# DataStructure

## DataStructure\Instruction.cs

* Mips246指令c# model

**内部成员变量**

|  |  |  |
| --- | --- | --- |
| 变量名 | 变量类型 | 注释 |
| Mnemonic | Mnemonic | 指令助记符 |
| Machine\_code | BitArray | 指令机器码 |
| Arg1,arg2,arg3 | String | 指令参数 |
| Address | Int | 指令地址 |
| AssemblerTable | Hashtable | 静态汇编表 |
| Regs |  |  |
| R\_type |  |  |
| IJ\_Type |  |  |
| isAlias |  |  |

**构造函数**

|  |  |  |
| --- | --- | --- |
| 函数参数特征 | 参数列表 | 注释 |
| static Instruction() | N/A | 静态构造函数，初始化静态汇编表， |
| Instruction(string mnemonic, string arg1, string arg2, string arg3) | Mnemonic  助记符  Arg1  参数1  Arg2  参数2  Arg3  参数3 | 助记符，参数1，参数2，参数3 |
| Instruction(BitArray machine\_code ,bool isAlias = false) |  |  |

**属性接口**

|  |  |  |
| --- | --- | --- |
| 变量名 | 变量类型 | 注释 |
| Mnemonic | Mnemonic | 指令助记符 |
| Machine\_code | BitArray | 指令机器码 |
| Arg1,arg2,arg3 | String | 指令参数 |
| Address | Int | 指令地址 |

**外部函数接口**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| Validate() | Void | N/A | 实例构造完成后调用此函数进行汇编or反汇编，填充未填充的成员。 |
| ToString() | Void | N/A | 重载，返回指令汇编形式。 |

**内部函数**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| InitBoolArray(string codestring) | BitArray | Codestring  汇编代码字符串 | 初始化bitArray |
| InitAssemblerTable() | Void | N/A | 初始化汇编表 |
| InitRegDic() |  |  |  |
| InitR\_type() |  |  |  |
| InitIJ\_type() |  |  |  |
| ToMachineCode() | Void | N/A | 汇编入口 |
| ToAsmCode() | Void | N/A | 反汇编入口 |
| getImm() |  |  |  |
| getShamt() |  |  |  |
| getAdress() |  |  |  |
| getRs() |  |  |  |
| getRt() |  |  |  |
| getRd() |  |  |  |
| getOP() |  |  |  |
| getfunc() |  |  |  |
| getMnemonic() |  |  |  |
| getDecValueFromBitarray(int begin, int length) |  |  |  |
| setRegMachineCode(int startPosition, string reg) | Void | StartPosition  起始位置  Reg  寄存器 | 将MachineCode第N位起的5位设置成对应的寄存器二进制码。 |
| setShamtMachineCode(int startPosition, string shamt) | Void | Startposition  起始位置  Shamt  Shamt | 将MachineCode第N位起的5位设置成对应的shamt二进制码。 |
| setImmediateMachineCode(int startPosition, string immediate) | Void | Startposition  起始位置  Immediate  立即数 | 将MachineCode第N位起的16位设置成对应的立即数二进制码。 |
| setOffsetMachineCode(int startPosition, string offset) | Void | Startposition  起始位置  Offset  偏移量 | 将Machine第N位起的16位设置成对应的偏移量二进制码。 |
| setAddressMachineCode(string address) | Void | Address  地址标签 | 将Machine第N位起的16位设置成对应的地址二进制码。 |
| OP\_XXX() | Void | N/A | 单条指令汇编处理代码。 |

## DataStructure\AssemblerFac.cs

## DataStructure\AssemblerIns.cs

## DataStructure\DataInstruction.cs

## DataStrucure\FourExp.cs

## DataStructure\FourExpFac.cs

## DataStructure\LabelStack.cs

## DataStructure\RegContent.cs

## DataStructure\VarTable.cs

# Assembler

## Assembler\Assembler.cs

* 汇编器核心代码

**内部成员变量**

|  |  |  |
| --- | --- | --- |
| 变量名 | 变量类型 | 注释 |
| sourceString | List<string> | 源代码行数组 |
| codelist | List<Instruction> | MIPS246指令C#对象数组 |
| sourceList | List<string[]> | 经空格处理后的Token序列数组（每个数组元素代表一条指令，其中的每个string代表一个token） |
| error | AssemblerErrorInfo | 汇编错误 |
| sourcepath | string | 输入路径 |
| outputpath | string | 输出路径 |
| linetable | Hashtable | 行与代码序列映射表 |
| codeindextable | Hashtable | 代码序列与行映射表 |
| addresstable | Hashtable | 地址标签与源代码行 |
| labeltable | Hashtable | 标签与行映射表 |
| rawSource | List<string> |  |
| StartAddress | Int | 其实地址（未使用） |

**构造函数**

|  |  |  |
| --- | --- | --- |
| 函数参数特征 | 参数列表 | 注释 |
| Assembler() | N/A | 默认构造函数 |
| Assembler(string sourcepath, string outputpath) | Sourcepath  源代码路径  Outputpath  输出代码路径 | 根据输入路径和输出路径作为参数，读入源代码，构造汇编器 |
| Assembler(List<string> sourceString) | SourceString  源代码行字符串数组 | 以行字符串作为参数构造汇编器 |
| Assembler(string sourceCode) | SourceCode  源代码裸文本 | 以裸文本作为参数构造汇编器 |

**属性接口**

|  |  |  |
| --- | --- | --- |
| 变量名 | 变量类型 | 注释 |
| SourceList | List<string[]> | 经空格处理后的Token序列数组（每个数组元素代表一条指令，其中的每个string代表一个token） |
| CodeList | List<Instruction> | MIPS246指令C#对象数组 |
| Error | AssemblerErrorInfo | 汇编错误 |
| Linetable | Hashtable | 指令地址 |
| CodeIndexTable | Hashtable | 行与代码序列映射表 |
| RawSource | List<string> |  |

**外部函数接口**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| DoAssemble() | Bool | N/A | 实例构造完成后调用此函数进行汇编or反汇编，填充未填充的成员。 |
| Display(bool isBinary) | Void | isBinary  是否以二进制输出 | 输出至Console  参数 |
| Output(bool isoutputCOE, string outputpath, bool isHEX) | void | isoutputCOE  以COE格式或TXT格式输出  isHEX  以二进制或十六进制输出 | 输出至文件 |
| DisplayError() | Void | N/A | 将汇编错误信息输出至控制台 |

**内部函数**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| LoadFile() | Bool | N/A | 读取文件至sourceList |
| LoadAddress() | Bool | N/A | 填充地址标签与源代码行映射表 |
| CheckWord() | Bool | N/A | 扫描token |
| WriteBackAddress() | Void | n/A | 将绝对地址写回相关指令对象 |
| CalculOffset() | Void | N/A | 计算相对偏移量 |
| InitInstructionAddress() | Void | N/A | 初始化地址表 |
| AssembleInstructions() | Void | N/A | 对每一条指令对象执行汇编过程 |
| SetAddress0(string label) | Void | Label  将label所指向的地址设为0地址 | 设置0地址（未使用） |
| CheckVariableName(string name) | Bool | Name  变量 | 检查变量名是否合法 |
| addAddresstable(string addressname, int address) | Bool | Addressname  地址标签名  Address  指令行 | 将标签名和地址记录在标签与地址行映射表中 |
| CheckRegister(string reg) | Bool | Reg  寄存器名 | 检查寄存器名是否合法 |
| CheckAddress(string addressname) | Bool | Addressname  地址标签名 | 检查地址名是否合法 |
| RemoveComment(string str) | String | Str  源代码 | 过滤代码中的注释 |
| CheckShamt(string i) | Bool | I  Shamt | 检查shamt值是否合法 |
| ConvertImmediate(int i, string str, out int intvalue, bool sign) | Bool | I  行号  Str  待转换的立即数文本  Intvalue  输出参数，转换后的值  Sign  是否为有符号数 | 转换文本至整型 |
| FormatHex(bool[] boolArray) | String | Bool[]  指令串 | 将boolArray转换为十六进制文本 |
| OP\_XXX() | Bool | I  行号 | 单条指令汇编源代码检查函数，并对instruction对象进行填充 |

## Assembler\AssemblerErrorInfo.cs

* 汇编器错误对象

**内部成员变量**

|  |  |  |
| --- | --- | --- |
| 变量名 | 变量类型 | 注释 |
| Line | Int | 错误行 |
| Assemblererror | AssemblerError | 错误类型 |
| description | String | 错误信息输出 |

**构造函数**

|  |  |  |
| --- | --- | --- |
| 函数参数特征 | 参数列表 | 注释 |
| AssemblerErrorInfo(int line, AssemblerError assemblererror) | Line  行号  Assemblererror  错误类型 | N/A |
| AssemblerErrorInfo(int line, AssemblerError assemblererror, string description) | Line  行号  Assemblererror  错误类型  Description  错误信息提示 | N/A |

**外部函数接口**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| ConsoleDisplay() | Void | N/A | 打印错误信息至控制台。 |
| ToString() | Void | N/A | 重载，返回错误信息。 |

# Compiler

# Disassembler

# ResulteComparer

## ResultComparer/ResultComparer

* 运行结果比较器

**外部函数接口**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| Compare(string path1, string path2) | String | Path1  运行结果路径1  Path2  运行结果路径2 | 对两个输出结果的路径进行比较，返回比较结果 |

* **内部函数**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| LoadFile(string path) | List<string> | Path  路径 | 读取文件内容 |

# TestCodeGeneator

## TestCodeGeneator\TestCodeGeneator.cs

* 汇编器核心代码

**内部成员变量**

|  |  |  |
| --- | --- | --- |
| 变量名 | 变量类型 | 注释 |
| Count | Int | 生成数量 |
| seed | int | 随机数种子 |
| codeList | List<Instruction> | 生成的指令数组 |
| r | Random | 随机数生成器 |
| cmdList | List<Mnemonic> | 目标指令数组 |

**构造函数**

|  |  |  |
| --- | --- | --- |
| 函数参数特征 | 参数列表 | 注释 |
| TestCodeGeneator () | N/A | 默认构造函数 |

**属性接口**

|  |  |  |
| --- | --- | --- |
| 变量名 | 变量类型 | 注释 |
| Count | Int | 生成数量 |
| seed | int | 随机数种子 |
| codeList | List<Instruction> | 生成的指令数组 |

**外部函数接口**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| ConfigGeneator(int num, List<Mnemonic> targetCMDList) | Void | Num  指令生成条数  Mnemonic  指令选择范围 | 配置随机代码生成器 |
| Generate() | Void | N/A | 生成随机指令 |

**内部函数**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 返回类型 | 参数列表 | 注释 |
| GenerateCMD(int cmdIndex) | Instruction | Int  生成指令序列 | 生成单条指令 |
| GenerateReg() | string | N/A | 随机生成寄存器 |
| GenerateImmediate(bool isSign) | String | Issign  是否为有符号数 | 随机生成立即数 |
| GenerateOffset() | String | n/A | 随机生成偏移量 |
| GenerateShamt() | String | N/A | 随机生成Shamt |
| GenerateAddress () | String | N/A | 随机生成地址 |

# Web