

SDS PROGRAM LIBRARY  
PROGRAM DESCRIPTION

Page 1 of 11

Model No. 705280-11A00

TITLE: SDS REAL-TIME BATCH MONITOR (RBM)

AUTHOR: SDS

ACCEPTED:

COMPUTER  
CONFIGURATION: SIGMA 7 BCM configuration plus RAD.  
Minimum 16K memory.PURPOSE: To provide a RAD-oriented version of SIGMA 5/7  
Basic Control Monitor.SUBROUTINES  
USED: none

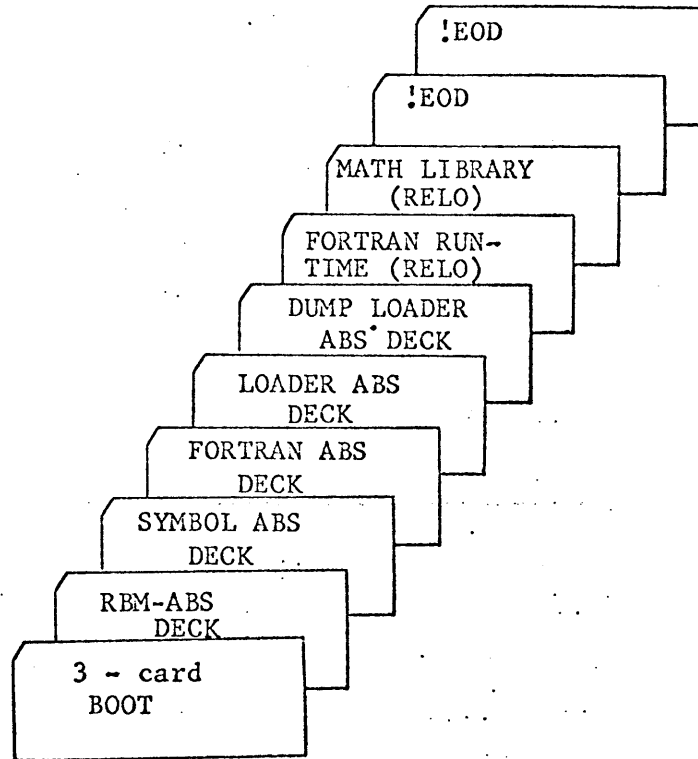
ACCURACY: N/A

SOURCE LANGUAGE: SYMBOL

DESCRIPTION: RBM is a modification of Version B of the Sigma 7 Basic Control Monitor, Catalog Number 704144. It provides the same services as BCM. It creates and operates in a RAD-oriented environment. It assumes a minimum of 16K of core memory during execution. The various parts of the system are available as relocatable decks. Each installation obtains the various decks and follows a SYSGEN procedure to build an installation-dependent system.

The final system will consist of card images and will be bootable from any sequential access device. It will have the following form:

## DESCRIPTION (cont'd)



The LOADER and DUMP LOADER are modified in RBM. The SYMBOL and FORTRAN processors are the standard versions of these processors as distributed by SDS. The installation will make ABS decks of these processors with the bias necessary for that particular installation (nominally X'1000'). The RBM-ABS deck is made by combining several relocatable decks through the use of the DUMP LOADER. These decks are (in order):

- |     |        |                               |
|-----|--------|-------------------------------|
| 1.  | 705281 | RBM MONITOR                   |
| 2.  | 705282 | RBM I/O INTERFACE             |
| 3.  | 705283 | RBM DISC HANDLER              |
| 4.  | 704852 | FORTRAN BCD CONVERSION        |
| 5.  | 704854 | DUMMY FORTRAN BCD CONVERSION  |
| 6.  | 704851 | 7-TRACK MAGNETIC TAPE HANDLER |
| 7.  | 704369 | MAG TAPE HANDLER              |
| 8.  | 704370 | LINE PRINTER HANDLER          |
| 9.  | 704371 | CARD READER HANDLER           |
| 10. | 704372 | CARD PUNCH HANDLER            |
| 11. | 704373 | TYPEWRITER HANDLER            |
| 12. | 704374 | PAPER TAPE HANDLER            |

DESCRIPTION (cont'd)

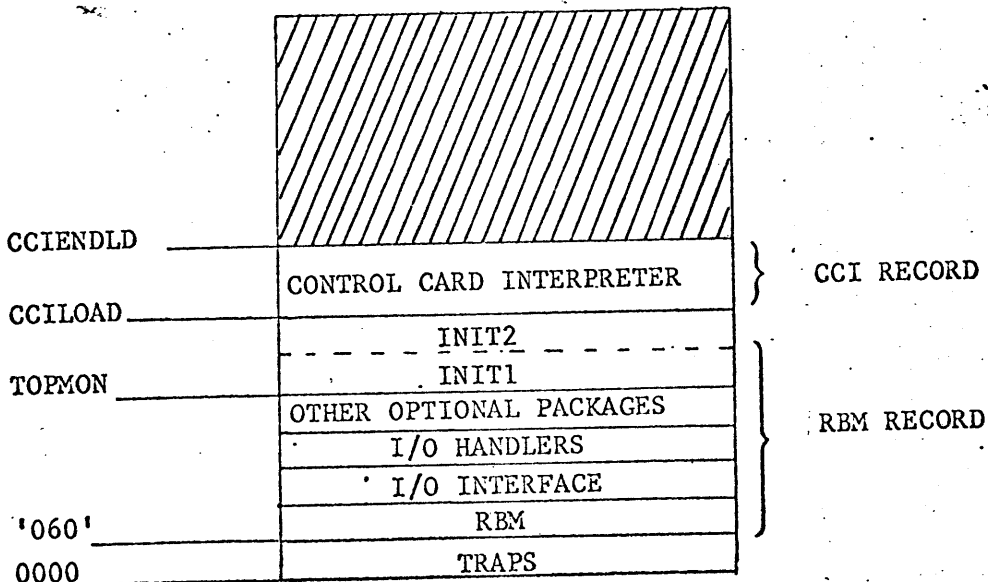
Decks (cont'd)

- 13. 704363 FLOATING POINT INSTRUCTION SIMULATOR
- 14. 704364 DECIMAL INSTRUCTION SIMULATOR
- 15. 704365 BYTE INSTRUCTION SIMULATOR
- 16. 704366 CONVERT INSTRUCTION SIMULATOR
- 17. 705284 RBM INITIALIZATION
- 18. 705285 RBM CONTROL CARD INTERPRETER

Programs with catalog numbers of '704xxx' are part of the BCM Software Package distributed under the catalog cover number 704127. Programs with catalog numbers of '705xxx' are part of the RBM Software Package distributed under the catalog cover number 705280.

To make a RBM ABS deck, all the relocatable decks with numbers 705xxx are required. The 704xxx decks are added as needed by the hardware configuration in the order shown. The total set of relocatable decks is then input to the DUMPING LOADER to form a single ABS deck.

A boot operation of the RBM system brings in the RBM ABS deck as shown below:



DESCRIPTION (cont'd)

The areas labelled INIT1 and INIT2 are logically separate programs, but are physically in the same relocatable deck. The initialization routine INIT2 gains control when the boot is complete. It performs the following operations:

1. When INIT2 gains control, a message is typed out to the typewriter asking for hardware channel and device changes. If none are required, type NO and new line. If changes are required, type YES and new line and a type-out of the type AAA = CNN will appear

where

AAA =	}	LPA	line printer
		CRA	card reader
		CPA	card punch
		PTR	paper tape reader
		PTP	paper tape punch
		9TA	9 track tape
		7TA	7 track tape
DCA	disc		

and

CNN is

C = IOP number 0-7  
 NN = DEVICE number

The AAA = CNN will be typed out in the order of the equipments listed for AAA. They will be typed out one at a time and then wait for a reply from the typewriter. If no change is desired for that device, just type new line. If a change is required, type CNN followed by a new line.

Example:

The disc is on MIOP 1 and all other equipment is on MIOP 0.

Answer YES N/L to question of change

```
LPA = 002
new line
CRA = 003
new line
.
.
.
7TA = 0E0
new line
DCA = 0F0
1F0 new line.
```

The disc will be permanently assigned to 1F0 in the DCT1 table and no further changes are required.

## DESCRIPTION (cont'd)

2. Following the request for hardware changes, a request to change the op label assignments will be made. Again, if none are required, type NO and new line. If changes are required, type YES and new line. A typeout of the form XX = will appear where:

XX = { C - Control command input  
OC - Operator's console  
LO - Listing output  
LL - Listing log  
DO - Diagnostic output  
PO - Punch output  
BO - Binary output  
LI - Library input  
SI - Source input  
BI - Binary input

After each op label is typed out, initialization will wait for a reply from the typewriter. If no change is required for that op label, type new line. If a change is required, input the change in the form of YYN and new line where:

YY = { TY - Typewriter  
PR - Paper tape reader  
PP - Paper tape punch  
CR - Card reader  
CP - Card Punch  
LP - Line printer  
DC - Disc  
7T - 7 track magnetic tape  
9T - 9 track magnetic tape

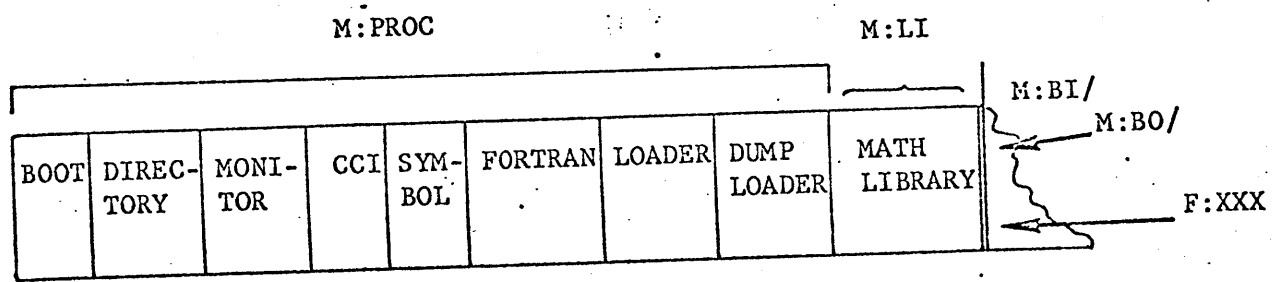
All changes are stored in OPLBT2 and OPLBT3 in the monitor and become the standard op label assignments for the system.

DESCRIPTION (cont'd)

- 3. Requests, via the typewriter, each of the processor decks, and in turn, reads them into available core above CCIENDLD and then writes each processor on the RAD as a single record.
- 4. Requests, via the typewriter, the library, reads the library, packs the cards three to a sector and writes them on the disc. (8 cards per sector for high speed RAD)
- 5. Stores various parameters into RBM and CCI concerning the information on disc.
- 6. Writes the area labelled "RBM" onto the disc.
- 7. Writes the area labelled "CCI record" onto the disc.
- 8. Writes a bootstrap record onto sector zero of the disc.
- 9. Transfers to INIT1.

INIT1 then begins the normal initialization sequence of RBM by outputting the typewriter message "RBM". Control may now be returned to this point at any time by booting from the RAD.

The format of the RAD at this time is:



## DESCRIPTION (cont'd)

With BOOT occupying sector zero and the other entries of M:PROC occupying as many sectors as necessary.

Logically, the operation of RBM and standard BCM are identical. Actually, CCI, the control card interpreter, is non-resident and is brought in from the RAD between execution phases of a job. In addition, if a processor call control card, e.g. !FORTRAN LS,LO,BO, is encountered by CCI for one of the processors on the RAD and if that particular processor was not introduced to the system immediately preceding e.g. !ABS FORTRAN, then the processor is loaded from the RAD file, M:PROC. (NOTE: To obtain the dump loader, use !DLOAD. Also, DLOAD uses PO for its ABS binary output.)

Following the loading of a processor, memory layout is as shown:

512 $\left( \frac{\text{TOPMON}+511}{512} \right)$	PROCESSOR
TOPMON	
	OTHER OPTIONAL ROUTINES
	I/O HANDLERS
	I/O INTERFACE
X'060'	RBM
	TRAPS

The operational labels BI, BO and LI are changed to have a nominal assignment of RAD hardware type; thus, the M:BI, M:BO and M:LI DCBs will be nominally assigned to RAD. The M:BI and M:BO nominal assignments may be changed with ASSIGN cards.

## DESCRIPTION (cont'd)

In addition, a pack/unpack routine is included which packs and unpacks BI/LI/BO in 6 card (180 word) blocks if these files are assigned to RAD (8 card (240 word) blocks for high speed RAD).

IF BI IS ASSIGNED TO RAD AT THE TIME A LOAD CARD IS ENCOUNTERED, RBM ASSUMES A COMPILE-AND-GO SITUATION IN WHICH COMPILATION WAS THE LAST OPERATION WHICH REFERENCED BO. EOD indicators are written on BO at its current position and it is then rewound to serve as the BI device for the loading operation.

A job to compile FORTRAN decks, assemble SYMBOL decks and then execute the two together would look as shown:

```

!JOB
!FORTRAN LS,BO
  < FORTRAN SOURCE >
  < FORTRAN SOURCE >
  .
  .
  .
!SYMBOL LO,BO
  < SYMBOL SOURCE >
!SYMBOL LO, BO
  < SYMBOL SOURCE >
!LOAD
!RUN
!DATA

```

The disc management in RBM is minimal. The system treats the disc as a temporary storage device with the exception of the M:PROC and M:LI files. The system maintains a table (DFTBL) which has ten entries (files). Each entry has five half-word items: first sector, last sector, current sector, sectors per record, and bytes per record. First and last sector are the bounds of the file on disc. Current sector is the last referenced position within the file and sectors per record and bytes per record are self-explanatory when one assumes fixed-length records.

The first two entries in DFTBL define M:PROC and M:LI. The items are set by INIT2 when the processors and library are put onto disc and are neverchanged. The third entry is for BI/BO and is set by CCI



DESCRIPTION (cont'd)

to begin at the end of M:LI and end at the physical end of the RAD whenever a JOB card is encountered. The remaining entries are for user files and are set by CCI as a result of processing ASSIGN cards as shown:

!ASSIGN DCB, (DEVICE, DCAFO,RECORDS,WORDS)

where WORDS is the number of words per record in the file and RECORDS is the number of records desired. CCI makes entries in DFTBL reserving sufficient contiguous area to contain the file with the first user entry beginning at the same place as BI/BO, the second beginning where the first ends, etc., etc.

The disc handler recognizes the following requests: READ, WRITE, REWIND, and POSITION RECORD. The function parameter tables for REWIND and POSITION RECORD are not changed from those of BCM. READ and WRITE parameter tables are as shown:

CODE 8	0 7	DCB	17
P1-P5 <sup>5</sup>		0 22	1 4 W 0
		ABN	17
		ERROR	17
		BUFFER	17
		BYTES	16
		KEY	17

## DESCRIPTION (cont'd)

where CODE is X'10' for READ, X'11' for WRITE.

DCB is the word address of the DCB.

P1 is set to indicate that ABN is present.

P2 is set to indicate that ERROR is present.

P3 is set to indicate that BUFFER is present.

P4 is set to indicate that BYTES is present.

P5 is set to indicate that KEY is present.

ABN, ERROR, BUFFER, and BYTES have the same meaning as defined in BCM. KEY is the word address of a cell containing the record to be referenced for this operation. The handler uses this number and sectors per record and first sector to calculate the disc address of the record. If KEY is absent (P5=0) or equal to zero, the handler advances the current sector pointer one record and performs the operation on that disc address.

RBM requires the addition of a sixth word to all existing DCBs. Monitor DCBs and DCBs built as a result of ASSIGN cards are taken care of by the system. If the user has DCBs assembled into his binary deck, he must reassemble and add a word of zero. This sixth word is used as shown:

DFN	16	RECORD #	16
-----	----	----------	----

where DFN is the disc file number or ordinal of the DFTBL entry for this file and RECORD # is the record number from the last request.

TITLE: RBM Source Mag Tape

ACCEPTED:

TABLE OF CONTENTS

<u>File No.:</u>	<u>Identification:</u>
1	705281 - RBM Monitor EOF
2	705282 - RBM I/O Interface EOF
3	705283 - RBM Disc Handler EOF
4	705284 - RBM Initialization EOF
5	705285 - RBM Control Card Interpreter EOF
6	705286 - RBM Relocating Loader EOF
7	705287 - RBM Dumping Loader EOF
8	705288 - RBM RADIO EOF
9	705289 - RBM RUNIO EOF EOF

# SDS PROGRAM LIBRARY PROGRAM DESCRIPTION

## A P P E N D I X

### I. SUPPORT OF HIGH SPEED (7212) OR SLOW SPEED (7204) RAD

The RAD allocation scheme in RBM is built around the total number of sectors available on the RAD ("raw sector count"), without regard to the number of sectors available on a track or band. In order to use this scheme and be compatible with both the 7212 and the 7204 it was necessary to introduce a set of constants and an algorithm to the monitor.

The constants used are as follows:

<u>NAME</u>	<u>7204 value</u>	<u>7212 value</u>
ENDBIBUF	BEGBIBUF + 179	BEGBIBUF + 239
BILIBYT	1440	1920
BILIBYT1	1440	1920
SECTOR	90	240
SECTOR1	90	256
SECTRK	16	82
SHIFTYPE	4	7
BILISEC	2	1
BILICDS	6	8
MAXDF	8192	5248
NOBYSP	360	1024
NOBYSL	360	960

where:

- ENDBIBUF      sets the BI/BO/LI pack/unpack buffer to 180 or 240 words
- BILIBYT      sets the number of bytes in the buffer to 1440 or 1920
- BILIBYT1     same as BILIBYT
- SECTOR       sets the sector size to 90 words or 240 words for BI/BO/LI pack and unpack
- SECTOR1      sets the sector size to 90 words or 256 words for CCI processing of user disc assignments
- SECTRK       sets the number of sectors / track or band to 16 and 82 (used in algorithm)

A P P E N D I X (cont'd)

where:

SHIFTYPE set to 4 or 7 (used by algorithm)  
 BILISEC sets the number of sectors / record for BI/BO/LI to 2 or 1  
 BILICDS sets the number of cards/record for BI/BO/LI to 6 or 8  
 MAXDF sets the number of sectors / disc to 8192 or 5248  
 NOBYSP sets the number of bytes / sector for processors to 360 or 1024  
 NOBYSL sets the number of bytes / sector for library to 360 or 960

The INITIALIZATION routine determines the type of RAD and then sets the above constants for the appropriate RAD.

Prior to any actual RAD read/writes, the "raw sector count" is sent to the algorithm (DISCART) which converts this value to the actual disc address. This disc address is then used in the SEEK command to position the disc for reads or writes. This value is kept in a half-word table (CURBDSEC) in the monitor disc file table.

II. CHANGING OF OPERATIONAL LABEL ASSIGNMENTS

This optional feature, provided during ground-zero initialization of Sigma 5/7 RBM, is especially convenient for the user who is operating with a minimum configuration (RAD, TYPEWRITER, PAPER TAPE READER/PUNCH). Upon completion of the initialization phase, RBM is ready to process jobs without the need of executing a series of console interrupts and !SYST KEYIN'S.

After the option to change hardware address is exercised, the message 'Change Op Label Assignments? Type YES or NO Followed By New Line' is printed on the typewriter. A reply of 'NO' or 'N/L' will cause initialization to continue leaving the operational labels set to standard assignments.

A 'YES' reply will cause each operational label followed by an equal sign to be output, one at a time, to the typewriter. The operator must reply with 'XXN N/L' for a change or N/L for no change, where XX = the first two letters of the appropriate device mnemonic:

TY - typewriter	CP - card punch
PR - paper tape reader	LP - line printer
PP - paper tape punch	DC - disc (RAD)
CR - card reader	9T - nine track mag tape
	7T - seven track mag tape

A P P E N D I X (cont'd)II. CHANGING OF OPERATIONAL LABEL ASSIGNMENTS (cont'd)

As an illustration, assume a minimum configuration consisting of a typewriter, paper tape reader/punch and a RAD:

CHANGE OPLABEL ASSIGNMENTS?

TYPE YES OR NO FOLLOWED BY NEW LINE

YES

C = TYN	Typewriter
OC = N/L	Standard assignment - typewriter
LO = TYN	Typewriter
LL = TYN	Typewriter
DO = TYN	Typewriter
PO = PPN	Paper tape punch
BO = N/L	Standard assignment - RAD
LI = N/L	Standard assignment - RAD
SI = PRN	Paper tape reader
BI = N/L	Standard assignment - RAD

Input for each operational label is compared to a table in initialization when found it stores the device pointer in the monitor tables OPLBT2 and OPLBT3. The assignments are permanent until changed with a !SYST KEYIN.

The initial assignments of op labels are as follows:

C	- Card reader	(CRN)
OC	- Typewriter	(TYN)
LO	- Line printer	(LPN)
LL	- Line printer	(LPN)
DO	- Line printer	(LPN)
PO	- Card punch	(CPN)
BO	- Disc	(DCN)
LI	- Disc	(DCN)
SI	- Card reader	(CRN)
BI	- Disc	(DCN)

APPENDIX (cont'd)III. RBM REAL-TIME CAPABILITY

RBM provides for both resident and non-resident foreground capability as well as the ability to checkpoint the background area of core to the RAD.

A. Assembly Parameters

Three monitor parameters are associated with foreground operations in RBM. They are as follows:

1. FORGRND EQU N

N = 0 if no foreground operations are to be included in the system

N = 1 if foreground operations are to be included in the system

2. NFORPAGE EQU N

N = number of pages of memory (at top of core) to be reserved for non-resident foreground

3. DCHKP EQU N

N = 0 if the checkpoint feature is not to be included in the system

N = 1 if checkpoint features are to be included in the system

B. Resident Foreground

Resident foreground programs must be included (as ROM decks) with the group of ROM decks that make up the RBM monitor. These decks, monitor and foreground, are then dumped in ABS form (with the dumping loader) to produce the monitor ABS deck. Four names (FOR0, FOR1, FOR2, and FOR3) are included in the foreground key-in table to provide key-in linkage for up to four imbedded foreground programs that have declared these four names (FOR0 thru FOR3) as DEFs. If non-resident foreground is also to be used in the system, the name FOR0 should not be used in the resident foreground programs as it will be needed to provide key-in linkage to the non-resident routine.

A P P E N D I X (cont'd)III. RBM REAL-TIME CAPABILITY (cont'd)C. Non-Resident Foreground

Non-resident foreground programs are introduced into the system from the background job stream as ROM decks and loaded with the loader. The "REAL" option is specified on the loader control card. This causes the loader to load the program into the reserved foreground area (pages reserved at top of core) and place the name and starting address of the routine into the FOR0 position of the key-in table. The program can then be initiated with a key-in.

D. Checkpoint

When the checkpoint feature is included in the system, a foreground program (resident or non-resident) may request the monitor to checkpoint the current background program and release all of background core for use by the foreground program. The monitor will save the background area on the disc (at the high end) and restore it when requested by the foreground program.

A CALL,9 6 is used to checkpoint the background and a  
CALL,9 7 is used to restore the background.

E. Foreground Initiation

The unsolicited key-in is used to initiate a real-time foreground program (resident or non-resident) and is of the form:

!F name,m

where

name - is the directory name of the pertinent foreground program

m - specifies the action to be taken. If m = I, the specified program is initiated. If m ≠ I, m is stored in the high order byte of the address word in the directory entry for the program, and system operation is not affected.



A P P E N D I X (cont'd)III. RBM REAL-TIME CAPABILITY (cont'd)F. Programmer Requirements

It is the responsibility of each foreground job to save the machine environment of an interrupted job, by storing the contents of location X'4F' (JIT table pointer), the current general registers, and the PSD. Each foreground job must provide eight contiguous memory locations for the foreground JIT (Job Information Table). The address of the first word of this table should be placed into location X'4F'. The monitor assumes responsibility for maintaining this table but all entries should initially be zero.

A foreground job may use monitor I/O services using monitor DCBs or establish and maintain its own DCBs. Also, the foreground routine may perform all of its I/O independent of monitor services. The foreground job can use any of the monitor functions except the monitor return functions.

Before returning control, the foreground job must restore the previous contents of X'4F' as well as the general registers and PSD.

Foreground programs are entered initially through the key-in linkage with a BAL,SR4 routine entry address.

Returns should be made with a

B \*SR4

to insure proper entrance and exit.

G. Programming Considerations

Foreground programs may cause the monitor to be re-entered in several ways. Each of these require an increase in STACKSIZ. For each foreground program that uses monitor CALs (concurrently), the increase must be 33 words. For each key-in-initiated foreground program that may be initiated while another such foreground program is in process, or for each device that may cause an I/O interrupt while an I/O interrupt is being processed for another device, the increase must be 47 words.

### III. RBM REAL-TIME CAPABILITY (cont'd)

#### G. Programming Considerations (cont'd)

The following restrictions should be considered in real-time operations:

The priority of a real-time program cannot be higher than the priority of the I/O interrupt if the program is going to use monitor I/O services.

#### H. Standard RBM Release

The standard release of RBM will be assembled with the foreground and checkpoint options included and NFORPAGE set for 1 page of non-resident foreground.

### IV. SYSTEM ACCOUNTING

RBM assumes that CLOCK 4 is available and running at 500 CPS for the purpose of time accounting. It maintains a separate cell in memory as a cumulative counter for each processor recognized by the system plus a total job time counter. The values of these counters are printed out at the end of each job on M:L0 providing the user with a written record of the total time spent in each processor and the total job times. The difference between the summation and the total job is due to control card processing.

A sample print-out is shown below:

```
***      4.544 (MIN) IN  SYMBOL
***      2.214 (MIN) IN  LOADER
***      0.012 (MIN) IN  XECUTION
***      7.372 (MIN) IN  THIS JOB
```