

ASSIGN MICI, (FILE, MBS, IDOUCI)  
METASYM CI, LB

1		UFLAGS	SET	1	UHIFLG EQUIS
2	*S*		SYSTEM	UTS	
3		*			
4		*			
5		*M*	MBS		MULTI-BATCH RESOURCE SCHEDULER
6		*			
7		*			
8		*P*	NAME:		MBS
9		*P*			
10		*P*	PURPOSE:		SCHEDULES BATCH JOBS FOR EXECUTION BASED ON THEIR
11		*P*			RESOURCE REQUIREMENTS
12		*P*			
13		*P*	DESCRIPTION:		
14		*P*			THE MBS MODULE, PART OF THE RBBAT GHOST, CONTAINS TWO
15		*P*			MAIN ROUTINES, MBSOP (REPART) AND MBS. THE FIRST OF
16		*P*			THESE, CALLED AS MBSOP BY RBBAT TO OPERATE ON NEWLY
17		*P*			SUBMITTED JOBS AND AS REPART BY MBS TO ACT ON ALL JOBS
18		*P*			WHEN THE PARTITION ATTRIBUTES HAVE BEEN CHANGED, IS
19		*P*			RESPONSIBLE FOR SETTING THE BITS IN BHPART THAT TELL
20		*P*			WHETHER A GIVEN JOB IS ABLE TO RUN IN A GIVEN
21		*P*			PARTITION BASED ON THE MATCH OF RESOURCES. THE MAIN
22		*P*			ROUTINE, OF COURSE IS MBS. IT IS CALLED BY RBBATM
23		*P*			EITHER AS A RESULT OF A GHOST FUNCTION CODE IN A COMBU
24		*P*			REQUESTING AN MBS FROM THE MONITOR, OR WHEN A NEW
25		*P*			BATCH JOB HAS BEEN ADDED TO RBBAT'S TABLES. RBBATM
26		*P*			ALSO CALLS MBS TO START UP BATCH JOBS AFTER A CRASH.
27		*P*			MBS PERFORMS THE FOLLOWING TASKS:
28		*P*			
29		*P*			1) PLILK - THE PARTITION ATTRIBUTE LOCK IS TESTED.
30		*P*			IF IT IS NON ZERO CONTROL IS CHANGING PARTITION
31		*P*			ATTRIBUTES AND MBS EXITS. IF IT IS ZERO MBS SETS IT
32		*P*			TO PREVENT ATTRIBUTE CHANGES WHILE MBS IS RUNNING.
33		*P*			
34		*P*			2) PLICHG - THE PARTITION ATTRIBUTE CHANGE FLAG IS
35		*P*			TESTED. IF IT IS ZERO MBS CONTINUES. IF IT IS NON-
36		*P*			ZERO, IT CONTAINS SET BITS FOR EACH PARTITION WHOSE
37		*P*			ATTRIBUTES HAVE BEEN CHANGED, AND MBS CALLS REPART TO



75 \*P\*  
76 \*P\*  
77 \*P\*  
78 \*P\*  
79 \*P\*  
80 \*P\*  
81 \*P\*  
82 \*P\*  
83 \*P\*  
84 \*P\*  
85 \*P\*  
86 \*P\*  
87 \*P\*  
88 \*P\*  
89 \*P\*  
90 \*P\*  
91 \*P\*  
92 \*P\*  
93 \*P\*  
94 \*P\*  
95 \*P\*  
96 \*P\*  
97 \*P\*  
98 \*P\*  
99 \*P\*  
100 \*P\*  
101 \*P\*  
102 \*P\*  
103 \*P\*  
104 \*P\*  
105 \*P\*  
106 \*P\*  
107 \*P\*  
108 \*P\*  
109 \*P\*  
110 \*P\*  
111 \*P\*

SELECTED.

8) THE ACCOUNT OF THE CJ IS CHECKED AGAINST THOSE OF ALL RUNNING JOBS TO SEE IF EITHER THE CJ OR A RUNNING JOB SPECIFIED ACCOUNT. IF THIS IS TRUE A NEW CJ IS SELECTED.

9) IF THE CJ SPECIFIED ORDER ALL RUNNING JOBS AND THOSE IN THE QUEUE ARE CHECKED AND IF ANY WITH THE SAME ACCOUNT HAVE A LOWER SYSID, A NEW CJ IS SELECTED

10) THE RESOURCES REQUESTED BY THE CJ ARE CHECKED AGAINST THOSE AVAILABLE IN THE SYSTEM AND IF ANY ARE UNAVAILABLE A NEW CJ IS SELECTED.

11) IF THE CJ SPECIFIED SERIAL NUMBERS FOR DISCS ON HIS LIMIT CARD THE FOLLOWING CHECKS ARE MADE:

A) ANY SERIAL NUMBER THE CJ REQUESTED EXCLUSIVE MUST NOT BE IN USE OR REQUESTED EITHER SHARED OR EXCLUSIVE BY ANY OTHER RUNNING USER.

B) ANY SERIAL NUMBER THE CJ REQUESTED SHARED MUST NOT BE IN USE OR REQUESTED BY ANY RUNNING JOB AS EXCLUSIVE.

C) THERE MUST BE FREE SPINDLES (BEYOND THOSE REQUESTED BY THE CJ) TO MOUNT ANY REQUESTED SHARED SPINDLES NOT ALREADY MOUNTED.

D) THERE MUST BE ROOM IN THE IN-CORE SERIAL NUMBER TABLES TO HOLD THE CJ'S SERIAL NUMBERS.

IF ANY OF THESE TESTS FAIL A NEW CJ IS SELECTED

12) THE CJ HAS NOW BEEN SELECTED TO RUN

112 \*P\*  
 113 \*P\*  
 114 \*P\*  
 115 \*P\*  
 116 \*P\*  
 117 \*P\*  
 118 \*P\*  
 119 \*P\*  
 120 \*P\*  
 121 \*P\*  
 122 \*P\*  
 123 \*P\*  
 124 \*P\*  
 125 \*P\*  
 126 \*P\*  
 127 \*P\*  
 128 \*P\*  
 129 \*P\*  
 130 \*P\*  
 131 \*P\*  
 132 \*P\*  
 133 \*P\*  
 134 \*P\*  
 135 \*P\*  
 136 \*P\*  
 137 \*P\*  
 138 \*P\*  
 139 \*P\*  
 140 \*P\*  
 141 \*P\*  
 142 \*  
 143 \*

13) A GETI TABLE SLOT IS REMOVED FROM THE FREE CHAIN AND THE JOBS RESOURCE INFORMATION (AND OTHER INFO) IS MOVED FROM THE RBBAT BATCH TABLES INTO IT.

14) AVRNGU IS INCREMENTED FOR ALL MOUNTED SHARED SPINDLES THE JOB HAS REQUESTED AND THE DEVICE OWNERSHIP BIT (ASPIN) IS SET FOR THESE SPINDLES. THESE BITS ARE TRANSFERED TO GIJASPN AND EVENTUALLY TO JIASPIN.

15) THE RESOURCES THAT THE JOB REQUESTED ARE ADDED TO THE BATCH CURRENT TOTALS

16) THE JOB'S SERIAL NUMBERS ARE MOVED TO THE CORE SERIAL NUMBER TABLES

17) THE JOB IS MOVED FROM ITS PRIORITY CHAIN TO RBBAT RUNNING CHAIN

18) THE BATCH BIT IS SET IN THE JOB'S UH:FLG2 AND MBS CALLS ADD1 TO START UP THE USER

19) FINALLY MBS RETURNS TO THREE ABOVE TO TRY TO START ANOTHER JOB.

REFERENCE I

MBS IS PART OF THE RBBAT GHOST WHICH ALSO CONTAINS RBBATM AND RBBATR. THE TABLES MBS USES (BATCH, GETI, RESOURCE, AVR, PARTITION, ETC.) ARE DESCRIBED IN THE DATA BASE TECH. MANUAL.

Line	Symbol	Page	Symbol	Description
144		PAGE		
145	*			
146	*			
147		DEF	MBS	THE MULTI-BATCH SCHEDULER
148		DEF	MBSOP	SET BHPART BITS FOR A GIVEN JOB BASED ON ITS RESOURCE REQUIREMENTS
149	*,*			
150	*			
151	*			
152		REF	M7	MASK
153		REF	M8	MASK
154		REF	YFF	MASK
155		REF	X80	TABLE BIT
156		REF	BT31T80	TABLE BITS
157		REF	NB31T80	TABLE BITS
158	*			
159		REF	TMPSTK	RBBAT'S STACK USED TO PUSH AND PULL
160		REF	DCT4	INPUT: DEVICE TYPE USED TO CHECK WHETHER A SERIAL NUMBER IS THE SAME RESOURCE TYPE AS ANOTHER
161	*,*			
162	*,*			
163		REF	UHIFLG	I/O: BATCH BIT IS STORED HERE TO MAKE NEW USER BATCH
164	*,*			
165		REF	SLIBPRI8	INPUT: BATCH DEFAULT EXECUTION PRIORITY PASSED TO ADD1 WHEN JOB IS STARTED
166	*,*			
167	*,*			
168	*			
169		REF	FIPRI	RBBAT IF! PRIORITY JOBS PRIORITY CHAIN INDEX
170	*,*			
171		REF	RUNPRI	RBBAT RUNNING JOBS PRIORITY CHAIN INDEX
172	*,*			
173	*			
174		REF	RRBIT	INPUT: RERUN BIT IN BWISDA
175		REF	JEBIT	INPUT: JOBERT SUBMITTED BIT IN BWISDA
176	*			
177		REF	SLIPI	INPUT: AMOUNT IN 256THS TO INCREMENT PRIORITY WHEN JOB IS PASSED OVER FOR SCHEDULING
178	*,*			
179	*,*			
180		REF	SIBUIS	I/O: NUMBER OF BATCH USERS RUNNING

181	*,*			USED TO DECIDE WHETHER TO SCHEDULE
182	*,*			INCREMENTED WHEN A NEW JOB IS STARTED
183		REF	SIBUAI5	INPUT: NUMBER OF BATCH USERS ALLOWED
184	*,*			TO RUN USED TO DECIDE WHETHER TO
185	*,*			SCHEDULE
186		REF	SIBFIS	I/O: NUMBER OF BATCH JOBS IN THE QUE
187	*,*			USED TO DECIDE WHETHER TO SCHEDULE.
188	*,*			DECREMENTED WHEN A NEW JOB IS RUN
189		REF	SIMBSF	OUTPUT: MULTI-BATCH SCHEDULE FLAG USE
190	*,*			WHEN MBS MUST WAIT FOR A GETI TABLE
191	*,*			ENTRY - CAUSES ANOTHER MBS IN 5 SEC
192	*			
193		REF	SNUL	STATE INDEX OF UNUSED USER SLOTS
194		REF	SB:HQ	I/O: STATE HEAD TABLE USED TO STEAL
195	*,*			USER NUMBER FOR NEW BATCH JOB
196		REF	SBITQ	I/O: STATE TAIL TABLE USED TO STEAL
197	*,*			USER NUMBER FOR NEW BATCH JOB
198		REF	UBIFL	I/O: USER STATE FLINK USED TO STEAL
199	*,*			USER NUMBER FOR NEW BATCH JOB
200		REF	UB:BL	I/O: USER STATE BLINK USED TO STEAL
201	*,*			USER NUMBER FOR NEW BATCH JOB
202	*			
203		REF	SV:RSIZ	HIGHEST INDEX INTO RESOURCE TABLES
204		REF	SB:RTY	INPUT: RESOURCE TABLE USED TO CONVER
205	*,*			RESOURCE INDEX TO DEVICE TYPE TO MOV
206	*,*			SERIAL NUMBER INFO TO CORE AND CHECK
207	*,*			WHETHER SERIAL NUMBERS ARE SAME DEVI
208		REF	SH:RTOT	INPUT: RESOURCE TABLE TOTAL ON SYSTE
209	*,*			USED TO CHECK HOW MANY CAN BE ALLOCA
210		REF	SH:ROCU	INPUT: RESOURCE TABLE ONLINE CURRENT
211	*,*			TO CHECK HOW MANY ARE IN USE
212		REF	SH:RGCU	INPUT: RESOURCE TABLE GHOST CURRENT
213	*,*			TO CHECK HOW MANY ARE IN USE
214		REF	SH:RBSUM	INPUT: RESOURCE TABLE MAX AVAILABLE
215	*,*			BATCH USED TO CHECK HOW MANY ARE
216	*,*			AVAILABLE
217		REF	SH:RBCU	I/O: RESOURCE TABLE BATCH CURRENT US

218	*,*			TO CHECK HOW MANY ARE AVAILABLE AND
219	*,*			INCREMENTED WHEN A NEW JOB IS SCHEDULED
220	*,*			THAT REQUESTED THEM
221		REF	SBIRBDF	INPUT: RESOURCE TABLE BATCH DEFAULT
222	*,*			USED TO KEEP B4WIRES UP TO DATE
223	*			
224		REF	AVRTBLSIZ	NUMBER OF TAPE ENTRIES IN THE AVR
225	*,*			TABLES USED TO FIND PACKS
226		REF	AVRTBLNE	NUMBER OF ENTRIES TOTAL IN THE AVR
227	*,*			TABLES USED TO FIND PACKS
228		REF	BATAPE	FIRST DCT INDEX THAT HAS AN AVR
229	*,*			TABLE ENTRY USED TO CONVERT DCTX
230	*,*			TO AVRX
231		REF	AVRTBL	INPUT: AVR SERIAL NUMBER/FLAG TABLE
232	*,*			WORD 0 USED TO COMPARE REQUESTED
233	*,*			SERIAL NUMBERS WITH RUNNING = WORD1
234	*,*			BIT 2 USED TO CHECK WHETHER WORD 0 CAN
235	*,*			BE TRUSTED
236		REF	AVRNOU	OUTPUT: AVR NUMBER OF USERS COUNTED
237	*,*			WHEN A JOB REQUESTS AN ALREADY MOUNTED
238	*,*			SHARED PACK
239		REF	AVRID	INPUT: AVR USER NUMBER = IF THIS IS
240	*,*			NON-ZERO THE PACK IS EXCLUSIVE
241	*			
242		REF	BH:HPRI	I/O: BATCH QUEUE TABLE IN RBBAT
243	*,*			PRIORITY HEADS USED TO MOVE FROM JOB
244	*,*			TO JOB AND RECHAIN TO NEW PRIORITIES
245		REF	BH:TPRI	I/O: BATCH QUEUE TABLE IN RBBAT
246	*,*			PRIORITY TAILS USED TO MOVE FROM JOB
247	*,*			TO JOB AND RECHAIN TO NEW PRIORITIES
248		REF	BH:LINK	I/O: BATCH QUEUE TABLE IN RBBAT
249	*,*			PRIORITY LINK USED TO MOVE FROM JOB
250	*,*			TO JOB AND RECHAIN TO NEW PRIORITIES
251		REF	BB:PI	I/O: BATCH QUEUE TABLE IN RBBAT
252	*,*			INCREMENTAL PRIORITY INCREASED BY
253	*,*			SLIP! WHEN JOB IS BYPASSED FOR
254	*,*			SCHEDULING

Line	Code	Field	Description
255		REF BBIRID	INPUT: BATCH QUEUE TABLE IN RBBAT
256	*,*		RBID USED TO CHECK LEGAL PARTITIONS
257	*,*		AND MOVED TO GIBIRID
258		REF BHISID	INPUT: BATCH QUEUE TABLE IN RBBAT
259	*,*		SYSID USED TO CHECK FOR ORDER RESTRI
260	*,*		AND MOVED TO PLHISID
261		REF BH:TIME	INPUT: BATCH QUEUE TABLE IN RBBAT
262	*,*		LIMIT CARD TIME USED TO SCHEDULE AND
263	*,*		MOVED TO GIH:TIM
264		REF BWISDA	I/O: BATCH QUEUE TABLE IN RBBAT
265	*,*		FLAGS AND STARTING DISC ADDRESS = JE
266	*,*		USED TO CHECK LEGAL PARTITIONS = SDA
267	*,*		MOVED TO GI:SDA = BWISDA IS CLEARED
268	*,*		UNLESS RRBIT IS SET
269		REF BD:ACCT	INPUT: BATCH QUEUE TABLE IN RBBAT
270	*,*		ACCOUNT USED FOR ACCOUNT CHECK
271		REF B&W:RES	I/O: BATCH QUEUE TABLE IN RBBAT
272	*,*		RESOURCES USED FOR SCHEDULING AND
273	*,*		KEPT UP TO DATE FOR DISPLAYS = MOVED
274	*,*		TO GI:RES = BYTE 0 IS PRIORITY MOVED
275	*,*		TO GIB:PRI
276		REF BH:SLNK	I/O: BATCH QUEUE TABLE IN RBBAT
277	*,*		SHARED SERIAL NUMBER HEAD USED TO MA
278	*,*		SERIAL NUMBER TESTS = MBS SORTS LIST
279	*,*		FOR RECOVERY AND RERUN PURPOSES
280		REF BH:XLNK	INPUT: BATCH QUEUE TABLE IN RBBAT
281	*,*		EXCLUSIVE SERIAL NUMBER HEAD USED TO
282	*,*		MAKE SERIAL NUMBER CHECKS
283		REF BH:PART	I/O: BATCH QUEUE TABLE IN RBBAT
284	*,*		PARTITION BITS: A BIT ON MEANS THE
285	*,*		JOB QUALIFIES FOR THE PARTITION. SE
286	*,*		BY MBSOP WHEN THE JOB IS SUBMITTED,
287	*,*		MODIFIED BY REPART WHEN PARTITION
288	*,*		ATTRIBUTES CHANGE, AND USED BY MBS
289	*,*		TO SEE IF THERE IS A PARTITION TO RU
290	*,*		THE JOB IN
291		REF S#WISER	INPUT: BATCH QUEUE TABLE IN RBBAT

292	*,*			SERIAL NUMBER USED TO MAKE SERIAL
293	*,*			NUMBER CHECKS - MOVED TO T SERIAL
294		REF	S#H:LNK	I/O: BATCH QUEUE TABLE IN RBBAT
295	*,*			SERIAL NUMBER LINK USED TO RUN SERIA
296	*,*			NUMBER CHAINS AND TO SORT SERIAL
297	*,*			NUMBERS FOR RERUN AND RECOVERY
298		REF	S#BITYP	INPUT: BATCH QUEUE TABLE IN RBBAT
299	*,*			RESOURCE INDEX USED TO MAKE SERIAL
300	*,*			NUMBER CHECKS - CONVERTED TO DEVICE
301	*,*			TYPE AND MOVED TO R SERIAL
302	*			
303		REF	GI:IFRE	I/O: GETI TABLE - HEAD OF FREE ENTRI
304	*,*			USED TO DECIDE WHETHER TO TRY TO SCH
305	*,*			ULE - GETI ENTRY TO FILL IN IS
306	*,*			REMOVED FROM HERE
307		REF	GI:SDA	OUTPUT: GETI TABLE - STARTING DISC AD
308	*,*			(AND JEBIT AND RRBIT IF SET) ARE
309	*,*			MOVED HERE FROM BW:SDA
310		REF	GI:RES	OUTPUT: GETI TABLE - RESOURCES FROM B
311	*,*			ARE MOVED HERE
312		REF	GI:ASPN	OUTPUT: GETI TABLE - DEVICE OWNERSHIP
313	*,*			BUILT BY MBS IN BI:ASPN ARE MOVED HE
314		REF	GIB:PRT	OUTPUT: GETI TABLE - PARTITION NUMBER
315	*,*			SELECTED BY MBS IS MOVED HERE
316		REF	GIH:TIM	OUTPUT: GETI TABLE - TIME FROM BH:TIM
317	*,*			IS MOVED HERE
318		REF	GIB:RID	OUTPUT: GETI TABLE - RBID FROM BB:RID
319	*,*			MOVED HERE
320		REF	GIB:IUN	OUTPUT: GETI TABLE - USER NUMBER TAKE
321	*,*			FROM SNULL IS PUT HERE - IT WILL BE
322	*,*			THE USER NUMBER OF THE NEW JOB
323		REF	GIB:SLN	OUTPUT: GETI TABLE - THE SHARED SERIA
324	*,*			NUMBER HEAD OF THE CHAIN FORMED WHEN
325	*,*			MBS MOVES SHARED SERIAL NUMBERS TO C
326	*,*			IS PUT HERE
327		REF	GIB:XLN	OUTPUT: GETI TABLE - THE EXCLUSIVE SE
328	*,*			NUMBER HEAD INTO L SERIAL IS PUT HERE

Line	Code	Label	Description
329	**		
330	REF	GIB:PRI	WHEN MBS MOVES THE SERIAL NUMBERS TO OUTPUT: GETI TABLE - THE REQUESTED RUN JOB PRIORITY IS MOVED HERE FROM B4W: <sup>10</sup>
331	**		
332	*		
333	REF	LPART	NUMBER OF PARTITIONS IN THE SYSTEM
334	REF	MBSOP#	LPART*SV:RSIZ USED TO THREAD THROUGH PLB:MAX AND PLB:MIN
335	**		
336	REF	PLILK	I/O: PARTITION CHANGE LOCK - IF SET CONTROL IS MESSING WITH THE PARTITION AND MBS CANT RUN - IF NOT MBS SETS I TO KEEP CONTROL OUT OF THE WAY
337	**		
338	**		
339	**		
340	REF	PLICHG	I/O: PARTITION CHANGE FLAGS SET BY CONTROL TO SHOW WHAT PARTITIONS HAVE BEEN CHANGED - MBS STORES USER NUMBER FROM SNULL INTO TOP BYTE
341	**		
342	**		
343	**		
344	REF	PLB:USR	OUTPUT: PARTITION TABLE - USER NUMBER FOR NEW JOB IS STORED HERE
345	**		
346	REF	PLH:TOL	OUTPUT: PARTITION TABLE - TOTAL JOBS RUN IN THIS PARTITION COUNTED UP BY MBS WHEN A JOB IS SCHEDULED IN THIS PARTITION
347	**		
348	**		
349	**		
350	REF	PLH:CUR	OUTPUT: PARTITION TABLE - TOTAL JOBS RUN IN THIS PARTITION SINCE CHANGE @ RECOVERY COUNTED UP BY MBS WHEN A JOB IS RUN IN THIS PARTITION
351	**		
352	**		
353	**		
354	REF	PLH:TL	INPUT: PARTITION TABLE - TIME LOWER LIMIT USED BY MBS TO SCHEDULE JOBS
355	**		
356	REF	PLH:TU	INPUT: PARTITION TABLE - TIME UPPER LIMIT USED BY MBS TO SCHEDULE JOBS
357	**		
358	REF	PLH:SID	OUTPUT: PARTITION TABLE - SYSID STORED IN BY MBS FROM BH:SID
359	**		
360	REF	PLD:ACT	OUTPUT: PARTITION TABLE - ACCOUNT STORED IN BY MBS FROM BD:ACCT
361	**		
362	REF	PLH:FLG	I/O: PARTITION TABLE - FLAG BITS: BIT0 = HOLD IN CORE
363	**		
364	**		
365	**		

366	*,*		BIT12 = CR JOBS LEGAL
367	*,*		BIT13 = JOBENT JOBS LEGAL
368	*,*		BIT14 = RB JOBS LEGAL
369	*,*		BIT15 = LOCKED
370	*,*		THE BITS ARE USED TO DECIDE LEGAL
371	*,*		PARTITIONS AND MAKE THE ACCOUNT CHECK
372	*,*	REF	INPUT: PARTITION TABLE = RESOURCE MA
373	*,*		USED BY MBS TO SCHEDULE JOBS
374	*,*	REF	INPUT: PARTITION TABLE = RESOURCE MI
375	*,*		USED BY MBS TO SCHEDULE JOBS
376	*,*	REF	OUTPUT: IN CORE SERIAL NUMBER TABLE
377	*,*		INTO WHICH MBS MOVES S#WISER
378	*,*	REF	I/O: IN CORE SERIAL NUMBER TABLE
379	*,*		WHICH MBS USES TO LINK UP IN CORE
380	*,*		SERIAL NUMBERS TO GIB:XLN AND GIB:SL
381	*,*	REF	OUTPUT: IN CORE SERIAL NUMBER TABLE
382	*,*		INTO WHICH MBS STORES THE DEVICE TYP
383	*,*		DERIVED FROM S#BITYP AND SB:RTY
384	*		
385	*,*	REF	I/O: RBBAT'S STATIC DATA AREA
386	*,*		8 WORD AREA USED TO BUILD RESOURCE
387	*,*		TABLE OF AVAILABLE TO BATCH
388	*,*	REF	I/O: RBBAT'S STATIC DATA AREA
389	*,*		4 WORD AREA USED TO BUILD DUMMY B4W:
390	*,*		WHICH IS INCREMENTED FOR SHARED
391	*,*		SERIAL NUMBERS NOT MOUNTED
392	*,*	REF	I/O: RBBAT'S STATIC DATA AREA
393	*,*		3 WORD AREA USED TO BUILD DEVICE OWN
394	*,*		ERSHIP BITS
395	*,*	REF	I/O: RBBAT'S STATIC DATA AREA
396	*,*		CELL USED TO HOLD WORD ADDRESS OF B4
397	*,*		FOR THE CANDIDATE JOB
398	*,*	REF	INPUT: RBBAT'S STATIC DATA AREA
399	*,*		CELL THAT WHEN SET SAYS RBBATR IS RU
400	*,*		AND MBS SHOULDN'T
401	*,*	REF	I/O: RBBAT'S STATIC DATA AREA
402	*,*		MBS SETS THIS CELL NON ZERO TO TURN

H01 11:11 SEP 08, 1975

403 \*,\*  
404 \*,\*  
405 \*  
406 \*  
407 \*,\*  
408 \*,\*  
409 \*,\*  
410 \*,\*  
411 \*,\*  
412 \*  
413 \*  
414 \*  
415 \*

REF ADD1  
REF FIND  
REF FIND1  
REF SVIV

12  
OFF THE SPECIAL SAVED ENVIRONMENT IT  
SAVES

CALLED: TO REPORT IBM STATE AND BEG  
SWAP OF NEW BATCH JOBS  
CALLED: TO START RUNNING BATCH QUEUE  
PRIORITY CHAINS  
CALLED: TO CONTINUE RUNNING BATCH QU  
PRIORITY CHAINS  
CALLED: TO SAVE A SPECIAL RBBAT ENVI

			PAGE	
			CSECT	
416				1
417	05 00000			
418	0000001P	S	BACCT EQU	BT31T00+31
419	0000002U	S	B8RDR EQU	BT31T00+32
420	0000001E	S	AVRD EQU	BT31T00+30
421	0000000U		R0 EQU	0
422	00000001		R1 EQU	1
423	00000002		R2 EQU	2
424	00000003		R3 EQU	3
425	00000004		R4 EQU	4
426	00000005		R5 EQU	5
427	00000006		R6 EQU	6
428	00000007		R7 EQU	7
429	00000008		R8 EQU	8
430	00000009		R9 EQU	9
431	0000000A		R10 EQU	10
432	0000000B		R11 EQU	11
433	0000000C		R12 EQU	12
434	0000000D		R13 EQU	13
435	0000000E		R14 EQU	14
436	0000000F		R15 EQU	15
437	00000001		PLK EQU	1
438	0000800U		PH8LD EQU	X'8000'
439	0000200U		PRUN EQU	X'2000'
440	0000400U		PACCT EQU	X'4000'
441	00000002		RBLGL EQU	2
442	00000008		LCLGL EQU	8
443	00000004		JELGL EQU	4

444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480

FFFFF S

PAGE

```

*
*
*F* NAME: MBSOP
*F*
*F* PURPOSE: SET BIT IN BH:PART FOR EACH PARTITION THE GIVEN JOB
*F* CAN RUN IN BASED ON RESOURCES REQUESTED AND JOB ORIGN
*F*
*
*F* NAME: REPART
*F*
*F* PURPOSE: REEVALUATE LEGAL PARTITION BITS IN BH:PART FOR ALL JOB
*F* IN THE BATCH QUEUE FOR THOSE PARTITIONS WHOSE ATTRIBUT
*F* HAVE BEEN CHANGED BY CONTROL
*F*
*
* MBSOP,REPART
* SET OR RESET BITS IN BH:PART
* BAL,11 MBSOP
* 2 = BH:PART INDEX
* ALL BITS ARE SET IF JOB FITS IN PARTITION
* AND RESET IF JOB DOESNIT FIT
* BAL,11 REPART
* 15= BH:PART BITS TO BE REEXAMINED
* THOSE BITS IN 10 ARE SET OR RESET APPROPRIATELY
* IN BH:PART FOR ALL JOBS IN THE QUEUE
*
*
* MBSOP# EQU SV:RSIZ*LPART (DEFINED BY SYSGEN)
*
* MBSOP#1 EQU MBSOP#+LPART=1
*
* MBSOP#1 CAN BE ADDED TO THE PLBIMIN/MAX INDEX
* OF RESOURCE N IN PARTITION 0 (WHICH OF COURSE ISNT
* REALLY THERE) TO OBTAIN THE INDEX OF RESOURCE N=1
    
```

```

481
482
483      00000000  S
484
485
486
487
488
489
490      05 00000
491      05 00000 022000030 A
492      05 00001 055000000 X
493      05 00002 32C000000 X
494      05 00003 22FFFFFFF A
495      05 00004 68000000B
496      05 00005
497      05 00005 221000000 N
498      05 00006 6A0000000 X
499      05 00007 E8000000B A
500      05 00008 32C000002 A
501      05 00009 25C000002 A
502      05 0000A 20C000000 N
503      05 0000B
504      05 0000B 22E000000 A
505      05 0000C 227000002 A
506      05 0000D 723400000 X
507      05 0000E 693000014
508      05 0000F 323000000 X
509      05 00010 313400000 X
510      05 00011 694000013
511      05 00012 207000004 A
512      05 00013 207000002 A
513      05 00014
514      05 00014 223000000 N
515      05 00015 517600000 X
516      05 00016 684000018
517      05 00017 49E600000 X
    
```

```

* IN PARTITION LPART
*
MBSOP#2 EQU MBSOP#1+1+SV:RSIZ
*
*MBSOP#2 IS THE PLBIMIN/MAX INDEX OF THE HIGHEST
*NUMBERED (SV:RSIZ+1) RESOURCE IN THE HIGHEST
*NUMBERED (LPART) PARTITION
*
*
MBSOP EQU *
LCI 3
PSM,5 TMPSTK
LW,12 BADRS
LI,15 *1
B REPART1
REPART EQU *
LI,1 FIPRI
BAL,0 FIND
B *11
LW,12 2
SLS,12 2
AI,12 B+WIRES
REPART1 EQU *
LI,R14 0
LI,R7 RBLGL
LB,R3 BB:RID,R2
BNEZ CKORG
LW,R3 JEBIT
CW,R3 BW:SDA,R2
BANZ *+2
AI,R7 LCLGL=JELGL
AI,R7 JELGL=RBLGL
CKORG EQU *
LI,R3 LPART
CH,R7 PLH:FLG,R3
BAZ *+2
BR,R14 BT31T00,R3
    
```



NO1

11:11 SEP 08, '75

555 05 00038 E940000A A  
 556 05 00039 6800003B  
 557 05 0003A  
 558 05 0003A 4BE60000 X  
 559 05 0003B  
 560 05 0003B 205FFFFFF N  
 561 05 0003C 64300037  
 562 05 0003D 205FFFFFF N  
 563 05 0003E 646E0001 A  
 564 05 0003F 680E0000 A  
 565 05 00040  
 566 05 00040 F2DC000C A  
 567 05 00041 21D00080 A  
 568 05 00042 E8400008 A  
 569 05 00043 72DC0000 X  
 570 05 00044 49D00000 X  
 571 05 00045 F5DC000C A  
 572 05 00046 4BD00000 X  
 573 05 00047 E8000008 A

CKRSB

CKRSG

SETRS

BANZ \*10  
 B CKRSG  
 EQU \*  
 AND,14 NB31T80,3  
 EQU \*  
 AI,5 \*(SVIRSIZ+1)  
 BDR,3 CKRS2  
 AI,5 MBSOP#1  
 BDR,6 1,7  
 B 0,7  
 EQU \*  
 LB,13 \*12,6  
 CI,13 X'80'  
 BAZ \*8  
 LB,13 SBIRBDF,6  
 BR,13 X80  
 STB,13 \*12,6  
 AND,13 M7  
 B \*8



596  
 597  
 598  
 599  
 600  
 601  
 602  
 603  
 604  
 605  
 606  
 607  
 608  
 609  
 610  
 611 05 00051 52660000 X  
 612 05 00052 556E0000 X  
 613 05 00053 69300058  
 614 05 00054 55780000 X  
 615 05 00055 21700000 A  
 616 05 00056 E920000B A  
 617 05 00057 55680000 X  
 618 05 00058 21700000 A  
 619 05 00059 E930000B A  
 620 05 0005A 55680000 X  
 621 05 0005B E800000B A

PAGE

```

*
*
*F* NAME: BATUNG
*F*
*F* PURPOSE: UNCHAINS A JOB ENTRY FROM A PRIORITY CHAIN FOR MBS
*F*
*
*
* REMOVE JOB ENTRY FROM PRIORITY QUEUE
* BAL,11 BATUNG
* 3=ENTRY INDEX, 4=PRIORITY
* 7=INDEX OF PREVIOUS ENTRY IN QUEUE.
* IF UNQUEUEING HEAD OF CHAIN, 7 IS EITHER 0 OR
* HA(BH:HPRI)-HA(BH:LINK)+PRIORITY (WHICH IS NEGATIVE)
BATUNG LH,6 BH:LINK,3 UNQUEUE
STH,6 BH:LINK,7 4=PRIORITY
BNEZ *+5
STH,7 BH:TPRI,4
CI,7 0
BG *11
STH,6 BH:TPRI,4
CI,7 0
BNEZ *11
STH,6 BH:HPRI,4
B *11
    
```





HO1 11111 SEP 08, 175

\*\*\*\* ILLEGAL AF  
\*\*\*\* ILLEGAL CF

679	05	0007A	22400000	N
680	05	0007B	72280000	X
681	05	0007C	683001CC	
682	05	0007D	72640000	X
683	05	0007E	75680000	X
684	05	0007F	69300081	
685	05	00080	75680000	X
686	05	00081	751C0000	X
687	05	00082	35200000	X
688	05	00083	00000000	A

LI,4	SNULL
LB,2	SB:HQ,4
BEZ	MBSX
LB,6	UB:FL,2
STB,6	SB:HQ,4
BNEZ	*+2
STB,6	SB:TQ,4
STB,1	UB:BL,6
STW,2	PL:CHG
ENABLE	

NULL STATE INDEX FOR FREE USER SLOTS  
TRY TO GET ONE

NEXT IN QUEUE  
SET NEW HEAD

NO HEAD => NO TAIL  
BL OF HEAD IS 0

\*\*\*\* UNDEF CBM  
\*\*\*\* ILLEGAL AF  
\*\*\*\* ILLEGAL CF



H01 11:11 SEP 08, '75

726	05	00099	32F0000E	A
727	05	0009A	683001CC	
728	05	0009B	22400001	N
729	05	0009C	22700000	A
730	05	0009D	55700000	X
731	05	0009E	680000A0	

LW,15	14
BEZ	MBSX
LI,4	FIPRI+1
LI,7	0
STH,7	BHILINK
B	MBSJ





H01

11:11 SEP 08, '75

792	05	000B6	6940n1C3	
793	05	000B7	31D0n020	N
794	05	000B8	6840n0BB	
795	05	000B9	51E40000	X
796	05	000BA	6920n09F	
797	05	000BB	6420n0B1	
798	05	000BC	31D0n020	N
799	05	000BD	6840n0C6	
800	05	000BE	2210n000	N
801	05	000BF	6A00n000	X
802	05	000C0	6#00n0C6	
803	05	000C1	11A40000	X
804	05	000C2	6930n000	X
805	05	000C3	51E40000	X
806	05	000C4	6920n09F	
807	05	000C5	6800n000	X

MBSA1

BANZ	MBSF
CW,13	B0RDR
BAZ	MBSA1
CH,14	PLH;SID,2
BG	MBSJO
BDR,2	MBSA
CW,13	B0RDR
BAZ	MBSR
LI,1	FIPRI
BAL,0	FIND
B	MBSR
CD,10	BD;ACCT,2
BNE	FIND1
CH,14	BH;SID,2
BG	MBSJO
B	FIND1

NEW ACCT DEPENDENT  
 NEW ORDER FLAG  
 NO  
 ORDER  
 WRONG WAY

NO ORDER, CHECK RESOURCES

808  
 809  
 810  
 811  
 812  
 813  
 814  
 815  
 816  
 817  
 818  
 819  
 820  
 821  
 822  
 823  
 824  
 825 05 000C6  
 826 05 000C6 22100003 A  
 827 05 000C7 22200000 A  
 828 05 000C8 75200000 X  
 829 05 000C9 3523FFFF N  
 830 05 000CA 641000C9  
 831 05 000CB 22600000 N  
 832 05 000CC 32C00000 X  
 833 05 000CD  
 834 05 000CD 6A800040  
 835 05 000CE 51DC0000 X  
 836 05 000CF 692001B4  
 837 05 000D0 75DC0000 X  
 838 05 000D1 646000CD  
 839  
 840  
 841  
 842  
 843  
 844

PAGE  
 \* DETERMINE IF ADEQUATE RESOURCES ARE AVAILABLE  
 \* SAME REGISTERS AS ACCT/ORDER CHECK EXCEPT:  
 \* 2 \* 0 ON ENTRY; BECOMES JISPIN  
 \* 8 \* RESOURCE REQUIREMENTS (4 BYTES)  
 \* 9 \* HEAD OF FOUND SERIAL NUMBERS CHAIN  
 \* (SHARED SERIAL NUMBERS FOUND IN AVRTBL  
 \* ARE UNLINKED FROM THE BB:SLNK CHAIN)

\*F\* NAME: MBSR (MBSRO)

\*F\* PURPOSE: CHECKS WHETHER SUFFICIENT RESOURCES ARE AVAILABLE TO RU  
 \*F\* THE CANDIDATE JOB

\*  
 \* MBSR EQU \*  
 LI,1 3 CLEAR BI:ASPN  
 LI,2 0  
 STB,2 BBI:CRE5  
 STW,2 BI:ASPN-1,1  
 BDR,1 \*-1  
 LI,6 SVIR5IZ  
 LW,12 BADR5  
 \* MBSRO EQU \*  
 BAL,8 SETRS  
 CH,13 BSL:CRE5,6  
 BG MBSNR  
 STB,13 BBI:CRE5,6  
 BDR,6 MBSRO

\*F\* NAME: MBSCS\*

\*F\* PURPOSE: PROCESSES THE SERIAL NUMBER LISTS FROM THE CANDIDATE  
 \*F\* JOB'S MOUNT OPTION. DETERMINES WHETHER SERIAL NUMBER



1111 SEP 08, 175

882	05	000D9	22C00000	N		LI,12	BHISLNK
883	05	000DA	6AB00191			BAL,11	RUNSER
884	05	000DB	22C00000	N		LI,12	BH:XLNK
885	05	000DC	6AB00191			BAL,11	RUNSER
886	05	000DD	22D00000	N		LI,13	BH:SLNK
887	05	000DE	22B000D6			LI,11	MBSCS#0
888	05	000DF	68000191			B	RUNSER
889	05	000EU			MBSCM	EQU	*
890	05	000E0	22100000	N		LI,1	HA(BH:XLNK)-HA(S#H:LNK)
891	05	000E1	30100003	A		AW,1	3
892	05	000E2	6AB0019D			BAL,11	RUNMNT
893	05	000E3	680000E6			B	MBSCM1
894	05	000E4	6800009F			B	MBSJO
895	05	000E5	680000E2			B	*=3
896	05	000E6			MBSCM1	EQU	*
897	05	000E6	52960000	X		LH,9	BHISLNK,3
898	05	000E7	22100000	N		LI,1	HA(BH:SLNK)-HA(S#H:LNK)
899	05	000E8	30100003	A		AW,1	3
900	05	000E9	6AB0019D		MBSCM2	BAL,11	RUNMNT
901	05	000EA	68000108			B	MBSSP3
902	05	000EB	680000EE			B	MBSIS
903	05	000EC	73140000	X		MTB,1	B8ICRES,2
904	05	000ED	680000E9			B	MBSCM2
905	05	000EE			MBSIS	EQU	*
906	05	000EE	12CC0000	X		LD,12	AVRTBL,6
907	05	000EF	48D0001E	N		AND,13	AVRD
908	05	000F0	693000F2			BNEZ	*=2
909	05	000F1	64B001A1			BDR,11	RUNMNT0
910	05	000F2	528C0000	X		LH,8	AVRID,6
911	05	000F3	6930009F			BNEZ	MBSJO
912	05	000F4	73100000	X		MTB,1	B8ICRES
913	05	000F5	2520027B	A		SCS,2	=5
914	05	000F6	72600002	A		LB,6	2
915	05	000F7	2560007D	A		SLS,6	=3
916	05	000F8	32840000	X		LW,8	BI:ASPN,2
917	05	000F9	498C0001	N		OR,8	BT31T00+1,6
918	05	000FA	35840000	X		STW,8	BI:ASPN,2

HO1 11:11 SEP 08, '75

919 05 000FB 00000000 A  
 \*\*\*\* UNDEF COM  
 \*\*\*\* ILLEGAL AF  
 \*\*\*\* ILLEGAL CF  
 920 05 000FC 52620000 X  
 921 05 000FD 556A0000 X  
 922 05 000FE 52D60000 X  
 923 05 000FF 55D20000 X  
 924 05 00100 55160000 X  
 925 05 00101 00000000 A

DISABLE  
 LH,6 S#H:LNK,1  
 STH,6 S#H:LNK,5  
 LH,13 BH:SLNK,3  
 STH,13 S#H:LNK,1  
 STH,1 BH:SLNK,3  
 ENABLE

\*\*\*\* UNDEF COM  
 \*\*\*\* ILLEGAL AF  
 \*\*\*\* ILLEGAL CF  
 926 05 00102 31100009 A  
 927 05 00103 69300105  
 928 05 00104 32900006 A  
 929 05 00105 31600000 A  
 930 05 00106 683000E9  
 931 05 00107 64B0019E

CW,1 9  
 BNE \*+2  
 LW,9 6  
 CW,6 13  
 BE MBSCM2  
 BDR,11 RUNMNT+1

932  
 933  
 934  
 935  
 936  
 937  
 938  
 939  
 940  
 941  
 942

\*  
 \*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*  
 \*

NAME: MBSSP3  
 PURPOSE: CHECKS WHETHER RESOURCES INCREMENTED BECAUSE REQUESTED SHARED SERIAL NUMBERS WERE NOT MOUNTED ARE AVAILABLE. CHECKS WHETHER THERE IS ROOM IN THE IN-CORE SERIAL NUM TABLES TO MOVE DOWN THE CUI'S SERIAL NUMBERS.

943 05 00108  
 944 05 00108 22100000 N  
 945 05 00109 72820000 X  
 946 05 0010A 51820000 X  
 947 05 0010B 69200184  
 948 05 0010C 64100109  
 949 05 0010D 22C00000 A

MBSSP3 EQU \$  
 LI,1 SV:RS1Z  
 LB,8 B&ICRES,1  
 CH,8 BSLICRES,1  
 BG MBSNR  
 BDR,1 \*-3  
 LI,12 0

H01

11:11 SEP 08, '75

32

950	05	0010E	32600009	A
951	05	0010F	68300111	
952	05	00110	6AB00182	
953	05	00111	22600000	N
954	05	00112	6AB00180	
955	05	00113	68000116	
956	05	00114	726C0000	X
957	05	00115	683001B4	
958	05	00116	64C00114	

LW,6	9
BEZ	*+2
BAL,11	CNTSER+2
LI,6	HA(BH:XLNK)-HA(S#H:LNK)
BAL,11	CNTSER
B	*+3
LB,6	LSERIAL,6
BEZ	MBSNR
BDR,12	*=2

959  
 960  
 961  
 962  
 963  
 964  
 965  
 966  
 967  
 968  
 969  
 970

PAGE

\*  
 \*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*  
 \*  
 \*

NAME: MBSRJ

PURPOSE: THE CANDIDATE JOB HAS BEEN SELECTED TO BE RUN. A GETI  
 TABLE SLOT IS TAKEN AND GIB:PRI, GIB:RID, AND GIM:TIM  
 ARE FILLED IN FROM B4WIRES(BYTE0), BB:RID, AND BH:TIME.

RESOURCES ARE AVAILABLE - RUN THE JOB

971 05 00117  
 972 05 00117 00000000 A  
 \*\*\*\* UNDEF COM  
 \*\*\*\* ILLEGAL AF  
 \*\*\*\* ILLEGAL CF

MBSRJ

EQU \$  
 DISABLF

973 05 00118 32100000 X  
 974 05 00119 25100078 A  
 975 05 0011A 46100000 X  
 976 05 0011B 45100000 X  
 977 05 0011C 00000000 A

LW,1 G1IFRE GET G1 TABLE SLOT  
 SLS,1 =8  
 XW,1 G1IFRE  
 AND,1 M8  
 ENABLE

\*\*\*\* UNDEF COM  
 \*\*\*\* ILLEGAL AF  
 \*\*\*\* ILLEGAL CF

978 05 0011D F2600000 X  
 979 05 0011E 75620000 X  
 980 05 0011F 72660000 X  
 981 05 00120 75620000 X  
 982 05 00121 52660000 X  
 983 05 00122 55620000 X  
 984 05 00123 72200000 X  
 985 05 00124 68300136

LB,6 \*BADRS GET PRI0  
 STB,R6 GIB:PRI,1 AND SAVE IN CORE  
 LB,6 BB:RID,3  
 STB,6 GIB:RID,1  
 LH,6 BH:TIME,3  
 STH,6 GIM:TIM,1  
 LB,2 BB:GRES  
 BEZ MBSSCH1

986  
 987  
 988  
 989

\*  
 \*  
 \*F\*  
 \*F\*

NAME: MBSSCH

HC1 11:11 SEP 08, '75

990  
 991  
 992  
 993  
 994  
 995 05 00125 225FFFFF N  
 996 05 00126 2260001F N  
 997 05 00127 2560027B A  
 998 05 00128 72200006 A  
 999 05 00129 45600000 X  
 1000 05 0012A 2520007D A  
 1001 05 0012B 20200001 A  
 1002 05 0012C 32ADFFFF N  
 1003 05 0012C 32ADFFFF N  
 1004 05 0012D 31A40000 X  
 1005 05 0012E 68400130  
 1006 05 0012F 531A0000 X  
 1007 05 00130 205FFFFF A  
 1008 05 00131 21500000 N  
 1009 05 00132 69100136  
 1010 05 00133 6420012D  
 1011 05 00134 22200020 A  
 1012 05 00135 6460012C  
 1014  
 1015  
 1016  
 1017  
 1018  
 1019  
 1020  
 1021  
 1022  
 1023 05 00136 22200004 A  
 1024 05 00137 32A00000 X  
 1025 05 00138 20AFFFFF A

\*F\*  
 \*F\*  
 \*F\*  
 \*  
 \*

PURPOSE: INCREMENTS AVRNOU FOR EACH PACK WHOSE DEVICE OWNERSHIP BIT HAS BEEN SET IN BIIASPN BY MBSCSN

LI,5 AVRTBLNE=1 INCREMENT SHARED SER#S  
 LI,6 AVRTBLNE+BATAPE=1+X'20'  
 SCS,6 =5  
 LB,2 6  
 AND,6 M8  
 SLS,2 =3  
 AI,2 1  
 MBSSCH EQU \*  
 LW,10 BIIASPN=1,6  
 MBSSCHO EQU \*  
 CW,10 BT31T00,2  
 BAZ \*+2  
 MTH,1 AVRNOU,5  
 AI,5 =1  
 CI,5 AVRTBLSIZ  
 BL MBSSCH1  
 BDR,2 MBSSCHO  
 LI,2 32  
 BDR,6 MBSSCH

\*  
 \*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*F\*  
 \*  
 \*

NAME: MBSSCH1  
 PURPOSE: MOVES RESOURCES FROM B4W,RES TO GI,RES AND DEVICE OWNERSHIP BITS FROM BIIASPN TO GI,ASPN

MBSSCH1 EQU \*  
 LI,2 4  
 LW,10 BADRS  
 AI,10 =1

1027	05	00139	32B00001	A
1028	05	0013A	25B00002	A
1029	05	0013B	20BFFFFFF	N
1030	05	0013C	B2C4000A	A
1031	05	0013D	B5C4000B	A
1032	05	0013E	6420013C	
1033	05	0013F	20B00000	N
1034	05	00140	35B00001	A
1035	05	00141	22200003	A
1036	05	00142	32C5FFFF	N
1037	05	00143	B5C4000B	A
1038				
1039				
1040				
1041				
1042				
1043				
1044				
1045				
1046				
1047				
1048				
1049				
1050				
1051				
1052				
1053		05 00144		
1054	05	00144	22600000	N
1055	05	00145	72CC0000	X
1056	05	00146	50CC0000	X
1057	05	00147	55CC0000	X
1058	05	00148	64600145	
1059	05	00149	22600000	N
1060	05	0014A	52560000	X
1061	05	0014B	4550000F	A
1062	05	0014C	315C0000	X
1063	05	0014D	6940014F	

LW,11	1
SLS,11	2
AI,11	GI:RES=1
LW,12	*10,2
STW,12	*11,2
BDR,2	*=2
AI,11	GI:ASPN=GI:RES
SW,11	1
LI,2	3
LW,12	BI:ASPN=1,2
STW,12	*11,2

\*  
\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*F\*  
\*  
\*  
\*

NAME: MBSGR

PURPOSE: ADDS THE NEW JOB'S RESOURCE REQUIREMENTS TO SH:RBCU. SELECTS A PARTITION TO RUN HIM IN, TAKING THE HIGHEST NUMBERED ONE HE FITS IN. MOVES THE REST OF THE JOB INFORMATION FROM THE BATCH QUEUE TABLES TO THE GETI TABLES. CHAINS THE JOB ENTRY TO THE RUNNING CHAIN. INCREMENTS S:IBUIS, DECREMENTS S:BFIS, SETS THE BATCH BIT IN UH:FLG2, AND CALLS ADD1 TO START THE NEW USER.

\* SWIPE RESOURCES

MBSGR	EGU	*
	LI,6	SVIRSIZ
	LB,12	B8ICRES,6
	AM,12	SH:RBCU,6
	STH,12	SH:RBCU,6
	BDR,6	*=3
	LI,6	LPART
	LH,5	BH:PART,3
	AND,5	15
	CW,5	BT31T80,6
	BANZ	*+2

FIND PARTITION TO RUN

MO1 11111 SEP 08, '75

1064 05 0014E 6460014C  
 1065 05 0014F 75620000 X  
 1066 05 00150 531C0000 X  
 1067 05 00151 531C0000 X  
 1068 05 00152 12A60000 X  
 1069 05 00153 15AC0000 X  
 1070 05 00154 32200000 X  
 1071 05 00155 55EC0000 X  
 1072 05 00156 32D60000 X  
 1073 05 00157 45D0001F N  
 1074 05 00158 52D0000D A  
 1075 05 00159 50DC0000 X  
 1076 05 0015A 20D02000 A  
 1077 05 0015B 55DC0000 X  
 1078 05 0015C 752C0000 X  
 1079 05 0015D 22800000 A  
 1080 05 0015E 75820000 X  
 1081 05 0015F 75820000 X  
 1082 05 00160 72F00000 X  
 1083 05 00161 32A00009 A  
 1084 05 00162 68300165  
 1085 05 00163 75F20000 X  
 1086 05 00164 6AB00185  
 1087 05 00165 52A60000 X  
 1088 05 00166 68300169  
 1089 05 00167 75F20000 X  
 1090 05 00168 6AB00185  
 1091 05 00169 75F00000 X  
 1092 05 0016A 32660000 X  
 1093 05 0016B 31600000 X  
 1094 05 0016C 6940016E  
 1095 05 0016D 49600000 X  
 1096 05 0016E 32200000 X  
 1097 05 0016F 00000000 A

BDR,6 \$=2  
 STB,6 GIB:PRY,1  
 MTH,1 PLH:CUR,6  
 MTH,1 PLH:TOL,6  
 LD,10 BDI:ACCT,3  
 STD,10 PLD:ACT,6  
 LW,2 PL:CHG  
 STH,14 PLH:SID,6  
 LW,13 BWISDA,3  
 AND,13 BACCT  
 LH,13 13  
 AM,13 PLH:FLG,6  
 AI,13 PRUN  
 STH,13 PLH:FLG,6  
 STB,2 PLB:USR,6  
 LI,8 0  
 STB,8 GIB:ISLN,1  
 STB,8 GIB:IXLN,1  
 LB,15 LSERIAL  
 LW,10 9  
 BEZ \$+3  
 STB,15 GIB:ISLN,1  
 BAL,11 MOV:SERV  
 LH,10 BH:IXLNK,3  
 BEZ \$+3  
 STB,15 GIB:IXLN,1  
 BAL,11 MOV:SERV  
 STB,15 LSERIAL  
 LW,6 BWISDA,3  
 CW,R6 RRB:IT  
 BANZ \$+2  
 AND,6 YFF  
 LW,2 PL:CHG  
 DISABLF

GET USER #  
 PUT USER ID  
 MOVE ONLY ACCT FLAG TO PLH:FLG  
 ADD IN PHOLD  
 AND PRUN  
 MOVE SERVS TO CORE

\*\*\*\* UNDEF C8M  
 \*\*\*\* ILLEGAL AF  
 \*\*\*\* ILLEGAL CF

MO1 11:11 SEP 08, '75

1098 05 00170 46660000 X  
 1099 05 00171 35620000 X  
 1100 05 00172 75220000 X  
 1101 05 00173 75260000 X  
 1102 05 00174 6AB00051  
 1103 05 00175 22400000 N  
 1104 05 00176 6AB00048  
 1105 05 00177 6A30005C  
 1106 05 00178 22F00000 N  
 \*\*\*\*\* UNDEF SYM  
 1107 05 00179 55F40000 X  
 1108 05 0017A 33100000 X  
 1109 05 0017B 33F00000 X  
 1110 05 0017C 32400002 A  
 1111 05 0017D 32F00000 X  
 1112 05 0017E 6AB00000 X  
 1113 05 0017F 6800006F  
 1114  
 1115  
 1116  
 1117  
 1118  
 1119  
 1120  
 1121  
 1122  
 1123 05 00180  
 1124 05 00180 30600003 A  
 1125 05 00181 20C00001 A  
 1126 05 00182 526C0000 X  
 1127 05 00183 69300181  
 1128 05 00184 E800000B A  
 1129  
 1130  
 1131  
 1132  
 1133

XW,6 BW:SDA,3  
 STW,6 GI:SDA,1  
 STB,2 GIB:UN,1  
 STB,2 BB:PI,3  
 BAL,11 BATUNG  
 LI,4 RUNPRI  
 BAL,11 BATQT  
 BAL,3 QHQ  
 LI,15 BAT  
 STH,15 UH:FLG,2  
 MTW,1 S:BUIS  
 MTW,-1 S:BFIS  
 LW,4 2  
 LW,15 SL:BPRI0  
 BAL,11 ADD1  
 B MBS1

UNQUEUE SYMB ENTRY  
 ADD TO SCHED QUEUE  
 PUT SLOT BACK IN QUEUE  
 SET BAT BIT  
 ADD1 NEEDS UN IN 4  
 DEFAULT BATCH PRIORITY

\*  
 \*  
 \*F\* NAME: CNTSER  
 \*F\*  
 \*F\* PURPOSE: COUNTS THE NUMBER OF SERIAL NUMBERS IN A JOB'S SHARED  
 \*F\* OR EXCLUSIVE CHAIN  
 \*F\*  
 \*  
 \*  
 CNTSER EQU \*  
 AW,6 3  
 AI,12 1  
 LH,6 S#H:LNK,6  
 BNEZ \*2  
 B \*11  
 \*  
 \*  
 \*F\* NAME: MOVSEERS  
 \*F\*  
 \*F\* PURPOSE: MOVES SERIAL NUMBERS FROM THE CHAINS IN RBBAT (S#W:SER

```

1134
1135
1136
1137
1138      05 00185
1139 05 00185 3250000F A
1140 05 00186 3260000A A
1141 05 00187 329C0000 X
1142 05 00188 359A0000 X
1143 05 00189 722C0000 X
1144 05 0018A 72940000 X
1145 05 0018B 759A0000 X
1146 05 0018C 72FA0000 X
1147 05 0018D 52AC0000 X
1148 05 0018E 69300185
1149 05 0018F 75AA0000 X
1150 05 00190 E800000B A
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161      05 00191
1162 05 00191 D254000D A
1163 05 00192 E830000B A
1164      05 00193
1165 05 00193 D266000C A
1166 05 00194 E830000B A
1167 05 00195 328A0000 X
1168      05 00196
1169 05 00196 318C0000 X
1170 05 00197 6830009F
    
```

\*F\* TO THE IN-CORE CHAINS (TSERIAL)

\*F\*

\*

\*

```

MOVSEERS EQU $
          LW,5 15
          LW,6 10
          LW,9 S#WISER,6
          STW,9 TSERIAL,5
          LB,2 S#B:TYP,6
          LB,9 SBIRTY,2
          STB,9 RSERIAL,5
          LB,15 LSERIAL,5
          LH,10 S#H:LNK,6
          BNEZ MOVSEERS
          STB,10 LSERIAL,5
          B *11
    
```

\*

\*

\*F\* NAME: RUNSER

\*F\*

\*F\* PURPOSE: COMPARES EACH SERIAL NUMBER IN A CANDIDATE JOB'S CHAIN  
 \*F\* WITH ALL SERIAL NUMBERS IN THE SHARED OR EXCLUSIVE CHAIN  
 \*F\* OF ALL RUNNING JOBS AND REJECTS THE CJ IF A MATCH IS FOUND

\*F\*

\*

\*

```

RUNSER EQU $
          LW,5 *13,2
          BEZ *11
RUNSER1 EQU $
          LW,6 *12,3
          BEZ *11
          LW,8 S#WISER,5
RUNSER2 EQU $
          CW,8 S#WISER,6
          BE MBSJO
    
```

H01 11:11 SEP 08, '75

1171 05 00198  
 1172 05 00198 526C0000 X  
 1173 05 00199 69300196  
 1174 05 0019A 525A0000 X  
 1175 05 0019B 69300193  
 1176 05 0019C E800000B A  
 1177  
 1178  
 1179  
 1180  
 1181  
 1182  
 1183  
 1184  
 1185  
 1186 05 0019D  
 1187 05 0019D 32500001 A  
 1188 05 0019E 521A0000 X  
 1189 05 0019F E830000B A  
 1190 05 001A0 22600000 N  
 1191 05 001A1  
 1192 05 001A1 32C20000 X  
 1193 05 001A2 72220000 X  
 1194 05 001A3 680001A6  
 1195 05 001A4  
 1196 05 001A4 19CC0000 X  
 1197 05 001A5 683001AC  
 1198 05 001A6  
 1199 05 001A6 206FFFFFF A  
 1200 05 001A7 21600000 N  
 1201 05 001A8 681001A4  
 1202 05 001A9 20B00001 A  
 1203 05 001AA 20B00001 A  
 1204 05 001AB E800000B A  
 1205 05 001AC  
 1206 05 001AC 72840000 X  
 1207 05 001AD 32200006 A

RUNSER3 EQU \*  
 LH,6 S#H:LNK,6  
 BNEZ RUNSER2  
 LH,5 S#H:LNK,5  
 BNEZ RUNSER1  
 B \*11

\*  
 \*  
 \*F\* NAME: RUNMNT  
 \*F\*  
 \*F\* PURPOSE: GETS THE NEXT SERIAL NUMBER FROM THE CJ'S CHAIN AND  
 \*F\* INFORMS THE CALLER WHETHER OR NOT IT IS MOUNTED AND IN  
 \*F\*  
 \*

RUNMNT EQU \*  
 LW,5 1  
 LH,1 S#H:LNK,5  
 BEZ \*11  
 LI,6 AVRTBLNE  
 RUNMNT0 EQU \*  
 LW,12 S#W:SER,1  
 LB,2 S#B:TYP,1  
 B RUNMNT15  
 RUNMNT1 EQU \*  
 CLM,12 AVRTBL,6  
 BCR,3 RUNMNT2  
 RUNMNT15 EQU \*  
 AI,6 \*1  
 CI,6 AVRTBLSIZ  
 BGE RUNMNT1  
 AI,11 1  
 RUNMNT18 AI,11 1  
 B \*11  
 RUNMNT2 EQU \*  
 LB,8 SBIRTY,2  
 LW,2 6

H01 11:11 SEP 08, '75

1208	05	001AE	20200000	N
1209	05	001AF	71840000	X
1210	05	001B0	6930009F	
1211	05	001B1	52CC0000	X
1212	05	001B2	693001AA	
1213	05	001B3	680001A1	

AI,2	BATAPE
CB,8	DCT4,2
BNE	MBSJO
LH,12	AVRNBU,6
BNEZ	RUNMNT18
B	RUNMNT0



HO1 11:11 SEP 08, '75

\*\*\*\* ILLEGAL CF

1246 05 001C2 644000A0

BDR,4 MBSJ



H01 11111 SEP 08, 175  
 1281 05 001CE 35200000 X  
 1282 05 001CF 46200000 X  
 1283 05 001D0 683001D2  
 1284 05 001D1 6A30005C  
 1285 05 001D2 00000000 A MBSX1  
 \*\*\*\* UNDEF CBM  
 \*\*\*\* ILLEGAL AF  
 \*\*\*\* ILLEGAL CF  
 1286 05 001D3 22B00001 A  
 1287 05 001D4 35B00000 X  
 1288 05 001D5 02200070 A  
 1289 05 001D6 0A500000 X  
 1290 05 001D7 E800000B A  
 1291

STW,2 PLILK  
 XW,2 PL;CHG  
 BEZ MBSX1  
 BAL,3 QHO  
 ENABLE  
  
 LI,11 1  
 STW,11 SVZ  
 LCI 7  
 PLM,5 TMPSTK  
 B \*11  
 END

UNLOCK TABLES  
 RELEASE USER # IF ANY  
 PUT BACK IN QUEUE

RETURN

CONTROL SECTION SUMMARY: 01 00000 PT 0 02 00000 PT 0 03 00000 PT 1 04 00000 PT 0  
 05 001D8 PT 1

\* SYMBOL VALUES

AVRD/0000001E S	BACCT/0000001F S	BATQT/05 00048	BATUNG/05 00051
BORDR/00000020 S	CKBRG/05 00014	CKRS/05 00033	CKRSB/05 0003A
CKRSG/05 0003B	CKRS1/05 00036	CKRS2/05 00037	CNTSER/05 00180
JELGL/00000004	LCLGL/00000008	MASK/FUNC	MBSA/05 000B1
MBSA1/05 000BB	MBSCKF/05 001C7	MBSCH/05 000E0	MBSCH1/05 000E6
MBSCH2/05 000E9	MBSCS#/05 000D2	MBSCS#0/05 000D6	MBSF/05 001C3
MBSF1/05 001C5	MBSGR/05 00144	MBSI/05 001B4	MBSIS/05 000EE
MBSJ/05 000A0	MBSJO/05 0009F	MBSJ2/05 000A6	MBSNR/05 001B4
MBSOP#1/FFFFFF S	MBSOP#2/00000000 S	MBSOP1/05 0001F	MBSOP2/05 00022
MBSOP3/05 00024	MBSOP4/05 0002B	MBSOP2/05 00099	MBSR/05 000C6
MBSRJ/05 00117	MBSRO/05 000CD	MBSOP2/05 00099	MBSR/05 000C6
MBSRCH1/05 00136	MBSRP3/05 00108	MBSOP2/05 00099	MBSRCH0/05 0012D
MBS1/05 0006F	MBS3/05 00084	MBSOP2/05 00099	MBSX1/05 001D2
PACCT/00004000	PHOLD/00008000	MBSOP2/05 00099	NMASK/FUNC
QHO/05 0005C	RBLGL/00000002	MBSOP2/05 00099	PRUN/00002000
RUNMNT/05 0019D	RUNMNT0/05 001A1	MBSOP2/05 00099	REPART/05 0000B
RUNMNT18/05 001AA	RUNMNT2/05 001AC	MBSOP2/05 00099	REPART1/05 0000B
RUNSER2/05 00196	RUNSER3/05 00198	MBSOP2/05 00099	RUNMNT15/05 001A6
R10/0000000A	R11/0000000B	MBSOP2/05 00099	RUNSER1/05 00193
R14/0000000E	R15/0000000F	MBSOP2/05 00099	R1/00000001
R4/00000004	R5/00000005	MBSOP2/05 00099	R13/0000000D
R8/00000008	R9/00000009	MBSOP2/05 00099	R3/00000003
		MBSOP2/05 00099	R7/00000007
		MBSOP2/05 00099	UFLAGS/00000001

\* EXTERNAL DEFINITIONS

MBS/05 00064 MBSOP/05 00000

\* PRIMARY REFERENCES

ADD1	AVRID	AVRNGU	AVRTBL	AVRTBLNE	AVRTBLSIZ	BADRS
BATAPE	BB:PI	BB:RID	BD:ACCT	BH:HPRI	BH:LINK	BH:PART
BH:SID	BH:SLNK	BH:TIME	BH:TPRI	BH:XLNK	BH:ASPN	BH:CREP
BT31T80	BW:SDA	B4W:RES	B8:CREP	DCT4	FIND	FIND1
FIPRI	GI:ASPN	GI:FRE	GI:RES	GI:SDA	GIB:PRI	GIB:PKT
GIB:RID	GIB:SLN	GIB:UN	GIB:XLN	GIH:TIM	JEBIT	LPART
LSERIAL	MBSOP#	M7	M8	NB31T80	PL:CHG	PL:ILK
PLB:MAX	PLB:MIN	PLB:USR	PLD:ACT	PLH:CUR	PLH:FLG	PLH:SID
PLH:TL	PLH:TBL	PLH:TU	RCVRFLG	RRBIT	RSERIAL	RUNPRI
SIBFIS	S:BUAIS	SIBUIS	S:MBSF	S#BITYP	S#H:LNK	S#W:SER
SB:HQ	SB:RBDP	SB:RTY	SBITQ	SH:RCBU	SH:RBSUM	SH:RGU

HO1	11:11	SEP 08, '75						
	SHIROCU	SHIRTOT	SLIBPRI0	SLIPI	SNULL	SVIRSIZ	SVIV <sup>46</sup>	
	SVZ	TMPSTK	TSERIAL	UB:BL	UB:FL	UH:FLG	X80	
	YFF							
* NO	SECONDARY REFERENCES							
* UNDEFINED SYMBOLS	BAT							
* ERROR SEVERITY LEVEL 3								
* ERROR LINES								
	678	688	919	925	972	977	1097	
	1106	1241	1245	1280	1285			