

调度程序说明

上海市计算技术研究所四室

1976.3.

调度程序的说明

在《731机操作系统简介》中已对这部分工作作了文字说明，这里给出有关框图的功能说明，工作单元说明和大部分框图，从功能说明中已经清楚的一些程序在此没有给出框图，它们都是比较次要的内容。

一. 框图功能说明

编号	程序名称	内存地址	功能说明
1	SELECT	62,0000 其中62的内存为20000, 以下同	机器出现了一级、二级中断,并转入本系统处理后,一般都转到这个选题程序。根据排队表,由它选出一个最优先的程序,恢复它的现场,并转入该程序工作。
2	SELECT1	62,0002	SELECT的一个内部入口,它根据排队表开始选题。
3	SELECT14	62,3100	从外部设备加工和传输程序中选题。
4	LOAD A	62,2440	从候算表A中装入一个题目。
5	LOAD B	62,2442	从候算表B中装入一个题目
6	LOAD	62,2310	对候算表中的题目作一些是否可以装入的检查
7	LOAD 1	62,2344	装入人工题目
8	LOAD 2	62,2400	装入自动题目
9	PRO	62,0700	对一个可装入的题目,进行具体的装入工作。前面5个子程序不过是作些准备工作。613中放好候算表中可装入的题目的地址,614放候算表的信息,为WN或WN4或WNB的内容。
10	JEND	63,2120	结束处理程序。进行题目的脱机绘图和脱机打印,做一些题目的扫尾工作,打印题目的统计信息,作系统的统计和该题目全部撤出。

续上表

编号	程序名称	内存地址	功能说明
11	OFFIN	7600	脱机输入后转来,置初始文件进入标志。如果二个题目队列有空,则引进该题目。凡是脱机输入的纸带,有关的题目必须通过排序命令预先排队。进入时L ₃ 中放题目名。
12	DW TJ	62,3000	为撤消候算表中的一个题目的命令ENDJOB < 题目名 > 提供的子程序。题目名在人机通讯的单元PAR (61,1054) 中。
13	PMI	62,3200	SELECT中的子程序,把监督信息区分配表中有关单元的48位清0,并把最后一批未满一个监督信息区的内容送脱印文件。
14	MRSRi		i=1,2,3,4,5,6。修改排队表状态的6个子程序。L ₃ 的0字节放要改变的状态,1字节放程序道号。其它字节均为0。i=1,2,3时,L ₃ 的0字节在要改变的状态位放"1",其余为0。i=4,5,6时,L ₃ 的0字节在要改变的状态位放"0",其余为1。
	MRSR1	7720	置申请
	MRSR2	7722	置挂位
	MRSR3	7724	置排不上队位
	MRSR4	7726	去申请
	MRSR5	7730	去挂位
	MRSR6	7732	去排不上队位

编号	程序名称	内存地址	功能说明
15	MRSRJ MRSR7 MRSR8 MRSR9 MRSR10 MRSR11 MRSR12	7700 7702 7704 7706 7710 7712	$j = 7, 8, 9, 10, 11, 12$ 。MRSRJ类似于上面的MRSR i , 而 j 对应于 i ($i = j - 6$)。不同的是 L_3 的1字节为0, 而 L_2 放题目名。返回时, $\omega = 0$ 表示题目名正确; $\omega \neq 0$, 表示题目名有错。
16	MTN	7761	MRSRJ使用的子程序。把题目名转换成道号。
17	ONCP	62, 2740	控打报到处理程序。由中断管理转来。
18	CR	62, 2036	对于要装入的题目, 检查是否能满足其内存和鼓的资源要求。
19	CR1	62, 2061	CR的一个内部入口, 只检查是否能满足内存的资源要求。
20	R3	62, 1762	RESOURCE1和RESOURCE2的一个公用子程序。
21	RESOURCE1	62, 2005	编译程序结束后, 根据目的程序的要求对内存重新分配。
22	RESOURCE2	62, 2771	编译程序结束后, 根据目的程序的要求对鼓重新分配。
23	CN	62, 3120	供OFFIN使用。检查输入的题目名是否在候算表中已经登记。

续上表

编号	程序名称	内存地址	功能说明
24	C TIME	62,3165	检查人工题目的开始时间是否大于或等于当地时间
25	ALLOCATOR	62,2630	修改内存的资源现状
26	CJMT	62,3040	根据新的排队字位置(在b15中)和老的排队字位置(在b13中)修改道号器JM和道号表JMT的内容。
27	CJMTB	62,3044	功能同上,不同的是b13中信息没有,而在b14中放目的程序的道号
28	CJMT1	62,3050	功能同上,不同的是,它只修改道号表。
29	COPY	62,3160	记副本目的程序,待增加
30	ERR1 ERR2	62,3140 62,3141	为打印信息用的子程序。ERR1打一个信息,即信息编号,ERR2打信息、编号和L2中的内容,它可以是题目名,也可以是别的内容。
31	PPSSI	62,1300	为打印系统统计信息的命令 PSSI 提供的加工程序
32	PJSI	62,1544	为JEND中打印题目统计信息的加工程序。

续上表

编号	程序名称	内存地址	功能说明																		
33	S3I	62,2530	为打印年月、日或时、分、秒时使用的子程序 L ₃ 中放 <table border="1"><tr><td>C1</td><td>C2</td><td>C3</td><td>C4</td><td>C5</td><td>C6</td></tr></table> 年/时 月/分 日/秒 L ₁ 的右指令放 5514 C。字符C可是：或 / 或 + 三种，结果在缓冲区中增加二个信息： <table><tr><td>□</td><td>□</td><td>□</td><td>□</td><td>C₁</td><td>C₂</td></tr><tr><td>C</td><td>C₃</td><td>C₄</td><td>C</td><td>C₅</td><td>C₆</td></tr></table>	C1	C2	C3	C4	C5	C6	□	□	□	□	C ₁	C ₂	C	C ₃	C ₄	C	C ₅	C ₆
C1	C2	C3	C4	C5	C6																
□	□	□	□	C ₁	C ₂																
C	C ₃	C ₄	C	C ₅	C ₆																
34	SADDB	62,2445	把 L ₀ 的内容送缓冲区，缓冲区的计数器加 2。																		
35	CYSZJ	62,2450	字节计数器 616 减 1，把 L ₂ 的 0 字节送到 L ₀ 的 (616) ₁₋₃ 字节。																		
36	DETOBI	62,2470	二位数的 2 → 10，初值在 L ₀ 中，浮点形式，结果也在 L ₀ 中。形式为 □ □ □ □ C ₁ C ₂																		
37	DETOBI1	62,2520	二位数的 2 → 10，初值在 L ₀ 的低位部分，定点形式，48 位可以是 0。结果同 DETOBI。																		
38	FORSPACE	62,2454	一个数据的位数若小于打印的位数，则前面填充。																		
39	TIMEO	62,2550	把某题目（它的排版字地址由 b16 指出）的中央处理机的时间（以秒为单位）化成字符形式： <table border="1"><tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr></table> 时 分 秒 保存在 L ₃ 和 d ₁ 中。																		

编号	程序名称	内存地址	功能说明																																											
40	TIME1	62,2557	功能同上,中央处理机的时间已放在L ₀ 中,定点形式,48位可以是0。																																											
41	SEVENT	62,2650	9个字节的2$\rightarrow$10. 9个字节存放形式如下: d₄: <table border="1"><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> c₉ c₈ c₇ d₃: <table border="1"><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> c₆ c₅ c₄ c₃ c₂ c₁ 加工结果在缓冲区中增加5个信息. 形式如下: <table border="1"><tr><td>xxx=xx</td><td>+xx+xx</td><td>+xx+xx</td><td>+xx+xx</td><td>+xx+xx</td></tr><tr><td><u>c₁</u></td><td><u>c₂</u></td><td><u>c₃</u></td><td><u>c₄</u></td><td><u>c₅</u></td></tr><tr><td></td><td></td><td></td><td><u>c₆</u></td><td><u>c₇</u></td></tr><tr><td></td><td></td><td></td><td></td><td><u>c₈</u></td></tr><tr><td></td><td></td><td></td><td></td><td><u>c₉</u></td></tr></table> 等号左边的数为右边9个数的总和:																			xxx=xx	+xx+xx	+xx+xx	+xx+xx	+xx+xx	<u>c₁</u>	<u>c₂</u>	<u>c₃</u>	<u>c₄</u>	<u>c₅</u>				<u>c₆</u>	<u>c₇</u>					<u>c₈</u>					<u>c₉</u>
xxx=xx	+xx+xx	+xx+xx	+xx+xx	+xx+xx																																										
<u>c₁</u>	<u>c₂</u>	<u>c₃</u>	<u>c₄</u>	<u>c₅</u>																																										
			<u>c₆</u>	<u>c₇</u>																																										
				<u>c₈</u>																																										
				<u>c₉</u>																																										
42	L ₂ /SEVENT	62,2710	L₂是SEVENT中的一个子程序入口。L₁中放三个字符: <table border="1"><tr><td>q₃</td><td>q₂</td><td>q₁</td><td></td><td></td><td></td></tr></table> 结果在缓冲区中增加2个信息. 形式如下: <table border="1"><tr><td><u> </u></td><td><u> </u></td><td><u> </u></td><td><u>+xx</u></td><td><u>+xx+xx</u></td></tr><tr><td></td><td></td><td></td><td><u>a₁</u></td><td><u>a₂</u></td></tr><tr><td></td><td></td><td></td><td></td><td><u>a₃</u></td></tr></table>	q ₃	q ₂	q ₁				<u> </u>	<u> </u>	<u> </u>	<u>+xx</u>	<u>+xx+xx</u>				<u>a₁</u>	<u>a₂</u>					<u>a₃</u>																						
q ₃	q ₂	q ₁																																												
<u> </u>	<u> </u>	<u> </u>	<u>+xx</u>	<u>+xx+xx</u>																																										
			<u>a₁</u>	<u>a₂</u>																																										
				<u>a₃</u>																																										
43	SHIFT	62,2610	把L ₂ . L ₃ 中的12个字节的内容全部加起来, 2 $\rightarrow$ 10. 送缓冲区																																											
44	引导程序	116040	系统调试用。输入符6纸带,纸带开始穿内存地址,最多6位八进制数,表示16位字地址。再穿一个冒号,接着穿输入的内容。每个单元(字)的内容穿16个八进制数,用逗号作分隔符。若地址是连续的,除第一个内容前面必须穿内存地址外,接下来内容前面可不必穿地址,而地址不连续时,必须穿地址和冒号。回车换行等字符跳过。最后穿11个以上的废码"#"和几个停码。																																											

续上表

编号	程序名称	内存地址	功能说明								
45	八进制输出程序	116400	系统调试用。八进制形式打印一组内存的信息,内存的位置由L₂给出,形式为: <table border="1"> <tr> <td>0</td><td>开始地址</td><td>0</td><td>结束地址</td></tr> <tr> <td>41</td><td></td><td>25</td><td>17</td></tr> </table> 输出的形式为32行一页。每页输出64个单元的内容,左边一半,右边一半。每个单元内容前有6位地址。	0	开始地址	0	结束地址	41		25	17
0	开始地址	0	结束地址								
41		25	17								

二 工作单元说明

工作单元符号	内存地址	说明
bd ₁	6776	变址器暂存单元
bm	7546	小档时的资源要求:1~5位是内存块数(9块),6~15位是鼓的组数(512组)。内存的块数为实际要求减1,每块2048字。
bn	7020	已经引入的题目中,自动题目的总数
bom	7014	人工/自动的标志。前者为0,后者为1。
CP1	7000	表示控制台使用情况。右边的1~24位表示7通道,左边的25~48位表示6通道。通道若占用,则相应位置就放占用该通道的解释程序在排队表中的地址,否则为0。

续上表

工作单元符号	内存地址	说 明										
CP ₂	7002	类似于CP ₁ ,也是表示控制台情况的。0字节和1字节的后3位分别放6通道和7通道解释程序的道号。第8和第16位为好坏标志,1表示好,0表示坏。不占用时相应的字节的后7位为全1。										
CP ₃	7034	表示时钟屏蔽情况,初值是23位为1,其余位均为0。对时命令后,变23位的1为0,同时运行表中相应位也变成0,即开放时钟中断。										
d, d ₁ -d ₆		临时工作单元										
JM	7022	道号器。0~3字节初态为0,如果有题目引进,且道号(道号的后三位)为 <i>i</i> (<i>i</i> =0,1,2,3),则在第 <i>i</i> 字节填上该题目在排队表中的相对(与6740之差)位置。它可能有四个值2,4,10,12(八进制)。另外二个字节放空闲,内存的上下界,第5字节放上界,第4字节放下界。										
JMT-LD	7060	JMT为道号表, JMT-LD 取表 JMT-ST 送表 为了从程序状态字中的“程序道号”较快地取到与这个程序道号相对应的那个题目的排队字,以及把修改好的内容送到排队字,建立了这个道号表,其内容在引进题目时形成,形式如下:										
JMT-ST	7064											
		<table border="1"> <tr> <td rowspan="2">取 7060 →</td><td>把0道题目的排队字送L₂</td><td>把1道题目的排队字送L₂</td></tr> <tr> <td>把2道题目的排队字送L₂</td><td>把3道题目的排队字送L₂</td></tr> <tr> <td rowspan="2">送 7064 →</td><td>把L₂的内容送0道题目的排队字</td><td>把L₂的内容送1道题目的排队字</td></tr> <tr> <td>把L₂的内容送2道题目的排队字</td><td>把L₂的内容送3道题目的排队字</td></tr> </table>	取 7060 →	把0道题目的排队字送L ₂	把1道题目的排队字送L ₂	把2道题目的排队字送L ₂	把3道题目的排队字送L ₂	送 7064 →	把L ₂ 的内容送0道题目的排队字	把L ₂ 的内容送1道题目的排队字	把L ₂ 的内容送2道题目的排队字	把L ₂ 的内容送3道题目的排队字
取 7060 →	把0道题目的排队字送L ₂	把1道题目的排队字送L ₂										
	把2道题目的排队字送L ₂	把3道题目的排队字送L ₂										
送 7064 →	把L ₂ 的内容送0道题目的排队字	把L ₂ 的内容送1道题目的排队字										
	把L ₂ 的内容送2道题目的排队字	把L ₂ 的内容送3道题目的排队字										

续上表

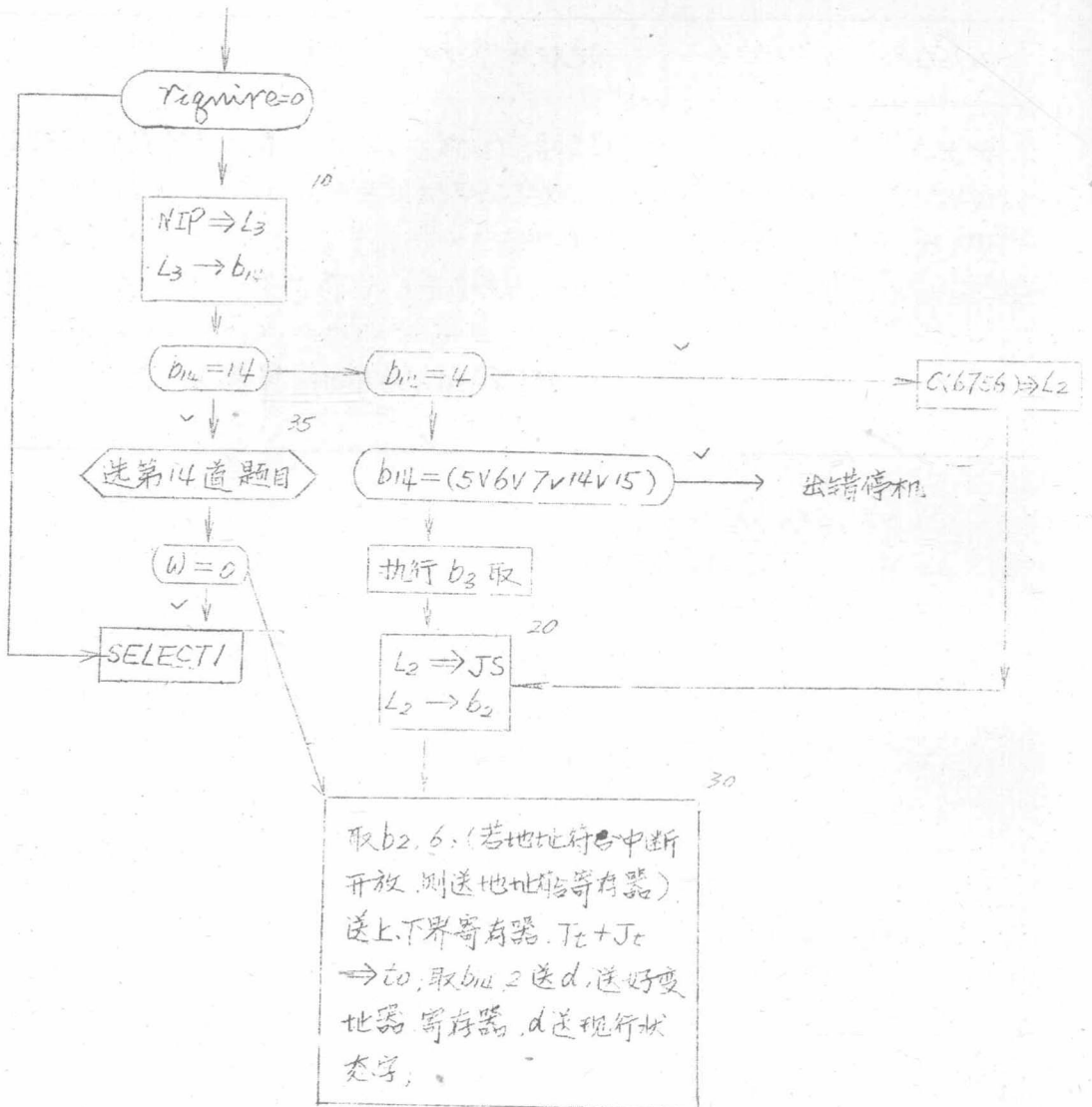
工作单元符号	内存地址	说 明								
		<table><tr><td>把8道题目的排队字送L₂</td><td>把9道题目排队字送L₁</td></tr><tr><td>把10道题目的排队字送L₂</td><td>把11道题目的排队字送L₂</td></tr><tr><td>把L₂的内容送8道题目的排队字</td><td>把L₂的内容送9道题目的排队字</td></tr><tr><td>把L₂的内容送10道题目的排队字</td><td>把L₂的内容送11道题目的排队字</td></tr></table>	把8道题目的排队字送L ₂	把9道题目排队字送L ₁	把10道题目的排队字送L ₂	把11道题目的排队字送L ₂	把L ₂ 的内容送8道题目的排队字	把L ₂ 的内容送9道题目的排队字	把L ₂ 的内容送10道题目的排队字	把L ₂ 的内容送11道题目的排队字
把8道题目的排队字送L ₂	把9道题目排队字送L ₁									
把10道题目的排队字送L ₂	把11道题目的排队字送L ₂									
把L ₂ 的内容送8道题目的排队字	把L ₂ 的内容送9道题目的排队字									
把L ₂ 的内容送10道题目的排队字	把L ₂ 的内容送11道题目的排队字									
JS	7030	放选中题目的排队字内容								
JW	7024	与JM有些类似, 0~3字节初态为全“1”。如果有题目引进, 且道号为i ($i=0, 1, 2, 3$) 则在第i字节放该题目在候算表中的相对位置, 字节的最高二位的意义为: 00表示 人工题目的候算表, 队列A, 01表示 人工题目的候算表, 队列B, 10表示 自动题目的候算表 低6位给出候算表的相对位置。另外二个字节放硬件具有的内存总容量, 第5字节放上界, 第4字节为下界。								
MA	7016	已经引入的题目中, 人工题目的总数。								
NIP	7026	被中断的那个程序的程序道号, 在最后四位								
Requiere	7012	只有二种状态: 0表示在一次中断以后, 排队表中的申请位均未改变, 可直接返回到原来被中断的程序继续执行; $\neq 0$表示申请位已有所改变, 必须从排队表中重新选出一个可执行的题目。								
SCHT	6740-6756	排队表								
SXSLP	7334	脱机输出文件名。常数。								

续上表

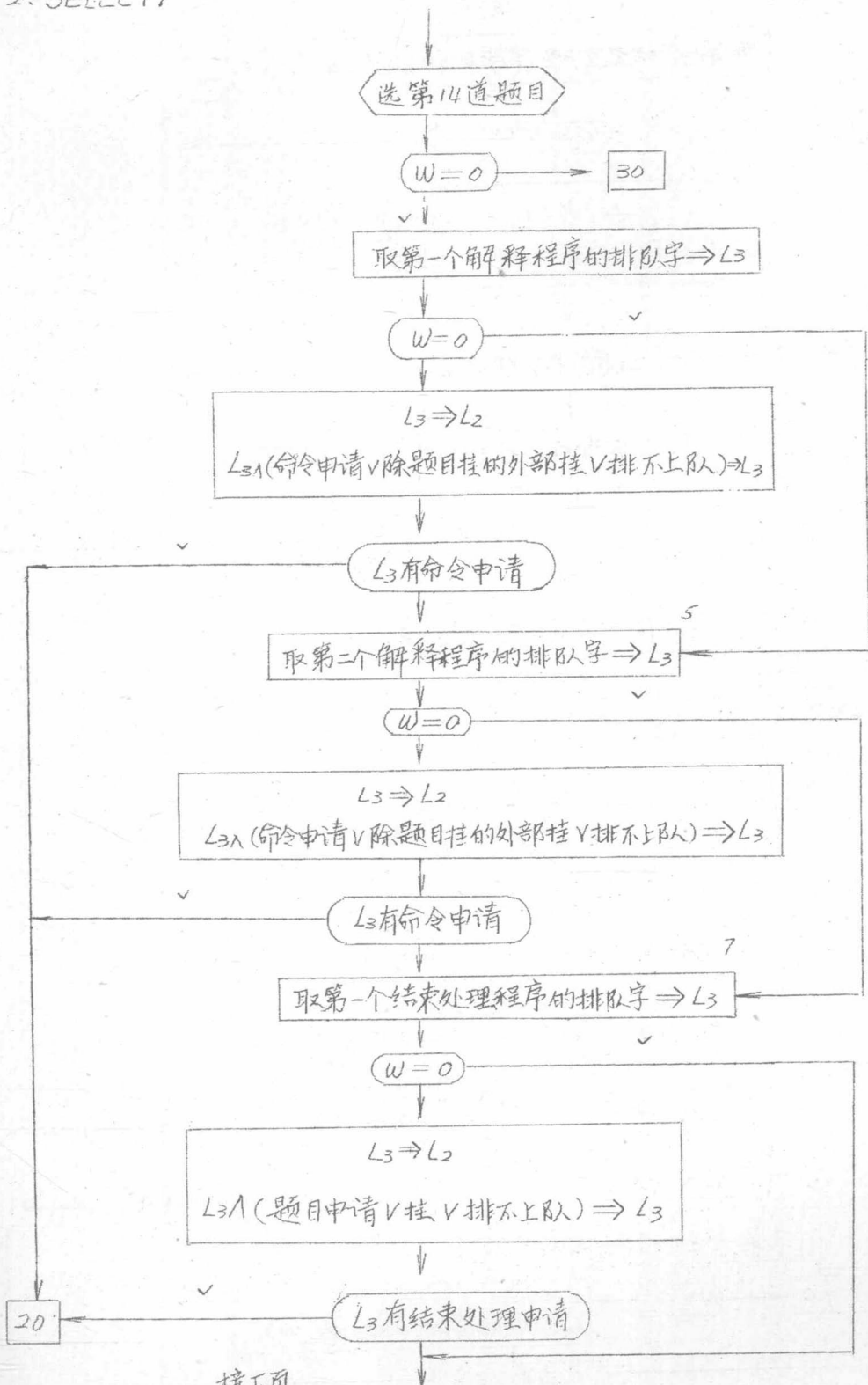
工作单元符号	内存地址	说明
SYSGP	7336	脱机绘图文件名。常数。
WN WNA WNB	7010 7004 7006	候算表计数器。1~6 位表示候算表中候算题目的个数,这里的个数为实际的2倍。即实际有 n 个题目,计数器中用 $2 \times n$ 表示。25~41 位表示各候算表的头地址。48 位是 6,7 通道标志: 0 表示 6 通道, 1 表示 7 通道。其它位均为 0。WNA, WNB 为人工候算表计数器, WN 为自动候算表计数器

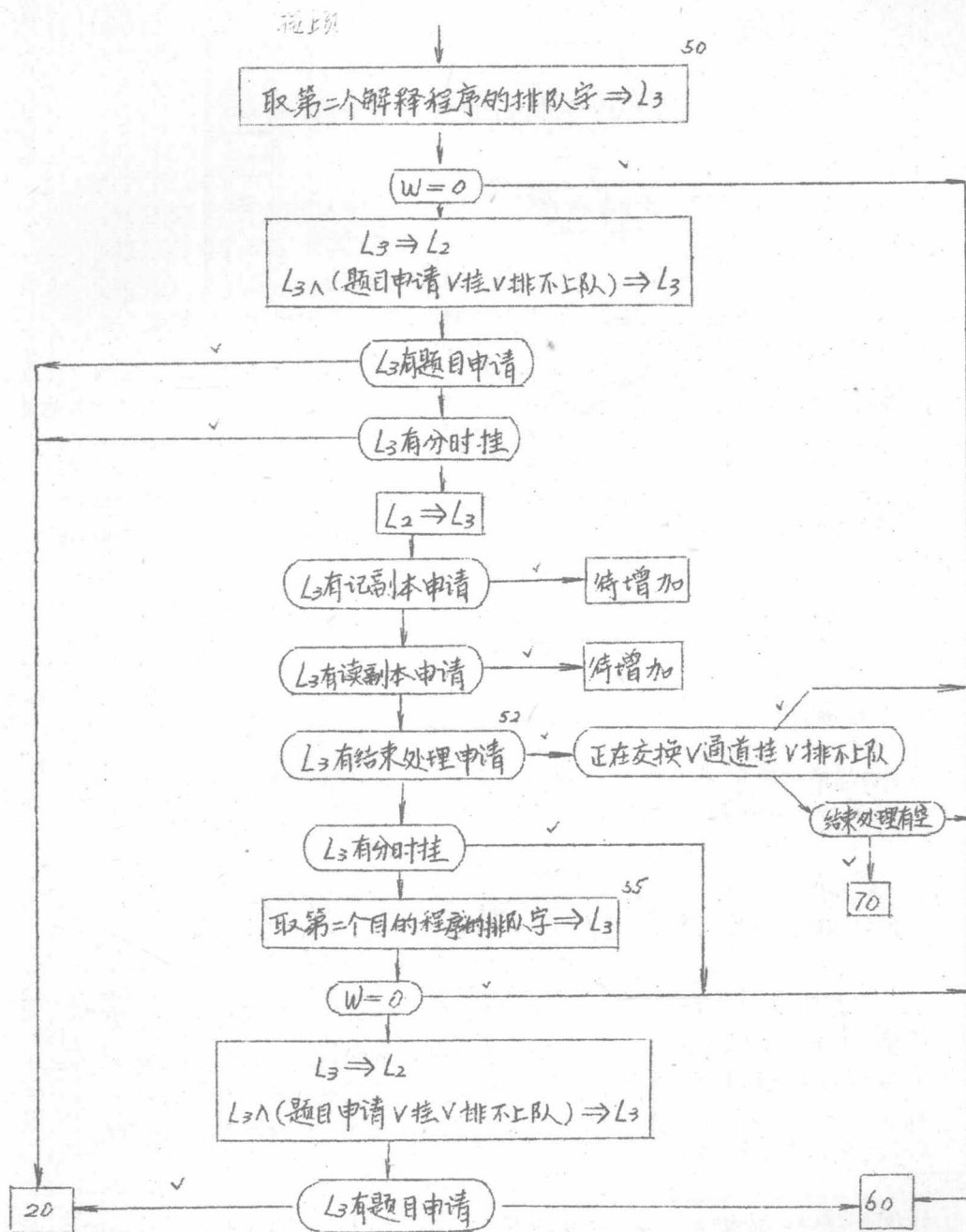
三. 框图

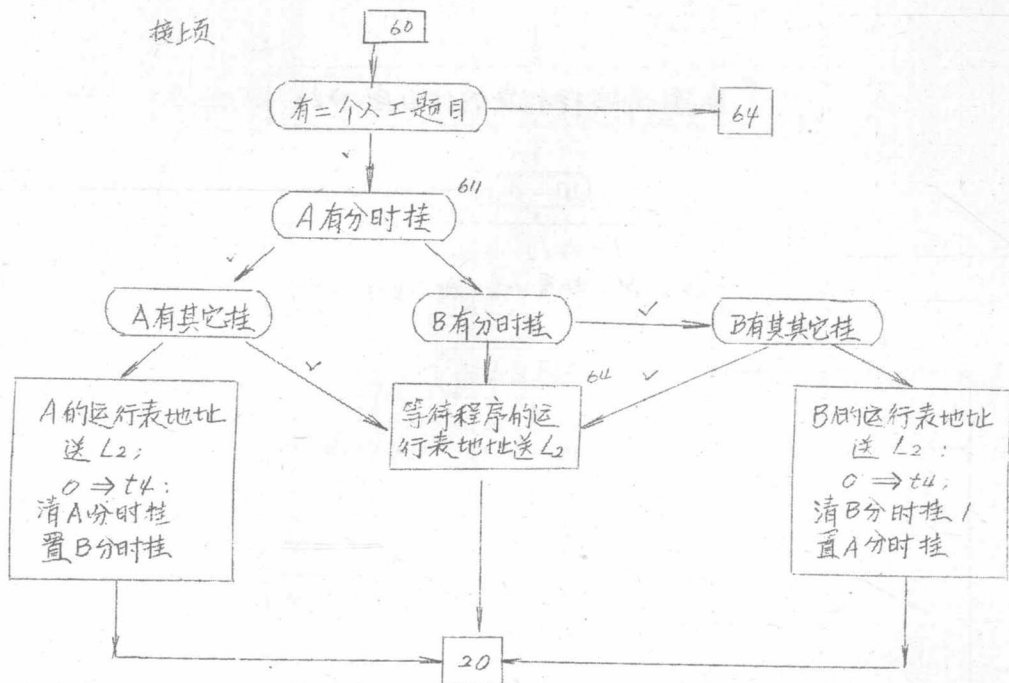
1. SELECT



2. SELECT1







A, B 分别为二个目的程序的排队字。

目的程序的道号 $\rightarrow 616$, $0 \Rightarrow R20 + (b16) \Rightarrow$ 赋值跟踪变量;
把最后一批监督信息送监督文件. 相应的监督信息区分配表第48位清零

清相应目的程序的内存, 释放控制台, 修改 CP1, CP2, 释放鼓,
置相应的候算表位置以“算完”标送。

送好结束处理程序的运行表(即已结束的那队题目); 解释程序名 $\Rightarrow$ 运行表第一个单元; 送好程序状态字, 解释程序和目的程序的输出时间, 带交换时间加起来送, 送时间限额(目的程序)和符合停机初值(全1);
排队字地址 $\Rightarrow b7$, 运行表地址送 $b8$; 7100 或 $7120 \rightarrow 615$; 释放内存; 修改 JM 的第4,5字节, 送 b_1, b_2, b_3, b_4, b_5 。

修改道号表和 JM 的 0, 1, 2, 3 中的某字节, 旧的解释程序的排队字和目的程序的运行表地址送新的排队字(即结束处理程序的排队字)。原来的排队字清零

接下页