

TEC-2指令系统概述

TEC-2机指令系统，采用6位操作码，故最多支持64条基本指令，其中53条指令已由设计者实现，微程序固化在ROM芯片中，其余11条留给实验人员执行实现。

TEC-2机的基本指令格式比较固定。从指令长度区分，有单字指令和双字指令，用户也可以实现三字指令；从操作数的个数区分，有无操作数指令、单操作数指令和双操作数指令；从支持的基本寻址方式区分，有寄存器寻址、寄存器间接寻址、立即数寻址、变址寻址、相对寻址、绝对寻址和堆栈寻址等方式；从指令功能上看，最常用的指令类型和运算是比较齐全的。

为了方便理解和记忆，把TEC-2机的指令格式归纳为如下形式：

15	10	9	8	7	4	3	0
操作码		条件码	目的寄存器 DR		源寄存器 SR		
			入、出端口地址/相对转移的位移量				
立即数/绝对地址/变址位移量							

单字指令仅用一个指令字。

双字指令用两个指令字，此时第二个指令字的内容可能是立即数、一个绝对地址或一个变址位移量。

第一个指令字分为三个主要部分。

最高6位是操作码。从这个意义上讲，TEC-2机的基本指令是固定长度的操作码结构。最多支持64条基本指令。

中间的两位，即9、8两位是条件码，只能把他用作条件转移指令的判断条件。这两位的值为00、01、10和11时，分别选择以处理机状态字中的C、Z、V和S的值作为判断条件。从这个意义上讲，我们也可以认为这两位是指令的扩展操作码。由于除了条件转移指令之外，其余的指令均不使用这两位，故我们可以用这两位扩展新的指令，而不会影响原来的指令功能。

最低的8位有多种用法：

(1) 这8位用于给出入/出指令的入/出端口地址。例如，已规定第一个串行口地址为80h和81h，第二个串行口地址为82h和83h。并行口、DMA口等都可以被分配几个合理的地址，以实现入/出接口的各种实验。

(2) 这8位用于给出相对寻址的位移量，其范围是从-128到+127之间，因此相对地址应在当前指令地址向前向后总共256个字的范围之内。实现相对地址计算时，这个位移量的最高位用符号位，补码形式，与16位的当前指令地址（放在IP寄存器中，即增量前的PC值）相加时，这一符号位要扩展到15~8这高8位上去。

(3) 这8位被分为两个4位的字段，用于给出的所用的通用寄存器编号。对双操作数指令，这里可是给出目的与源两个操作数所在的寄存器编号。对单操作数指令，只用源或目的中的一个操作数，此时，可能用到某一个4位字段，另一个4位字段不用。需要强调的一点是，寄存器用于给出操作数、操作数地址，或用作变址寄存器。

参考资料：胡敏、钱兴贤《计算机组成接口与通信实验指导》

TEC-2机指令系统详述

从操作数个数和指令功能两个角度，把TEC-2机基本指令区分为如下6类分别说明：

1. 无操作数指令，共11条

	15	10 9	8 7	4 3	0
格式:	OP	不用	不用	不用	

NOP	空操作
PSHF	状态字入栈
POPF	状态字出栈
EI	开中断, $INTE \leftarrow 1$
DI	关中断, $INTE \leftarrow 0$
STC	进位置1, $C \leftarrow 1$
CLC	进位置0, $C \leftarrow 0$
RET	子程序返回, $PC \leftarrow [SP]$, $SP \leftarrow SP-1$
IRET	中断返回, $STR \leftarrow [SP]$, $SP \leftarrow SP-1$, $PC \leftarrow [SP]$, $SP \leftarrow SP-1$
LDMC	装入微指令代码, 所用参数为 R1: 微码在主存中首地址 R2: 微指令条数 R3: 微码写入的控存地址, 把主存中给出的一段微代码写入控存中。
HALT	动态停机指令, $PC \leftarrow IP$

2. 单操作数指令，有两种格式，共12条

	15	10 9	8 7	4 3	0
格式 1:	OP	不用	不用	SR	

MUL SR	无符号乘, $R1 * SR \rightarrow R0R1$, 根据R1的值置状态位
DIV SR	无符号除, $R0R1 / SR \rightarrow R0$ (余数) R1 (商), 根据R1的值置状态位

	15	10 9	8 7	4 3	0
格式 2	OP	不用	DR	不用	

PUSH DR	压入DR
---------	------

POP DR	弹出DR
INC DR	$DR \leftarrow DR + 1$
DEC DR	$DR \leftarrow DR - 1$
NOT DR	DR求反, $DR \leftarrow DR$
SHL DR	DR左移, 最低位补0, 最高位移入C
ASR DR	DR算术右移, 最高位保持不变, 最低位移入C
SHR DR	DR逻辑右移, 最高位补入0, 最低位移入C
RCL DR	DR与C循环左移, C移入最低位, 最高位移入C
RCR DR	DR与C循环右移, C移入最高位, 最低位移入C

3. 双操作数指令, 有两种格式, 共17条

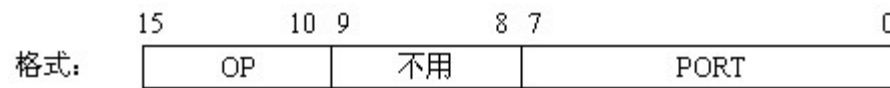
	15	10 9	8 7	4 3	0
格式 1	OP	不用	DR	SR	

ADD DR, SR	$DR \leftarrow DR + SR$
ADC DR, SR	$DR \leftarrow DR + SR + C$
SUB DR, SR	$DR \leftarrow DR - SR$
SBB DR, SR	$DR \leftarrow DR - SR - C$
CMP DR, SR	$DR - SR$
AND DR, SR	$DR \leftarrow DR \text{ and } SR$
OR DR, SR	$DR \leftarrow DR \text{ or } SR$
XOR DR, SR	$DR \leftarrow DR \text{ xor } SR$
TEST DR, SR	$DR \& SR$
MOV DR, SR	$DR \leftarrow SR$
MOV DR, [SR]	$DR \leftarrow [SR]$
MOV [DR], SR	$[DR] \leftarrow SR$

	15	10 9	8 7	4 3	0
格式 2:	OP	不用	DR	SR	
	DATA/ADR				

MOV DR, DATA	$DR \leftarrow DATA$
MOV DR, [ADR]	$DR \leftarrow [ADR]$
MOV [ADR], SR	$[ADR] \leftarrow SR$
MOV DR, DATA[SR]	$DR \leftarrow [DATA + SR]$
MOV DATA[SR], DR	$[DATA + SR] \leftarrow DR$

4. I/O指令, 输入输出指令各一条

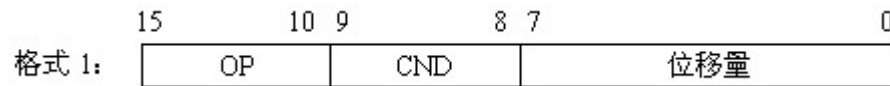


IN PORT $R0 \leftarrow [PORT]$, 从外设读入一字节到R0低8位

OUT PORT $[PORT] \leftarrow R0$, 把R0的低8位数据写到外设

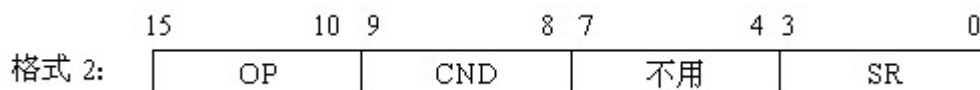
5. 转移指令，三种格式，共6条

相对转移指令



JR ADR 无条件相对转移到ADR, ADR为原PC值+位移量

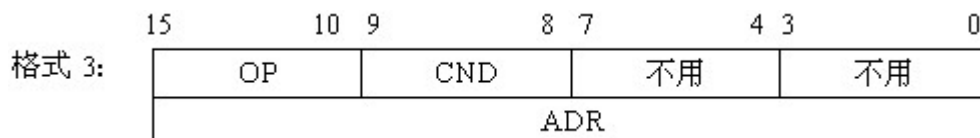
JR CND, ADR 当条件满足时相对转移到ADR, ADR为原PC值+位移量



JP SR 无条件转SR所指的地址

JP CND, SR 当条件满足时转SR所指的地址

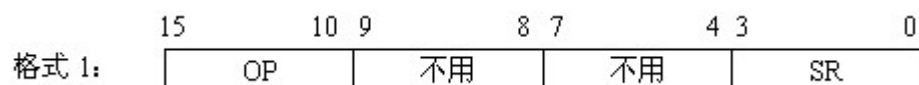
按绝对地址实现的转移指令



JP ADR 无条件转移到ADR地址

JP CND, ADR 当条件满足时转移到ADR

6. 子程序调用指令，两种格式，共两条



CALL SR 调用SR指明的子程序, $SP \leftarrow SP-1$, $[SP] \leftarrow PC$, $PC \leftarrow SR$



CALL ADR 调用通过ADR指明的子程序, $SP \leftarrow SP-1$, $[SP] \leftarrow PC$,
 $PC \leftarrow ADR$

上述指令说明中用到的符号包括:

SR 源操作数寄存器	PC 程序计数器
DR 目的操作数寄存器	IP PC增量前的值, 当前指令地址
OP 指令操作码	SP 堆栈指针
CND 条件转移指令所用的判断条件。	

对于无条件转移指令, CND位无用。这里要特别说明的一点, 是条件码CND为两位, 编码00, 01, 10, 11分别表明要判C, Z, V, S四个标志位, 而条件转移指令操作码的最低1位用来表明是按这某一标志位为0还是为1才能实现转移, 即同一条汇编指令包含两个操作码子, 如JP C, SR和JP NC, SR汇编指令名均为JP, 但依据条件分别是C和NC, 其指令操作码分别是100010和100011。

参考资料: 胡敏、钱兴贤《计算机组成接口与通信实验指导》