	文档	名		实验指导书_lab2.1				
	版本	号		0.4				
Ŧ**\'\\ =	创建。	人	计算机组成原理教学组					
更新记录	创建日	期		2022/1/1				
更新历史								
序号	更新日期	更新人	版本号	更新内容				
1	2022/1/1	陈颖琪	0.1	RISC-V CPU模型虚拟仿真				
2	2022/2/24	陈颖琪	0.2	仿真平台说明,添加ripes				
3	2022/3/29	陈颖琪	0.3	修正了表1中的两处错误				
4	2023/3/14	陈颖琪	0.4	仿真平台3略				

文档错误反馈:

本文档出现错误请联系:

Yingqichen@sjtu.edu.cn

实验二 之 RISC-V CPU 虚拟仿真部分

1、实验目的

- 1、熟悉指令和汇编代码。
- 2、产生指令存储器初始值文件。

2、实验要求

2.1 仿真平台

仿真软件 1: Ripes v2.2.4

RISC-V 32/64b 单周期及流水线架构 CPU 模型仿真软件,下载链接:

https://jbox.sjtu.edu.cn/l/B1IA1R

本部分实验可以此平台为主。

仿真平台 2: WebRISCV

RISC_V32/64b 五级流水线架构 CPU 模型在线仿真平台,主要用于理解流水线执行过程。 在后续的 lab5 实验中也会使用。

平台链接: http://10.119.1.50:81/ (校内网,校外访问需连接交大 VPN)

或 https://webriscv.dii.unisi.it/index.php (备用)

两个仿真平台均可使用。

2.2 仿真测试代码段

本次实验用到如下汇编代码段,可直接于仿真平台1、2输入用,供参考。

lui x10, 0

ori x4, x10, 0

addi x25, x0, 1

addi x26, x0, 2

addi x27, x0, 3

addi x28, x0, 4

sw x25, 0(x4)

sw x26, 4(x4)

sw x27, 8(x4)

sw x28, 12(x4)

addi x5, x0, 4

```
call:
jal sum
sw x12, 0(x4)
lw x19, 0(x4)
sub x18, x19, x12
addi x5, x0, 3
loop2:
addi x5, x5, -1
ori x18, x5, -1
xori x18, x18, 1365
addi x19, x0, -1
andi x20, x19, -1
or x16, x20, x19
xor x18, x20, x19
and x17, x20, x16
beq x5, x0, shift
j loop2
shift:
addi x5, x0, -1
slli x18, x5, 15
slli x18, x18, 16
srai x18, x18, 16
srli x18, x18, 15
fi:
j fi
sum:
add x18, x0, x0
loop:
lw x19, 0(x4)
addi x4, x4, 4
add x18, x18, x19
addi x5, x5, -1
bne x5, x0, loop
slli x12, x18, 0
jr ra
```

2.3 虚拟仿真步骤与要求

补全表格理解代码段含义。

- 1、选择仿真平台1(也可选择其他平台,做对比)输入上述测试代码段;
- 2、设置单周期 CPU 模式;
- 3、通过单步执行仿真理解代码执行的操作,得出其每条指令机器码,补全表 1 中对指令的描述,或者指令执行后的输出,填入表格 1,补全后四列(红色字体)内容;
- 4、提交完整表格。

表 1 测试用代码段功能描述表

		T	T	用作的权力	1		T
Ad	标	仿真平台 2	仿真平台	机器码	机器码	指	描述
dre	号	输入代码(这	2 显示代	(BIN)	(HEX)	*	
ss(样输入才不	码			类	
HE		会出错)				型	
X)							
0		lui x10, 0	lui a0, 0		00000537	U	#initialize x10 =base
			,	1.1E+23			address 0
4		ori x4, x10, 0	oritp, a0, 0		00056213	1	#x4<- base address
-		011 14, 110, 0	011(p, 00, 0	1E+17	00030213	'	x10 + offset 0 =0
		- dd:2E0 1	- dd: -0		00100-03	_	
8		addi x25, x0, 1	addi s9,	1E+30	00100c93	R	#initialize x25 = 1
			x0, 1			_	
С		addi x26, x0, 2	addi s10,	1.1E+21	00200d13	R	
			x0, 2				
10		addi x27, x0, 3	addi s11,	1.11E+3	00300d93	R	
			x0, 3	1			
14		addi x28, x0, 4	addi t3,	1.11E+3	00400e13	R	
			x0, 4	1			
18		sw x25, 0(x4)	sw s9,	1.01E+3	01922023	S	#[0] = 1
			0(tp)	0			
1c		sw x26, 4(x4)	sw s10,	1.11E+3	01a22223	S	# [4] = 2
			4(tp)	1			
20		sw x27, 8(x4)	sw s11,	1.11E+3	01b22423	S	
		, , , ,	8(tp)	1			
24		sw x28, 12(x4)	sw t3,	_	01c22623	S	
		3W X20, 12(X+)	12(tp)	1E+24	01022023		
28		addi x5, x0, 4	addi t0,		00400293	R	# x5 = 4,循环次数
20		audi x5, x0, 4	•	1E+24	00400293	N	# X3 - 4, //目が八致
2 -	C-II	Calledal access	x0, 4		05 4000 - f		# # f + i
2c	Call	Call: jal sum	jalra, 128	1E+24	054000ef	I	# call function sum
	:					_	跳转到 pc = 80
30		sw x12, 0(x4)	sw a2,	1.01E+1	00c22023	S	#[16] <- 0x0000000a
			0(tp)	7			(x12=0x0000000a)
34		lw x19, 0(x4)	lw s3,	1.11E+3	00022983	I	#x19<-[16] (0x10)
			0(tp)	1			([16]=0x0000000a)
38		sub x18, x19,	sub s2, s3,	1.11E+3	40c98933	R	#x18= 0
		x12	a2	1			
3c		addi x5, x0, 3	addi t0,	1.11E+2	00300293	R	#x5=3
			x0, 3	3			
40	loo	loop2:addi x5,	addi t0, t0,	45.07	Fff28293	R	
	p2:	x5, -1	-1	1E+24			
44	<u> </u>	ori x18, x5, -1	ori s2, t0, -		Fff2e913	ı	#x18= 0xfffffff , (x18
			1	1E+30			= x5 or 12bit 立即数
		1	-				- NJ OI 120IL 立时双

							有符号扩展 0xffffffff)
48		xori x18, x18,	xori s2, s2,	1.11E+2	55594913	R	#X18=0xfffffaaa
		1365	1365	3			
4c		addi x19, x0, -		110111	Fff00993	R	#X19=0xfffffff
		1	x0, -1	1			
50		andi x20, x19,	andi s4,		Fff9fa13	R	#X20=0xffffffff
		-1	s3, -1	1E+11			(X20=0xfffffff and
							0xfffffff)
54		or x16, x20,	or a6, s4,	15.17	013a6833	R	#X16=0xffffffff
		x19	s3	1E+17			
58		xor x18, x20,	xor s2, s4,	1E+22	013a4933	R	#X18=0
		x19	s3	16+22			
5c		and x17, x20,	and a7, s4,	1E+24	010a78b3	R	#X17=0xffffffff
		x16	a6	11.124			
60		beq x5, x0,	beq t0, x0,		00028463	S	#Ifx5 = 0
		shift	104	1.11E+3		В	Goto shift after
				1			finished loop2 4
							times, goto pc= 68
64		j loop2	jal x0, 64	1.11E+3	Fddff06f	I	#Loop Loop2 for 4
	_	_		1			times, goto pc=40
68	shift	shift:addi x5,	addi t0,	1E+19	Fff00293	R	#X5=0xffffffff
	:	x0, -1	x0, -1		_		
6c		slli x18, x5, 15	slli s2, t0, 15	1E+15	00f29913	I	#X18=0xffff8000
70		slli x18, x18,	slli s2, s2,	1.1E+23	01091913	I	#X18=0x80000000
		16	16	1.11.723			
74		srai x18, x18,	srai s2, s2,	1E+17	41095913	I	#X18=0xffff8000
		16	16	10.11			
78		srli x18, x18,	srli s2, s2,	1E+30	00f95913	1	#X18=0x0001ffff
		15	15	12.00			
7c	finis	finish:j finish	jal x0, 124	1.1E+21	0000006f	I	#Endhere
	h:						
80	sum	sum:add x18,	add s2, x0,	1.11E+3	00000933	R	#X18 = 0x0001ffff
	:	x0, x0	x0	1			
84	loo	loop:lw x19,	lw s3,	1.11E+3	00022983	I	#X19 <- [x4]
	p:	0(x4)	0(tp)	1 015.0	00400040		
88		addi x4, x4,4	additp,	1.01E+3	00420213	R	#x4 <-(x4+4)
0.0		add v10 v10	tp,4	1 115+2	01200022	D	#V10= v10 + [v4]
8c		add x18, x18, x19	add s2, s2, s3	1.11E+3 1	01390933	R	#X18= x18 + [x4], X18= 0x0001ffff
90		addi x5, x5, -1	addi t0, t0,	1.11E+3	Fff28293	R	#x5 <- (x5-1),循环次
90		auui x3, x3, -1	-1	1.116+3	11120233	, n	数-1
94		bne x5, x0,	bne t0, x0,		Fe0298e3	S	#loop 循环累加
J 7		loop	132	1E+24	1023003	В	次,结果存于: x18
		,00p	132		İ		以, 和水(丁): X10

98	slli x12, x18, 0	slli a2, s2,			00091613	1	#X12<- x18 , X12 =
		0		1E+24			0x0000000a,函数调
							用结果存于: x12
9c	Jr ra	jalr	х0,	1E+24	00008067	1	#函数 sum 调用返回,
		0(ra)		10+24			回到 pc = [ra]

3 指令存储器 IP 例化初始化文件产生

完成表 1 后,即可产生 vivado FPGA 开发平台指令存储器 IP 例化时的初始化文件: sc_irom.coe。以便为后续 CPU 核设计提供必要的支持。该文件为文本格式,但是以 coe 作为后缀。格式如下。第一行表示机器码是 16 进制格式。第二行等号后开始列出代码段的机器码,可以用空格或者回车符分隔每条指令最终以分号(半角字符)结束。

memory_initialization_radix=16; memory_initialization_vector=00000537 00056213

00100c93

0

0

0

0

0

;

将前面补全的表 1 中的 16 进制机器码逐行写入, 存为 sc_irom.coe。以备后续实验使用。你也可以通过其他汇编器产生机器码后自己编写小程序自动存为该格式文件。

4 思考

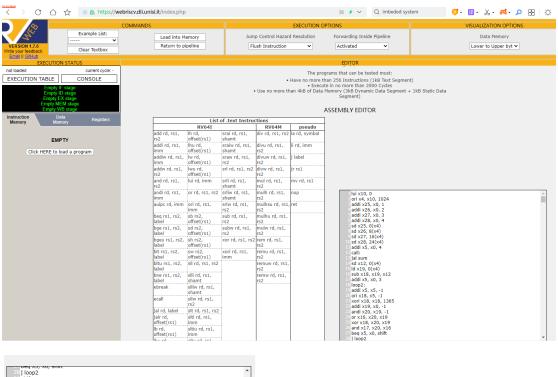
以上代码段最开始将数据 1, 2, 3, 4 写到了哪些存储器地址单元?可以查看 memory 页面,找到被更新的单元,作截图。

Ripes 软件中是将.data 段起始地址默认设定为 0x10000000 的。如果用 Ripes 来仿真,为了便于在.Data 段直接观测,如何修改测试程序的前两句,才可以将 1,2,3,4 写到 0x10000000 为基址的.Data 段起始位置?修改后完成仿真,查看 memory 页面,找到.data 段的起始位置,看看是否被改写?从 0x10000000 开始的几个单元是否被更新?理解对应的语句功能含义。

如果使用仿真平台 2 完成以上仿真的话直接用该代码段是否可以?提示:程序加载与仿真界面截图,如图 2 和图 3。如果在仿真平台 2 上进行仿真,第 2 句可改为 ori x4, x10, 1024,其他不变。因为该平台数据存储器(data 段)设置的偏移量为地址 1024 处开始。为便于在界面直接观察数据存储器内的变化结果,可以做此修改。

5 附录

仿真平台 2 测试程序的加载界面截图示意如图 2、图 3 所示,供参考。



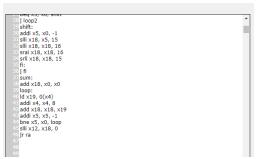
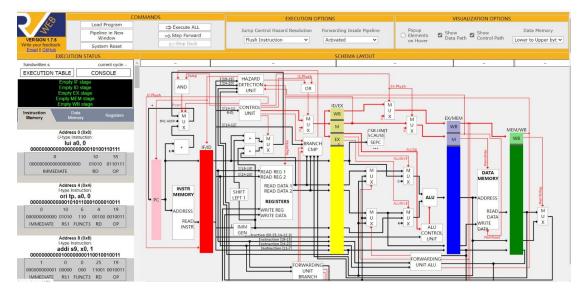


图 2



仿真平台2测试程序加载后的指令和机器码截图,供参考。

Address 0 (0x0) U-type Instruction: lui a0, 0 000000000000000000000001010011	10111 000000	Address 16 (0 I-type Instruct addi s11, x	ion: 0, 3	00000	S-type I	s 32 (0x20 Instruction 11, 8(tp) 0001001	:	00011
0 10	55 3	0 0	27 19	8	27	4	2	35
		•		0000000	01000 11011	00100	010	0100011
IMMEDIATE RD			11011 0010011	IMMED	IATE RS2	RS1 FU	JNCT3	OP
IMMEDIATE RD	OP	E RS1 FUNC	T3 RD OP					
Address 4 (0x4)		A -d-d 20 (0	4.45		Addres	s 36 (0x24	1)	
I-type Instruction:		Address 20 (0 I-type Instruct				Instruction		
ori tp, a0, 0		addi t3, x0			sw t3	3, 12(tp)		
0000000000000101011000100001	0011 0000000	001000000000		00000	00011100001	0001001	10001	00011
0 10 6 4	19 4	0 0	28 19	12	28	4	2	35
000000000000 01010 110 00100 0	0000000001	100 00000 000	11100 0010011	0000000	01100 11100	00100	010	0100011
IMMEDIATE RS1 FUNCT3 RD				IMMED	IATE RS2	RS1 FL	JNCT3	OP
IMINIEDIATE KST FONCTS KD	OP IMMEDIAT	E KST FUNC	IS KD OP					
Address 8 (0x8) I-type Instruction: addi s9, x0, 1		Address 24 (0 S-type Instruc sw s9, 0(1	ion: p)	00000	Address 40 (0x28) I-type Instruction: addi t0, x0, 4			10011
0000000000010000000011001001		110010010001			0000100000			
1 0 0 25	19 0	25 4	2 35	4	0	0	5	19
000000000001 00000 000 11001 0	0000000000	000 11001 00100	010 0100011		00100 00000			0010011
IMMEDIATE RS1 FUNCT3 RD	OP IMMEDIAT	E RS2 RS1	FUNCT3 OP	IMMED	IATE RS1	FUNCT3	RD	OP
Address 12 (0xc) I-type Instruction: addi s10, x0, 2 0000000001010000000011010001	10011 000000	Address 28 (0 S-type Instruc SW s10, 4(ion: tp)	Address 44 (0x2c) UJ-type Instruction: jal ra, 128 000001010100000000000000011101111			01111	
2 0 0 26				1	42		1	111
2 0 0 20		26 4	2 35	000000	000000000101	010 0	0001	1101111
00000000010 00000 000 11010 0	00000000	100 11010 00100		000000	ADDRESS	1010 0	RD	OP
IMMEDIATE RS1 FUNCT3 RD	OP IMMEDIAT	E RS2 RS1	FUNCT3 OP		ADDRESS		KD.	UP
Address 48 (0x30) S-type Instruction:		Address 64 (0 I-type Instruct				80 (0x50) struction:		
sw a2, 0(tp) 00000000110000100010000001	00011 1111111	addi t0, t0 1111100101000		11111	andi s4 11111111001	1 , s3, -1 11111010	000010	0011
	00011 1111111 35 -1	addi t0, t0 1111100101000 5 0		11111			20	0 011
000000001100001000100000001 0 12 4 2	35 -1	1111100101000 5 0	001010010011 5 19	-1	11111111001 19	11111010 7	20	19
000000001100001000100000001 0 12 4 2 0000000000000 01100 00100 010	35 -1 0100011 111111111	1111100101000 5 0 111 00101 000	001010010011 5 19 00101 0010011	-1 11111111	11111111001 19 1111 10011	11111010 7 111 1	20 0100 0	19 010011
000000001100001000100000001 0 12 4 2	35 -1	1111100101000 5 0 111 00101 000	001010010011 5 19 00101 0010011	-1	11111111001 19 1111 10011	11111010 7 111 1	20	19
000000001100001000100000001 0 12 4 2 0000000000000 01100 00100 010	35 0100011 OP IIII111111111111111111111111111111111	1111100101000 5 0 111 00101 000	001010010011 5 19 00101 0010011 T3 RD OP x44) ion: -1	-1 11111111 IMMEDIA	11111111001 19 11111 10011 ATE RS1 F Address R-type In	7 111 1 FUNCT3 84 (0x54) struction: s4, s3	20 0100 0 RD	19 0010011 OP
000000001100001000100000001 0 12 4 2 000000000000 01100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: Iw s3, 0(tp)	35 0100011 OP IIII111111111111111111111111111111111	1111100101000 5 0 111 00101 000 TE RS1 FUNC Address 68 (0 1-type Instruct ori s2, t0,	001010010011 5 19 00101 0010011 T3 RD OP x44) ion: -1	-1 11111111 IMMEDIA	11111111001 19 11111 10011 ATE RS1 F Address R-type In or a6,	7 111 1 FUNCT3 84 (0x54) struction: s4, s3	20 0100 0 RD	19 0010011 OP
000000001100001000100000001 0 12 4 2 00000000000 01100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction:	35 -1 0100011 0P IIIIIIIIIIIIIIIIIIIIIIIIIIIII	1111100101000 5 0 11 00101 000 E RS1 FUNC Address 68 (0 1-type Instruct ori s2, t0, 1111100101110 5 6	001010010011 5 19 00101 0010011 T3 RD OP x44) ion: -1 100100010011 18 19	-1 11111111 IMMEDIA	11111111001 19 1111 10011 ATE RS1 F Address R-type In or a6, 00100111010	7 111 1 EUNCT3 84 (0x54) struction: s4, s3 00110100 6	20 0100 0 RD	19 0010011 OP 0011
000000001100001000100000001 0 12 4 2 0000000000000001100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: Iw s3, 0(tp) 000000000000000010001010101100 0 4 2 19 00000000000000000000 010 10011	00011 1111111 3 -1 0000011 111111111	1111100101000 5 0 111 00101 000 E RS1 FUNC Address 68 (0 I-type Instruct ori s2, t0, 1111100101110 5 6 111 00101 110	001010010011 5 19 00101 0010011 73 RD OP x44) on: -1 100100010011 18 19 10010 0010011	000000 0 0000000 1	11111111001 19 11111 10011 ATE RS1 F Address R-type In Or a6, 00100111010 19 20 10011 10100	7 111 1 EUNCT3 84 (0x54) struction: s4, s3 00110100 6	20 0100 0 RD	19 0010011 OP 0011 51
000000001100001000100000001 0 12 4 2 0000000000000001100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: Iw s3, 0(tp) 000000000000000001000101001100 0 4 2 19 00000000000000000000 010 10011	35 -1 0100011 0P IIIIIIIIIIIIIIIIIIIIIIIIIIIII	1111100101000 5 0 111 00101 000 E RS1 FUNC Address 68 (0 I-type Instruct ori s2, t0, 1111100101110 5 6 111 00101 110	001010010011 5 19 00101 0010011 73 RD OP x44) on: -1 100100010011 18 19 10010 0010011	000000 0 0000000 1	11111111001 19 11111 10011 ATE RS1 F Address R-type In Or a6, 00100111010 19 20 10011 10100	11111010 7 111 1 SUNCT3 84 (0x54) astruction: s4, s3 00110100 6 110 1	20 0100 0 RD 000011 16	19 0010011 OP 0011 51 0110011
000000001100001000100000001 0 12 4 2 00000000000000001100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: Iw s3, 0(tp) 0000000000000000100010010100100 0 4 2 19 00000000000000000000100 010 100110	00011 11111111 3 -1 000011 OP IMMEDIAT	1111100101000 5 0 111 00101 000 E RS1 FUNC Address 68 (0 I-type Instruct ori s2, t0, 1111100101110 5 6 111 00101 110	001010010011 5 19 00101 0010011 T3 RD OP x44) ion: -1 100100010011 18 19 10010 0010011 T3 RD OP x48) ion: 1365	000000 0 0000000 1 FUNCT7	11111111001 19 1111 10011 ATE RS1 F Address R-type in or a6, 00100111010 19 20 10011 10100 RS2 RS1 I	7 1111 1 15UNCT3 84 (0x54) 854, s3 10110100 6 110 1 1FUNCT3 88 (0x58) 88 (0x58) 8struction: 9, s4, s3	20 0100 0 RD 000011 16 10000 0 RD	19 0010011 OP 0011 51 0110011 OP
000000001100001000100000001 0 12 4 2 0000000000000001100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: IW \$3, 0(tp) 00000000000000000100010001000100 0 4 2 19 0000000000000000000100 100 10011 IMMEDIATE RS1 FUNCT3 RD Address 56 (0x38) R-type Instruction: sub \$2, \$3, \$2 0100000011001100010001001001	35 -1 1111111111 OP INMEDIAT 00011 11111111 3 -1 0000011	1111100101000 5 0 111 00101 000 E RS1 FUNC Address 68 (6 1-type Instruct or s2, t0, 1111100101 110 5 6 111 00101 110 E RS1 FUNC Address 72 (6 1-type Instruct xori s2, s2,	001010010011 5 19 00101 0010011 73 RD OP x44) on: -1 100100010011 18 19 10010 0010011 73 RD OP x48) on: 1365	000000 0 0000000 1 FUNCT7	11111111001 19 11111 10011 ATE RS1 F Address R-type In or a6, 00100111010 19 20 10011 10100 RS2 RS1 I Address R-type In xor s2	7 1111 1 15UNCT3 84 (0x54) 854, s3 10110100 6 110 1 1FUNCT3 88 (0x58) 88 (0x58) 8struction: 9, s4, s3	20 0100 0 RD 000011 16 10000 0 RD	19 0010011 OP 0011 51 0110011 OP
000000001100001000100000001 0 12 4 2 0000000000000010100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction:	00011 111111111 3 -1 1000011 0P IMMEDIAT	1111100101000 5 0 111 00101 0000 FE RS1 FUNC Address 68 (0 1-type Instruct or is 2, t0, 1111100101110 5 6 111 00101 110 FE RS1 FUNC Address 72 (0 1-type Instruct xor is 2, s2, 1010110010101	001010010011 5 19 00101 0010011 73 RD OP x44) on: -1 100100010011 18 19 10010 0010011 73 RD OP x48) on: 1365 1100100010011 18 19	000000 000000001 FUNCT7	111111111001 19 11111 10011 ATE RS1 F Address R-type in or a6, 00100111010 19 20 10011 10100 RS2 RS1 Address R-type in xor s2 00100111010	7 1111010 7 111 1 EUNCT3 84 (0x54) estruction: s4, s3 00110100 6 110 1 FUNCT3 88 (0x58) estruction: , s4, s3 00100100 4	20 0100 0 RD 000011 16 10000 0 RD	19 0010011 OP 0011 51 0110011 OP
000000001100001000100000001 0 12 4 2 0000000000000010100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: Iw s3, 0(tp) 000000000000000001000101001100 0 4 2 19 00000000000000000000100 010 10011 IMMEDIATE RS1 FUNCT3 RD Address 56 (0x38) R-type Instruction: sub s2, s3, a2 010000001100110011000100101 32 12 19 0 18 0100000 01100 10011 000 10010	00011 1111111 0P 111111111111111111111111111111111111	1111100101000 5 0 111 00101 0000 12 RS1 FUNC Address 68 (0 1-type Instruct ori s2, 10, 1111100101110 5 6 11 00101 110 12 RS1 FUNC Address 72 (0 1-type Instruct xori s2, s2, 10101100101010 18 4 01 10010 100	001010010011 5 19 00101 0010011 73 RD OP x44) 001: -1 100100010011 18 19 10010 0010011 73 RD OP x48) 001: 1365 1100100010011 18 19 10010 0010011	000000 1 FUNCT7 000000 0 0000000 1	11111111001 19 1111 10011 ATE RS1 F Address R-type in or a6, 00100111010 19 20 10011 10100 RS2 RS1 i Address R-type in xor s2 0100111010 19 20 10011 10100	7 1111 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	20 0100 0 RD 000011 16 10000 0 RD	0011 0P 0010011 0P 0011 51 0110011 0P
000000001100001000100000001 0 12 4 2 0000000000000010100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction:	00011 111111111 3 -1 1000011 0P IMMEDIAT	1111100101000 5 0 111 00101 0000 12 RS1 FUNC Address 68 (0 1-type Instruct ori s2, 10, 1111100101110 5 6 11 00101 110 12 RS1 FUNC Address 72 (0 1-type Instruct xori s2, s2, 10101100101010 18 4 01 10010 100	001010010011 5 19 00101 0010011 73 RD OP x44) 001: -1 100100010011 18 19 10010 0010011 73 RD OP x48) 001: 1365 1100100010011 18 19 10010 0010011	000000 1 FUNCT7 000000 0 0000000 1	11111111001 19 1111 10011 ATE RS1 F Address R-type in or a6, 00100111010 19 20 10011 10100 RS2 RS1 i Address R-type in xor s2 00100111010	7 1111010 7 111 1 EUNCT3 84 (0x54) estruction: s4, s3 00110100 6 110 1 FUNCT3 88 (0x58) estruction: , s4, s3 00100100 4	20 0100 0 RD 000011 16 10000 0 RD	19 0010011 OP 0011 51 0110011 OP
000000001100001000100000001 0 12 4 2 0000000000000010100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: Iw s3, 0(tp) 00000000000000001000101001100 0 4 2 19 00000000000000000000100 010 10011 IMMEDIATE RS1 FUNCT3 RD Address 56 (0x38) R-type Instruction: sub s2, s3, a2 010000001100110011000100101 32 12 19 0 18 0100000 01100 10011 000 10010	35 -1 1111111111 OP INMEDIAT 00011 11111111 3 -1 0000011 OP INMEDIAT 10011 0101010 51 1365 0110011 OP IMMEDIAT	1111100101000 5 0 111 00101 0000 12 RS1 FUNC Address 68 (0 1-type Instruct ori s2, 10, 1111100101110 5 6 11 00101 110 12 RS1 FUNC Address 72 (0 1-type Instruct xori s2, s2, 10101100101010 18 4 01 10010 100	001010010011 5 19 00101 0010011 13 RD OP x44) on: -1 100100010011 18 19 10010 0010011 13 RD OP x48) on: 1365 1100100010011 18 19 10010 0010011 18 19 10010 0010011 18 19 10010 0010011 18 19 10010 0010011 18 19 10010 0010011 173 RD OP	000000 0 0000000 0 0000000 1 FUNCT7	11111111001 19 1111 10011 ATE RS1 F Address R-type in or a6, 00100111010 19 20 10011 10100 RS2 RS1 Address R-type in xor s2 xor s2 20100111010 19 20 10011 10100 RS2 RS1 Address R-type in xor s2 xor	7 111 1 1 5 UNCT3 84 (0x54) struction: \$4, \$3 10110100 6 110 1 5 UNCT3 88 (0x58) struction: \$4, \$3, \$0100100 4 100 1 5 UNCT3 92 (0x5c) struction: \$4, \$4, \$5, \$6, \$6, \$7, \$6, \$7, \$6, \$7, \$7, \$7, \$7, \$7, \$7, \$7, \$7, \$7, \$7	20 0100 0 RD 000011 16 10000 0 RD 18 10010 0 RD	19 0010011 OP 0011 51 0110011 OP 0011 51 0110011 OP
000000001100001000100000001 0 12 4 2 000000000000011100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: Iw s3, 0(tp) 000000000000000000100010011001100 0 4 2 19 0000000000000000000100 010 10011 IMMEDIATE RS1 FUNCT3 RD Address 56 (0x38) R-type Instruction: sub s2, s3, a2 0100000011001010101001001001 32 12 19 0 18 0100000 01100 10011 000 10010 FUNCT7 RS2 RS1 FUNCT3 RD Address 60 (0x3c) I-type Instruction: addi t0, x0, 3 0000000000110000000000010100	35 -1 1111111111 OP INMEDIAT 00011 11111111 3 -1 0000011 OP INMEDIAT 10011 0101010 51 1365 0110011 OP IMMEDIAT	1111100101000 5 0 111 00101 0000 EE RS1 FUNC Address 68 (6 1-type Instruct or is 2, 10, 1111100101 110 5 6 111 00101 110 EE RS1 FUNC Address 72 (6 1-type Instruct xor is 2, s2, 101011001001 18 4 101 10010 100 EE RS1 FUNC Address 76 (6 1-type Instruct addi s3, x0	001010010011 5 19 00101 0010011 13 RD OP x44) on: -1 100100010011 18 19 10010 0010011 13 RD OP x48) on: 1365 1100100010011 18 19 10010 0010011 18 19 10010 0010011 18 19 10010 0010011 18 19 10010 0010011 18 19 10010 0010011 173 RD OP	000000 0 0000000 0 0000000 1 FUNCT7	11111111001 19 1111 10011 ATE RS1 F Address R-type in or a6, 0100111010 19 20 10011 10100 RS2 RS1 i Address R-type in xor s2 0100111010 RS2 RS1 i Address R-type in xor s2 01001110100 RS2 RS1 i Address R-type in xor s2 01001110100 RS2 RS1 i	7 1111 1 7 111 1 5UNCT3 84 (0x54) struction: \$4, \$3 10110100 6 110 1 FUNCT3 88 (0x58) struction: , \$4, \$3 10100100 4 100 1 FUNCT3 92 (0x5c) struction: , \$4, \$4, \$6	20 0100 0 RD 000011 16 10000 0 RD 18 10010 0 RD	19 0010011 OP 0011 51 0110011 OP 0011 51 0110011 OP
000000001100001000100000001 0 12 4 2 000000000000011100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: Iw s3, 0(tp) 000000000000000010001001100100 0 4 2 19 00000000000000000100 010 10011 IMMEDIATE RS1 FUNCT3 RD Address 56 (0x38) R-type Instruction: sub s2, s3, a2 010000001100100110001001001 32 12 19 0 18 0100000 01100 10011 000 10010 FUNCT7 RS2 RS1 FUNCT3 RD Address 60 (0x3c) I-type Instruction: addi t0, x0, 3 0000000000110000000000101000 3 0 0 5	35 -1 0100011 0P IIIIIIIIII OP IIIIIIIIIII OP IIIIIIIIII	1111100101000 5 0 111 00101 000 1E RS1 FUNC Address 68 (0 1-type Instruct or is 2, t0, 1111100101110 5 6 111 00101 110 1E RS1 FUNC Address 72 (0 1-type Instruct xor is 2, s2, 1010110101010 18 4 101 10010 100 1E RS1 FUNC Address 76 (0 1-type Instruct addi s3, x6 1111100000000 0 0	001010010011 5 19 00101 0010011 13 RD OP x44) on: -1 100100010011 18 19 10010 0010011 13 RD OP x48) on: 1365 1100100010011 18 19 10010 0010011 18 19 10010 0010011 18 19 10010 0010011 19 19 19	000000 0 0000000 0 0000000 1 FUNCT7	11111111001 19 1111 10011 ATE RS1 F Address R-type In 07 a6, 00100111010 19 20 10011 10100 RS2 RS1 II Address R-type In xor s2 00100111010 19 20 10011 10100 RS2 RS1 II Address R-type In xor s2 0100111010 RS2 RS1 II Address R-type In and a7 00100001010	7 1111010 10 7 1111 1 1 111 1 1 1 111 1 1 1	20 0100 0 RD 000011 16 100000 0 RD 010011 17	19 0011 51 0110011 OP 0011 51 01110011 OP
0000000001100001000100000001 0 12 4 2 00000000000000001100 00100 010 IMMEDIATE RS2 RS1 FUNCT3 Address 52 (0x34) I-type Instruction: w s3, 0(tp) 000000000000000100010011001100 0 4 2 19 0000000000000000100 010 10011 IMMEDIATE RS1 FUNCT3 RD Address 56 (0x38) R-type Instruction: sub s2, s3, a2 01000000110010011001100100101 32 12 19 0 18 0100000 01100 10011 000 10010 FUNCT7 RS2 RS1 FUNCT3 RD Address 60 (0x3c) I-type Instruction: addi t0, x0, 3 0000000000110000000000010100000000000	35 -1 111111111 OP INMEDIAT 00011 1111111 3 -1 0000011 0101010 51 1365 01010011 OP INMEDIAT 10011 1365 01010011 OP INMEDIAT	1111100101000 15 0 111 00101 0000 1E RS1 FUNC Address 68 (6 1-type Instruct 25 6 111 00101 110 15 6 111 00101 110 16 RS1 FUNC Address 72 (6 1-type Instruct 27 10101100101010 18 4 101 10010 100 18 4 101 10010 100 19 RS1 FUNC Address 76 (6 1-type Instruct 28 11111000000000 0 0 011 00000 0000	001010010011 5 19 00101 0010011 13 RD OP x44) on: -1 100100010011 18 19 10010 0010011 1365 1100100010011 18 19 10010 0010011 18 19 10010 0010011 173 RD OP x48) on: 110011 0010011 18 19 10010 0010011 19 19 10011 0010011	000000 1 FUNCT7 000000 1 FUNCT7	Address R-type in or a6, 00100111010 RS2 RS1 Address R-type in vor 82, 00100111010 RS2 RS1 Address R-type in xor s2 010011 10100 RS2 RS1 Address R-type in xor s2 010011 10100 RS2 RS1 Address R-type in xor s2 010011 10100 RS2 RS1 Address R-type in xor s2 0100011 10100	7 1111010 10 7 1111 1 1 111 1 1 1 111 1 1 1	20 0100 0 RD 000011 16 100000 0 RD 010011 17	19 0011 51 00011 51 0011 51 0011 51 0011 51 0011 51 0011 51

Address 96 (0x60) SB-type Instruction:

beq t0, x0, 104 000000000000001010001100011

4 0 5 0 99 000000000100 00000 00101 000 1100011 IMMEDIATE RS2 RS1 FUNCT3 OP

Address 100 (0x64)
UJ-type Instruction:

jal x0, 64 1111110111011111111111000001101111

-18 0 111 1111111111111111101110 00000 1101111 ADDRESS RD OP

Address 104 (0x68) I-type Instruction:

addi t0, x0, -1

111111111111100000000001010010011

-1 0 0 5 19

111111111111 00000 000 00101 0010011

IMMEDIATE RS1 FUNCT3 RD OP

Address 108 (0x6c) I-type Instruction:

slli s2, t0, 15 000000011110010100110010010011

15	5	1	18	19
000000001111	00101	001	10010	0010011
IMMEDIATE	RS1	FUNCT3	RD	OP

Address 112 (0x70) I-type Instruction:

slli s2, s2, 16

0000001000010010001100100010011

16 18 1 18 19 000000010000 10010 001 10010 0010011 IMMEDIATE RS1 FUNCT3 RD OP

> Address 116 (0x74) I-type Instruction:

srai s2, s2, 16

10000001000010010101100100010011

-2032 18 5 18 19 100000010000 10010 101 10010 0010011 IMMEDIATE RS1 FUNCT3 RD OP

> Address 120 (0x78) I-type Instruction:

srli s2, s2, 15

00000000111110010101100100010011

15 18 5 18 19 000000001111 10010 101 10010 0010011 IMMEDIATE RS1 FUNCT3 RD OP

Address 124 (0x7c)
UJ-type Instruction:

jal x0, 124

0000000000000000000000001101111

0	0	111
000000000000000000000000000000000000000	00000	1101111
ADDRESS	RD	OP

Address 128 (0x80) R-type Instruction:

add s2, x0, x0 0000000000000000000010010011

0	0	0	0	18	51
0000000	00000	00000	000	10010	0110011
FUNCT7	RS2	RS1	FUNCT3	RD	OP

Address 132 (0x84) I-type Instruction: Iw s3, 0(tp)

00000000000000100010100110000011

0 4 2 19 3 000000000000 00100 010 10011 0000011 IMMEDIATE RS1 FUNCT3 RD OP

> Address 136 (0x88) I-type Instruction:

addi tp, tp, 4

000000001000010000001000010011

4 4 0 4 19 000000000100 00100 000 00100 0010011 IMMEDIATE RS1 FUNCT3 RD OP

> Address 140 (0x8c) R-type Instruction:

add s2, s2, s3 00000001001110010000100100110011

0 19 18 0 18 51 0000000 10011 10010 000 10010 0110011 FUNCT7 RS2 RS1 FUNCT3 RD OP Address 144 (0x90) I-type Instruction:

addi t0, t0, -1 11111111111110010100001010010011

-1 5 0 5 19 111111111111 00101 000 00101 0010011 IMMEDIATE RS1 FUNCT3 RD OP

> Address 148 (0x94) SB-type Instruction:

bne t0, x0, 132 111111110000000101001100011100011

-8 0 5 1 99 1111111111000 00000 00101 001 1100011 IMMEDIATE RS2 RS1 FUNCT3 OP

> Address 152 (0x98) I-type Instruction:

slli a2, s2, 0 000000000001001001011000010011

0 18 1 12 19 000000000000 10010 001 01100 0010011 IMMEDIATE RS1 FUNCT3 RD OP

> Address 156 (0x9c) I-type Instruction:

jalr x0, 0(ra) 00000000000000001100111

0 1 0 0 103 000000000000 00001 000 00000 1100111 IMMEDIATE RS1 FUNCT3 RD OP