

汇编生成主流程 (convert)

1. 初始化与函数头部生成：
- 规范化函数名，生成汇编标签和段声明

• 输出 `.balign 4`、`.global`、`.section .text` 等指令

• 生成函数入口标签，保存寄存器（`push {r4-r10, fp, lr}`），设置帧指针（`add fp, sp, #32`）

• 若有溢出变量（spilled temps），为其分配栈空间（`sub sp, sp, #N`）
2. 遍历基本块与四元式：
- 对每个基本块，遍历其四元式（QuadStm）列表

• 对每条四元式，根据类型（kind）分情况处理
3. 四元式到汇编的转换：
- **LABEL**：输出汇编标签

• **MOVE**：处理寄存器/内存间的数据移动，必要时加载/存储溢出变量

• **MOVE_BINOP**：二元操作（如加减乘），支持与LOAD/STORE合并优化（如 `ldr/str [base, offset]`）

• **LOAD/STORE**：内存读写，处理溢出变量的加载与存储

• **CALL/MOVE_CALL/EXTCALL/MOVE_EXTCALL**：函数调用，处理参数与返回值

• **JUMP/CJUMP**：无条件/条件跳转，优化多余的连续标签跳转

• **RETURN**：恢复栈帧，弹出寄存器，返回
4. 溢出变量处理：
- 对于分配到栈上的临时变量，使用 `ldr / str` 指令在需要时加载/保存

• 统一使用r9/r10作为溢出变量的临时寄存器

Git Graph

