

课程综合设计 1-MIPS 汇编器设计

QIndomitable Mips Assembler(QMA)

王晓强

目 录

汇编功能

支持指令

支持伪指令

其他数据定义方式

反汇编功能

示例 ASM

汇编功能

支持如下指令:

add

addu

and

clo

clz

div

divu

mul

mult

multu

madd

maddu

msub

nor

or

sll

sllv

sra

srav

srl

srlv

sub

subu

xor

slt

sltu

mfhi

mflo

mthi

mtlo

mfc0

mfc1

mtc0

mtc1

jalr

jr

addi

addiu

andi

ori

lui

xori

slti

sltiu

beq

bgez

bgezal

bgtz

blez

bltza

bltz

bne

lb

lbu

lh

lhu

lw

sb

sh

sw

j

jal

支持如下伪指令:

地址加载

la

数据定义, 代码段起点

.text

.data <ADDRESS>

.word

.half

.byte

// <ADDRESS>,数据均支持 10 进制, 0x/0X 前导 16 进制,

// 0 前导 8 进制;

数据定义, 代码段起点其他定义方式:

代码段起点:

#baseAddr XXXXXXXXX; // 16 进制, ByteAddress, 无<冒号>

...

//code segment

...

#DataAddre: XXXXXXXXX; // 16 进制, ByteAddress, 有<冒号>

// 无前导, 0x/0X 前导 16 进制, 只解析 16 进制

// ASCII 定义,单引号或双引号表示的字符串面值,不解析转义字符
// 可以出现标号, 支持 la 加载, 但暂时不支持 Macro 定义
dd //32 位
dw //16 位
db //8 位
// 缺省 32 位
// 以上定义方式从定义开始, 作用范围到<分号>结束, 中间允许用
// <逗号>分割, 超过定义宽度的数据会被截断, 整个数据段保证 4 字
// 节对齐, 不足用 0 填充;

反汇编功能

只支持指令的反汇编, COE 加载, BIN 加载

不支持伪指令和任何数据定义的反汇编

示例 ASM

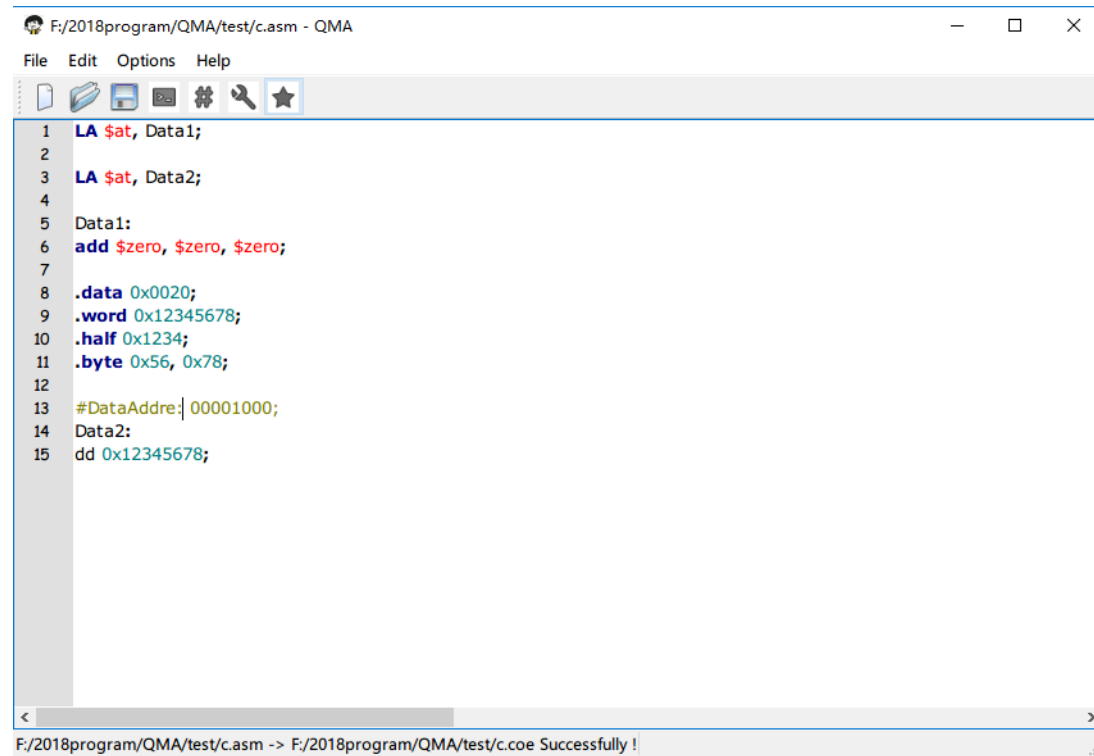
./sample/a.asm

```
F:/2018program/QMA/test/a.asm - QMA
File Edit Options Help
245 add $t1, $zero, $t6;
246 add $t1, $t1, $v0;
247 label_9:
248 lw $a1, 0($v1);
249 add $t3, $a1, $a1;
250 add $t3, $t3, $t3;
251 sw $t3, 0($v1);
252 sw $a2, 4($v1);
253 j label_10;
254
255 #DataAddr: 00001000; //?????? 00001000H ?????
256 Data1: //??? 1???
257 dd FFFFFFF0, 000002AB, 80000000, 0000003F, 00000001, FFFF0000, 0000FFFF, 80000000,
258 00000000, 11111111, 22222222, 33333333, 44444444, 55555555, 66666666, 77777777,
259 88888888, 99999999, AAAAAAAAAA, BBBBBBBB, CCCCCCCC, DDDDDDDD, EEEEEEEE, FFFFFFFF;
260
261 db 0x55,0x56,0x57,0x58; //dw??? ?????0x41,0x42,0x1234?
262 dw 0x5678, 0x1234; //dw??? ?????0x55??00001060
263
264 dd 0x12345678; //dw ?????0x12345678?00001068
265
266 #DataAddr: 00002000; //?????? 00002000H ?????
267 Data2: //??? 2???
268 557EF7E0, D7BDFBD9, D7BDFBD9, DFCFFCFB, DFCFBFFF, F7F3DFFF, FFFDF3D, FFF9DB9,
269 FFFFCF, DFCFFCFB, DFCFBFFF, D7BDFBD9, D7BDFBD9, D7BDFBD9, FFF9DB9, 007E0FFF
0 Warning ! 0 Error ! Enjoy Yourself !
```

./sample/b.asm

```
F:/2018program/QMA/test/b.asm - QMA
File Edit Options Help
113 add $s1, $s1, $t6;
114 and $s1, $s1, $s4;
115 add $t1, $t1, $v0;
116 beq $t1, $at, label_8;
117 j label_9;
118 label_8:
119 add $t1, $zero, $t6;
120 add $t1, $t1, $v0;
121 label_9:
122 lw $a1, 0($v1);
123 add $t3, $a1, $a1;
124 add $t3, $t3, $t3;
125 sw $t3, 0($v1);
126 sw $a2, 4($v1);
127 j label_10;
128
129 .data 0x0200;
130 .word 0xf0000000, 0x000002AB, 0x80000000, 0x0000003F, 0x00000001, 0xFFFF0000, 0x0000FFFF, 0x80000000;
131
132 .word 0x00000000, 0x11111111, 0x22222222, 0x33333333, 0x44444444, 0x55555555, 0x66666666, 0x77777777;
133 .word 0x88888888, 0x99999999, 0xaaaaaaaa, 0xbbbbbbbb, 0xcccccccc, 0xdddddddd, 0xeeeeeeee, 0xffffffff;
134 .word 0x557EF7E0, 0xD7BDFBD9, 0xD7BDFBD9, 0xDFCFFCFB, 0DFCFBFFF, 0xF7F3DFFF, 0xFFFFDF3D, 0xFFFF9DB9;
135 .word 0xFFFFFCF, 0DFCFFCFB, 0DFCFBFFF, 0xD7B9FFF, 0xD7BDFBD9, 0xD7BDFBD9, 0xFFFF07E0, 0x007E0FFF;
136 .word 0x03bdf020, 0x03def820, 0x08002300, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000;
137
0 Warning ! 0 Error ! Enjoy Yourself !
```


./sample/c.asm



The screenshot shows a window titled "F:/2018program/QMA/test/c.asm - QMA". The menu bar includes "File", "Edit", "Options", and "Help". The toolbar contains icons for file operations (new, open, save, print), a search icon, and a star icon. The main text area displays assembly code with line numbers 1 through 15. The code includes instructions like "LA \$at, Data1;", "LA \$at, Data2;", "add \$zero, \$zero, \$zero;", and data definitions like ".data 0x0020;", ".word 0x12345678;", ".half 0x1234;", ".byte 0x56, 0x78;". A comment "#DataAddre: 00001000;" is also present. The status bar at the bottom shows the command "F:/2018program/QMA/test/c.asm -> F:/2018program/QMA/test/c.coe Successfully !" and a small icon on the right.

```
1 LA $at, Data1;
2
3 LA $at, Data2;
4
5 Data1:
6 add $zero, $zero, $zero;
7
8 .data 0x0020;
9 .word 0x12345678;
10 .half 0x1234;
11 .byte 0x56, 0x78;
12
13 #DataAddre: 00001000;
14 Data2:
15 dd 0x12345678;
```

F:/2018program/QMA/test/c.asm -> F:/2018program/QMA/test/c.coe Successfully !