

您的位置

# MIPS 指令集(共31条)

更新于2010-04-12 16:52:27

文章出处:与非网

关键字： 指令集

MIPS 指令集(共31条)									
助记符	指令格式						示例	示例含义	
Bit #	31..26	25..21	20..16	15..11	10..6	5..0			
R-type	op	rs	rt	rd	shamt	func			
add	000000	rs	rt	rd	00000	100000	add \$1,\$2,\$3	\$1=\$2+\$3	rd <- rs + rt ;
addu	000000	rs	rt	rd	00000	100001	addu \$1,\$2,\$3	\$1=\$2+\$3	rd <- rs + rt ; 符号数
sub	000000	rs	rt	rd	00000	100010	sub \$1,\$2,\$3	\$1=\$2-\$3	rd <- rs - rt ; 其
subu	000000	rs	rt	rd	00000	100011	subu \$1,\$2,\$3	\$1=\$2-\$3	rd <- rs - rt ; 其 号数
and	000000	rs	rt	rd	00000	100100	and \$1,\$2,\$3	\$1=\$2 & \$3	rd <- rs & rt ;
or	000000	rs	rt	rd	00000	100101	or \$1,\$2,\$3	\$1=\$2   \$3	rd <- rs   rt ; 其
xor	000000	rs	rt	rd	00000	100110	xor \$1,\$2,\$3	\$1=\$2 ^ \$3	rd <- rs xor rt ; (或)
nor	000000	rs	rt	rd	00000	100111	nor \$1,\$2,\$3	\$1=~(\$2   \$3)	rd <- not(rs   rt) rd=\$1(或非)
slt	000000	rs	rt	rd	00000	101010	slt \$1,\$2,\$3	if(\$2<\$3) \$1=1 else \$1=0	if (rs < rt) rd=1 e 中rs=\$2, rt=\$3,
sltu	000000	rs	rt	rd	00000	101011	sltu \$1,\$2,\$3	if(\$2<\$3) \$1=1 else \$1=0	if (rs < rt) rd=1 e 中rs=\$2, rt=\$3, (无符号数)
sll	000000	00000	rt	rd	shamt	000000	sll \$1,\$2,10	\$1=\$2<<10	rd <- rt << shamt 也就是指令中的
srl	000000	00000	rt	rd	shamt	000010	srl \$1,\$2,10	\$1=\$2>>10	rd <- rt >> shamt rd=\$1
sra	000000	00000	rt	rd	shamt	000011	sra \$1,\$2,10	\$1=\$2>>10	rd <- rt >> shamt 留 其中rt=\$2, rd=\$1
sllv	000000	rs	rt	rd	00000	000100	sllv \$1,\$2,\$3	\$1=\$2<<\$3	rd <- rt << rs ;
srlv	000000	rs	rt	rd	00000	000110	srlv \$1,\$2,\$3	\$1=\$2>>\$3	rd <- rt >> rs ; rd=\$1
srav	000000	rs	rt	rd	00000	000111	srav \$1,\$2,\$3	\$1=\$2>>\$3	rd <- rt >> rs ; 其中rs=\$3, rt=\$
jr	000000	rs	00000	00000	00000	001000	jr \$31	goto \$31	PC <- rs

I-type	op	rs	rt	immediate			
addi	001000	rs	rt	immediate	addi \$1,\$2,100	\$1=\$2+100	rt <- rs + (sign-extended)immediate 中rt=\$1,rs=\$2
addiu	001001	rs	rt	immediate	addiu \$1,\$2,100	\$1=\$2+100	rt <- rs + (zero-extended)immediate 中rt=\$1,rs=\$2
andi	001100	rs	rt	immediate	andi \$1,\$2,10	\$1=\$2 & 10	rt <- rs & (zero-extended)immediate 中rt=\$1,rs=\$2
ori	001101	rs	rt	immediate	andi \$1,\$2,10	\$1=\$2   10	rt <- rs   (zero-extended)immediate 中rt=\$1,rs=\$2
xori	001110	rs	rt	immediate	andi \$1,\$2,10	\$1=\$2 ^ 10	rt <- rs xor (zero-extended)immediate 中rt=\$1,rs=\$2
lui	001111	00000	rt	immediate	lui \$1,100	\$1=100*65536	rt <- immediate*2 <sup>16</sup> 标寄存器高16位，目标寄存器低16位
lw	100011	rs	rt	immediate	lw \$1,10(\$2)	\$1=memory[\$2+10]	rt <- memory[rs+immediate] extend)immediate
sw	101011	rs	rt	immediate	sw \$1,10(\$2)	memory[\$2+10]=\$1	memory[rs+(sign-extended)immediate] rt ; rt=\$1,rs=\$2
beq	000100	rs	rt	immediate	beq \$1,\$2,10	if(\$1==\$2) goto PC+4+40	if (rs == rt) PC <- (PC+4)+immediate
bne	000101	rs	rt	immediate	bne \$1,\$2,10	if(\$1!=\$2) goto PC+4+40	if (rs != rt) PC <- (PC+4)+immediate
slti	001010	rs	rt	immediate	slti \$1,\$2,10	if(\$2<10) \$1=1 else \$1=0	if (rs <(sign-extended)immediate) \$1=1 ; 其中rs=\$2, rt=\$1
sltiu	001011	rs	rt	immediate	sltiu \$1,\$2,10	if(\$2<10) \$1=1 else \$1=0	if (rs <(zero-extended)immediate) \$1=1 ; 其中rs=\$2, rt=\$1
J-type	op	address					
j	000010	address			j 10000	goto 10000	PC <- (PC+4)[31..28],address[28..0]
jal	000011	address			jal 10000	\$31<-(PC+4); goto 10000	\$31<-(PC+4); PC <- (PC+4)[31..28],address[28..0]

注 意: 因为MIPS16只有16个16位的寄存器，所以JAL指令中\$31改成\$15，所有立即数均无需扩展，LUI指令直接就是将立即数付给RT寄存器。

评论

更多评论

用户信息: 匿名用户

 正在读取数据.....

