

# 五、目标代码生成

## (1. RISC-V 概述)

魏恒峰

hfwei@nju.edu.cn

2023 年 06 月 02 日



RISC: Reduced Instruction Set Computer (精简指令集计算机)



V: Five

CISC: Complex Instruction Set Computer (复杂指令集计算机)

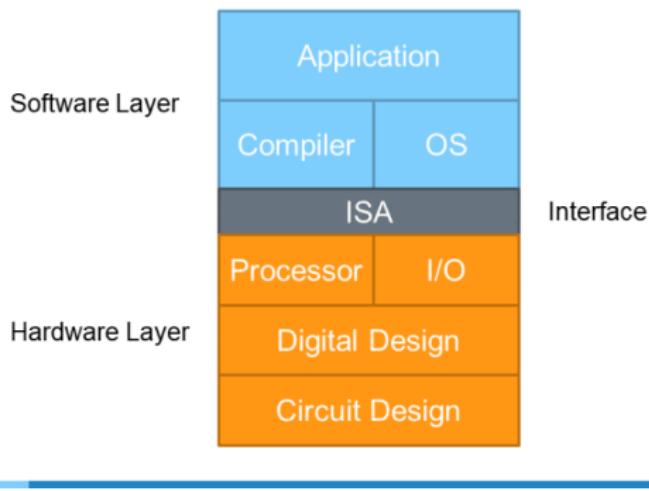


RISC: Each instruction performs only one function.

CISC: SUBL val, %eax ; %eax <- %eax - val)



# ISA (Instruction Set Architecture) as the Software/Hardware Interface



In general, an ISA defines the supported [instructions](#), [data types](#), [registers](#), the hardware support for managing [main memory](#), fundamental features (such as the [memory consistency](#), [addressing modes](#), [virtual memory](#)), and the [input/output](#) model of a family of implementations of the ISA.

# 2017 ACM Turing Award



“Hennessy and Patterson created a systematic and quantitative approach to designing faster, lower power, and **reduced instruction set computer (RISC)** microprocessors.”



[About RISC-V](#)[Membership](#)[RISC-V Exchange](#)[Technical](#)[News & Events](#)[Community](#)

## Announcements

### First-ever RISC-V Summit Europe Will Demonstrate Technical and Commercial Momentum Across Industries

The Barcelona RISC-V Summit from June 5-9 will focus on industries such as Automotive, High-Performance Compute/Data Center, and Security; Call for Submissions and Sponsorships are now open.

[READ MORE](#)

## Join RISC-V International



RISC-V International comprises a large member organization building the first open, collaborative community of software and hardware innovators powering innovation at the edge forward. Through various events and workshops, RISC-V International is changing the way the industry works together and collaborates – creating a new kind of open hardware and software ecosystem. Become a member today and help pioneer the industry's future!

## Premier Members



成为资本 CHENGING CAPITAL



# 2019 年国际芯片大会



“RISC-V 很可能发展成为世界主流 CPU 之一，在 CPU 领域形成 Intel、ARM、RISC-V 三分天下的格局。” — 中国工程院院士 倪光南



## 新长征路上的摇滚

“万里长征第一步 ...”

# RISC-V Assembly Language Programming

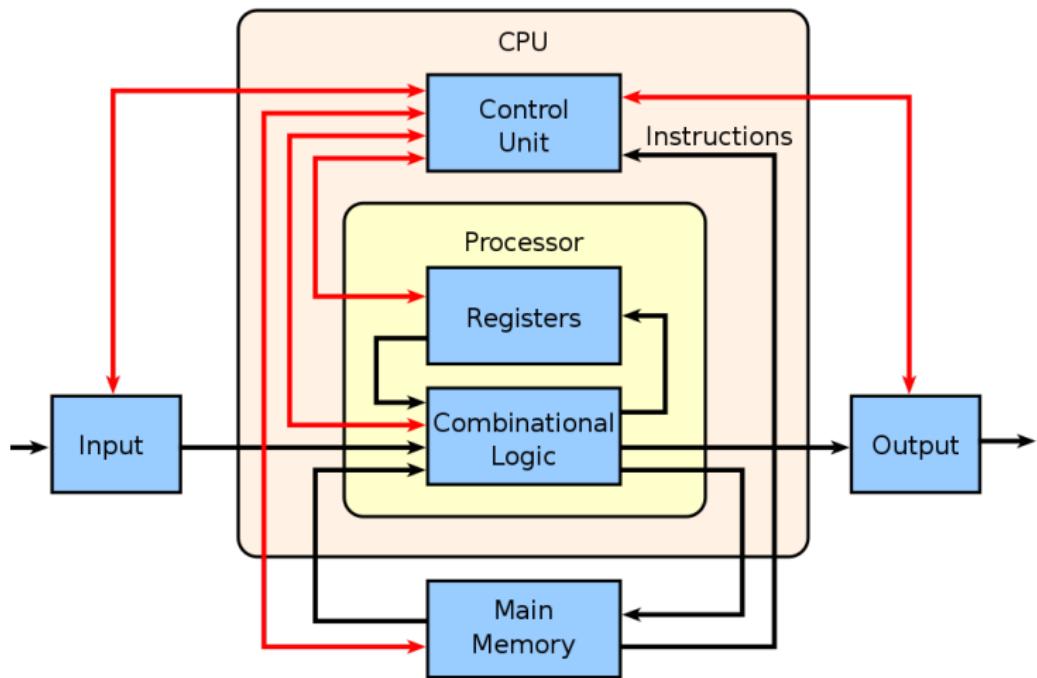
Using the ESP32-C3 and QEMU



Warren Gay



CONCEPT & CODE



Registers (zero, pc)

Memory (Data, Instructions)

I/O

add.asm

addi.asm

add-sub.asm

## ecall.asm

## data.asm

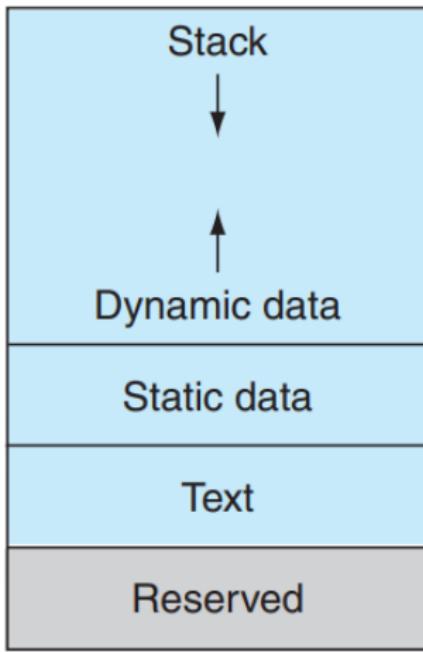
\$sp → 7fff fffc<sub>hex</sub>

\$gp → 1000 8000<sub>hex</sub>

1000 0000<sub>hex</sub>

pc → 0040 0000<sub>hex</sub>

0



array.asm

## branch-max.asm

## array-index-bit.asm

## proc-max.asm

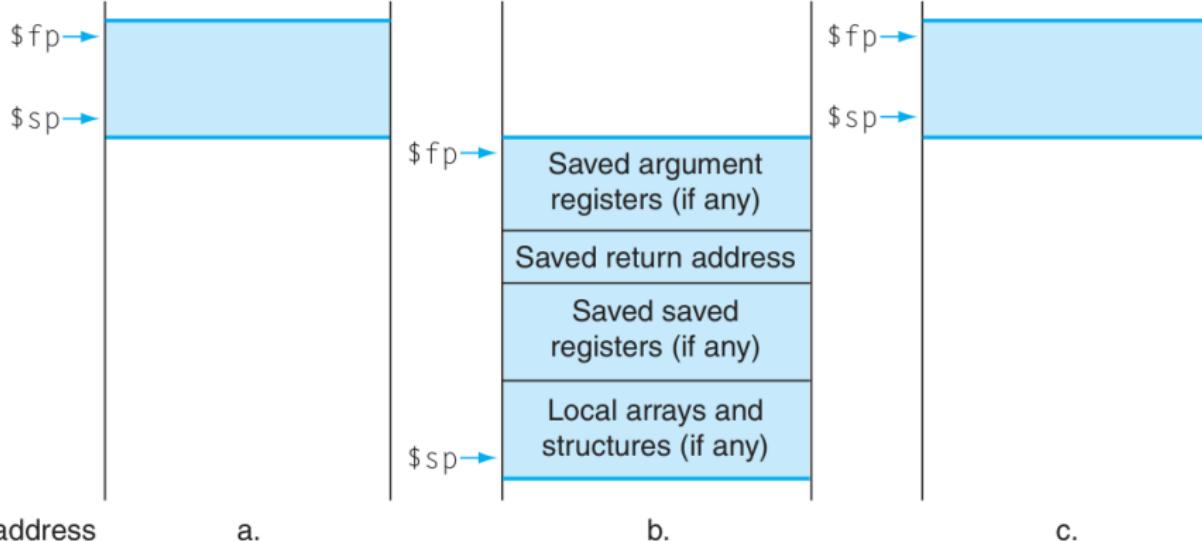
Registers	Symbolic names	Description
x0	zero	Hardwired zero
x1	ra	Return address
x2	sp	Stack pointer
x3	gp	Global pointer
x4	tp	Thread pointer
x5-x7	t0-t2	Temporary registers
x8-x9	s0-s1	Saved registers
x10-x11	a0-a1	Function arguments and return values
x12-x17	a2-a7	Function arguments
x18-x27	s2-s11	Saved registers
x28-x31	t3-t6	Temporary registers

## proc-fact.asm

## RISC-V 调用约定 (Calling Convention)

Registers	Symbolic names	Description	Saver
x0	zero	Hardwired zero	—
x1	ra	Return address	Caller
x2	sp	Stack pointer	Callee
x3	gp	Global pointer	—
x4	tp	Thread pointer	—
x5-x7	t0-t2	Temporary registers	Caller
x8-x9	s0-s1	Saved registers	Callee
x10-x11	a0-a1	Function arguments and return values	Caller
x12-x17	a2-a7	Function arguments	Caller
x18-x27	s2-s11	Saved registers	Callee
x28-x31	t3-t6	Temporary registers	Caller

High address



Low address

a.

b.

c.

bubblesort.asm

bubblesort @ CompilerExplorer

```
book-cover.s •
Users > edson > Desktop > book-cover.s
1   .file  "book-cover.c"
2   .option nopic
3   .attribute arch, "rv32i2p0"
4   .attribute unaligned_access, 0
5   .attribute stack_align, 16
6   .text
7   .align 2
8   compute_the_answer_to_the_ultimate_question_of_life_the_universe_and_everything:
9     li a0,42
10    ret
11    .align 2
12    .globl do_something_1000_times
13    .type do_something_1000_times, @function
14 do_something_1000_times:
15    addi sp,sp,-16
16    sw $0,8(sp)
17    sw ra,12(sp)
18    li $0,1000
19 .L6:
20    addi $0,$0,-1
21    call do_something
22    bne $0,zero,.L6
23    lw ra,12(sp)
24    lw $0,8(sp)
25    addi sp,sp,16
26    jr ra
27    .section .rodata.str1.4,"aMS",@progbits,1
28    .align 2
29 .LC0:
30    .ascii "There are 10 types of people in this world "
31    .asciz "those who understand binary and those who don't"
32    .align 2
33 .LC1:
34    .string "Assembly language you must learn!"
35    .align 2
36 .LC2:
37    .ascii "The Unicamp CS course was created in 1969 - "
38    .asciz "The first one in Brazil!"
39 .LC3:
40    .byte 78, 105, 99, 101, 33, 32, 89, 111, 117, 32, 107
41    .byte 110, 111, 119, 32, 65, 83, 67, 73, 73, 33, 0
```

# An Introduction to Assembly Programming with RISC-V

An Introduction to Assembly Programming with RISC-V



Gabriel Hjort Blindell

# Instruction Selection

Principles, Methods, and Applications

 Springer

Sebastian Hack

Register Allocation for  
Programs in SSA Form



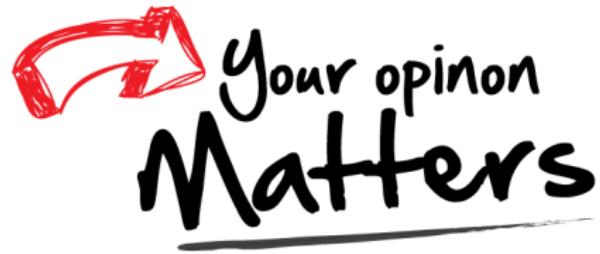
universitätsverlag karlsruhe

JUST  
HAVE  
FUN



“因为我的病就是没有感觉”

# Thank You!



Office 926

hfwei@nju.edu.cn