F(x) Protocol 源码审计快速指引

本指引已经把“前端/路由/Facet / PoolManager / BasePool / TickLogic / Flashloan / Swap / NFT / Keeper / Oracle / Revenue” 等参与者全部列出，尽量覆盖 所有路径（开仓、减仓、短仓、再平衡、清算、fxUSD 铸/销、稳定币补偿、参数管理、奖励/收入分配、预言机更新、白名单路由与退款）。

注：每个 participant 下两行 注释分别为：职责 / 实现要点。

1) 开仓（Long） — 完整路径（用户用任意代币 -> 转换 -> 闪电贷 -> 注入池 -> 铸 fxUSD -> 还贷 -> NFT）

sequenceDiagram

participant User as User(frontend)

作用：准备参数并提交 tx（borrowAmount/fxUSDAmount/swapData）

要点：前端负责杠杆/滑点预估与 minOut

participant FacetOpen as "PositionOperateFlashLoanFacetV2.sol:openOrAddPositionFlashLoanV2"

作用：开仓入口（构建路由，调用 transferInAndConvert、发 flashloan）

要点：把用户入金转换为抵押并发起 \_invokeFlashLoan

participant LibRouterIn as "LibRouter.sol:transferInAndConvert"

作用：入金路由，用户 token -> 抵押（例如 USDC->WBTC）

要点：走白名单目标合约，严格 minOut 校验

participant Morpho as "MorphoFlashLoanFacetBase.sol:\_invokeFlashLoan"

作用：发起闪电贷并回调 Facet

要点：封装 flashloan 并把借入资产交给回调

participant FacetCB as "PositionOperateFlashLoanFacetV2.sol:onOpenOrAddPositionFlashLoanV2"

作用：回调逻辑（托管 NFT -> PoolManager.operate -> 接收 fxUSD）

要点：临时托管 NFT，调用 PoolManager.operate 更新仓位，收到 fxUSD 后 swap 归还闪电贷

participant NFT as "IERC721"

作用：托管/归还头寸 NFT（xPosition）

要点：transferFrom/transfer 保证回调期间头寸控制权

participant PoolMgr as "PoolManager.sol:operate"

作用：统一入口（缩放、委托 BasePool、更新全局账本）

要点：会调用 \_handleSupply/\_handleBorrow/\_changePoolCollateral/\_changePoolDebts

participant BasePool as "BasePool.sol:operate"

作用：池内具体仓位增减、阈值校验与份额更新

要点：刷新 index/price，若处于清算态则 revert ErrorPositionInLiquidationMode

participant PosLogic as "PositionLogic.sol:\_mintPosition"

作用：创建或更新仓位 NFT 的内部逻辑

要点：写入仓位快照、抵押/债务份额

participant HandleBorrow as "PoolManager.sol:\_handleBorrow"

作用：按借款铸造 fxUSD 给调用方（Facet）

要点：计算借款费并 mint fxUSD 到 Facet 地址

participant FacetCheck as "PositionOperateFlashLoanFacetV2.sol:\_checkPositionDebtRatio"

作用：Facet 层对债务比合规检查

要点：对比前端期望的范围，若不在范围可 revert/退款

participant FacetSwap as "PositionOperateFlashLoanFacetV2.sol:\_swap"

作用：用 fxUSD 通过白名单路由兑换回抵押币并还贷

要点：白名单+minOut 校验，可能分两步：fxUSD->USDC->WBTC

participant RouterRefund as "LibRouter.sol:refundERC20"

作用：退款多余资产到收入池或指定地址

要点：清理残留 token，避免合约残留造成问题

User->>FacetOpen: openOrAddPositionFlashLoanV2(params)

FacetOpen->>LibRouterIn: transferInAndConvert(userToken -> WBTC)

FacetOpen->>Morpho: \_invokeFlashLoan(borrowAmount)

Morpho-->>FacetCB: flashloan callback

FacetCB->>NFT: transferFrom(user -> facet)

FacetCB->>PoolMgr: operate(newColl>0, newDebt>0)

PoolMgr->>BasePool: operate(...)

BasePool->>PosLogic: \_mintPosition(...)

PoolMgr->>HandleBorrow: \_handleBorrow (mint fxUSD -> Facet)

FacetCB->>FacetCheck: \_checkPositionDebtRatio()

FacetCB->>FacetSwap: \_swap(fxUSD -> WBTC)

FacetSwap-->>FacetCB: WBTC

FacetCB->>Morpho: repayFlashLoan(WBTC)

FacetCB->>RouterRefund: refundERC20(剩余 -> revenue)

FacetCB->>NFT: transfer(user)

FacetCB-->>User: return tx / xPosition id

2) 减仓 / 平仓（Long） — 完整路径（闪电贷借入抵押 -> 兑换 -> 还债 -> 更新池账本 -> 转出用户资产）

sequenceDiagram

participant User as User(frontend)

作用：用户发起减仓/平仓请求（目标 token/数量）

要点：前端计算需借入多少抵押以覆盖短时偿还

participant FacetClose as "PositionOperateFlashLoanFacetV2.sol:closeOrRemovePositionFlashLoanV2"

作用：减仓入口，发起借入抵押的闪电贷

要点：构建 flashloan 并传回调参数

participant Morpho as "MorphoFlashLoanFacetBase.sol:\_invokeFlashLoan"

作用：闪电贷发起与回调

要点：提供借入抵押供回调使用

participant FacetCBClose as "PositionOperateFlashLoanFacetV2.sol:onCloseOrRemovePositionFlashLoanV2"

作用：回调中把抵押兑换 fxUSD 并调用 PoolManager.operate(提抵押/还债)

要点：托管 NFT、兑换、还债、计算剩余返还

participant NFT as "IERC721"

作用：临时托管/归还 NFT

要点：保证回调操作时仓位不可被并发操作

participant FacetSwap as "PositionOperateFlashLoanFacetV2.sol:\_swap"

作用：兑换抵押 -> fxUSD 或稳定币以偿还债务

要点：走白名单路由，避免被恶意路由利用

participant PoolMgr as "PoolManager.sol:operate"

作用：处理 newColl<0/newDebt<0（提抵押并还债）

要点：最终调用 \_handleRepay 燃烧 fxUSD 或接受稳定

participant HandleRepay as "PoolManager.sol:\_handleRepay"

作用：燃烧 fxUSD（优先）或接受稳定币偿还并回调 onRebalanceWithStable

要点：更新全局债务并释放对应抵押份额

participant LibRouterOut as "LibRouter.sol:convertAndTransferOut"

作用：把最终资产（例如 USDC 或 ETH）转回给用户

要点：执行路由并确保 minOut

participant LibRouterRefund as "LibRouter.sol:refundERC20"

作用：退款剩余到收入池

要点：避免合约残留

User->>FacetClose: closeOrRemovePositionFlashLoanV2(params)

FacetClose->>Morpho: \_invokeFlashLoan(borrowCollateral)

Morpho-->>FacetCBClose: flashloan callback

FacetCBClose->>NFT: transferFrom(user -> facet)

FacetCBClose->>FacetSwap: \_swap(collateral -> fxUSD)

FacetCBClose->>PoolMgr: operate(newColl<0, newDebt<0)

PoolMgr->>HandleRepay: \_handleRepay (burn fxUSD / accept stable)

FacetCBClose->>LibRouterOut: convertAndTransferOut(userTargetToken)

FacetCBClose->>LibRouterRefund: refundERC20(剩余 -> revenue)

FacetCBClose->>NFT: transfer(user)

FacetCBClose-->>User: return tx / result

3) Short（做空） — 开/加/减/平 & 清算（完整路径）

sequenceDiagram

participant User as User(frontend)

作用：发起短仓操作（借入负债代币 / 提交抵押）

要点：短仓路由与债务代币映射需在前端确定

participant ShortFacetOpen as "ShortPositionOperateFlashLoanFacet.sol:openOrAddShortPositionFlashLoan"

作用：短仓入口（flashloan 借入负债代币）

要点：准备 flashloan 用于开/加短仓

participant Morpho as "MorphoFlashLoanFacetBase.sol:\_invokeFlashLoan"

作用：发起闪电贷（借入负债代币）

要点：回调到短仓 Facet

participant ShortFacetCB as "ShortPositionOperateFlashLoanFacet.sol:onOpenOrAddShortPositionFlashLoan"

作用：回调处理（托管 NFT/调用 ShortPoolManager.operate/兑换）

要点：兑换路径可能是借入负债代币 -> 卖出为抵押 / fxUSD

participant ShortPoolMgr as "ShortPoolManager.sol:operate"

作用：短池开仓/加仓/减仓入口（含 CreditNote 与对手池协同）

要点：短池内维护 credit note、对手池配对与净额计算

participant ShortPosLogic as "ShortPositionLogic.sol"

作用：短仓头寸内部管理（NFT/仓位记录）

要点：记录负债份额与抵押份额，暴露 getPositionDebtRatio

participant ShortSwap as "ShortPositionOperateFlashLoanFacet.sol:\_swap"

作用：短仓侧路由兑换（负债代币 <-> fxUSD/稳定币）

要点：确保白名单和 minOut

participant ShortClose as "ShortPositionOperateFlashLoanFacet.sol:closeOrRemoveShortPositionFlashLoan"

作用：平短仓入口，同样会发起 flashloan 用于周转

要点：回调执行还债、释放抵押

User->>ShortFacetOpen: openOrAddShortPositionFlashLoan(params)

ShortFacetOpen->>Morpho: \_invokeFlashLoan(borrowDebtToken)

Morpho-->>ShortFacetCB: flashloan callback

ShortFacetCB->>ShortPoolMgr: operate(...)

ShortPoolMgr->>ShortPosLogic: update position / mint NFT

ShortFacetCB->>ShortSwap: \_swap(routes)

ShortFacetCB-->>User: return tx / shortPosition id

4) 再平衡（Tick 级 / 全局） — 温和去杠杆路径、奖励与稳定币补偿

sequenceDiagram

participant Trigger as Keeper/Caller

作用：触发再平衡（可以是守护者/策略/用户）

要点：传入 maxFxUSD / maxStable 用于限额控制

participant FxBase as "FxUSDBasePool.sol:rebalance"

作用：基础池入口，拼装参数并调用 PoolManager.rebalance

要点：会拆解为 maxFxUSD 和 maxStable

participant PoolMgrReb as "PoolManager.sol:rebalance"

作用：再平衡调度（tick 或 全局）

要点：会调用 \_beforeRebalanceOrLiquidate / \_afterRebalanceOrLiquidate

participant Before as "PoolManager.sol:\_beforeRebalanceOrLiquidate"

作用：准备计价/缩放因子/限制

要点：读取预言机价格、计算抵押缩放因子

participant BasePoolReb as "BasePool.sol:rebalance"

作用：池内遍历 candidate Tick 并对每个 Tick 执行 \_liquidateTick

要点：跳过可清算 Tick / 坏债 / 尘额

participant TickLogic as "TickLogic.sol:\_liquidateTick"

作用：单 Tick 的去杠杆/份额转移/奖励计算

要点：返回 rawDebt/rawColl/bonus 等供聚合

participant After as "PoolManager.sol:\_afterRebalanceOrLiquidate"

作用：结算（优先火 fxUSD -> 稳定补偿 -> 转抵押 -> 记账）

要点：计算协议费用并转移抵押到接收者

participant LibRouterStable as "LibRouter.sol:onRebalanceWithStable"

作用：当需要用稳定币补偿时的路由入口

要点：接收稳定币并在池侧完成结算

Trigger->>FxBase: rebalance(maxFxUSD,maxStable)

FxBase->>PoolMgrReb: rebalance(...)

PoolMgrReb->>Before: \_beforeRebalanceOrLiquidate()

PoolMgrReb->>BasePoolReb: rebalance(...)

BasePoolReb->>TickLogic: \_liquidateTick(...)

BasePoolReb-->>PoolMgrReb: aggregated rawDebts/rawColls

PoolMgrReb->>After: \_afterRebalanceOrLiquidate()

After->>LibRouterStable: onRebalanceWithStable(if needed)

After-->>Trigger: events / receipts

5) 清算（强制去杠杆） — 触及清算阈值后的强制清算流程

sequenceDiagram

participant Liquidator as Keeper/Actor

作用：发起清算请求（针对超清算阈值的 Tick/Position）

要点：需满足最小清算额 MIN\_LIQUIDATE\_DEBTS

participant FxBase as "FxUSDBasePool.sol:liquidate"

作用：基础池清算入口（拼装 maxFxUSD/maxStable）

要点：会检查并拒绝过小清算请求

participant PoolMgrLiq as "PoolManager.sol:liquidate"

作用：清算调度（调拨储备、遍历 Tick）

要点：调用 \_beforeRebalanceOrLiquidate / BasePool.liquidate

participant BasePoolLiq as "BasePool.sol:liquidate"

作用：逐 Tick 清算并累加奖励/短缺补足

要点：调用 TickLogic.\_liquidateTick 更新份额

participant TickLogic as "TickLogic.sol:\_liquidateTick"

作用：计算清算体量、奖励抵押与来源

要点：输出 rawDebts/rawColl/bonusRawColls/bonusFromReserve

participant After as "PoolManager.sol:\_afterRebalanceOrLiquidate"

作用：清算后的燃烧/稳定补偿/转抵押/发事件

要点：先燃烧 fxUSD，再用稳定币补齐；抵押扣除清算费用转给接收者

Liquidator->>FxBase: liquidate(maxFxUSD,maxStable)

FxBase->>PoolMgrLiq: liquidate(...)

PoolMgrLiq->>PoolMgrLiq: \_beforeRebalanceOrLiquidate()

PoolMgrLiq->>BasePoolLiq: liquidate(...)

BasePoolLiq->>TickLogic: \_liquidateTick(...)

BasePoolLiq-->>PoolMgrLiq: aggregated rawDebts/rawColls/bonus

PoolMgrLiq->>After: \_afterRebalanceOrLiquidate()

After-->>Liquidator: Liquidate event / transfers

6) fxUSD 铸造 / 销毁 与 稳定币补偿（跨图说明）

sequenceDiagram

participant Facet as Facet (any facet caller)

作用：通过 PoolManager.\_handleBorrow/\_handleRepay 获取或偿还 fxUSD

要点：借出时 mint fxUSD 到调用合约；还款时优先燃烧 fxUSD

participant PoolMgrBorrow as "PoolManager.sol:\_handleBorrow"

作用：铸造 fxUSD 并计入借款费用

要点：更新 pool 的 rawDebts/全局债务份额

participant PoolMgrRepay as "PoolManager.sol:\_handleRepay"

作用：燃烧 fxUSD 或接收稳定币用于偿还债务

要点：当 fxUSD 不足且 stableRepayAllowed 时接受稳定并标记 onRebalanceWithStable

participant StableRouter as "LibRouter.sol:onRebalanceWithStable"

作用：把稳定币按协议规则注入 / 兑换为抵押用于补偿

要点：确保稳定币路径安全并触发后续结算

Facet->>PoolMgrBorrow: request borrow (mint fxUSD)

PoolMgrBorrow-->>Facet: fxUSD minted

Facet->>PoolMgrRepay: repay fxUSD (burn) or stable

PoolMgrRepay->>StableRouter: onRebalanceWithStable(if stable)

PoolMgrRepay-->>Facet: position updated / debts decreased

7) Oracle / 价格更新 与 风险定价（参与者与触发）

sequenceDiagram

participant Oracle as PriceOracle

作用：提供抵押与兑换价格（getExchangePrice/getLiquidatePrice）

要点：Pool 在每次 operate/rebalance/liquidate 前会读取 oracle 值

participant Pool as BasePool / PoolManager

作用：读取价格以计算 index/债务比/清算价

要点：价格读取影响 newRawColl/newRawDebt 以及清算决策

participant Keeper as Keeper/Actor

作用：触发再平衡/清算/预警

要点：通常托管在 off-chain 策略中，触发合约对应方法

Keeper->>Oracle: request price feed / trigger update

Oracle-->>Keeper: updated price

Keeper->>Pool: call rebalance/liquidate/operate (uses price)

Pool->>Oracle: getExchangePrice/getLiquidatePrice()

8) 管理/权限/参数变更（Admin 操作路径）

sequenceDiagram

participant Admin as ProtocolOwner

作用：管理阈值、白名单、pause、reserve 参数

要点：具有权限控制函数（onlyOwner/onlyRole）

participant Governance as GovernanceContract

作用：治理/多签调用（可选）

要点：某些参数建议由治理投票变更

participant PoolMgr as "PoolManager.sol"

作用：接收并执行参数变更（rebalanceRatioData / permissionedLiquidationThreshold / capacities）

要点：变更会影响 operate/rebalance/liquidate 的行为

Admin->>PoolMgr: setRebalanceParams(...) / setLiquidateParams(...)

PoolMgr-->>Admin: event / ack

Governance->>PoolMgr: propose & execute (if applicable)

9) 收益 / 费用 / 所得分配（收入流向）

sequenceDiagram

participant PoolMgr as PoolManager

作用：在 \_afterRebalanceOrLiquidate 与 handleBorrow 中计入费/奖励

要点：扣费逻辑写在 \_handleSupply/\_handleBorrow/\_afterRebalanceOrLiquidate

participant Revenue as RevenuePool / Treasury

作用：接收协议收入（借款费、清算费用、再平衡奖励）

要点：收入可用于买回/分发/保险储备

participant Rewards as RewardsDistributor

作用：分发给 LP / stakers（如果有）

要点：驱动激励逻辑（可由 PoolManager 触发）

PoolMgr->>Revenue: transfer fees

PoolMgr->>Rewards: distribute rewards (if applicable)

10) 综合概览（High-level unified view，便于把所有流连起来看）

sequenceDiagram

participant User as User(frontend)

participant Facet as Facet (open/close short/long)

participant Router as LibRouter

participant Flash as MorphoFlashLoanFacetBase

participant PoolMgr as PoolManager

participant BasePool as BasePool / FxUSDBasePool

participant TickLogic as TickLogic

participant Oracle as PriceOracle

participant NFT as IERC721 (xPosition)

participant Revenue as RevenuePool

participant Keeper as Keeper/Actor

User->>Facet: open/close/rebalance/liquidate/short

Facet->>Router: transferIn/convert & swap calls

Facet->>Flash: invokeFlashLoan (if needed)

Flash-->>Facet: callback (onOpen/onClose)

Facet->>PoolMgr: operate / rebalance / liquidate

PoolMgr->>BasePool: delegate operate/rebalance/liquidate

BasePool->>TickLogic: \_liquidateTick / tick ops

PoolMgr->>Revenue: collect fees

BasePool->>Oracle: getExchangePrice/getLiquidatePrice

Keeper->>PoolMgr: call rebalance/liquidate (trigger)

NFT->>User: xPosition minted / transferred