

FAQ of flipflop.fun

A. About

A1. flipflop.fun和pump.fun的区别是什么？

FFF（Flipflop.fun）和 pump.fun 都是去中心化代币发行平台，但它们在机制、设计目标和功能上存在显著差异。以下是两者之间的核心区别：

1. 铸造机制

- **pump.fun**：用户可以快速创建代币并在平台上交易，代币价格由Bonding Curve（**绑定曲线模型**）决定，随着购买量增加，价格逐步上升。
- **FFF**：采用 **Proof of Mint (PoM)** 机制，铸造代币的数量和难度通过动态调整来控制，类似于比特币的挖矿难度调整，价格随着铸造速度加快而上升，其目的旨在确保公平性并防止 Sybil 攻击。

2. 难度调整

- **pump.fun**：没有难度调整机制，代币的铸造和交易主要由市场需求驱动。
- **FFF**：通过 **Milestone** 和 **Checkpoint** 的时间划分，动态调整铸造难度，确保铸造过程的稳定性，避免快速铸造导致的不公平分配。

3. 流动性管理

- **pump.fun**：当代币达到一定市值时，平台会自动将流动性一次性添加到去中心化交易所（如 Raydium）。
- **FFF**：当完成目标Milestone时（默认为铸造完成代币总量的25%时），发币人（Issuer）或发币人委托的市值管理者（Value manager）可根据市场情况有计划的将铸造的固定费用有计划注入去中心化交易所（Raydium）。
 - 在未完成目标Milestone时，铸造费用冻结，只用于可能发生的refund。
 - 铸造费用的钱包（Token Vault）中的资金只能用于流动性管理，任何人无法提现。
 - 目标Milestone完成之后的所有铸造获的费用也进入Token Vault，用于对去中心化交易所中的流动池提供持续性支持。

4. 防止 Sybil 攻击

- **pump.fun**：没有特别的机制来防止 Sybil 攻击，容易受到恶意账户的操控。
- **FFF**：PoM 机制通过增加快速铸造的成本（减少铸造数量）来抑制 Sybil 和Bot攻击，确保代币分配的公平性。

5. 社区共识建立

- **pump.fun**：代币发行和交易速度很快。没有用于建立社区共识的时间，只有Bump and Dump。
- **FFF**：发币人可以事先设定铸造的时间节奏，PoM机制保障该铸造的时间节奏的有效实施，为社区形成共识提供充足的时间。

6. 参与者保障

- **pump.fun** : 没有任何保障, 参与者如赌博。
- **FFF** : 在未完成目标Milestone时, 参与者可随时Refund, 获得本金, 零风险。

7. 费用结构

- **pump.fun** : 创建代币免费, 铸造根据Bonding Curve支付铸造费, 交易需要支付交易手续费, 毕业需要缴纳费用。
- **FFF** : 创建代币免费, 铸造根据PoM机制支付铸造费, 交易免费, 毕业即Refund需要缴纳费用。

8. 代币分配

- **pump.fun** : 代币分配主要由用户购买行为决定, 早期购买者可能获得更多利益。
- **FFF** : 通过 PoM 机制动态调整代币分配, 旨在实现更公平的分配, 减少早期集中铸造的影响。

9. 技术实现

- **pump.fun** : 基于 Solana 区块链, 使用bonding curve绑定曲线模型实现快速代币创建和交易。
- **FFF** : 基于 PoM 机制, 可在以太坊或 Solana 等区块链上实现, 具体取决于部署选择。

10. 目标用户

- **pump.fun** : 主要面向 meme 代币爱好者和希望快速创建代币的用户。
- **FFF** : 平台提供两种发币参数: meme币参数和标准参数。同时面向meme代币爱好者和希望通过公平铸造机制发行代币的项目方和社区。

11. 风险管理

- **pump.fun** : 存在 rug pull (项目方跑路) 等风险, 尽管平台设计试图减少这些问题。
- **FFF** : 通过 PoM 机制和流动性管理, 减少市场操纵和欺诈行为, 提升市场稳定性。

12. 社区驱动

- **pump.fun** : 强调社区参与和 meme 文化, 鼓励快速交易和市场炒作, 事实上造成大量Bot作弊和Sybil攻击, 导致真正的玩家损失巨大, 极不公平。
- **FFF**

13. 代币价格确定

- **pump.fun** : 通过绑定曲线模型, 价格随需求变化, 早期购买者可能以较低价格进入。
- **FFF** : 代币价格由铸造的速度决定, 如果按设定的铸造时间进度计划进行铸造, 晚期参与者铸造成本和早期参与者一样。

14. 代币经济学

- **pump.fun** : 代币经济学较为简单, 主要依赖绑定曲线模型驱动。
- **FFF** : 代币经济学通过 PoM 机制设计, 旨在实现公平分配和市场稳定性, 减少投机性行为。

总结

FFF (Flipflop.fun) 与 pump.fun 在代币发行和交易机制上有着本质区别。FFF 通过 **Proof of Mint (PoM)** 机制引入难度调整和流动性管理，旨在解决铸造 (Fair Mint) 中的公平性缺失、Sybil 攻击、共识建立时间不足、欺诈和市场价格管理等问题。pump.fun 更专注于快速代币创建和交易，适合 meme 代币和短期投机赌博场景。两者在设计理念上有着不同追求。

A2. Flipflop.fun 和铭文 (Inscription) 的区别是什么？

铭文 (Inscription) 常用于创建独特的数字资产或存储需要不可篡改和可验证的信息。例如，在比特币网络中，通过“Ordinals”协议，铭文可以将数据嵌入交易输出中，从而在比特币区块链上创建可追踪的独特资产。而 Flipflop.fun 则是一个专注于代币铸造和分发的特定平台。以下是两者之间的主要区别：

1. 目的

- **Flipflop.fun :**
一个专为代币铸造和分发设计的平台，旨在通过 PoM (Proof of Mint) 机制确保公平性与稳定性，优化代币发行过程。
- **铭文 :**
一个更广泛的区块链概念，用于记录各种类型的数据，不局限于代币发行，可能用于 NFT 创建、数据存储等多种用途。

2. 功能

- **Flipflop.fun :**
通过动态难度调整 (PoM) 控制代币铸造过程，能够防止恶意操作 (如 Bot 滥用) 并优化代币的分发效率。
- **铭文 :**
仅仅是将数据写入区块链的基础过程，没有内置的难度调整或代币发行管理机制，因此容易受到大量 Bot 恶意操作和 Sybil 攻击的影响。

3. 范围

- **Flipflop.fun :**
是一个构建在区块链技术之上的工具或服务，专注于代币发行的特定用例，应用范围相对较窄但更专注。
- **铭文 :**
是区块链技术本身的基础操作，适用范围更广，可用于多种场景，不仅仅局限于代币铸造。

4. 流动性

- **Flipflop.fun :**
用户在铸造代币时支付的费用会进入去中心化交易所的流动池，为生态系统参与者提供流动性支持。
- **铭文 :**
铸造铭文时需要支付 Gas 费，但这些费用全部由矿工收取，未对参与者或社区赋能。

总之，Flipflop.fun 和铭文 (Inscription) 在**目的、功能、范围和流动性管理**上存在显著差异。Flipflop.fun 通过 PoM 机制实现公平的代币发行并支持流动池赋能社区，而铭文作为一个基础的区块链数据记录方式，功能更通用但缺乏动态调整和参与者激励机制。

A3. Flipflop.fun 解决了至今为止区块链 Token 发行平台的哪些问题？

Flipflop.fun（简称 FFF）作为一个创新的去中心化代币发行平台，针对传统区块链 Token 发行平台（包括类似 pump.fun 这样的平台）中存在的多种问题，提出了独特的解决方案。以下是 FFF 解决的主要问题及其具体机制：

1. Sybil 攻击和 Bot 作弊

- **问题：**传统平台容易受到 Sybil 攻击（通过创建大量虚假账户）和 Bot 作弊的干扰，导致普通用户无法公平参与代币发行。
- **解决方案：**FFF 引入 **Proof of Mint (PoM)** 机制，通过动态调整铸造难度，快速铸造的成本会显著上升（铸造数量减少），从而有效抑制恶意用户和机器人的操纵，确保公平性。

2. 缺乏社区共识建立时间

- **问题：**许多平台（如 pump.fun）代币发行速度过快，社区没有足够时间形成共识，导致价格剧烈波动和投机行为盛行。
- **解决方案：**FFF 通过 **Milestone 和 Checkpoint** 的分阶段时间设计，延长了代币铸造过程。这种设计为社区提供了充分的时间来建立共识，减少短期投机并增强市场稳定性。

3. 参与者风险高

- **问题：**传统平台通常不提供退款机制，参与者可能因项目失败或市场崩盘而损失全部资金，风险极高。
- **解决方案：**FFF 的 **Refund 机制** 允许参与者在目标 Milestone 结束前随时退款，确保本金安全，大幅降低参与者的财务风险。

4. 流动性管理不善

- **问题：**在许多平台上，流动性池的资金不足或管理不当，导致代币上线后市场价值难以维持。
- **解决方案：**FFF 将铸造费用直接注入流动池，发币人和专业的市值管理者可以根据市场情况，更有效的利用项目资金，避免一次性流动性注入完毕后，无后续资金的问题，可为市场提供持续的流动性支持，优化市场价值管理。

5. 市场操纵和欺诈行为

- **问题：**恶意用户或机器人通过快速铸造和交易操纵市场，项目方也可能通过不透明的操作实施欺诈。
- **解决方案：**PoM 机制增加快速铸造的难度和成本，抑制市场操纵；同时，FFF 的透明机制和流动性管理减少了欺诈行为的空间。

6. 代币分配不公平

- **问题：**传统平台（如 pump.fun）中，早期参与者可以低成本获取大量代币，后期参与者处于明显劣势，分配不均。
- **解决方案：**FFF 的 PoM 机制动态调整铸造难度，使快速集中铸造的成本上升，而如果按时间节奏铸造，则可确保早期和后期参与者的代币分配更加均衡，成本一致，更有利于共建中长期共识，避免 Bump and dump。

7. 费用结构不合理

- **问题：**许多平台的费用结构不透明或过于昂贵，如pumpfun的水涨船高的铸币费用和过高的交易手续费，影响用户体验和参与意愿。
- **解决方案：**FFF 提供透明且合理的费用结构：创建代币免费，铸造需支付铸造费用，交易免费。这种设计降低了参与门槛，同时保持平台运营的可持续性。

8. 缺乏长期激励

- **问题：**传统代币平台往往聚焦短期投机，忽视社区的长期建设和参与者的持续激励。
- **解决方案：**FFF 通过引导铸造时间、提供退款保障以及持续注入流动性，鼓励社区成员长期参与，促进生态系统的健康发展。

PoM针对绝大多数Token发行平台遇到的 **Sybil 攻击和 Bot 作弊、缺乏社区共识、参与者风险高、流动性不足、市场操纵与欺诈、代币分配不公、费用结构不合理** 以及 **缺乏长期激励** 等问题，构建了一个更公平、安全且可持续的代币发行环境。相比类似 pump.fun 的平台，FFF 在公平性、风险管理和社区驱动方面展现了显著优势。

A4. flipflop.fun平台上发行的代币周期分为几个阶段？

Flipflop.fun（FFF）平台上发行的代币周期主要分为三个阶段：**启动期（Launch）、共识期（Mint/Refund）和增长期（Graduated/Rebase）**。以下是对每个阶段的详细说明：

1. 启动期（Launch）

这是代币发行的初始阶段。发币人（Issuer）在 FFF 平台上创建代币。

- **特点：**
 - 操作简单，只需要半分钟即可发行一个Solana的SPL代币
 - 发币人可以选择不同的发币参数，平台提供 meme 币参数组合或标准参数组合，只需选择一个就行，无需了解复杂的PoM原理和参数意义。
 - 发币人可以设定铸造的时间节奏，如平均每分钟的铸造次数。

2. 共识期（Mint/Refund）

在此阶段，参与者开始铸造代币。平台采用 **Proof of Mint (PoM)** 机制，根据铸造速度动态调整难度，以确保公平性和稳定性。

- **特点：**
 - 在目标 Milestone 结束前，参与者可随时选择退款，保障本金安全。
 - 通过 PoM机制，铸造将按发币人设定的进度推进，从而为社区提供充足时间形成共识，减少短期投机行为。

3. 增长期（Graduated/Rebase）

当代币铸造达到目标 Milestone（如完成 25% 的代币总量）时，平台进入增长期，代币正式进入市场交易阶段。

- **特点：**
 - **毕业（Graduated）**：目标 Milestone 完成后，代币“毕业”，平台将铸造费用注入流动池（如 Raydium），初始化流动性，代币可在去中心化交易所交易。

- **重置 (Rebase)** : 此阶段中, 代币铸造还在进行, 去中心化交易所的交易也在进行, 两者之间产生一种动态平衡, 因为铸造成本只增不减的机制, 使得市场价格稳步上涨。
- 此阶段重点在于流动性管理和社区扩展, 标志着代币从发行的一级市场向市场交易的二级市场过渡, 而之前共识期时建立的社区, 为增长提供了保障。

A5. flipflop.fun平台上发行的代币总量是如何计算的？

Flipflop.fun (简称 FFF) 平台上发行的代币总量 (**Total Supply**) 是通过一个基于 **Milestone** (里程碑)、**Checkpoint** (检查点) 和 **Reduction Factor** (衰减系数) 的机制计算的。这种设计类似于比特币的挖矿奖励递减模式, 通过多个 Checkpoint 分阶段释放代币, 并在每个 Milestone 中根据缩减系数逐步减少铸造量。

公式1: 代币的总量通过以下公式计算：

$$\text{Total Supply} = C \cdot T_0 \cdot \frac{1 - f^E}{1 - f}$$

参数说明：

- C : 每个 Milestone 中的 Checkpoint 数量。
- T_0 : 第一个 Milestone 中每个 Checkpoint 的目标铸造量。
- f : 缩减系数 (Reduction Factor), 通常 ($f < 1$), 表示每个 Milestone 铸造量的缩减比例。
- E : 总的 Milestone 数量。

这个公式是一个等比数列的求和公式, 反映了代币铸造量随 Milestone 递增而逐步减少的特性。

公式2: 最大供应量 (Max Supply) 的计算方法

如果 Milestone 数量 (E) 趋向无穷大, 代币总量会收敛到一个理论上的最大值, 即 **Max Supply**。其公式为：

$$\text{Max Supply} = \frac{C \cdot T_0}{1 - f}$$

这表示在无限 Milestone 的情况下, 代币总量的上限。

举例1

flipflop.fun平台上一个标准代币的发行参数如下：

- 每个 Milestone 中的 Checkpoint 数量 ($C = 250$)
- 第一个 Milestone 中每个 Checkpoint 的目标铸造量 ($T_0 = 200,000$) 代币
- 缩减系数 ($f = 0.5$) (即每个 Milestone 铸造量减少 50%)

使用 Max Supply 公式：
$$\text{Max Supply} = \frac{C \cdot T_0}{1 - f} = \frac{250 \cdot 200,000}{1 - 0.50} = \frac{50,000,000}{0.50} = 100,000,000 \text{ 代币}$$

这意味着即使时间无限增加, 代币总量也不会超过这个值。这种机制确保了代币发行的可控性和经济学上的可持续性。

举例2

flipflop.fun平台上一个meme代币的发行参数如下：

- 每个 Milestone 中的 Checkpoint 数量 ($C = 250$)
- 第一个 Milestone 中每个 Checkpoint 的目标铸造量 ($T_0 = 1,000,000$) 代币
- 缩减系数 ($f = 0.75$) (即每个 Milestone 铸造量减少 25%)

使用 Max Supply 公式 : $\text{Max Supply} = \frac{C \cdot T_0}{1 - f} = \frac{250 \cdot 1,000,000}{1 - 0.75} = \frac{250,000,000}{0.25} = 1,000,000,000$ 代币

总结 : 在flipflop.fun平台上, 一个meme代币的最大供应量为 1,000,000,000(1Billion) 代币, 一个标准代币的最大供应量为 100,000,000(100Million) 代币。

A6.什么是milestone, checkpoint, 以及target milestone ?

在Flipflop.fun的代币铸造机制中, **Milestone (里程碑)**、**Checkpoint (检查点)** 和 **Target Milestone (目标里程碑)** 是核心的时间划分和调控单元, 用于实现动态铸造难度调整和公平分配。以下是具体说明 :

1. Milestone (里程碑)

- **定义**

Milestone是代币铸造周期中的最高级时间单位, 每个Milestone包含固定数量的**Checkpoint**, 并按照**缩减系数 (Reduction Factor)** 逐步减少铸造总量。

- **关键属性**

- **Checkpoint数量 :** 每个Milestone包含的Checkpoint数量固定 (例如250个) 。
- **铸造量缩减 :** 每个新Milestone的**目标铸造量**和**基础铸造量**会按缩减系数 (如0.50) 减少, 实现类似比特币的减半机制。
- **难度独立调整 :** 每个Checkpoint的铸造难度根据实际铸造速度动态调整, 但Milestone之间铸造量的缩减是固定预设的。

- **示例**

假设初始Milestone的目标铸造量为1,000,000代币, 缩减系数为0.75 :

- Milestone 1: 1,000,000 代币
- Milestone 2: 750,000 代币 ($1,000,000 \times 0.75$)
- Milestone 3: 562,500 代币 ($750,000 \times 0.75$)

2. Checkpoint (检查点)

- **定义**

Checkpoint是Milestone内的基本时间单元, 每个Checkpoint对应一个**动态难度调整周期**, 用于根据实际铸造速度调整铸造成本和代币分配。

- **关键机制**

- **难度调整 :** 若Checkpoint的实际完成时间短于目标时间, 则增加难度系数, 减少单次铸造量, 提高代币成本 ; 反之则保持难度不变。
- **铸造量计算 :** 单次铸造量 = 基础铸造量 / 当前难度系数
- **时间控制 :** 目标Checkpoint时间通常设置为固定值 (如10分钟) , 实际时间由市场参与度决定。

- **示例**

- 目标Checkpoint时间：10分钟
- 实际完成时间：8分钟 → 难度系数增加，单次铸造量减少
- 实际完成时间：12分钟 → 难度系数不变

3. Target Milestone（目标里程碑）

- 定义
Target Milestone是代币发行的关键阶段目标（默认为完成首个Milestone，如果是meme版的代币，即铸造总量的25%，如果是标准版的代币，则为铸造总量的50%，因为两这个衰减系数不同）。

Target Milestone的完成标志着代币从铸造期（一级市场）进入流通期（二级市场）。
- 核心作用
 1. **流动性解锁**：完成时，铸造费用将注入去中心化交易所（如Raydium），建立初始流动性池。
 2. **风险控制**：在目标Milestone完成前，参与者可随时**全额退款（Refund）**，确保本金零风险。
 3. **共识形成**：为目标Milestone预留充足时间（如数天），防止短期投机，促进社区共识建立。
- 规则
 - 未完成时：铸造费用冻结，仅用于潜在退款。
 - 完成后：后续铸造费用持续注入流动性池，支持长期市场稳定。

总结对比

概念	作用	调控对象	时间跨度示例
Milestone	分阶段减少铸造总量	代币经济学模型	数天至数周
Checkpoint	动态调整单次铸造成本和数量	铸造速度与公平性	10分钟至数小时
Target Milestone	划分铸造期与流通期，控制流动性释放	市场风险与参与者保障	首个Milestone完成

A7. 什么是fomo系数（或难度系数）？

A7. 什么是FOMO系数（难度系数）？

FOMO系数（即**难度系数**）是Flipflop.fun平台用于动态调节代币铸造成本和数量的核心参数，其设计灵感源自比特币的挖矿难度机制。该系数通过实时反馈市场参与热度，抑制快速铸造行为，确保代币分配的公平性。

1. 定义与作用

- 定义
FOMO系数（**Fear Of Missing Out Coefficient**）是一个动态数值，反映当前铸造阶段的竞争强度。
 - 初始值为 **1**，随铸造速度变化自动调整。
 - **系数升高** → 单次铸造获得的代币减少 → 代币成本上升。
 - **系数保持** → 代币成本稳定。
- 核心作用

- **抑制投机**：通过提高快速铸造的成本，防止Bot和Sybil攻击者垄断早期代币。
- **公平锚定**：确保按设定节奏铸造时，早期与后期参与者的单位代币成本一致。
- **市场平衡**：高系数时参与者可能放缓铸造，低系数时吸引新参与者，形成动态平衡。

2. 调整机制

FOMO系数在每个**Checkpoint（检查点）**结束时更新，规则如下：

条件	调整公式	效果说明
Checkpoint实际时间 < 目标时间	新系数 = 旧系数 × (1 + (1 - 实际时间/目标时间)/100)	铸造过快 → 系数上升，抑制速度
Checkpoint实际时间 ≥ 目标时间	新系数 = 旧系数	铸造达标 → 系数维持稳定

公式示例

- 目标Checkpoint时间：600秒（10分钟）
- 实际完成时间：400秒
- 调整幅度： $(1 - 400/600)/100 = 0.00333$
- 新系数： $旧系数 \times 1.00333$

3. 对参与者的影响

- **成本透明**
单次铸造量 = 基础铸造量 / FOMO系数
(例：基础1000代币，系数1.2 → 实际获得833代币)
- **风险对冲**
 - **Target Milestone前**：可随时按**原始成本退款**，避免高系数阶段的潜在亏损。
 - **Target Milestone后**：系数持续升高推高代币锚定价格，支撑二级市场价格。

• 博弈策略

市场行为	FOMO系数变化	参与者策略建议
大量Bot快速铸造	急剧上升	暂停铸造，等待社区共识形成
按目标节奏平稳铸造	缓慢上升	持续参与，均衡获取代币
铸造冷清，时间超目标	维持不变	低成本介入机会

4. 案例

铸造参数：

- 基础铸造量（\$M_0\$）：100代币/次
- Checkpoint目标时间：600秒（10分钟）
- 初始FOMO系数：1.0

Checkpoint	实际完成时间 (秒)	FOMO系数变化	当前FOMO系数	实际获得代币	代币成本 (ETH)
9	300	+0.5%	1.015086	98.51	0.001015
10	200	+0.6667%	1.021854	97.86	0.001022
16	200	+0.6667%	1.028666	97.21	0.001029

关键数据解读

1. 系数递增性

- 快速铸造（Checkpoint 9-10）导致系数连续跳升，代币成本从0.001015 ETH升至0.001022 ETH。
- 持续加速（Checkpoint 16）使成本进一步增至0.001029 ETH，涨幅达2.9%。

2. 抑制效果量化

行为	代币获取量下降	成本上升
3倍速铸造（200秒/Checkpoint）	2.79%	+2.9%
2倍速铸造（300秒/Checkpoint）	1.49%	+1.5%

设计意图

通过该案例可清晰看出：

- 非线性惩罚**：速度越快，成本增幅越显著（200秒铸造的惩罚力度是300秒的1.9倍）
- 动态锚定**：按目标节奏（600秒/Checkpoint）铸造时，代币成本保持恒定。

总结

FOMO系数通过**动态成本锚定**和**反超发机制**，将市场热度转化为可量化的调控参数，既保留Fair Mint的开放性，又通过算法实现去中心化公平。其设计本质是**让速度与公平达成纳什均衡**，而非简单限制参与者行为。

A8. 每次铸造的代币成本多少？怎么计算？

在Flipflop.fun平台中，代币铸造成本通过**动态难度机制（FOMO系数）**和**阶段衰减模型**共同决定，计算公式透明且可验证。以下是详细计算规则：

1. 核心公式

单次铸造成本 = 固定铸造费 / 实际获得代币数量

具体展开为：

$$p = \frac{P_0}{M} = \frac{P_0 \cdot d}{M_0 \cdot f^{(m-1)}}$$

参数说明

符号	定义	示例值
\$p\$	单位代币成本（ETH/代币）	0.001022 ETH
\$P_0\$	单次铸造固定费用（ETH）	0.1 ETH
\$d\$	当前FOMO系数（难度系数）	1.021854
\$M_0\$	初始基础铸造量（代币/次）	100 代币
\$f\$	缩减系数（按Milestone衰减）	0.75
\$m\$	当前Milestone编号	1（首个阶段）

2. 计算步骤

1. 确定基础铸造量

当前Milestone的基础铸造量 = ($M_0 \cdot f^{\{m-1\}}$)
(例如：第2个Milestone时, ($M_0=100, f=0.75 \rightarrow 100 \times 0.75=75$)代币)

2. 计算实际获得代币

实际获得代币 = 基础铸造量 / FOMO系数
(例如：基础75代币, 系数1.2 $\rightarrow 75/1.2=62.5$ 代币)

3. 计算单位成本

成本 = 固定铸造费 / 实际获得代币
(例如：0.1 ETH /62.5=0.0016 ETH/代币)

3. 动态影响因素

因素	对成本的影响	调控逻辑
FOMO系数升高 (d↑)	成本线性增加	抑制快速铸造
Milestone推进 (m↑)	成本指数增加 (因基础量衰减)	实现代币稀缺性
铸造速度达标	成本保持稳定	维持预设经济模型

4. 实际案例

- 参数
($P_0=0.1$) ETH, ($M_0=100$), ($f=1$)（首Milestone不衰减）, ($d=1.021854$)
- 计算过程
 - 基础铸造量 = ($100 \times 1^{\{1-1\}}=100$)
 - 实际获得代币 = ($100 / 1.021854 \approx 97.86$)
 - 单位成本 = ($0.1 / 97.86 \approx 0.001022$) ETH

5. 成本控制机制

- 风险对冲

- **Target Milestone前**：支持全额退款，锁定成本为历史最低值。
- **铸造进度查询**：平台实时公开当前FOMO系数和基础铸造量。

• 博弈平衡

市场行为	成本变化	推荐策略
Bot抢跑导致d飙升	短期成本骤增	暂停铸造，等待系数稳定
社区按目标节奏铸造	成本按预设曲线缓步上升	定期参与，均摊成本

总结

代币成本由**固定费用**、**FOMO系数**和**Milestone阶段**三重变量动态决定，通过链上公式实时计算。此设计将市场博弈转化为可预测的数学模型，既保留公平性，又通过算法抑制投机行为。

A9. 什么是毕业？

要了解毕业，首先需要了解flipflop.fun平台上代币的生命周期，为了便于理解，将代币发行比作一次太空火箭发射任务，"毕业"如同火箭成功脱离地球大气层并进入预定轨道——这是从"地面控制"到"自主运行"的转折点。具体对应关系如下：

1. 发射阶段 vs 代币阶段

火箭发射阶段	代币阶段	核心规则
发射台准备	启动期 (Launch)	工程师设定参数（发币人配置代币），燃料加注（初始流动性准备）
助推器燃烧	共识期 (Mint/Refund)	火箭升空（铸造开始），可紧急中止（退款），实时调整推力（FOMO系数动态调控）
轨道注入	毕业 (Graduated)	脱离助推器（达到Target Milestone），展开太阳能板（流动性注入），进入自主运行（自由交易市场）

2. "毕业"的核心意义

- **动力系统切换**
火箭抛弃助推器（关闭退款通道），转而依赖轨道引擎（流动性池）维持运行，如同代币从铸造依赖转为市场供需驱动。
- **任务模式升级**
 - 毕业前：地面控制中心（项目方）主导所有修正（动态调整参数）；
 - 毕业后：进入自动驾驶模式（社区自治），仅靠惯性导航（市场机制）维持。
- **风险性质转变**

阶段	风险类型	应对措施
----	------	------

阶段	风险类型	应对措施
升空阶段	技术故障（铸造失衡）	紧急中止（退款）
轨道运行	宇宙辐射（市场波动）	自主防护（社区治理）

3. 为何需要"毕业"机制？

- **资源优化**
助推器（早期铸造费）的燃料耗尽后必须分离，避免 dead weight（无效资金滞留），将资源集中到核心引擎（流动性池）。
- **使命验证**
只有进入轨道的火箭（代币）才证明其具备宇宙探索能力（市场价值），未达标者将坠入大气层销毁（项目终止）。
- **去中心化过渡**
如同NASA移交控制权给国际空间站，毕业标志着从中心化调控（项目方主导）到去中心化自治（持币者治理）的转变。

4. 实际案例

以SpaceX火箭发射流程为例：

- **Target Milestone**：二级火箭分离（类比铸造25%总量）
- **毕业动作**：释放卫星组（流动性注入），开启通信模块（交易所上市）
- **失败后果**：火箭坠入海洋（代币永不流通）

阶段	航天器状态	代币经济映射
一级火箭分离	抛弃初始推进器	完成首阶段铸造，部分退款
二级火箭入轨	卫星组部署	流动性注入，开盘价锚定
太阳帆板展开	能源系统自给	市场交易活跃，价格发现

总结

"毕业"是代币经济中的"轨道注入机动"——它通过**流动性释放+退款通道关闭**，像火箭进入预定轨道一样，标志着项目从"试验阶段"正式升级为"可持续运行实体"。唯有通过这一关卡的代币，才有资格在加密货币的宇宙中开启星辰大海的征程。

A10. 为什么铸造成本会只升不降？

Flipflop.fun的铸造成本呈现**单向增长趋势**，这是由**FOMO系数**和**Milestone衰减机制**的双重叠加效应决定的。以下通过核心公式分阶段解析其必然性：

1. 价格公式与增长因子

代币铸造价格的完整公式为：

$$p = \frac{P_0 \cdot d}{M_0 \cdot f^{(m-1)}}$$

变量解释

变量	定义	变动方向	对价格影响
d	FOMO系数（难度系数）	只增不减	线性增长
f	Milestone衰减系数（ $0 < f < 1$ ）	固定衰减	指数增长
m	当前Milestone编号	持续递增	指数增长

双重增长机制

- **FOMO系数单向性**： d 仅在铸造速度**快于目标**时增加，而不会因速度过慢而降低。
- **Milestone衰减必然性**：每个新Milestone的基础铸造量 $M_0 \cdot f^{(m-1)}$ 必然减少（因 $f < 1$ ），导致分母缩小。

2. 全生命周期价格曲线

阶段1：目标Milestone前（ $m=1$ ）

- 公式简化为： $p = \frac{P_0 \cdot d}{M_0}$
- **增长驱动**：仅依赖 d 上升（铸造速度超过目标）
- **示例(见POM论文)**
 - $P_0=0.1$ ETH, $M_0=100$
 - 当 d 从1.0升至1.028666（16次Checkpoint加速），价格从0.001 ETH升至0.001029 ETH，涨幅2.9%

阶段2：目标Milestone后（ $m \geq 2$ ）

- 公式展开： $p = \frac{P_0 \cdot d}{M_0 \cdot \frac{1}{f^{(m-1)}}}$
- **指数增长触发**：
 - 当 $f=0.75$ 时，每进入新Milestone，分母缩小25% → 价格跳跃式上涨
 - 叠加 d 的持续上升，形成**线性×指数**复合增长

数值模拟

Milestone	$f^{(m-1)}$	FOMO系数 d	价格（ETH）	环比涨幅
1（毕业）	1.0	1.03	0.00103	-
2	0.75	1.05	0.00147	+42.7%
3	0.5625	1.08	0.00192	+30.6%

3. 不可逆性证明

定理

在以下条件下，价格 p 严格单调递增：

1. $d_{n+1} \geq d_n$ (FOMO系数非递减)
2. $f < 1$ (Milestone衰减存在)
3. m 随铸造进度递增

推导

- 对任意两个时间点 $t_1 < t_2$:
 - 若 t_2 与 t_1 在同一Milestone :

$$d_{t_2} \geq d_{t_1} \rightarrow p_{t_2} \geq p_{t_1}$$
 - 若 t_2 进入新Milestone :

$$m \uparrow \rightarrow f^{(m-1)} \downarrow \rightarrow p_{t_2} = p_{t_1} \cdot \frac{d_{t_2}}{d_{t_1}} \cdot \frac{1}{f} > p_{t_1}$$

4. 设计意图与经济学意义

1. 抗通胀模型

通过价格单向增长，使后期铸造者需支付更高成本，抑制代币超发（类似比特币减半但更陡峭）。

2. 公平性强化

早期参与者无法通过快速铸造获得成本优势，消除时间套利空间。

3. 市场稳定性

持续升高的铸造成本为二级市场价格提供**动态地板价**支撑，防止恶意做空。

4. 博弈均衡

价格增长曲线迫使参与者选择：

- **快速铸造**：产生市场FOMO效应，推高价格
- **慢速铸造**：提高公平性，让早期和晚期的参与者成本趋于均衡
- **Refund**：Refund机制让在目标Milestone前的参与者可以获得全额退款，避免高成本的风险，同时降低FOMO预期，阻止早期Bot和Sybil攻击。

总结

铸造价格的单向增长是**算法预设的经济规律**，通过FOMO系数与Milestone衰减的**双重锁死机制**，将代币发行转化为一个**时间成本严格递增的稀缺性游戏**。这既是对比特币"挖矿难度-减半"模型的升级，也为去中心化代币发行提供了抗操纵的新范式。

A11. 如何保障铸造阶段参与者的利益？

Flipflop.fun通过**多重机制设计**确保铸造阶段的公平性、安全性与收益可预期性，核心保障措施如下：

1. 零风险退款机制（Refund）

- 规则：
 - 在**目标Milestone完成前**（默认模式下为铸造总量的50%，meme模式下为总量的25%），参与者可随时**全额退回本金**（扣除5%Refund费和0-5%的推荐码费用）。
 - 退款时需确保钱包内持有**全部原始铸造代币**，若部分转移则需补足差额。
- 设计意图：

- 消除早期参与的不确定性风险，防止项目方跑路（Rug Pull）或市场突变导致损失。
- 抑制Bot通过批量账户操控市场。

- **示例：**

某代币目标Milestone为3天，若期间价格波动剧烈，30%参与者选择退款，系统自动从流动性储备金中兑付，不影响剩余70%持有者权益。

2. 动态难度调控（FOMO系数）

- **反Sybil攻击：**

- **公式：**铸造量 = 基础量 / FOMO系数，快速铸造导致系数上升，单次收益递减。
- **效果：**Bot批量操作时，其**边际成本**会快速提升，**边际收益**迅速降低，最终无利可图。

- **公平锚定：**

- 若按设定的时间节奏铸造，FOMO系数保持稳定，确保**早期与晚期参与者成本基本一致**。

3. 流动性硬锁定与计划注入

- **资金安全：**

- 铸造费用存入名叫**Token Vault**的**PDA账户**，该账户中的资金仅能用于向去中心化交易所（如Raydium）添加流动性和交易，任何人无法提现。
- 未达目标Milestone前，资金**完全冻结**，不可提取或挪用。

- **流动性策略：**

- **目标达成后：**由发币人或发币人委托的市值管理人将**Token Vault**中的代币和SOL有计划的注入流动性池，并销毁Liquidity Token，避免流动性底池枯竭。
- **持续铸造阶段：**后续铸造费用持续性注入流动池，对**流动性支撑**产生持久性支持。

4. 透明化链上验证

- **实时可查数据：**

- **FOMO系数、当前铸造量、剩余可退款额度**等参数全链上公开。
- 参与者可通过区块链浏览器（如Solscan）验证合约状态。

- **智能合约（链上程序）约束：**

- 代币分配规则、流动性释放条件均通过代码强制执行，规避人为干预风险。

5. 反操纵博弈设计

- **延迟收益惩罚：**

- 试图在早期囤积代币后集中抛售的参与者，将因后续铸造难度升高而**拉高自身成本**，或因后期参与者集体**Refund**而导致发行失败。

- **反信息不对称与社区共识：**

- 每個代幣在發行時設定了**目標鑄造時間**，强制预留社区讨论与传播时间，避免信息不对称。

总结

Flipflop.fun通过**Refund保险阀**、**FOMO系数调控**、**流动性硬锁定与全链透明**四重机制，构建了去中心化代币发行中最完善的参与者保护体系。其本质是将传统金融的**风险对冲工具**（如期权、止损）编码为区块链原生协议，使普通用户能在低风险环境中参与早期投资。

A12. Token Vault在参与流动性管理时，产生的交易盈亏有谁承担？提供流动性产生的费用收入如何分配？

1. 交易盈亏的承担方

- **完全去中心化承担：**
Token Vault注入流动性池后，且**流动性代币（LP Token）被销毁**的话，这意味着：
 - **盈亏归属全体持币者：**流动性池的资产增值/贬值直接影响代币市场价格，盈亏通过市场波动由所有持币者共同承担。
 - **无单一控制方：**销毁LP Token后，无人能单独提取流动性池资产，避免项目方或做市商操控。

2. 流动性手续费的分配机制

- **全数留存流动性池：**
Raydium等DEX的交易手续费（通常0.25%）**100%注入**Token Vault账户（即PDA账户），用于：
 - **自动做市（AMM）：**增加池深度，提升大额交易稳定性。
 - **抵抗无常损失：**通过手续费复投抵消部分波动风险。
- **设计意图：**
 - **去中心化激励：**避免手续费被项目方垄断，确保流动性增长惠及整个生态系统。
 - **长期稳定性：**手续费再投资形成**流动性护城河**，降低恶意砸盘可行性。

3. 流动性管理的核心规则

环节	规则	经济影响
流动性注入	发币人/市值管理人按预设节奏（如每铸造100万代币）从Token Vault提取资金注入	避免一次性抛压，形成阶梯式价格支撑
LP Token 销毁	销毁LP Token，可确保流动性池所有权不可被回收，但如果不销毁，Token Vault的PDA账户的特点可确保资金无法被提取，仍旧只能用于流动性管理	消除项目方跑路风险，在完全机械式的全自动销毁机制和根据市场情况专业且灵活的进行流动性管理之间寻找平衡
手续费再投资	交易手续费自动累积至流动池，不分配给任何地址	增强市场深度，使代币获得类似比特币的"自我强化"流动性特征

总结

Token Vault通过**双重保障机制**实现流动性管理的去中心化与安全性：

1. 盈亏共担：

- 若销毁LP Token，盈亏由全体持币者通过市场波动共同承担；
- 即使保留LP Token，PDA账户的特性也确保资金**仅能用于流动性管理**，无法被提取或挪用。

2. 手续费循环：

- 交易手续费100%回流至Token Vault账户，用于持续增强流动池深度，形成**自我强化**的流动性护城河。

3. 灵活风控平衡：

- 通过销毁LP Token实现完全去中心化，或依托PDA账户的不可篡改性保留专业管理空间，在**机械执行与市场适应性**间取得最优解。

这一设计使Flipflop.fun成为首个兼顾**去中心化公平性**与**专业流动性管理能力**的代币发行协议。

A13. 如何让社区成员更公平的获得代币？

Flipflop.fun通过**算法约束**和**经济激励**的双重设计，从**铸造机制**、**风险控制**和**社区治理**三方面保障公平性：

1. 抗Bot的动态铸造规则

• FOMO系数调控：

- **快速铸造惩罚**：Bot或Sybil攻击者批量操作时，铸造量随速度提升**边际递减**。
- **公平节奏锚定**：按预设时间铸造时，所有参与者的单位成本一致，消除早期优势。

• Checkpoint时间锁：

- 每个Checkpoint设置最低持续时间（如10分钟），强制间隔确保散户参与机会。

2. 去中心化风险对冲工具

• 零风险退款（Refund）：

- 目标Milestone前可**全额退款**（扣除5%费用），消除项目失败风险。
- 示例：某Meme币在3天内达成25%铸造量，30%参与者选择退出，系统自动兑付无挤兑风险。

• 流动性硬锁定：

- 铸造费用存入**不可篡改的PDA账户**，仅能用于流动性管理，杜绝资金挪用。

3. 透明化博弈设计

• 链上数据全公开：

- 实时可查**FOMO系数**、**铸造进度**、**退款池余额**，规避暗箱操作。
- 通过Solscan等区块链浏览器验证合约状态，确保规则执行不可篡改。

- **反垄断分配模型：**
 - **延迟收益惩罚：**囤积代币者面临后续铸造成本飙升，抑制集中抛售。
 - **手续费回流机制：**流动性池收益100%用于增强市场深度，惠及全体持币者。

4. 对比传统模式的公平性升级

公平维度	pump.fun	Flipflop.fun
抗Bot能力	无动态调控，Bot收益高	FOMO系数使Bot边际收益趋零
退款保障	无退款机制	目标Milestone前全额退款
流动性控制	机械式的流动性操作，无法获得持续支持，流动池很容易被抽干	专业、灵活、可持续的流动性管理机制
成本一致性	早期参与者成本低	按节奏铸造则成本一致

总结

Flipflop.fun通过**FOMO系数抑制投机、Refund机制对冲风险、全链透明消除信息差**，重新定义了去中心化代币发行的公平标准。其本质是将比特币的"工作量证明"升级为"参与量证明"（Proof of Mint），让每个社区成员无论资金规模或技术能力，都能在算法保障下公平竞争。

A14. 产量递减机制是如何实现的？

Flipflop.fun的产量递减机制通过**预设结构性衰减（Milestone缩减）和动态铸造调控（FOMO系数）**双重实现，具体规则如下：

1. 结构性递减：基于Milestone的预设衰减

- **缩减系数（Reduction Factor）**
每个新Milestone的**基础铸造量**按固定比例缩减，公式为：
$$\text{基础铸造量} = M_0 \cdot f^{(m-1)}$$
 - M_0 ：首个Milestone的基础铸造量
 - f ：缩减系数（ $0 < f < 1$ ，如0.75表示每Milestone减少25%）
 - m ：当前Milestone编号

• 示例

Milestone	基础铸造量（ $M_0=100$ ， $f=0.75$ ）	环比减少
1	100 代币	-
2	75 代币	25%
3	56.25 代币	25%

2. 动态递减：同一Milestone内的FOMO调控

- 规则：
 - 铸造速度 > 目标 → FOMO系数永久性递增 → 实际铸造量 = 基础铸造量 / FOMO系数
 - 铸造速度 ≤ 目标 → FOMO系数维持不变 → 铸造量稳定

• 示例

Checkpoint	FOMO系数	实际铸造量（基础量100）	效果
1	1.0	100 代币	基准值
10	1.021854	97.86 代币	速度过快，产量减少
16	1.028666	97.21 代币	进一步递减

• 不可逆性对比

类型	触发条件	是否可逆
Milestone递减	跨阶段预设	不可逆，只能递减
FOMO调控	单Milestone内速度	不可逆，只递增

3. 双重递减的协同效应

- 抗Bot设计：
 - Bot快速铸造触发永久升高，后续收益持续递减。
 - 即使Bot停止攻击，其历史行为仍导致该Milestone内所有参与者收益受损。
- 长期通缩模型：

Milestone缩减与FOMO递增形成指数+线性复合衰减曲线，比比特币的单纯减半模型更陡峭。

总结

产量递减通过Milestone结构性缩减（全局不可逆）与FOMO系数递增（局部不可逆）双重锁定，形成全网首个全不可逆通缩模型。这种设计既抑制短期投机，又确保代币稀缺性随时间和参与度严格递增，为公平发行树立新范式。

A15. 铸造成本是按每个Checkpoint递增，还是每次铸造都会递增？

Flipflop.fun的铸造成本递增机制以Checkpoint为调整单位，而非每次铸造单独调整。具体规则如下：

1. Checkpoint级别的成本更新

- 调整周期：
 - FOMO系数（难度系数）仅在每个Checkpoint结束时根据实际铸造速度重新计算。
 - 若本Checkpoint完成时间短于目标时间 → FOMO系数永久性递增。
 - 若完成时间等于或长于目标时间 → FOMO系数维持不变。

- **示例**（引用pom paper 2.3.1节数据）：

Checkpoint	目标时间	实际时间	FOMO系数变化	下一Checkpoint成本
1	600秒	500秒	+0.00167	0.0010167 ETH
2	600秒	400秒	+0.00333	0.0010333 ETH
3	600秒	600秒	无变化	0.0010333 ETH

2. 同一Checkpoint内成本固定

- **规则：**
每个Checkpoint内的**所有铸造操作**使用相同的FOMO系数，因此单次成本固定。
- **示例：**
 - Checkpoint 2的FOMO系数为1.0333，基础铸造量100代币：
 - 第1次铸造：100 /1.0333 ≈96.78代币 → 成本0.0010333 ETH
 - 第50次铸造：仍为96.78代币 → 成本不变

3. 与Milestone递减的叠加效应

- **全局衰减：**
每个新Milestone的**基础铸造量**按缩减系数减少（如25%），导致**跨Milestone成本跳升**。
- **示例**（参数：\$M_0=100\$, \$f=0.75\$）：

Milestone	Checkpoint	FOMO系数	实际铸造量	单位成本（ETH）
1	10	1.02	98.04	0.00102
2	1	1.02	73.53	0.00136（↑33%）

4. 对比其他模型的优势

平台	成本调整频率	公平性缺陷
pump.fun	每次铸造（Bonding Curve）	早期参与者成本极低，晚期极高
比特币	每2016区块（约2周）	调整周期过长，无法抑制短期算力波动
Flipflop	每个Checkpoint	平衡实时性与抗操纵能力

总结

铸造成本**按Checkpoint递增**而非每次铸造，这种设计：

1. **抑制高频操作**：Bot无法通过单次快速铸造立即获利，需承担整个Checkpoint的成本上升风险。
2. **降低计算复杂度**：参与者可预测当前Checkpoint内所有铸造的固定成本。
3. **增强公平性**：社区有明确的时间窗口（Checkpoint周期）调整参与策略。

A16. 每个Checkpoint中的铸造次数是一定的吗？

不固定。每个Checkpoint的铸造次数由**目标铸造量（T）和动态单次铸造量（M）**共同决定，实际次数需满足总铸造量不超过目标值。以下是关键机制和案例解析：

1. 核心计算规则

- **单次铸造量：**
 $M = \frac{M_b}{d}$
 - M_b ：基础铸造量（按Milestone衰减）
 - d ：当前FOMO系数
- **铸造次数：**
 $n = \left\lfloor \frac{T}{M} \right\rfloor$
（向下取整，确保总铸造量 $n \cdot M \leq T$ ）
- **剩余未完成量：**
 $\text{剩余量} = T - n \cdot M$

2. 用户案例解析

参数	值
目标铸造量（T）	13,000 代币
基础铸造量（M_b）	1,000 代币
FOMO系数（d）	1.33

- **计算过程：**
 1. 单次铸造量： $M = 1,000 / 1.33 \approx 751.8797$
 2. 最大次数： $n = \left\lfloor 13,000 / 751.8797 \right\rfloor = 17$ 次
 3. 总铸造量： $17 \times 751.8797 \approx 12,781.95$
 4. 剩余量： $13,000 - 12,781.95 = 218.05$

结论：

- 该Checkpoint内铸造次数为17次（非固定值）
- 剩余量会计入后续Checkpoint或通过难度调整补偿

3. 扩展案例

案例1：FOMO系数升高（抑制快速铸造）

参数	值
目标铸造量（T）	13,000 代币
基础铸造量（M_b）	1,000 代币
FOMO系数（d）	1.5（因铸造速度过快升高）

- 计算：
 1. $M = 1,000 / 1.5 \approx 666.6667$
 2. $n = \lfloor 13,000 / 666.6667 \rfloor = 19$ 次
 3. 总铸造量： $19 \times 666.6667 \approx 12,666.67$
 4. 剩余量： $13,000 - 12,666.67 = 333.33$

影响：

- 铸造次数从17次→19次，但单次量减少，总铸造量仍不足目标
- 更高的FOMO系数导致后续Checkpoint成本进一步上升

案例2：FOMO系数不变（匀速铸造）

参数	值
目标铸造量（T）	13,000 代币
基础铸造量（M_b）	1,000 代币
FOMO系数（d）	1.0（铸造速度达标）

- 计算：
 1. $M = 1,000 / 1.0 = 1,000$
 2. $n = \lfloor 13,000 / 1,000 \rfloor = 13$ 次
 3. 总铸造量： $13 \times 1,000 = 13,000$
 4. 剩余量： 0

影响：

- 按目标节奏铸造时，次数最少且无剩余量，成本稳定

4. 剩余量处理机制

定期销毁，过高的FOMO系数可能导致大量的未铸造剩余，即产生通缩，从而降低FOMO带来的副作用。

5. 设计总结

特性	说明
动态次数	次数由d和M_b动态计算，非固定值
硬顶约束	总铸造量永不超过T，剩余量定期集中销毁
抗Bot设计	Bot加速铸造导致d↑→M↓→n↑，但总收益受T限制且成本飙升

结论

每个Checkpoint的铸造次数**不固定**，而是通过**目标量约束**和**FOMO系数调控**动态计算。这种设计既保证了代币分发的精确通缩，又通过数学规则将操纵空间压缩到极限，成为首个实现"算法公平发行"的机制。

A17. 达到目标Milestone后，铸造成本和市场价格的Rebase机制是如何防止死亡螺旋的？

Flipflop.fun通过**铸造成本锚定**与**流动性自强化**的双重Rebase机制，打破传统代币的下跌恶性循环。以下是核心原理与抗螺旋设计：

1. Rebase机制的核心逻辑

- 铸造成本单向上行：
每个Milestone的**基础铸造量衰减**（如25%）叠加**FOMO系数递增**，使铸造成本 p 持续上升，公式为：
$$p = \frac{P_0 \cdot d}{M_0 \cdot f^{(m-1)}}$$
 - d 只增不减， $f < 1 \rightarrow$ 成本曲线**严格单调递增**。
- 市场价格动态锚定：
 - 地板价支撑**：市场价格若低于当前铸造成本，套利者倾向于**买入代币**而不是**参与铸造**，推高价格，铸造减速。
 - 天花板限制**：市场价格高于铸造成本时，套利者倾向于**参与铸造**而不是**买入代币**，铸造量增加，并且带来流动性注入的增加，推动市场价格。

2. 抗死亡螺旋的三重防线

防线1：流动性护城河

- Token Vault自动注资**：
每笔铸造费持续注入流动池，形成**买盘深度累加**。
- 手续费再投资**：
DEX交易手续费100%回流至流动池，增强抗跌能力（类似比特币矿工费维护网络安全）。

防线2：成本-价格博弈均衡

市场行为	Rebase响应	经济效果
价格下跌接近铸造成本	套利者买入代币 \rightarrow 需求增加	价格回升至成本线以上
价格暴跌远低于成本	铸造暂停 \rightarrow 供应停止	避免抛压进一步恶化
价格上涨脱离成本	铸造量增加 \rightarrow 流动性注入抑制泡沫	维持合理估值区间

防线3：反抛售激励

- 延迟抛售惩罚**：
囤币者若选择持有至后续Milestone，将面临：
 - 更高铸造成本** \rightarrow 新入场者需支付更高价格，降低抛售竞争压力。
 - 流动性深度增长** \rightarrow 大额抛售被深度吸收，冲击成本升高。

3. 与传统模型的对比验证

场景	无Rebase机制（如pump.fun）	Flipflop.fun Rebase机制
价格下跌	恐慌抛售 \rightarrow 流动性枯竭 \rightarrow 死亡螺旋	套利买入铸造 \rightarrow 流动性注入支撑

场景	无Rebase机制（如pump.fun）	Flipflop.fun Rebase机制
Bot批量砸盘	无成本锚定，价格自由落体	砸盘价低于成本触发自动买入平衡
长期冷清	流动性耗尽，代币归零	铸造暂停，Token Vault储备维持底线价值

总结

Rebase机制通过**成本刚性上升**和**流动性自适应注入**，构建了三级反脆弱体系：

- 1. **短期**：套利博弈快速修复价格偏离
- 2. **中期**：流动性护城河吸收异常波动
- 3. **长期**：成本锚定为代币赋予内在价值底线

这种设计将传统金融的**做市商护盘**和**价值存储**功能编码为链上协议，使得死亡螺旋在数学上被消除，而非依赖人为干预。

A18. 未来是否计划支持非Solana链？

Flipflop.fun将基于协议**可移植性**设计，在2024-2026年逐步实现**全生态多链部署**，具体规划如下：

1. 已确定支持的区块链

类别	区块链名单	技术适配进展
EVM兼容链	Ethereum主网、BSC、Arbitrum、Base、Avalanche、Polygon	已完成以太坊Sepolia测试网验证
高性能L1链	Solana（已支持）、Sui、Aptos	Solana测试网运行中，Move语言评估中
模块化链	Celestia（数据可用层）、Fuel（并行执行）	架构研究阶段

2. 跨链实现的核心技术

- **统一协议层：**
PoM（Proof of Mint）核心算法封装为**跨链智能合约模块**，支持：
 - EVM：Solidity/Vyper版本
 - Move：Sui/Aptos定制版本
 - SVM：Solana原生程序
- **链间状态同步：**
通过Wormhole、LayerZero等跨链桥实现**铸造进度同步**，确保多链版本代币总量一致。
- **流动性聚合：**
在不同链的DEX（如Raydium、Uniswap、PancakeSwap）间建立**统一流动性池**，通过SLP协议共享深度。

3. 各链部署路线图

阶段	时间	目标
Q3 2024	Solana主网优化, EVM测试网上线 (Arbitrum Sepolia)	验证多链PoM算法稳定性
Q4 2024	支持Ethereum主网、BSC、Avalanche	实现EVM全生态覆盖
Q1 2025	Sui/Aptos主网部署	打通Move生态, 支持高性能并行铸造
Q2 2025	跨链流动性聚合 (基于Wormhole)	用户可在任意链参与铸造并跨链交易

4. 多链战略的优势

- **抗单链风险：**
避免因Solana网络拥堵或以太坊Gas波动导致生态瘫痪，提升协议鲁棒性。
- **生态协同：**
 - EVM链吸引DeFi传统用户，Move链服务高频交易场景，Solana聚焦Meme文化社区。
 - 跨链代币标准（如ERC-1155与SPL代币互转）释放组合创新可能。
- **费用优化：**
用户可自主选择最低Gas链参与铸造（如Base链铸造+Solana交易）。

5. 社区治理角色

- **链部署投票：**
通过治理代币发起提案，社区决定扩展目标链优先级（如优先支持Aptos还是Avalanche）。
- **参数分链定制：**
各链可独立设置各自系统参数。

总结

Flipflop.fun的**多链蓝图**不仅是技术扩展，更是构建**抗脆弱代币经济网络**的战略选择。通过EVM+Move+SVM三大技术栈覆盖、跨链状态一致性保障、以及社区驱动的参数治理，将PoM机制打造为首个真正**链无关的公平发行标准**。

A19. 平台治理代币与去中心化治理机制

Flipflop.fun将在条件成熟时（未定）发行原生治理代币，用于实现社区驱动 的协议升级与参数治理。

A20. Flipflop.fun的未来路线图是什么？

Flipflop.fun的未来发展将围绕**AI驱动发行、社交化入口、全链生态和闭环交易市场**四大核心展开，规划草案如下：

1. AI Agent驱动的一级市场（2025 Q1）

- **智能参数配置：**
AI Agent基于链上数据（如FOMO系数、价格波动、Meme趋势）自动优化代币发行参数，包括但不限于：
 - 动态调整**Milestone**缩减系数
 - 预测最佳**目标铸造时间**
 - 实时识别并拦截Sybil攻击模式
- **AI运营顾问：**
 - 为项目方提供**代币发行建议**，生成从创意到落地的全套AI服务并提供实时顾问。
 - 为参与者推荐**铸造策略**（如最优成本窗口期）。
 - 为运营方提供推广策略，并通过AI Agent自动实施。
 - 为市值管理者提供**资金管理建议**，并通过AI Agent自动实施。
- **首个性：**
全球首个将AI深度整合至代币发行全流程的平台，实现从“人工博弈”到“算法均衡”的跨越。

2. Telegram/Discord Mini App集成（2024 Q3）

- **功能亮点：**
 - **一键铸造：**通过Telegram聊天界面直接参与代币铸造，无需切换钱包。
 - **社区治理：**集成投票机器人，基于Telegram群组发起参数调整提案。
 - **社交化传播：**分享铸造成就至频道，吸引裂变用户。

3. 全链生态扩展（2024-2025）

阶段	目标链	关键进展
2024 Q3	EVM测试网（Arbitrum Sepolia）	验证AI Agent跨链适应性
2024 Q4	Ethereum主网、BSC、Avalanche	支持ERC-404代币标准，实现半同质化资产发行
2025 Q1	Sui、Aptos（Move VM）	推出并行铸造功能，支持高并发Meme季
2025 Q2	Celestia+Fuel模块化架构	实现数据可用性与执行层分离，降低90%存储成本

4. 去中心化交易所（FlipDEX）上线（2025 Q2）

- **核心功能：**
 - **流动性聚合：**整合Raydium（Solana）、Uniswap（EVM）、Cetus（Move）等DEX深度。
 - **跨链无缝交易：**通过Wormhole实现代币跨链兑换（如SOL链铸造→ETH链卖出）。
 - **Rebase增强池：**自动将交易手续费转换为Token Vault注资，形成反脆弱流动性。
- **一级-二级市场闭环：**
 - **首发优势：**Flipflop.fun铸造的代币优先上线FlipDEX，享受零上市费。
 - **成本锚定做市：**根据实时铸造成本调整做市算法，减少价格脱钩风险。

总结

Flipflop.fun的路线图以**AI重构发行逻辑、社交捕获流量、全链覆盖场景、DEX闭环价值**为核心，致力于成为Web3时代的“代币经济操作系统”。通过技术堆栈的垂直整合与生态的水平扩展，最终实现**公平性、效率与规模**的“不可能三角”突破。

B. Launch

B1. 如何发行一个代币？

只需6步，1分钟完成发行：

- 1. 登录测试网
访问 <https://test.flipflop.fun>，连接钱包（支持Phantom、Solflare等Solana链钱包）。
- 2. 创建代币
 - 点击左侧菜单 **Launch Token**
 - 填写基础信息：代币名称、符号、图标（支持URL或上传）
 - 可选：设置铸造启动时间、社交媒体链接
- 3. 选择模型
 - **Standard**：最大总量为100,000,000枚，适合项目方
 - **Meme**：最大总量为1,000,000,000枚，适合社区币
- 4. 链上发布
点击 **Create Token**，签署钱包交易（支付测试网Gas费）。
- 5. 等待确认
约30秒后，链上完成代币合约部署。
- 6. 查看并管理代币
进入 **Tools → My Deployment**，查看代币地址、铸造进度、流动性设置。

注意：

- 测试网代币无实际价值，仅供功能验证

B2. Standard和Meme模式有什么区别？如何选择？

Flipflop.fun预置了两种代币经济模型，分别针对不同场景优化，以下是核心差异与选择策略：

1. 核心参数对比

不同点

参数	Standard模式	Meme模式
最大供应量	100,000,000（1亿枚）	1,000,000,000（10亿枚）

参数	Standard模式	Meme模式
Milestone衰减系数	50%（每阶段供应量减半）	25%（每阶段供应量减少25%）
毕业时总铸造费用	1,000 - 4,457 SOL	250 - 1,114.25 SOL
毕业时铸造代币量	50,000,000枚（50%）	250,000,000枚（25%）
DEX初始价格范围	0.00008 - 0.000353 SOL/代币	0.000004 - 0.000018 SOL/代币
单次铸造费用	0.2 SOL	0.01 SOL（降低95%）
单Checkpoint目标铸造量	200,000代币	1,000,000代币（5倍于Standard）

相同点

参数	配置
目标完成时间	约5.79天
初始单次铸造量	10,000代币
目标Milestone	1（完成即毕业）
每个Milestone的Checkpoints	250个
单Checkpoint目标时间	2000秒（约33分钟）
流动性注入比例	20% 铸造费注入DEX

2. 经济模型差异

- Standard模式
 - 强通缩模型：每Milestone代币供应量减半，适合需通过**稀缺性增值**的项目（如DeFi协议、工具型代币）。
 - 高成本过滤投机：单次铸造费0.2 SOL，抑制短期炒作，吸引长期持有者。
 - 典型用例：DAO治理代币、基础设施积分。
- Meme模式
 - 高流通性设计：10亿总供应量+低单价（0.000004 SOL起），适合需要**社区裂变传播**的代币。
 - 超低参与门槛：单次铸造费仅0.01 SOL，1 SOL可铸造10万枚，吸引大众参与。
 - 典型用例：动物币、社交文化Meme代币。

3. 选择策略

维度	Standard模式	Meme模式
项目目标	长期生态建设、价值存储	短期热度引爆、社区文化传播
代币分配	控制流通量，避免通胀	鼓励广泛分发，快速形成持币基数
社区类型	开发者、机构投资者为主	散户、KOL、Meme爱好者为主

维度	Standard模式	Meme模式
风险偏好	低波动性，抗下跌	高波动性，追求爆发增长

4. 操作建议

- 项目方选择
 - 融资+生态控制 → 选Standard模式，通过通缩模型提升代币价值。
 - 冷启动社区 → 选Meme模式，利用低价策略激发用户裂变。
- 普通用户选择
 - 稳健收益 → 参与Standard铸造（如0.2 SOL/次），关注长期价值。
 - 短期投机 → 选择Meme模式（0.01 SOL/次），利用高数量博取涨幅。

B3. 如果Standard和Meme模式都无法满足我的需求，想定制代币的参数，可以吗？

可以，请与平台联系。

B4. 发币时预计代币价格以及预计的铸造总费用是怎么计算的？为什么是一个范围？

Flipflop.fun的代币价格和总铸造费用范围通过**流动性池初始化模型**和**FOMO系数动态性**共同决定，以下是详细计算过程：

1. 核心公式与参数

流动性池初始化价格公式

根据论文定义，代币初始价格由注入流动性池的资金和代币量决定：

$$Price = \frac{0.90 \cdot TotalFee}{InitLiquidity}$$

- TotalFee**：铸造总费用（扣除协议费5%和推荐费0-5%后，90-95%注入流动性池，此处取90%下限）
- InitLiquidity**：流动性池初始代币量

关键参数定义

参数	说明	Standard模式值	Meme模式值
\$P_0\$	单次铸造费用	0.2 SOL	0.01 SOL
\$T_0\$	每个Checkpoint的目标铸造量	200,000 代币	1,000,000 代币
\$M_0\$	单次铸造的基础铸造量	10,000 代币	10,000 代币
\$C\$	每个Milestone的Checkpoints数	250	250
\$E\$	总Milestone数（发币时默认1）	1	1
\$f\$	Milestone衰减系数	50%	75%

参数	说明	Standard模式值	Meme模式值
r_l	流动性池占比（总供应量）	20%	20%

2. 分步计算示例（Standard模式）

步骤1：计算初始流动性池代币量（InitLiquidity）

$$\text{InitLiquidity} = C \cdot T_0 \cdot r_l \cdot \frac{1 - f^E}{(1 - f)(1 - r_l)}$$

代入参数：

- $C=250$, $T_0=200,000$, $r_l=0.2$, $f=0.5$, $E=1$
- $$\text{InitLiquidity} = 250 \cdot 200,000 \cdot 0.2 \cdot \frac{1 - 0.5^1}{(1 - 0.5)(1 - 0.2)} = 12,500,000 \text{ 代币}$$

步骤2：计算铸造总费用范围（TotalFee）

$$\text{TotalFee} \in \left[\frac{P_0 \cdot T_0}{M_0} \cdot C_e, \frac{P_0 \cdot T_0}{M_0} \cdot 101 \cdot (1.01^{C_e} - 1) \right]$$

其中 $C_e = E \cdot C = 250$

- 下限（无FOMO递增）：**

$$\text{TotalFee}_{\min} = \frac{0.2 \cdot 200,000}{10,000} \cdot 250 = 4 \cdot 250 = 1,000 \text{ SOL}$$
- 上限（FOMO每Checkpoint递增1%）：**

$$\text{TotalFee}_{\max} = \frac{0.2 \cdot 200,000}{10,000} \cdot 101 \cdot (1.01^{250} - 1) \approx 4,457 \text{ SOL}$$

步骤3：计算初始价格范围

- 最低价格（ P_{low} ）：**

$$P_{\text{low}} = \frac{0.90 \cdot \text{TotalFee}_{\min}}{\text{InitLiquidity}} = \frac{0.90 \cdot 1,000}{12,500,000} \approx 0.000072 \text{ SOL/代币}$$
- 最高价格（ P_{high} ）：**

$$P_{\text{high}} = \frac{0.90 \cdot \text{TotalFee}_{\max}}{\text{InitLiquidity}} = \frac{0.90 \cdot 4,457}{12,500,000} \approx 0.000321 \text{ SOL/代币}$$

3. 范围形成的核心原因

- FOMO系数动态性：**
 - 若铸造速度快于目标，FOMO系数按每Checkpoint最高1%递增，导致总费用指数增长。
 - 若速度达标，FOMO系数保持1，总费用取最小值。
- 流动性池比例约束：**

20%的初始流动性比例锚定代币总量，价格与总费用呈线性关系。

4. 设计意义与用户策略

- 价格锚定：**初始价格由算法强制约束，避免项目方操纵。
- 参与策略：**

- **低风险用户**：按目标节奏铸造，获得最低成本价（0.000288 SOL）。
- **高风险用户**：抢跑铸造，支付更高成本（最高0.001283 SOL）以获取早期筹码。

总结

Flipflop.fun通过**刚性流动性比例**和**动态费用模型**，将代币初始价格范围转化为可量化的数学区间。这种设计既保障了去中心化发行的公平性，又为市场博弈保留了弹性空间。

B5. 在启动代币时，如何设置name和symbol？有什么限制？

在Flipflop.fun（基于Solana链）发行代币时，名称和符号需遵循以下硬性规则：

1. 代币名称（Name）的限制

- **长度限制**：最多支持 **32个字符**（包括空格和符号）。
- **允许字符**：
 - ASCII字母（A-Z, a-z）
 - 数字（0-9）
 - ASCII标点符号（如 **!, @, #, \$, %** 等）
 - 空格（但不能连续多个空格）
 - **单个表情emoji符号**（如 🚀、💎 等）
- **禁止行为**：
 - 名称不能全为空格（如" "）。
 - 不能包含非ASCII字符（如中文、日文）。
 - 不能包含连续空格（如" "）。

有效示例：

- Dragon Coin 🐉
- DeFi_Protocol_2023
- Meme Token!

无效示例：

- （全空格）
- 比特币（含非ASCII字符）
- Hello World（连续空格）

2. 代币符号（Symbol）的限制

- **长度限制**：最多支持 **10个字符**（建议3-5字符）。
- **允许字符**：
 - ASCII字母（A-Z, a-z）
 - 数字（0-9）
 - **单个emoji表情符号**（如 🚀、💎 等）
- **禁止行为**：
 - 符号不能包含空格、标点符号或其他特殊字符（如 \$, @）。
 - 不能使用多个表情符号（如 🚀💎 ）。

有效示例：

- DRGN
- M202
- 🚀 (仅一个表情符号)

无效示例：

- DEFI\$ (含标点符号)
- BT C (含空格)
- 🚀 🌟 (多个表情符号)

3. 代币Metadata (URI) 的规则

- **格式要求：**必须以 `https://` 开头。
- **长度限制：**URI链接不超过 **200个字符**。

有效示例：

- `https://node1.irys.xyz/F7qkMeZGbDwZbbo6E6Xk0ahuUlGEcXElWgGnH5zm4z`

无效示例：

- `ftp://example.com/icon.png` (协议不支持)
- `data:image/png;base64,...` (非https/ipfs协议)

4. metadata格式

代币metadata Uri打开后的格式如下示例：

```
{
  name: "Satoshi",
  image:
"https://gateway.irys.xyz/fzuVcFKiJaeuRzz3JSWXs3aC3JPxYyggDQCzel_uv",
  symbol: "SATO",
  description: "Hello Satoshi",
  creator: {
    name: "Flipflop",
    site: "https://app.flipflop.fun"
  },
  extensions: {
    website: "",
    twitter: "https://x.com/shanghai",
    telegram: "",
    discord: ""
  }
}
```

其中creator和extensions字段可根据需要自定义。name, image, symbol, description字段是必须的。

5. name和symbol是否可以重复？

- name+symbol的组合必须唯一：即使符号相同，只要名称不同，仍可创建新代币
- symbol不能重复，且大小写敏感
- name大小写不敏感，非大小写完全一致的name可以重复

有效示例：

- Symbol **SAT0**和**Sato**算作重名，谁先发行谁先拥有。
- Name **Satoshi token**和**satoshi token**不算作重名，可以重复。

设计意图与用户指导：

- 避免混淆：防止恶意用户创建名称/符号高度相似的代币（如BTC和Btc），有效防止仿盘代币的混淆攻击。
- 灵活性与唯一性的平衡：
 - 项目方可复用短符号（如MOON），但需搭配独特名称（如Moon Project vs Moon Meme）。
 - 符号建议全大写（如SOL），兼容市场惯例。
- 操作建议：
 - 发行前通过Solana浏览器（如Solscan）查询名称和符号组合是否已被占用。
 - 若符号已被占用，可添加后缀（如DRGN2）或改用更具体的名称（如Dragon Protocol）。该设计的目的是为了确保Symbol的唯一性，避免重复，以免混淆。

B6. 代币的metadata和图像数据保存在哪里？会丢失吗？

在 Flipflop.fun 平台上发行的代币，其 metadata（元数据）和图像数据的存储设计优先考虑去中心化和持久性。

存储位置

1. 首选存储：Irys 网络

- 代币的 metadata 和图像数据优先存储在 **Irys 网络** 上。Irys 是一种专为去中心化存储优化的解决方案，旨在提供高效、低成本且永久的数据存储。
- **特点**：Irys 通过与 Arweave 网络的集成，提供永久存储，同时优化了数据上传的速度和成本。它支持快速交易确认，适合需要频繁上传数据的场景，如代币 metadata 和图像。

2. 备选存储：Arweave 网络

- 如果 Irys 网络不可用或出于其他考虑，metadata 和图像数据会存储在 **Arweave 网络** 上。Arweave 是一个去中心化的永久存储网络，专注于数据的长期保存。
- **特点**：Arweave 使用“一次支付，永久存储”的模型，用户支付一笔费用后，数据将被永久存储在区块链上。

简介

- **Irys 网络**
 - Irys（前身为 Bundlr）是一个基于 Arweave 的 Layer 2 解决方案，旨在提升去中心化存储的效率。它通过批量处理数据上传并优化费用结构，降低了存储成本，同时保持了数据的永久性。

- **优势**：快速上传、低成本、与 Arweave 无缝集成，适合大规模数据存储需求（如 NFT 或代币 metadata）。

- **Arweave 网络**

- Arweave 是一个去中心化的存储协议，通过“区块编织”（Blockweave）技术实现数据的永久保存。用户支付一次性费用，数据会被多个节点存储，确保长期可用性。
- **优势**：数据不可篡改、永久存储、无需持续维护费用。

数据会丢失吗？

- **不会丢失：**

- 无论是存储在 Irys 还是 Arweave 上，代币的 metadata 和图像数据都采用去中心化的永久存储机制。一旦上传成功，数据会被分布式存储在多个节点上，不依赖单一服务器，避免了传统中心化存储（如 AWS）可能面临的数据丢失风险。
- **Irys 的保障**：Irys 依赖 Arweave 的底层存储技术，确保数据的持久性。即使 Irys 网络本身出现问题，数据仍可通过 Arweave 访问。
- **Arweave 的保障**：Arweave 的设计目标是数据保存至少 200 年，通过经济激励机制鼓励节点长期存储数据。

B7. 我可以发行多个代币吗？

可以。

只要遵守name和symbol的命名规则，就可以发行多个代币。

B8. 如何找到我发行的所有代币？

在APP中点击菜单中Tools-> My Deployments，即可查看所有发行的代币。

B9. 通过flipflop.fun发行的代币安全性如何？代币验证器的打分如何？

Flipflop.fun通过智能合约对Solana的SPL代币权限进行严格管控，确保代币发行的安全性与去中心化。以下是完整的权限处理机制：

1. Mint权限（代币增发）

- **初始设置：**
 - 代币创建时立即**丢弃Mint权限**，确保总量固定。
 - 所有铸造的代币自动转入**Mint Token Vault（PDA账户）**，由Flipflop.fun程序控制。
 - 以上模式被称为：**预铸造+PDA托管模式**
- **安全规则：**
 - **无人可增发**：即使项目方也无法修改代币总量。
 - **空投由合约执行**：仅通过程序逻辑分配代币，避免人为操控。
 - 支持所有限制Mint权限尚未丢弃的代币的DEX。

2. Freeze权限（代币冻结）

- **初始设置：**
 - 代币创建时**丢弃Freeze权限**，禁止任何账户冻结代币。

- **安全影响：**
 - 用户资产绝对安全，不受项目方或第三方冻结威胁。
 - 支持所有限制Freeze权限尚未丢弃的代币的DEX。

3. Close权限（代币关闭）

- **初始设置：**
 - 代币创建后，发币人**保留Close权限**，但受以下限制：
 - **未开始铸造，或所有代币都被Refund时：**可关闭代币（退回账户租金）。
 - **铸造启动后：**永久禁用Close权限，确保代币持续存在。
- **设计意图：**
 - 防止项目方在募集资金后恶意销毁代币。

4. Metadata更新权限

- **可控的灵活性：**
 - **允许更新：**发币人可修改除name和symbol外的所有Metadata（如社交链接、Banner图片）。
 - **权限关闭：**提供**一键禁用**功能，锁定后Metadata永久不可更改。
- **安全边界：**
 - 禁止修改name和symbol，防止重名攻击。

5. 权限对比（Flipflop.fun vs 传统SPL代币）

权限	传统SPL代币	Flipflop.fun
Mint	项目方可保留增发权限	永久丢弃，总量固定
Freeze	项目方可冻结用户资产	永久丢弃，资产永不冻结
Close	项目方可随时关闭代币	仅铸造前允许，铸造后禁用
Metadata	项目方可任意修改所有信息	仅开放非关键字段，可永久锁定

6. 安全设计总结

1. **抗增发：**通过丢弃Mint权限，杜绝通胀风险。
2. **抗冻结：**用户资产完全由私钥控制，不受第三方干预。
3. **防跑路：**铸造开始后禁用Close权限，确保代币长期存在。
4. **透明可验证：**所有权限变更记录在链上，用户可通过区块链浏览器（如Solscan）验证。

Flipflop.fun通过上述设计，将代币控制权从项目方转移至智能合约和社区，实现了真正的去中心化发行。

B10. 既然mint权限被丢弃了，又是如何实现mint代币的？

Flipflop.fun通过创新的**预铸造+PDA托管**模式，在丢弃Mint权限的前提下实现安全的代币分发，以下是详细机制：

1. 代币发行的两阶段机制

阶段1：预铸造与权限销毁

- **预铸造所有代币：**
代币创建时，**一次性预铸全部供应量**（如1亿枚）并转入代币铸造池**Mint Token Vault**（PDA账户）。
- **立即丢弃Mint权限：**
预铸造完成后，智能合约自动**永久销毁Mint权限**，确保无人能增发代币。

阶段2：动态分发（空投）

- **PDA账户托管：**
代币铸造吃池**Mint Token Vault**由Flipflop.fun的智能合约控制，按以下流程分配代币：
 - 用户铸造时，合约从Vault中提取对应数量代币空投至用户账户。
 - 按**流动池占比**参数，将用于流动池的代币空投至Token Vault账户。
 - **无人可手动操作：**因为不拥有PDA账户的owner权限，包括项目方在内的任何角色均无法直接转移Mint Token Vault中的代币。

2. 关键安全设计

设计	说明
总量硬顶	预铸造后丢弃Mint权限，代币总量恒等于初始设置（如1亿枚）。
去中心化分发	仅通过智能合约逻辑分配代币，规避人为操控风险。
PDA不可篡改	Mint Token Vault 作为PDA账户，没有私钥，无人可提取或挪用其中代币。

3. 代币铸造池（Mint Token Vault）销毁机制（毕业后）

- **功能目的：**
允许社区在代币毕业后（完成目标Milestone），按需销毁Vault中未分发的代币，实现：
 - **总量缩减：**提升代币稀缺性。
 - **停止铸造：**彻底终止后续分发。
 - **仅限未分发代币：**已空投给用户的代币不受影响。

*举例

某meme币，最大总量为10亿枚，在毕业时，有2.5亿枚代币已经分发，此时，社区认为2.5亿枚已经足够，于是发起销毁操作。销毁所有的代币铸造吃池（Mint Token Vault）中的代币后，代币总量变为2.5亿枚，后续铸造将停止。

4. 与传统模型的对比优势

特性	传统SPL代币	Flipflop.fun
增发风险	项目方可保留Mint权限随意增发	预铸造后权限永久销毁，总量固定
分发控制	依赖项目方手动转账	智能合约自动空投，无人为干预
透明度	需信任项目方操作	所有分发记录链上可查

5. 用户验证方式

- **区块链浏览器查询：**

通过Solscan等工具验证代币的：

1. **Mint权限状态**：显示为`null`（已丢弃）。
2. **Mint Token Vault余额**：实时查看未分发代币数量。
3. **销毁交易记录**：毕业后销毁操作公开透明。

总结

Flipflop.fun通过**预铸造+权限销毁+PDA托管**的三重保障，在杜绝增发风险的同时实现公平分发。毕业后可选的代币销毁功能，为社区提供了灵活的通货紧缩工具，真正实现了「代码约束权力」的去中心化理念。

B11. 为什么区块链浏览器中代币持有人排行中第一个账户的持有量这么大？

该账户即代币铸造池（Mint Token Vault）的地址，因为flipflop采用**预铸造+PDA托管**模式，所以这个账户中持有的代币总量为初始设置的总量，而不是实际发行的总量。

B12. 如果发币后，发现需要修改，我能在链上删除原来已经发行的代币吗？

1. 可修改的内容

- **允许修改：**

- **Metadata**（代币元数据）：
 - 社交链接（Twitter、Discord等）
 - Banner图片
 - 项目描述
 - 官网URL

- **禁止修改：**

- **代币名称（Name）**
- **代币符号（Symbol）**
- **Mint账户（代币核心身份）**

2. 如何彻底“删除”已发行的代币？

如果代币尚未被铸造（即无人参与），可以执行以下操作：

1. 关闭并回收资源：

- 使用**Close**指令关闭除**Mint账户**外的所有关联账户（如Metadata账户）。具体操作：菜单->Tools -> My Deployments -> 选择代币 -> Close Token
- **回收SOL租金**：关闭账户后，部分存储租金SOL会退回。

2. 重新创建代币：

- 由于**Mint账户**无法删除，需**重新部署新代币**（使用新的Name/Symbol）。
- 旧代币将永久存在于链上（但已无铸造等功能，因关联账户已关闭）。

3. 关键限制与注意事项

场景	是否可行	解决方案
修改Name/Symbol	✗ 不可行	必须创建新代币
删除Mint账户	✗ 不可行（Solana底层限制）	旧Mint账户永久存在，但可弃用
代币已被铸造	✗ 不可删除	仅能通过社区治理调整后续规则（如销毁）

4. 操作建议

- **发行前严格校验**：确保Name/Symbol符合规则且无拼写错误。
- **利用Metadata灵活性**：通过更新社交链接、Banner等维护项目形象。
- **紧急情况处理**：若需彻底重置，应在**零铸造时**立即关闭旧代币并重新部署。

5. 设计意图

- **安全性**：防止恶意项目方随意篡改代币身份（如仿盘骗局）。
- **灵活性**：通过Metadata更新满足项目发展需求。
- **链上透明**：所有修改记录永久可查，避免中心化操控。

总结：Flipflop.fun在保障去中心化安全的前提下，通过智能合约为用户方提供了合理的调整空间，但核心代币属性（Name/Symbol/Mint）一经部署即不可篡改，确保用户资产可信。

B13. 如何估算铸造的计划完成时间，以及每小时或每分钟计划次数？

Flipflop.fun的铸造时间估算公式基于**等比数列求和模型**与**动态衰减机制**构建，以下是详细的数学推导：

1. 总时间公式

每个Milestone的时间为 $C \cdot t_{\text{check}}$ （ t_{check} 为单Checkpoint目标时间），因此总时间：

$$T_{\text{total}} = E \cdot C \cdot t_{\text{check}}$$

2. 案例1（Standard模式）

- **参数：**
 $C = 250$, $t_{\text{check}} = 2000$ 秒, $E = 1$
- $T_{\text{total}} = 250 \cdot 2000 \cdot 1 = 500,000$ 秒 \approx **5.79天**

3. 案例2（Meme模式）

- **参数：**
 $C = 250$, $t_{\text{check}} = 2000$ 秒, $E = 1$
- $T_{\text{total}} = 250 \cdot 2000 \cdot 1 = 500,000$ 秒 \approx **5.79天**

B14. 代币的metadata中的社交信息能更新吗？更新metadata时要收费吗？

Flipflop.fun的代币metadata是可更新的，用户可以随时修改其中的社交信息（如Twitter、Discord等）。

1. 更新流程

- **用户操作**：用户在APP中点击菜单中Tools-> My Deployments -> 选择代币 -> Update Metadata。
- **智能合约执行**：智能合约会检查用户是否拥有Update Metadata权限，无Update Metadata权限，无法更新。

2. 费用

- **更新费用**：更新metadata的费用为0.1 SOL。
- **小技巧**：用户可以在发币时就设置好metadata，以减少更新时的手续费。注意：代币主页的背景图无法在发币时设置，只能在发币后更新。

B15. 如何防止metadata随意更新给社区造成不良影响？

发币人可以随时关闭metadata的更新权限，以防止metadata随意更新给社区造成不良影响。

1. 关闭更新权限

- **用户操作**：用户在APP中点击菜单中Tools-> My Deployments -> 选择代币 -> Authority -> Revoke Metadata Update。当该权限显示为Mutable时，说明该权限可以被更新，如果显示为Immutable时，说明该权限不可被更新。

B16. 我可以设定一个代币的启动时间吗？

可以。

在发币时，可以设置一个未来的时间，当该时间到达后，代币才会开始铸造。

在该时间之前，用户可以照常生成URC推荐码。

FAQ of flipflop.fun

C. Mint and refund

C1. 如何找到代币信息？

您可以通过以下四种途径获取代币信息并进入铸造页面：

1. 二维码扫描

直接扫描其他用户推荐图片右下角的动态二维码，自动跳转至代币铸造页面。

2. 社交链接直达

点击社交媒体（Twitter/Telegram等）中的推荐链接，支持一键跳转。

3. 平台精准搜索

在flipflop.fun应用首页搜索框输入代币名称、合约地址或符号（需区分大小写），实时获取项目信息。

4. 社交网络追踪

在flipflop.fun应用"社区"板块关注项目方或KOL主页，实时查看其推荐代币的动态更新。

C2. 如何铸造代币？

铸造流程（三步完成）：

1. 进入代币主页

通过上述任一方式定位目标代币页面。

2. 输入验证信息

点击"铸造"按钮，在弹窗中输入有效的URC码（用户推荐码）。

3. 确认铸造操作

连接钱包后点击"开始铸造"，支付Gas费并签署交易，系统将实时分配代币至您的钱包。

提示：铸造前请确保钱包有足够SOL余额并保持网络畅通。

C3. 铸造代币必须要有URC吗？如何获取？

■ URC必要性说明

URC（用户推荐码）是铸造代币的必需凭证，其设计包含双重安全保障：

1. 反自动化攻击

采用链上加密机制，阻断Bot批量获取。

2. 社区信任体系

通过真实用户间的推荐关系，建立强大的真人社区。

■ URC获取方式

若您未持有URC，可通过以下途径获取：

- **社区互动获取**

在项目官方社区（TG群/Discord）向管理员或活跃成员申请

- **线下活动兑换**

参与项目方举办的AMA或meetup活动获取高质量URC码

■ 链下防窃听机制

URC采用端到端加密传输技术，全程通过链下渠道（如图片/私信）分发，防止链上数据监听。每个URC码确保被真人使用。

C4. 能否使用自己生成的URC码？

系统限制使用本人URC码进行铸造。

C5. 如何计算mint所得的代币成本？

Flipflop.fun的铸造成本通过**动态算法模型**实时计算，核心公式与参数如下：

1. 核心公式

单次铸造的每代币成本：

$$p = \frac{P_0 \cdot d}{M_0 \cdot f^{(m-1)}}$$

总铸造成本（完成目标Milestone）：

$$\text{TotalCost} = \frac{P_0 \cdot T_0}{M_0} \cdot \sum_{i=1}^{C_e} d_i$$

参数说明：

符号	定义	示例（Standard模式）
P_0	单次铸造固定费用	0.2 SOL
d	当前FOMO系数（难度系数）	初始为1，铸造过快时递增
M_0	单次基础铸造量	10,000 代币
f	Milestone衰减系数	0.5（Standard模式）
m	当前Milestone编号	1（首个Milestone）
T_0	单Checkpoint目标铸造量	200,000 代币
C_e	总Checkpoints数（目标Milestone）	250

2. 分步计算示例（Standard模式）

场景1：按目标速度铸造（ $d=1$ ）

- 单次成本：
$$p = \frac{0.2 \cdot 1}{10,000 \cdot 0.5^0} = \frac{0.2}{10,000} = 0.00002 \text{ \text{SOL/代币}}$$
- 总成本：
$$\text{TotalCost} = \frac{0.2 \cdot 200,000}{10,000} \cdot 250 \cdot 1 = 4 \cdot 250 = 1,000 \text{ \text{SOL}}$$

场景2：快速铸造（ d 升至1.5）

- 单次成本：
$$p = \frac{0.2 \cdot 1.5}{10,000 \cdot 0.5^0} = 0.00003 \text{ \text{SOL/代币}}$$
- 总成本：
$$\text{TotalCost} = 4 \cdot 250 \cdot 1.5 = 1,500 \text{ \text{SOL}}$$

3. 影响成本的关键因素

因素	对成本的影响	示例
铸造速度	速度越快 → d 递增 → 成本上升	加速50% → 成本+50%
Milestone衰减	新Milestone启动 → 基础量 M_0 减少 → 成本指数上升	第2个Milestone成本 ×2（ $f=0.5$ ）

C6. 如何找到我铸造的所有代币？

在flipflop.fun的应用中，点击菜单-> My minted tokens，即可看到所有铸造的代币。

D. Unique Referral Code (URC)

D1. 如何生成URC推荐码？

在flipflop.fun应用中，进入代币页面，点击紫色的“生成URC”按钮，链接钱包签名，即可生成URC推荐码。

如果验证码已经生成，同样的方式，可以查看验证码已经被使用的次数，以及专属的推荐链接，可以将推荐链接发给社区，点击该链接，可以直接铸造，无需再输入URC码。

如果验证码的使用次数达到上限（50次），可以点击“激活”按钮，再次激活验证码。需要注意的是，两次激活验证码之间的时间间隔不能短于24小时。

D2. URC有什么好处？

使用URC铸造的人和推广URC的推荐人，可通过以下方式获得直接和间接收益：

D3. URC推荐码对推荐人和使用者的好处及计算公式

1. 推荐人（URC提供者）的好处

- **奖励机制：**推荐人可获得被推荐人节省费用的 **20%** 作为奖励。
- **公式：**

$$\text{推荐人奖励} = 0.2 \cdot (P_0 - \text{Fee}) = 0.2 \cdot P_0 \cdot k \cdot \left(1 - \frac{1}{d}\right)$$
 - P_0 ：单次铸造固定费用
 - k ：折扣率（由推荐人代币持有比例决定）
 - d ：当前难度系数（FOMO系数）

2. 被推荐人（URC使用者）的好处

- **费用折扣：**根据推荐人的代币持有比例，享受 **0%~25%** 的铸造费用折扣。
- **公式：**

$$\text{实际费用} = P_0 \cdot \left(1 + \frac{k}{d} - k\right)$$
- **折扣率 (k) 由推荐人代币持有比例 (r) 决定：**

$$k = \begin{cases} 0\% & r < 0.2\% \\ 5\% & 0.2\% \leq r < 0.4\% \\ 10\% & 0.4\% \leq r < 0.6\% \\ 15\% & 0.6\% \leq r < 0.8\% \\ 20\% & 0.8\% \leq r < 1\% \\ 25\% & r \geq 1\% \end{cases}$$

3. 示例计算

参数设定

- 单次铸造固定费用 $P_0 = 0.1 \text{ ETH}$
- 推荐人代币持有比例 $r = 0.5\%$ （对应 $k = 10\%$ ）
- 当前难度系数 $d = 1.2$

步骤1：计算被推荐人实际费用

$$\text{Fee} = 0.1 \cdot \left(1 + \frac{0.1}{1.2} - 0.1\right) = 0.1 \cdot (1 + 0.0833 - 0.1) = 0.1 \cdot 0.9833 =$$

0.09833\ \text{ETH} \\$\\$

- **节省费用：**
 $\$P_0 - \text{Fee} = 0.1 - 0.09833 = 0.00167\ \text{ETH} \$\$$

步骤2：计算推荐人奖励

$\$ \text{推荐人奖励} = 0.2 \cdot 0.00167 = 0.000334\ \text{ETH} \$\$$

4. 不同场景对比

场景	难度系数 (d)	折扣率 (k)	实际费用 (ETH)	推荐人奖励 (ETH)
无折扣 ((k=0%))	1.2	0%	0	0 (无折扣率时无奖励)
标准示例	1.2	10%	0.09833	0.000334
高难度 ((d=2.0))	2.0	10%	$0.1 \cdot (1 + 0.1/2 - 0.1) = 0.095$	$0.2 \cdot (0.1 - 0.095) = 0.001$
最大折扣 ((k=25%))	1.0	25%	$0.1 \cdot (1 + 0.25/1 - 0.25) = 0.1$	0 (无难度系数时无奖励)
极高难度 ((d=100.0))	100.0	25%	$0.1 \cdot (1 + 0.25/100 - 0.25) = 0.07525$	$0.2 \cdot (0.1 - 0.07525) = 0.00495$

5. 结论

5.1. 推荐人收益：

- 奖励与 **节省费用** 正相关，难度系数 d 越高，节省幅度越大（如 $d=2.0$ 时奖励增至0.001 ETH）。
- 推荐人持有更多代币（提升 k ）可显著增加收益（如 $k=25\%$ 时奖励上限更高）。

5.2. 被推荐人收益：

- 费用节省依赖 **k 和 d 的比值**，高难度时折扣效果更显著（如 $d=2.0$ 时费用节省5%）。
- 若难度系数 $d=1.0$ （无难度增长），最大折扣 $k=25\%$ 也无法降低费用，需结合动态难度参与。

6. 设计目的

6.1. 双赢设计：

- 推荐人通过推广获得 **持续收益**（每笔铸造均分奖励），激励社区裂变。
- 被推荐人享受 **成本优化**，尤其在高难度阶段（如市场火热时）节省显著。

6.2. 社区驱动增长：

- **激励持币**：更高的持币比例意味着更大的折扣吸引力，促使推荐人长期持有代币，而非抛售。
- **提升共识**：URC推广者通过吸引更多人参与铸造，推动社区规模扩大，从而增强代币的共识价值。
- **正向循环**：社区成员越多，代币需求可能上升，价格稳定性增强，间接提升推荐人的持币价值。

6.3. 抵消FOMO拉动的铸造成本上升：

当铸造速度过快时，FOMO系数（难度系数）上升，导致单个Token成本增加，但使用推荐人的URC可以降低铸造成本，激励用户更多的参与社区互动。

6.4. 无需许可的推广机制：

- **链上运行**：URC机制完全基于链上智能合约运行，无需中心化审核，推广者可自由发展自己的“子社区”。
- **低门槛参与**：任何用户均可自行生成URC并推广，收益完全透明且即时结算。
- **自动计算**：不同推广人的URC所能产生的折扣和推广收益，只取决于持币量，任何人无法手动调整该数据。

7. 总结

URC机制通过**链上算法**将社区推广与动态成本控制结合，实现以下目标：

1. **经济激励**：推广者收益与持币量、市场热度正相关，形成可持续增长飞轮。
2. **公平性保障**：折扣率仅由持币比例决定，避免人为操纵。
3. **抗波动设计**：高难度阶段通过折扣抵消成本上升，维持用户参与意愿。
4. **去中心化治理**：完全依赖代码规则，无需信任中介，符合Web3精神。

该机制是首个将**持币权益**与**社区推广**深度绑定的去中心化解决方案，为代币经济模型提供创新范式。

D4. 因使用URC会给铸造者折扣，并给推荐人奖励，会对毕业时的总铸造费用产生什么影响？

1. 核心机制公式

URC机制通过双重调节影响净铸造费，**净铸造费用公式**：

- **被推荐人**：支付费用降低（公式： $\text{Fee} = P_0 \cdot (1 + \frac{k}{d} - k)$ ）
- **推荐人**：获得节省费用的20%（公式： $0.2 \cdot P_0 \cdot k \cdot (1 - \frac{1}{d})$ ）
- **系统净影响**：每笔URC铸造的实际净收入为：
 $\text{NetFee} = \text{Fee} - \text{推荐人奖励} = P_0 \cdot \left(1 - 1.2k + \frac{1.2k}{d}\right)$ 其中：
 - P_0 ：单次铸造固定费用（Standard=0.2 SOL，Meme=0.01 SOL）
 - k ：折扣率（0%~25%）
 - d ：FOMO系数(难度系数)

2. 对总铸造费用的影响

原总铸造费用范围（无URC）： $\text{TotalFee} \in \left[\frac{P_0 T_0}{M_0} C_e, \frac{P_0 T_0}{M_0} \cdot 101 \cdot (1.01^{C_e} - 1) \right]$ 引入URC后，需将 P_0 替换为 NetFee ，修正范围为： $\text{TotalFee}_{\text{URC}} \in \left[\frac{P_0 (1 - 1.2k + \frac{1.2k}{d}) T_0}{M_0} C_e, \frac{P_0 (1 - 1.2k + \frac{1.2k}{d}) T_0}{M_0} \cdot 101 \cdot (1.01^{C_e} - 1) \right]$

3. 两种模式参数对比

参数	Standard模式	Meme模式
单次费用 P_0	0.2 SOL	0.01 SOL

参数	Standard模式	Meme模式
目标铸造量\$T_0\$	200,000代币	1,000,000代币
基础铸造量\$M_0\$	10,000代币	10,000代币
Checkpoints数\$C_e\$	250	250
原费用范围	[1,000, 4,457] SOL	[250, 1,114.25] SOL

4. 极端场景计算 (k=25%, d→∞)

当k=25%, d→∞时, 总铸造费范围变为 : $\$ \text{TotalFee}_{\text{URC}} \in \left[0.7 \cdot \frac{P_0 T_0}{M_0 C_e}, 0.7 \cdot \frac{P_0 T_0}{M_0} \cdot 101 \cdot (1.01^{C_e} - 1) \right] \$$

4.1 Standard模式

- 净费用 : $\$ \text{NetFee} = P_0 \cdot (1 - 1.2k) = 0.2 \cdot (1 - 1.2 \cdot 0.25) = 0.14 \text{ SOL} \$$
- 流动性池影响 : 减少30% SOL, $(0.2 - 0.14) / 0.2 = 30\%$
- 总铸造费范围 : $\$ [0.7 \cdot 1,000, 0.7 \cdot 4,457] = [700, 3,119.9] \text{ SOL} \$$

3.2 Meme模式

- 净费用 : $\$ \text{NetFee} = 0.01 \cdot 0.7 = 0.007 \text{ SOL} \$$
- 流动性池影响 : 减少30%
- 总铸造费范围 : $\$ [0.7 \cdot 250, 0.7 \cdot 1,114.25] = [175, 780] \text{ SOL} \$$

4. 极端情况下经济影响对比

影响维度	Standard模式	Meme模式
单次净收入	0.2→0.14 SOL (-30%)	0.01→0.007 SOL (-30%)
总费用上限	4,457→3,119.9 SOL	1,114.25→780 SOL
总费用下限	1,000→700 SOL	250→175 SOL
推荐人收益/次	0.01 SOL	0.0005 SOL

5. 社区增长补偿 :

- 假设URC带来100%更多参与者, 实际流动性池规模 : $\$ \text{ActualLP} = 0.7 \cdot 2 = 1.4 \cdot \text{原始值} \$$

6. 结论

在极端情况下, URC机制在两种模式下会导致总费用范围缩小30%, 但通过 :

- 社区增长带来交易量补偿
- 模式差异化设计平衡激励

最终形成**可控的成本转移**（从流动性池转向社区激励），而非系统性风险。建议持续监控 k 和 d 的实际分布，必要时动态调整参数。

总结

URC机制通过**费用再分配**（折扣+奖励）降低净铸造收入，导致毕业时总费用范围同比收缩约**0-30%**（具体取决于 k 和 d 的分布）。这一设计以牺牲部分流动性初始资金为代价，换取更快的社区冷启动和长期生态活力，属于典型的**用短期成本换长期增长策略**。

D5. 所有URC都一样吗？不同URC有什么区别？

1. URC核心差异维度

不同URC的本质区别体现在以下三方面：

差异维度	影响因素	可变范围	智能合约控制方式
折扣力度	推荐人持币比例(r)	0%-25%	通过实时查询持币比例自动计算
奖励分成	平台治理规则	固定20%分成	不可修改的智能合约常数

2. 经济价值差异公式

单个URC的经济价值(EV)可量化为： $EV = \sum_{i=1}^n \left[\frac{0.2P_{0k_i}(1-\frac{1}{d_i})}{(1+s)^t} \right]$ 其中：

- k_i = $\min(25\%, \text{推荐人持币量}/\text{当前流通量})$
- s = 风险折现率（建议取值15%）
- t = 铸造发生的时间周期

3. 典型URC类型对比

URC类型	持币比例	折扣率	每日使用上限	预估收益（Standard模式）
鲸鱼URC	$\geq 1\%$	25%	50次	18-22 SOL/日
社区领袖URC	0.6-0.8%	15%	50次	3-5 SOL/日
散户URC	0.2-0.4%	5%	50次	0.1-0.3 SOL/日
系统默认URC	0%	0%	50次	0 SOL

4. 对参与方的影响

对铸造者：

- 选择高折扣URC可节省：
 - 最高25% 费用（当 $d \rightarrow \infty$ 时节省0.05 SOL/次）
 - 典型节省3-8%（平均 $d \approx 1.5$ 时）

对推荐人：

- 收益取决于：
 - 持币量（决定k）
 - 推广能力（影响使用次数）
 - 市场热度（影响d值）
-

D6. 如何获取高质量的URC？

1. 增加持币量

- 提高钱包内代币数量，直接提升URC折扣等级（\$k\$值）
- 持币比例 $\geq 1\%$ 可获得最高25%折扣权限

2. 社区互动

- 实时监控社区动态，抢先获取高持币用户发布的URC
- 参与项目AMA、测试网活动等获取专属URC

3. 加入大户子社区

- 通过Discord/Telegram等加入持币大户的专属社群
- 获取稳定高折扣URC（通常要求质押一定代币）

4. 链上监控

- 使用区块链浏览器追踪大额持币地址
- 设置交易警报捕获优质URC生成交易

5. 时间策略

- 在市场冷清期（\$d\$值较低）时更易获取高性价比URC
 - 项目早期阶段URC质量普遍较高
-

D7. 如果我设置了一个将来的代币启动时间，在启动前，能生成urc并进行推广预热吗？

- **预启动URC生成**：在代币启动时间（T0）之前，项目方可随时生成有效URC
 - **预热期特性**：
 - URC可正常分发推广
 - 铸造功能锁定至启动时间自动解锁
-

D8. 为什么我分享的URC失效了？

1. URC失效的核心原因

- **使用次数耗尽**：每个URC默认50次使用上限
- **时间锁限制**：重新激活需间隔 ≥ 24 小时

2. 防滥用设计原理

机制	技术实现	防护效果
次数限制（50次）	智能合约计数器	防止单URC无限刷量
24小时冷却期	时间戳验证（Solana Slot计算）	抑制Bot高频生成新URC
唯一性标识	PDA账户地址绑定	确保每个URC独立计数

3. 链上验证方法

用户可通过App或Solana浏览器查询：

- 剩余次数
- 下次激活时间

D9. 相同推荐人推荐不同的项目时，每个项目的URC是一样的吗？

1. URC核心生成机制

- 基础构成要素：

$\text{URC}_{\text{default}} = \text{TokenSymbol} \oplus \text{Shorten}(\text{ReferrerPubkey})$

其中：

- \oplus 表示连接符"_"
- $\text{Shorten}()$ 取公钥首尾各8字符

- 示例：

```
# 输入
token = "SATO"
pubkey = "5UtbDq2jGZF9DQJ71VoeGVRd85DRRosmNWxhoPY9tgNi"

# 输出
URC_default = "SATO_5UtbDq2joPY9tgNi" # 首8+尾8字符
```

2. 自定义URC规则

类型	字符限制	唯一性校验范围	示例
默认URC	固定格式	全链	SATO_5UtbD...tgNi
自定义URC	3-24字符(字母数字)	同代币项目内	SATO_ToTheMoon

3. 跨项目URC特性

- **完全独立：**
 - 相同推荐人在不同项目的URC互不影响
 - 每个URC绑定到特定代币合约地址

4. 关键注意事项

1. 自定义URC不可包含：
 - 特殊符号（除"_"外）
 - 建议保留代币符号作为前缀（如"SATO_"）

D10. 如何找到并管理我推广的代币？

为方便用户管理自己推广的代币，flipflop.fun提供了“我的URC”功能。

在app中点击: [菜单](#) -> [Tools](#) -> [我的URC](#)，即可看到所有推广的代币。

D11. 如何查看URC码的信息，如有效性，铸造折扣，使用次数，下次激活时间等？

在App中点击: [菜单](#) -> [Tools](#) -> [Validate URC](#)，输入URC码，即可验证URC码的有效性和铸造折扣。

D12. 如何查询有多少人用我的urc铸造，我获得了多少奖励？

在App中点击: [菜单](#) -> [Tools](#) -> [我的URC](#)，选中一个推广的代币，点击[Bonus Detail](#)

D13. 如果用我的urc铸造的人Refund了，会扣除我已经获得的佣金吗？

1. 核心规则

- **佣金保护原则：**推荐人已获奖励不会被追回
- **Refund扣减来源：**从退款申请人的本金中优先扣除应返还的推荐奖励

2. 资金流模型

```
\begin{aligned}
&\&\text{RefundAmount} = \text{MintFee} - \text{RewardFee} - \\
&\text{RefundFee} \\
\end{aligned}
```

如果铸造时没有使用折扣URC，则不会扣除RewardFee。

D14. 我如何分享一个我喜欢的代币？推特分享，图片分享，链接分享。

点击代币页面右上角[Share](#)按钮，你有三种分享方式：

- 复制推荐链接；
- 转发到Twitter (X)；

- 生成分享图片，图片中有代币的信息和推荐链接的二维码。

D15. 收到一个朋友的推广图，我该怎么办？

用手机扫描该图片右下角的二维码，进入代币页面，开始铸造。

E. Refund

E1. 如何Refund？

在App中点击: 菜单 -> Tools -> My Minted Tokens, 选中一个铸造的代币，点击Refund按钮，即可Refund。

E2. Refund时的扣费如何计算？我Refund后能得到多少资金？

1. Refund时的扣费计算

Refund时的扣费主要包括两部分：

- **Refund Fee（退款手续费）**：固定为退款金额的5%。
- **URC奖励返还**（如果铸造时使用了推荐码）：需扣除推荐人已获得的奖励。

计算公式：

$$\text{RefundAmount} = \text{MintFee} - \text{RewardFee} - \text{RefundFee}$$

其中：

- **MintFee**：用户最初支付的铸造费用（如0.1 SOL），多次铸造会累计。
- **RewardFee**：铸造时使用URC推荐码产生的推荐人累计奖励（若未使用URC，此项为0）。
- **RefundFee**：退款手续费，为MintFee的5%，由平台收取。

示例：

- 用户支付铸造费 0.1 SOL，未使用URC：
 - 退款金额 = 0.1 SOL - 0 - (0.1 × 5%) = 0.095 SOL
- 用户支付铸造费 0.1 SOL，使用URC且推荐人奖励为 0.001 SOL：
 - 退款金额 = 0.1 SOL - 0.001 SOL - (0.1 × 5%) = 0.094 SOL

2. 资金退回流程

- **退回金额**：用户实际收到的资金为扣除上述费用后的净值（见公式）。
- **代币处理**：退回的代币会被**立即销毁**，代币总量相应减少。
- **限制条件**：
 - 必须确保钱包内代币余额与铸造时数量**完全一致**，否则无法退款。
 - 仅限在****共识期（目标Milestone完成前）****内发起Refund。

E3. Refund在代币的生命周期中始终有效吗？

Refund只在共识期内（即目标Milestone达到之前）有效。

E4. Refund后，退回的代币去哪了？会减少代币总量吗？

退回的代币被立即销毁，因此Refund会减少代币总量。

E5. Refund时，提示我数量和铸造时的数量不符，无法Refund，怎么办？

Refund时，必须确保钱包内代币的余额等于铸造时的数量。

如果数量不符，请务必从其他地方获取代币，确保与铸造时数量相等。

如果您的代币不是通过铸造所得，不能Refund。

F. 毕业与市值增长

F1. 在共识期内，任何人都能添加流动性吗？代币能交易吗？

根据Flipflop.fun的机制设计，关于共识期内的流动性和交易规则如下：

1. 代币转账

- **允许转账**：代币在铸造启动后即可自由转账（即发币后即开放转账功能）。
- **技术基础**：代币遵循Solana的SPL标准，默认支持转账，除非智能合约额外限制（但Flipflop.fun未设置此类限制）。

2. 流动性添加

- **开放添加**：任何用户（包括项目方或社区成员）均可主动在Raydium等DEX创建流动性池。
- **风险提示**：
 - 共识期内代币价格波动剧烈，因缺乏官方流动性支撑，可能面临高滑点或低深度问题。
 - 官方流动性池（由Token Vault资金支持）仅在**目标Milestone完成后**由发币人或市值管理者（VM）按计划注入。



3. 交易可行性

- **可交易但风险高**：虽然代币可自由买卖，但共识期内交易依赖社区自发形成的流动性，可能存在以下问题：
 - 价格易受操纵（如大户抛售导致暴跌）。
 - 流动性不足时难以成交。

4. 官方建议

- 项目方通常建议等待**目标Milestone完成后**，通过官方流动性池交易，以获得更稳定的市场环境。

总结

-  **可以**：任何人在共识期内添加流动性或交易。
-  **但需注意**：非官方流动性池风险较高，建议谨慎参与。
- **官方流动性**：目标Milestone完成后由Token Vault资金支持的专业流动性管理启动。

F2. flipflop.fun支持哪些dex的池子？

Flipflop.fun **支持所有 Solana 生态的 DEX**，但主要针对 **Raydium** 进行优化，并具有自动创建流动池的机制。具体规则如下：

1. 支持的 DEX

 **所有 Solana DEX 均可自由添加流动性**，包括但不限于：

- **Raydium**（支持最新的CPMM协议，即cp-swap）
- **Orca**
- **Jupiter（聚合器，依赖底层 DEX）**
- **Saber（稳定币 AMM）**
- **Mercurial**
- **Lifinity**
- **其他 Solana AMM 协议**





2. 毕业时的自动流动性创建（仅 Raydium的CPMM）

- **如果毕业时（目标 Milestone 完成）Raydium 上仍无该代币的流动池**，Flipflop.fun 的智能合约会 **自动在 Raydium 上创建 SOL/代币 交易对**，并注入初始流动性。
- **资金来源**：
 - **Token Vault** 中的 SOL（来自铸造费用）。
 - **Token Vault** 中预留的代币（按预设比例，如 20% 总供应量）。
- **LP Token 处理**：
 - **默认销毁**（确保无人可提取流动性资金）。
 - 也可由 **市值管理者（VM）** 选择保留，用于后续流动性管理。

3. 其他 DEX 需手动添加流动性

- 如果社区或项目方希望在 **Orca、Jupiter 等其他 DEX** 上提供流动性，需 **手动操作**。
- Flipflop.fun **不会自动在这些平台创建池子**，但代币仍可正常交易。

总结

DEX	支持情况	流动性创建方式
Raydium(CPMM)	 主要支持	智能合约自动创建（毕业时）
Orca	 支持	需手动添加
Jupiter	 支持（聚合交易）	依赖底层 DEX 流动性
其他 Solana DEX	 支持	需手动添加

F3. 毕业时流动性池子是自动创建的吗？有多少币和资金会进入池子

Flipflop.fun 在代币 毕业（完成目标 Milestone）时，会 自动在 Raydium 上创建流动性池（如果尚未有人创建），并注入初始流动性。具体规则如下：

1. 流动性池是否自动创建？

✅ 是的，默认自动创建（仅限 Raydium）。

- **触发条件**：当目标 Milestone 完成时，若 Raydium 上 没有该代币的 SOL 交易对，智能合约会自动执行以下操作：
 1. **创建 SOL/代币 交易对**（如 SOL/FFF）。
 2. **注入初始流动性**（资金来自 Token Vault）。
 3. **创建CPMM池的费用**约0.2SOL，由Token Vault支付
- **如果已有流动性池**：则不会重复创建。

2. 多少代币和资金进入池子？

流动性池的初始资金由 **Token Vault** 提供，具体金额由发币人或流动性管理者确定。

注：实际SOL和代币的注入量可能因 Refund、URC 奖励扣除等因素略有浮动。

3. LP Token 的处理

- **可选保留**：发币人可委托 **市值管理者（VM）** 保留 LP Token，用于后续流动性调整（如做市策略）。

4. 为什么选择 20% 流动性比例？

- **足够深度**：20% 代币 + 对应 SOL 能提供合理的初始交易深度，减少滑点。
- **抗操纵**：避免流动性占比过低导致价格容易被操控。
- **剩余代币用途**：
 - 80% 代币通过铸造分发（共识期 + 增长期）。

F4. 流动池锁仓吗？

Flipflop.fun 的流动性池 **默认不自动锁仓**（即 LP Token 不会被自动销毁），而是由 **发币人（Issuer）或市值管理人（VM）** 决定是否销毁或保留 LP Token。

1. 流动性池的 LP Token 处理方式

选项	操作方式	影响
销毁	发币人或 VM 主动销毁 LP Token	<ul style="list-style-type: none">- 流动性永久锁定，无人可撤资。- 增强社区信任，避免 Rug Pull 风险。

选项	操作方式	影响
保留	由 VM 管理 LP Token，用于动态调整流动性（如做市、对冲无常损失等）。	<ul style="list-style-type: none">- 灵活应对市场波动，优化资金效率。- 需高度信任 VM。

2. 设计意图

(1) 灵活性优先

- 适应不同项目需求：
 - **Meme 币**：可能选择销毁 LP Token，确保去中心化和抗操纵。
 - **长期生态项目**：可能保留 LP Token，用于专业做市或应对极端行情。
- **市值管理人 (VM) 的作用**：
 - 专业的 VM 可通过调整流动性深度，减少价格波动，提升市场稳定性。

(2) 抗风险与信任平衡

- **销毁 LP Token**：
 - **优点**：完全去中心化，杜绝项目方撤资跑路 (Rug Pull)。
 - **缺点**：失去流动性调控能力，若市场暴跌无法干预。
- **保留 LP Token**：
 - **优点**：可通过增减流动性平滑价格曲线。
 - **缺点**：需依赖 VM 的道德和专业性 (Token Vault 的无法提现功能降低了 VM 的作恶可能性)。

(3) 社区治理方向

- 可引入 **DAO 投票**，即由一个多签账户作为 VM，让持币者共同决定市值管理，以及 LP Token 的处置方式 (如部分销毁或保留)。

(4) AI Agent

- 我们正在评估使用 **AI Agent** 进行市值管理的可行性。

3. 为什么不像其他平台强制锁仓？

- **避免过度僵化**：强制锁仓可能限制项目应对市场变化的能力 (例如需要临时增加流动性应对 FUD)。
- **信任与效率权衡**：
 - Flipflop.fun 的设计理念是 **"代码约束 + 灵活治理"**，而非一刀切的规则。
 - 通过透明化 VM 操作 (如链上记录资金动向) 降低信任成本。
- **设计本质**：在去中心化信任与市场效率之间寻求最优解，并赋予人最大的决定权，而非简单牺牲一方。

F6. 谁能管理项目钱包Token Vault中的资产？

Flipflop.fun 的 **Token Vault** (项目资金池) 采用 **多重权限管理** 设计，确保资金安全且用途透明。以下是具体规则：

1. 管理权限归属

角色	权限范围	操作限制
智能合约	自动执行资金分配（如毕业时注入流动性）。	完全去中心化，无人可干预。
发币人（Issuer）	可设置 市值管理人（VM） ，但 无法直接动用资金 。	仅能通过智能合约预设规则操作（如授权 VM）。
市值管理人（VM）	若被授权，可管理流动性（如调整做市策略），但 无法提现或转移资金 。	资金仅能用于链上流动性操作（如 DEX 添加/移除流动性），且所有记录公开。
社区治理（未来）	计划引入 DAO 投票机制，对重大资金使用进行表决（如更改 VM）。	尚未实现，需治理代币支持。
AI Agent（未来）	计划引入 AI Agent，对资金使用进行决策并执行交易。	调研中

2. 资金安全设计

- **Token Vault 是 PDA 账户：**
 - 无私钥控制，仅能通过 **Flipflop.fun 智能合约** 调用资金。
 - 即使发币人或 VM 也无法直接提现，确保资金不会被挪用。
- **资金用途强制限制：**
 - 只能用于 **流动性管理**（如 Raydium 添加/移除流动性）或 **Refund 退款**。
 - 无法转账至个人钱包或其他合约。

3. 市值管理人（VM）的职责与约束

- **职责：**
 - 根据市场情况调整流动性（如对抗暴跌或过度波动）。
 - 执行发币人预设的做市策略（如阶梯式注入流动性）。
- **约束：**
 - **24 小时交易限额：**发币人可设定 VM 单日最大操作额度，防止过度干预。
 - **紧急冻结：**发币人或未来 DAO 可投票冻结 VM 权限（如发现恶意行为）。

4. 设计意图

- **去中心化与可控性的平衡：**
 - 发币人不能随意操控资金，但可通过专业 VM 优化市场表现。
 - **VM 权限受限**，避免中心化作恶风险。
- **透明化运作：**
 - 所有资金流动记录在链上可查（如通过 Solscan）。
 - 社区可监督 VM 的操作合规性。

F7. 共识期内，Token Mint中的资金可以使用吗？什么时候才能使用？

1. 共识期内（目标 Milestone 完成前）

- ❌ 不可使用：
 - Token Vault 中的资金（铸造费 + 代币）在共识期内 **完全冻结**，仅用于：
 - Refund 退款**：如果用户选择退款，资金从 Vault 中扣除并返还。
 - 无人可动用**，包括发币人、市值管理人（VM）或平台。

2. 可使用资金的触发条件

✅ 仅当目标 Milestone 完成（代币毕业）后，资金才会按规则释放：

- 流动性注入：
 - 由市值管理人决定将Token Vault中的SOL铸造费 + 代币注入 Raydium 创建流动池。
- 后续铸造费分配：
 - 毕业后的铸造费仍进入 Token Vault，但可由 **市值管理人（VM）** 按计划用于：
 - 追加流动性。
 - 对抗市场波动（如暴跌时买入支撑）。

3. 设计意图

- 用户资金安全保障**：
 - 共识期内资金锁定，确保 Refund 机制始终可执行，避免项目方挪用。
- 去中心化启动**：
 - 毕业前依赖社区自发交易，毕业后由智能合约自动分配资金，减少人为干预。

总结

阶段	资金状态	可使用场景
共识期内	❌ 冻结	仅限 Refund 退款。
毕业后	✅ 解锁	自动注入流动性 + VM 管理后续资金。

F8. 如何授权第三方流动性管理者（VM）操作流动性钱包？

操作步骤：

- 进入管理页面
 - 登录 Flipflop.fun 网页或 App，点击 **菜单 → Tools → My Deployment**。
- 选择代币
 - 找到需要管理的代币，点击进入详情页。
- 设置市值管理人（VM）
 - 点击 **Authority**，在 **“委托市值管理账号”** 输入框中填写 VM 的 Solana 钱包地址。
- 确认授权
 - 连接发币人钱包，签署交易完成授权。

注意：

- 默认 VM 是发币人自己**（即发币人地址），如需更换需手动输入第三方地址。
- 授权后，VM 可管理流动性，但 **无法提现资金**，仅能按规则操作流动性池。

F9. 发行方可以在共识期或发行期就指定VM吗？

是的，发行方可以随时指定市值管理人（VM）：

1. 授权时间

- **共识期内**：随时通过代币管理页面（**Authority**）设置或更换VM。
- **发行期（毕业后）**：仍可调整VM，但需重新签署授权交易。

2. 默认设置

- 若未主动指定，发币人自身钱包地址将作为默认VM。

3. 操作限制

- VM权限仅限流动性管理（如DEX做市），**无法提取Token Vault资金**。
-

F10. Token Vault中的代币部分是怎么获得的？

1. 代币来源

Token Vault中的代币通过 **预铸造（Pre-Mint）机制** 自动生成，规则如下：

- **初始总量**：根据代币模式（Standard/Meme）预设（如Standard=100M，Meme=1B）。
- **预铸造分配**：
 - **100%** 存入 **Mint Token Vault**（用于用户铸造和流动性分发）。

2. 计算

- **流动性代币（20%）**：
每次用户铸造一定数量代币，**Mint Token Vault**都会同时向用户账户和Token Vault空投代币，其中 \$\$
TokenVault代币数量 = $\frac{\text{用户铸造的代币数量}}{1 - \text{流动性代币比例}(20\%)}$
\$\$
-

F11. Token Vault中的sol是如何获得的？

1. SOL的来源

Token Vault中的SOL全部来自用户的**铸造费用**，扣除以下两部分后存入：

1. **协议费用（5%）**：固定扣除，用于平台维护。
2. **URC推荐奖励（0-5%）**：若铸造时使用了推荐码，则按规则扣除奖励（最高5%）。

2. 计算公式

存入Token Vault的SOL = 用户支付铸造费 - 协议费用（5%） - URC奖励（0-5%）

- **举例**（Standard模式，单次铸造费0.2 SOL）：
 - **无URC**： $0.2 \text{ SOL} - 0.01 \text{ SOL (5\%)} = 0.19 \text{ SOL}$ 存入Vault。
 - **有URC（奖励3%）**： $0.2 \text{ SOL} - 0.01 \text{ SOL (5\%)} - 0.006 \text{ SOL (3\%)} = 0.184 \text{ SOL}$ 存入Vault。
-

F12. Token Vault中的资产可以withdraw吗？

不能。

F13. Token Vault中的资产可以如何使用？

1. 允许的操作

Token Vault中的资产（SOL和代币）**仅限以下流动性管理用途**：

1. 添加流动性

- 将SOL和代币注入DEX（如Raydium）创建/扩大流动性池。

2. 解除流动性

- 从DEX撤出流动性（需配合做市策略，如调整池子深度）。

3. 买入代币

- 用SOL在市场上买入代币（用于支撑价格或回购销毁）。

4. 卖出代币

- 卖出代币换取SOL（用于对冲抛压或平衡资金）。

5. 销毁LP Token

- 销毁流动性凭证（永久锁定流动性，防止撤资）。

2. 禁止的操作

- **❌ 提现至个人钱包**
- **❌ 转账至非合约授权的地址**
- **❌ 用于与流动性无关的支出**

3. 执行条件

- **发币人或VM操作**：需通过Flipflop.fun智能合约执行，所有交易链上可查。
- **智能合约管控**：仅支持上述5类操作的函数调用，其他行为自动拒绝。

4. 设计意图

- **专款专用**：确保资金100%服务于代币流动性，杜绝挪用风险。
 - **去中心化监督**：所有操作公开透明，社区可通过区块链浏览器验证。
-

F14. 技术上如何确保Token Vault中的资金安全？

Token Vault账户是一个PDA账户，owner是Solana程序本身，任何人无法改变其账户余额。

1. PDA账户（Program Derived Address）

- **无私钥控制**：Token Vault是智能合约生成的PDA账户，**没有私钥**，无法通过外部签名操作资金。
- **唯一操作权限**：仅Flipflop.fun的智能合约程序可调用该账户，人工或其他合约无法干预。

2. 智能合约硬编码规则

资金仅允许以下链上操作（通过预置合约函数）：

1. **流动性管理**（添加/解除DEX流动性）。
 2. **市场操作**（买入/卖出代币）。
 3. **LP Token销毁**。
 4. **Refund退款**。
- 其他任何转账或提现请求均会被自动拒绝。**

3. 链上透明性

- 所有资金流动记录在Solana链上公开可查（如通过Solscan）。
- 社区可实时监控Token Vault余额和操作历史。

4. 安全边界

- **物理隔离**：即使Flipflop.fun团队也无法触碰资金，完全由代码控制。
- **抗攻击设计**：合约已通过审计，关键函数（如资金转移）设置多重校验。

总结：

通过**PDA账户 + 智能合约白名单函数 + 全链透明**三重机制，确保资金「可用不可盗」。

F15. 发行方如何防止委托的VM（市值管理人）渎职？

Flipflop.fun 通过 **双重风控机制** 防止 VM 滥用权限，确保资金安全：

1. 紧急冻结功能

- **即时生效**：
发币人可随时在代币管理页面（**Authority**）**冻结VM权限**，立即终止其所有操作权限。
- **适用场景**：
 - VM 异常交易（如频繁撤流动性）。
 - 社区投票质疑其行为（未来DAO治理）。

2. 24小时交易限额

- **额度管控**：
发币人可设定VM **每24小时的最大资金操作量**（如最多动用Vault中10%的SOL）。
- **触发拦截**：
若VM尝试超限操作，智能合约自动拒绝交易。

3. 设计意图

- **权力制衡**：
VM可专业管理流动性，但发币人保留紧急制动权。

- **风险兜底：**
限额机制避免单次恶意操作造成重大损失。

F16. 默认的市值经理（VM）是谁？

发币人。

F17. VM如何查看管理的所有代币？

操作步骤：

1. **登录Flipflop.fun**
 - 访问官网或打开App，连接VM的钱包（需已授权为市值管理人）。
2. **进入代币管理页面**
 - 点击 **菜单** → **Token Management** → **Delegated Tokens**。
3. **查看代币列表**
 - 页面将自动显示 **所有被授权管理的代币**（包括代币名称、余额、流动性状态等）。

注意：

- 仅显示已授权给该VM地址的代币，其他代币不可见。
- 需确保钱包地址与发币人授权的VM地址一致。

F18. 毕业后，如果社区担心代币铸造永无止境，希望终止铸造，该怎么办？

解决方案：销毁铸造池（Mint Token Vault）

1. **操作步骤：**

- 发币人或VM通过 **Flipflop.fun 的智能合约** 发起销毁：
 - 进入 **菜单** → **Token Management** → **Delegated Tokens**，选择目标代币。
 - 点击 **Burn Mint Token Vault**，输入希望销毁的数量，确认交易并支付Gas费。
- **效果：**
 - 按指定的数量，销毁 **Mint Token Vault** 中剩余未铸造的代币。

2. **前提条件：**

- 需确保代币已 **完成目标Milestone（毕业）**，否则需先等待毕业。
- 销毁后，**已流通的代币**不受影响，仍可正常交易。

示例：

某Meme币初始总量10亿枚，已铸造2.5亿枚后，社区投票决定销毁剩余7.5亿枚，彻底停止铸造，总量为2.5亿枚。