

#### 百年同僚

- 多周期CPU设计举例
- 1.根据指令功能设计数据通路
  - (1) 取指令操作,列出部件关系表

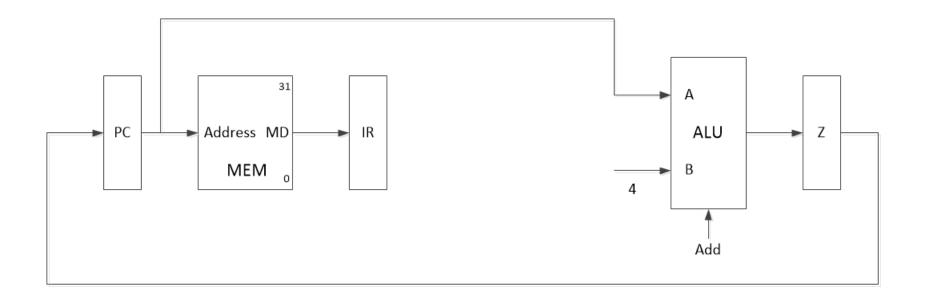
MEM←PC,IR←MEM,PC←PC+4、Z

- 所需部件: PC、MEM、IR、ALU(完成PC增值)
- 部件之间数据输入输出关系如下表:

	PC	MEN	I	IR	Al	_U	Z	
		Address MD			А	В		
Fetch	Z	PC		MD	PC	4	ALU	



	PC	MEN	I	IR	Al	_U	Z	
		Address MD			А	В		
Fetch	Z	PC		MD	PC	4	ALU	





### 列出部件关系表

(2) Addu rd,rs,rt ;rd←rs+rt

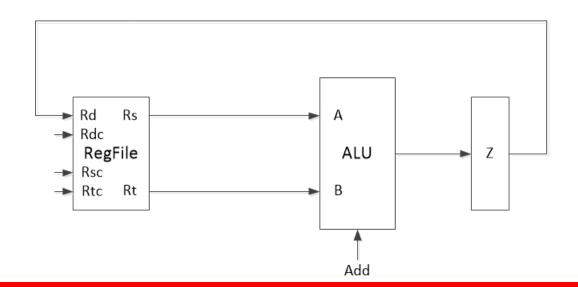
所需部件: Regfile、ALU、Z

部件之间数据输入输出关系如下表:

	PC	MEM		IR	AL	.U	Z	RegFile		
		Address MD			Α	В		Rd		
Fetch	Z	PC		MD	PC	4	ALU			
Addu					Rs	Rt	ALU	Z		



	PC	MEM		IR	AL	.U	Z	RegFile		
		Address MD			А	В		Rd		
Fetch	Z	PC		MD	PC	4	ALU			
Addu					Rs	Rt	ALU	Z		





### (3) Subu rd,rs,rt ;rd←rs-rt

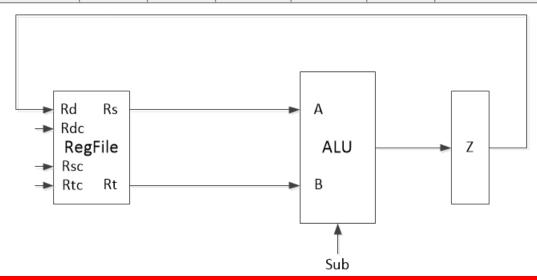
● 所需部件: Regfile、ALU、Z

● 部件之间数据输入输出关系如下表:

	PC	MEM		IR	AL	.U	Z	RegFile		
		Address MD			Α	В		Rd		
Fetch	Z	PC		MD	PC	4	ALU			
Addu					Rs	Rt	ALU	Z		
Subu					Rs	Rt	ALU	Z		



	PC	MEM		IR	AL	_U	Z	RegFile		
		Address MD			Α	В		Rd		
Fetch	Z	PC		MD	PC	4	ALU			
Addu					Rs	Rt	ALU	Z		
Subu					Rs	Rt	ALU	Z		





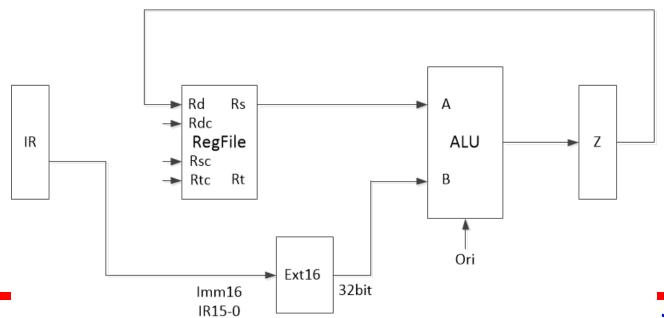
### (4) Ori rt,rs,imm16 ;rt←rs ∨ext.imm16

- 所需部件: Regfile、ALU、Ext16、Z
- 部件之间数据输入输出关系如下表:

	PC	ME	M	IR	AL	_U	Z	RegFile	Ext16		
		Address	MD		А	В		Rd			
Fetch	Z	PC		MD	PC	4	ALU				
Addu					Rs	Rt	ALU	Z			
Subu					Rs	Rt	ALU	Z			
Ori					Rs Ext16		ALU	Z	imm16		



	PC	ME	M	IR	1	ALU	Z	RegFile	Ext16		
		Address	MD		А	В		Rd			
Fetch	Z	PC		MD	PC	4	ALU				
Addu		3		26 25	21	20 16	15			0	
Subu			Ori(001101	)	rs	rt	_	imm16			
Ori					Rs	Ext16	ALU	Z	imm16		



四听入学电信学院



百年同僚

### 鬥(5) Sll rd,rt,sa ;rd←rt左移sa位

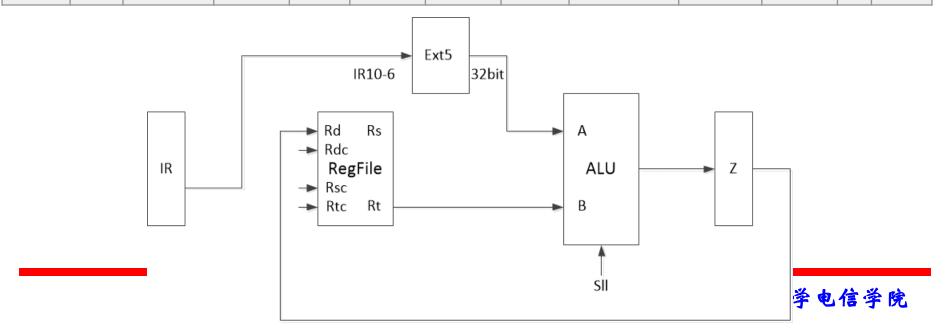
● 所需部件: Regfile、ALU、Z、Ext5

部件之间数据输入输出关系如下表:

	PC	ME	M	IR	AL	_U	Z	RegFile	Ext16	Ext5	
		Address	MD		Α	В		Rd			
Fetch	Z	PC		MD	PC	4	ALU				
Addu					Rs	Rt	ALU	Z			
Subu					Rs	Rt	ALU	Z			
Ori					Rs	Ext16	ALU	Z	imm16		
SII					Ext5	Rt	ALU	Z		IR(Sa)	



	PC		ME	M	IR		Al	LU		Z	R	egFile	Ext16	Ext5	
		Addre	ess	MD		A	4	В				Rd			
Fetch	Z	PC	,		MD	Р	С	4		ALU					
Addu			31	26	25	21	20	16	15	11	10	6	5	0	
Subu			SII	(000000)	0000	00		rt		rd		sa	000000		
Ori						Rs		Ext16	3	ALU		Z	imm16		
SII					Ex		t5	Rt		ALU		Z		IR(Sa)	





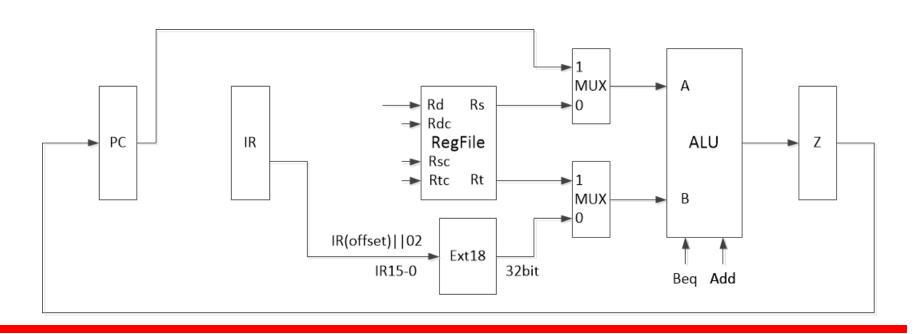
## (6)Beq rs,rt,offset; rs=rt,PC←PC+Sign ext(offset||0²)否则PC←PC+4

- 所需部件: PC、Regfile、ALU、Ext18、Z
- 部件之间数据输入输出关系如下表:

	PC	ME	M	IR	Al	_U	Z	RegFile	Ext16	Ext5	Ext18
		Address	MD		Α	В		Rd			
Fetch	Z	PC		MD	PC	4	ALU				
Addu					Rs	Rt	ALU	Z			
Subu					Rs	Rt	ALU	Z			
Ori					Rs	Ext16	ALU	Z	imm16		
SII					Ext5	Rt	ALU	Z		IR(Sa)	
Beq	Z				PC	Ext18	ALU				Offset

同济大学电信学院

	PC	N	EM	IR		AL	_U	Z	RegFile	Ext16	Ext5	Ext18	
		Address	MD		А		В		Rd				
Fetch	Z	PC		MD	PC		4	ALU					
Addu		3:	26	5 25	21	20	16 1	5			0		
Subu			Beq(000100)	rs	5		rt	1	offse	t			
Ori					Rs		Ext16	ALU	Z	imm16			
SII					Ext5		Rt	ALU	Z		IR(Sa)		
Beq	Z				PC		Ext18	ALU				Offset	



#### 同游大学电信学院



### **百年**风海 TON (IT WITH SITY ) target;

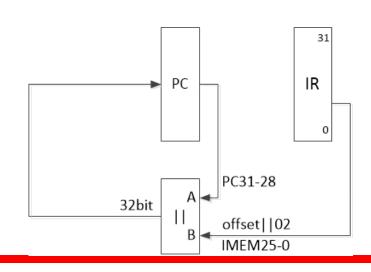
PC←PC<sub>31-28</sub>||instr\_index||0<sup>2</sup>

● 所需部件: PC、||

部件之间数据输入输出关系如下表:

	РС	MEI	VI	IR	А	LU	Z	RegFile	Ext16	Ext5	Ext18		
		Address	MD		А	В		Rd				А	В
Fetch	Z	PC		MD	PC	4	ALU						
Addu					Rs	Rt	ALU	Z					
Subu					Rs	Rt	ALU	Z					
Ori					Rs	Ext16	ALU	Z	imm16				
SII					Ext5	Rt	ALU	Z		IR(Sa)			
Beq	Z				PC	Ext18	ALU				Offset		
J	II											PC 31-28	IR 25-0

	РС	MEI	M	IR	А	LU	Z	RegFile	Ext16	Ext5	Ext18		I
		Address	MD		А	В		Rd				А	В
Fetch	Z	PC		MD	PC	4	ALU						
Addu					Rs	Rt	ALU	Z					
Subu					Rs	Rt	ALU	Z					
Ori		3	31	26 2	25			_			0		
SII			J(0000	10)				instr_index	(				
Beq	Z				PC	Ext18	ALU				Offset		
J	II											PC 31-28	IR 25-0



#### 同济大学电信学院



### 百年风海 Lw rt,offset(base) ;rt←[rs+Sign\_ext\_offset]

- 所需部件: MEM、Regfile、ALU、Z、Ext16
- 部件之间数据输入输出关系如下表:

	PC	MEN	И	IR	Α	LU	Z	RegFile	Ext16	Ext5	Ext18		II
		Address	MD		А	В		Rd				А	В
Fetch	Z	PC		MD	PC	4	ALU						
Addu					Rs	Rt	ALU	Z					
Subu					Rs	Rt	ALU	Z					
Ori					Rs	Ext16	ALU	Z	imm16				
SII					Ext5	Rt	ALU	Z		IR(Sa)			
Beq	Z				PC	Ext18	ALU				Offset		
J	II											PC 31-28	IR 25-0
Lw		Z			Rs	Ext16	ALU	MD	Offset				

		Address	MD		А	В		Rd				А	В
Fetch	Z	PC		MD	PC	4	ALU						
Addu					Rs	Rt	ALU	Z					
Subu					Rs	Rt	ALU	Z					
Ori					Rs	Ext16	ALU	Z	imm16				
SII			31	26	25	21 20	16	15		ID/0a)	0		
Beq	Z		Lw(100	0011)	base	2	rt		offs	set			
J	II											PC 31-28	IR 25-0
Lw		Z			Rs	Ext16	ALU	MD	Offset				
		ddress MD -		IR 0	offse IMEM1	Rsc Rtc	File Rt			A ALU -		31 Z 0	
							1						院

RegFile

Z

PC

MEM

IR

ALU

Ext16

Ext5

Ext18

 $\parallel$ 

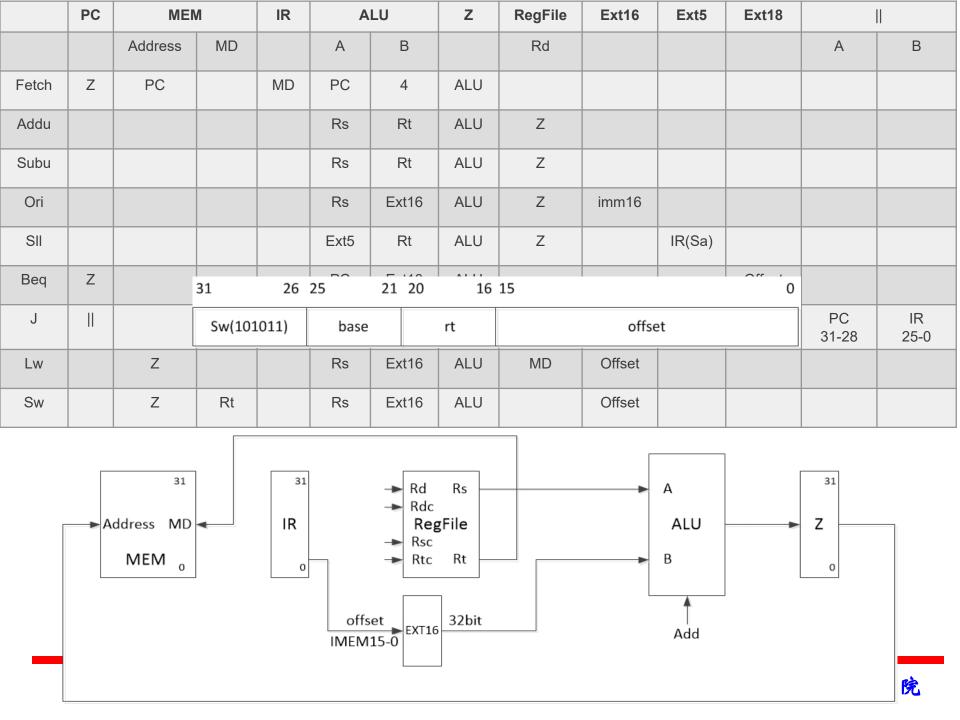


(9) Sw rt,offset(base);[base+Sign ext offset]←rt

所需部件: MEM、Regfile、ALU、Z、Ext16

部件之间数据输入输出关系如下表:

	РС	MEN	/	IR	А	LU	Z	RegFile	Ext16	Ext5	Ext18		II
		Address	MD		А	В		Rd				А	В
Fetch	Z	PC		MD	PC	4	ALU						
Addu					Rs	Rt	ALU	Z					
Subu					Rs	Rt	ALU	Z					
Ori					Rs	Ext16	ALU	Z	imm16				
SII					Ext5	Rt	ALU	Z		IR(Sa)			
Beq	Z				PC	Ext18	ALU				Offset		
J	II											PC 31-28	IR 25-0
Lw		Z			Rs	Ext16	ALU	MD	Offset				
Sw		Z	Rt		Rs	Ext16	ALU		Offset				





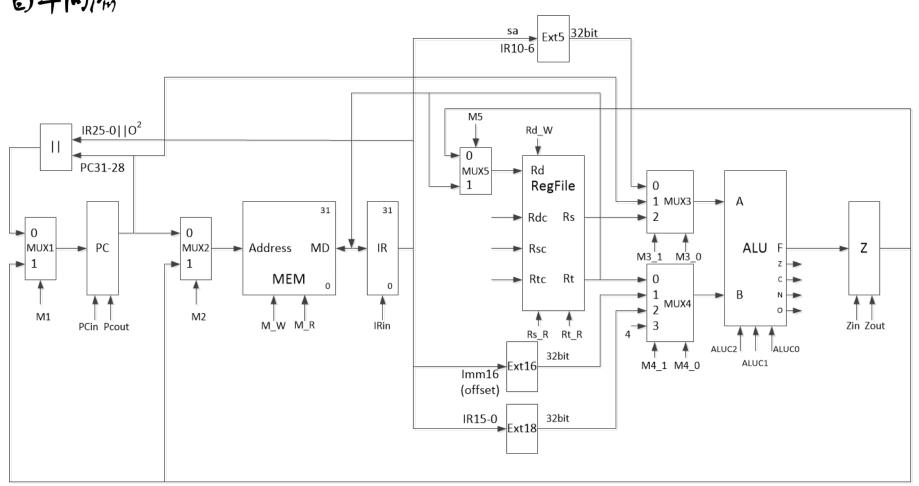
## 如何构成8条指令的数据通路?

	PC	MEN	Л	IR	А	LU	Z	RegFile	Ext16	Ext5	Ext18		II
		Address	MD		А	В		Rd				А	В
Fetch	Z	PC		MD	PC	4	ALU						
Addu					Rs	Rt	ALU	Z					
Subu					Rs	Rt	ALU	Z					
Ori					Rs	Ext16	ALU	Z	imm16				
SII					Ext5	Rt	ALU	Z		IR(Sa)			
Beq	Z				PC	Ext18	ALU				Offset		
J	II											PC 31-28	IR 25-0
Lw		Z			Rs	Ext16	ALU	MD	Offset				
Sw		Z	Rt		Rs	Ext16	ALU		Offset				

同济大学电信学院

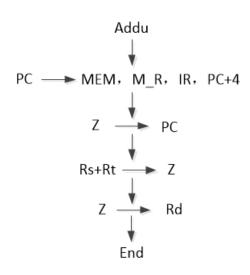


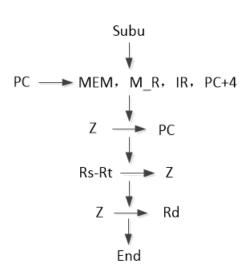
百年同濟

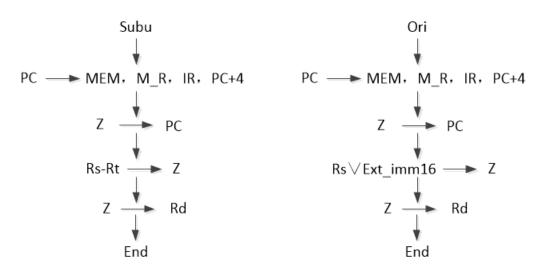




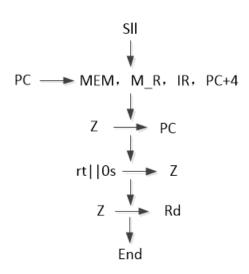
### 2.绘制指令流程图

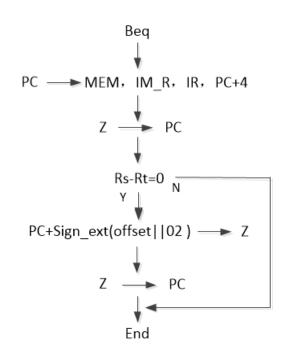


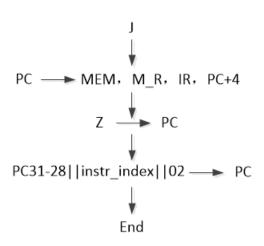




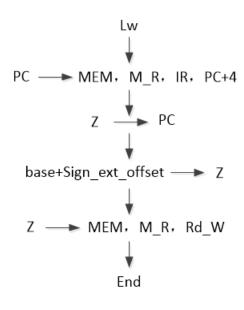


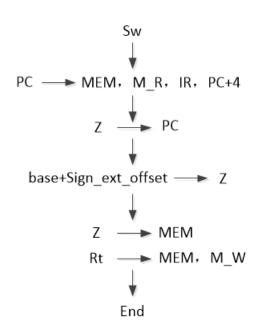














### 3.编排指令操作时间表

依据各条机器指令的操作流程图将每条指令执行所需的控制信号填入下表。

假设: ALU控制如下表。

	ALUC2	ALUC1	ALUC0
Add	0	0	0
Sub	0	0	1
Ori	0	1	0
SII	0	1	1

				Ad	ldu						Sub	ou						Or	i						SII						В	eq						J					Lw	1					Sw		
	Т	1	T2	T	3 T	4	T5	T.	1 .	T2	T3	3 7	Г4	T5	Т	1 7	2	Т3	T4	Т	5	T1	Т	2	Т3	T4	Т.	5	T1	T2	2 T	3 -	T4	T5	T1	T	2 T	3 7	Γ4 ·	T5	T1	T2	Т3	3 T	4 T	5	T1	T2	Т3	T4	T5
PCout																																																			
M2																																																			1
M_R																																																			
IRin																																																			
PCin																																																			
Rsc4-0																																																			
Rtc4-0																																																			
Rs_R																																																			
Rt_R																																																			
M3_1																																																			
M3_0																																																			
M4_1																																																			
M4_0																																																			
ALUC2																																																			
ALUC1																																																			
ALUC0																																																			
Zin																																																			
Zout																																																			
M5																																																			
Rdc4-0																																																			
Rd_W																																																			
M_W																																																			I
M1																																																			

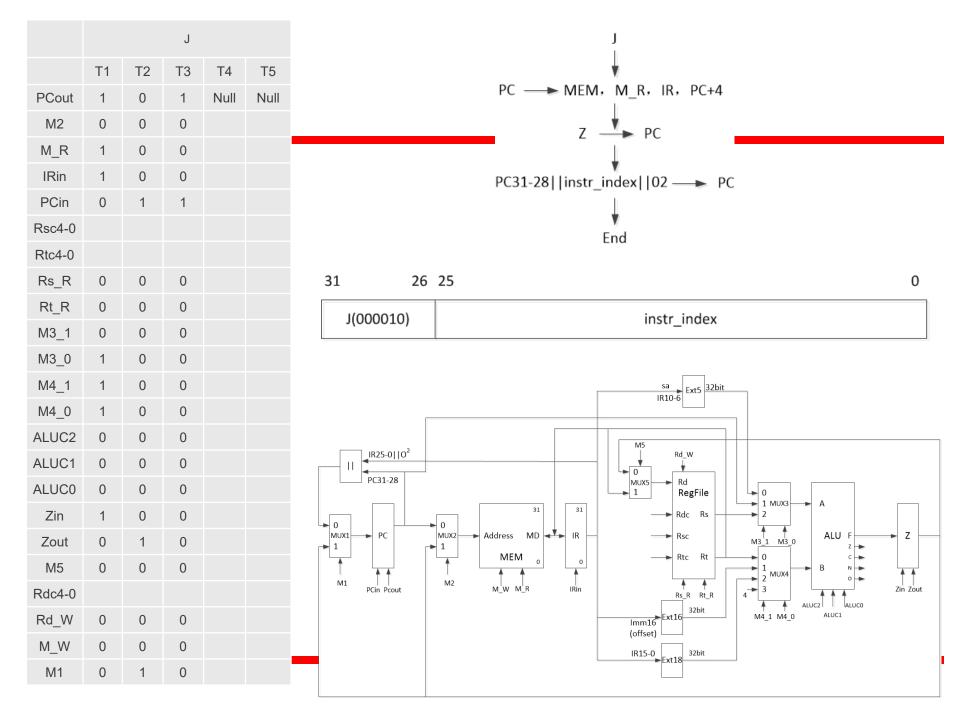
			Add	u									Ad	du					
	T1	T2	Т3	T4	T5					,	200	- M	· \	M D ID D	~ . <b>4</b>				
PCout	1	0	0	0	Null					1	PC.	IVIE	_IVI, 	M_R,IR,P	L+4				
M2	0	0	0	0								Z		<b>▶</b> PC					
M_R	1	0	0	0								Rc±	Rt —	/ > Z					
IRin	1	0	0	0										,					
PCin	0	1	0	0								7	z —	► Rd					
Rsc4-0			IR25-21										En	nd					
Rtc4-0			IR20-16																
Rs_R	0	0	1	0		:	31	26	25	2	1	20	16	15 1	1 10	0 6	5		0
Rt_R	0	0	1	0			Addu(0000	200)		rs		rt		rd		00000	1	00001	
M3_1	0	0	1	0			Adddood										1		
M3_0	1	0	0	0															
M4_1	1	0	0	0										sa Ext5 32	oit				
M4_0	1	0	0	0															
ALUC2	0	0	0	0			IR25-0  O <sup>2</sup>						N	15   Rd_W 					
ALUC1	0	0	0	0			PC31-28	1					0 MU 1	JX5 → Rd		0			
ALUC0	0	0	0	0					, [	3:	1	31		Rdc Rs		1 MUX3 2	Α		$\neg$
Zin	1	0	1	0		0 MUX1 1	PC PC	0 MUX2	2 A	ddress MD	) <mark>~'</mark>	IR —		Rsc		M3_1 M3_0	ALU F	-	z
Zout	0	1	0	1			<b>A A</b>	1		MEM ,	0	0		—► Rtc Rt		0 1 2 MUX4	c B N	•	<b>.</b> .
M5	0	0	0	0		M1	PCin Pcout	M2		M_W M_R		IRin		Rs_R Rt_R		3	*	Zi	n Zout
Rdc4-0				IR15-11										m16 Ext16 32bit		↑ ↑ ALU M4_1 M4_0	ALUC1	JC0	
Rd_W	0	0	0	1										15-0 5 110 32bit					
M_W	0	0	0	0										Ext18					
M1	0	1	0	0															

			Sub	u							Sı	ıbu				
	T1	T2	Т3	T4	T5							<b>V</b>				
PCout	1	0	0	0	Null				PC	-	MEM,	M_R, IR	, PC	+4		
M2	0	0	0	0								<b>\</b>				
M_R	1	0	0	0							Z —	PC				
IRin	1	0	0	0							Rs-Rt -	<b>♥</b> Z				
PCin	0	1	0	0												
Rsc4-0			IR25-21								Z —	▼   Rd				
Rtc4-0			IR20-16								E	<b>♥</b> nd				
Rs_R	0	0	1	0		31	26	25	21	20	16		11	10	5 5	0
Rt_R	0	0	1	0		6.	-l(000000)				1			00000	100	011
M3_1	0	0	1	0		St	ıbu(000000)		rs		rt	rd		00000	1000	J11
M3_0	1	0	0	0								63		1.5		
M4_1	1	0	0	0								IR10-6	Ext5 32	bit		
M4_0	1	0	1	0												
ALUC2		0	0	0			IR25-0  O <sup>2</sup>						i_w ↓			
ALUC1	0	0	0	0			PC31-28					MUX5	Rd RegFile	0		
ALUC0		0	1	0						31	31		dc Rs	1 MUX3 2	A	
Zin	1	0	1	0		0 MUX:		0 MUX2 1	Address [	MD 👈	IR —	— <b>▶</b> R	sc	M3_1 M3_0	ALU F	Z
Zout	0	1	0	1		1	<b>A</b> • •	1	MEM	0	0	R	tc Rt	0 1 2 MUX4	B N	
M5	0	0	0	0		M1	PCin Pcout	M2	M_W M_F	1	IRin		R Rt_R	3	ALUC2 ALUCC	Zin Zout
Rdc4-0	0	0	0	IR15-11								Imm16 Ext16	32bit	M4_1 M4_0	ALUC2   ALUC1	
Rd_W	0	0	0	1								(offset)	32bit			
M_W	0	0	0	0							4	►Ext18				
M1	0	1	0	0												

			Ori			Ori	
	T1	T2	Т3	T4		₩.	
PCout	1	0	0	0	I	PC → MEM, M_R, IR, PC+4	
M2	0	0	0	0		₩.	
M_R	1	0	0	0		Z PC	
IRin	1	0	0	0		Rs∨Ext_imm16 — Z	
PCin	0	1	0	0		NS V EXC_IIIIII 2	
Rsc4-0			IR25-21			Z <del>▼</del> Rd	
Rtc4-0						<b>↓</b>	
Rs_R	0	0	1	0		End	
Rt_R	0	0	0	0	31	26 25 21 20 16 15	0
M3_1	0	0	1	0	0=:/00110	01)	
M3_0	1	0	0	0	Ori(00110	01) rs rt imm16	
M4_1	1	0	0	0		sa	
M4_0	1	0	1	0		sa   Ext5   32bit	
ALUC2	0	0	0	0			
ALUC1	0	0	1	0	IR25-0  0 <sup>2</sup>	M5 Rd_W	
ALUC0	0	0	0	0	PC31-28	Mux5 → Rd	
Zin	1	0	1	0		1 RegFile 0 1 MUX3 A A Rdc Rs 2	
Zout	0	1	0	1	0 MUX1 PC	0 MUX2 Address MD IR Rsc ALU F	z
M5	0	0	0	0		MEM 0 0 Rtc Rt 0	
Rdc4-0				IR20-16	M1 PCin Pcout	M <sub>2</sub> ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑	Zin Zout
Rd_W	0	0	0	1		Imm16 Ext16 32bit M4_1 M4_0 ALUC2 ALUC1	
M_W	0	0	0	0		(offset)	
M1	0	1	0	0		IR15-0 Ext18 32bit	

			SII			SII
	T1	T2	Т3	T4	T5	<b>*</b>
PCout	1	0	0	0	Null	PC → MEM, M_R, IR, PC+4
M2	0	0	0	0		$Z \xrightarrow{\qquad} PC$ $rt     0s \xrightarrow{\qquad} Z$ $Z \xrightarrow{\qquad} Rd$
M_R	1	0	0	0		
IRin	1	0	0	0		rt  0s Z
PCin	0	1	0	0		_ +
Rsc4-0						Z Rd
Rtc4-0			IR20-16			♥ End
Rs_R	0	0	0	0		
Rt_R	0	0	1	0		31 26 25 21 20 16 15 11 10 6 5 0
M3_1	0	0	0	0		SII(000000) 00000 rt rd sa 000000
M3_0	1	0	0	0		
M4_1	1	0	0	0		Sa
M4_0	1	0	0	0		IK10-6
ALUC2	0	0	0	0		
ALUC1	0	0	1	0		IR25-0  O <sup>2</sup> Rd_W
ALUC0	0	0	1	0		PC31-28 Rd RegFile 0
Zin	1	0	1	0		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
Zout	0	1	0	1		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
M5	0	0	0	0		MEM o o Rtc Rt 0 1 2 MUX4 B N D D D D D D D D D D D D D D D D D D
Rdc4-0				IR15-11		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
Rd_W	0	0	0	1		
M_W	0	0	0	0		(offset)
M1	0	1	0	0		IR15-0 Ext18 32bit

							Beq
			ı	Beq (Z=1)			<b>*</b>
		T1	T2	Т3	T4	T5	PC → MEM, IM_R, IR, PC+4
	PCout	1	0	0	1	0	Z PC
_	M2	0	0	0	0	0	$\downarrow$
1	M_R	1	0	0	0	0	Rs-Rt=0 N
	- IRin	1	0	0	0	0	Y
	PCin	0	1	0	0	1	PC+Sign_ext(offset  02) — ➤ Z
	Rsc4-0			IR25-21			i evengui_exit(eniset  e2 )
	Rtc4-0			IR20-16			Z PC
	Rs_R	0	0	1	0	0	
	- Rt_R	0	0	1	0	0	End
	— M3_1	0	0	1	0	0	31 26 25 21 20 16 15 0
	— M3_0	1	0	0	1	0	Par/(000100)
	_ M4_1	1	0	0	1	0	Beq(000100) rs rt offset
	_ M4_0	1	0	0	0	0	sa   Ext5   32bit
	ALUC2	0	0	0	0	0	
	ALUC1	0	0	0	0	0	M5
	ALUC0	0	0	1	0	0	
	Zin	1	0	0	1	0	1 RegFile 0
	Zout	0	1	0	0	1	31   31   Rdc Rs   2
	M5	0	0	0	0	0	MUX1 PC MUX2 Address MD IR Rsc M3 1 M3 0 ALU F Z Rtc Rt D 0 C C
	Rdc4-0						1 MUX4 > B N
	Rd_W	0	0	0	0	0	Rs_R Rt_R 4 ALUC2 ALUC0
	M_W	0	0	0	0	0	
ı	M1	0	1	0	0	1	IR15-0 Ext18 32bit



						Lw	
			Lw				
	T1	T2	Т3	T4	T5	PC —► MEM, M_R, IR, PC+4	
PCout	1	0	0	0	Null	<b>V</b>	
M2	0	0	0	1		Z PC	
M_R	1	0	0	1			
IRin	1	0	0	0		base+Sign_ext_offset ──► Z	
PCin	0	1	0	0		. ↓	
Rsc4-0			IR25-21			Z —► MEM, M_R, Rd_W	
Rtc4-0							
Rs_R	0	0	1	0		End	
Rt_R	0	0	0	0		31 26 25 21 20 16 15	0
M3_1	0	0	1	0		Lw(100011) base rt offset	
M3_0	1	0	0	0			
M4_1	1	0	0	0		sa IR10-6 Ext5 32bit	
M4_0	1	0	1	0			
ALUC2	0	0	0	0		M5	
ALUC1	0	0	0	0		IR25-0  O <sup>2</sup>	
ALUC0	0	0	0	0		PC31-28  MUX5  Rd  RegFile  1 MUX3  A	
Zin	1	0	1	0		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	
Zout	0	1	0	1		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	> Z
M5	0	0	0	1		MEM 0 0 1 1 MUX4 B B N B N B N B N B N B N B N B N B N	<b>↑ ↑</b>
Rdc4-0				IR20-16		PCIn Pcout M_W M_R IRin Rs_R Rt_R 4 3	Zin Zout
Rd_W	0	0	0	1		Imm16 Ext16 32bit M4_1 M4_0 ALUC1	
M_W	0	0	0	0		(offset) IR15-0 State 32bit	
M1	0	1	0	0		Ext18	

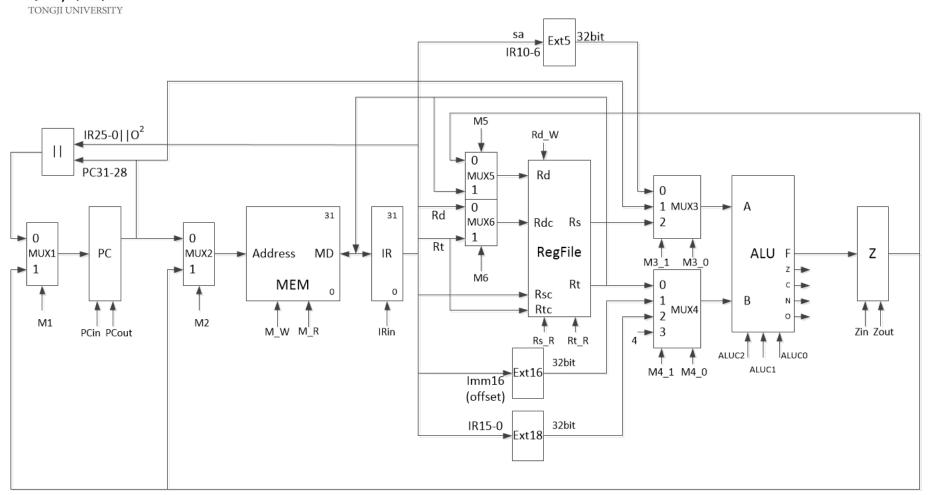
							Sw												
		0 0 0 0 1 R 1 0 0 0 D 0 0 D 0 0 0 D 0 0 0 D 0 0 0 D 0 0 0 D 0 0 0 0			•														
		T1	T2	Т3	T4	T5	PC → MEM, M_R, IR, PC+4												
	PCout					Null	Z PC												
	M2						1												
Ī	M_R						base+Sign_ext_offset ── Z  Z ── MEM												
	- IRin																		
	PCin	0																	
ı	Rsc4-0						Rt —► MEM, M_W												
	Rtc4-0				IR20-16		1												
	Rs_R	0	0	1	0		End												
	Rt_R	0	0	0	1		31 26 25 21 20 16 15	0											
	M3_1	0	0	1	0		Lw(100011)												
	M3_0	1	0	0	0		Lw(100011) base rt offset												
	M4_1	1	0	0	0		$ \begin{array}{c} sa \\ \hline IR10-6 \end{array} $ Ext5 $\frac{32bit}{}$												
	M4_0	1	0	1	0														
1	ALUC2	0	0	0	0		M5												
1	ALUC1	0	0	0	0		IR25-0  0 <sup>2</sup>   IR25-0  0 <sup>2</sup>   Rd_W   0												
1	ALUC0	0	0	0	0		PC31-28  MUX5  Rd  RegFile  0  1 MUX3  A												
	Zin	1	0	1	0		31 31 Rdc Rs 2												
	Zout	0	1	0	1		1   MEM   MEM   MS_1 MS_0   Z	Z											
	M5	0	0	0	0		M1 M2 B N D D D D D D D D D D D D D D D D D D	<b>†</b> †											
ı	Rdc4-0						PCin Pcout M_W M_R IRin TT 3 Zii	in Zout											
	Rd_W	0	0	0	0														
	M_W	0	0	0	1		IR15-0 Ext18 32bit	l l											
	M1	0	1	0	0														

	Addu			:	Subu			Ori			SII			Beq			J		Lw			Sw				
	T1	T2	T3	T4	T5	Т3	T4	T5	Т3	T4	T5	Т3	T4	T5	Т3	T4	T5									
PCout	1	0	0	0	Null	0	1	0	1	Null	Null	0	0	Null	0	0	Null									
M2	0	0	0	0		0	0		0	0		0	0		0	0	0	0			0	1		0	1	
M_R	1	0	0	0		0	0		0	0		0	0		0	0	0	0			0	1		0	0	•
IRin	1	0	0	0		0	0		0	0		0	0		0	0	0	0			0	0		0	0	
PCin	0	1	0	0		0	0		0	0		0	0		0	0	1	1			0	0		0	0	
Rsc4-0			IR25-21			IR25-21			IR25-21						IR25-21						IR25-21			IR25-21		
Rtc4-0			IR20-16			IR20-16						IR20-16			IR20-16										IR20-16	
Rs_R	0	0	1	0		1	0		1	0		0	0		1	0	0	0			1	0		1	0	
Rt_R	0	0	1	0		1	0		0	0		1	0		1	0	0	0			0	0		0	1	
M3_1	0	0	1	0		1	0		1	0		0	0		1	0	0	0			1	0		1	0	
M3_0	1	0	0	0		0	0		0	0		0	0		0	1	0	0			0	0		0	0	
M4_1	1	0	0	0		0	0		0	0		0	0		0	1	0	0			0	0		0	0	
M4_0	1	0	0	0		0	0		1	0		0	0		0	0	0	0			1	0		1	0	
ALUC2	0	0	0	0		0	0		0	0		0	0		0	0	0	0			0	0		0	0	
ALUC1	0	0	0	0		0	0		1	0		1	0		0	0	0	0			0	0		0	0	
ALUC0	0	0	0	0		1	0		0	0		1	0		1	0	0	0			0	0		0	0	
Zin	1	0	1	0		1	0		1	0		1	0		0	1	0	0			1	0		1	0	
Zout	0	1	0	1		0	1		0	1		0	1		0	0	1	0			0	1		0	1	
M5	0	0	0	0		0	0		0	0		0	0		0	0	0	0			0	1		0	0	
Rdc4-0				IR15-11		I	IR15-11			IR20-16			IR15-11								I	IR20-16				
Rd_W	0	0	0	1		0	1		0	1		0	1		0	0	0	0			0	1		0	0	
M_W	0	0	0	0		0	0		0	0		0	0		0	0	0	0			0	0		0	1	
M1	0	1	0	0		0	0		0	0		0	0		0	0	1	0			0	0		0	0	•



注意: Rdc的输入有两个来源, IM15-11和IM20-16, 所以, 在Rdc的输入端要加一个MUX6。





			Addu				Subu			Ori			SII		Beq (	(Z=1)	)		J			Lw			Sw	
	T1	T2	Т3	T4	T5	Т3	T4	T5	Т3	T4	T5	Т3	T4	T5	Т3	T4	T5									
PCout	1	0	0	0	Null	0	1	0	1	Null	Null	0	0	Null	0	0	Null									
M2	0	0	0	0		0	0		0	0		0	0		0	0	0	0			0	1		0	1	
M_R	1	0	0	0		0	0		0	0		0	0		0	0	0	0			0	1		0	0	
IRin	1	0	0	0		0	0		0	0		0	0		0	0	0	0			0	0		0	0	
PCin	0	1	0	0		0	0		0	0		0	0		0	0	1	1			0	0		0	0	
Rsc4-0			IR25-21			IR25-21			IR25-21						IR25-21						IR25-21			IR25-21		
Rtc4-0			IR20-16			IR20-16						IR20-16			IR20-16										IR20-16	
Rs_R	0	0	1	0		1	0		1	0		0	0		1	0	0	0			1	0		1	0	
Rt_R	0	0	1	0		1	0		0	0		1	0		1	0	0	0			0	0		0	1	
M3_1	0	0	1	0		1	0		1	0		0	0		1	0	0	0			1	0		1	0	
M3_0	1	0	0	0		0	0		0	0		0	0		0	1	0	0			0	0		0	0	
M4_1	1	0	0	0		0	0		0	0		0	0		0	1	0	0			0	0		0	0	
M4_0	1	0	0	0		0	0		1	0		0	0		0	0	0	0			1	0		1	0	
ALUC2	0	0	0	0		0	0		0	0		0	0		0	0	0	0			0	0		0	0	
ALUC1	0	0	0	0		0	0		1	0		1	0		0	0	0	0			0	0		0	0	
ALUC0	0	0	0	0		1	0		0	0		1	0		1	0	0	0			0	0		0	0	
Zin	1	0	1	0		1	0		1	0		1	0		0	1	0	0			1	0		1	0	
Zout	0	1	0	1		0	1		0	1		0	1		0	0	1	0			0	1		0	1	
M5	0	0	0	0		0	0		0	0		0	0		0	0	0	0			0	1		0	0	
Rdc4-0				IR15-11			IR15-11			IR20-16			IR15-11									IR20-16				
Rd_W	0	0	0	1		0	1		0	1		0	1		0	0	0	0			0	1		0	0	
M_W	0	0	0	0		0	0		0	0		0	0		0	0	0	0			0	0		0	1	
M1	0	1	0	0		0	0		0	0		0	0		0	0	1	0			0	0		0	0	
M6	0	0	0	0		0	0		0	1		0	0		0	0	0	0			0	1		0	0	



#### 4.进行微操作综合

按照所有机器指令的操作时间表,把相同的 微操作综合起来,得到每个微操作的逻辑表达 式。本例共有24个控制信号。

PCout=T1+T3J+T4Beq

M2=T4(Lw+Sw)

M R=T1+T4Lw

IRin=T1



百年同為

Rsc4-0=IR25-21

Rtc4-0=IR20-16

Rs R=T3(Addu+Subu+Ori+Beq+J)

Rt R=T3(Addu+Subu+SII+Beq)+T4Sw

M3 1=T3(Addu+Subu+Ori+Beq+Lw+Sw)

M3 0=T1+T4Beq

M4 1=T1+T4Beq

 $M4_0=T1+T3(Ori+Lw+Sw)$ 



ALUC2=0

ALUC1=T3(Ori+SII)

ALUC0=T3(Subu+SII+Beq)

Zin=T1+T3(Addu+Subu+Ori+SII+Lw+Sw)+T4Beq

Zout=T2+T4(Addu+Subu+Ori+SII+Lw+Sw)+T5Beq

M5=T4Lw

Rdc4-0=IR15-11(Addu+Subu+SII)+IR20-16(Ori+Lw)

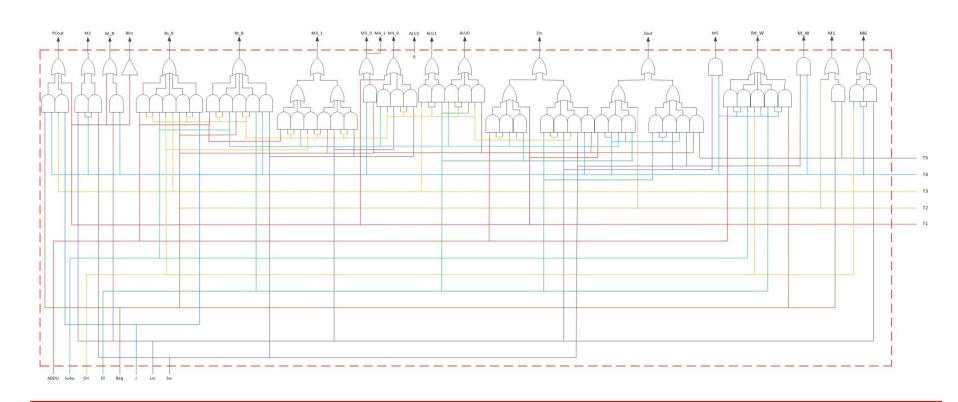
Rd\_W=T4(Addu+Subu+Ori+SII+Lw)



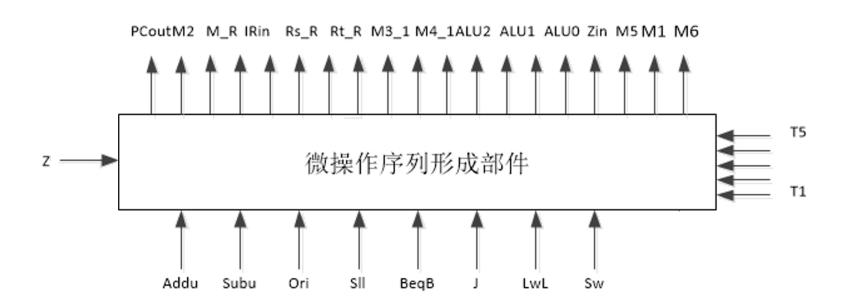
M\_W=T4Sw M1=T2+T5BeqZ M6=T4(Ori+Lw)



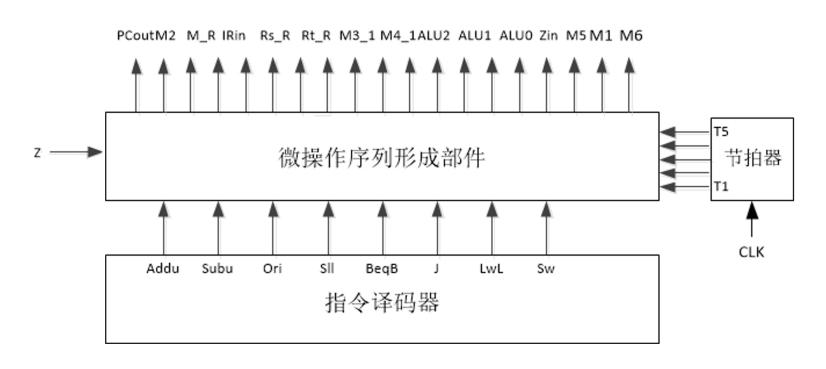
#### 5.构成微操作序列形成部件的组合逻辑网络

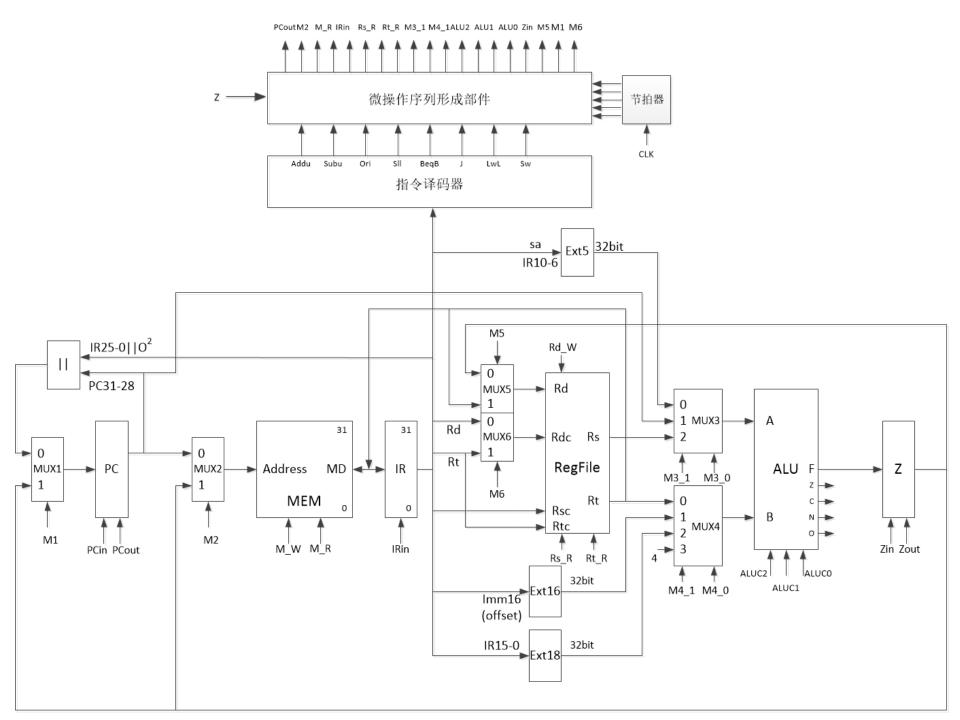








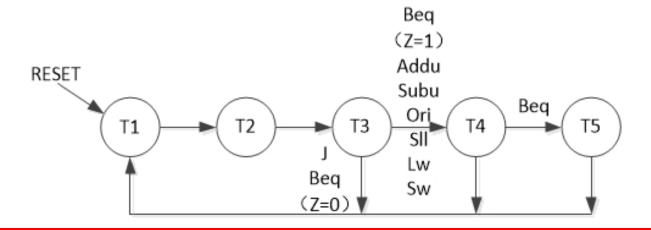






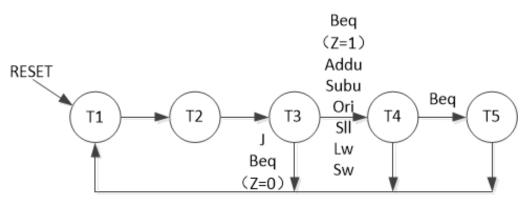
#### ● 多周期CPU的控制部件的状态转移图

以8条基础指令为例,从操作时间表中可以看到,跳转指令j用三个周期,条件转移指令beq用五个周期,其余指令addu、subu、ori、sll、Lw、Sw均用四个周期。五个状态分别有用T1、T2、T3、T4、T5表示,有限状态转移图如下图:





#### ● 8条指令有限状态机状态转换表



当前	输入				下个
状态					状态
T	Addu, Subu, Ori,	J	Beq	Z	T'
	S11, Lw, Sw				
T1	X	X	X	X	T2
T2	X	X	X	X	Т3
Т3	0	0	1	0	T1
Т3	0	0	1	1	T4
Т3	0	1	0	0	T1
Т3	1	0	0	0	T4
T4	1	X	0	X	T1
T4	0	X	1	X	T5
T5	X	X	X	X	T1



#### 状态转换的逻辑表达式:

<u></u>	i前状:	态	输入	下个状态					
t2	t1	t0	Addu, Subu, Ori, Sll, Lw, Sw	J	Beq	Z	t2'	t1'	t0'
0	0	0	х	Х	Х	Х	0	0	1
0	0	1	х	Х	х	Х	0	1	0
0	1	0	0	0	1	0	0	0	0
0	1	0	0	0	1	1	0	1	1
0	1	0	0	1	0	0	0	0	0
0	1	0	1	0	0	0	0	1	1
0	1	1	1	Х	0	Х	0	0	0
0	1	1	0	Х	1	Х	1	0	0
1	0	0	Х	Х	Х	Х	0	0	0

$$t0' = \overline{t2}\,\overline{t1}\,\overline{t0} + \overline{t2}\,t1\,\overline{t0}\,Beq\,Z + \overline{t2}\,t1\,\overline{t0}\,(Addu + Subu + Ori + Sll + Lw + Sw)$$

$$t1' = \overline{t2}\,\overline{t1}\,t0 + \overline{t2}\,t1\,\overline{t0}\,Beq\,Z + \overline{t2}\,t1\,\overline{t0}\,(Addu + Subu + Ori + Sll + Lw + Sw)$$

$$\mathsf{t2'} = \overline{\mathsf{t2}}\,\mathsf{t1}\,\mathsf{t0}\,\mathsf{Be}\,q$$



#### ● 多周期CPU控制部件逻辑结构

