指令格式：

|  |
| --- |
| 操作码（8位） |

|  |  |
| --- | --- |
| 操作码（8位） | 操作数（8位） |

指令集

|  |  |  |  |
| --- | --- | --- | --- |
| 指令类别 | 助记符 | 指令长度 | 功能 |
| 访存 | Ld, R0, add | 16 | 读取内存单元add至R0 |
| Ld, R1, add | 16 | 读取内存单元add至R1 |
| St, R0, add | 16 | 将R0内容存入内存单元add |
| St, R1, add | 16 | 将R1内容存入内存单元add |
| 算术运算 | Add, add(R0 + R1 -> R2 -> RAM(add)) | 16 | R0与R1内容**相加**，结果存入R2，并存入内存单元add |
| Sub, add(R0 - R1 -> R2 -> RAM(add)) | 16 | R0与R1内容**相减**，结果存入R2，并存入内存单元add |
| Mul, add(R0 \* R1 -> R2 -> RAM(add)) | 16 | R0与R1内容**相乘**，结果存入R2，并存入内存单元add |
| Div, add(R0 / R1 -> R2 -> RAM(add)) | 16 | R0与R1内容**相除**，结果存入R2，并存入内存单元add |
| Mod, add(R0 % R1 -> R2 -> RAM(add)) | 16 | R0内容对R1**取模**，结果存入R2，并存入内存单元add |
| Rs, add(R0>>1 -> R2 -> RAM(add)) | 16 | R0内容**算术右移**，结果存入R2，并存入内存单元add |
| Ls, add(R0<<1 -> R2 -> RAM(add)) | 16 | R0内容**逻辑左移**，结果存入R2，并存入内存单元add |
| 逻辑运算 | Xor, add(R0 ^ R1 -> R2 -> RAM(add)) | 16 | R0与R1内容**相异或**，结果存入R2，并存入内存单元add |
| And, add(R0 & R1 -> R2 -> RAM(add)) | 16 | R0与R1内容**相与**，结果存入R2，并存入内存单元add |
| Or, add(R0 | R1 -> R2 -> RAM(add)) | 16 | R0与R1内容**相或**，结果存入R2，并存入内存单元add |
| Not, add(!R0 -> R2 -> RAM(add)) | 16 | R0内容**取非**，结果存入R2，并存入内存单元add |
| 跳转 | Jmp, add | 16 | 无条件跳转到指定地址 |
| Jz, add | 16 | R0内容**等于**R1则跳转到指定地址 |
| Jnz, add | 16 | R0内容**不等于**R1则跳转到指定地址 |
| Jl, add | 16 | （有符号数）R0内容**小于**R1则跳转到指定地址 |
| Jg, add | 16 | （有符号数）R0内容**大于**R1则跳转到指定地址 |
| Ja, add | 16 | （无符号数）R0内容**小于**R1则跳转到指定地址 |
| Jc, add | 16 | （无符号数）R0内容**大于**R1则跳转到指定地址 |
| 寄存器  间转移 | Mov, R2, R0 (R2 -> R0) | 8 | 将R2内容存入R0 |
| Mov, R2, R1 (R2 -> R1) | 8 | 将R2内容存入R1 |
| 递归功能 | Ld, SP, add | 16 | 读取内存单元add至栈指针寄存器SP |
| Push (R0 -> RAM(\*SP)) | 8 | 入栈（将R0内容存入RAM(\*SP)） |
| Pop (RAM(\*SP) -> R0) | 8 | 出栈（将add(SP)存入R0） |
| Exec, func | 16 | 执行函数func |
| Ret | 8 | 退出函数 |
| 循环功能 | Ld, CX, add | 16 | 读取内存单元add至循环变量寄存器CX |
| Exec, loop | 8 | 执行循环loop |
| 其他 | Inc, R0 (R0 + 1 -> R0) | 16 | R0内容自增（R0++），并存入内存单元add |
| Hlt | 8 | 停机指令 |