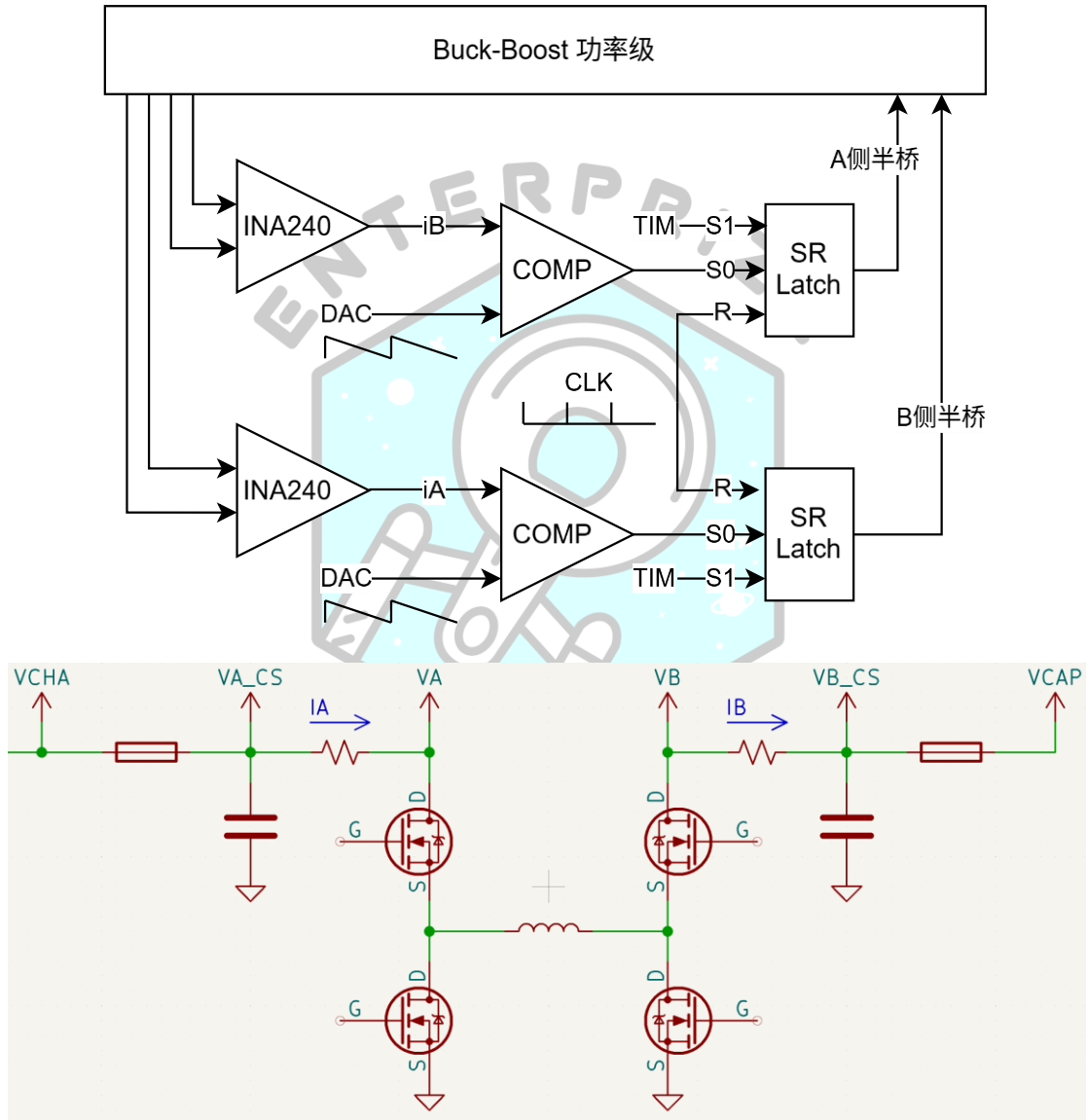
除了上文提到的影响较大的问题，峰值电流控制需要较高的电流测量带宽（如上文提到的 INA240A1 选型部分），并且由于模拟信号较多，对 PCB 布局有更高的要求。



3.3.5 峰值电流 Buck-Boost 硬件架构

对于 RoboMaster 超级电容控制器场景下，要实现电压升降与电流的双向控制，结合上文提到的 Buck-Boost 架构，需要详细分析需求来降低峰值电流控制的不足带来的影响并且最大程度发挥其优势。在本设计中通过调整 SR 锁存器、比较器处的逻辑，最大化降低了斜坡补偿的需求，并保证了不同模式间的平滑切换。

控制架构



如上图所示，对于 Buck-Boost 架构的控制，使用两套峰值电流环路，使用 INA240A1 作为电流感应放大器进行高位采样，采样电阻位于上管 D 极与输入/输出之间。当上管导通时，通过采样电阻的电流即为电感电流。

每个 SR 锁存器有一个 Reset 源和两个 Set 源，Reset 源被固定频率的时钟触发，S0 被比

较器触发，S1 的作用是做占空比保护或屏蔽比较器输入从而实现固定占空比或上管常开。具体实现方式将在下文“3.3.7 MCU 外设配置”部分详细说明。

接下来将通过分析不同升降压模式下的时序图来详细介绍控制方式，整个功率级一直工作在 CCM 模式下。

最小化斜坡补偿

对于斜坡补偿的需求，由于理论上至少有一半的占空比会出现次谐波震荡，所以最好可以令功率级工作在理论上无需斜坡补偿的区间（50%-200%）（详见“3.3.6 时序图”），以此减小斜坡补偿的需求和影响。

控制器 A 侧的电压为电池电压，B 侧为电容组电压，电容组的能量计算方式为：

$$E = 0.5 \times C \times U^2$$

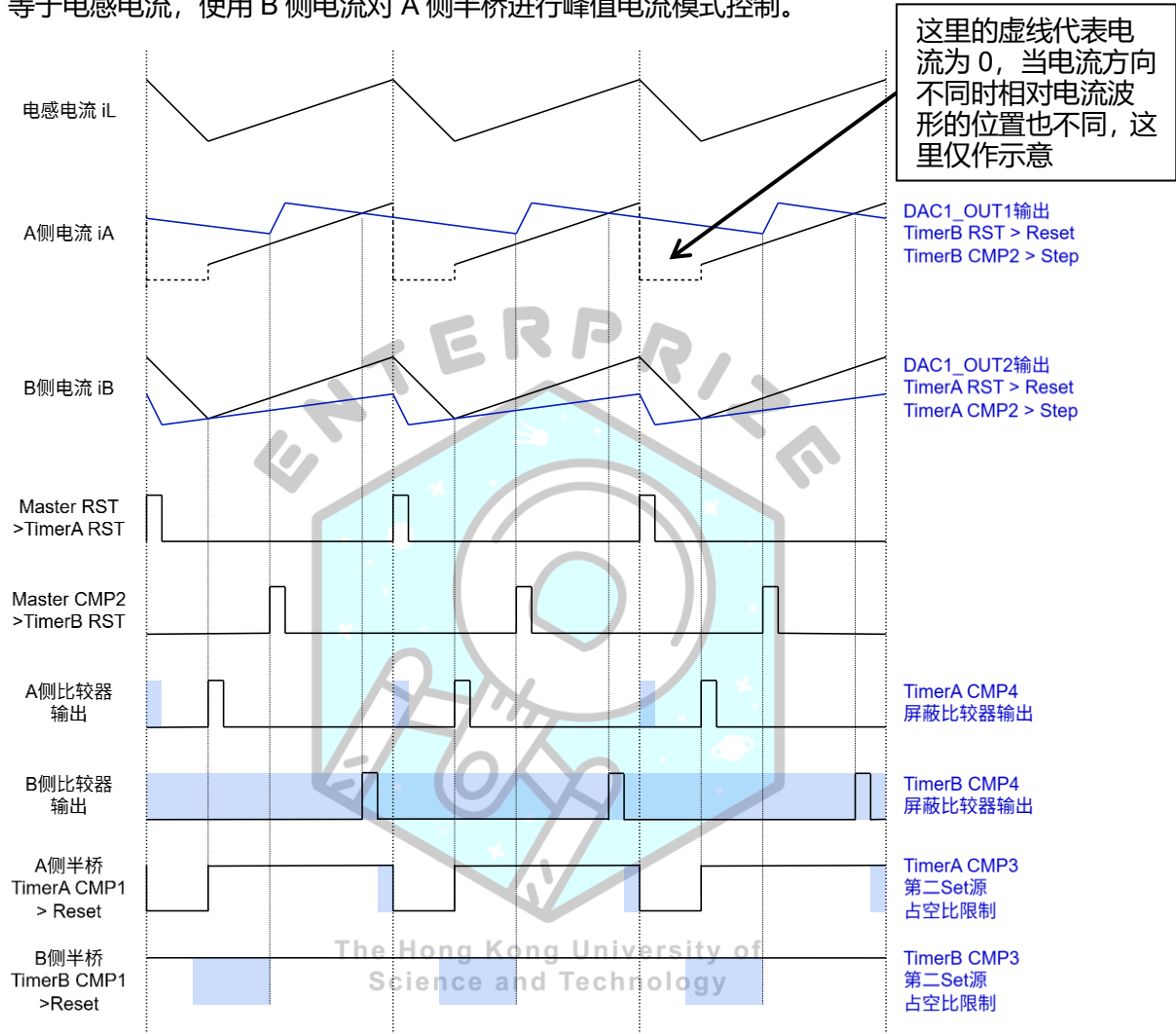
假设电容组最高电压为 29V（详见“3.6 超级电容组设计”），那么电容组在 10V 时仅存有 11.9%的能量，且此时受到电压限制，放电功率极大受限，所以在电容控制器控制算法中设定 10V 为低压保护的阈值，电容组电压小于 10V 时控制器会逐步限制电容组电流至很低，在 5V 左右彻底关断输出。

所以，可以认为电容组绝大多数时间都工作在 10V-29V 的区间，假设电池电压在 21V-26V 的区间内，可以算出最小占空比为 38.5%，最大占空比为 138.1%，大部分覆盖了理论上无需斜坡补偿的区间，最小化了斜坡补偿的需求（仅需坡度较小的补偿），保证了在此区间内峰值电流控制的稳定。

3.3.6 时序图

Buck 模式 (iB 谷值电流)

Buck 模式下, A 侧半桥以 250kHz 频率开关, B 侧上管常开, 此时 B 侧高位采样电阻电流等于电感电流, 使用 B 侧电流对 A 侧半桥进行峰值电流模式控制。

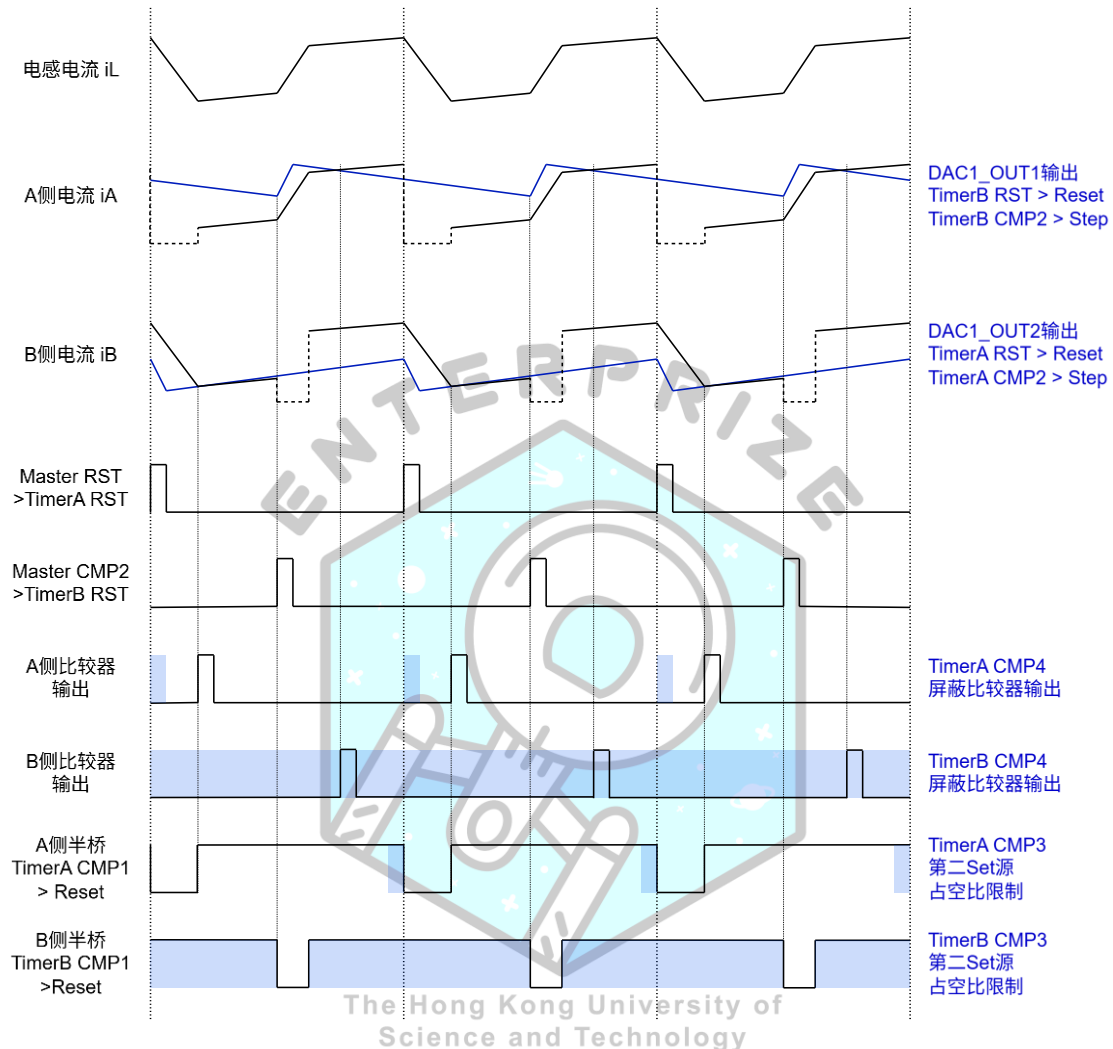


如图所示, 因为 Buck 模式下只用到了 B 侧电流控制 A 侧半桥, 所以对 A 侧电流对应的 B 侧比较器的输出进行屏蔽; 同时将第二 Set 源 (上一节中的 S1) 触发时间设置为 0, 以实现 B 侧半桥上管常开。具体运行过程分为两阶段:

1. Master 复位事件同时触发 TimerA 的复位事件, TimerA CMP1 (=0) 触发 A 侧半桥和 DAC Reset, 电感电流开始下降, 电流斜率绝对值为 V_{out}/L
2. 电流下降到比较器阈值触发 A 侧比较器, 进而触发 A 侧半桥 Set; 如果比较器触发过晚 (超过 99%周期), 则会由 TimerA CMP3 触发 Set, 防止占空比过小。触发后电感电流开始上升, 电流斜率绝对值为 $(V_{in} - V_{out})/L$

Buck-Boost ($V_{in} > V_{out}$) 模式 (i_B 谷值电流)

Buck-Boost ($V_{in} > V_{out}$) 模式下, AB 侧半桥都以 250kHz 频率开关, 且周期相位相错 180° , 当 B 侧上管打开时, B 侧高位采样电阻电流等于电感电流, 使用 B 侧电流对 A 侧半桥进行峰值电流模式控制, 同时 B 侧保持固定占空比。

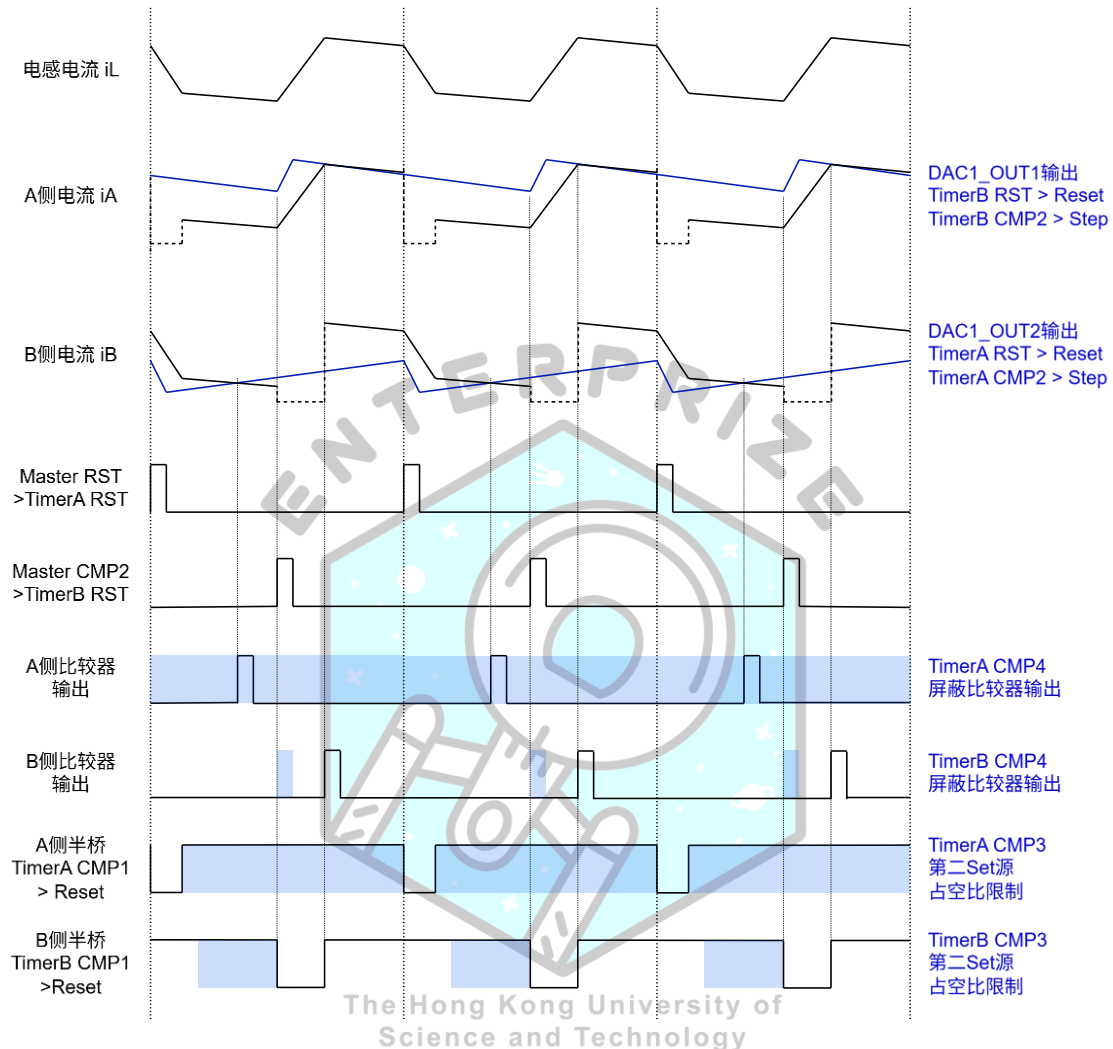


如图所示, Buck-Boost ($V_{in} > V_{out}$) 模式下将 B 侧半桥设置为固定占空比, A 侧半桥使用 B 侧电流进行控制。通过屏蔽 B 侧比较器的输出, 并把 TimerB CMP3 设置为在 16%周期时触发以实现 B 侧 84%固定占空比输出。具体运行过程分为四阶段:

1. Master 复位事件同时触发 TimerA 的复位事件, TimerA CMP1 ($=0$) 触发 A 侧半桥和 DAC Reset, 电感电流开始下降, 电流斜率绝对值为 V_{out}/L
2. 电流下降到比较器阈值触发 A 侧比较器, 进而触发 A 侧半桥 Set; 触发后电感电流开始上升, 电流斜率绝对值为 $(V_{in} - V_{out})/L$
3. Master CMP2 在半周期处触发 TimerB 复位事件, 进而触发 B 侧半桥 Reset, 电感电流开始上升, 电流斜率绝对值为 V_{in}/L
4. TimerB CMP3 触发 B 侧半桥 Set, 电感电流上升, 电流斜率绝对值为 $(V_{in} - V_{out})/L$

Buck-Boost ($V_{in} < V_{out}$) 模式 (i_A 峰值电流)

Buck-Boost ($V_{in} < V_{out}$) 模式下, AB 侧半桥都以 250kHz 频率开关, 且周期相位相错 180° , 当 A 侧上管打开时, A 侧高位采样电阻电流等于电感电流, 使用 A 侧电流对 B 侧半桥进行峰值电流模式控制, 同时 A 侧保持固定占空比。

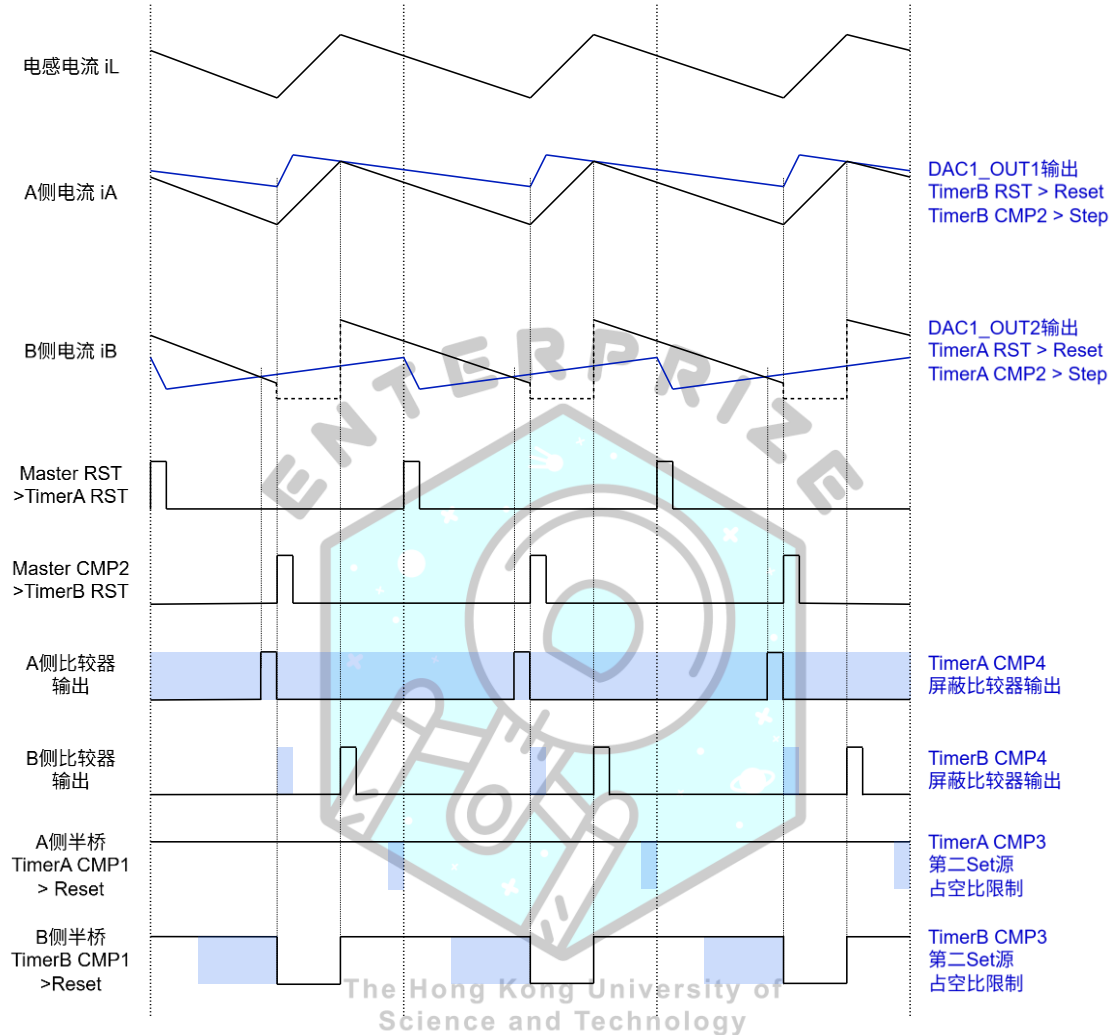


如图所示, Buck-Boost ($V_{in} < V_{out}$) 模式下将 A 侧半桥设置为固定占空比, B 侧半桥使用 A 侧电流进行控制。通过屏蔽 A 侧比较器的输出, 并把 TimerA CMP3 设置为在 16%周期时触发以实现 A 侧 84%固定占空比输出。具体运行过程分为四阶段:

1. Master 复位事件同时触发 TimerA 的复位事件, TimerA CMP1 ($=0$) 触发 A 侧半桥和 Reset, 电感电流开始下降, 电流斜率绝对值为 V_{out}/L
2. TimerA CMP3 触发 A 侧半桥 Set, 电感电流下降, 电流斜率绝对值为 $(V_{out} - V_{in})/L$
3. Master CMP2 在半周期处触发 TimerB 复位事件, 进而触发 B 侧半桥和 DAC Reset, 电感电流开始上升, 电流斜率绝对值为 V_{in}/L
4. 电流上升到比较器阈值触发 B 侧比较器, 进而触发 B 侧半桥 Set; 触发后电感电流开始下降, 电流斜率绝对值为 $(V_{out} - V_{in})/L$

Boost 模式 (iA 峰值电流)

Boost 模式下, V 侧半桥以 250kHz 频率开关, A 侧上管常开, 此时 A 侧高位采样电阻电流等于电感电流, 使用 A 侧电流对 B 侧半桥进行峰值电流模式控制。

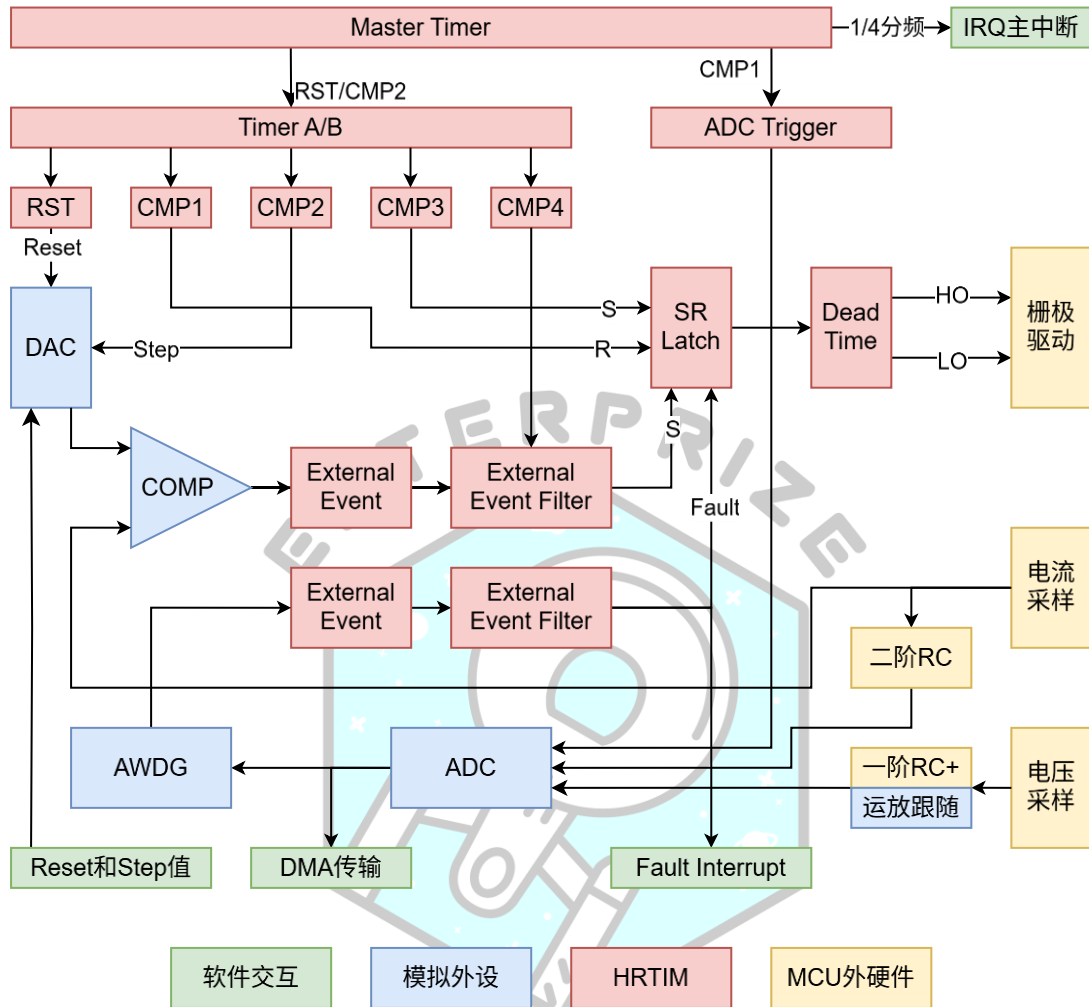


如图所示, 因为 Buck 模式下只用到了 B 侧电流控制 A 侧半桥, 所以对 A 侧电流对应的 B 侧比较器的输出进行屏蔽; 同时将第二 Set 源 (上一节中的 S1) 触发时间设置为 0, 以实现 B 侧半桥上管常开。具体运行过程分为两阶段:

1. Master CMP2 在半周期触发 TimerB 的复位事件, TimerB CMP1 ($=0$) 触发 B 侧半桥和 DAC Reset, 电感电流开始上升, 电流斜率绝对值为 V_{in}/L
2. 电流上升到比较器阈值触发 B 侧比较器, 进而触发 B 侧半桥 Set; 如果比较器触发过晚 (超过 55%周期), 则会由 TimerA CMP3 触发 Set, 防止占空比过小导致 Boost 电压过高。触发后电感电流开始下降, 电流斜率绝对值为 $(V_{out} - V_{in})/L$

3.3.7 MCU 外设配置

外设配置框图



HRTIM 配置

The Hong Kong University of Science and Technology

STM32G474 的 HRTIM 可以提供 MCU 主频 32 倍的倍频速度，即 $170\text{MHz} \times 32 = 5.44\text{GHz}$ ，分辨率可达 184ps，非常有助于在固定占空比模式下输出精确的占空比。HRTIM 输出的本质为一个多输入的 SR 锁存器，是实现峰值电流控制的关键。

为了使开关频率为 250kHz，将 Master Timer 和 Timer A、Timer B 的周期配置为 21760。软件环路计算速度设定为 4 开关周期一次，所以将 Master Timer 的 Repetition Counter 配置为 4-1，并启用全局中断，在代码中可以在对应的 IRQ_Handler 中执行环路计算和保护等功能的代码。

Time Base Setting

Prescaler Ratio	HRTIM Clock Multiplied by 32 (HRTIM Clock...)
fHRCK Equivalent Freque...	5.44E9 Hz
Period	21760
Resulting PWM Frequency	250000 Hz
Repetition Counter	4-1
Mode	The timer operates in continuous (free-runni...

为了实现 Buck Boost 两侧半桥交错输出, 需要将 Timer A 和 Timer B 的相位相错。将 Timer A 的复位源设置为 Master Timer 的复位事件, 将 Timer B 的复位源设置为 Master Timer CMP2, CMP2 的值设置为周期一半, 这样就实现了两个子 Timer 的 180°交错输出。

Update Trigger Sources Selection ...	1
1st Update Trigger Source	Register update is triggered by the master timer update
Reset Update	Update by Timer reset / roll-over disabled
Resynchronized Update	Update taken into account immediately
Reset Trigger Sources Selection : ...	1
1st Reset Trigger Source	The timer counter is reset upon master timer period event
Interrupt Requests Sources Select...	0
Number of Timer A Internal DMA R...	0

Update Trigger Sources Selection : Please en...	1
1st Update Trigger Source	Register update is triggered by the master timer update
Reset Update	Update by Timer reset / roll-over disabled
Resynchronized Update	Update taken into account immediately
Reset Trigger Sources Selection : Please ent...	1
1st Reset Trigger Source	The timer counter is reset upon master timer Compare 2..
Interrupt Requests Sources Selection : Pleas...	0
Number of Timer B Internal DMA Request So...	0

虽然两个子 Timer 的相位不同, 但为了保证其寄存器值更新的同步, 将两个 Timer 的 Update 触发源都设置为 Master Timer 的复位事件。

CMP1 和 CMP3

CMP1 用于在每个周期开始时对 SR 锁存器进行 Reset, CMP3 则用于固定占空比或占空比限制的 Set。

根据 AN4539 中对于不同比较单元优先级的描述, 为了实现上管常开 (即 100%占空比), 需要 Set 源的占空比比 Reset 源高, 所以这里 CMP1 和 CMP3 间不能互换; 因为 CMP1 和 CMP3 都可能出现值为 0 的情况, 所以不能与 CMP2 或 4 互换 (后者最小值为 96)。

- when CMP1 value is equal to the PER value, the duty cycle is 100% (CMP1 set event wins over PER reset event: refer to the hardware priority scheme detailed in the reference manual: when two simultaneous events occur, priority is the following: **CMP4 → CMP3 → CMP2 → MP1 → PER**)

CMP2 和 CMP4

由于 DAC 在每个周期 Reset 后的上升需要一定时间，在这个过程中比较器可能意外触发，所以使用对应子 Timer 的 CMP4 比较单元来屏蔽比较器的输入。

在上升后，使用 CMP2 来触发 DAC 开始输出三角波以进行斜坡补偿，CMP3 的值一直恒定为周期的 5%。

在峰值电流控制下，通过测试 DAC 输出的上升时间，选择 CMP4 的值为周期的 6%。因为模式切换会在 94% 占空比前进入其他模式，所以实际此值可以取更大。如下图中绿色波形曲线为 DAC 输出，黄色为 MCU 输出的半桥控制信号：



在固定占空比或上管常开模式下，需要完全屏蔽比较器的输入，所以对应地，需要将 CMP4 设置为占空比值或与周期相等的值。

ADC 采样时间

由于加入无线充电系统后，要采样的量更多，使得采样时间较为紧张，因为 ADC 引脚都有小电容或运放跟随，所以采样时间可以较短。计算如下

根据 STM32G474RB 的数据手册，在 MCU 主频为 170MHz 时，ADC 时钟最快为 42.5Mhz (4 分频)。

由于每个开关周期需要将所有量采样一次 (超级电容系统 5+无线充电系统 2)，使用 ADC1 和 2 一起采样，单个周期至少需要有进行 4 次采样的时间。

每个开关周期内 ADC 时钟的周期共有 $42.5\text{MHz}/250\text{kHz} = 170\text{cycle}$ ，每次采样时间为“采样时间+12.5cycle”，所以将 ADC 采样时间配置为 12.5Cycle，总时间占用开关周期的 58.8%

ADC Trigger 配置

由于电压和电流采样在进入 ADC 前都有进行 RC 低通滤波，且截止频率都为开关频率的 1/10 左右，对于开关频率的信号（例如滤波前信号电感电流波形，滤波进入 ADC 后为每侧平均电流）分别有-20dB（电压）和-40dB（电流）的衰减，所以对于采样时机要求不是很严格，仅需在周期开始后一段时间内触发采样，计算如下：

测试说明，电流采样比电压采样更容易受到开关噪声的干扰，故将 iA, iB 两个电流采样设定为 ADC1 和 2 的第一个采样 Rank。

为了最小化噪声，电流采样应在两次开关间，即周期的 1/4 或 3/4 处采样。考虑到电流采样二阶 RC 滤波的相位延迟为 180°，将采样时间设定在 1/4 周期处。

将采样时间的 12.5cycle 对齐 1/4 周期，需要提前 6.25cycle 开始触发采样。每个 cycle 对应的 HRTIM counter 数为 $21760/170 = 128$ ，故须在 1/4 周期上在提前 $128 \times 6.25 = 800$ 进行采样。

综上所述，ADC Trigger (Master Timer CMP2) 的值应为 $21760/4 - 800 = 4640$

双 ADC 配置

STM32G4 系列支持 ADC1 和 2 的双 ADC 同步采样来提升采样结果的同步（例如同步采样电流和电压），但是本系统中使用双 ADC 的主要原因是因为其可以将两个 ADC 采样的数据（各 12bit）存入一个 32 为寄存器中，由于计算 uint16 和 uint32 的开销相同，双 ADC 采样在进行结果的累加平均时可以将开销减半。（以每开关周期进行采样，四开关周期进行一次平均计算为例，12bit 的 ADC 原始数据进行四次加和最大为 14bit，不会对高 16 位数据造成干扰。）

由于双 ADC 要求两个 ADC 的 Rank 数和每个 Rank 的采样时间相同，故引入一个冗余量来满足此要求，实际每个 ADC 配置四个 Rank，采样时间均为 12.5cycle。

Number Of Conversion	4
External Trigger Conversion Source	High Resolution Timer Trigger 1 event
External Trigger Conversion Edge	Trigger detection on the rising edge
Rank	1
Channel	Channel 9
Sampling Time	12.5 Cycles
Offset Number	No offset
Monitored by	Analog Watchdog 2
Rank	2
Channel	Channel 8
Sampling Time	12.5 Cycles
Offset Number	No offset
Monitored by	Analog Watchdog 3
Rank	3
Channel	Channel 12
Sampling Time	12.5 Cycles
Offset Number	No offset
Monitored by	None
Rank	4
Channel	Channel 3
Sampling Time	12.5 Cycles
Offset Number	No offset

仅在 ADC1 中启用 DMA，数据为 32bit (word)，优先级为 Very High，模式为 Circular，将两个 ADC 的数据合并传输至一个变量中。

DMA Request	Channel	Direction	Priority
ADC1	DMA1 Channel 1	Peripheral To Memory	Very High

Add
Delete

DMA Request Settings

Mode Circular	Increment Address <input type="checkbox"/>	Peripheral	Memory
		<input type="checkbox"/>	<input checked="" type="checkbox"/>
Data Width	Word	Word	Word

使能 DMA Continuous Requests, 使得每次采样后都进行 DMA 传输, 并将 Overrun behavior 设置为 overwritten 保证采样数据一直在更新。

DMA Continuous Requests Enabled
 Overrun behaviour Overrun data overwritten

软件启动采样的代码为：

```
void initADC()
{
    //校准 ADC, ADC1 和 ADC2 用于同步采样, ADC4 用于监测温度和辅助电源
    HAL_ADCEx_Calibration_Start(&hadc1, ADC_SINGLE_ENDED);
    HAL_ADCEx_Calibration_Start(&hadc2, ADC_SINGLE_ENDED);
    HAL_ADCEx_Calibration_Start(&hadc4, ADC_SINGLE_ENDED);
    HAL_Delay(50);
    //双 ADC 同步采样
    HAL_ADCEx_MultiModeStart_DMA(&hadc1, adcData.rawData12, 4 * HRTIM_INT_SCALER);
    HAL_ADC_Start(&hadc2);
}
```

DAC 配置

配置 DAC 时将其调为高频模式，根据 RM0440 中对与 HRTIM 到 DAC 互联矩阵的描述，将两个 Trigger 分别配置为对应 Timer 中 Dual DAC Trigger 的 RST 和 STEP 信号源。Polarity 根据 AB 侧不同配置选择上升或下降。Reset 和 Step 两个数据由软件更新：

```
__attribute__((section(".code_in_ram"))) inline void setInductorCurrent()
{
    HAL_DACEx_SawtoothWaveGenerate(&hdac1, DAC_CHANNEL_2,
        DAC_SAWTOOTH_POLARITY_INCREMENT,
        PEAKI_TO_DACVAL((psData.iLTarget - 1.25f)), 180);
    HAL_DACEx_SawtoothWaveGenerate(&hdac1, DAC_CHANNEL_1,
        DAC_SAWTOOTH_POLARITY_INCREMENT,
        PEAKI_TO_DACVAL(-(psData.iLTarget + 1.25f)), 180);
}
```

▼ DAC Out1 Settings

Mode selected	Normal Mode
Output Buffer	Enable
DAC High Frequency	Mode Above 160MHz
DMA Double Data	Disable
Signed Format	Disable
Trigger	HRTimer Trigger RST Out2 event
Trigger2	HRTimer Trigger STEP Out2 event
Wave generation mode	Sawtooth wave generation
Polarity	Sawtooth polarity decrement
Reset Data	2300
Step Data	500
User Trimming	Factory trimming

COMP 和 OPAMP 配置

运放设定为跟跟随器模式，并使用 High speed 模式提升压摆率。

Mode

Power Mode
User Trimming

在运放的输出端对引脚进行复用，将相同引脚用于 ADC 采样。不使用内部通道直连的原因是外边连接可以通过测试点测量电压波形，方便调试。



比较器启用外部输出方便调试，由于其他功能都在 HRTIM 外设中实现，所以其余设置不做特殊配置。

Input [+]

Input [-]

ExternalOutput

比较器与比较器触发的半桥控制信号如下图所示（由于是测试版本有轻微震荡），其中紫色波形为比较器输出，黄色为 MCU 输出的半桥控制信号，比较器触发半桥 Reset：



电流感应放大器输出的电感电流波形（INA240 输出）：



对于运放，在启用前调用校准函数，最终模拟外设（除 ADC 外）初始化代码如下：

```
void initAnalog()
{
    HAL_OPAMP_SelfCalibrate(&hopamp1); // 采样电压跟随
    HAL_OPAMP_SelfCalibrate(&hopamp2); // 采样电压跟随
    HAL_OPAMP_SelfCalibrate(&hopamp3); // 采样电压跟随
    HAL_OPAMP_SelfCalibrate(&hopamp4); // 参考电压跟随
}
```

```
HAL_Delay(50);  
  
HAL_OPAMP_Start(&hopamp1);  
HAL_OPAMP_Start(&hopamp2);  
HAL_OPAMP_Start(&hopamp3);  
HAL_OPAMP_Start(&hopamp4);  
HAL_DAC_Start(&hdac1, DAC_CHANNEL_2);  
HAL_DAC_Start(&hdac1, DAC_CHANNEL_1);  
HAL_COMP_Start(&hcomp2);  
HAL_COMP_Start(&hcomp3);  
}
```



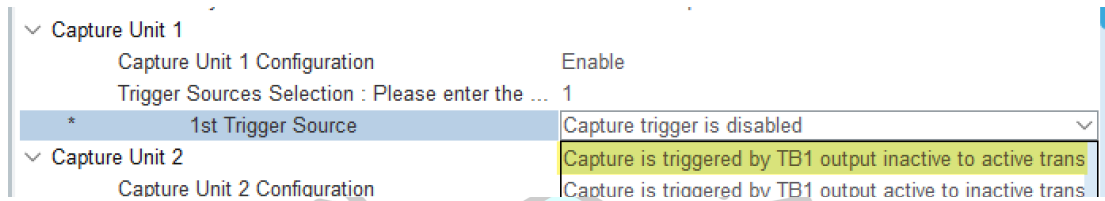
The Hong Kong University of
Science and Technology

3.3.8 升降压模式切换

占空比测量

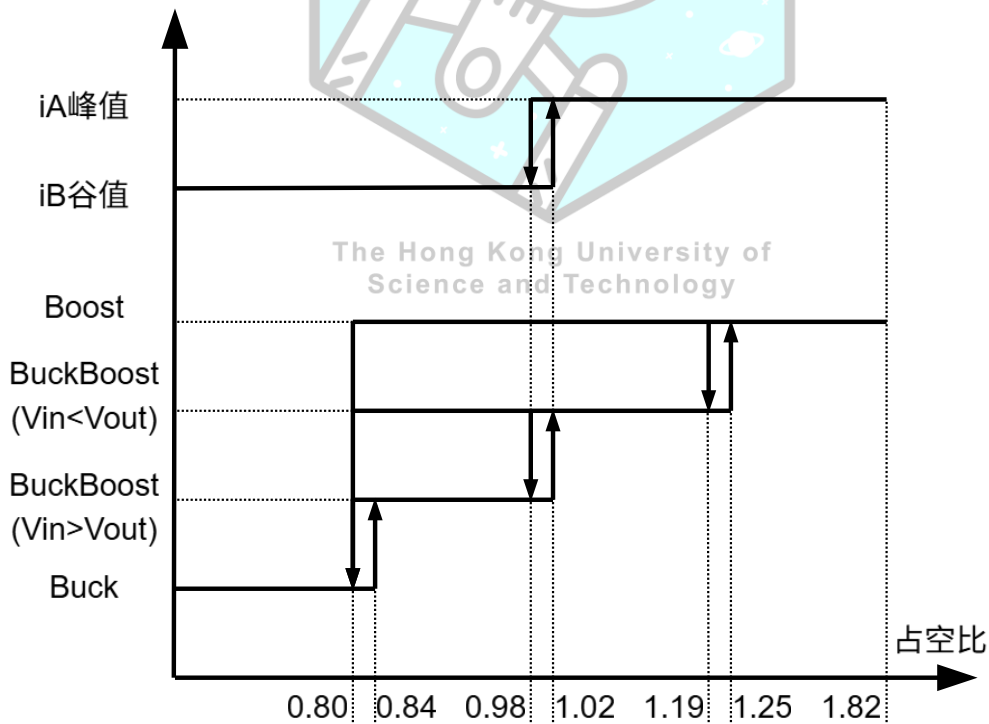
要进行状态切换首先要获知功率级的占空比。由于峰值电流模式控制的是电感电流的大小，并不是直接作用于占空比，所以需要观察。

一种测量方式是使用HRTIM的 Capture Unit 获取半桥控制信号输出是 Timer 的 Counter 值，通过 DMA 传入变量中，通过与周期值相除获取单边占空比，最终两侧占空比相除获取整体占空比。但是此方法测量结果噪音较大，如果直接用于控制会引起不同运行模式之间的震荡。



所以在本系统中使用了另一种更为直接的方式，通过观测两侧电压 v_A v_B 并将其相除，获取整体占空比。由于电压测量噪音极小，所以可以获得较为稳定的测量结果，并且仅需一次除法即可完成计算。此方案缺点是由于 MOS 管、采样电阻的压降，占空比会有一定偏差，但可以通过不同模式间留有余量的方式来保证模式间可以正常切换。

状态切换状态机



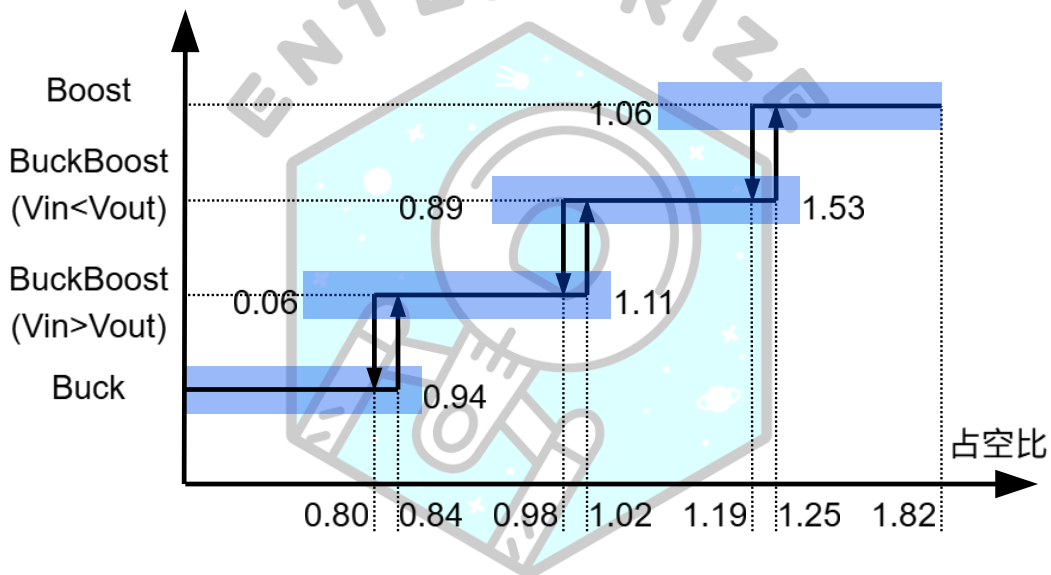
如图即为不同模式间的状态切换方式。在开关频率 1/4 的中断中，处理完 ADC 后通过 $D=vB/vA$ 的方式估算出占空比后，将 D 输入状态机进行不同模式的切换。

在不同模式间切换时，加入了施密特触发来防止在切换时震荡。例如 Buck 切换到 BuckBoost ($V_{in} > V_{out}$) 时，占空比阈值为 0.84，而相反阈值则为 0.80。

对于除 Buck 以外的模式，都可以通过一定阈值直接切换回 Buck 模式，这样做的原因是加快短路保护的响应速度。由于每 4 开关周期进行一次软件环路计算，例如限制功率级处于 Boost 模式，则至少需要 3 个软件周期或 12 个开关周期才能回到 Buck 模式，如果对每个模式都设置直接切换至 Buck 模式的阈值，响应仅需 1 个软件周期或 4 个开关周期。这种做法会提升不同模式下短路保护表现的一致性。

不同模式的运行范围

为了防止占空比偏差导致无法正常在模式间进行切换，在不同模式间留有一定冗余，如图：



忽略上文提到的任意模式可直接切换至 Buck 模式的通路；图中蓝色部分为每个模式下允许的最大和最小占空比，可以充分保证切换不会出现异常。例如 Buck 模式下占空比高于 0.84 就会切换到 BuckBoost，但实际 Buck 模式最高允许占空比为 0.94。

切换模式半桥输出

根据时序图和模式切换的需求，可见半桥输出效果需要达成三种模式：

- 周期开始 Reset，对侧峰值电流控制 Set，并且有最小占空比限制 (Set)
- 固定占空比输出
- 常开

在占空比状态机后，根据现有模式进行相应的寄存器设定，具体代码如下：

```
__attribute__((section(".code_in_ram"))) void modeStateMachine()
{
```

```

//根据电压计算占空比
psData.dutyByVoltage = adcData.vB / adcData.vA;

//根据占空比进行状态切换
switch(psData.dcdcMode)
{
case BUCK:
    if(psData.dutyByVoltage > 0.84f)
        psData.dcdcMode = BUCKBOOST;
    break;
case BUCKBOOST:
    if(psData.dutyByVoltage < 0.80f)
        psData.dcdcMode = BUCK;
    else if(psData.dutyByVoltage > 1.02f)
        psData.dcdcMode = BOOSTBUCK;
    break;
case BOOSTBUCK:
    if(psData.dutyByVoltage < 0.82f)
        psData.dcdcMode = BUCK;
    else if(psData.dutyByVoltage < 0.98f)
        psData.dcdcMode = BUCKBOOST;
    else if(psData.dutyByVoltage > 1.25f)
        psData.dcdcMode = BOOST;
    break;
case BOOST:
    if(psData.dutyByVoltage < 0.82f)
        psData.dcdcMode = BUCK;
    else if(psData.dutyByVoltage < 1.19f)
        psData.dcdcMode = BOOSTBUCK;
    break;
default:
    break;
}

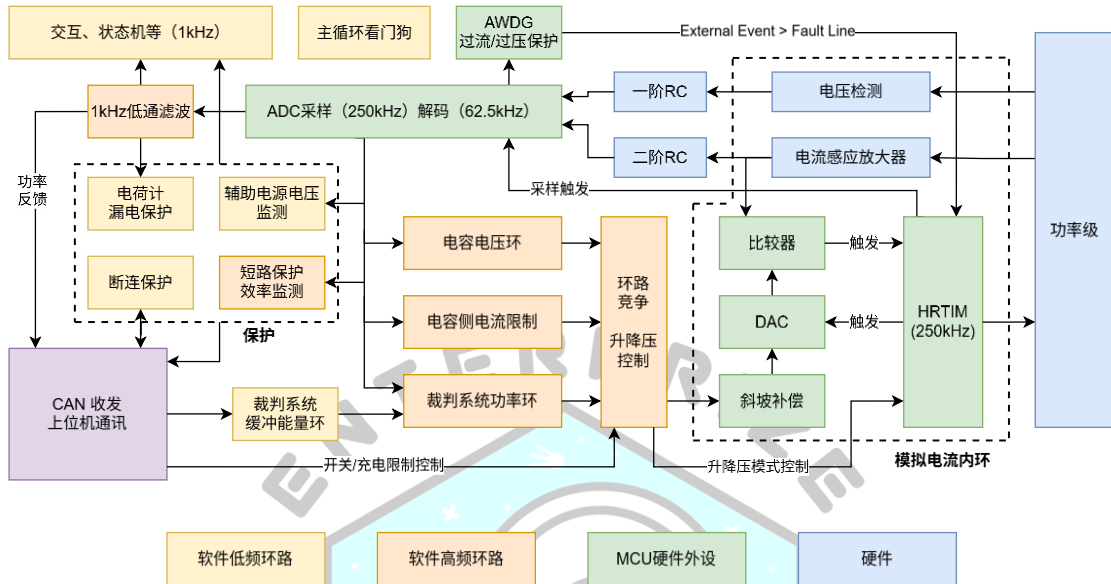
//根据状态操作 HRTIM 寄存器，并分别计算 A 和 B 的占空比
switch(psData.dcdcMode)
{
case BUCK:
    //A 侧限制 94%占空比
    __HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_A,
        HRTIM_COMPAREUNIT_4, HRTIM_PERIOD * 0.06f);
    //A 侧限制 0.5%占空比
    __HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_A,
        HRTIM_COMPAREUNIT_3, HRTIM_PERIOD * 0.995f);
    //B 侧常开
    __HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_B,
        HRTIM_COMPAREUNIT_3, 0U);
    //更新状态和 AB 占空比
    psData.pcmMode = IB_VALLEY;
    break;
case BUCKBOOST:
    //A 侧限制 94%占空比

```

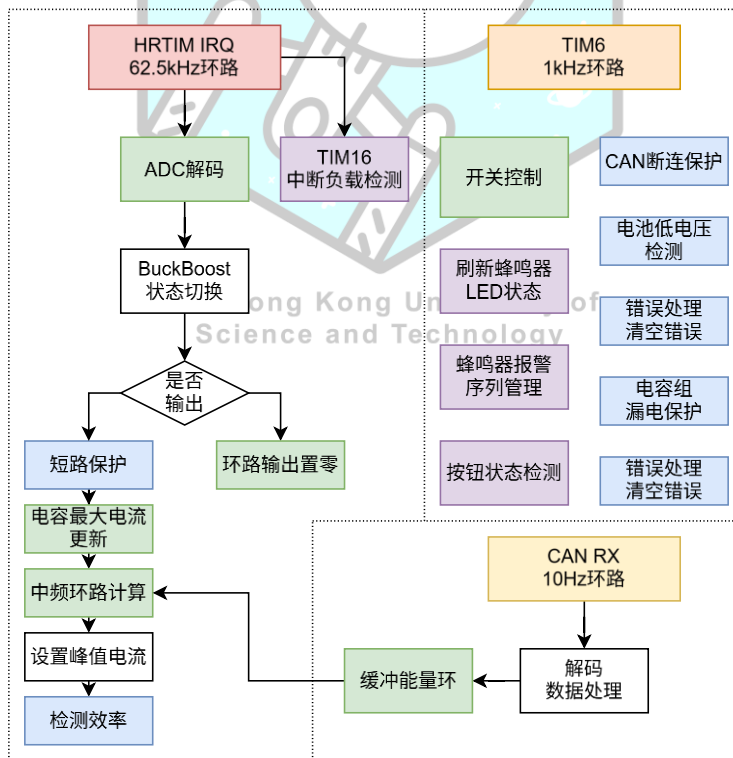
```
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_A,  
                        HRTIM_COMPAREUNIT_4, HRTIM_PERIOD * 0.06f);  
//A 侧限制 5%占空比  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_A,  
                        HRTIM_COMPAREUNIT_3, HRTIM_PERIOD * 0.95f);  
//B 侧固定 84%占空比  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_B,  
                        HRTIM_COMPAREUNIT_3, HRTIM_PERIOD * 0.16f);  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_B,  
                        HRTIM_COMPAREUNIT_4, HRTIM_PERIOD * 0.24f);  
//更新状态和 AB 占空比  
psData.pcmMode = IB_VALLEY;  
break;  
case BOOSTBUCK:  
//A 侧固定 84%占空比  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_A,  
                        HRTIM_COMPAREUNIT_3, HRTIM_PERIOD * 0.16f);  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_A,  
                        HRTIM_COMPAREUNIT_4, HRTIM_PERIOD * 0.24f);  
//B 侧限制 94%占空比  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_B,  
                        HRTIM_COMPAREUNIT_4, HRTIM_PERIOD * 0.06f);  
//B 侧限制 55%占空比  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_B,  
                        HRTIM_COMPAREUNIT_3, HRTIM_PERIOD * 0.45f);  
//更新状态和 AB 占空比  
psData.pcmMode = IA_PEAK;  
break;  
case BOOST:  
//A 侧常开  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_A,  
                        HRTIM_COMPAREUNIT_3, 0U);  
//B 侧限制 94%占空比  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_B,  
                        HRTIM_COMPAREUNIT_4, HRTIM_PERIOD * 0.06f);  
//B 侧限制 55%占空比  
__HAL_HRTIM_SETCOMPARE(&hhrtim1, HRTIM_TIMERINDEX_TIMER_B,  
                        HRTIM_COMPAREUNIT_3, HRTIM_PERIOD * 0.45f);  
//更新状态和 AB 占空比  
psData.pcmMode = IA_PEAK;  
break;  
default:  
break;  
}  
}
```

3.3.9 数字外环设计与实现

数字外环以开关频率 1/4 运行 (62.5kHz)，在 HRTIM Master Timer 的 IRQ_Handler 中运行。本节将介绍数字环路的实现方式。



流程框图



如图，超级电容系统共有三个由不同中断源产生的环路，主循环仅用于处理看门狗，本部分将重点介绍图中环路控制相关内容（绿色内容）。

ADC 解码

上文提到双 ADC 采样数据 (32 位) 由 DMA 传输至 uint32_t 数组中, 每周期 4 个采样×4 个周期, 数组大小为 16。

采样顺序为: ADC1 iA iR vA vWPT(无线充电输入电压)

ADC2 iB iB(占位) vB iWPT(无线充电电流)

首先对 4 个周期的数据进行累加, 得到 sumData[4], 再分别根据校准的斜率和截距解码, 例如对于 iA 和 iB 的解码, 数据分别存在一个 uint32 的高 16 位和低 16 位:

```
adcData.iA = (1-ADC_ISENSE_ALPHA) * adcData.iA +
    ADC_ISENSE_ALPHA * ((uint16_t)adcData.sumData[0] * ADC_IA_K + ADC_IA_B);
adcData.iB = (1-ADC_ISENSE_ALPHA) * adcData.iB +
    ADC_ISENSE_ALPHA * (*(uint16_t*)((uint8_t*)&adcData.sumData[0] + 2) *
    ADC_IB_K + ADC_IB_B);
```

在解码同时, 对电流进行一阶软件滤波, 电压量实测噪音很小故不进行额外滤波。

在直接测量量解码后进行间接量的计算, 例如底盘功率、裁判系统功率、电容组功率等

```
adcData.iChassis = adcData.iR - adcData.iA;
adcData.pReferee = adcData.vA * adcData.iR;
adcData.pChassis = adcData.vA * adcData.iChassis;
```

最终, 对特定量进行截止频率约为 1kHz 的软件低通滤波以供给低频环路进行相关运算, 防止 1kHz 环路直接使用 62.5kHz 环路的数据造成额外的噪声。以底盘功率为例, 电容控制器需要以 1kHz 的频率向主控板反馈底盘功率, 所以需要 pChassis 进行低通滤波后使用

```
adcData.pChassislf = MF_TO_LF_ALPHA * adcData.pChassis +
    (1-MF_TO_LF_ALPHA) * adcData.pChassislf;
```

电容内阻补偿

由于电容组内阻约为 100-300mΩ 级别, 在大电流放电时会产生明显的压降, 意味着测量电压小于电容组内的实际电压, 会造成放电时电流限制过早等问题。

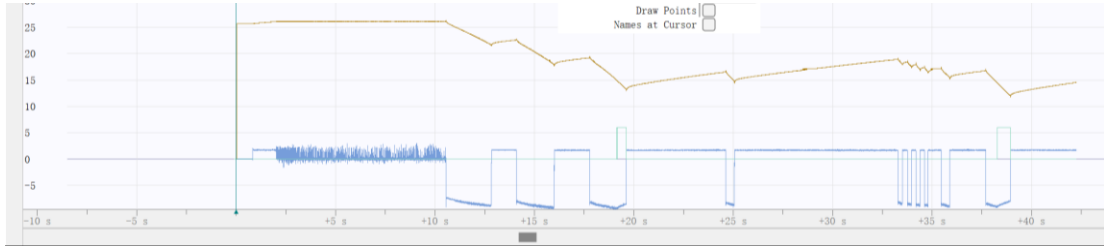
具体做法为根据电容组侧电流 iB 在测量的 vB 上增加偏置, 这种简单的补偿方法实测有非常好的效果, 其实现仅需一行代码, 其后 vCap 用于电容组相关计算, 而 vB 继续用于功率级本身相关的计算:

```
adcData.vCap = adcData.vB - adcData.iB * CAPARR_DCR;
```

具体的调试方法为:

- 将 CAPARR_DCR 设为 0, 在调试软件中绘制 vCap 的曲线
- 对电容组进行反复正反充放电切换 (例如+5A 到-10A), 同时调整 CAPARR_DCR 的值, 直到近似消除切换处的电压突变。

如图, 棕色曲线为补偿后的电容电压, 蓝色曲线为电流, 可见基本消除了电容电压的突变。

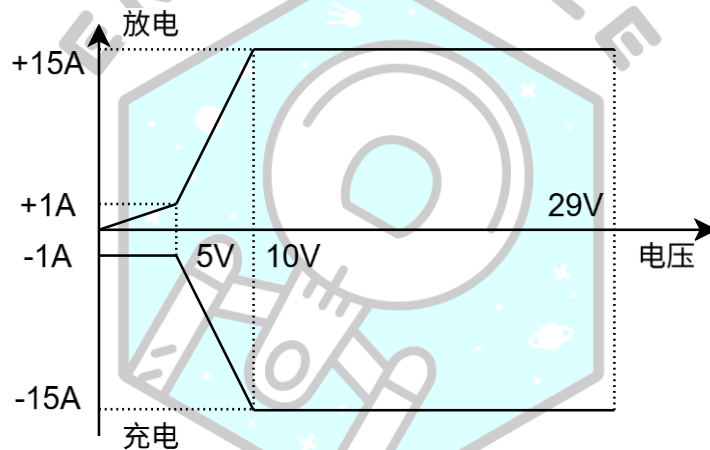


更新电容最大电流

电容组电压处于正常范围时，电容组电流限制为裁判系统 CM01 的 15A。

由于电容组低电压时放电功率极大受限，所以需要在低压时严格限制电容组电流。

以 29V 电容组为例，电容组电压 10V 时电容组能量为 11.8%，以理论最大放电功率为 160W 放电仅能维持 1 秒左右；5V 时电容组能量为 3%，几乎耗尽。所以将电容组低压限制设定为 10V，放电截止电压设定为 5V。对充放电电流进行限制：



如图可见在充放电时对电流的限制并不对称，如低压时允许较低功率充电，但几乎不允许放电。

The Hong Kong University of Science and Technology

中频环路 1 – 裁判系统功率环

裁判系统功率环的核心为增量式 PID，其输入为“裁判系统目标功率/底盘电流”与“当前裁判系统端电流”，输出为增量式的电感电流。

其中增量式 PID 的写法延续了队伍去年开源中的写法，即将 P 项分为 pOnMeasurment 和 pOnTarget 两项，分别进行调整以实现目标突变情况下的快响应、低过冲。实际调整中后者约为前者的 1/2。

中频环路 2 – 电容组电压环

电容电压环主要由一个增量式的 I 环实现 (PD 等为 0)，这样做的原因如下：

- 经过测试单 I 控制完全可以较好地限制电压

- 62.5kHz 环路需要尽可能减小运算量
- 在只有 I 的情况下，通过调整 k_I 值，使得输出端悬空时输出电压刚好会有轻微震荡，而当电容组接入时则不会震荡，这样可以通过轻微震荡的特征来判断电容组断连以进行保护，具体保护将在下文中说明。

中频环路 3 – 电容组电流限制环

由于峰值电流模式不能精确控制平均电流，故不能使用数控方式中直接限制电流环的目标值来实现限流。实测中使用电流纹波估算限流的方式并不能很好限制输出电流，故额外引入了两个环路来限制电流输出。

电容组电流限制环主要由两个增量式的 I 环实现（PD 等为 0），将正反电流分别做限制，选择 I 环的具体原因与上文的前两条原因相同：

- 经过测试单 I 控制完全可以较好地限制电流
- 62.5kHz 环路需要尽可能减小运算量

推测单 I 环即可较好限制的原因推测如下：虽然 I 环会引入过冲，但此过冲其实是电感电流峰值的过冲，由于实际观测量为输出电流（经过了约 $500\mu\text{F}$ 的电容），过冲部分的电流带来的额外电荷可能被滤波电容吸收，故在观测输出电流时过充较小。

中频环路 – 竞争与输出

在分别计算完上文提到的三个环路后，将结果进行相互比较，最终输出一个目标电流的插值，流程如下：

- 首先将目标电流差值设置为裁判系统功率环的结果
- 将目标值与电容组电压环输出进行比较，如果后者输出较小则直接输出结果，跳过后续环路
- 将目标值与电容组电流环结果进行比较，以输出电流限制环为下限，输入电流限制环为上限，对目标值进行限制。
- 将目标值（电流目标增量）累加到限制的峰值电流目标上
- 对峰值电流目标的绝对值进行限制，限制为电感的饱和电流以下（饱和电流 30A 左右，实际限制为 25A），或在缓启动时进行电流限制。

```
mfLoop.deltaIL = mfLoop.iRPID.getOutput();
if(mfLoop.dIL_VCap_Max < mfLoop.deltaIL)
{
    mfLoop.deltaIL = mfLoop.dIL_VCap_Max;
    ctrlData.limitFactor = CAPARR_VOLTAGE_MAX;
}
else if(mfLoop.dIL_IB_Positive < mfLoop.deltaIL)
{
    mfLoop.deltaIL = mfLoop.dIL_IB_Positive;
```

```

ctrlData.limitFactor = IB_POSITIVE;
}
else if(mfLoop.dIL_IB_Negative > mfLoop.deltaIL)
{
mfLoop.deltaIL = mfLoop.dIL_IB_Negative;
ctrlData.limitFactor = IB_NEGATIVE;
}
psData.iLTarget += mfLoop.deltaIL;
psData.iLTarget = M_CLAMP(psData.iLTarget, -psData.iLLimit, psData.iLLimit);

```

裁判系统缓冲能量环

由于电流采样、电压采样都有一定误差，且功率限制以电源管理模块测得为准，所以需要电容控制器的功率误差作时时校准以从裁判系统中获取最大能量。

由于从上赛季开始裁判系统串口移除了功率反馈，所以这里使用与队伍上赛季开源的电容控制器相同的方法，通过裁判系统缓冲能量对功率进行时时校准。

缓冲能量反馈频率为 10Hz，经由主控板通过 CAN，与其他信息一同反馈给电容控制器。由于数据源刷新率为 10Hz，缓冲能量环也在 10Hz 环路中运行。

缓冲能量环使用位置式 PID 实现，输入为当前缓冲能量，目标值设定为 57J，目标值选择主要考虑了距离缓冲能量上限 60J，可以更快达成闭环，同时留有较多余量。将输出直接以偏置的方式增加到原有的目标功率上输入裁判系统功率环。

开关控制与缓启动

裁判系统底盘断电时，电容不能向底盘继续供电，故需要对开关进行控制。开关控制状态机在 1kHz 环路中运行，其以底盘电压为控制量进行施密特触发。

当功率级断电时，控制器依然由裁判系统功率环控制，由于此时裁判系统不会提供能量，地盘电压会急剧降低，使用此特性来实现开关控制，逻辑如下：

- 当功率级断电时，如果底盘电压大于 20V，且没有进入错误保护，功率级则会启动
- 当功率级启动时，如果底盘电压小于 18V 时，功率级会关闭

为了防止上电瞬间的电流尖峰过大，当启动时，会限制裁判系统环的目标功率。

DAC 输出

当通过中频环路计算出得出电感电流目标后，需要通过 DAC 输出来将其转换为峰值电流内环的输入。

峰值电流内环控制的是功率级的峰值电流，但是当在 BuckBoost ($V_{in} < V_{out}$) 与 BuckBoost ($V_{in} > V_{out}$) 模式切换时，峰值电流控制源会在 i_A 峰值与 i_B 谷值间切换，虽然配合外环闭环速度很快，但还是不可避免地在切换模式引入电流波动。

由于不切换峰值/谷值控制时峰值电流的偏差并不重要（实测输出电压很低时为 1.4A 左右，

完全可由外环补偿)，所以仅需在 BuckBoost (Vin<Vout) 与 BuckBoost (Vin>Vout) 模式切换时对峰峰值电流做补偿。计算如下：

当模式切换时，输入输出电压几乎相等，都约等于 24V 左右，此时一侧为固定占空比 84%，另一侧占空比也约为 84%。

$$\text{此时电感的电流纹波约为: } \frac{24\text{V}}{10\mu\text{H}} \times \left(16\% \times \frac{1}{250\text{kHz}}\right) = 1.54\text{A}$$

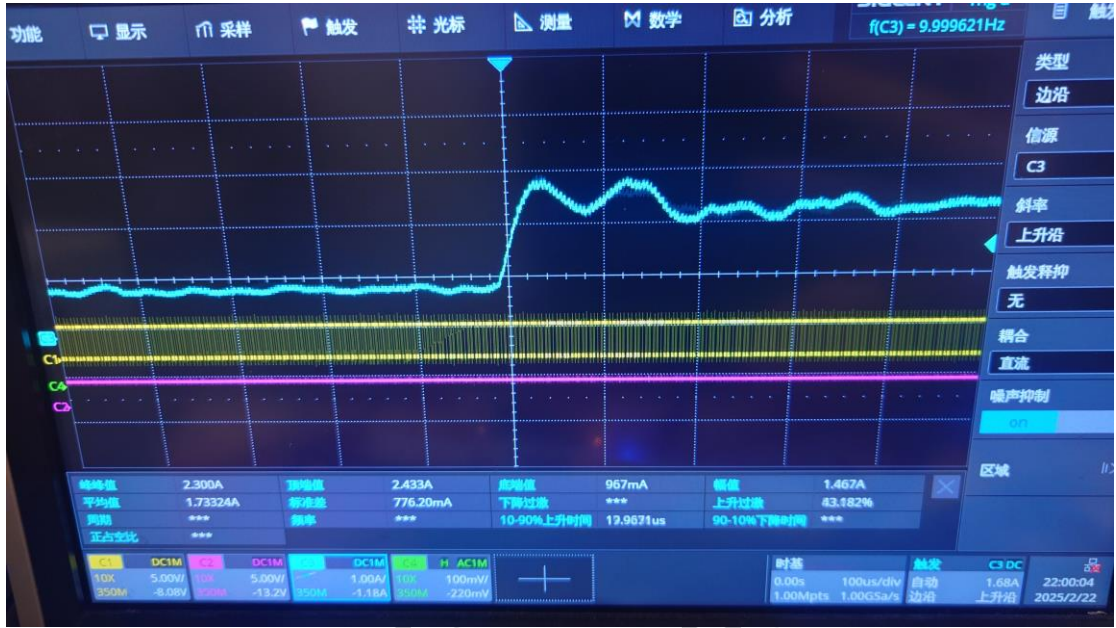
在得到电流目标后，将其分别进行正反纹波 1/2 的偏置，并通过换算得到最终的 DAC 输出值。由于 DAC 误差、斜坡补偿等原因，目标电流值仍有不可消除的偏差，但偏差值较小，完全可以由外环进行补偿（峰值电流环绝对对准度相对不重要）。

```
#define PEAKI_TO_DACVAL(x) ((x) * (HW_RSENSE * HW_IAMP_GAIN * 4096.0f / ADC_VREF) + 2048U)
HAL_DACEx_SawtoothWaveGenerate(&hdac1, DAC_CHANNEL_2,
    DAC_SAWTOOTH_POLARITY_INCREMENT,
    PEAKI_TO_DACVAL((psData.iLTarget - HALF_IPP)), 180);
HAL_DACEx_SawtoothWaveGenerate(&hdac1, DAC_CHANNEL_1,
    DAC_SAWTOOTH_POLARITY_INCREMENT,
    PEAKI_TO_DACVAL(-(psData.iLTarget + HALF_IPP)), 180);
```

响应速度

在实现峰值电流内环+数字外环后，电流响应速度极快，在 PID 参数留有一定余量的情况下，外部电流的响应（经过 500uF 左右的滤波电容）可达 20μs 级别。测试中使用电流探头测量外部电路，每 50ms 改变一次目标值，阶跃 1.5A 左右，结果如下图：

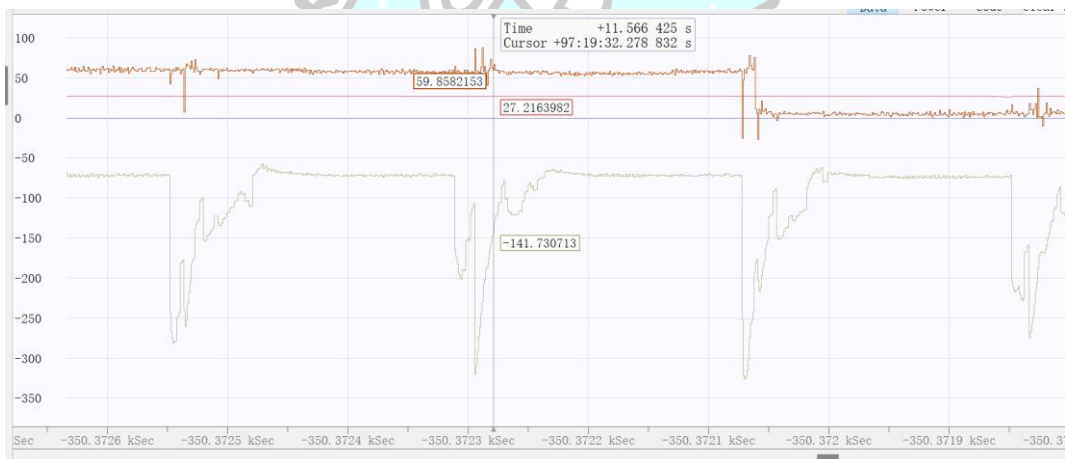




动能回收

在进行动能回收时，只需要控制裁判系统端电路不为负，即可将能量导入电容组中。因此使用前文实现的响应极快的环路控制即可完成目标。

下图为测试时的波形曲线，上面的曲线为裁判系统功率，下面的曲线为底盘功率（方便观察在绘图时加了-120W 的偏置）。测试使用四个 3508 电机的底盘，悬空加速并突然刹车，瞬时反冲功率超过 200W 左右，如图可见裁判系统端功率很好地完成了闭环，并几乎将全部的刹车的反冲能量导入了电容中（表现为裁判系统功率始终为正）。



3.3.10 充电限制和纯电容模式

引入充电限制的原因

结合文章开始提到的单局能量限制的应对方法，需要对电容控制策略做出改变，通过引入电容充电限制，例如在电容组充电至 90%后不再从裁判系统获取能量主动充电来实现如下效果：

- 剩余 10%能量留给动能回收，防止电容充满时动能回收能量倒灌入裁判系统造成浪费
- 减少首先充入电容，再从电容流入底盘的能量，这样可以减小功率变换带来的损耗
- 一定程度上可以延长电容组寿命

使用场景

充电限制模式在正常运行时启动，但是当机器人即将投入战斗中或即将飞坡（需要较高爆发能量），充电限制模式将关闭以最大化存储的能量。

具体实现方式

充电限制功能由主控板控制，主控板将通过开关 bit 和一个 uint8 格式的阈值来分别控制此功能的开关和阈值。当受到开启信号后，将容量百分比阈值通过电容能量 $E = \frac{1}{2} \times C \times U^2$ 的关系映射到电容电压，作为控制的输入量。

在中频环路中通过阈值电压的施密特触发来改变裁判系统功率环的目标值，逻辑如下：

- 当电压上升到阈值电压+0.1V 时，将裁判系统目标功率设置为 6W（大约为电容控制器+底盘的静态功耗，此值设置太小会导致有能量倒灌入裁判系统造成浪费）
- 当电压下降到阈值电压-0.1V 时，将恢复裁判系统正常的功率目标值，即功率限制+裁判系统能量环的校准偏置值

纯电容模式

由于无线充电系统的引入，在充电前电容组内需要有较多的空余能量以供接收电能。具体做法是通过修改裁判系统的目标功率为很低的值，使得底盘优先从电容中获取能量，直到消耗完大部分能量后转入裁判系统供电。此模式由操作手主动触发（例如返回补给区时将电容放空以供无线充电）。

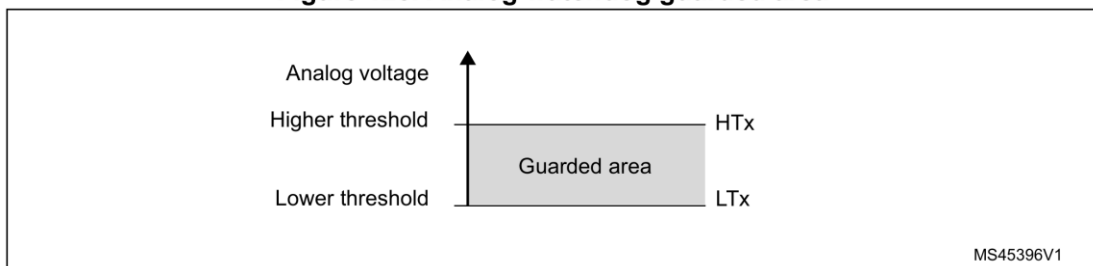
3.4 保护

由于超级电容系统涉及大功率的电压/电流变化，且电容组在满电时存储 2000J 的能量，如果出现故障风险极大。所以在超级电容系统中需要完善的保护系统。

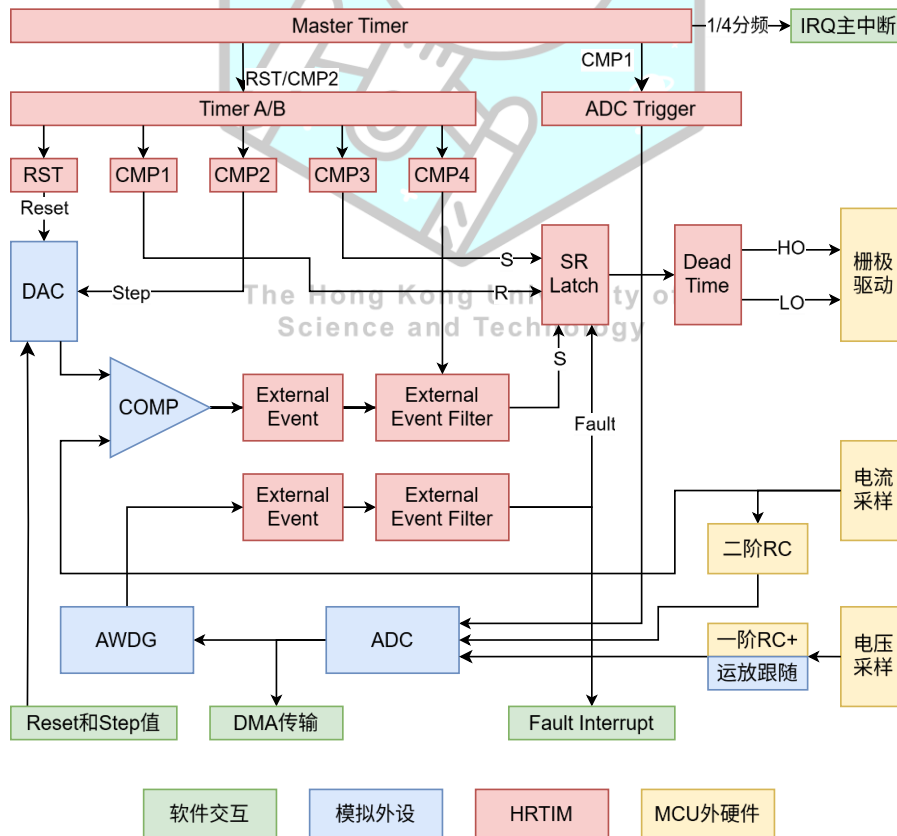
3.4.1 过压与过流保护

为了减小软件计算开销，并且提升保护响应速度，过压与过流保护使用模拟看门狗 AWDG 实现。这种方式在硬件上对 ADC 采样结果进行比较，如果超出阈值可触发特定事件的信号。

Figure 125. Analog watchdog guarded area



如上文所示的框图中所示，AWDG 信号接入 HRTIM 的 External Event，在通过过滤（事件连续发生超过几次）后直接关断功率级，同时触发软件中断以进行相应处理。



触发链

```
iA > ADC1_Channel19 > ADC1_AWDG2 > HRTIM1_ExternalEvent2 > HRTIM1_FaultLine2
iR > ADC1_Channel18 > ADC1_AWDG3 > HRTIM1_ExternalEvent3 > HRTIM1_FaultLine3
vA > ADC1_Channel12 > ADC1_AWDG1 > HRTIM1_ExternalEvent1 > HRTIM1_FaultLine1
iB > ADC2_Channel15 > ADC2_AWDG2 > HRTIM1_ExternalEvent5 > HRTIM1_FaultLine5
vB > ADC2_Channel13 > ADC2_AWDG1 > HRTIM1_ExternalEvent4 > HRTIM1_FaultLine4
```

外设配置

根据 RM0440 描述，AWDG1 和 AWDG2/3 不同，AWDG1 精度为 12 位，AWDG2/3 只能监测高 8 位，所以对于电压 vA 和 vB，使用监测精度较高的 AWDG1，对于精度要求相对不高的电流，使用 AWDG2/3

在 ADC 配置页面使能 AWDG 并绑定相关通道，滤波、中断等处理在 HRTIM 中实现。

<ul style="list-style-type: none"> Analogue Watchdog 1 <ul style="list-style-type: none"> Enable Analog WatchDog1 Mode <input checked="" type="checkbox"/> Watchdog Mode: Single regular channel Analog WatchDog Channel: Channel 12 High Threshold: 4095 Low Threshold: 0 Interrupt Mode: Disabled Filtering: None Analogue Watchdog 2 <ul style="list-style-type: none"> Enable Analog WatchDog2 Mode <input checked="" type="checkbox"/> Watchdog Mode: Single channel High Threshold: 4095 Low Threshold: 0 Interrupt Mode: Disabled

将 AWDG 输出链接至 HRTIM 的 External Event

<ul style="list-style-type: none"> External Event 1 <ul style="list-style-type: none"> Event Configuration: Enable External Event 1 Source: External Event Source 4 - ADC1_AWD1 Event Polarity: External event is active high Event Sensitivity: External event is active on level Event Fast Mode: External Event is re-synchronized by the HRTIM logic before acting on outp...

将 External Event 输出链接至 Fault Line

<ul style="list-style-type: none"> Fault Line 1 <ul style="list-style-type: none"> Line Configuration: Configure Fault Line Source: Fault input is External Event 1 Polarity: External event is active high Filter: fSAMPLING= fHRTIM, N=8 Lock: Fault settings bits are read/write Line Ctl: Enable Fault Line Counter on Faults (Threshold): 0 Reset Mode: Fault counter is reset on each reset / roll-over event Blanking Source Type: Reset-Aligned Window - Blanking window starts on Tim...

软件配置

根据 RM0440 描述，在启用 ADC 前配置 ADC1 和 2 的 TR1/2/3 寄存器来调整阈值，具体代码如下所示，其中电压值上限按理论值计算，下限为 0，电流设置较为宽松设置为 27A

左右，计算方式为： $I_{limit} = \frac{0xE0-2048U}{4096U} \times V_{Ref} \times \frac{1}{20V/V \times 2m\Omega}$

```
void configAWDG()
{
    ADC1->TR3 = 0x00E00020;
    ADC1->TR2 = 0x00E00020;
    ADC2->TR2 = 0x00E00020;
    ADC1->TR1 = (uint32_t)(OVP_A/2.9f/16.0f*4095.0f) << 16;
    ADC2->TR1 = (uint32_t)(OVP_B/2.9f/16.0f*4095.0f) << 16;
    HRTIM1->sCommonRegs.IER |= 0b11111;
}
```

在配置完阈值后，通过 HRTIM 的 IER 寄存器启用全部五个中断源。

	6	5	4	3	2	1	0
	FLT6 IE	SYSFL TIE	FLT5IE	FLT4IE	FLT3IE	FLT2IE	FLT1IE
	rw	rw	rw	rw	rw	rw	rw

在软件的 IRQ_Handler 中进行软件上的错误处理，通过读取 HRTIM 的 ISR 获知错误源，并进行错误状态更新，以 vA 过压保护为例：（此处关断输出的函数只是为了更新状态，实际输出由硬件触发关断）

```
if(HRTIM1->sCommonRegs.ISR & HRTIM_FLAG_FLT1) //vA 过压保护触发
{
    errorData.errorCode |= ERROR_OVP_A;
    errorData.errorLevel = ERROR_RECOVER_AUTO;
    HRTIM::disableOutputAB();
}
```

Science and Technology

	6	5	4	3	2	1	0
	FLT6	SYSFL T	FLT5	FLT4	FLT3	FLT2	FLT1
	r	r	r	r	r	r	r

3.4.2 短路与功率级保护

短路保护

短路保护判定条件沿用了队伍上赛季开源的电容控制器的做法，通过电压小于一定阈值（如 5V）、电流大于一定阈值（如 5A）判断。由于涉及两个条件，且其中一个条件可能被经常触发，故不适用 AWDG 和错误中断实现。

以 A 端短路保护代码为例，当两个阈值条件都满足时，开始进行错误计数器累加，同时更新状态，当错误计数器累加带一定值后触发功率级关断，更新相关错误状态，并记录触发时的电流和电压以供调试。

```
if(adcData.vA <= SCP_VOLTAGE && -adcData.iA >= SCP_CURRENT)
{
    //裁判系统端或底盘端短路
    errorData.errorLevel = WARNING;
    errorData.shortCircuitCnt +=600;
    if(errorData.shortCircuitCnt > 1100){
        HRTIM::disableOutputAB();
        errorData.errorCode |= ERROR_SCP_A;
        errorData.errorLevel = ERROR_RECOVER_MANUAL;
        errorData.errorVoltage = adcData.vA;
        errorData.errorCurrent = -adcData.iA;
    }
}
```

例如上电时某瞬间 A 端过流条件被触发 1 次，此时计数器值为 600，后续无触发，那么计数器会在低频环路中逐渐减小，当减为 0 时清除 WARNING 状态；但是如果短时间内触发次数达到阈值，则会触发关断。

得益于峰值电流本身限制电流峰值的特性和较快的闭环速度，本设计的短路保护阈值较比数控方案可以设置地很低。测试中 B 端电压在 29V 左右（电容组满电电压），将输出短路，短路后保护瞬间触发，短路过程中无明显火花，且多次触发也不会损坏功率级。

效率监测

当 A 侧电流绝对值大于 0.5A 时，根据电流流向计算功率级转换效率，如果效率过低则进行关断保护

辅助供电监测

由于本设计中电源网络较多，当 MCU 正常工作时只能说明 3.3V 网络供电正常，此时如果 3.3VA、10V 或 5V 网络故障会导致严重后果，本设计使用四个电阻（分别指向三个电源网络和 GND）组成分压网络，当所有电源电压正常时，ADC 读取值应为 1.41V 左右。排除仅使用烧录器供电的情况，当此电压不正常时可以通过禁止启用功率级输出来保护控制器。

3.4.3 通讯断连保护

当电容组至主控板的 CAN 信号断连时，需要进行错误处理来防止超功率等情况发生（例如功率限制更新不及时等因素引起）。

软件设计上，在每次 CAN RX 中断中更新一个时间戳（方便调试），并更新连接状态，在 1kHz 的环路中将时间戳与当前时刻作比较，如果超过 500ms 即视为断连。

断连后将所有接收到的数据清空，并关闭所有特殊模式（纯电容模式和充电限制），将裁判系统功率目标值设定为 37W（考虑到今年虚弱模式的加入，此阈值后续可能还需要调整）。

3.4.4 电容组漏电/断连保护

由于 RM 比赛中电容组一般为自行设计并手工焊接，有概率出现接触不良或漏电（如均衡保护失效）等情况，此时需要电容控制器对其进行保护。原理主要根据公式： $\Delta Q = C\Delta U$

保护算法同时监测两个量，分别为电荷量与电压。具体实现如下：

- 在 1kHz 环路中积分电流获得电荷量（实测 62.5kHz 环路中积分噪声较大），在 62.5kHz 环路中监测电压的最大值和最小值。
- 如果电荷量达到一定阈值，与电压差值进行一次相除，为了防止电压不变而 0 为分母的情况，实际上使用“电压差/电荷差”与容量阈值的倒数做对比。
- 如果电压差达到一定阈值，与电荷量进行一次相除，为了电荷量电压不变而 0 为分母的情况，实际上使用“电荷差/电压差”与容量阈值做对比。
- 如果长时间二者都没有达到阈值，则清空两个量，重新开始积分。
- 根据触发情况累加错误计数器，当达到一定阈值认为电容组故障。

这种算法实测可以很好地处理电容组断连、漏电等情况。当等效容量过低则认为电容组断连，容量过高认为电容组漏电。

断连测试

前文提到电容组不连接时 vB 的电压环被有意调整为轻微震荡，此时电压差会很快达到阈值，由于未连接电容，电荷量积分几乎为 0，可以触发保护。

漏电测试

将电容组输出并联到电子负载，以恒流模式放电，实测当放电电流 1A 左右时开始触发保护。

实际使用中错误保护触发不会直接关断输出，而是会通过蜂鸣器提醒用户，如果频繁触发则说明需要检查接线或更换电容组。

3.4.5 错误分级处理与其他保护

错误分级

为了更好地处理错误，本设计将错误分级，并且为每个错误都分配对应的 errorCode：

```
enum ErrorLevel
{
    NO_ERROR = 0,           // 无错误
    ERROR_RECOVER_AUTO = 1, // 错误，可通过自动恢复
    ERROR_RECOVER_MANUAL = 2, // 错误，可通过发信息恢复
    ERROR_UNRECOVERABLE = 3, // 错误，不可恢复
    WARNING              // 警告
};
```

当 WARNING 等级错误触发时（一般为短路保护的条件下达成但未达到触发阈值），1kHz 环路中会进行错误计数器的减小。

当 ERROR_RECOVER_AUTO（过流或过压）触发后，会在 5s 后尝试重启进行恢复。

当 ERROR_RECOVER_MANUAL（短路保护）触发后，将不会自动恢复，可以通过 CAN 向电容发起错误清除命令或通过按钮手动清除。

当 ERROR_UNRECOVERABLE（功率级错误）触发后，不会自动恢复，需要拆掉控制器进行错误排查。

UID 检测

由于不同控制器的校准参数不同，在初始化中会读取 STM32 中的 96 为 UID 并进行对比，如果与校准数据中不相符则停止初始化并报警。

过温检测

硬件上使用 NTC 电阻进行过温检测，实际使用中发现电容控制器发热并不严重，故此故障保护暂时没有启用，仅作硬件预留。

3.5 通讯与交互

3.5.1 CAN 通讯

超级电容控制器与主控板通讯主要通过 CAN 进行,数据格式大致与队伍上赛季的开源相同,并作出一定调整。例如将错误码改为状态码,反馈更多信息,并加入裁判系统功率反馈。

反馈信息 (电容>主控板)

在 1kHz 环路中会将电容电量、状态信息、底盘功率、裁判系统功率、最大可输出功率等信息进行打包,并以 1kHz 的速度反馈给主控板

```
static void generateTxDataNew(TxDataNew &td)
{
    td = {};
    td.statusCode = 0x00;
    td.statusCode = (psData.outputABEnabled << 7) |
        (ctrlData.refLoop.useNewFeedbackMessage << 6) |
        (ctrlData.wptStatus << 4) |
        (((ctrlData.limitFactor >= 4) ? 0b11 : (ctrlData.limitFactor & 0x03))
        << 2) |
        (errorData.errorLevel & 0x03);
    td.capEnergy = (adcData.vCap1f*adcData.vCap1f *
        (1/(CAPARR_MAX_VOLTAGE*CAPARR_MAX_VOLTAGE))) * 250U;
#ifdef WPT_HARDWARE
    if(psData.outputEEnabled)
        td.chassisPower = (adcData.pChassis1f - adcData.pWPT1f) * 64U + 16384U;
    else
        td.chassisPower = adcData.pChassis1f * 64U + 16384U;
#else
    td.chassisPower = adcData.pChassis1f * 64U + 16384U;
#endif
    td.refereePower = adcData.pReferee1f * 64U + 16384U;
    td.chassisPowerLimit = CAPARR::getMaxPowerFeedback() + rxData1.refereePowerLimit;
}
```

考虑到上位机需要得知无线充电功率来进行能量相关计算,所以在无线充电启用时将其累加在底盘功率上(性质相同)。

信息包长度为 8Byte,并根据新老数据格式不同向不同 CAN 地址发送信息以作区分。

```
struct TxDataNew {
    // 0x052 (useNewFeedbackMessage = 1)
    uint8_t statusCode; // 状态信息
    uint16_t chassisPower; // 底盘功率, 功率*64+16384 (-256W~+768W, 精度 0.015625)
    uint16_t refereePower; // 裁判系统功率, 功率*64+16384 (-256W~+768W, 精度 0.015625)
    uint16_t chassisPowerLimit; // 底盘最大可用功率(包括裁判系统)
    uint8_t capEnergy; // 电容现有能量, 0-255
} __attribute__((packed));
```

接收信息 (主控板>电容)

主控板以 10Hz 频率向电容发出指令，格式为：

```
struct RxData {
    uint8_t enableDCDC: 1;           // 允许启动 DCDC
    uint8_t systemRestart: 1;       // 系统重启
    uint8_t resv0: 3;
    uint8_t clearError: 1;          // 手动清除可清除的错误
    uint8_t enableActiveChargingLimit: 1; // 是否启用主动充电限制
    uint8_t useNewFeedbackMessage: 1; // 是否使用新的反馈消息格式

    uint16_t refereePowerLimit;     // 裁判限制功率，单位 W
    uint16_t refereeEnergyBuffer;   // 裁判能量缓冲，单位 J
    uint8_t activeChargingLimitRatio; // 主动充电限制比例（能量），0-255
    int16_t resv2;
} __attribute__((packed));
```

当收到命令后，进行解包并进行相应操作，例如当 systemRestart 为 1 时，会直接触发系统的重启：

```
if(rd.systemRestart)
{
    HRTIM::disableOutputAB();
    __disable_irq();
    while (true)
        NVIC_SystemReset();
}
```

CAN 总线保护

STM32G4 系列的 CAN 总线在错误累计一定数量后会进入 Bus Off 状态并不会自动恢复，所有在每次发送数据前通过读取 PSR 寄存器中的 BO 位来判断是否触发，如果触发则写入 CCCR 寄存器中的 INIT 位来恢复。

The Hong Kong University of
Science and Technology

3.5.2 交互

按钮交互

电容控制器在硬件上留有一个按钮，允许在不连接调试线的情况下进行操作，设计上长按 3 秒以上会进行 NVIC_SystemReset 重启整个系统，长按 1 秒以上、2 秒以下则会清除 ERROR_RECOVER_MANUAL 级别的错误。

蜂鸣器与 LED

电容控制器内置蜂鸣器三个 WS2812 LED 进行状态反馈。

在系统启动和功率级开关时，蜂鸣器会通过不同声音来发出提示。

进入错误保护时，根据错误等级不同，通过不同数量的高音+低音组合来提升错误等级和错误点。例如 B 侧短路保护报警音位 4 高+2 低。

3.6 超级电容组设计

3.6.1 需求分析

RoboMaster 规则限定超级电容组的要求如下

- 标称容量不得超过 2000J，实测容量不得超过 2200J
- 供电电压不能超过 30V

在实际设计过程中，对电容组还有其他要求

- 要有完善的保护外壳
- 体积尽可能小
- 实际容量尽可能大
- 放电功率尽可能大

3.6.2 电容选型

单体电压选择

现在可以购买到的超级电容单体电压一般有 2.5V、2.7V 和 3.0V 三种（锂离子电容由于功率过小且不能完全放电不考虑在内）。由于本赛季队内对于机器人体积压缩要求很高，所以优先选能量密度更高的 3.0V 电压单体。

单体组合方式

电容组总能量的及算法方式为

$$E = \frac{1}{2} \times \frac{C}{b} \times (bU)^2 < 2000J$$

其中 C 为单串容量， b 为串数， U 为单体电压。由于单体耐压越高，理论上最大放电功率越大（放电电流被 CM01 限制在 16A），所以满电电压尽量接近 30V，所以这里选择 10 串的配置来达到规则限定下的最大电压 30V。

根据公式反推，单体容量最大为 44.44F，在进行反复的配对组合尝试后，最终选择使用 3.0V 34F 的单体+10F 的单体来使得单串容量为 44F，此时 $E = 1980J < 2000J$ ，符合规则要求。

这种电容单体组合方式有 ETN TV 或 CDA 品牌可选，经测试发现 CDA 的电容单体容量表现都更为优秀，所以选择 CDA 的 CXP-3R106R-TWQ 和 CXP-3R0346R-TW 两个型号。

根据数据手册计算，理论最大瞬时电流为 $8.96 + 21.93 = 30.89A$ ，整体内阻为 $(68m\Omega \parallel$

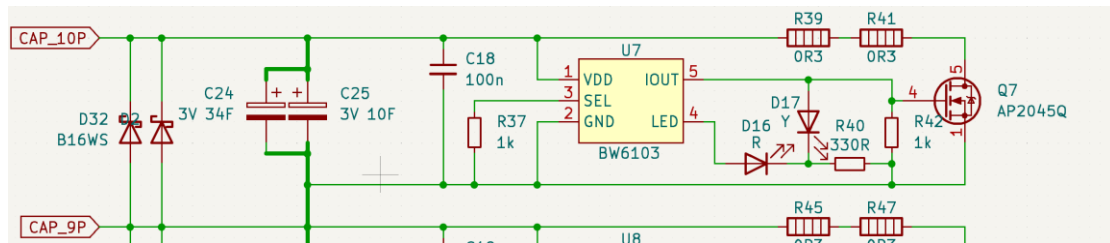
$45\text{m}\Omega \times 10 = 271\text{m}\Omega$ ，实测内阻约为 $150\text{m}\Omega$ 左右。总体性能可以满足要求。

3.6.3 原理图与 Layout

保护电路

均衡电路考虑电路精简性选用了 BW6103 芯片，为了实现更大放电功率使用了外置 MOS 管串联大功率电阻进行放电。

由于 BW6103 的警示灯在电容电压很高 (3.25V) 时才会亮起，所以在控制 MOS 的栅极上也加入一个黄色 LED 来指示泄放状态，方便调试，BW6103 的输出引脚电流可达 200mA ，足够驱动一个额外的 LED。



当某些电容单体在满电被放电均衡，其中的电荷量会减少，这会导致在整个电容组放电至接近 0V 时某些电容单体出现负压的情况，所以使用两个低 VF 的肖特基势垒二极管（图中 B16WS）进行保护。

外置 MOS 管选择耐压 20V 最大电流 65A 峰值 $\text{DFN}3\times 3$ 的 AP2045Q，选择原因是其栅极开启电压阈值较低，可以保证电容单体达到泄放电压时完全开启。

在整个电容组的输出放置一个 30A 的快断保险丝，放置电容组输出意外短路，此保险丝比电容控制器的输出保险丝 20A 要大，原因是电容组保险丝熔断更加危险（由于内部依然带电，保险丝熔断难以放电）。

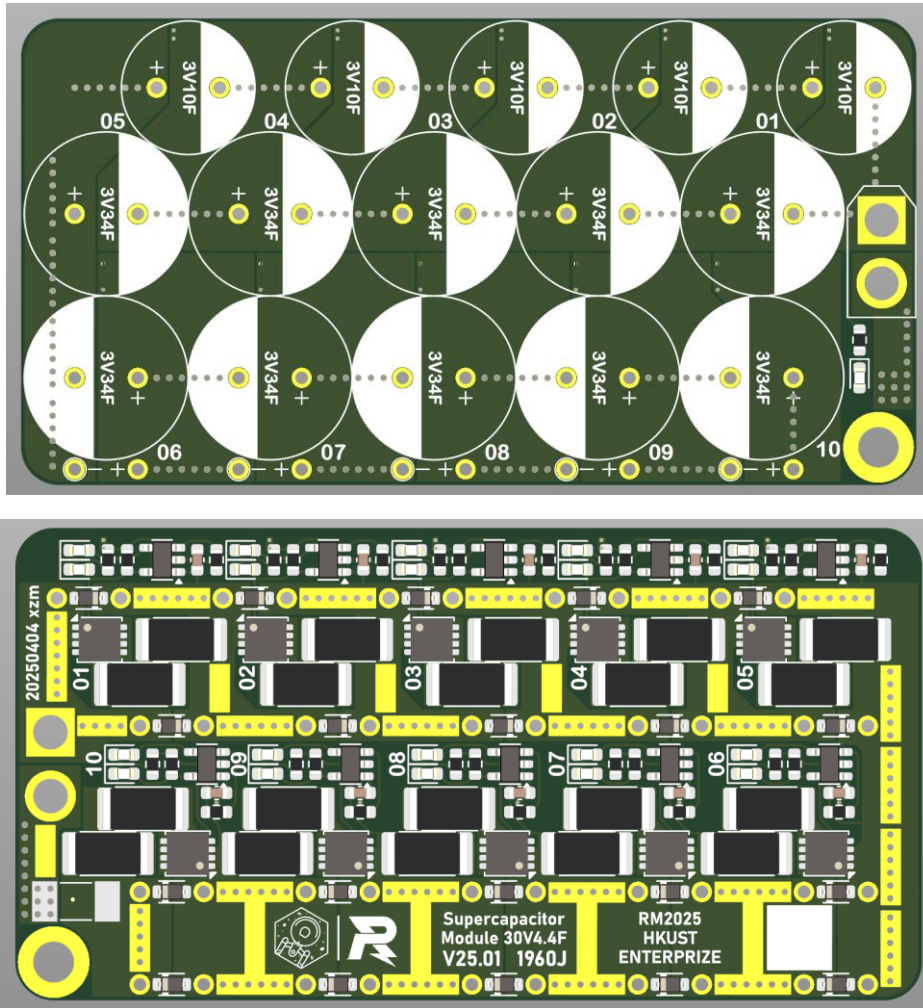
泄放电阻选用两个 2512 封装 1.5Ω 的厚膜电阻串联，最大放电电流约为 1A

PCB Layout

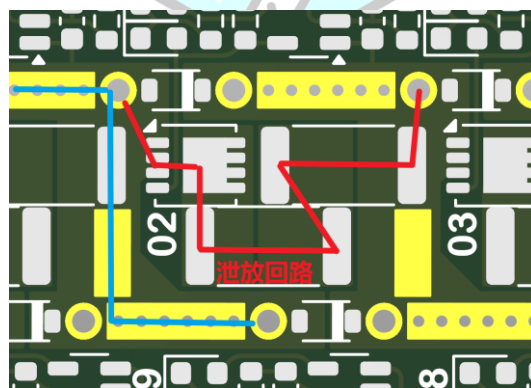
Layout 上主要考虑因素如下

- 主电流回路要尽可能粗，减小损耗
- 保护电路尽量对称排版，方便 Layout 和维修，并且降低焊接时焊错元件的风险
- 保护电路的功率回路尽量短

整体 Layout 如下：



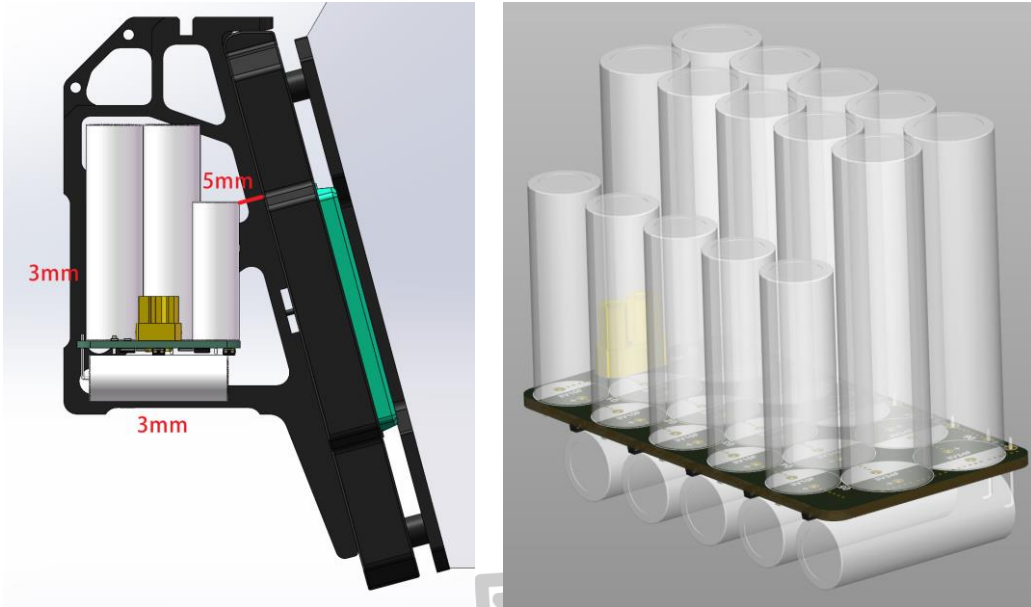
如图，电容泄放电流的 Layout 基本对称，并保证了泄放回路较短，对大功率回路在背面进行了开窗处理，正面全部为功率回路的覆铜。



电容组排布

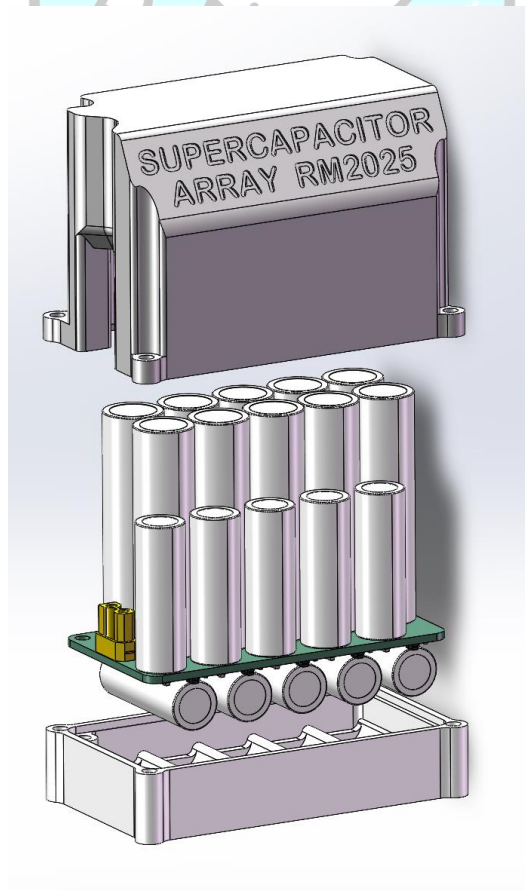
电容组的排布如右图所示，34F 的单体并列排布，10F 的单体一侧与 34F 的单体交错排布，另一侧在背面横向焊接，这样做可以最大程度地缩小体积。

这样的排布方式可以将电容组的体积完全控制在装甲板支架之内，相当于整车无需再底盘上预留额外空间，大大减小了超级电容系统的体积占用。



3.6.4 保护外壳

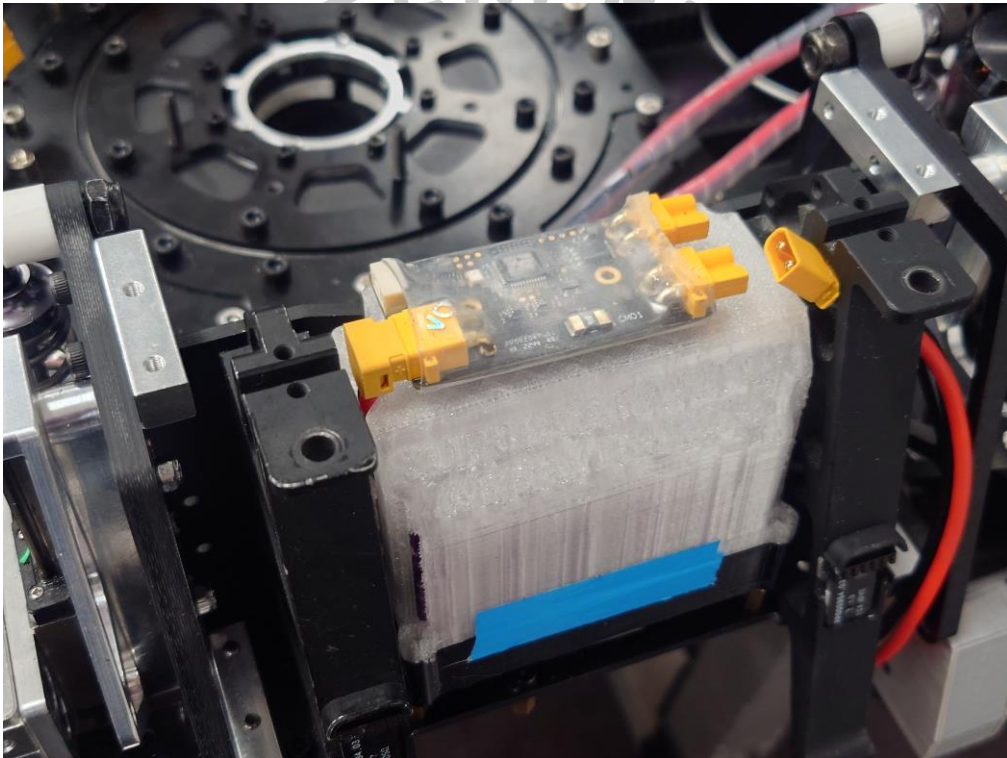
由于电容组单体较为瘦高，电容组外壳需要提供完善的保护和支持。设计时上下两个外壳为 TPU 材质，完全贴合电容单体。加之电容组内部使用 UV 胶进行单体与 PCB 板之间的加固，整个电容组的抗摔能力较强。



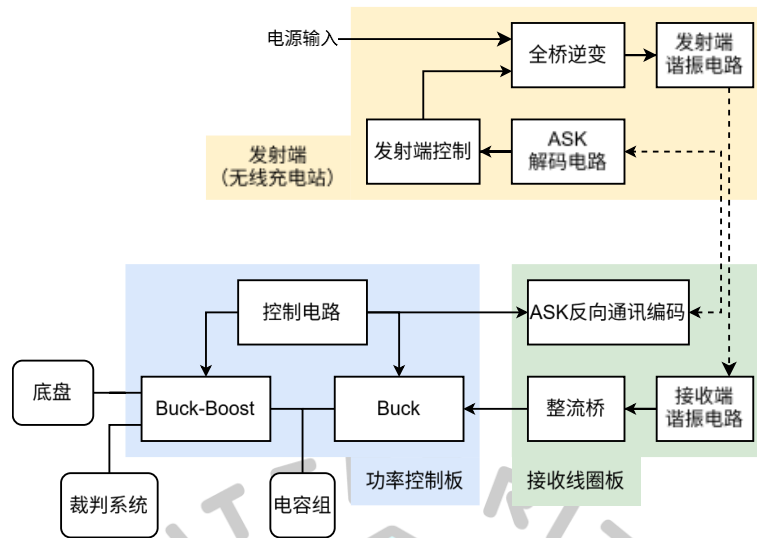
焊接好的实物图



安装效果展示 (完全嵌入装甲板内)



4. 无线充电系统

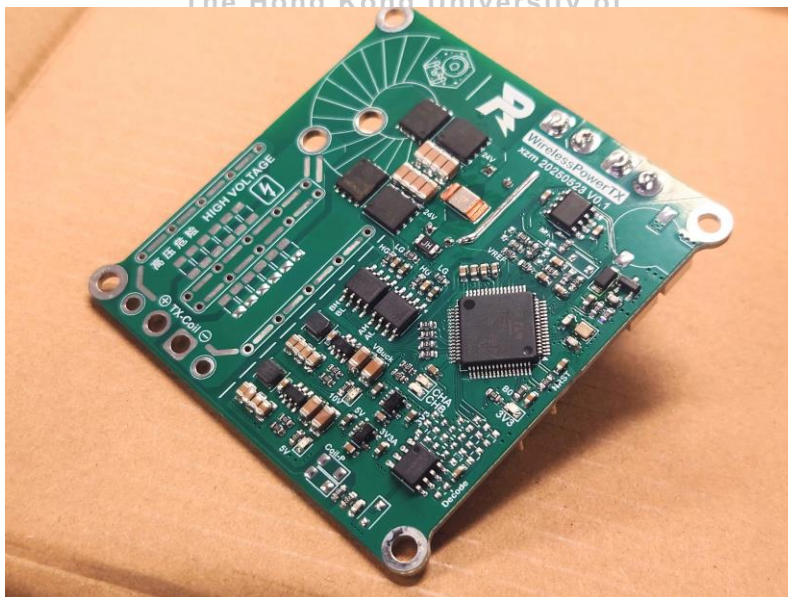


无线充电系统功率传输使用全桥逆变配合 LCC-S 架构实现功率传输, 通讯方面模仿 Qi 协议进行 ASK 和 FSK 通讯。本部分将主要介绍无线充电系统的发射端、接收端硬件、控制思路以及通讯的实现方式。

4.1 发射端

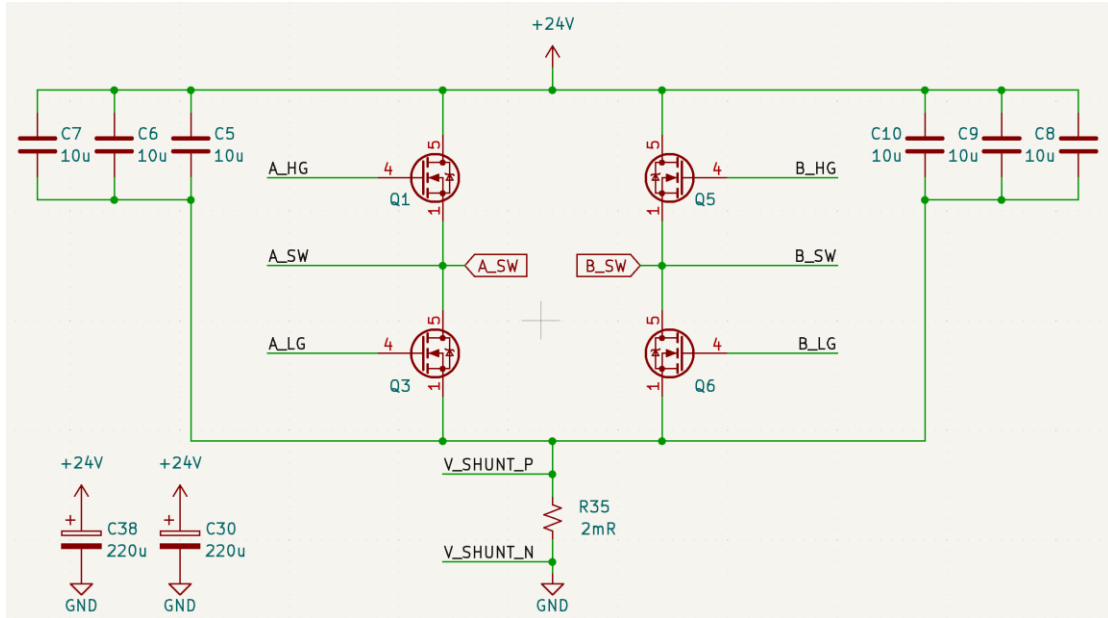
总体上, 发射端控制板主要由功率级 (逆变全桥)、谐振电路、采样与反馈电路、辅助供电、ASK 解码几大部分组成。MCU 选用与功率控制板上相同的 STM32474RBT3, 输入为 24V, 并预留有 24V 输出接口与 CAN 通讯电路 (如果对位机构需要电机)。

为了加快设计与迭代, 无线充电系统硬件方案很多与功率控制板相似, 说明会较为简略。



4.1.1 功率级及驱动

全桥部分原理图



MOS 选型

全桥使用了四颗 BSC070N10 作为开关管, 由于无线充电谐振频率不高(100-150kHz 级别), 所以选型相对较为宽松。Vgs 10V 下 30nC 的栅极电荷量、6.0mΩ内阻、最大过流 80A、VDS 耐压 100V 对于此应用完全足够。

滤波电容

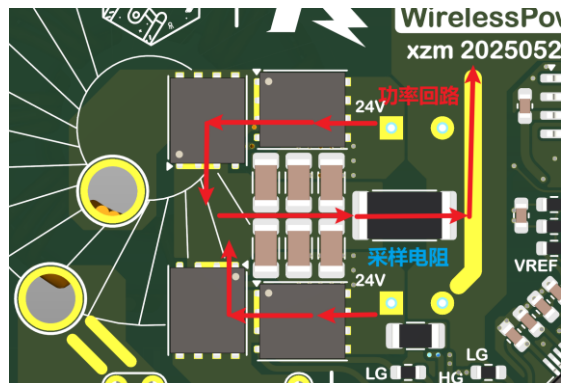
滤波电容每侧采用三颗 50V 10μF 1206 的贴片电容滤除高频噪声, 母线上采用两个 220μF 固态电容用来稳压和滤除低频干扰。

电流采样

电流采样选择在两个低位 MOS 的 S 极进行低位采样, 阻值选择 2mΩ (与功控板一致)。

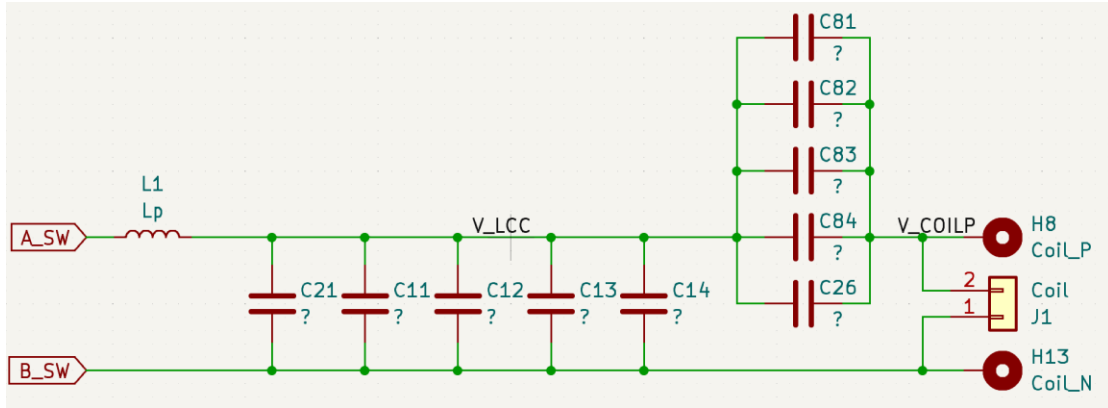
PCB 布局

PCB 布局上最小化了大功率回路的面积, 供电从背面通过固态电容的滤波进入高位 MOS, 陶瓷电容紧贴半桥的两端。



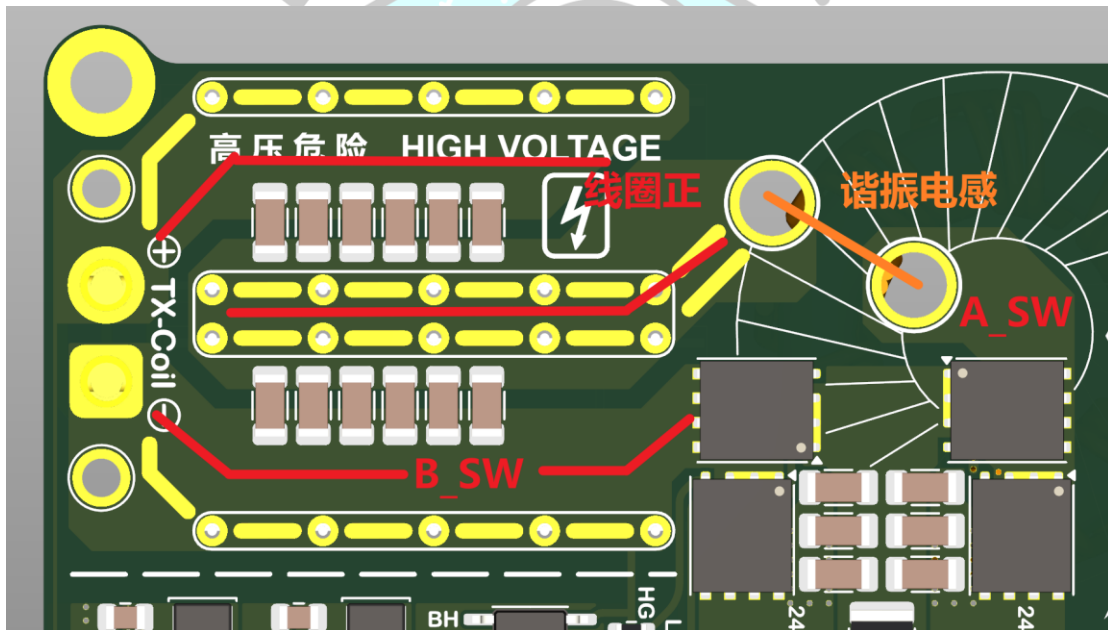
4.1.2 谐振电路

原理图



发射端谐振电路为 LCC 拓扑，通过谐振电感和电容后输出连接到线圈。

PCB 布局



谐振电感选型

在第一版测试板中，发现谐振电感电流很大且发热较大，所以在这一版中选用 90125 大小的铁硅铝电感。

后来在与华南虎战队的交流中得知此处需要使用铁粉芯磁芯并配合利兹线制作谐振电感，更换后效果极好，几乎没有发热的同时传输效率提升至 85%-86%。

谐振电容选型

谐振电容预留了 CBB 电容与 1206 尺寸 COG（由于 X7R 和 X5R 容值会变所以只能使用

COG) 电容的焊盘位。CBB 电容的优势是价格较低，且耐压很高，但缺点是受热时稳定性较差，且过热严重会直接导致电容报废。而高耐压的 COG 电容价格较高，但性能更加优秀。在测试后发现 CBB 电容发热比较严重，且出现过电容过热失效的情况，考虑到高耐压的 COG 电容用量很小，后续全部采用 COG 材质，耐压至少 100V 的谐振电容。

线圈选型

规则限制线圈能量交互面不得大于 10000 平方毫米，选择外径 80mm，内径 30mm，感值为 14uH 的线圈，发射接收端对称。

实测加入隔磁片后自感值约为 20μH，通过正反串接的方法测得线圈相距 2cm 时互感约为 5 μH。

谐振参数计算

为了满足理论无功为 0 的要求，需要满足如下关系：

$$\omega L_p = \frac{1}{\omega C_p}$$

$$\omega C_s = \frac{1}{\omega(L_{coil} - L_p)}$$

为了尽量减小电感与电容的取值，将谐振频率最大化，取规则上限的 148.5kHz

$$\omega = 2\pi f_s$$

在传输最大功率 80W 时，接收端电源需要可以达到 30V。发射端通过移相全桥，等效电压为 0V-24V，将谐振电感取 6μH，电压增益为 3.3 左右

$$Gain = \frac{L_{coil}}{L_p}$$

谐振电感取 6μH 后，实际使用电桥测量感值在 100kHz 下为 5.77μH，根据以上关系可以计算其他计算参数：

$$C_p = 199.07\text{nF} \quad C_s = 80.72\text{nF}$$

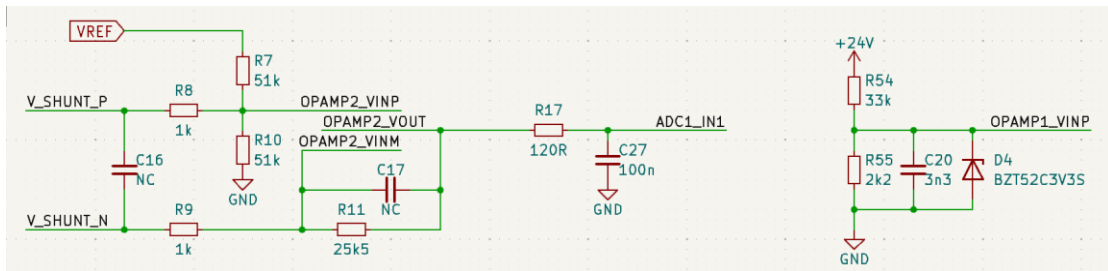
同时，接收端 S 电容参数为

$$\omega L_s = \frac{1}{\omega C_s} \rightarrow C_s = 57.43\text{nF}$$

4.1.3 采样与反馈

采样与反馈部分需要采样全桥的电流和输入母线电压以计算发射端功率。与功率控制板相似，无线充电发射端也充分利用了 STM32G4 的内置运放。

电压和电流采样



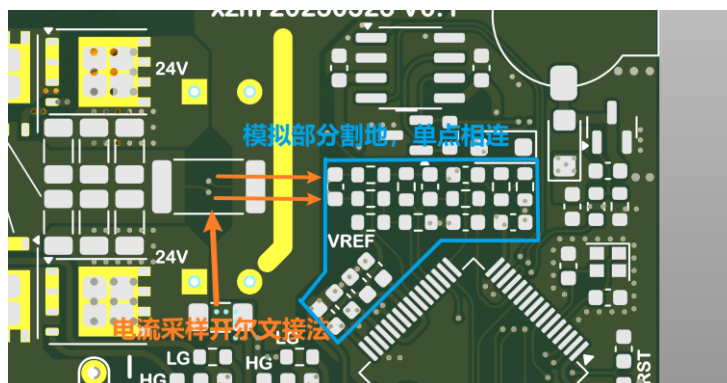
电压和电流采样都可以使用 MCU 内置运放实现，其中电压采样经过分压电阻和滤波电容后进入运放跟随，随后进入 ADC 采样，而电流采样使用内部运放搭建差分放大器，参数如下：

- 放大倍数 $25.5k\Omega / 1k\Omega = 25.5 V/V$
- 偏置电压 $2.9V / 2 = 1.45V$
- 理论量程绝对值 $\frac{2.9V}{2} \div (2m\Omega \times 25.5 V/V) = \pm 28.43A$
- 理论分辨率 $\frac{2.9V}{4096} \div (2m\Omega \times 25.5 V/V) = 13.9mA$

电压采样：

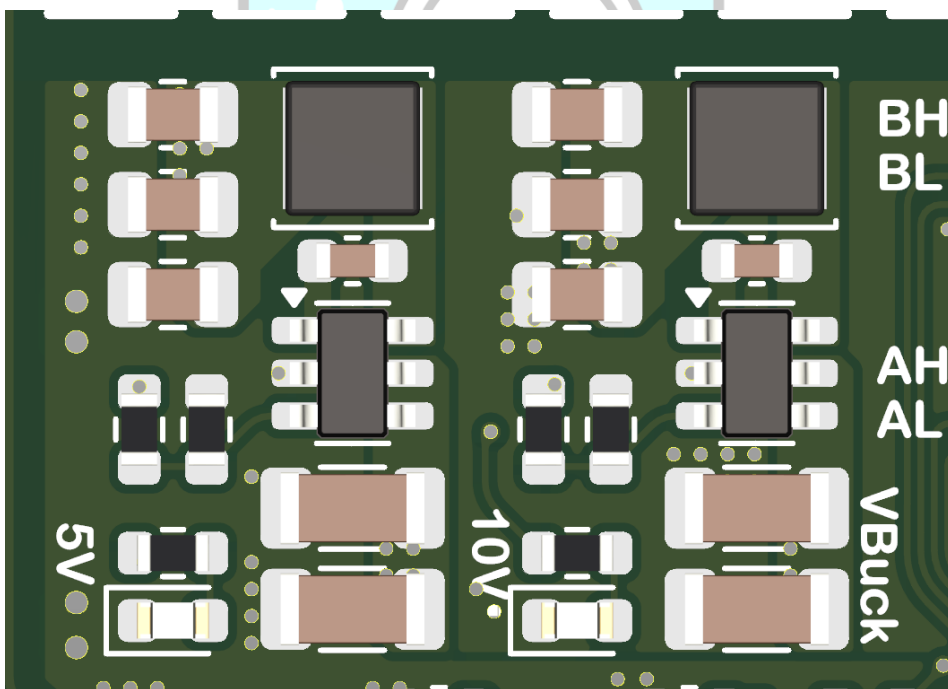
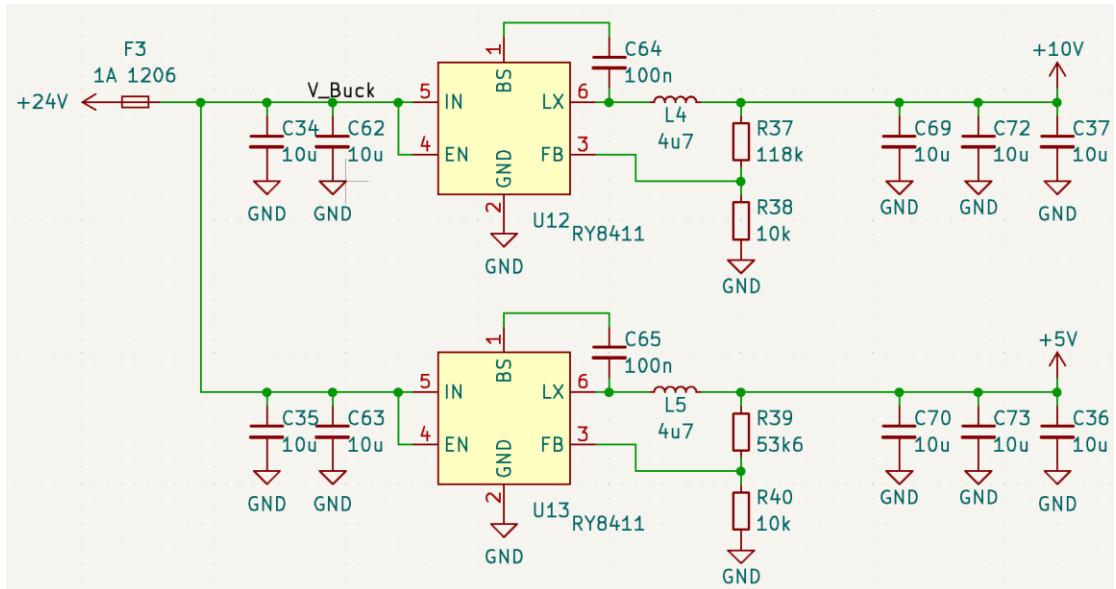
- 缩放倍率 $(33k\Omega + 2.2k\Omega) / 2.2k\Omega = 16V/V$
- 最大量程 $2.9V \times 16 = 46.4V$
- 一阶滤波截止频率： $\frac{1}{2\pi(33k\Omega \parallel 2.2k\Omega) \times 3.3nF} \approx 23.38kHz$
- ADC 理论分辨率： $46.4V / 4096 = 11.328mV$

PCB 布局



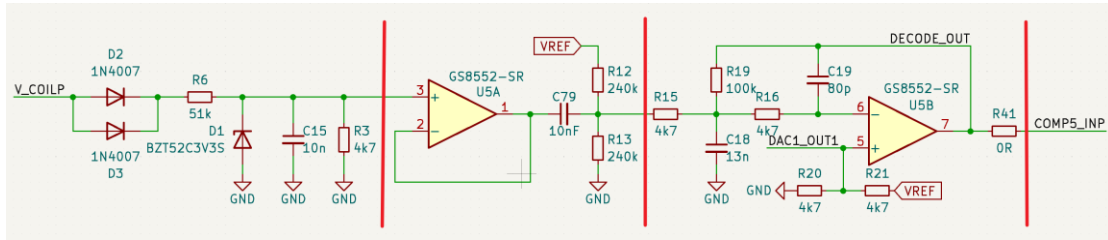
4.1.4 辅助供电

辅助供电采用与功率控制板相同的配置，使用两个 RY8411 分别产生 10V 和 5V



为了进一步减小噪声，Buck 输入地与功率地相连，而数字地与功率地在 Buck 输出电容处再相连。其它设计与功控板相似。

4.1.5 ASK 解码电路



ASK 通讯通过将信息编码在线圈电压幅值上来进行反向通讯，解码电路共分为四个阶段。

峰值取样电路

考虑到线圈电压峰峰值最高可达 180V 左右，峰值取样电路使用两个 SS36 串联（与图中不同，因为实测 1N4007 反向恢复过慢效果不佳），达到共 120V 的耐压。

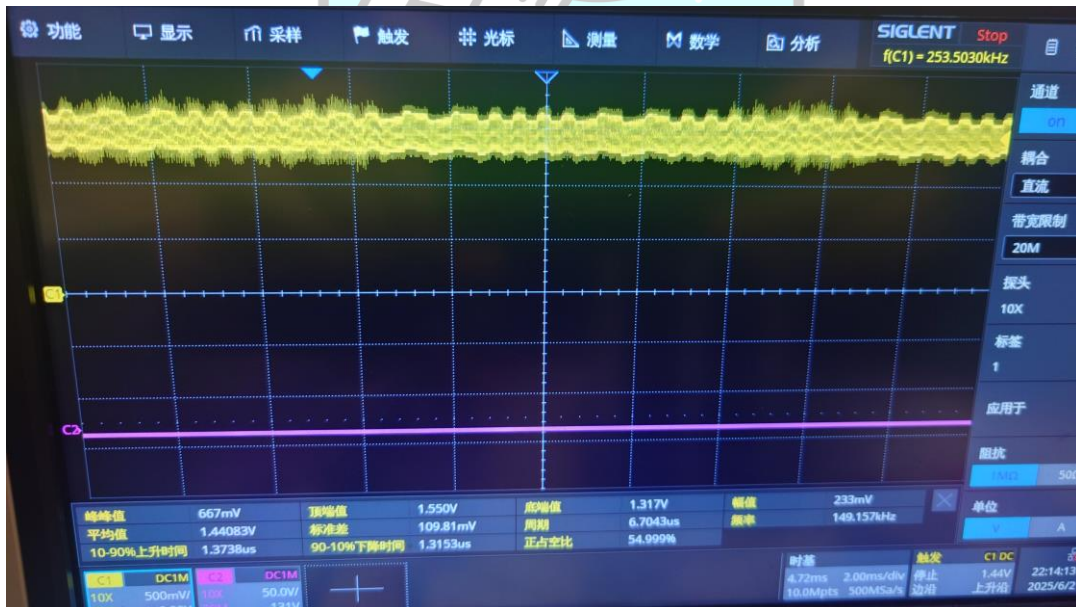
通过分压电阻后向一个电容从进行充电，进行峰值取样，由于滤波截止频率较高，实际波形在峰值附近有较大波动，此波动将在后级电路中滤波处理。

在运放输入端并联一个稳压二极管，防止电压过高损坏运放。

经过反复尝试，分压电阻选择 51kΩ 和 4.7kΩ，电容选择 10nF

电压跟随器

对峰值取样电路的输出波形进行跟随，减小后级的输入阻抗，输出波形如下：



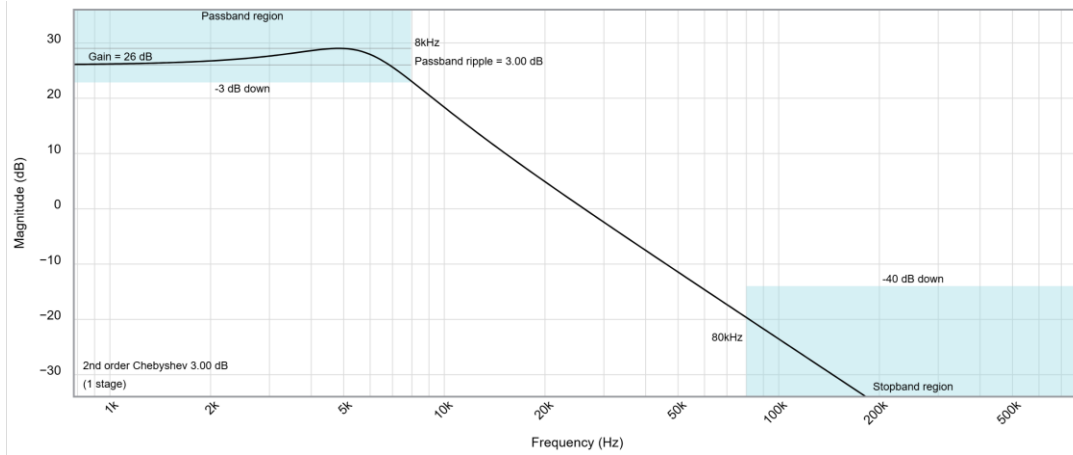
电压归中

滤除直流分量，并把信号的平均值限定在 1.45V 左右（参考电压一半）

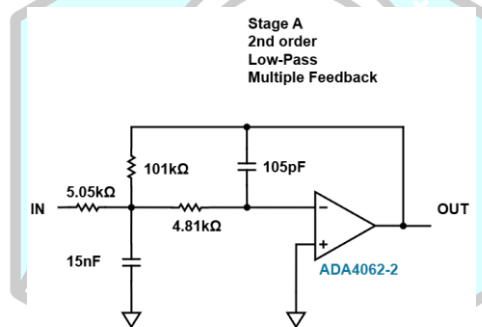
高通滤波电阻取 240kΩ（等效 120kΩ），电容选择 10nF，截止频率在 100Hz 左右。

滤波放大电路

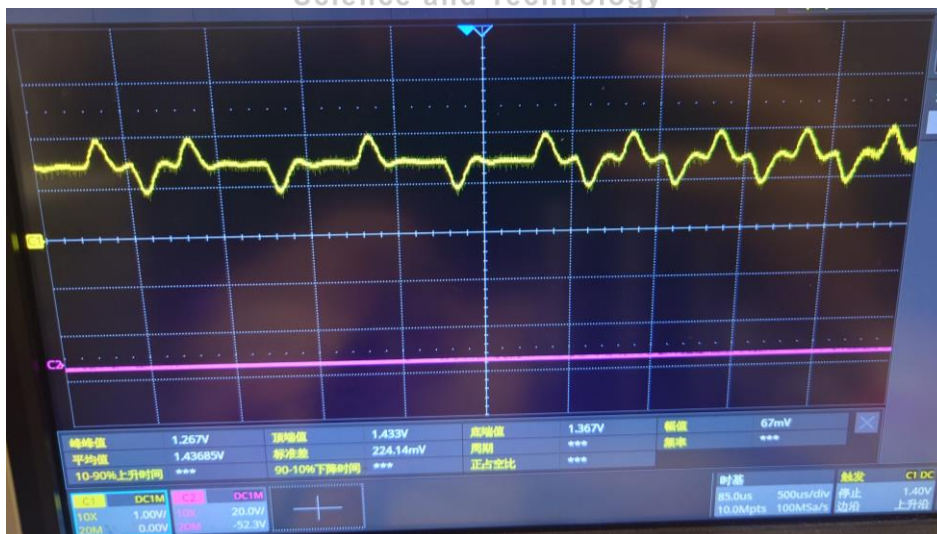
滤波放大电路的目标是滤除谐振频率的信号，只保留 8kHz 以下 (ASK 通讯) 的信号。使用 Multiple Feedback 架构的二阶有源滤波，使用 Analog Device 的 Analog Filter Wizard 进行计算，设计结果如下：



滤波器参数如下：

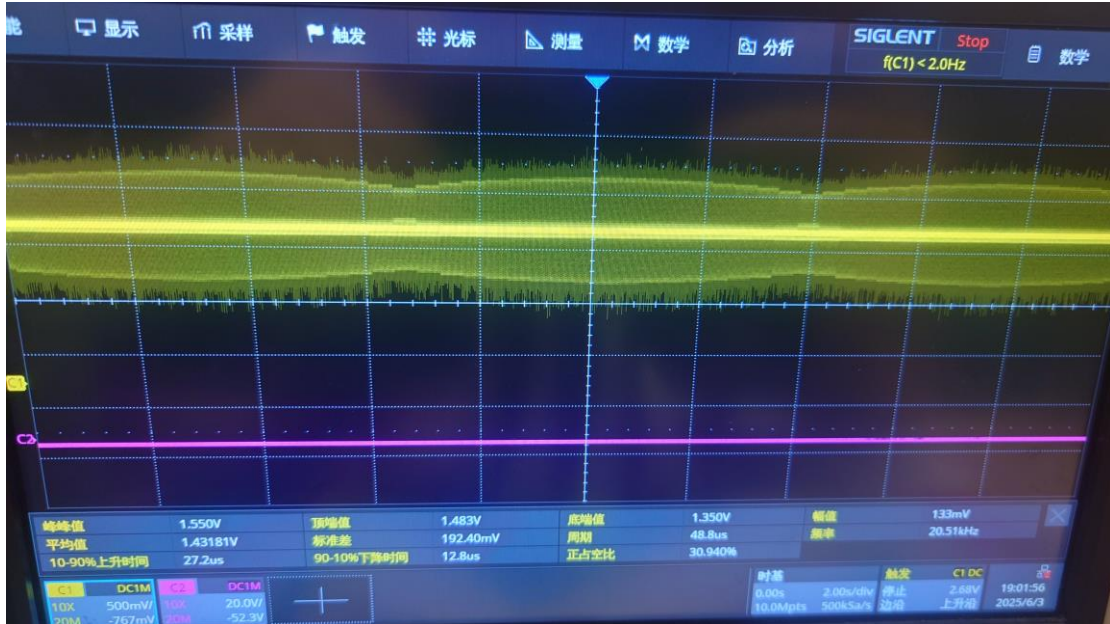


运放的正输入端为输出的电压偏置，这里使用两个电阻对参考电压进行分压，输出偏置为 1.45V。输出结果如下图所示 (此波形与 Qi 协议常规波形不同，下文会具体说明)：



功率波动测试

为了测试通讯的稳定性，将发射功率在 10W-100W 间以 10W 步进反复切换，硬件解码电路输出波形幅值波动较小，表现比较稳定：



4.1.6 功率级硬件配置

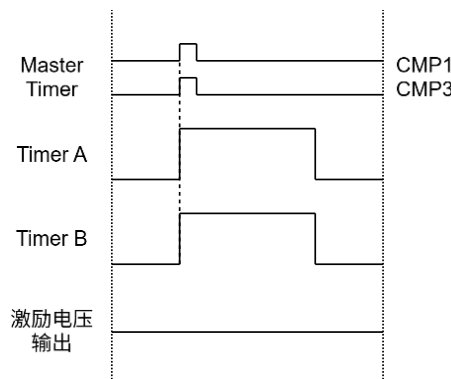
功率级通过移相全桥的方式实现输出功率的控制，通过 STM32474 的 HRTIM 进行控制。

HRTIM 配置与移相全桥实现

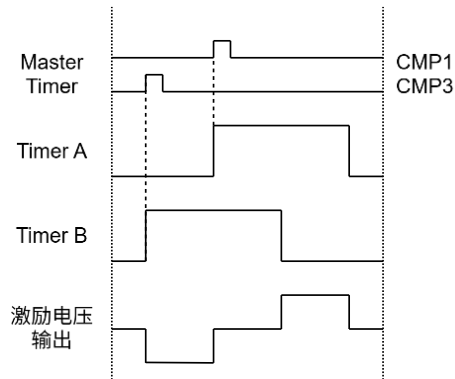
使用 TimerA 和 TimerB 分别控制两侧半桥的输出，固定占空比为 50%，通过 Master Timer 来动态控制两个子 Timer 的相位，实现中心对齐的移相全桥输出。

周期 148.5kHz，周期配置为 36633

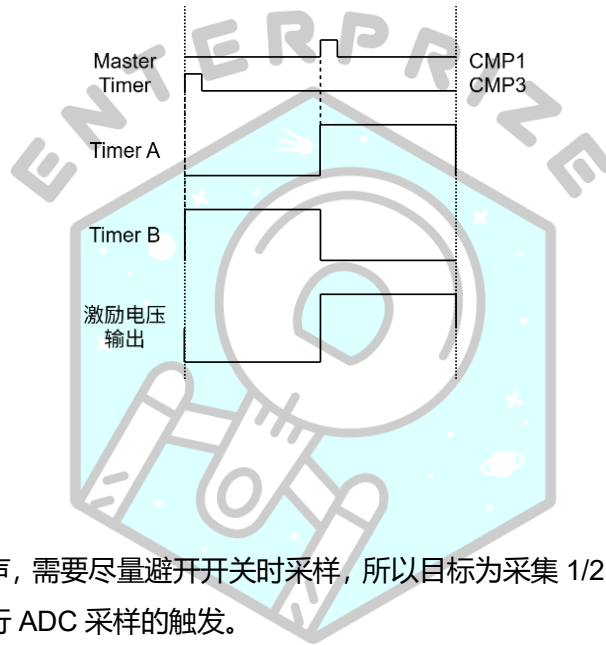
使用 Master Timer 的 CMP1 和 CMP2 来触发两个子 Timer 的复位,例如当 CMP1 和 CMP3 都为 1/4 周期时，可实现相当于 0V 电压的输出



例如当 CMP1 和 CMP2 分别为 12.5%和 37.5%时，相当于激励电压为 $24V \times 0.5 = 12V$



例如当 CMP1 和 CMP2 分别为 0%和 50%时，相当于激励电压为 $24V \times 1.0 = 24V$



ADC 触发

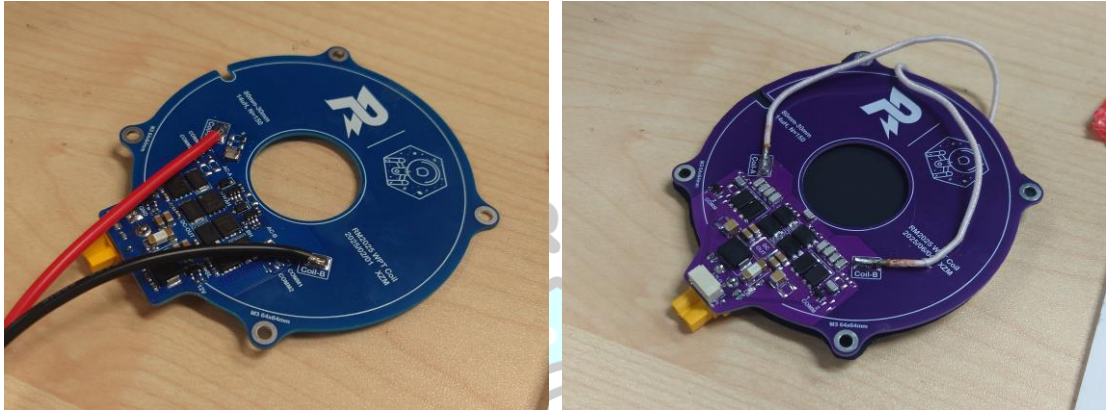
为了最小化采样噪声，需要尽量避开开关时采样，所以目标为采集 1/2 处的量，使用 Master Timer 的 CMP2 进行 ADC 采样的触发。

由于一阶滤波相位延迟 90° ，在每周期的 3/4 处采样，对应 Counter 值为 27475，ADC 采样时间为 12.5Cycle，为了最小化电流误差，应提前 6.25cycle 开始采样，ADC 频率为 $170Mhz/4$ ，6.25Cycle 对应 HRTIM 的 counter 偏移值为 800，故 CMP2 值为 $27475 - 800$ 。

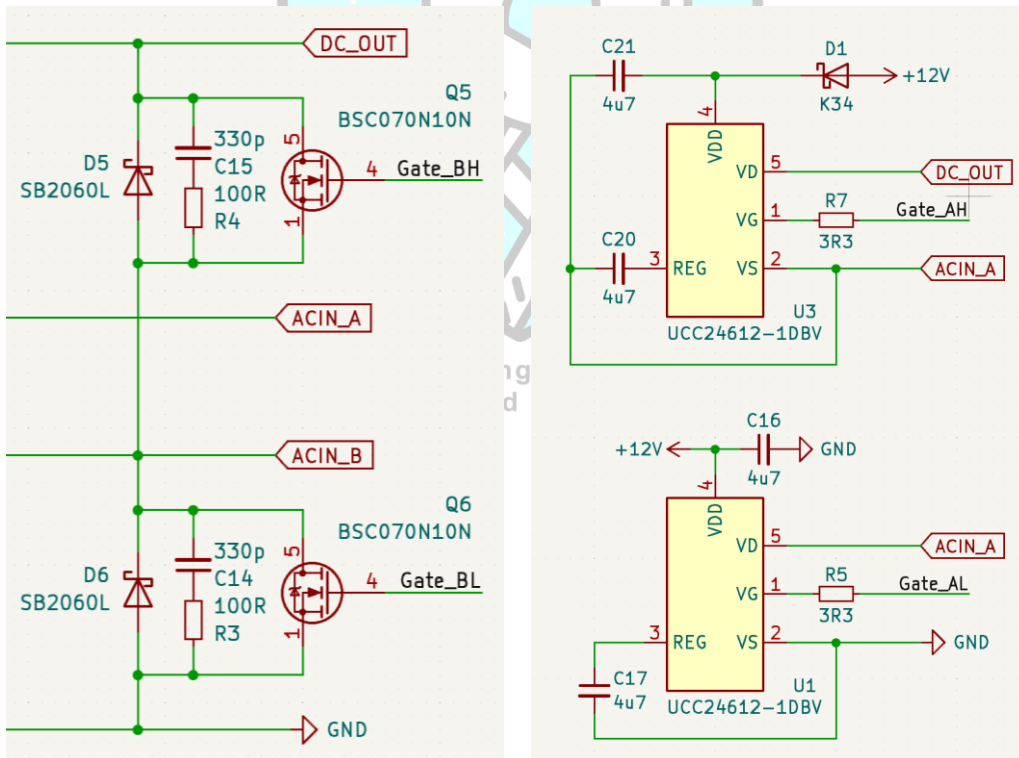
4.2 接收端

4.2.1 接收端线圈板

线圈板共进行两次迭代，第一版为主动全桥测试版（左图），由于测试发现主动全桥对效率提升很小，所以在第二版删去了主动全桥电路（右图）。

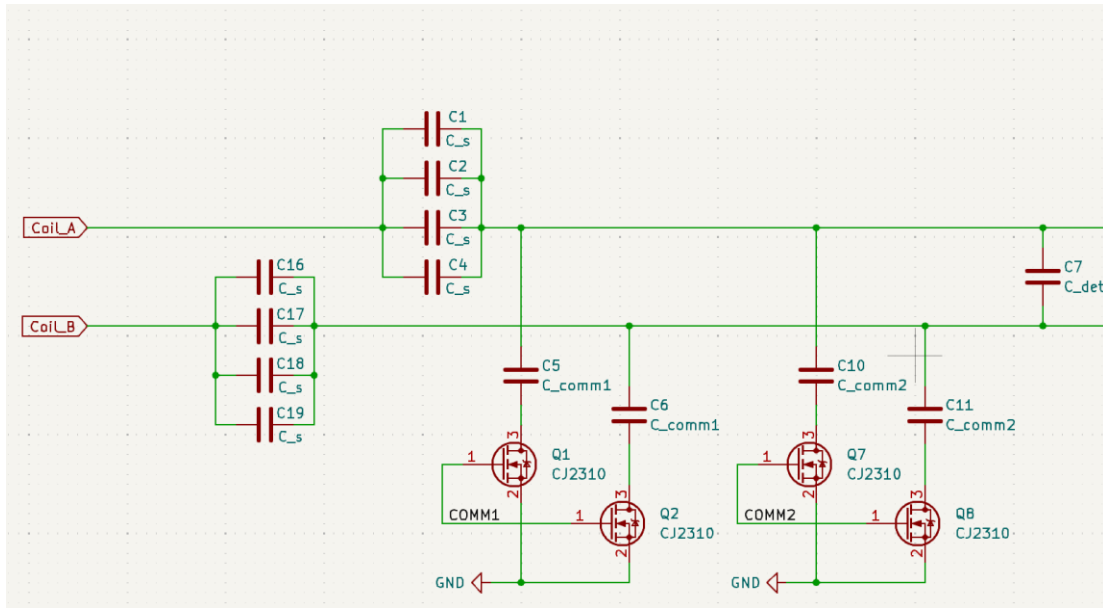


主动全桥



主动全桥整流通过四个 UCC24612 同步整流控制器实现，如图所示为整流桥的一侧，参考数据手册分别实现高位与低位的开关。但是后来测试后发现器对整体效率提升很小，且需要提供 10V 供电（增大复杂度），整体收益较小，故在后续版本中移除。

ASK 编码电路与谐振



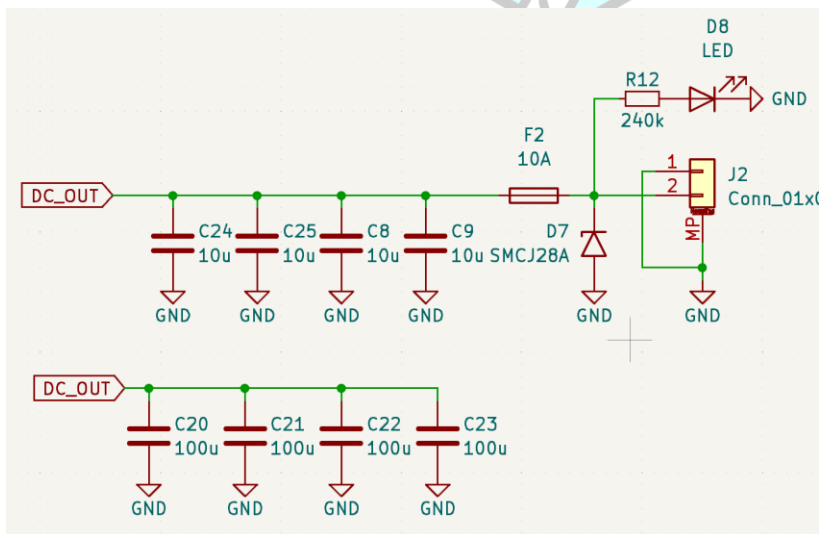
串联谐振电容放置在线圈两端，这样设计是为了可以使用两个 50V 耐压谐振电容串联，等效为一个容量减半的 100V 耐压电容以降低成本。

ASK 编码电路通过放置两对不同大小的谐振电容，使得在功率较低时可以加大通讯信号的幅值便于解码，但是实际测试中发现一对 6.8nF 电容已经可以实现稳定通讯。

开关 MOS 管使用 CJ2310，耐压 60V，最大电流 3A，防止击穿。

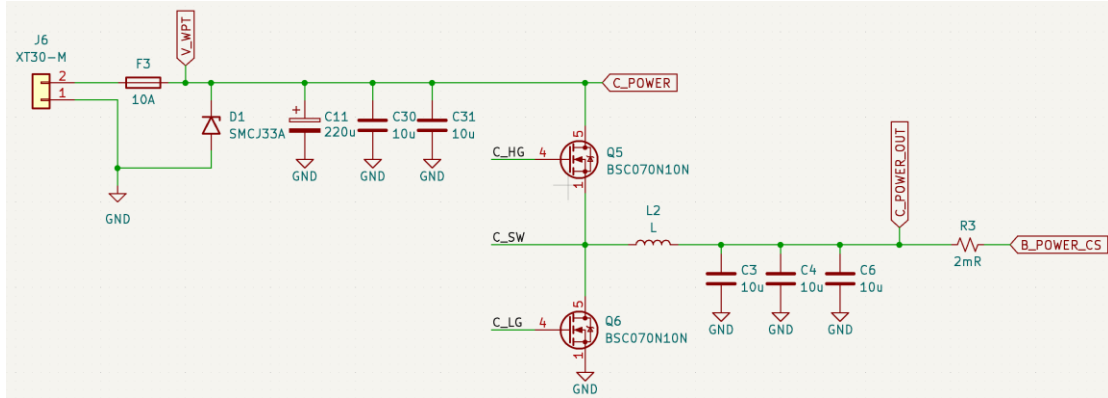
输出滤波

输出部分使用小电容+大电容的方式进行滤波和储能，并通过 TVS 和保险丝进行防护，当发射端失控时，DC_OUT 电压升高，超过 TVS 击穿电压后会触发保险丝熔断保证输出安全：



4.2.2 功率控制板无线充电部分

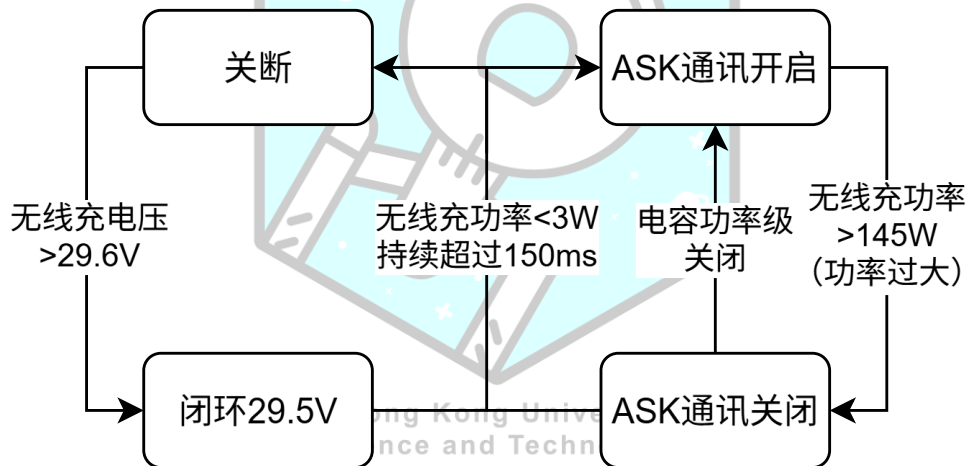
功率级



无线充电功率 Buck 电路使用 STM32G474 HRTIM 的 TimerE 进行控制。将无线充电全桥输入降压给电容组充电。功率电路如图，其选型和控制等与功控板相似，不再详细说明。

控制方式

无线充电 Buck 使用功率环直接控制，其核心为增量式 PID，闭环无线充电输入的电压：



此曲线可以实现将无线充电输入端电压控制在 29.5V，保证其电压一直高于电容组电压，可以通过 Buck 给电容组充电。

当功率超过 145W 时，将触发功率过大的保护，具体方式为直接关断 ASK 的输出，从而使发射端关断输出，具体通讯相关的内容将在后文具体说明。

4.3 控制与通讯算法

4.3.1 FSK 通讯

FSK 通讯通过发射端轻微改变谐振频率实现，接收端通过检测频率来解码信息。具体实现上，接收端将接收线圈的一极经过电阻分压进入 STM32 内置的比较器比较，并通过比较器输出触发一个 Timer 的 Capture Unit 进行采样，将当前的 Timer Counter 通过 DMA 传入一个数组中。最后通过软件计算频率。

由于本设计的控制无需 FSK（即发射端到接收端的正向通讯），故仅对上述方法做了可行性验证，并做了硬件预留。

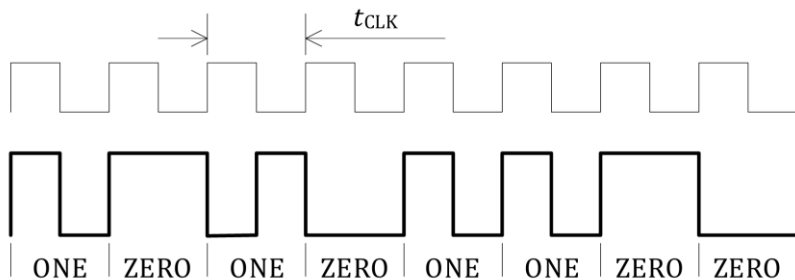
4.3.2 ASK 通讯

ASK 通讯通过接收端开关通讯电容来影响发射线圈的幅值，并通过上文所示的 ASK 解码电路解出通讯结果。

通讯格式

由于硬件原理限制，ASK 信道只能传输交流信号，本设计中直接使用 Qi 协议中的通讯格式：

Figure 3. Example of a differential bi-phase encoding scheme



如图，以 2kHz 的频率进行输出传输，每周期开始时进行一次电平翻转，如果 bit 为 1，在周期一半时在进行一次翻转。

解码电路输出的解码方式

如上文中的波形图所示，本设计中的解码电路将信号的上升沿转化为上凸的波形，将下降沿转换为下凹的波形，这与符合 Qi 协议的一般解码电路不同。

如图为 NXP 的 AN4701 中解码电路的输出，在开始处，由于隔直通交的截止频率过低等原因，导致电压偏置回中值电压的速度过慢，此方法通过比较器可以直接获得源通讯波形相同的波形，但是在调试过程中，发现此方式难以在功率变化时做到稳定解码；考虑到发射端

MCU 的负载较低，所以在本设计中直接通过快速的 ADC 采样来识别解码电路中的“上凸”和“下凹”特性。

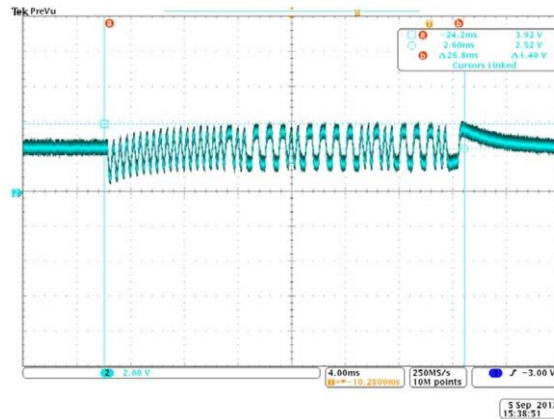
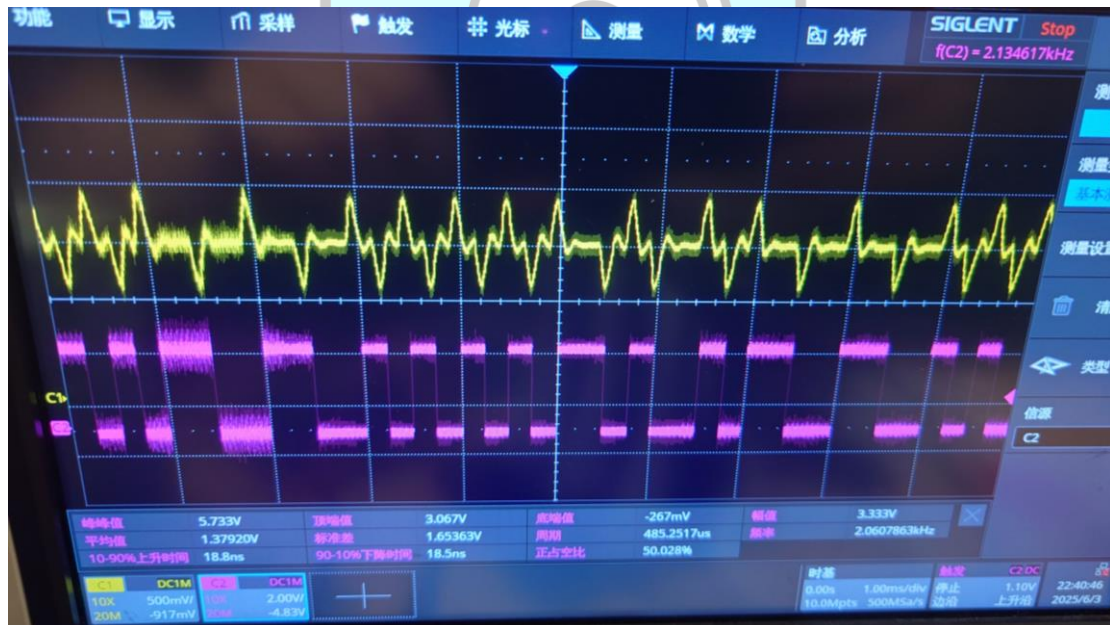


Figure 13. Communication wave after the second-order low-pass filter

具体实现上，在高频环路中以开关频率的一半（主中断频率）采样解码电路的输出，当电压大于或小于一定阈值时，对计数器进行累加，当计数器达到一个阈值，则认为电平翻转，最终的解码效果非常稳定，波形如图（黄色波形为硬件解码电路输出，紫色波形为软件解码结果输出）：

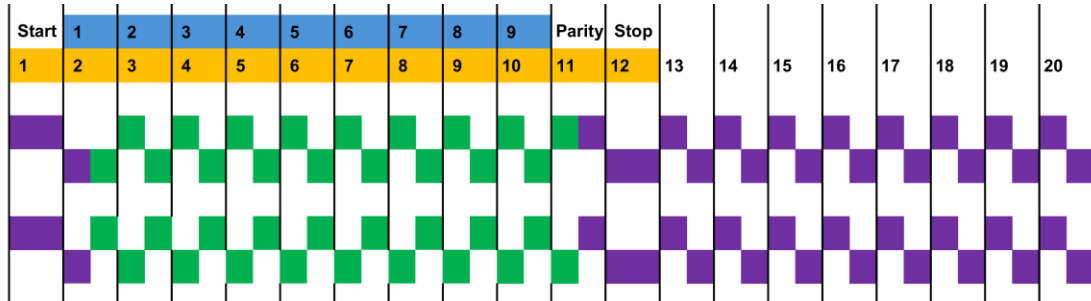


实现此效果的稳定阈值为：

```
uint16_t upperThreshold = 2310U; // 上限阈值
uint16_t lowerThreshold = 1766U; // 下限阈值
uint16_t cntThreshold = 6U; // 连续采样次数阈值
```

数据包格式

为了保证整个无线充电系统的安全，接收端需要以尽可能快的频率进行功率反馈。本设计中以 20bit 为周期（100Hz 频率），12bit 包长发送反馈数据。数据包格式如下：



其中橙色 12bit 代表数据包，蓝色代表有效数据，长度为 9bit。每个数据包中：

- 1bit 起始位 (0)
- 9bit 数据位
- 1bit 校验位 (如果 1 的个数为奇数则为 1)
- 1bit 停止位 (0)

在解包时，通过识别固定不会变的波形（如图所示的紫色部分），特征为“8 个连续先高后低的 1+一个双高的 0”即为起始位。

此格式最大化利用了包的长度，仅会出现一种情况导致误识别起始位（即上图中上面的数据包），但此时解码得到的数据仍然可以保持结果相同。

有效数据内容

有效数据长度为 9bit，为了最大化功率反馈的精度以供发射端进行效率监测，将其中 8bit 用于传输功率信息。规定功率量程为 150W，将接收端读取到的 0-150W 数据映射到 0-255 的值上进行反馈。

剩余 1bit 用于接收端向发射端反馈开关需求，例如当电容组充满电时，此 bit 为 0，通知发送端停止功率输出，仅输出最小功率维持通讯。

功率接收端/ASK 发射端编码

由于功率控制板中已有多个中断源，为了不引入额外的中断，将原有的 1kHz 低频中断改为 4kHz，1kHz 中断中的其他代码通过一个计数器分散到四次中断中运行。在 4kHz 的中断中，每 40 次触发（100Hz）会触发一次打包。

打包时会首先将功率数据转化为 uint8，与开关 1bit 组合，并计算校验位

```

packedPower = (uint8_t)(wptPower * (255.0f/100.0f)); // 将功率转换为 0-255 范围
parity = (powerRequirement & 0b1) | (packedPower << 1);
parity ^= (parity >> 8);
parity ^= (parity >> 4);
parity ^= (parity >> 2);
parity ^= (parity >> 1);

askData.txMessage = 0x0
    | (powerRequirement & 0b1) // 功率要求
    | (packedPower << 1)      // uint8 格式的功率
    | ((parity & 0b1) << 9); // 奇偶校验
    
```

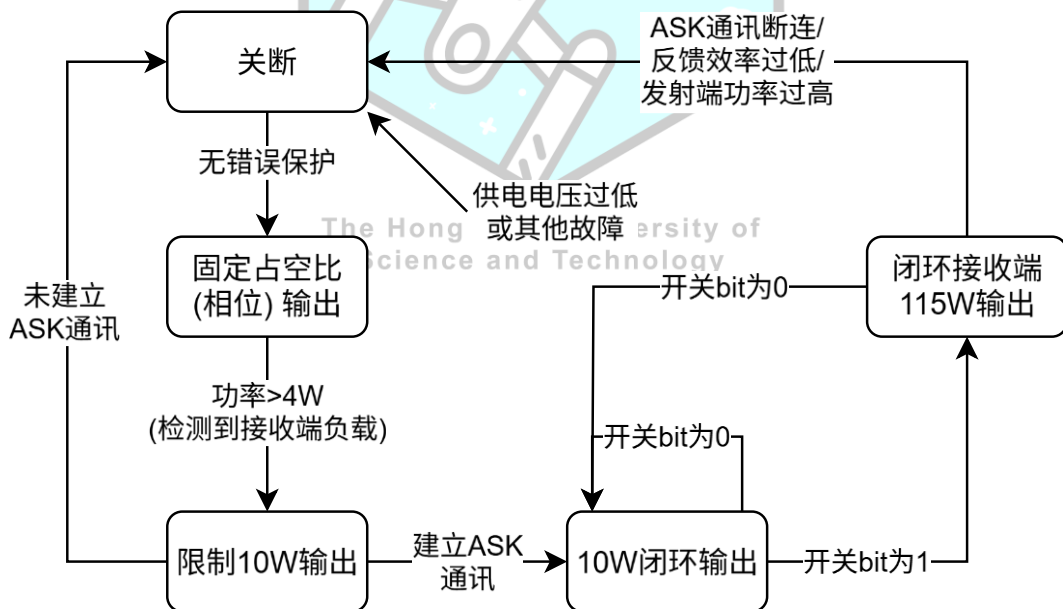
同时，通过写入 GPIO 的 BSRR 寄存器将输出拉高（开始时一定为高）

在之后每当 index 为偶数时进行一次 GPIO 翻转，当 index 为奇数时根据数据包内容判断是否需要翻转。

功率发射端/ASK 接收端解码

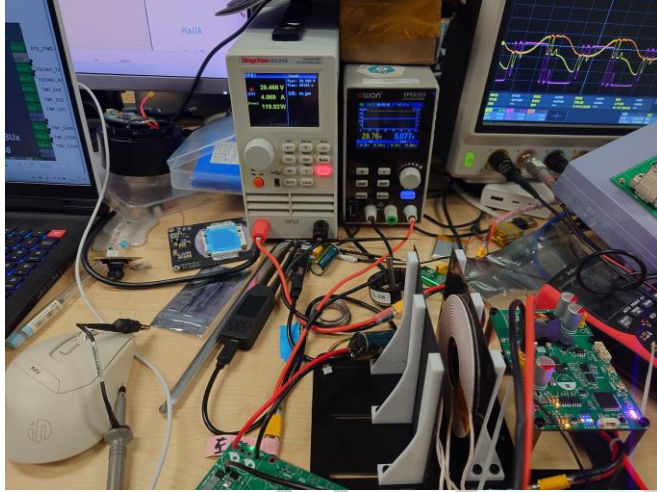
接收端在通过 ADC 解码出信号的上升与下降沿后，通过 Timer 来获取脉冲长度，最终获得 1 或 0 的序列，寻找包头后即可解码出功率接收端的信息。

4.3.3 发射端状态机

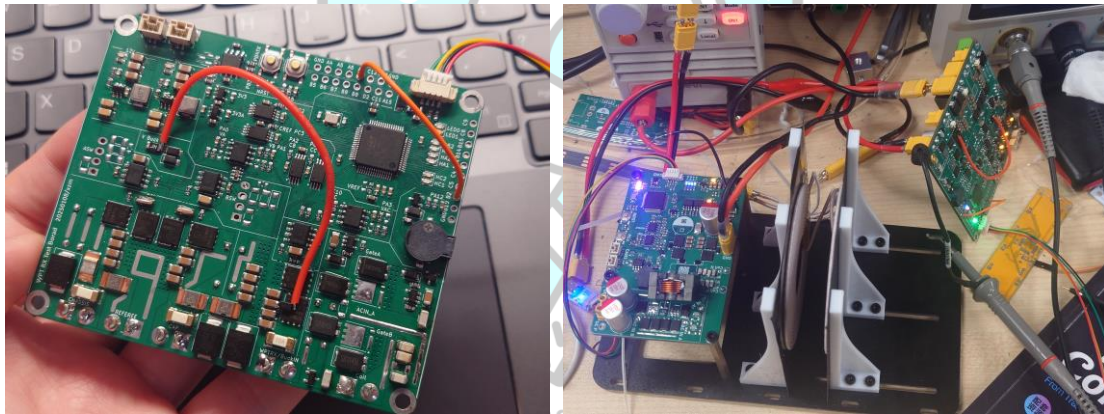


5 迭代/测试记录

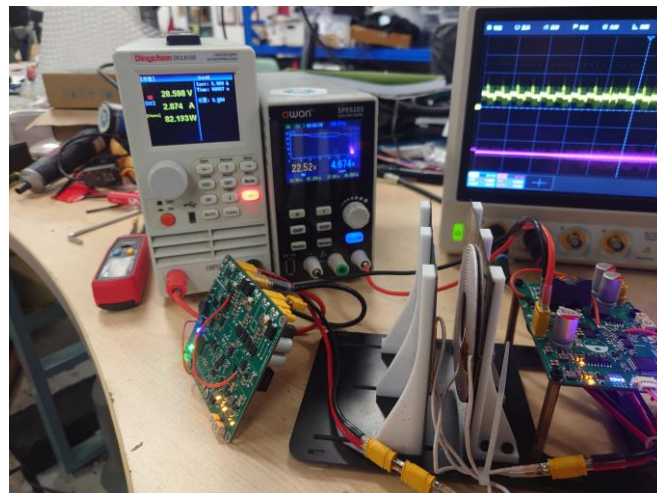
无线充电发射端 V1 功率传输测试，接收端功率 119.92W，效率 82.12%



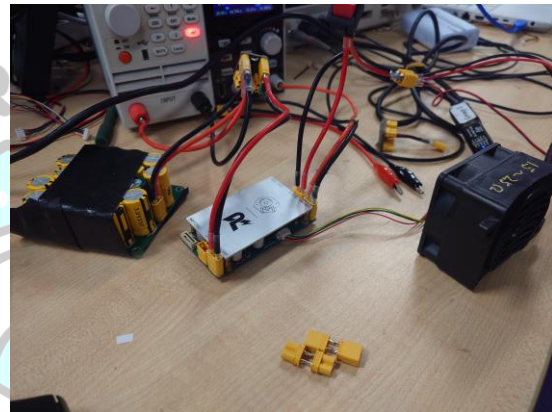
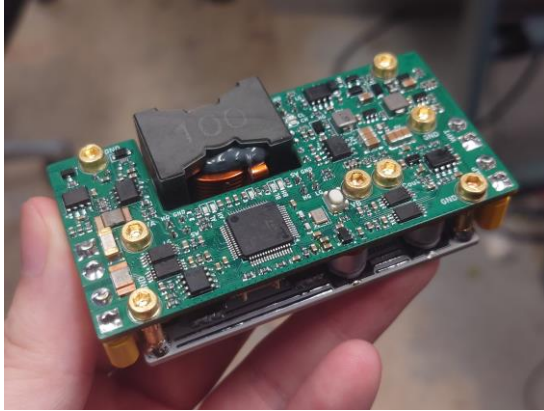
功率控制板 V1 (超级电容系统+无线充电接收端)，与发射端 V1 共同测试功率传输



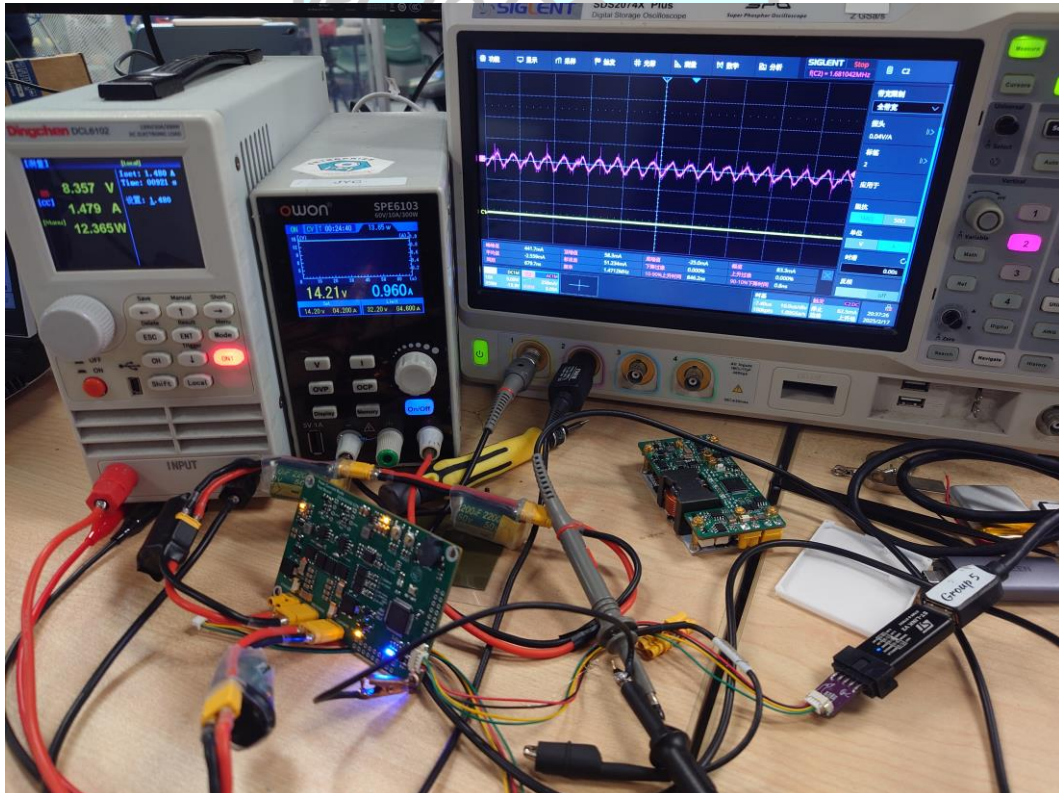
无线充电发射端 V1 与功率控制板 V1 的通讯测试



功率控制板 V2，其中超级电容部分为数控方案



功率控制板 V3，超级电容部分峰值电路控制方案的验证板



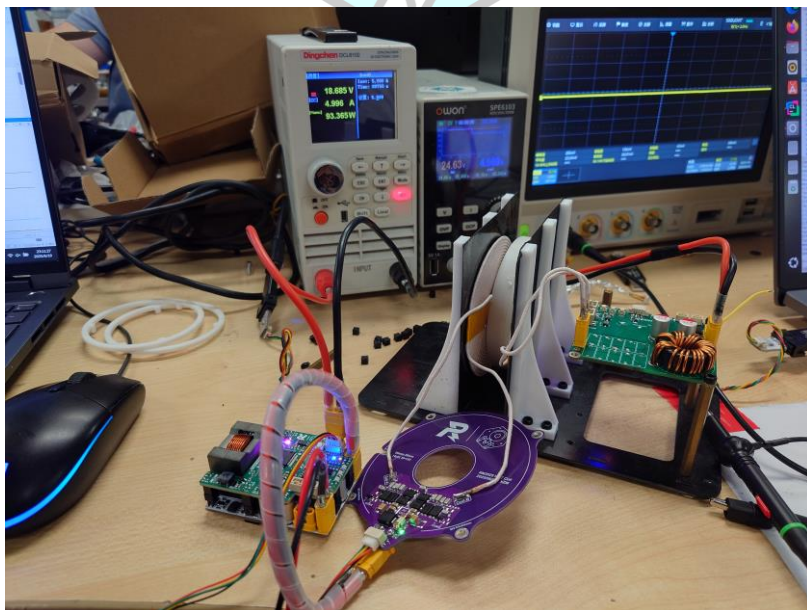
无线充电发射端 V2 与线圈板 V2



功率控制板 V4



无线充电发射端 V2 与线圈板 V2、功率控制板 V4 进行通讯和功率测试



6 未来改进方向

由于本赛季研发压力较大，同时也需要负责工程机器人、英雄机器人相关 PCB 绘制与主控板、哨兵交换机、新电池缓启动等项目的研发，整体研发时间较为紧张，所以尽管进行了多次迭代，本方案仍有许多不足。

无线充电系统稳定性尚未得到充分验证

无线充电系统在一月末左右完成初代研发，但是因为综合考虑当时的规则，认为其收益很小而放弃研发。在 5 月末青工会获知无线充电会加强后又重启研发，至六月初完成第二版的全功能测试，在临近 UC 时才安装在机器人上进行测试。由于测试机会较少，所以其在机器人上运行的稳定性目前还没有得到充分验证。

无线充电系统的相关理论计算不严谨

受限于本人的理论知识水平有限，且研发时间较短，对于无线充电，尤其是谐振计算部分，本文和实际中的理论计算十分粗略，而对移相全桥相关的计算和发射端功率闭环也没有考虑线性化等问题。在未来研发中将努力学习相关知识并学习工业中的成熟设计，将无线充电的理论计算与实践过程更加严谨。

代码优化不足

由于在开始超级电容控制器和无线充电系统研发前为纯硬件，并没有大型嵌入式项目的开发经验，C++代码相关知识也仅限于课本。这在很多时候导致写出的代码运行较为低效。未来在逐步优化软件代码的同时，也会探索硬件提升运行速度的方式：例如本赛季曾经尝试过将 62.5kHz 环路的代码放入 STM32G474 的 CCMRAM 中运行；并且也有探索过使用 FMAC 进行电压环计算。在未来的开发过程中将会继续对它们的探索。

无线充电系统与超级电容系统融合有待改进

由于无线充电系统和超级电容系统同时随电容组进行充放电操作，其中的保护逻辑仍然有待改进，例如当电容功率级关闭且无线充电发射端失控时（例如充电时被击杀导致底盘断电，且发射端出现故障），会不可避免地造成保险丝熔断。

硬件方案并未进行深度优化

对于很多芯片的选型，例如 MOS 管、栅极驱动等，本赛季主要都沿用了上赛季电容控制器的相似选型以求稳定，并节约开发时间。而对于一些参数的调整，例如功率 MOS 管的栅极电阻大小、上升时间和死区等也未进行大量测试和精细调整。

无线充电接口短路风险

无线充电接口通过 Buck 电路向电容组接口输出。Buck 上管的体二极管会导致没有无线充电的情况下电容组的电压灌入无线充电接口，当电容电压较快波动时（例如未连接电容组）上电，体二极管会通过较大瞬时电流，多次后会损坏 MOS 管（表现为无法开启）。并且由于无线充电接口需要延伸至线圈板，功率走线较长且比较靠外，增加了额外的短路风险，在今后考虑通过加入背对背 MOS 等方式进行更充分的保护。



The Hong Kong University of
Science and Technology

7 参考资料

- [1] STM32 AN5306: Operational Amplifier (OPAMP) usage in STM32G4 Series
- [2] STM32 RM0440: STM32G4 series advanced Arm®-based 32-bit MCUs
- [3] STM32 AN5497: Introduction to the buck current mode with the B-G474E-DPOW1 Discovery kit
- [4] STM32 AN4539: HRTIM cookbook
- [5] Qi Specification: Communications Physical Layer, Version 2.0 April 2023
- [6] NXP AN4701: Demodulating Communication Signals of Qi-Compliant Low-Power Wireless Charger Using MC56F8006 DSC
- [7] 【RM2024 超级电容控制器硬件+软件完全开源】 (史上最详细的开源!)香港科技大学 ENTERPRIZE 战队 <https://bbs.robomaster.com/article/54127?source=4>
- [8] 【RM2023-数控超级电容方案开源】香港科技大学-ENTERPRIZE 战队 <https://bbs.robomaster.com/article/9452?source=4>
- [9] STM32G474xB STM32G474xC STM32G474xE Datasheet

8 致谢

感谢 ENTERPRIZE 战队硬件组学长 JYC RY OFR 为硬件部门打下的坚实基础，并在本赛季为超级电容与无线充电系统的设计调试中提供了巨大帮助

感谢 ENTERPRIZE 战队硬件组队员 ZZY 承担了大部分构建、调试发射端代码的工作

感谢吉林大学、中国石油大学（华东）、华南理工大学在交流中分享的宝贵经验



The Hong Kong University of
Science and Technology