

Program Logic

Version 8.1

IBM System/360 Time Sharing System

Operator Task and Bulk I/O

This publication describes the internal logic of the Operator Task, Batch Monitor, BULKIO Preprocessor, BULKIO Task, and certain command and macro instruction processing routines associated with the above tasks, which operate under the IBM System/360 Time Sharing System. The intended audience is the system programmer concerned with the maintenance of these system tasks.

The book consists of an introduction to main operator control and nonconversational operations, and a description of the Operator Task and Batch Monitor, the BULKIO Preprocessor, the BULKIO Task, the command routines, macro instruction handling, and time shared utility programs. Logic flowcharts are provided for all the routines discussed. Appendixes detail control blocks used, external modules called, a module directory, the active remote table, and the Operator Task and BULKIO modules using RCR macro instructions.

Prerequisite Publications

The reader must be familiar with the information contained in:

IBM System/360 Time Sharing System:
Concepts and Facilities, GC28-2003
System Messages, GC28-2037
System Control Blocks Program Logic Manual,
GY28-2011

PREFACE

This publication describes the internal logic of the Operator Task, Batch Monitor, BULKIO Preprocessor, and BULKIO task programs, as well as certain command and macro instruction processors. The programs may be initiated by the System Operator or by the Batch Monitor as background tasks.

The manual discusses:

Batch Monitor and Operator Task (TASKID 001).

BULKIO Preprocessor.

BULKIO Task (TASKID 002).

Command Processors.

Macro Instruction Processors.

Time Shared Utility Processors.

A brief introduction to nonconversational processing appears before the discussions

of the individual programs. The manual provides flowcharts for the above programs and appendixes that detail data areas and modules called by these programs.

This manual requires familiarity with overall TSS/360 techniques and terminology, as contained in:

IBM System/360 Time Sharing System: Concepts and Facilities, GC28-2003.
System Messages, GC28-2037.
System Control Blocks Program Logic Manual, GY28-2011.

In addition, familiarity with the following publication is highly recommended:

IBM System/360 Time Sharing System: Command System Program Logic Manual, GY28-2013.

Fifth Edition (September 1971)

This is a major revision of, and makes obsolete, GY28-2047-3, and TNL GN28-3133.

This edition is current with Version 8, Modification 1 of IBM System/360 Time Sharing System, and remains in effect for all subsequent versions or modifications unless otherwise noted. Significant changes or additions to this publication will be provided in new editions or Technical Newsletters. Before using this publication, refer to the latest edition of IBM System/360 Time Sharing System: Addendum, GC28-2043, which may contain information pertinent to the topics covered in this edition. The Addendum also lists the editions of all TSS/360 publications that are applicable and current.

This publication was prepared for production using an IBM computer to update the text and to control the page and line format. Page impressions for photo-offset printing were obtained from an IBM 1403 Printer using a special print chain.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form is provided at the back of this publication for reader's comments. If the form has been removed, comments may be addressed to IBM Corporation, Department 643, Neighborhood Road, Kingston, N.Y. 12401

SECTION 1: INTRODUCTION TO MAIN OPERATOR CONTROL AND NONCONVERSATIONAL OPERATIONS	1
Main Operator Control	1
Nonconversational Operations	1
Task Initiation	1
Task Termination	1
Express Batch	2
SECTION 2: OPERATOR TASK AND BATCH MONITOR	3
Operator Task General Operation	3
Main Operator's Housekeeping Routine -- MOHR (CZACB)	3
Main Operator Communications Program -- MOCP (CZACA)	6
Batch Monitor General Operations	7
Batch Monitor (CZABA)	7
SECTION 3: BULKIO PREPROCESSOR	13
General Operations	13
BULKIO Preprocessor (CZABD)	13
SECTION 4: BULKIO TASK	15
BULKIO Initialization Routine (CZAWS)	16
BULKIO Control Routine (CZAWT)	17
BULKIO Master Service (CZAWW)	17
BULKIO Attend Alert Routine (CZAWU)	18
BULKIO Input Start (CZAWX)	19
BULKIO Input Service (CZAWZ)	21
FORTRAN Data Set Conversion (CZAWC)	23
BULKIO Input Closeout Routine (CZAWV)	23
BULKIO OUTPUT Service (CZAWY)	24
BULKIO Message Handler Routine (CZAWM)	26
BULKIO ACK Data Set Handler (CZAWN)	26
BULKIO ABEND Recovery (CZAWA)	27
MSAM Input Recovery Procedure (WABE)	27
VAM Input Recovery Procedure (WABV)	28
MSAM Output Recovery Procedure (WAMO)	28
VAM Output Recovery Procedure (WAVO)	29
Initialization Recovery Procedure (WAINIT)	29
SECTION 5: COMMAND ROUTINES	30
ASNBD Command Routine (CZABK)	30
FORCE Command Routine (CZACG)	32
HOLD/DROP Command Routine (CZCMO)	32
LABEL Command Routine (CZABX)	34
MSG/BCST Command Routine (CZACF)	34
PRINT Routine (CZABG)	35
PUNCH Routine (CZABH)	37
RT Routine (CZABF)	38
SHUTDOWN Command Routine (CZACN)	39
WT Routine (CZABI)	40
SECTION 6: MACRO INSTRUCTION HANDLING	43
XWTO Routine (CZABQ)	43
SECTION 7: TIME SHARED UTILITY PROGRAMS	45
PATCLEAR	45
PATFIX	45
DMPRST	45
TIME SHARED UTILITY MODULE	45
PATCLEAR Processor (CZUFO)	45
DMPRST Processor (CZUFA)	46
Time Shared Utility Module (CZUBC)	48
PATFIX Routine (CZUPF)	49

SECTION 8: FLOWCHARTS	51
APPENDIX A: CONTROL BLOCKS REFERENCED TO CALLING MODULES232
APPENDIX B: EXTERNAL CALLED MODULES REFERENCED TO PLMS233
APPENDIX C: MODULE DIRECTORY234
APPENDIX D: ACTIVE REMOTE TABLE238
APPENDIX E: OPERATOR TASK AND BULKIO MODULES USING RCR MACROS239
INDEX240

ILLUSTRATIONS

Figure 1. General Diagram of MOHR Operations	4
Figure 2. General diagram of MOCP operations	5
Figure 3. Typical batch monitor operation during its time slice at entry point CZABAB	8
Figure 4. Typical batch monitor during handling of a private I/O request	8
Figure 5. BULKIO preprocessor operational flow diagram	13
Figure 6. CZAWY ABORT Codes	25
Figure 7. Command Routines	30
Figure 8. Macro Instructions	43
Figure 9. Time shared utility module usage (CZUFA and CZUFO)	45
Chart AA. MOHR - CZACB	52
Chart BB. Main Operator Communications Program - CZACA (Page 1 of 4)	53
Chart DD. Batch Monitor - CZABA (Page 1 of 16)	57
Chart EE. BULKIO Preprocessor - CZABD (Page 1 of 16)	73
Chart FF. Initialization - CZAWS (Page 1 of 7)	89
Chart GG. Control - CZAWT	96
Chart HH. Master Service Routine - CZAWW	97
Chart II. Attend Alert Routine - CZAWU (Page 1 of 7)	98
Chart JJ. Input Start - CZAWX (Page 1 of 4)	105
Chart KK. Input Service - CZAWZ (Page 1 of 10)	109
Chart KL. FORTRAN Data Set Converter -- CZAWC	119
Chart LL. Input Closeout - CZAWV (Page 1 of 3)	120
Chart MM. Output Service - CZAWY (Page 1 of 8)	123
Chart NN. Message Handler - CZAWM	131
Chart OA. ACK Data Set Handler - CZAWN (Page 1 of 4)	132
Chart OO. Bulk I/O Recovery Controller - CZAWA (Page 1 of 15)	136
Chart PP. Assign BULKIO Device Command Processor - CZABK (Page 1 of 4)	151
Chart QQ. FORCE - CZACG	155
Chart RR. HOLD/DROP - CZCMO (Page 1 of 5)	156
Chart RS. LABEL Command - CZABX (Page 1 of 4)	161
Chart SS. MESSAGE/BROADCAST - CZACF (Page 1 of 2)	165
Chart UU. RT - CZABF (Page 1 of 3)	167
Chart VV. SHUTDOWN - CZACN	170
Chart WW. PUNCH - CZABH (Page 1 of 3)	171
Chart XX. PRINT - CZABG (Page 1 of 7)	174
Chart YY. WT - (Write Tape) - CZABI (Page 1 of 7)	181
Chart ZA. XWTO - CZABQ (Page 1 of 2)	188
Chart ZB. PATCLEAR - CZUFO (Page 1 of 3)	190
Chart ZC. TSU DUMP/RESTORE - CZUFA (Page 1 of 8)	193
Chart ZD. TSU Processor - CZUBC (Page 1 of 3)	201
Chart ZE. PATFIX Routine - CZUPF (Page 1 of 28)	204

SECTION 1: INTRODUCTION TO MAIN OPERATOR CONTROL AND NONCONVERSATIONAL OPERATIONS

MAIN OPERATOR CONTROL

The Command System creates the system operator's task at system startup; identifies him to the system; and enables him to communicate with the Command System. It also handles messages to and from the operator, and maintains a data set (called the operator log) of messages.

NONCONVERSATIONAL OPERATIONS

The Command System contains two modules that direct nonconversational operations in the time-sharing system: the batch monitor and the BULKIO preprocessor. (Nonconversational describes tasks that are executed without direct communication with a user; they are similar to the batch jobs of a conventional computing system.)

The batch monitor initiates nonconversational tasks, keeps status records about them, and handles their termination. The BULKIO preprocessor assists in preparing for the execution of WT, RT, PUNCH, and PRINT commands, WT, PU, and PR macros, and DATASET entry from a high-speed card reader, primarily by checking parameters. The modules which carry out these commands are unique because they are always executed in separate nonconversational tasks, independent of the task that initiated them. Thus the system can execute such commands at any available time, and the user (or his task) does not have to wait for input/output operations.

Task Initiation

The user has three ways to start nonconversational tasks: he may issue an EXECUTE or BACK command from his terminal after he has logged on; he may issue a bulk input/output command (PUNCH, WT, or PRINT) or macro (PU, WT, or PR); he may submit a punched sequence of commands to the operator who will input them via a high-speed card reader or the RT command which reads a data set (data or a command sequence) from a tape volume.

For an EXECUTE or BACK command, the command analyzer and executor (see "Command System PLM") reads the command entered at the terminal and, after checking the command name, calls the EXECUTE or BACK command routine. EXECUTE validates the command operand, and sends a request to the batch monitor to start a nonconversational task. The batch monitor requests a task

status index (TSI) for the new task, and initiates the task by sending a message to it; the resulting interrupt acts in the same manner as the initial attention interrupt of a conversational task. The task monitor and CS of the new task respond to the interrupt by performing the necessary task initialization, and CS then fetches its commands from the prestored data set that serves as SYSIN for the nonconversational task. BACK changes the mode of a task from conversational to nonconversational; it uses the batch monitor only to ensure that a new background task is permissible at this time and, if it is, sets the TSI to indicate nonconversational mode. Commands will subsequently be read from the data set specified in the BACK command.

PUNCH, PRINT, and WT commands may appear in conversational and nonconversational tasks; RT commands may be issued only in the operator's conversational task. For each of these commands, the command analyzer and executor validates the command name, and calls the BULKIO preprocessor. The preprocessor ensures that the command operands are acceptable; the user will be prompted, if necessary, and the preprocessor concludes with a request to the batch monitor for task initiation. The batch monitor operates similarly for an EXECUTE. There is no actual SYSIN for bulk input/output tasks. The SYSIN is a sequence of object code supplied as a part of CS; it is not the usual prestored series of commands.

When a Punch, Print or Write Tape request is made, the batch monitor initiates nonconversational processing by first determining whether the input volume is public or private. If public, the batch monitor sends a VSEND message to task 002, the BULKIO task, which performs the actual processing within its task. If the input volume is private, the batch monitor sends a VSEND message to the appropriate command processor for private I/O requests (PUNCH, PRINT, WT). These modules build and execute a separate background job that will process the request.

Task Termination

Nonconversational tasks may be terminated either normally or abnormally. Normal termination is the same as a conversational task termination except that the SYSOUT data set, which contains all system communications to a task, must be printed. The SYSOUT data set will be listed at the computer center, and subsequently sent to

the user. Another difference is that LOGOFF, when processing a nonconversational task, notifies the batch monitor that the task has ended. This notification is required to permit the batch monitor to update its records in the Batch Work Queue data set (SYSOPER0.SYSBWQ).

A nonconversational task will be abnormally terminated either by a CANCEL command, or whenever an uncorrectable error is detected. In this case, CS forces a premature logoff. The SYSOUT of the task will be printed, and it will include a statement that reveals the cause of the abnormal termination. A batch request, from a conversational user, may be canceled (via CANCEL) before it is assigned to a task by the Batch Monitor.

Express Batch

Express Batch processing allows a series of sub-tasks to operate in a single task environment. Instead of performing a full LOGON-LOGOFF for each sub-task, sub-task accounting is handled by the RCR macros operating on the sub-task user IDs in the input stream. The Express Batch initiates processing with a full LOGON, and terminates with a full LOGOFF. Note that a sub-task referencing a private device will not be serviced in an Express Batch environment; that task will be abended (type 1) and the next sub-task will be processed.

Main Operator Control establishes the main operator's task at system startup time and handles messages to and from the system operator. Main Operator Control consists of two routines:

1. Main Operator Housekeeping Routine (MOHR) -- This initiates the main operator's task at system startup. (See Figure 1.)
2. Main Operator Communications Program (MOCP) -- This processes messages from other tasks to the operator and the operator's replies to those messages. (See Figure 2.)

Note: "Main Operator" is synonymous with "system operator." The latter term is used in the external description of the time-sharing system.

The Batch Monitor directs nonconversational operations in TSS/360. It initiates nonconversational operations, and accounts for all such operations in or awaiting execution. Then, if an operation involves private I/O devices, the batch monitor initiates a task. If not, the operation is assigned to the BULKIO task.

OPERATOR TASK GENERAL OPERATION

MOHR operates at system startup time only. It initiates the main operator's routine, and it obtains, through the MSGWR routine, the current date and time from the operator and stores these values for system use. MOHR then prompts the operator for the devices, if any, to be assigned to the BULKIO task. Also, MOHR opens the system operator log, calls the Batch Monitor Initialization routine, and asks the main operator to verify the system hardware configuration. When the operator indicates, by a reply message, that the configuration is set, MOHR issues SVCs to enable the starting of other tasks and, if the BULKIO task was assigned any devices, to initialize the BULKIO task. As its final action, MOHR calls MSGWR to tell the operator that startup is now complete.

When entered after system startup, MOCP processes messages to the operator from other tasks and handles the operator's replies. It issues, to the operator, messages received from another task. If the message requires a reply, an identifying number is issued along with the message text. This number is used by the operator

in his reply, thus linking message and reply together. MOCP then makes an entry in the system operator log. It records the text of the message just issued, the identification of the message sender, and the current system time. When a numbered reply is received, MOCP sends it to the appropriate task. MOCP scans the list of messages awaiting replies for those for which no reply has been made for too long a period; when one is found, it prompts the operator for a reply. This prompting will recur until the operator finally replies with the proper identifying number.

MOCP is activated when XIP receives an incoming message or when the operator enters the REPLY command.

Main Operator's Housekeeping Routine -- MOHR (CZACB)

MOHR performs initialization functions for the main operator's task, allows subsequent tasks to be initiated, and activates the BULKIO task. The routine operates only during startup. (See Chart AA.)

Entry Points:

CZACB2 -- To initialize SYSLOG and the Batch Monitor, and set the system clock.

CZACB3 -- To permit other tasks to begin; to activate the BULKIO task; and to SIR the Batch Monitor timer.

Modules Called:

ABEND (CZACP) -- ABEND type 2 for system error.

ASNBD (CZABK) -- Obtains devices for the BULKIO task.

Batch Monitor Initialization (CZABA) -- Initializes the Batch Monitor.

DDEF (CZAEA4) -- Creates a JFCB for SYSLOG.

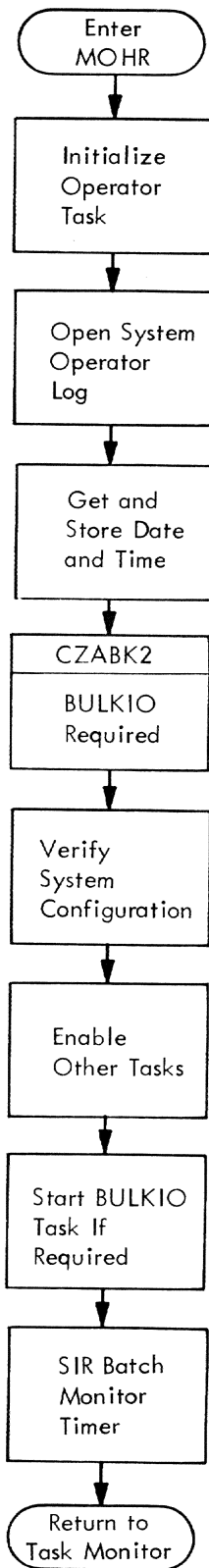
MSGWR (CZAAD2) -- Sends messages to operator requesting date and time, and indicating end of startup.

NEXTPAR (CZAAC1) -- Scans date and time entered by operator.

OPEN (CZCLA) -- Opens the SYSLOG.

QLE (CZCJQ) -- Enqueues linkage entry to CZABL1.

SCRTSI -- Creates BULKIO task.



SETTOD -- Sets time in system.

SETYMD -- Sets date in system.

SIR (CZCJS) -- Initializes batch monitor timer.

VSEND -- Sends the configuration message to the operator, then a message to activate the BULKIO task.

XTRSYS, SETSYS -- Allow initiation of other tasks.

Exits: The routine normally returns to the task monitor, via the RETURN macro instruction. If a system error occurs, the routine exits to ABEND.

Operation: Before calling MOHR, VMTI-2 has already prepared for initialization of the operator's task by calling LOGON. Two entry points to MOHR are called in sequence to provide various task initialization functions.

MOHR now prompts the operator for the current date and time so that elapsed timer readings can be related to clock time. When the date is obtained, it is checked for proper format, converted to a binary serial day (the number of microseconds since March 1, 1900), and set in the system. Similarly, the time is obtained, validated, converted to microseconds (in binary), and set in the system. At CZACB2, MOHR first initializes the SYSLOG data set, in which are recorded communications between system and operator, by issuing a DDEF macro instruction to create a JFCB for the data set, and then cataloging it. The SYSLOG data control block (DCB, which is located in MOCP and defined there as an external symbol) is then opened. Then MOHR calls CZABK2 to prompt the operator whether the BULKIO task is to be initialized, and, if so, with which devices. Next, MOHR calls CZABA1 to initialize the batch monitor. Then MOHR sends a message to the operator asking him to issue a REPLY command when the system hardware configuration has been set. MOHR does not await the reply. Instead, it calls QLE to enqueue linkage to CZABL1, then returns control to the task monitor.

Upon receipt of the operator's reply to the configuration message, control passes to MOHR at CZACB3 for the final initialization. MOCP sends the operator's reply to MOHR by a VSEND: then upon entry at CZACB3, issues an SETSYS macro instruction to allow initiation of other tasks. If the BULKIO task is to be initialized, it issues a SCRISI SVC to create the BULKIO task; and sends a message via VSEND to activate the BULKIO task. Finally, MOHR SIRs the batch monitor timer for entry to CZABA6, writes a

Figure 1. General Diagram of MOHR Operations

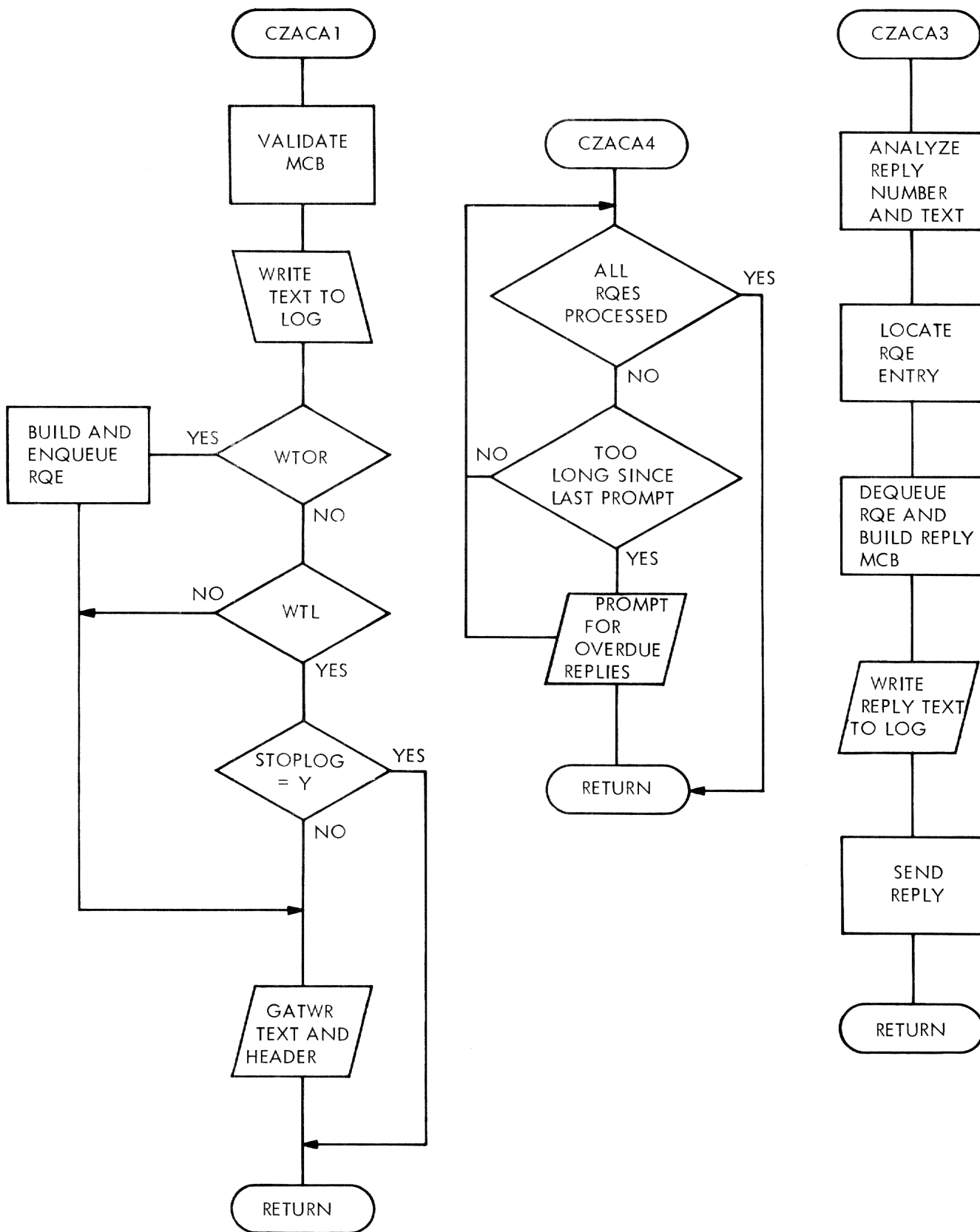


Figure 2. General diagram of MOCP operations

message to the operator telling him that startup is complete, and returns control to the task monitor.

Error Conditions: MOHR issues these minor system error codes for SYSER:

<u>SYSER Code</u>	<u>ABEND Message</u>	<u>Significance</u>
050302601	SYSTEM ERROR- BAD RETURN CODE FROM MSGWR	Invalid return code from MSGWR.
050302602	SYSTEM ERROR- BAD RETURN CODE FROM MSGWR	Invalid return code from MSGWR when asking for date.
050302603	SYSTEM ERROR- BAD RETURN CODE OR NEFARIOUS VSEND	VSEND not successful or invalid return code.
050302604	SYSTEM ERROR- NEFARIOUS RETURN CODE FROM MSGWR	Invalid MSGWR return code or MSGWR not successful.
050302605	SYSTEM ERROR- INVALID TASK ID	SCRTSI did not create a task with taskid of 2
050302606	SYSTEM ERROR- INVALID DDEF	Invalid return code from DDEF or DDEF was unsuccessful.
050302607	SYSTEM ERROR- INVALID SCAN	Invalid SCAN return code.
050302608	SYSTEM ERROR- INVALID SCAN	Invalid return code after scan of time response, from MSGWR.
050302609	SYSTEM ERROR- BULKIO INITIALIZATION VSEND	Invalid return code from the VSEND issued to activate the BULKIO task, or an unsuccessful VSEND.

System Control Block Usage: Interrupt
Storage Area (CHAISA)

Main Operator Communications Program --
MOCP (CZACA)

MOCP processes WTO, WTL, WTOA, and WTOR messages to the operator, replies from the operator, and prompting messages for overdue replies. (See Chart BB.)

Entry Points:

CZACA1 -- Processes WTO, WTL, WTOA, WTOR
MCBs.

CZACA3 -- Processes replies to WTOR.

CZACA4 -- Processes overdue replies.

CZACA5 -- Put message in SYSLOG.

Modules Called:

CLOSE (CZCLBC) -- Closes (Type T) SYSLOG.

CZCJQS -- Queues linkage to CZACB3 (MOHR).

EBCDIME -- Converts system time to YMD/HMS
format.

FREEMAIN (CZCHA3) -- Releases storage.

GATWR (CZATC1) -- Writes messages to the
operator's terminal.

GDV (CZASDX) -- Determines STOPLOG value.

OPEN (CZCLAO) -- Opens SYSLOG.

PRMPT (CZATJ) -- Writes error or prompting
messages.

REDTIM -- Gets current system time.

Internal Subroutines:

CAIN -- Generates entry code.

CAWRLOG -- Writes the message and header to
SYSLOG (in CSECT CZACA4).

Exits: The routine returns to the caller
via the RETURN macro instruction.

Operation: At CZACA1, the entry to process incoming MCBs, register 1 contains the address of the MCB containing the message. The system time is obtained, and the MCB and OPH are checked for consistency. When the MCB is a cancellation for a WTOR message, the corresponding Reply Queue entry is erased; the MCB is converted to an action message (WTOA) which instructs the operator to ignore the specific WTOR message. For a high priority action message to the operator (WTOA), arrows are placed in front of the message text. At this point, the routine assigns a reply number for any message requiring a reply. The message length is calculated, and the routine writes the message to SYSLOG. If a reply is expected (WTOR), the routine completes the Reply Queue Header and puts the MCB on the Reply Queue. Then, unless the message is WTL and the default value of STOPLOG is 'Y', the message is written to the terminal. Finally, unless a reply is expected (WTOR), the MCB storage is released.

At CZACA3, the entry to process replies to WTOR requests, the reply number and text are checked for validity, and the Reply Queue is scanned for a matching reply. The routine now dequeues the RQE, builds an

MCB, and writes the reply to SYSLOG. The RQE storage is now freed. If the sending and receiving Task IDs are the same, linkage is set to CZACB3 (MOHR); otherwise, the reply is sent via VSEND.

At CZACA4, the entry to prompt for overdue replies, the routine obtains the system time and calculates the time limit for each RQE. If the last prompt time is less than this time limit, the operator is prompted for a reply and the last-prompt time is updated to the current time.

At entry CZACA5, the input parameters, including current system time, are moved into the header; subroutine CAWRLOG is called to write a message to the operator log. CAWRLOG writes the preceding record, via PUT. Then, it builds the log entry header: text is moved into the buffer, to a maximum of 132 bytes, and written to the log via PUT; additional 132-byte records are also written to completely transfer the text. When each quarter-page is written, a type 'T' close is performed on SYSLOG so that the log will be current should the system crash.

In all cases, exit is made to the calling program.

Error Conditions: Diagnostic messages are issued for the following errors:

At CZACA1:

1. No text in the message.
2. The reply code is non-zero.
3. A reply is expected to a WTL message.
4. A reply is indicated, but no reply code.
5. A reply is expected, but no MEB address.
6. The reply code is greater than the maximum.

At CZACA3:

1. There is no input.
2. The user is not privileged to use the command.
3. The reply number is invalid.
4. The reply length is too great.
5. No matching reply is found.

System Control Block Usage:
Data Control Block (CHADCB)

Message Control Block (CHAMCB)

Operator Header (CHAOPH)

Reply Checking Table (CHARET)

Reply Queue (CHARQE)

System Activity and Resources Table (CHASAR)

System Log (CHALOG)

Task Common (CHATCM)

BATCH MONITOR GENERAL OPERATIONS

The Batch Monitor is invoked by the Main Operator Housekeeping Routine (MOHR) as a part of the operator task during system startup. Batch Monitor and the operator task comprise task 001. The Batch Monitor maintains a record of every nonconversational task or job being executed or awaiting execution. Each time a request to initiate a nonconversational operation is received via external interrupt, a subprocessor of Batch Monitor enters the request in the batch work queue (BWQ) and sets an interrupt flag (see Figures 3 and 4). After all such interrupts have been processed, the Batch Monitor will attempt to initiate independent nonconversational tasks and dispatch job requests to the BULKIO task.

The entry remains in the BWQ until the task or job is finished. At that time, the LOGOFF command routine (or the BULKIO task) informs the Batch Monitor, and the entry for the task or job is deleted from the BWQ. Batch Monitor checks for waiting requests on each of its time slices. When no requests exist, Batch Monitor puts itself in the wait state.

Batch Monitor (CZABA)

The Batch Monitor consists of subprocessors with separate entry points to perform the following functions:

- To initiate all nonconversational tasks and jobs, except those started by the BACK command routine;
- To assign a batch sequence number (BSN) to each nonconversational operation;
- To scan the work queue for initiation requests from the BULKIO preprocessor. (See Chart DD.)

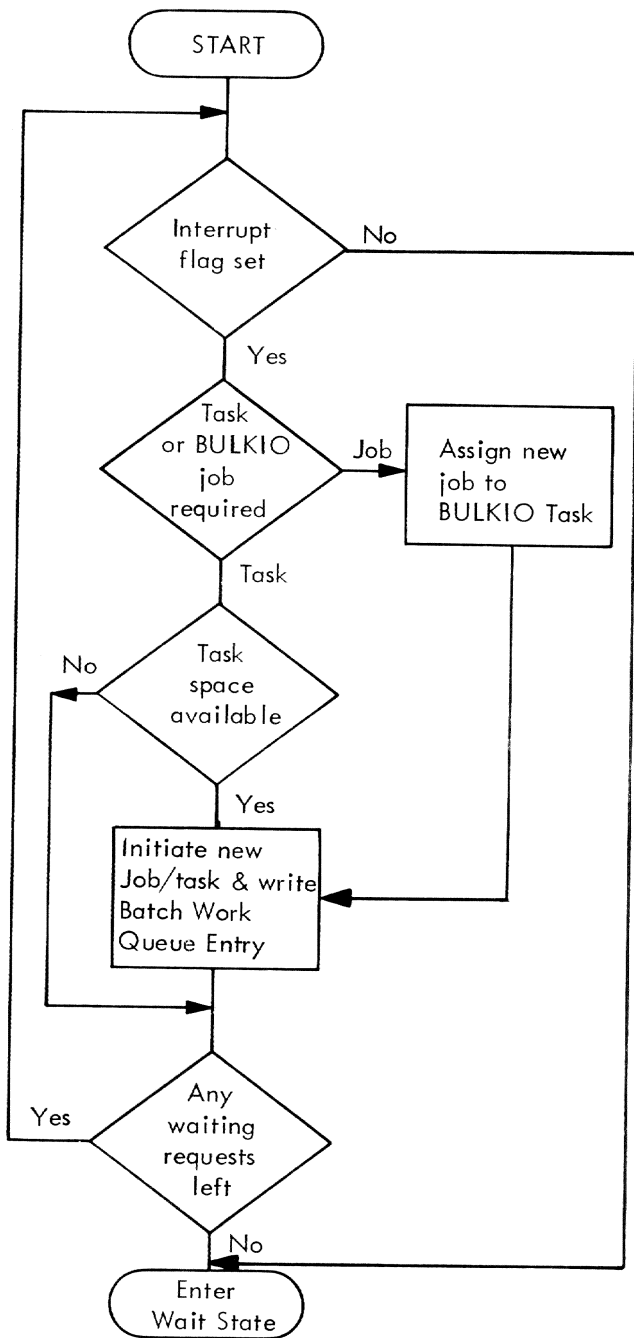


Figure 3. Typical batch monitor operation during its time slice at entry point CZABAB

Entry Points:

CZABA1 -- Entry point to the general control processor of the Batch Monitor. This entry point is used at system startup to initialize the Batch Monitor for new tasks.

CZABA2 -- Entry point to the EXECUTE sub-processor to secure a batch sequence

number for the task and insert an entry into the BWQ.

CZABA3 -- Entry point to the CANCEL sub-processor to cancel operations.

CZABA4 -- Entry point to the BULKIO sub-processor to handle initiation requests from CZABD for PRINT, PUNCH, WT and RT operations.

CZABA5 -- Entry point to the BACK sub-processor to assign a batch sequence number (BSN) to a background task and create an ART Table entry for it.

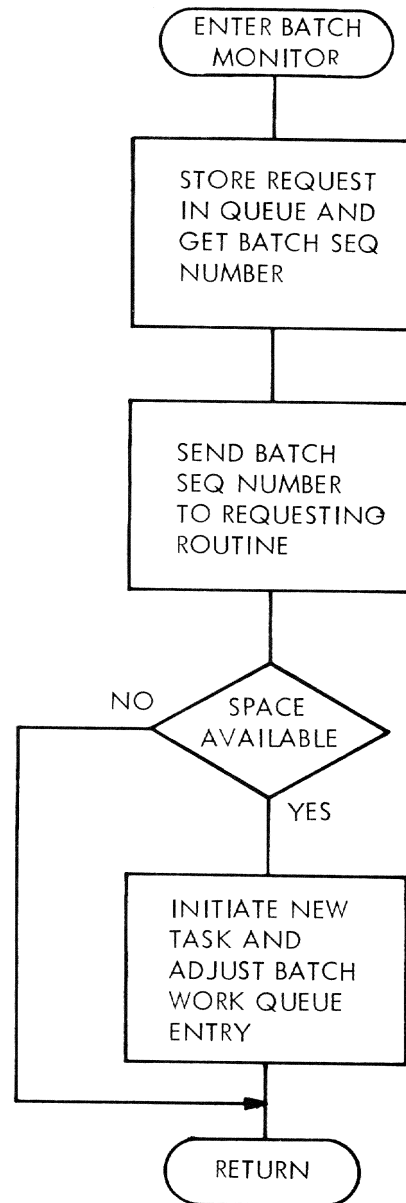


Figure 4. Typical batch monitor during handling of a private I/O request

CZABA6 -- Entry point to timer interrupt routine to scan BWQ for private print or punch task entries awaiting use of devices assigned to the BULKIO task.

CZABA7 -- Entry point to the LOGOFF subprocessor to complete LOGOFF for a BULKIO job or task.

CZABA8 -- Entry point to the DIRECT command processor. This allows BWQ output entries that are queued for RJE stations to be re-routed to other RJE stations or to an on-line printer.

CZABA9 -- Entry point to the SHUTDOWN subprocessor to force an ABEND for each active task in the BWQ or ART Table.

CZABAB -- Entry point to the Batch monitor processor. This entry is used to process the entries in the batch work queue (BWQ) and to initiate BULKIO operations and EXECUTE tasks.

CZABAC -- Entry point for EODAD.

CZABAD -- Entry point to SYNAD exit.

CZABAE -- Entry point from Batch Wait routine (CZABL).

CZABAF -- A flag defined as an entry point to allow MOCP to set this interrupt flag to on.

CZABL1 -- Batch Wait routine to put Batch Monitor into TWAIT state awaiting interrupt; a unique CSECT.

CZABL2 -- An interrupt flag defined as an entry point.

CZABAQ -- Entry to BWQ DCB for EXHIBIT processing in main operator task.

Modules Called:

ADDCAT (CZCFA1) -- Creates a data set descriptor in the user's catalog.

ALFNUM (CZAAC3P) -- Validates parameter string.

CLOSE (CZCLBC) -- Closes a data set.

DDEF (CZAEA4) -- Creates a JFCB in the TDT.

DDEF (CZAEA5) -- Defines TSS*****.RJEACK

DELCAT (CZCFD1) -- Removes a data set descriptor from the user's catalog.

DELREC (CZCPH1) -- Deletes an entry from the BWQ data set.

ERASE (CZAEJ3) -- Erases data sets.

FIND (CJCOJ1) -- Finds a valid member in the acknowledgement data set.

FINDDS (CZAEC1) -- Finds BWQ JFCB.

FREEMAIN (CZAHA3) -- Releases virtual storage.

GATWR (CZATC1) -- Sends messages to the console.

GETMAIN (CZCGA3) -- Gets virtual storage for the ART table.

GETSBLOCK (CZCFG4) -- Updates sharing descriptor.

LOCATE (CZCFL1) -- Locates a specified output data set name.

MOCP (CZACA4) -- Interrupts the Main Operator Task.

OBEY (CZASA1) -- Permits BWQ to all.

OPEN (CZCLA0) -- Opens the BWQ data set.

PROMPT (CZATJ1) -- Sends messages to the operator.

QLE (CZCJQS) -- Enqueues an entry in the Task Monitor.

READ (CZCPE1) -- Reads BWQ entries.

REL (CZAFJ3) -- Releases the ACK data set.

SARD Processor (CZAYE1) -- Gathers display data for EXHIBIT command.

SHUTDOWN (CZACN3) -- Exits when all tasks have been shutdown.

SIR (CZCJSA) - Specifies reentry to CZABA6 by timer interrupt.

WRITE (CZCPE1) -- Writes entries to the BWQ data set.

Exits: The batch monitor normally exits to the task monitor. When shutdown is complete, the routine exits to SHUTDOWN. If a minor system error occurs, the routine indicates the error and attempts to continue processing.

Operation: The batch monitor, at CZABA1, is called by the main operator housekeeping routine (MOHR) as part of the Operator task at system startup, to initialize the batch monitor task. The batch monitor initialization processor creates the active remote task (ART) table and performs a DDEF on the batch work queue (BWQ). After the batch monitor has been initialized, it sends a VSEND message to the operator, and returns to MOHR.

If the batch work queue already exists, the initialization processor makes a pass through the queue to initiate any tasks left remaining at shutdown. It erases unnecessary SYSIN and SYSOUT data sets, turns off in-service indicators, deletes unnecessary batch work queue entries and updates the batch work queue data set.

At periodic intervals, the batch monitor processor (CZABAB) checks for waiting requests to process tasks. These requests are indicated by an interrupt flag set by one of the batch monitor subprocessors each time a request to initiate a task is received by the subprocessor from a command routine. The subprocessor will set the interrupt flag and return control to the task monitor. The task monitor will then call the batch monitor processor at CZABAB to initiate the task.

If no interrupt has occurred, and all nonconversational tasks have been abended due to a shutdown, the batch monitor exits via a no-return call to SHUTDOWN. If, however, shutdown is not completed the batch monitor returns to the wait state. Any interrupts occurring while shutdown is in progress will cause the batch monitor to return control to the task monitor.

The batch monitor processor (CZABAB) processes BWQ data set entries; it searches, sequentially, for jobs or tasks awaiting initiation. For BULKIO jobs, the type of entry is first determined (private or public). Public entries can be either print jobs or punch jobs. For a public entry, the BULKCOMM Table's S-entry list is searched for a printer or punch device-type match to the BWQ entry. If found, the S-entry is completed from data in the BWQ. If no match is found, the BWQ entry remains enqueued; processing continues with the next BWQ entry.

Private device entries can be print, punch, read tape, or write tape jobs. For each private entry a search is made through SDAT to determine if the devices required by the pending task are available. If so, they are reserved, a task is created, a "do not dismount the user's private volume" message is issued at the console, and a VSEND is executed to the new task's external interrupt processor to initiate that task. If devices are not available, the entry remains enqueued; processing continues with the next BWQ entry. Messages are issued at the console stating "BULKIO JOBS REQUIRING PRIVATE DEVICES
. . .PRINTnnnn,PUNCHnnnn."

If the specific BWQ entry is not a BULKIO type job, the Batch processing logic is called to determine if a TSI slot is available and if the maximum number of tasks

(other than BACK) have not been reached. If yes, it then sends a message via VSEND to the new task's external interrupt processor to initiate the task. If an entry can not be dispatched, it remains on the queue; processing continues with the next BWQ entry.

The following is a description of the various subprocessors which set interrupt flags to indicate that a request for task processing has been sent to the batch monitor from various command routines.

The BULKIO Preprocessor (CZABD) either calls, or VSENDS to, the batch monitor at entry point CZABA4 (BULKIO subprocessor). At this entry point the subprocessor stores the request and the device codes in the batch work queue, assigns a BSN to the operation, and sends the batch sequence number to the BULKIO preprocessor. The subprocessor then sets an interrupt flag indicating to the batch monitor that a BULKIO operation is pending.

The timer interrupt processor (CZABA6) acts as an interrupt processor to perform a scan of the BWQ for private operation entries requiring devices assigned to the BULKIO task. The interrupt pending flag is set to notify the batch monitor to return from the TWAIT state.

CZABA3 is the entry point for the interrupt processor handling requests from the CANCEL command routine. The subprocessor searches the batch work queue or the ART Table for the operation to be canceled. The cancel flag is set and a message is sent to CANCEL the active task. A message is then sent to the CANCEL command to record the command's acceptance.

The execute subprocessor (CZABA2) accepts requests from the EXECUTE command routine. The subprocessor stores these requests in the BWQ, assigns a BSN to the task and VSENDS the BSN to the EXECUTE command routine. This subprocessor also fields requests from the BULKIO task's input closeout routine (CZAWV) to assign a BSN to a SYSIN command procedure detected during a card read operation except for an RJE request from BULKIO. In this case, the BSN is assigned by the BULKIO task.

The background subprocessor (CZABA5) accepts requests from the BACK command routine. The third-level TSI availability is determined, the requests are stored in the ART table, and a BSN is assigned to the task. A VSEND is then made to the command routine recording the acceptance of the request. If the maximum number of tasks has been reached, the command routine is informed (via VSEND) that the request cannot be accepted.

The LOGOFF subprocessor (CZABA7) accepts messages from tasks being ABENDED or needing to be logged off or from the BULKIO task to signal the end of a job. The ART table and BWQ are searched to locate the task requesting to be logged off. When located, the entry is removed from the ART table or deleted from the BWQ, unless the BULKIO task is being ABENDED. In this case, the BULKCOMM-initialized flag is set off, and the active task count is decremented.

When a SYSIN data set is associated with the task being deleted, the SYSIN data set is erased.

The DIRECT command processor, CZABA8, is entered from the command analyzer and executor to process the operator's DIRECT command which reroutes RJE output to either an on-line device or another RJE device. The routine validates the first parameter and checks for a second. If the second parameter is present, it is validated. Following this, or if no second parameter was present, the routine finds the BWQ entry for the device requested in parameter 1, and either converts all output to on-line devices, or, if a specific device was mentioned, routes the output to that RJE station. Control is then returned to the calling program.

The shutdown subprocessor (CZABA9) is the interrupt subprocessor to handle messages from the SHUTDOWN command routine. This subprocessor sets the task initiation inhibit flag, sets the shutdown flag in all in-service BWQ entries, and sends a message to all active tasks and to the BULKIO task, to force an ABEND.

CZABL1, the Batch Monitor Wait routine, sets an enter code for SVC 121, and links to CZABAE. At this point, control is passed to ENTERO, within CZABAB, where an entry is put on the task monitor privileged queue. If an interruption results, this process is repeated.

CZABL2 is a direct link to SVC 229, the TWAIT processor.

Error Conditions: Routine issues these minor system errors:

SYSER Code	Message	Significance
050202501	ABEND None	SYNAD exit taken when searching batch work queue following a GET; processing continues with another GET.
050202502	None	SYNAD exit taken after unsuccessful VISAM write

with retrieval; processing continues as if WRITE had been successful.

050202503	None	Pre Logon Abend prior to Optask/Batch Monitor initialization in start up. System will loop in CZABA PSECT after issuing RUN.
050202505	None	Invalid return code returned by the SIR macro instruction.
050202514	None	The BULKIO routine requesting task initiation did not set batch work queue field correctly.
050202515	None	An uncataloged data set for a read tape task has an incorrect tape operand and does not indicate 7, 9, or 7DE track.
050202828	None	LOCATE unable to return data set descriptor level SBLOCK for this task.
050202530	None	ADDCAT could not complete addition to catalog.

System Control Block Usage:
Assign BULKIO Device DSECT (CHAABD)

Batch Work Queue (CHABWQ)
BULKCOMM S-Entry (CHASET)
BULKCOMM Table (CHABCT)
Catalog Common Area (CHACDS)
Catalog S-Block (CHACCC)
Data Control Block (CHADCB)
Device Available Table (CHAHED)
Message Control Block (CHAMCB)
Message Event Block (CHAMEB)
Screen Routines Common Area (CHASCA)
Subqueue Entry (CHAVE)
Subqueue Header (CHAAHD)
Symbolic Device Allocation Table (CHASDA)
System Activity and Resource Table (CHASAR)

System Common (CHASCM)

Task Common (CHATCM)

Task Data Definition Table (CHATDT)

Task Status Index (CHATSI)

The BULKIO preprocessor directs some nonconversational processing in TSS/360, by assisting the batch monitor in implementing the WT, PU, and PR macros and the WT, PUNCH, RT, and PRINT command routines.

GENERAL OPERATIONS

The BULKIO preprocessor receives all requests for card punching, printing, tape reading, and tape writing from the command routines PRINT, PUNCH, RT, and WT, and from the PR, PU, and WT macro instructions via type-I linkage. It validates each response and passes it on to the Batch Monitor (see Figure 5). The preprocessor also processes the DATASET control card, which marks the start of a data set read in by input service from a high-speed card reader. The Command Analyzer and Executor calls BULKIO preprocessor when a PUNCH, PRINT, RT, or WT command is encountered.

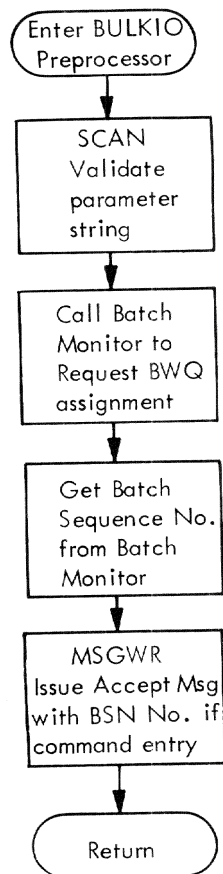


Figure 5. BULKIO preprocessor operational flow diagram

BULKIO Preprocessor (CZABD)

The BULKIO preprocessor processes RT, WT, PRINT, and PUNCH commands and the WT, PR, and PU macro instructions, and validates DATASET cards. If the parameters are specified correctly, a request message is created and passed to the Batch Monitor to initiate a nonconversational operation. (See Chart EE.)

The BULKIO preprocessor has eight entry points: one to each executive routine used to process commands; three secondary entry points to the PR, PU and WT executive routines to process the macro expansion; and an entry point to an executive routine to process parameters on a DATASET card (except for the user identification).

Entry Points:

- CZABD1 -- To DATASET executive routine.
- CZABD2 -- For RT command routine.
- CZABD3 -- For PR macro instruction.
- CZABD4 -- For PRINT command routine.
- CZABD6 -- For PUNCH command routine.
- CZABD7 -- For PU macro instruction.
- CZABD8 -- For WT command routine.
- CZABD9 -- For WT macro instruction.
- CZABDA -- For RJE SYNAD.
- CZABDB -- For RJE EODAD.

Modules Called:

- ABEND (CZACP1) -- Terminates the conversational or nonconversational user task.
- Batch Monitor (CZABA4) -- Sends request message to the Batch Monitor and awaits reply when requesting user task is SYS-OPER0, TASKID=0001.
- CATALOG (CZAEI2) -- Temporarily catalogs a non-VAM data set for erasure, if requested, after task completion.
- CHEKDS (CZAAC2) -- Validates dsname.
- CHKNUM (CZAAC5) -- Checks for a valid numeric value and converts to binary.
- CLOSE (CZCLBC) -- Closes ACK data set.
- DDEF (CZAEA5) -- Defines TSS*****.RJEACK.

FIND (CZCOJ1) -- Finds valid RJE station ID.

FINDDS (CZAEC1) -- Finds a JFCB or creates one.

LOCATE (CZCFL1) -- Locates a catalog entry (two calls).

MSGWR (CZAAD2) -- Sends message to SYSOUT (three calls).

NEXTPAR (CZAAC1) -- Gets next parameter and the subsequent delimiter.

OPEN (CZCLA0) -- Opens ACK data set.

PRMPT (CZATJ1) -- Issues error messages.

READ (CZCPE) -- Reads RJE station ID.

RELEASE (CZAFJ3) -- Releases JFCB for uncataloged data set and for cataloged private volume data sets used by RT, PRINT, PUNCH, and WT (CZABF, CZABG, CZABH, CZABI).

VSENDR (CZABQ1) -- Sends request message to the Batch Monitor and awaits reply, only if requesting user task is not SYSOPER0 (TASKID=0001).

Operation: The BULKIO Preprocessor consists of executive routines (DATASET, RT, PRINT, PUNCH, and WT); some common processing subroutines; and several parameter processing routines, one for each parameter possible in the commands. These parameter processing routines may be used for similar parameters from different macro instructions and commands.

Each executive routine builds and sends to the batch monitor a message containing the message code, batch monitor taskid, BULKIO preprocessor taskid, activity code, userid, password, charge number and associated parameters for the particular command. After receiving the reply message from the batch monitor, the BULKIO preprocessor issues a command acceptance message, giving the batch sequence number assigned by the batch monitor, and returns to the calling routine.

The executive routine performs house-keeping and parameter checking for the commands. At each entry point the common processing subroutine INITL is called to initialize the routine. For each parameter, any necessary flags and switches are set, and CNPAR, the common processing routine, is called. CNPAR processes the appropriate parameter and returns control to the executive routine.

If the parameter has been defaulted or must be prompted for, CNPAR will return a

code to the executive routine indicating this. The executive routine then calls CNPAR at its second entry point, CPPAR, to either prompt for the parameter or insert the default parameter.

Error Conditions: The BULKIO Preprocessor issues these minor system error codes for SYSER:

<u>SYSER Code</u>	<u>ABEND Message</u>	<u>Significance</u>
050202801	INVALID OR UNEXPECTED RET CODE FROM A CALLED ROUTINE	Invalid tape type in system common.
050202812	Same as above	Invalid return code from LOCATE.
050202813	Same as above	Invalid return code from CHEKDS.
050202814	Same as above	Invalid return code from FINDDS.
050202816	Same as above	Invalid return code from VSENDR.
050202817	INVALID OR UNEXPECTED RET CODE FROM A CALLED ROUTINE	Invalid return code from MSGWR (when no response was expected by the user).
050202818	Same as above	Invalid return code from CHKNUM.
050202819	Same as above	Invalid return code from MSGWR (when a response was expected by the user).
050202820	Same as above	Invalid return code from NEXTPA.

System Control Block Usage:
Batch Work Queue (CHABWQ)

Catalog S-Block (CHACCC)

Data Control Block (CHADCB)

Interrupt Storage Area (CHAISA)

RJE Station Entry (CHASTA)

System Common (CHASCM)

Task Common (CHATCM)

Task Data Definition Table (CHATDT)

User Table (CHAUSE)

The BULKIO task (TASKID=002) accomplishes nonconversational input/output operations on public unit record devices and is responsible for printing from, punching from, and reading cards into system public disk storage (VSAM and VISAM only). BULKIO is created, at system startup, if the operator assigns system devices (printers, punches, or readers). Once created, BULKIO remains active in the system.

If the operator does not create the BULKIO task at startup, he may create it later with the ASNBD (Assign Bulkio I/O Devices) command. ASNBD assigns devices to, and creates, the BULKIO task. Once assigned, the device is allocated to BULKIO until it is deleted (by ASNBD), system shutdown occurs, or the task is abnormally terminated (ABEND).

Remote Job Entry (RJE) is an extension of the BULK I/O task which allows reading cards from, and printing data to an RJE device using the MSAM modules for physical I/O.

The BULKIO task and the batch monitor (in task 001) share the BULKCOMM table (CHABCT). An active table entry (CHASET) describes the current status of an assigned device, including control information for MSAM. The Bulk I/O performance table (CHABPT) is used to establish appropriate cycle times based on the speed of the active devices assigned to task 002.

To distinguish individual operations performed by BULKIO from the continuous BULKIO task, an individual operation is termed a "job." While a job is assigned to BULKIO, the S-entry in the BULKCOMM table also contains all control and status information for the job.

- Notice that the BULKIO task encompasses public I/O jobs on unit record devices. Private I/O requests become independent nonconversational tasks, initiated by the Batch Monitor and are completely independent of the BULKIO task.
- Output job assignment to BULKIO is done by the Batch Monitor. After Batch Monitor scans the BWQ and ascertains a public request, the information is transferred to the proper entry in BULKCOMM.
- Input jobs (card reader) to BULKIO are not put into the BWQ. No command requ-

est is necessary. The system operator manually performs a not-ready-to-ready operation on the card reader. The BULKIO Input Service routine fields the asynchronous interrupt and reads in the cards. Card input data sets are put on public storage; command strings are assigned a SYSIN sequence number by the BULKIO task and recorded in a BWQ entry for later nonconversational execution.

BULKIO is a multiprogramming task. During its time slice, a dispatcher routine, Master Service, scans the S-entries in BULKCOMM and dispatches to the input and output service routines.

Each service routine fills (PRINT, PUNCH) or empties (read cards) the data buffers provided by MSAM. MSAM subsequently empties (PRINT, PUNCH) or fills (read cards) the buffers while the next job entry is being serviced by BULKIO.

The BULKIO task is cyclic: When it has provided work for all its active unit record and RJE devices, it becomes dormant (enters a wait state) until one of its devices is ready for more work.

BULKIO has its own recovery routine for abnormal termination attempts. Most common error conditions can be recovered from, leaving the BULKIO task actively running and deleting the faulty device and its job. Serious error conditions cause the task to be deleted, but the operator can recreate it with an ASNBD command. The BULKIO Task contains the following modules:

ABEND Recovery (CZAWA)
 Attend Alert Routine (CZAWU)
 Control Routine (CZAWT)
 Initialization Routine (CZAWS)
 Input Closeout Routine (CZAWV)
 Input Service (CZAWZ)
 Input Start (CZAWX)
 Master Service (CZAWW)
 Message Handler Routine (CZAWM)
 Output Service (CZAWY)

BULKIO Initialization Routine (CZAWS)

This routine initializes the BULKIO task upon receipt of the first ASNBD VSEND from the operator task. Thereafter, it processes subsequent ASNBDs by adding or deleting unit record devices from the BULKIO task in response to the ASNBD command. It can also be called by BULKIO ABEND recovery (CZAWA) to dynamically add a device which was deleted by an MSAM ABEND. CZAWS also receives wake-up VSENDS from batch monitor to BULKIO. (See Chart FF.)

Entry Points:

- CZAWS1 -- Entered from ASNBD and wake-up VSENDS.
- CZAWS2 -- ASNBD message area.
- CZAWS3 -- Entered by ABEND recovery (CZAWA) to add deleted device.
- CZAWS4 -- Entered to clean up if CZAWS abnormally terminates via ABEND.
- CZAWS5 -- First RJE S-entry.
- CZAWS6 -- Message length for ASNBD confirmation.

Modules Called:

- BMSG (CZAWM1) -- Writes messages to operator.
- CLOSE (CZCLBC) -- Closes device DCB.
- DDEF (CZAEA4) -- Defines MSAM device.
- ERASE (CZAEJ3) -- Erases ACK data set.
- FIND (CZCOJ1) -- Finds valid ACK member.
- FREEMAIN (CZCHA3) -- Releases storage used for input MCB.
- Input Closeout (CZAWV1) -- Releases reader, deletes S-entry, cleans up job in progress.
- OPEN (CZCLA0) -- Initializes MSAM data set.
- QLE (CZAJQS) -- Queues on Task Monitor.
- RELEASE (CZAFJ3) -- Returns MSAM device for ASNBD delete or aborted ASNBD add.
- SIR (CZCJSA) -- Specifies interrupt routines for CZAWT's VSEND to itself and for asynchronous interrupts from readers.
- VMTI-2 (CZATD2) -- Performs system task initialization.

Exits: This routine normally returns to the caller. If a SIR cannot be done for CZAWT's VSEND to itself, SYSER 052702601 is

issued and CZAWA1 is called for a terminal ABEND. If CZAWS abnormally terminates in such a way that recovery can be made, CZAWA calls CZAWS3 to clean up and return to the original caller.

Operation: When CZAWS initializes the BULKIO task, it does the following: indicates to the system that this task has special purpose ABEND and DDEF handling requirements; calls CZATD (VMTI-2) to perform system task initialization; and SIRs the ICB for CZAWTs VSEND to itself.

An ASNBD VSEND can either designate a series of adds and deletes, or can be a default ASNBD indicating that all available non-RJE unit record devices are to be acquired for the BULKIO task. For the non-default case, each device request is handled individually -- errors detected on one request usually do not affect those following.

For an add, BULKCOMM is searched to ensure that the requested device does not already belong to BULKIO. An empty S-entry slot is found in the course of the search and an MSAM DCB is built there; the device is defined and the DCB opened. In the case of a reader, an ICB for asynchronous interrupts is built and a SIR is issued. For an RJE device, two S-entry slots are found, and two S-entries are built.

For a delete, BULKCOMM is searched to make certain the BULKIO owns the device. If the S-entry is not currently assigned, a RELEASE is done and the S-entry is marked deleted. If the S-entry is assigned for output entries, flags are set such that Output Service (CZAWY) will delete the device on completion of normal BULKIO processing. If the output device is an RJE station, and it is printing Acknowledgement Data Sets (ACK), then the device will not be deleted. For input entries, Input Closeout (CZAWY) is called immediately to delete the device and clean up the current job if there is one; the S-entry is marked deleted. For an RJE device, both S-entries are deleted.

For the default case, the available device tables (CHAHED, CHAAHD, CHAAVE) are used to index into the SDAT where a search is made for available unit record devices. An available device is set unavailable and then added as if originally an ASNBD add had been requested for it.

When the ASNBD VSEND has been processed, a message is sent to the operator indicating which ASNBDs were successful, and giving the associated S-entry numbers; a wake-up VSEND is then sent to the Batch Monitor. A QLE to Control (CZAWT) is done if this

was the first ASNBD (that is, BULKIO was just initialized).

System Control Block Usage:
ASNBD Entry (CHAABD)

Availability Table Header (CHAHED)

Batch Work Queue (CHABWQ)

BULKCOMM (CHABCT)

Bulk I/O Performance Table (CHABPT)

Data Control Block (CHADCB)

DCB Header (CHADHD)

Interrupt Storage Area (CHAISA)

Message Control Block (CHAMCB)

S-entry (CHASET)

Symbolic Device Table Entry (CHASDA)

Task Common (CHATCM)

Task Data Definition Table (CHATDT)

Task Status Index (CHATSI)

BULKIO Control Routine (CZAWT)

This routine determines the manner and frequency of cycling of the BULKIO task. It also gathers statistics on timing for this task. (See Chart GG.)

Entry Points:

CZAWT1 -- Main entry point; entered because of QLE from CZAWS during initialization or because of VSEND to CZAWT by CZAWT.

CZAWT2 -- ICB for VSEND to self; SIR from CZAWS.

Modules Called:

BULKIO ABEND Recovery (CZAWA1) -- Is called following a SYSER when the hardware timer cannot be initialized via SIR.

Master Service (CZAWW1) -- Cycles through S-entries to dispatch input/output.

Exits: The routine normally exits to the task monitor. If unable to initialize (via SIR) the timer, a SYSER 052702701, and a call to CZAWA1 for BULKIO terminal ABEND, are made.

Operation: The BULKIO task can operate in either of two modes: interrupt-driven (CHBBIO='08') or timer-driven (CHBBIO='00'). During either mode, an external interrupt from the operator task or an asynchronous interrupt from a device

assigned to BULKIO will initiate a BULKIO cycle.

For timer-driven mode, BULKIO uses the BULKIO Performance Table (CHBBPT) to determine when its fastest active device will be ready for more work. CZAWT then issues a SIR macro instruction which sets the timer for the appropriate interval so that BULKIO can awaken in time to keep all of its active devices busy.

For interrupt-driven mode, synchronous interrupts from the BULKIO devices are enabled. However, the supervisor will not awaken BULKIO for a synchronous interrupt unless the interrupt is for the last currently outstanding I/O request for a BULKIO device. When awakened, BULKIO returns to the task monitor so that all previously queued interrupts can be processed. The last one that is processed is the QLE by CZAWT to itself.

When CZAWT is reentered, timing statistics are collected and Master Service is called to cycle through the S-entries. On return, if no S-entries are active, or if BULKIO is in interrupt-driven mode, all interrupts are enabled and a TWAIT macro instruction is issued. If BULKIO is in timer-driven mode, the real time used during this cycle is calculated. If this time exceeds the BULKIO base time (BCTTIM), interrupts are enabled and a VSEND is issued to resume processing. If the base time has been exceeded, all but synchronous interrupts are enabled, the remaining time is entered (by issuing SIR) into the timer, and a TWAIT is issued. Thus, in timer-driven mode BULKIO cycles only once per base time, unless an external or asynchronous interrupt causes it to awaken earlier.

Note: The system default for BULKIO is interrupt-driven mode because BULKIO usually cycles most efficiently in this mode. However, an installation may wish to run BULKIO in timer-driven mode in order to maximize throughput time at the expense of general system performance or, conversely, to maximize user response time at the expense of slowed operation of unit record devices. To operate BULKIO in timer-driven mode, the installation must set CHBBIO = X'00' and adjust BCTDEF and CHBBPT as required.

System Control Block Usage:
BULKCOMM (CHABCT)

BULKIO Master Service (CZAWW)

This routine is the S-entry scanner/dispatcher for the BULKIO task. (See Chart HH.)

Entry Points:

- CZAWW1 -- Main entry, from CZAWT.
- CZAWW2 -- From CZAWA to continue after ABEND.
- CZAWW6 -- Normal return from CZAWU.
- CZAWW9 -- From CZAWY and CZAWZ as normal return.
- CZAWWA -- From CZAWU as error and bypass-of-dispatching return.

Modules Called:

- ABEND Recovery (CZAWA1) -- Abnormally terminates the task via ABEND for certain error conditions.
- ACK Data Set Handler (CZAWN2) -- Writes ACK data set message to RJE station ID.
- Attend Alert (CZAWU1) -- Is called if the S-entry alert flag, SETOTB, is on for initialization of a new print or punch from public VAM job, for changing current access position within data set, for changing MSAM device options, or for request to complete these operations.
- BMSG (CZAWM1) -- Sends messages to devices.
- Input Closeout (CZAWV1) -- Deletes an S-entry.
- Input Service (CZAWZ1) -- Is called if a read cards job is assigned to current S-entry and its card reader.
- Input Service (CZAWZ3) -- Disconnects a device.
- Output Service (CZAWY1) -- Is called if print or punch from public VAM job is assigned to the current S-entry with device.

Exits: BULKIO Master Service normally returns to Task Control (CZAWT).

Operation: The routine scans the BULKCOMM Table (CHABCT) in ascending order for all full S-entries. Except for the infrequent calls to CZAWU described above, CZAWW dispatches control to the input or output service routine, and, upon return, attempts to get service for the next higher waiting S-entry. When the upper limit is reached, a type-I return is executed back to CZAWT.

This same logic processes RJE device S-entries except when: 1) an RJE reader S-entry indicates that the line must be enabled for dial-up; 2) an RJE printer S-entry calls for printing the Acknowledgment Data Set messages to the RJE station ID. To enable the line, RJELC (SVC 232) enables the line for dial-up; a message is issued

to indicate that the line is ready for dial-up; the original CZAWW logic is entered as for any device. When the ACKs are printed, CZAWN2 is called to perform the printing; control returns, within CZAWW, to process the next S-entry.

There are no messages or system errors from CZAWW.

System Control Block Usage:
BULKCOMM Table (CHABCT)

S-entry (CHASET)

BULKIO Attend Alert Routine (CZAWU)

This routine performs operations on output jobs. (See Chart II.)

Entry Points:

- CZAWU1 -- Normal entry.
- CZAWU2 -- EODAD entry.
- CZAWU3 -- SYNAD entry.
- CZAWU4 -- ABEND entry.

Modules Called:

- CLOSE (CZCLBC) -- Closes printer or punch DCB.
- DDEF (CZAEA5) -- Defines user's public VAM data set.
- FINISH (CZCMH1) -- Semi-closes printer or punch DCB.
- GET (CZCPB) -- Retrieves record from data set.
- INTINQ (CZCJIA) -- Hastens passing of printer or punch asynchronous interrupt to CZCMD2.

Message Handler (CZAWM) -- Writes messages to operator.

OPEN (CZCLAO) -- Opens BULKIO's DCB for user VAM data set to be printed or punched; opens MSAM printer or punch device DCB.

PAIR (CZACS1) -- Prevents SDAT from being locked if ABEND occurs.

PAIR (CZACS2) -- Is called when danger of locking SDAT is over.

SETL (CZCOT) -- Positions to start of data set.

SETUR (CZCMD1) -- Sets up printer or punch for a job.

Exits: Routine exits to Master Service (CZAWW).

Operation: This routine performs several functions for the public VAM BULKIO task. CZAWU is used for output (printing and punching) from public VAM only, and does the following for normal initialization of each output job:

- It validates the BULKIO-requester's userid via RCR OPEN (an error return from RCR OPEN causes the job to be aborted).
- It defines the user's public VAM data set (an error return from DDEF will abort the job).
- It issues an RCR OPEN macro instruction to validate the user and to set up accounting for job resources.
- It opens the VAM DCB 'SETDDC' in the applicable BULKIO S-entry that is used for accessing the user's data set. For an RJE job, it adjusts the record length in the MSAM DCB to five (5) greater than the record length of a VAM DCB.
- It writes a 'CZAWU' message to the operator, informing him of an output job starting, with BSN, device SDA, and fully qualified dsname.
- It issues an MSAM SETUR macro instruction to set up the output printer or punch.
- It then returns to CZAWW, BULKIO Master Service Dispatcher, and will probably be called repetitively to reissue the MSAM SETUR macro instruction until the return code changes from 4 to 0.

CZAWU has the facility to do job-oriented operations only via PCS 'SET' statements, or by installation-written command procedures, from the system operator into the applicable S-entry. During initialization of a new job as described above, between 'SETUR' and return to CZAWW, it will start an output of a VISAM line data set from the key specified in the 'SETETL' field, if that field is non-zero (EBCD); or output of a VSAM data set from the relative record number whose positive binary value is in the 'SETMOV' field. A negative value is illegal at this time since the current record accessed before any 'GETS' is the first record in the data set.

After a job has been started and partially printed or punched, CZAWU will, if the alert flag 'SETOTB' is on:

1. Skip output of undesired VISAM line records and resume printing or punching from the record whose key is in

the 'SETETL' field if the 'SETPNT' flag is also on.

2. Skip output of undesired VSAM records and resume printing or punching from the relative record number whose positive or negative binary value is in the 'SETMOV' field if the 'SETAMT' flag is also on.
3. Restart the printing or punching from the beginning of the user's data set if the 'SETRES' flag is also on.
4. Stop the printing or punching if the 'SETTOP' flag is also on.
5. Set the MSAM inhibit-message option by setting on the printer or punch 'DCBINHMS' flag if the 'SETINH' flag is also on.
6. Turn off the MSAM inhibit-message option by clearing the printer or punch 'DCBINHMS' flag if the 'SETINH' flag is also off, providing it was set on at some previous time.
7. Set the MSAM combine-reader-and-punch option by setting on the reader 'DCBCOMBI' flag if the 'SETPUN' flag is also on.
8. Turn off the MSAM combine-reader-and-punch option by clearing the reader 'DCBCOMBI' flag if the 'SETPUN' flag is also off, providing it was set on at some previous time.
9. Change the number of printer or punch data buffers used by MSAM in 'SDAMRB' to the binary value set into the 'SETBUF' field if the 'SETCBU' flag is also on.
10. Change the printer spacing to the EBCDIC value set into the 'SETPRS' field if the 'SETXPS' flag is also on. (CZAWU actually changes the DCB, CZAWU recognizes the operator's request, finishes the MSAM DCB, and notifies CZAWY.)

The reason for alerting (calling) CZAWU must be an accompanying condition such as a new job, an S-entry flag in the SETFL1 to SETFL4 bytes, or one of these flags plus a value set into a S-entry field by the system operator. The reason for the alerting of CZAWU is determined in CZAWU itself; CZAWW recognizes only the alert flag, not the alert intention.

BULKIO Input Start (CZAWX)

This routine performs the initialization requirements for a read cards job. (See Chart JJ.)

Entry Points:

- CZAWX1 -- Fields asynchronous interrupts, DIRs the ICB, and calls CZAWZ1 to request task initiation.
- CZAWX2 -- From CZAWZ to perform initialization service for new read cards job.
- CZAWX3 -- Simulated asynchronous entry used by CZAWA to delete the ICB and to return to Master Service or Task Monitor.
- CZAWX4 -- Provides direct return to task monitor.

Modules Called:

- ABEND Recovery (CZAWA) -- Is called for an illegal or bad return code from DIR.
- BMSG (CZAWM) -- Sends messages to operator.
- BULKIO Preprocessor (CZABD) -- Validates the parameters on the DATASET card.
- CLOSE (CZALB) -- Closes the DCB for a non-unique data set.
- DDEF (CZAEA5) -- Creates a JFCB associated with the DCB for the data set.
- DIR (CZCJDA) -- Deletes the Interrupt Control Block (ICB).
- Input Closeout (CZAWV) -- Erases a partially created data set.
- Input Service (CZAWZ1) -- Reads cards.
- Master Service (CZAWW) -- Is called if an ABEND occurs in a read cards operation.
- OPEN (CZCLA0) -- Opens DCB for output.
- RELEASE (CZAFJ3) -- Releases the JFCB for a non-unique data set.

Exits: BULKIO Input Starter normally exits to the caller. If an invalid return code is received from DIR, BULKIO will be terminated by a call to CZAWA. The following messages may be issued before return:

- MSG1 DSN=UID*****.+1 to 35 CHAR DSNAME
- MSG2 SDA=XXXX NOT ASSIGNABLE. JOB TERMINATED
- MSG4 DIR RC=XXXX SDA=XXXX. READ CARDS ABORTED.
- MSG5 SDA NOT ASSIGNABLE. JOB TERMINATED
- MSG6 UID=XXXXXXXX NOT ALLOWED BATCH WORK
- MSG7 DDEF RC=XXXX MSGID=XXXX
- MSG8 CZABD RC=XXXX

MSG9 UID=XXXXXXXX INVALID. RCR XXXXX ERROR

MSG10 NO USERID

MSG14 DSNAME NOT UNIQUE.DSN=UID****.(+1 to 35 CHAR DSNAME)

Operation: Input Start performs the initialization requirements for a read-cards job. It has both synchronous and asynchronous entry points. An entry at this routine's asynchronous entry point is a signal that a batch task is being initiated. The first function performed is to establish the address of the S-entry for this device and store it in the BULKCOMM table for use by CZAWZ (Input Service). Following this, a DIR is performed against the device. This disables all interrupts from this device until a SIR is performed. If DIR returns a code other than 0 or C, a diagnostic is sent to the operator and the BULKIO ABEND module is invoked to remove the BULKIO task from the system. Next, the number of active S-entries count is incremented by one. Flags used by CZAWZ are initialized and the Read Cards module (CZAWZ) is called. Upon return, CZAWX returns to the Task Monitor.

For an RJE entry, the routine validates the S-entry pair for the device, and the interruption; and invalid interruption causes the deletion of the S-entry, and a message to the operator. The device is marked available. If, at this point, it is determined that the interruption shows a loop caused by a return code of X'4' from the GET macro instruction, output to the device is halted, the count of active S-entries is incremented, and return is made to the caller. If the loop did not occur, MSAM posting is called, and return is made to the caller.

When CZAWZ encounters a LOGON card, it calls the synchronous entry point of CZAWX for task initialization. CZAWX1 will determine if a SYSIN, DATASET, or EXPRESS is requested and perform the following:

For SYSIN:

1. Validate the userid via the 'RCR OPEN' macro;
2. Set start byte to 1, stop byte to 80, EBCDIC flag on, END option flag on;
3. Generate a unique dsname=userid**. SYSIxxxx where xxxx is a number from 0000 to 9999;
4. Define a VSAM data set and open its associated DCB.

For EXPRESS BATCH: Determine if the VAM DCB is open. If it is:

1. Validate the user via RCR OPEN;
2. Return to the caller.

If it is not:

1. Set the start byte to 1, stop byte to 80, EBCDIC flag on, END flag on;
2. Generate a unique DSNAME=SYSOPER0.SYSIXXXX, where XXXX is a number from 0000 to 9999;
3. DDEF for a VSAM data set card and open its associated DCB.

For DATASET:

1. Validate the userid (via RCR OPEN);
2. Call CZABD (BULKIO Preprocessor) to validate the parameters on the DATASET card;
3. If the replace option is specified, check to see if there is an old version of the data set, and if so, erase it;
4. Define VSAM or VISAM data sets, depending upon the option specified by user, and open the DCB for the data set.

After the data set has been opened, the JFCB associated with this data set is checked to determine whether the disposition is old or new. When the disposition is new, processing continues. If the disposition is old, the following steps are performed:

For SYSIN and EXPRESS:

1. Close the DCB and release the JFCB for the data set;
2. Increment the DSNAME by 1;
3. Issue DDEF and OPEN again;
4. If after 5 such attempts the disposition remains old, a diagnostic is issued and the job is flushed through the card reader.

For DATASET:

1. Close the DCB and release the JFCB for the data set;
2. Issue a diagnostic;
3. Move the input to the next LOGON/DATASET.

System Control Block Usage:
Active User List (CHAAUL)

BULKCOMM (CHABCT)

Communications Area (CHACOM)

Data Control Block (CHADCB)

DCB Header (CHADHD)

Interrupt Control Block (CHAICB)

Interrupt Storage Area (CHAISA)

S-entry (CHASET)

Shared Dataset Member (CHASDM)

Task Common (CHATCM)

Task Definition Table (CHATDT)

Task Status Index (CHATSI)

User Table (CHAUSE)

BULKIO Input Service (CZAWZ)

This routine is called by Master Service (CZAWW) or Input Start (CZAWX) to build a SYSIN for execution in the background, or to build a VSAM or VISAM data set. (See Chart KK.)

Entry Points:

CZAWZ1 -- Main entry point; from Master Service or Input Start after receiving asynchronous interrupt from card reader.

CZAWZ2 -- SYNAD for VISAM PUT to issue an error message and to terminate the job in progress.

CZAWZ3 -- Disconnect RJE line.

Modules Called:

BULKIO Message File (CZAWM1), via BMSG macro -- Writes messages to the operator.

BULKIO Recovery (CZAWA) -- Abnormally terminates BULKIO task via ABEND; called if an invalid S-entry is detected or if a nonzero return code is received from SIR.

EXHIBIT Processor (CZAYF1) -- For "...STATUS" control card.

FINISH (CZCMH) -- Zeros the MSAM device buffers.

FINISH (MSAM) -- Results follow:

<u>Return Code</u>	<u>Significance</u>	<u>Action</u>
0	successful	Processing continues.
4	incomplete	Test for pending interrupt; if one exists, return to caller; otherwise, link to CZAJIA to service interrupt; then return.
8	error	If error is permanent, abort job in progress and delete job; otherwise reissue FINISH, flag, and return.

FTN Convert (CZAWC1) -- Converts line data set to FORTRAN source.

GET (MSAM) -- Results follow:

<u>Return Code</u>	<u>Significance</u>	<u>Action</u>
0	successful	Processing continues.
4	incomplete	Test for pending interrupt; when none, return to caller; otherwise, link to CZAJIA to service interrupt.
8	error	If error is permanent write message, terminate job and return; if not permanent, check error options and terminate job, skip error and continue, or accept error as specified.
C	end of file	Issue FINISH macro instruction and SIR device;
10	validity check	abort job in progress; return.
14	hang up	Issue MSAM Finish, Clean up, reenable line.

Input Closeout (CZAWV) -- Closes output data set DCB for SYSIN VSEND to Batch Monitor to start background execution; deletes a device.

Input Start (CZAWX) -- Validates LOGON or DATASET card; builds an open output data set DCB.

SIR -- zero -return code, successful, continue processing; nonzero, abnormally terminate BULKIO task.

SIR (CZCJS) -- Sets an interrupt control block equal to the asynchronous entry point address in CZAWX.

VAM PUT (CZCOS) -- Writes VSAM records.

VAM PUT (CZCPA) -- Writes VISAM records.

Exits: The routine normally returns to the caller. When an error occurs, the job is aborted, a message is sent to the operator, and return is made to the caller.

Operation: On entry to CZAWZ1, the input S-entry is validated, and an input record is read in by MSAM GET. Then, if this is an RJE task, the RJE station ID is validated via a search through the validation data set which is a member of the Acknowledgment Data Set (ACK). Assuming that no errors occur, there are two main paths in the module. The first path initializes the program. It is entered when a LOGON or DATASET card is recognized. This path sets switches and initializes counters. A call is made to Input Start (CZAWX) to: validate the user ID and batch privileges; check the parameters on a DATASET card; construct and open the output DCB; and, catalog the data set.

The second path builds the output data set using VSAM or VISAM PUTS. It is entered after every MSAM GET, unless the input record is a LOGON or DATASET card. In this path the appropriate bytes are put into the output record; for a SYSIN, bytes 1 through 80 are put out; for a data set, the bytes are specified in the DATASET card. When a LOGOFF or %ENDDDS card is encountered, a call is made to Input Closeout (CZAWV) to close the output data set and, for a SYSIN VSEND, to the Batch Monitor to queue the SYSIN for execution in the background.

For Remote Job Entry processing, CZAWZ recognizes and processes six control cards: RJSTART, CONTINUE, STATUS, OUTPUT, DELETE, and RJEND.

RJSTART -- CZAWZ validates the station identification (id) to ensure that it is a valid RJE station id, not currently active on the system.

CONTINUE -- If there is an output job in progress to the requesting station id, CZAWZ sets S-entry flags to indicate "continue, cancel, or restart" operations.

STATUS -- CZAWZ calls the EXHIBIT processor (CZAYF1) to search the Batch Work Queue for the status of all jobs associated with the requesting RJE station id. The EXHIBIT processor calls CZAWN to write

the status information into the Acknowledgement Data Set.

OUTPUT -- CZAWZ marks the RJE station's output S-entry to indicate the order of output for print jobs.

DELETE -- CZAWZ sends a VSEND to the Batch Monitor to cancel the specified batch job.

RJEND -- CZAWZ calls Input Closeout (CZAWV) to disconnect the RJE station id.

SYSIN processing can be extended to provide for the packing of one or more SYSINS into one data set. This extension is signaled by the EXPRESS and END cards. The EXPRESS and END cards bracket the SYSINS to be packed into one data set. The Express Batch SYSIN data set permits an installation to define a time limit for on-line execution for each SYSIN within the Express Batch data set.

At entry CZAWZ3, the S-entry pair is validated at WZCHECK, and the RJE line is disconnected by WZRJEDIS. WZRJEDIS is a subroutine which cleans up the input and output S-entries for the device, and sends a disconnect message to the operator.

System Control Block Usage:
BULKCOMM (CHABCT)

Data Control Block (CHADCB)

Data Event Block (CHADEB)

Data Event Control Block (CHADEC)

Interrupt Control Block (CHAICB)

S-entry (CHASET)

Symbolic Device Allocation Table (CHASDA)

FORTRAN Data Set Conversion (CZAWC)

This routine converts line data sets to FORTRAN source data sets. (See Chart KL.)

Entry Point: CZAWC1 -- Main entry.

Modules Called: None.

Exits: Control is returned to the caller; no return codes or messages result.

Operation: On entry to CZAWC1, register 1 must contain a pointer to a two-word parameter list: word 1 contains a pointer to the input buffer; word 2 contains a pointer to the output buffer. (If either pointer is zero, or if the input buffer is empty, control returns with no further processing.) CZAWC then formats the record in the output buffer so that it will be

acceptable to the FORTRAN compiler. Records are processed according to the following decision table; processing depends upon the contents of the two buffers.

input buffer	zeros	✓	✓				
	comment		✓	✓			
	START card			✓	✓		
output buffer	neither comment nor START					✓	✓
	comment	✓	✓		✓	✓	
	non-comment	✓	✓		✓		
action taken		A	B	A	B	A	B
		A	B	A	B	A	C

- The record is output as is.
- The trailing blanks are stripped off from column 73; the resulting record length is adjusted.
- A hyphen is placed in column 73; the record length is reduced by 7.

System Control Block Usage: None.

BULKIO Input Closeout Routine (CZAWV)

This routine performs a cleanup operation for Card Read and BULKIO Initialization. (See Chart LL.)

Entry Points: CZAWV1 -- Normal entry.

Modules Called:

BMSG (CZAWM) -- Sends messages to operator.

BULKIO ABEND (CZAWA1) -- Handles ABEND conditions.

CLOSE (CZCLBC) -- Closes a DCB for VAM or MSAM.

DIR (CZAJDA) -- Deletes interrupt request.

ERASE (CZAEJ3) -- Erases partially created data sets or SYSINS.

RELEASE (CZAFJ3) -- Releases the JFCB associated with the DCB.

VSEND -- Sends task initiation request to the Batch Monitor EXECUTE subprocessor and CZABA2.

Exits: Control is returned to the caller. When an error occurs, a diagnostic message is issued.

Operation: The routine determines whether to close and release the card reader. If the card reader is to be deleted, its associated S-entry is marked deleted, its DCB is closed, and its JFCB is released.

Next, the bypass flag is checked. When it is on, it indicates that there are no outstanding read card jobs; the flag is reset and a return is made.

When the bypass flag is off, the throwaway flag is checked: if it is on, the VAM data set currently being processed is erased, flags are reset, and control is returned to the caller; if it is off, the Express Mode flag is checked. If this flag is on, one more check is made to determine whether to close the Express data set and VSEND for task initiation, or to charge this user (via RCR RATION) and return control. If the job is non-Express, Input Closeout performs the following:

1. Closes the DCB and releases the JFCB for the VAM data set.
2. Uses the RCR macro instructions for charge purposes.
3. Issues a VSEND to the batch monitor (CZABAB) for task initiation if the job is a SYSIN, and sends a confirmation message to the operator.
4. Sends a 'job complete' message to the operator, for data sets.

For any illegal return codes, a diagnostic message is sent to the operator and return is made to the caller.

System Control Block Usage:

Active User List (CHAAUL)

Batch Work Queue (CHABWQ)

BULKCOMM (CHABCT)

Data Control Block (CHADCB)

Interrupt Storage Area (CHAISA)

S-entry (CHASET)

Task Common (CHATCM)

Task Definition Table (CHATDT)

Task Status Index (CHATSJ)

User Table (CHAUSE)

BULKIO OUTPUT Service (CZAWY)

This routine transfers a user's VAM data set to a printer or punch by using MSAM. (See Chart MM.)

Entry Points:

CZAWY1 -- Main entry, from Master Service (CZAWW).

CZAWY2 -- VAM SYNAD.

CZAWY3 -- VAM EODAD.

CZAWY4 -- Output cleanup by ABEND recovery (CZAWA).

CZAWY5 -- Message Control Block area, used by CZAWU to abort a job.

Modules Called:

BMSG (CZAWM1) -- Writes messages to operator.

CLOSE (CZCLBC) -- Closes user's VAM data set.

FINISH (CZCMH1) -- Completes MSAM output.

Initialization (CZAWS3) -- Adds a deleted device.

Input Service (CZAWZ3) -- Re-enables RJE line.

Master Service (CZAWW2) -- ABEND entry point in Master Service.

MSAM PUT -- Writes record on printer or punch.

RELEASE (CZAFJ3) -- Releases user's VAM data set or MSAM device being deleted.

VAM GET -- Gets user record to be output.

VSEND -- Sends 'job complete' message to Batch Monitor.

Exits: The routine normally returns to Master Service. If an error occurred, return is to ABEND entry point of Master Service to cleanup after VAM ABEND on an output job.

Operation: Output Service transfers a user's VAM data set, using MSAM, to a printer or punch. It converts the records, if required, and either validates the user's edit control characters or supplies its own. Headings, page numbers, and non-standard number of lines per page are supplied if the user desires.

When CZAWY is first called by CZAWW (Master Service) to begin a job, the input data set will have already been defined and opened by CZAWU (Attend Alert), and the output data set by CZAWS (Initialization). The VAM RECFM and DSORG are validated and S-entry fields and flags are initialized from the DCB. A GET is performed on the first user record; if he specified the EDIT option, the Control Character (CC) of this

record determines whether American National Standard for Information Interchange, ANSI X3.4-1968, hereinafter referred to as ASCII or machine will be used; if he specified the heading option, the first record will be saved to head each page. All MSAM records are variable length with either ASCII or machine edit (machine, if user did not request his own edit). A standard set of CC's for headers, trailers and default CC's saved. Finally the job heading lines or cards are put out.

The main work of CZAWY is done in the following loop: A VAM GET is done on the user's data set to locate a record; an MSAM PUT is done on the device data set to locate space for the output record; the VAM record is moved to the MSAM record with whatever converting is necessary. New pages, headings, and page-numbering are done when required. This continues until MSAM indicates an I/O WAIT is required; at this point CZAWY returns to CZAWW where some other S-entry may be processed. At some future time CZAWW redispaches CZAWY for the waiting S-entry and the PUT is reissued. If MSAM can complete the PUT, processing continues in the above loop; if not, control is returned to CZAWW.

Converting the record consists of selecting that part determined by the start byte-end byte option, and prefixing it with a variable length field and control character. The control character may have been supplied by the user or it may be supplied by CZAWY, if no edit was specified or if the user's CC was invalid. Line data set records or records produced by the DATA command are rearranged because of the keyboard/card-reader character, which is itself translated to a blank (terminal) or "C" (card reader).

When EODAD is reached on the user's data set, the data set is closed and released. Trailer records are put out and the device data set is processed by FINISH. FINISH, like PUT, may produce a return to CZAWW because of an MSAM I/O wait. When FINISH is done, the device is released if a delete had been requested. RCR RATION and CLOSE are issued to charge the user for his output; a VSEND is done to inform the batch monitor that this job is complete, and an EOJ WTO is sent to the operator. Then a final return is done to CZAWW until the next output job is initiated.

If CZAWY or CZAWU has a VAM ABEND on an output job, CZAWA (Recovery) calls CZAWY4 to clean up. If a job is canceled, Batch Monitor sets on SETTOP and SETEOD, which forces CZAWY to go to its EODAD routine on the next GET. If CZAWY or CZAWU detects a permanent I/O error, SETPER and SETDEL will be set causing the job to be aborted and the device deleted. If the operator does

an ASNBD delete on a device with an output job in progress, CZAWS (initialization) will turn on SETDEL forcing device deletion after the job in progress is completed. An ASNBD delete, on an input job in progress, causes the job to be aborted, and the device to be deleted.

The following errors cause the job to be aborted with no user output: VAM DSORG not VS or VI; VAM RECFM not F or V; fixed length records with length less than start byte; edit option specified, but first user CC is neither ASCII nor machine. The following errors cause the job to be aborted with an indeterminate amount of output: bad return code from a called routine; VAM SYNAD; MSAM I/O error, attempted ABEND by a called routine or VAM EODAD on user's data set (implies empty user data set). The table in Figure 6 shows the ABORT codes for these errors.

System Control Block Usage:
Active User Table Entry (CHAAUL)

Assign BULKIO Device DSECT (CHAABD)

BULKCOMM (CHABCT)

Data Control Block (CHADCB)

Data Event Block (CHADEB)

Data Event Control Block (CHADEC)

DCB Header (CHADHD)

Interrupt Storage Area (CHAISA)

Message Control Block (CHAMCB)

S-entry (CHASET)

Code	Meaning
01	VAM DSORG not VS or VI.
02	VAM RECFM not F or V.
04	Fixed length records with length less than start byte.
08	EDIT option specified, but first user CC is not ASA or machine.
10	Bad return code from a called routine.
20	VAM SYNAD.
40	MSAM I/O error.
80	Attempted ABEND by called routine, or VAM EODAD on user's data set.

Figure 6. CZAWY ABORT Codes

Shared Data Set Member (CHASDM)

Task Common (CHATCM)

Task Data Definition Table (CHATDT)

Task Status Index (CHATSII)

User Table (CHAUSE)

BULKIO Message Handler Routine (CZAWM)

This routine constructs a message from skeleton message pieces located in CZAWM and the caller's CSECT, and fills in variables designated by the caller. The resulting message is sent to the operator to record the BULKIO task status. (See Chart NN.)

Entry Points: CZAWM1 -- Call by BMSG macro instruction.

Modules Called:

CZAWN1 -- Writes to RJE terminal.

XWTO (CZABQ1) -- Writes to a WTO or WTL message.

CZAWN1 -- Writes to RJE terminal.

Exits: The routine normally exits to the caller. If the caller designated an invalid message piece (that is, bad message identifier), an error message will be put out giving the address where BMSG was last invoked; then control returns to the caller.

Operation: On entry, at CZAWM1, the message and message length are validated and moved to the message area. A carriage return character is inserted following the last text character. CZAWM then flags the message type (WTO, WTL, RJE) and calls CZABQ1 to send the message. If the message was sent to an RJE device, CZAWN1 is called to write to that station. Control returns to the caller.

CZAWM constructs its messages from message pieces which are found in the two csects. Common message pieces are in the first csect, following the code, and less common pieces are in the second csect. The routine is entered with pointers to specific message pieces which are assembled into a complete message.

System Control Block Usage:

BULKCOMM (CHABCT)

BULKIO Message DSECT (CHABWM)

BULKIO ACK Data Set Handler (CZAWN)

This routine handles messages from the RJE station to the Acknowledgment Data Set

(ACK), and messages from the ACK to the RJE station. (See Chart OA.)

Entry Points:

CZAWN1 -- Entered from CZAWM to write messages to the ACK.

CZAWN2 -- Entered from CZAWW to write messages from an ACK to a specified RJE station.

CZAWN3 -- EODAD routine when all messages have been written from the ACK.

CZAWN4 -- SYNAD

Modules Called:

Write (CZCPE1) -- Writes to ACK.

Stow (CZCOK1) -- Puts new ACK member in POD.

Input Closeout (CZAWV1) -- Deletes S-entries.

MSAM Finish (CZCMH1) -- Cleans up MSAM buffers.

Input Service (CZAWZ1) -- Re-enables RJE line.

Find (CZCOJ1) -- Finds an ACK member.

XWTO (CZABQ1) -- Sends messages to the operator.

Exits: Control is returned to the calling routine. When an error occurs, a diagnostic message is issued.

Operation: On entry to CZAWN1, from CZAWM, this routine gets the Acknowledgment Data Set (ACK) DCB, ensures that the message is to a valid RJE station ID, and that there have been no errors in processing the ACK. (Should any of these errors occur, control is immediately returned to the caller.) Next, FIND is called to locate the ACK member; SETL positions past the last record of the member, or to the beginning of a new member; WRITE writes the message to the ACK. Had the ACK member been new, the routine would build a DCB before the Write operation; then, following the write, it would stow the member in the POD. If there is an input S-entry, it is chained to its corresponding output S-entry. Control returns to the caller.

At CZAWN2 the active S-entry count is incremented; PUT issues the message to the station ID. FIND is called to locate the ACK member. Then, GET locates further output records which are also written, via PUT, to the RJE station ID. When all messages have been written, control passes to CZAWN3 for EODAD processing. If there are no messages to be output, MSAM Finish

cleans up the buffer; the active S-entry count is reduced (twice for an S-entry pair); control returns to the caller (CZAWW).

At CZAWN3, to process EODAD conditions when GET has output all messages, MSAM Finish cleans up the buffer. Then, the ACK member is found and deleted; the count of active S-entries is reduced. If the station ID is the symbolic device address, the line is re-enabled by CZAWZ3. Otherwise, the S-entry pair is deleted. In any case, control returns to the caller.

At CZAWN4, the SYNAD routine, an error message is written to the station ID; buffers are emptied; S-entries are deleted; and control returns to the caller.

System Control Block Usage:
BULKCOMM Table (CHABCT)

BULKIO Message DSECT (CHABWM)

Data Control Block (CHADCB)

S-entry (CHASET)

BULKIO ABEND Recovery (CZAWA)

This module determines the recovery procedure to be used when an ABEND occurs in the BULKIO task. (See Chart 00.)

Entry Points: CZAWA1 -- Normal entry.

If an ABEND occurs during CZAWA processing, processing resumes at the CZAWA instruction that follows the point of ABEND. This establishes a recursive entry procedure that changes the entry point with further processing.

Modules Called:
Message Handler (CZAWM1) -- Prints message to operator.

Internal Subroutines:
WABAOUT -- Prints message giving reason for ABEND and traces information.

WABLOCK -- Locks BULKCOMM table.

Exits: Control is passed to a recovery procedure; there is no return to this routine.

Operation: The routine tests two fields in the BULKCOMM table (CHABCT): BCTBIO and BCTCALL. BCTBIO indicates the action being performed, and BCTCALL indicates the BULKIO module performing the action at the time of the ABEND. The first two characters of BCTCALL determine the type of recovery:

V-, VAM recovery

M-, MSAM recovery

other, ABEND - no recovery possible.
The BULK I/O task is removed from the system.

Control is passed to one of the recovery procedures listed below; there is no return to this routine.

1. WABV - VAM input recovery
2. WABE - MSAM input recovery
3. WAVO - VAM output recovery
4. WAMO - MSAM output recovery
5. WAINIT - initialization recovery
6. WABF - RJE recovery from MSAM.
7. WABN - RJE recovery from ABEND while processing the ACK data set.
8. WABVEX - RJE recovery from ABEND while processing ..STATUS control card.

Current recovery procedures are employed to recover from a VAM ABEND on an RJE device: WABV for input; WAVO for output.

The routine also keeps a count of ABENDS since startup in BCTAKT. If the count exceeds a limit, the BULKIO task is abended. The recovery procedures which are given control may cause an ABEND. Control is passed to a recursive address in the recovery routine, if the count of recursive ABENDS does not exceed the specified limit.

MSAM Input Recovery Procedure (WABE)

This routine deletes a card reader and, for a non-RJE device, reassigns it to the BULKIO task. The VAM data set, which was built during the time of the ABEND, is erased.

Entry Points: WABE -- Normal entry.

Modules Called:
CLOSE (CZCLBC) -- Closes device or data set DCB.

ERASE (CZAEJ3) -- Erases VAM data set.

Initialization (CZAWS3) -- Assigns device.

Input Start (CZAWX3) -- Starts BULKIO task cycling after recovery is complete.

RELEASE (CZAFJ3) -- Releases device JFCB.

Internal Subroutines:
WABDEL C -- Returns external storage used by VAM routines.

WABEDEL -- Marks device S-entry deleted, and reduces count of S-entries with jobs, and updates accounting information.

WABRD -- Sets the open flag in DCB off, and sets count of open DCBs in JFCB for device to zero.

WACAT -- Removes lock on catalog.

WARJFB -- Removes JFCB from TDT chain.

WARELEAS -- Marks device available in SDAT, and reduces count of S-entries with jobs, and updates accounting information.

VAM Input Recovery Procedure (WABV)

This routine removes the VAM data set being built at the time of the ABEND. Job dependent flags are reset.

Entry Points: WABV -- Normal entry.

Modules Called: ERASE (CZAEJ3) -- Erases VAM data set.

Internal Subroutines:

WABRED -- Returns external storage used by VAM routines.

WABVI -- Resets VAM interlocks.

WACAT -- Removes lock on catalog.

MSAM Output Recovery Procedure (WAMO)

This routine restores the BULKIO task to a workable state after an ABEND has occurred during an MSAM operation. The routine performs tests to determine what action must be taken and then performs the clean-up operation.

Entry Points: WAMO -- Normal entry.

Modules Called:

BMSG (CZAWM) -- Sends diagnostics to operator.

Close (CZCLB) -- Closes DCB for MSAM and VAM.

Initialization (CZAWS3) -- Is called for dynamic ASNBD of device just deleted.

Master Service (CZAWW2) -- Returns control to BULKIO after ABEND recovery.

PRTHX -- Converts binary to printable hexadecimal.

RELEASE (CZAFJ) -- Releases JFCB for MSAM and VAM.

REP -- Establishes recursiveness.

VSEND -- Notifies Batch Monitor of job cancellation.

Internal Subroutines:

WABORT -- Terminates BULKIO.

WABRO -- Resets open flags in JFCB and DCB.

WABVI -- Releases VAM interlocks.

WAEXIST -- Searches for a JFCB.

WARCR -- Is called for charge purposes.

WARELEAS -- Marks device available in SDAT chain.

WARJFB -- Removes JFCB from TDT chain.

Operation: The device DCB is checked. If it is open, recursiveness is established and the DCB is closed. If the 'close' abnormally terminates, the recursive address manually closes the DCB and its associated JFCB.

If the DCB is not open, the 'close' will be skipped and only a release on the device is attempted. No release is attempted for an RJE device.

Before RELEASE is called, recursiveness is established. If RELEASE abnormally terminates, the recursive address calls a subroutine (WABRO), which searches the TDT chain and removes the device JFCB from the chain. Also, the WARELEAS subroutine is called to mark the device available in the SDAT. Both subroutines are also called if RELEASE returns an error code.

Next, the batch monitor is notified, via VSEND, that the job in progress has been aborted.

Following this, a search is made of the TDT chain for the VAM JFCB associated with the output job. If none is found the RCR subroutine is called for charge purposes; the S-entry is deleted, counts of S-entries are adjusted; for non-RJE devices, CZAWS (BULKIO initialization) is called for dynamic re-add of the device; the BULKCOMM is unlocked and control is given to the CZAWW2 (Master Service).

If a VAM JFCB is found, its associated DCB is checked to see if it is open. If it is, recursiveness is established and CLOSE is called. If CLOSE abnormally terminates, the recursive address locates the JFCB for the data set. Once found, a test is made to see if the data set is shared. If it is, the interlock release subroutine (WABVI) is called. If not, the open flags in the JFCB and DCB are reset and control is returned to main line logic.

If the DCB is not open, CLOSE is not called and only a release is attempted. If RELEASE abnormally terminates or returns an error code, the subroutine WARJFB is called to remove the JFCB from the TDT chain. The routine then performs the same functions mentioned above when no VAM JFCB is found.

VAM Output Recovery Procedure (WAVO)

This routine restores the BULKIO task to a workable state after an ABEND has occurred during a VAM operation.

Entry Points: WAVO -- Normal entry.

Modules Called:

BMSG (CZAWM) -- Sends message to operator.

CLOSE (CZCLB) -- Closes DCB associated with VAM data set.

Output Service SYNAD (CZAWY4) -- Ends job in which ABEND occurred.

PRTHX -- Converts binary to printable hexadecimal.

RELEASE (CZAFJ3) -- Releases JFCB associated with VAM data set.

Internal Subroutines:

WABRD -- Resets open flags in JFCB and associated DCB

WABVI -- Releases VAM interlocks.

WACAT -- Unlocks catalog interlocks.

WAEXIST -- Searches TDT chain for a JFCB.

WARJFB -- Removes JFCB from TDT chain.

Operation: The routine checks for catalog interlocks. To do this, a call is made to WACAT, a subroutine of the ABEND Recovery module.

The next test checks if the ABEND occurred during DDEF. If it did, flags are reset, locks unlocked and control given to CZAWY4 (the SYNAD of Output Service). If it did not, the TDT chain is searched for the JFCB for the data set being processed. If none is found, cleanup is the same as if the ABEND occurred in DDEF.

If ABEND did not occur in DDEF and a JFCB is found, the DCB for the data set is tested to determine if it is open. If the DCB is closed, recursiveness is established and RELEASE is called to release the JFCB for the data set. If RELEASE abnormally terminates, the recursive address uses the subroutine WARJFB to remove the JFCB from the TDT chain. WARJFB is also used if RELEASE returns an error code. Flags and locks are then reset and control is given to CZAWY4.

If the DCB is open, CLOSE is called. If CLOSE abnormally terminates, the recursive address locates the JFCB associated with the data set and performs the following: if the data set is shared, WABVI, the interlock release subroutine, is called to release VAM interlocks, and the open flags in the DCB and JFCB are reset. If the data set is not shared, only the open flags are reset. Processing is then the same as that in the above paragraph involving the RELEASE logic when the DCB was not open.

Initialization Recovery Procedure (WAINIT)

This routine restores the BULKIO task to a workable state after an ABEND has occurred during BULKIO initialization.

Entry Points: WAINIT -- Normal entry.

Modules Called:

BMSG (CZAWM) -- Writes message to operator.

CLOSE (CZCLB) -- Closes device DCB.

PRTHX -- Converts binary to EBCDIC

RELEASE (CZAFJ3) -- Releases device JFCB.

Internal Subroutines:

WABORT -- Terminates BULKIO.

WABRO -- Resets open flags in JFCB and DCB.

WARJFB -- Removes JFCB from TDT chain.

WARELEAS -- Marks device available in SDAT table.

Operation: The first check will determine whether the ABEND occurred during DDEF. If it did, a diagnostic is issued and tests are made to determine the point to which control is to be returned.

If the ABEND did not occur during DDEF, a check will be made to determine if the SETUR DCB is open. If so, an error message is issued and BULKIO is aborted. If the SETUR DCB is not open, a test is made to see if the device DCB is. If it is not, only a RELEASE is attempted. If it is, both CLOSE and RELEASE are invoked. If either CLOSE or RELEASE abnormally terminates or returns an error code, subroutines are used to accomplish the desired ends. For a CLOSE ABEND, the subroutine WABRO, which resets the open flags in the DCB and JFCB, is used. For a RELEASE ABEND or error return code, the subroutines WARJFB and WARELEAS are used to remove the device JFCB from the TDT chain, then mark the device available in the SDAT. Tests are then performed to determine the point to which control is to be returned.

SECTION 5: COMMAND ROUTINES

This section describes those command routines used only by an operator (class A), and those command routines within the BULKIO preprocessor. Each routine performs those functions requested by a specific command. Figure 7 contains the names and functions of these commands.

For those commands issued by the operator, the Command Analyzer and Executor (CA&E) links to the appropriate command routine when their corresponding command names appear in SYSIN. Typically, the CA&E links to the command routine which employs SCAN routines to isolate and validate command parameters. The command routine then performs the requested functions and returns control to the CA&E.

The PUNCH, PRINT, RT, and WT commands are handled by the BULKIO preprocessor. When the CA&E encounters one of these commands, it passes control to the preprocessor which validates the parameters and requests batch initiation from the batch monitor. The batch monitor assigns a batch sequence number and initiates a batch operation independent of the requesting task.

If the data to be printed or punched resides on system public VAM storage, the operation is assigned to the BULKIO task. Data sets and nonconversational SYSIN command streams that are read in through a high speed reader are handled by the BULKIO task.

If the data to be printed or punched resides on a private VAM or SAM volume, the operation is assigned to CZABG or CZABH, respectively, in an independently created nonconversational task. Since RT and WT involve private SAM volumes, they are also assigned to independently created nonconversational tasks (CZABF and CZABI, respectively).

ASNBD Command Routine (CZABK)

The Assign BULKIO Device (ASNBD) command routine obtains card readers, card punches, printers, and RJE devices for the BULKIO task. It requests the release of these devices when the operator does not want them assigned to the BULKIO task. (See Chart PP.)

Entry Points:

CZABK1 -- Command entry.

Routine Name	Symbolic Name	Purpose
ASNBD	CZABK	To obtain or delete card readers, punches, printers, and RJE devices for the BULKIO task.
FORCE	CZACG	To terminate a specific conversational task.
HOLD/DROP	CZCM0	To discontinue or to reestablish the availability of a device for TSS/360 usage.
LABEL	CZABX	To label or unlabel a tape.
MSG/BCST	CZACF	To write a message to a specific conversational user or to all conversational users.
PRINT	CZABD4	To print a data set.
PUNCH	CZABD6	To punch a data set onto cards.
RT	CZABD2	To read a tape data set into storage and to catalog it.
SHUTDOWN	CZACN	To terminate all tasks prior to physical shutdown of the system.
WT	CZABD8	To write a data set onto tape.

Figure 7. Command Routines

CZABK2 -- MOHR entry at startup.

Modules Called:

ATPOL macro -- Checks for pending attention interrupt.

PRMPT (CZATJ1) -- Issues messages to the operator and obtains responses where applicable.

SCRISI macro -- Creates the BULKIO task (TASKID=2).

VSEND macro -- Sends messages to the BULKIO task specifying devices to be added or deleted.

Exits: ASNBD always exits to the caller.

Operation: Entry occurs at CZABK1 when ASNBD is entered by the operator to change unit record assignment to the BULKIO task. A diagnostic message occurs if one of the following invalid requests is used:

1. Not valid SDAT entry or entry permanently locked.
2. Not 2540 Card Reader, 2540 Card Punch, 1403 Printer or 2780 RJE device.
3. Not available for an add request.
4. Duplicate requests (add and/or delete) for same device.

The ASNBD is ignored if there is too much prompting, either SCRISI or VSEND is unsuccessful, an attention interrupt occurs, or the operand is defaulted. If the ASNBD command is issued by a task other than the operator task (TID 001), the command is canceled. A correct operand is accepted, unless more than 16 devices are requested. Then only the first 16 devices are accepted.

Entry occurs at CZABK2 when MOHR (CZACB) calls during startup for device assignment. The operator may request no devices, specific device, or all unit record devices that are available. In the last two cases, an acceptable request causes a zero return code to MOHR. MOHR does the SCRISI of BULKIO and a VSEND specifying the devices to be used. An unsuccessful SCRISI or VSEND will cause MOHR to call CZABK2 again for cleanup (make devices in SDAT available again) and to present a diagnostic message.

At either entry, if Bulk I/O initialization is already in progress, a message is issued and the command is canceled.

MOHR receives a return code of 4 if no devices are to be assigned to BULKIO. Devices can only be assigned, and not deleted, at startup.

Error Conditions: Invalid operands cause diagnostic messages to the operator, as described in "OPERATOR'S GUIDE" and listed below.

CZABK001 ENTER BULKIO DEVICES.

CZABK002 INVALID REQUEST (gmmn NOT \$01). REENTER ENTIRE OPERAND.

Inserts: 1) UNIT REC 5) UNLOCKED
2) AVAIL 6) UNIQUE
3) EXISTENT 7) ADD DEV
4) VAL CHAR 8) IN AHD

CZABK003 \$01. DEVICE ASSIGNMENT UNCHANGED FOR ASNBD.\$02.

Inserts \$01: 1) MOHR VSEND NOT ACPT
2) SCRISI NOT TASK 2
3) TOO MUCH PROMPTING
4) ATTENTION INTERRUPT
5) DEFAULTED DEV LIST
6) VSEND RC0, NO TASK 2
7) VSEND RC4, NOT TAKEN
8) VSEND RC ILLEGAL

Inserts \$02: 1) 'NO' ASSUMED DURING STARTUP
2) (ALL BLANKS IF NOT STARTUP)

CZABK004 \$01 \$02 ACCEPTED.

Inserts \$01: BLANK OR MAX IS 16 DEVICES
\$02: OPERAND IS REPEATED

Delete requests for devices are accepted without regard to their prior assignment to BULKIO. If an unassigned device is deleted, the BULKIO issues a diagnostic message to the operator. A unit record device add request is accepted if the device is available, and the equivalent of a 'secure' operation is performed to make it unavailable to any task by BULKIO.

System Control Block Usage:
Assign BULKIO Device (CHAABD)

Available Device Table (CHAHED)

BULKCOMM (CHABCT)

Interrupt Storage Area (CHAISA)

Message Control Block (CHAMCB)

Subqueue Entry (CHAAVE)

Subqueue Header (CHAAHD)

Symbolic Device Allocation Table (CHASDA)

Task Common (CHATCM)

FORCE Command Routine (CZACG)

FORCE terminates a conversational task, as directed by the system operator. FORCE operates in conversational mode only. (See Chart QQ.)

Entry Points:

CZACG1 -- Normal entry.

CZACG2 -- Flag entry.

Modules Called:

ALFNUM (CZAAC3) -- Validates syntax of userid.

MSGWR (CZAAD2) -- Issues messages to system operator.

NEXTPAR (CZAAC1) -- Locates command operand.

VSEND -- Sends messages to user's conversational task.

Exits: FORCE normally returns to the calling routine, via the RETURN macro instruction. If a system error occurs, a SYSER is issued and FORCE exits to the calling routine.

Operation: The routine first checks the Task Common table to determine the 'A' privilege class of the command user. If the privilege is not A, a message is issued indicating that the user is not privileged to issue the command.

The routine then gets the command operand (userid), supplied by the system operator, and validates it using SCAN routines. If the userid is valid, the user table is read to get the identification (taskid) of the specified user's conversational task. This taskid, after being verified to ensure it is not the operator's taskid, and that it represents a conversational task, is moved to a message control block (MCB) along with the FORCE taskid, and sent, via VSEND, to the task being eliminated. Finally, a message is sent to the system operator informing him that the forced shutdown of the task is complete.

After each prompting and diagnostic message is issued, a check is made for an attention interrupt. If one has occurred, the command is canceled and control is returned to the calling routine.

Error Conditions: The routine checks for an acceptable operand and notifies the operator, through user prompter, of any errors found. For the messages issued, see System Messages. FORCE also issues these minor system error codes:

SYSER Code Significance

050302901 Invalid return code from NEXTPAR when checking userid. Invalid return code from ALFNUM when checking USERID.

050302903 Error or invalid return code from MSGWR.

050302904 VSEND to log off task was unsuccessful.

System Control Block Usage: Active User Table (CHAAUL)

Data Control Block (CHADCB)

Message Control Block (CHAMCB)

System Common (CHASCM)

Task Common (CHATCM)

HOLD/DROP Command Routine (CZCM0)

This routine discontinues or re-establishes the availability of a device for TSS/360 usage. HOLD marks the system tables to prevent a malfunctioning device from being assigned to a user; DROP returns a device to the system for assignment.

Entry Points:

CZCM04 -- DROP command entry from a BPKD labeled CZCM0411.

CZCM07 -- HOLD command entry from a BPKD labeled CZCM0711.

CZCM0E -- Return from the Task Monitor following an I/O operation.

Modules Called:

PAIR (CZACS1) -- Puts an entry in the ABEND Interlock Release Table.

PRMPT (CZATJ1) -- Issues messages to the operator.

RELEASE (CZCAD3) -- Releases a device.

VMER (CZCRX2) -- Records Inboard errors.

VMSDR (CZCRY) -- Records Outboard errors.

Exits: This routine returns to the caller via the RETURN macro instruction.

Operation: The HOLD and DROP processors operate independently within this module so they are discussed separately. Discussions of the subroutines (SETUP, DOIICAL, and CZCM0E) follow HOLD and DROP.

HOLD: The HOLD routine, entered at CZCM07 (from the BPKD labeled CZCM0711), first calls the SETUP routine to process the device list provided by the command analyz-

er. Then, the first device is tested to see if it can be, or should be, held. (A partitioned or SYSOPERO device, taskid 01 or 02, can not be held; a system-dedicated, public, or in-use device can only be held at the discretion of the system operator who will be prompted when such a device is encountered.)

Now, if the device is a terminal, the CKALOC macro instruction is issued. CKALOC return codes, which determine the device type, cause processing as follows:

- 0 -- Device not in use
HOLD builds an IORCB and issues an IOCAL to disable the line; the SDAT is marked "HELD", the device group table is set to indicate a malfunctioning device.
- 1 -- Invalid return code
If this code is returned from CKALOC, the HOLD is canceled.
- 2 -- MTT Terminal
The HOLD is canceled; an MTT terminal can not be held.
- 3 -- Invalid return code
The HOLD is canceled.
- 4 -- RTAM SYSIN Terminal
System tables are marked as for return code 0; ABEND is called via the VSEND macro instruction.
- 5 -- Device assigned to task, but not SYSIN device
The system tables are marked as for return code 0; HOLD purges I/O, and disables the terminal.
- 6 -- Task SYSIN device
The system tables are marked as for return code 0; HOLD purges I/O, and calls ABEND via VSEND.

After taking this action for a terminal device, HOLD zeros the taskid, userid, and valid in the SDAT (if the device is not a system device), and puts the device SDA in the confirmation message. If there are more devices, this process is repeated; otherwise, control is returned to the caller.

DROP: The DROP routine, entered at CZCM04 (from the BPKD labeled CZCM0411), first calls SETUP to process the device list provided by the command analyzer. Then, the first device is tested to see if it can be, or should be, dropped. (If the device was not held, it can not be dropped; if the device is being used by a class 'E' user, the operator is prompted to see if the device should be dropped.) Then the path to the device is marked by the SPATH macro

instruction. Processing proceeds depending on the device type.

- terminal -- An IORCB is built; an IOCAL is issued to enable the terminal line.
- all devices -- The "held" flag is turned off in the SDAT.
- system-dedicated -- The available flag in the SDAT is left off.
- public -- The available flag is turned on.
- private -- If it is not a demountable device (tape or disk) the available flag is turned on. For a demountable device, RELEASE is called to insure that a waiting task gets the device.

The SDA of the dropped device is placed in the confirmation message, and the next device in the device list is processed. When all devices are processed, control returns to the caller.

SETUP: The SETUP routine converts input parameters to a usable form for the HOLD and DROP routines. An ABEND Interlock Table entry will be created if the calling task (must be task 001) does not have one. Then, the input parameters are checked and converted to hexadecimal.

DOILOCAL: This is the common I/O routine for HOLD and DROP. DOILOCAL builds an IORCB and a CCW list (a DROP for a dedicated device contains a SAD, ENABLE, and PREPARE; all other CCW lists contain a SAD and ENABLE). Then, a DECB is built; the IOCAL is executed. CKALOC is called to return the device to the system; control returns to the caller.

CZCMOE: This routine is an I/O posting routine called by the Task Monitor when HOLD/DROP I/O is complete. CZCMOE checks for errors: VMER is called to record inboard errors; VMSDR is called to record outboard errors.

System Control Block Usage:
Data Event Congrol Block (CHADEC)

- Interrupt Storage Area (CHAISA)
- I/O Request Control Block (CHAIOR)
- I/O Statistical Data Table (CHASDT)
- Message Control Block (CHAMCB)
- Symbolic Device Allocation Table (CHASDA)
- Task Common (CHATCM)

LABEL Command Routine (CZABX)

LABEL allows the operator to add or remove labels on tape volumes in conversational or nonconversational mode. (See Chart RS.)

Entry Points:

CZABX1 -- Normal entry.

CZABXX -- BPKD entry point.

Modules Called:

CLOSE (CZCLBC) -- Closes the SYSIN data set.

DDEF (CZAE4) -- Creates the SYSIN data set.

MTREQ (CZCAA1) -- Mounts the volume.

OBEY (CZASA1) -- Executes the SYSIN data set.

OPEN (CZCLA0) -- Opens the SYSIN data set.

PRMPT (CZATJ1) -- Writes diagnostic messages to the operator.

REL (CZAFJ3) -- Releases the ddname of the SYSIN data set.

RELEAS (CZCAD) -- Releases the tape device.

Volume Label Reader (CZCWX3) -- Label/unlabels a tape volume.

VSAM PUT (CZCOS) -- Writes the data set to public VAM storage.

XWTO (CZABQ1) -- Writes messages to the operator (WTO).

Exits: LABEL exits to the caller by a RETURN macro instruction. Diagnostic messages notify the operator of errors.

Operation: CZABX, entered at the CZABX1 entry point, first validates the userid (it must be SYSOPER0), and the command parameters. When the command is initiated from the conversational operator task (TASKID = 1), a SYSIN data set consisting of LOGON, SECURE, LABEL, RET, and LOGOFF commands is created to form a new nonconversational task. The SYSIN data set is named SYS-OLTNN, where NN are numeric characters used to create a unique data set name. CZABX initiates the SYSIN task by requesting the OBEY of an EXECUTE command. This data set is erased at LOGOFF by the RET command.

The new SYSIN data set is now entered as a nonconversational task. MTREQ mounts the tape (with a ring, if the tape is file-protected); CZCWX3 (BSAM) performs the label processing. When no label is specified, the 80 character buffer contains

blanks. This buffer is used to contain the 80 character tape label when one is specified. On return from CZCWX, the device is released and a completion message is issued to the operator.

When the command is not issued from the conversational operator's task, the SYSIN data set is not created, and the program logic performs the labeling procedure as shown above.

Error Conditions: Diagnostic messages are issued for invalid command parameters, and for unauthorized use of the command.

System Control Block Usage: Data Control Block (CHADCB)

Symbolic Device Allocation Table (CHASDA)

System Common (CHASCM)

Task Common (CHATCM)

Task Data Definition Table (CHATDT)

MSG/BCST Command Routine (CZACF)

This routine processes both MSG and BCST commands. If called via MSG, the routine sends a message to a specified conversational user, and writes it in the operator's log (SYSLOG). If called via BCST, the routine sends a message to all currently active conversational users before writing it in SYSLOG. This routine operates in conversational mode only and is restricted to users with privilege class A. (See Chart SS.)

Entry Points:

CZACF1 -- Entry for MSG command

CZACF2 -- Entry for BCST command

CZACF3 -- SYNAD entry

CZACF4 -- EODAD entry

Modules Called:

EBCDTIME -- Gets time and date.

ESETL -- Ends sequential retrieval.

GET -- Gets user table record.

MSGWR (CZAAD2) -- Issues messages to operator.

NEXTPAR (CZAAC1) -- Gets command operand and validates keyword and userid.

PUT -- Inserts message in SYSLOG.

READ (CZCPE) -- Reads specific user table record.

VSEND -- Sends message to another task.

Exits: The routine normally returns to the calling routine, via the RETURN macro instruction. If a system error occurs, the routine exits through SYSER.

Operation: When processing a MSG command, the routine gets and validates the keyword and user identification (userid) supplied as the command operand. The entry for that user is then located in the user table and checked to ensure he has an active conversational task. If so, the message text is fetched and inserted, along with the current date and time, in a message control block (MCB); the message is sent, via VSEND, to the specified user's task; and the message, date, and time are inserted in SYSLOG. The routine then returns control.

When processing a BCST command, the routine builds an MCB with the message text and the current date and time. (Note that although many of the coding sequences used for MSG are shared by BCST, the processing is shown separately on the accompanying flowcharts for clarity.) The user table is now opened and the routine scans the table entries looking for active conversational tasks. (The operator's own task will be ignored.) When a conversational task is found, its task identification (taskid) is extracted from the user table entry and used to send the message, via VSEND, to that task. The routine continues scanning the table and issuing the message until it has read the entire table or until an attention interrupt occurs. At that point the user table is closed; the message, with date and time, is inserted in SYSLOG; and a confirmation message, if appropriate, is issued.

Error Conditions: The routine checks for an acceptable operand and notifies the operator through user prompter, of any errors found. For the messages issued, see System Messages. This routine also issues this major system error:

<u>SYSER Code</u>	<u>ABEND Message</u>	<u>Significance</u>
050303001	SYSTEM ERROR	I/O error when reading user table or writing in SYSLOG.

System Control Block Usage:
Data Control Block (CHADCB)

Message Control Block (CHAMCB)

Task Common (CHATCM)

User Table (CHAUSE)

PRINT Routine (CZABG)

This routine prints a specified data set from a private VAM or SAM volume to a high-speed printer at the computer installation. (See Chart XX.)

Entry Points:

CZABG1 -- Normal entry

CZABG2 -- EODAD entry for input data set

CZABG3 -- SYNAD entry from I/O (read error)

CZABG4 -- SYNAD entry from printer as a data set

Modules Called:

ABEND (CZACP1) -- Prepares task for abnormal termination.

BSAM Convert (CZCWA1) -- Converts to BSAM.

CHECK (CZCRCS) -- Tests for completion of read operation.

CLOSE (CZCLB) -- Closes input and output data sets

DDEF (CZAEA4) -- Defines input data set and output printer as a data set.

FINDJFCB (CZAEB1) -- Finds input data set JFCB.

FREEMAIN (CZCHA3) -- Returns buffers to buffer pool.

GET (CZCPB) -- Gets record from sequential data set.

GETMAIN (CZCHA2) -- Gets input buffers.

INTINQ (CZCJIA) -- Waits for not ready/ready interruption on printer, after SETUR.

LOCATE (CZCFL1) -- Determines DSORG of input data set.

LOGOFF (CZAFN1) -- Logs task off system.

MSGWR (CZAAD2) -- Issues messages.

OPEN (CZCLA) -- Opens input and output data sets.

PUT (CZCOS) -- Writes output record (using MSAM).

RCR (RATION) -- Maintains an accounting of the number of records printed for the user.

READ (CZCRA) -- Reads input record from tape.

SETUR (CZCMD1) -- Insures proper forms mounted on printer.

VMTI-2 (CZATD2) -- Logs on task, opens SYSOUT, and opens SYSMLF.

XWTO (CZABQ1) -- Sends abnormal task end message to operator (WTL).

Exits: PRINT normally exits to LOGOFF, via a call. If a system error occurs, PRINT exits to ABEND.

Operation: This routine operates independently of the task that issued the PRINT command. CZATD2 is called to logon the task, open its SYSOUT, and open SYSMLF. LOCATE is invoked to determine the data set organization by interrogating the returned catalog S-Block. PRINT next obtains the symbolic device address (SDA), contained in the batch work queue (BWQ), of the devices to be used by the task. The SDAs are implanted in the DDEF parameter lists. The routine then calls DDEF to define the input and output data sets, opens those data sets (via OPEN) and obtains buffers (via GETMAIN), if needed.

Before initiating printing, the SETUR macro instruction is invoked to insure that proper forms are mounted on the output device.

After setting up the initial print line to include the user identification (userid) and batch sequence number (bsn) associated with this task, PRINT inspects the spacing operand to decide which of three printing loops to use. If EDIT was not specified, the spacing operand serves as a control character for line spacing. In this case, page heading and line numbers are printed if requested.

If EDIT was specified, the subroutine, GETRCDD, is called to get the first input records. If there is a request for a printer for a TSSS dump tape, special flags are set to allow processing of a multifile tape when CZABG2 (EOD) is entered. Then, or if no dump is requested, a check is made to see if ASCII or machine code control characters are used. ASCII control characters are converted to machine code and tested if the output record is from the previously read input record (the first time through, this contains the userid and bsn generated by PRINT). This sequence is necessary because ASCII control characters cause a space then print, but machine code control characters cause a print then space. By stepping one record ahead, the routine achieves proper spacing. The loop starting at location B1 is used for machine code control characters.

Looping now continues until the input data has been exhausted. Two PRINT subroutines, GETRCDD and PUTRCDD, get the input records and print the output records. After the last record has been printed, PRINT checks to see if error records occurred or if records were skipped. If not, it sets an indicator in task common so that the task's SYSOUT will not be printed (because there are no errors, it will not be needed). The RCR (RATION) macro is then invoked to account for the number of records printed by the user before exiting to LOGOFF. Otherwise, it issues messages to SYSOUT to show the number of error records, skipped records, etc.

If the task must be terminated prematurely, a message to that effect will be sent to the operator (via WTL), written on SYSOUT, and also, if possible, printed. Except for a system error, exit will still be made to LOGOFF.

Error Conditions: The routine writes an error message on SYSOUT for each error; the messages that may appear are listed in System Messages.

For any of the following errors, the PRINT task is immediately terminated and the message PRINT TASK BSN xxxx - ABNORMALLY TERMINATED is sent to the operator and to SYSOUT:

- DDEF unable to define input or output data set.
- Input data set not SAM, VSAM, or VISAM.
- VSAM or VISAM record format is undefined.
- SAM data set not on magnetic tape.
- EDIT specified, but neither USASI nor machine code control characters are indicated.

If the length of an input record is less than the number of the first byte to be printed, the record is skipped. If there is a read error on tape input, the record is written in hexadecimal format on SYSOUT, along with an error message. User option (accept, skip, or end) decides the next action. If an invalid machine code or USASI control character is detected, the record is written in hexadecimal format on SYSOUT: the machine code control character is replaced with a print and single space; the USASI control character with a single space and print. The routine issues these minor system errors:

<u>SYSER Code</u>	<u>ABEND Message</u>	<u>Explanation</u>
050203101	INVALID OR	Invalid return ,

UNEXPECTED RET code received
CODE FROM A from a called
CALLED ROUTINE module.

050203102 Same as above Error return code
received from a
called module.

050203003 ERROR ON Unexpected
INTINQ FOR asynchronous
ASYNCHRONOUS interrupt from
INTERRUPT printer.

050203104 S BLOCK DSD does not
RETURNED BY describe a
LOCATE WAS private or a
NOT DSD public data set.

050203105 DSORG IN DSORG in DSD not
SBLOCK NOT VI, VS, or
VI, VS, OR BSAM.
PS.

System Control Block Usage:
Batch Work Queue (CHABWQ)

Catalog SBLOCK (CHACCC)

Data Control Block (CHADCB)

Data Event Control Block (CHADEC)

Interrupt Control Block (CHAICB)

Interrupt Storage Area (CHAISA)

Task Common (CHATCM)

Task Data Definition Table (CHATDT)

User Table (CHAUSE)

PUNCH Routine (CZABH)

This routine punches a specified VSAM or
VISAM data set from a private VAM volume
onto cards. (See Chart WW.)

Entry Points:

CZABH1 -- Normal entry.

CZABH2 -- EODAD entry for input data set.

CZABH3 -- SYNAD entry for input data set.

CZABH4 -- SYNAD entry for card punch as
a data set.

Modules Called:

CLOSE (CZCLB) -- Closes input and output
data sets.

DDEF (CZAEA4) -- Defines input data set and
output punch as a data set.

GET (CZCPB) -- Reads a logical record from
sequential data set.

INTINQ (CZCJIA) -- Waits for not ready/
ready interruption on punch, after
SETUR.

LOGOFF (CZAFN1) -- Terminates PUNCH task.

MSGWR (CZAAD2) -- Issues messages to
SYSOUT.

OPEN (CZCLA) -- Opens input and output
data sets.

PUT (CZCOS) -- Writes a logical record in
output data set.

RCR (RATION) -- Maintains an accounting of
the number of cards punched by user.

SETUR (CZCMD1) -- Insures proper forms
mounted on output device.

VMTI-2 (CZATD2) -- Logs on task, opens its
SYSOUT, and opens SYSMLF.

XWTO (CZABQ1) -- Sends abnormal task end
message to operator (WTL).

Exits: The routine normally exits to
LOGOFF, via a call. If a system error
occurs, the routine exits via ABEND.

Operation: This routine operates as a
separate task, entirely independent from
the task that issued the PUNCH command.
The first actions performed are, therefore,
those needed to initiate a new task.
CZATD2 is called to logon the task, open
its SYSOUT, and open SYSMLF.

The routine next obtains the symbolic
device address (SDA) of the devices to be
used by the task from the device code
fields in the batch work queue (BWQ). The
SDAs are then implanted in the DDEF para-
meter lists. The routine then calls DDEF
to define the input and output data sets
and opens those data sets (via OPEN).

The SETUR macro instruction is then
invoked to insure that the proper card form
is mounted in the output device.

The input data set is now read, one log-
ical record at a time, using the subrou-
tine, GETRCD. After each read, the EDIT
option is tested to see if a specific
stacker was requested or if the record has
a control character. In the latter case,
the character is checked and, if USASI,
converted to machine code before being
placed in the output identification card,
containing user identification and batch
sequence number, which will be punched
automatically as the first card of the out-
put card deck.

When all input has been processed, input
and output data sets are closed and a test

is made to see if the task has concluded normally. If so, and end of task message is issued to SYSOUT (via MSGWR).

The routine finally writes on SYSOUT the number of records that were read, punched, and skipped, and the number of error records. RCR (RATION) is then invoked to maintain an accounting of the number of records punched by the user. Exit is then taken to the LOGOFF command routine, which will terminate the task.

Error Conditions: The routine writes an error message on SYSOUT for each error; the messages that may appear are listed in System Messages.

For any of the following errors, the PUNCH task is immediately terminated and the message PUNCH TASK BSN xxxx TERMINATED BEFORE COMPLETION is sent to the operator:

- DDEF unable to define input or output data set.
- Input records undefined.
- EDIT specified but input record format does not have USASI or machine code control character.
- Input data set organization not VSAM or VISAM.

If the length of an input record is less than the number of the first byte to be punched, the record is skipped. If an invalid machine code or ASCII control character is detected, the record is written in hexadecimal form on SYSOUT; an error message will indicate that this is a data error, and the invalid control character will be replaced with a control character to punch and select stacker 1. This routine issues these minor system errors:

<u>SYSER Code</u>	<u>ABEND Message</u>	<u>Explanation</u>
050203201	INVALID OR UNEXPECTED RETURN CODE FROM A CALLED ROUTINE.	Invalid return code received from a called module.
050203202	Same as above	Error return code received from a called module.
050203203	ERROR ON INTINQ FOR ASYNCHRONOUS INTERRUPT	Unexpected asynchronous interrupt from punch.

System Control Block Usage:
Batch Work Queue (CHABWQ)

- Data Control Block (CHADCB)
- Data Event Control Block (CIADEC)
- Interrupt Control Block (CHAICB)
- Interrupt Storage Area (CHAISA)
- Task Common (CHATCM)
- User Table (CHAUSE)

RT Routine (CZABF)

The RT routine reads a data set from a private magnetic tape volume, enters it in system storage and catalogs it under the name specified in the operand. The RT routine operates in nonconversational mode and in the operator's task only. (See Chart UU.)

Entry Points:

- CZABF1 -- Main entry point.
- CZABF2 -- EODAD entry point for tape input data set.
- CZABF3 -- SYNAD entry point for tape input data set.
- CZABF4 -- VAM output data set SYNAD entry point.

Modules Called:

- CHECK (CZCRC) -- Checks the last read into input buffer.
- CLOSE (CZCLB) -- Closes input and output data sets.
- DDEF (CZAEA5) -- Creates a JFCB entry in the TDT, and requests external storage allocation space for output data set, and to create a JFCB for a tape input data set.
- FREEBUF (CZCNA) -- Releases input buffers.
- GETBUF (CZCMA) -- Gets input buffers.
- GETMAIN (CZCGA2) -- Obtains an area of virtual storage for output records.
- LOGOFF (CZAFN1) -- Logs off (RT) task.
- MSGWR (CZAAD2) -- Issues messages on SYSOUT.
- OPEN (CZCLA) -- Opens input and output data sets.
- PUT (CZCOS) -- Writes output record.
- RCR (RATION) -- Maintains an accounting of the number of records read by the user.

READ (CZCRA) -- Reads the input tape.

VMTI-2 (CZATD2) -- Logs on task, opens SYS-
OUT, opens SYSMLF.

WRITE (CZCRAS) -- Writes input tape.

XWTO (CZABQ1) -- Sends devices allocated
message to the Batch Monitor and abnorm-
al task end message to the operator.

Exits: The routine normally terminates via
a call to LOGOFF. If a system error
occurs, the routine exits to ABEND.

Operation: This routine operates as a
separate task entirely independent of the
operator's task. The first actions per-
formed are, therefore, those needed to
initiate a new task.

CZATD2 is called to logon the task, open
its SYSOUT, and open SYSMLF.

The routine then obtains the symbolic
device address (SDA) of the input device
from the batch work queue (BWQ) and inserts
the SDA into the input DDEF parameter list.

DDEF is called to create JFCBs for the
input and output data sets. The input data
set is opened, and the input buffers are
obtained via GETBUF. The output data set
is opened, and an area of virtual storage
is obtained in which to build output rec-
ords. After each logical record is read, a
test is made to determine if a line data
set is to be created. If so, the output
record length and line number are inserted
in the output. The logical input record is
now moved into the output record area, the
line number is incremented if the output is
a line data set, and the output record is
written.

When all the input data has been read,
the input and output data sets are closed.
The output data set is cataloged, the input
buffers are released through FREEBUF, and
an end of task message is written on SYS-
OUT. If the task is not completed normal-
ly, an abnormal task end message is sent to
the system operator, via WTL, and is writ-
ten on SYSOUT, the record counts are writ-
ten on SYSOUT, RCR (RATION) is invoked to
account for the number of records read by
the user, and RT terminates with a call to
LOGOFF.

Note: RCR RATION is invoked for both norm-
al and abnormal task end.

Error Conditions: The routine checks for
acceptable operands and notifies the user,
through MSGWR, of any errors found. For
the messages issued, see System Messages.
This routine issues these minor system
errors:

<u>SYSER Code</u>	<u>ABEND Message</u>	<u>Significance</u>
050203001	INVALID CODE FROM A CALLED ROUTINE	Invalid return code from DDEF, MSCWR, or VSEND routines called by RT.
050203002	INVALID CODE FROM A CALLED ROUTINE	Error return code from MSGWR or VSEND routines called by RT.

System Control Block Usage:
Batch Work Queue (CHABWQ)

Data Control Block (CHADCB)

Data Event Control Block (CHADEC)

Interrupt Storage Area (CHAISA)

Task Common (CHATCM)

User Table (CHAUSE)

SHUTDOWN Command Routine (CZACN)

This routine enables the system operator
to terminate operation of the time-sharing
system. SHUTDOWN operates in conversational
mode only. (See Chart VV.)

Entry Points:

CZACN1 -- Normal entry

CZACN2 -- EODAD entry

CZACN3 -- Batch Monitor entry

Modules Called:

CATFLUSH (CZCFX3) -- Copies all catalog
entries to USERCAT.

CLOSE (CZCLB) -- Closes user table and
SYSLOG.

CZABA9 -- Shuts down nonconversational
tasks.

EBCDIME -- Gets current time and date and
converts it from binary to EBCDIC.

ESETL (CZCPD) -- Releases interlock imposed
by SETL.

GET (CZCPB) -- Reads record in user table.

LOGOFF (CZAFN2) -- Logs off operator's
task.

MOSP (CZACA5) -- Writes to SYSLOG.

MSGWR (CZAAD2) -- Sends message to opera-
tor's terminal.

PRINT (CZABD3) -- Provides for printing of
SYSLOG.

SETL (CZCPC) -- Points to specified record in user table.

TSEND -- Ends time slice.

VSEND -- Sends intertask message.

Exits: The routine normally exits to LOGOFF (CZAFN2). If a system error occurs the routine exits to SYSER.

Operation: The routine first checks the Task Common table to determine the privilege class of the command user. If the privilege is not A, a message is issued indicating that the user is not privileged to issue the command.

The initiation of any new task is inhibited by setting a flag (SCMITI) in the System Common. Then the Batch Monitor is called at CZABA9 to terminate all nonconversational tasks. When control is returned, SHUTDOWN begins processing conversational operations; CZACA5 then writes the shutdown message to SYSLOG.

All active conversational tasks (except the operator's task) are found by searching the user table. Each entry in the user table is retrieved and checked to determine if it has an active conversational task in the system. When an active task is found, its task identification (taskid) is taken from the user table entry, and a VSEND is issued to notify this conversational user that his task is being terminated because of a system shutdown. The external interrupt processor types out the shutdown message; then it calls ABEND for a privileged task, or LOGOFF for a nonprivileged task.

After a shutdown message has been sent to each conversational task, the user table is again searched to determine if all conversational tasks have been removed from the system. (ABEND sets the USEACT flag to zero to indicate this fact). If any conversational task remains active, a TSEND SVC is issued to end the time slice of the operator's task. When the next time slice begins for the operator's task, the same check is made in the user table and, if the task is still active, the procedure for ending the operator's time slice is repeated up to 16 times for one active task. If a task remains active after 16 TSENDS in the operator task, processing continues so that the system can be shut down.

When all conversational tasks other than the operator's have been abnormally terminated, a return is made to the Task Monitor and SHUTDOWN waits for a call at CZACN3 from the Batch Monitor indicating that all nonconversational tasks have been terminated.

When the Batch Monitor calls SHUTDOWN at CZACN3, the current time and date are written into the operator's log (SYSLOG) via CZACA5. After the time of the completion of shutdown has been recorded, SYSLOG is closed. Then a PR macro instruction is issued to provide printed copy of SYSLOG after the next system startup. A message is sent, via MSGWR, to the operator's terminal to notify him that shutdown has been completed. Then, CATFLUSH (CZCFX3) is called to update the user catalogs (USER-CAT) from the SCRATCHCAT members. SCRATCHCAT is then erased. Finally, the operator's task is logged off with a call to LOGOFF.

Error Conditions: Routine issues these minor system errors:

<u>SYSER Code</u>	<u>ABEND Message</u>	<u>Significance</u>
050303801	None	VSEND cannot be completed successfully.
050303803	None	PRINT request for SYSLOG not accepted by BULKIO preprocessor.
050303805	None	Invalid return code received from a called module.

Processing continues after all SYSERS.

System Control Block Usage:
Data Control Block (CHADCB)

Message Control Block (CHAMCB)

Message Event Block (CHAMEB)

System Common (CHASCM)

System Operator Log (CHALOG)

Task Common (CHATCM)

User Table (CHAUSE)

WT Routine (CZABI)

This routine writes a specified data set onto a private magnetic tape volume in proper format for off-line printing. (See Chart YY.)

Entry Points:

CZABI1 -- Normal entry.

CZABI2 -- EODAD entry for VAM input data set.

CZABI3 -- SYNAD entry for VAM input data set.

CZABI4 -- SYNAD entry for BSAM output tape data set.

Modules Called:

CATALOG (CZAEI2) -- Catalogs the data set on tape.

CATALOG (CZAEI3) -- Catalogs a data set on VAM storage.

CHECK (CZCRC) -- Tests for completion of write operation.

CLOSE (CZCLB) -- Closes input and output data sets.

DDEF (CZAEA4) -- Defines input and output data sets.

ERASE (CZAEJ3) -- Erases a VAM data set, if requested by the user.

FREEBUF (CZCNA) -- Returns buffers to buffer pool.

GET (CZCPB) -- Reads a logical record from VAM.

GETBUF (CZCMA) -- Gets output buffers.

LOGOFF (CZAFN1) -- Logs task off system.

MSGWR (CZAAD2) -- Sends message to SYSOUT.

OPEN (CZCLA) -- Opens input and output data sets.

RCR (RATION) -- Maintains an accounting of the number of records written by the user.

WRITE (CZCRA) -- Writes a record on tape.

WTL (CZABQ1) -- Sends an ABEND message to the operator task.

VMTI-2 (CZATD2) -- Logs on task, opens SYSOUT, and opens SYSMLF.

Exits: The routine normally exits to LOGOFF, via a call. If a system error occurs, the routine exits to ABEND.

Operation: This routine operates as a separate task, entirely independent from the task that issued the WT command. The first actions performed are therefore those needed to initiate the new task: CZATD2 is called to logon the task, open its SYSOUT, and open SYSMLF.

The routine next obtains the symbolic device address (SDA) of the private input device, and the output device from the batch work queue. It enters those SDAs in the DDEF parameter list. As the last steps before writing, the routine calls DDEF to define the input or output data sets, opens

them (via OPEN) and obtains output buffers (via GETBUF).

The routine now builds a blank print line to provide for initial page positioning, and tests if editing has been specified. If not, the page header is set up as required and the routine reads in a record (using included subroutine GETRCD); tests for a new page; and then writes the record onto tape (using included subroutine PUTRCD). When a new page is indicated, the routine will issue the page number and page header if they have been requested. This reading, testing, and writing process continues until the input data set has been exhausted.

If editing is indicated, the routine distinguishes between ASCII and machine code control characters. It converts each ASCII code to machine code, checks that it is valid, and moves it to the output record for writing. The data in the output record is from the previously read input record (except on the first time through when the output record is the blank line generated by WT). This order is necessary since machine code control characters cause a print then space, but ASCII codes specify space then print. By stepping a record ahead, the routine achieves proper spacing. For machine codes, the routine checks that the control character is valid, then moves it to the output record.

After all input records have been read and then written onto tape, the output buffers are released, the input and output data sets are closed, and, if requested, the output data set is cataloged. The routine writes on SYSOUT the number of records read, written, and skipped, and the number of error records. RCR (RATION) is invoked to maintain the accounting for the number of records written before exiting to the LOGOFF command routine to terminate the task.

Error Conditions: The routine writes an error message on SYSOUT for each error; the messages that may appear are listed in System Messages.

For any of the following errors, the WT task is immediately terminated and the message WRITE TAPE TASK BSN xxxx ABNORMALLY TERMINATED is sent to the operator:

- DDEF unable to define input or output data set.
- Input data set organization not VSAM or VISAM.
- Input record format is undefined.

- EDIT specified, but neither FORTRAN nor machine code control characters.¹
- Characters are indicated.

If the length of an input record is less than the number of the first byte to be written, the record is skipped. If an invalid machine code or ASCII control character is detected, the record is written in hexadecimal format on SYSOUT; the machine code control character is replaced with a write and a single space; the USASI control character with a single space and write. An accompanying message will note the faulty control character as a data error. This routine issues these minor system errors:

 FORTRAN control characters are control characters defined by American National Standard FORTRAN, ANSI X3.9-1966.

<u>SYSER Code</u>	<u>ABEND Message</u>	<u>Significance</u>
050203301	INVALID OR UNEXPECTED CODE FROM A CALLED ROUTINE.	Invalid return code received from a called module.
050203302	Same as above.	Error return code from a called module.

System Control Block Usage:
 Batch Work Queue (CHABWQ)

Data Control Block (CHADCB)

Task Common (CHATCM)

User Table (CHAUSE)

SECTION 6: MACRO INSTRUCTION HANDLING

This section describes the handling of those command system macro instructions that use BULKIO preprocessor initiation, and those macro instructions used to communicate with other tasks (as well as with the system operator). Figure 8 contains the names and functions of the macro instructions.

The WT, PR, and PU macro instructions are handled by the BULKIO preprocessor routines that process the corresponding WT, PRINT, and PUNCH commands. As with the command, the preprocessor calls the batch monitor to initiate an independent background operation. Unlike command processing, the Command Analyzer and Executor (CZASA) is bypassed as control goes directly to the BULKIO preprocessor.

The command routine itself performs in the same way whether it is processing a command or a macro instruction, with the following two exceptions:

1. The command routine issues no message if it has been entered by a macro instruction. Instead, it supplies a return code to show if an error was found and, in some cases, to indicate

the type of error. Error handling is left to the object program that issued the macro instruction.

2. The macro expansion links to a special entry in the processing routine to allow the routine to set indicators to inhibit messages.

The VSENDR, WTO, WTL, WTOA, and WTOR macro instructions provide communication with the system operator, and with specific tasks.

XWTO Routine (CZABQ)

The XWTO routine performs the actions requested by five macro instructions: WTO, WTL, WTOA, WTOR and VSENDR. WTO, WTL, WTOR and WTOA each involve some sort of communication with the system operator. The VSENDR macro instruction is used to communicate with any specified task. The expansion of these macro instructions results in linkage to XWTO, with an operation code set to indicate which macro instruction is to be processed. XWTO will then carry out the requested action before returning control to the calling object program. (See Chart ZA.)

Macro Instruction Name	Function	Routine to Which Macro Instruction Links
PU	To punch a specified data set onto cards.	PUNCH command routine in CZABD7
PR	To print a data set on a printer.	PRINT command routine in CZABD3
VSENDR	To send an inter-task message and await the response.	XWTO routine, CZABQ
WT	To write a data set onto tape for subsequent offline printing.	WT command routine in CZABD9
WTL	To write a message in the operator log.	XWTO routine, CZABQ
WTO	To write a message to the console and to the operator log.	XWTO routine, CZABQ
WTOR	To write a message to the console and to the operator log, and await a response.	XWTO routine, CZABQ
WTOA	To write a message to the operator which indicates some operator action. (It is preceded by arrows.)	XWTO routine, CZABQ

Figure 8. Macro instructions

Entry Points: CZABQ1 -- Normal entry.

Modules Called:

- ABEND (CZACP) -- Disconnects task.
- ATPOL -- Checks for Attention Interruption.
- AWAIT (CEAP7) -- Places task in delay state.
- ITI -- Disables all except program and SVC task interrupts by setting interrupt inhibition lock byte (ISALCK).
- PTI -- Enables task interrupts by resetting interrupt inhibition lock byte (ISALCK)
- REDTIM (CEAS6) -- Reads the current system time.
- VSEND (CEAQ5) -- Sends information to another task.

Exits: This routine always returns to the calling routine, via RETURN.

Operation: The macro expansion places, in general register 0, a hexadecimal code that indicates the macro instruction to be processed: 0 for WTO, 4 for WTL, 8 for WTOR, C for VSENDR, and 10 for WTOA.

On entry, all synchronous I/O interrupts are inhibited. A code in register zero determines which subprocessor is to be given control. In the cases of WTO, WTOA and WTL, the number of calls occurring within the last 0.2 seconds is checked. If in excess of 14, the task is abnormally terminated.

In every case, the routine builds a message control block (MCB), inserting in it the text that is to be sent. The message is then transmitted, via VSEND, to the system operator's task or, for VSENDR only, to the caller-specified task. For WTO, WTOA and WTL macro instructions, this concludes XWTO processing.

For WTOR and VSENDR, the routine sets up an MCB and, because a reply is expected, a message event control block (MEB). After transmitting the message, the routine issues an AWAIT to put its own task in a delay state that continues until an interruption is received. When an interrupt occurs, XWTO checks first that it is appro-

priate and then that the reply has actually been received (the MEB shows this). If the interruption was an attention interruption from the terminal, a cancellation message is sent to the operator task so that it may remove the reply queue entry. The reply text is now moved to the area specified in the macro instruction operands (unless protected), and XWTO returns to the calling routine. On exit, the original VPSW masks are restored.

Error Conditions: A hexadecimal code will be returned in the fourth byte of general register 15.

<u>Return Code</u>	<u>Significance</u>
00	No error detected.
04	Attention interrupt occurred during delay state.
0C	Invalid message length; no message sent. Or, reply length exceeds specified length plus 7 additional characters (the 7 extra allow for doubleword usage); message was sent but reply not returned; reply area protected. Applies for WTOR or VSENDR only.
10	Invalid reply text or length (WTOR and VSENDR).
14	Task with specified task does not exist (for VSENDR only).
18	Task with specified task identification is not accepting message (for VSENDR only).
1C	Operator task is sending task.
20	Invalid macro code in register 0.

System Control Block Usage:
Interrupt Storage Area (CHAISA)

- Message Control Block (CHAMCB)
- Message Event Control Block (CHAMEB)
- Operator Header (CHAOPH)
- Task Common (CHATCM)
- Virtual Program Status Word (CHAVPS)

SECTION 7: TIME SHARED UTILITY PROGRAMS

The Time Shared Utility Programs (PATCLEAR, PATFIX, and DUMP/RESTORE) process the PATCLEAR, PATFIX, and DMPRST commands in a time-sharing environment. The commands, entered from a terminal or as a background task, interface with the command system through a BULTIN in the system profile, and a BPKD in the PSECT of each processor. The processors may handle their associated commands in the same task or, if requested, call the time shared utility module to create a separate background task which will reenter the command processor for final processing. (PATFIX does not call the time shared utility module.)

PATCLEAR

The PATCLEAR program (CZUFO) zeros those bytes in the Page Assignment Table (PAT) which represent data or DSCB pages on VAM2 2311 or 2314 disk packs, freeing the pack for reuse. If the task is to be run in a separate nonconversational mode, CZUBC is called to initiate the new task which in turn calls PATCLEAR for processing. If the task is to remain conversational, or in the same nonconversational task in which it was entered, PATCLEAR handles all processing for the request.

PATFIX

The PATFIX program (CZUPF) checks Page Assignment Tables for errors on the specified volumes; it also constructs new Page Assignment Tables from the DSCB and data pages of the volumes. Then, if the user requests, these new tables are put on the volumes in place of the old tables. Also, an error report is built showing all the errors found in the Page Assignment Tables.

DMPRST

The DMPRST program (CZUFA) dumps/restores data on a direct access volume mounted on an IBM 2311 disk storage device, or 2314 disk storage facility. The data contents of the direct access volume, except for the home address, can be dumped to another 2311 or 2314 disk, or to a magnetic tape, and can be restored to a direct access device of the same type as the source volume. As with PATCLEAR, DMPRST calls CZUBC to initiate a requested background task.

TIME SHARED UTILITY MODULE

The time shared utility module (CZUBC) creates a separate background task by building a VISAM data set, from a supplied command string, and by executing that data set via an OBEY macro instruction. The new task now reenters the appropriate command processor. (See Figure 9.)

PATCLEAR Processor (CZUFO)

CZUFO initializes PAT entries for data or DSCB pages in VAM-2 format (Chart ZB.)

Entry Points:

CZUFO1 -- command entry

CZUFO2 -- macro entry

CZUFO3 -- BPKD macro label

Modules Called:

FREEMAIN (CZCGA) -- Frees VM pages.

MTREQ (CZCAA) -- Mounts the volume and returns the SDA.

PRMPT (CZATJ) -- Prompts the user with completion messages.

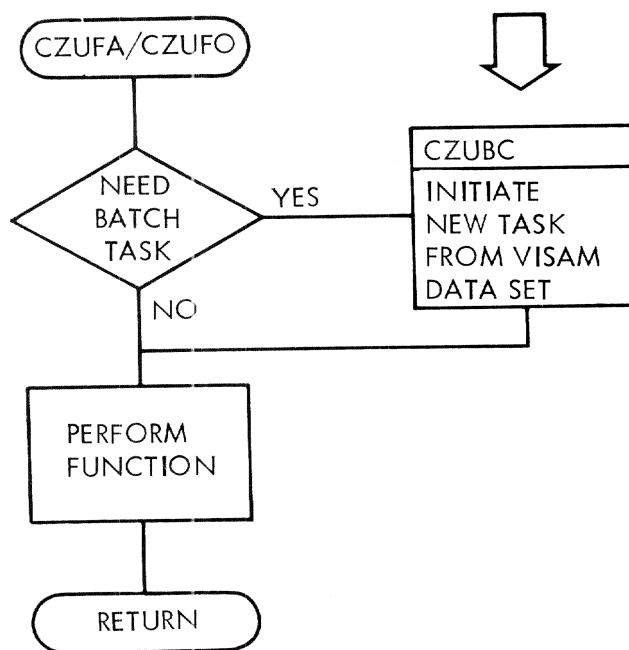


Figure 9. Time shared utility module usage (CZUFA and CZUFO)

RELEASES (CZCAD) -- Releases 2311/2314 disk devices.

TSU Module (CZUBC) -- Initiates a background task.

VAMINIT (CZCEQ) -- Reads in the PAT.

Exits: Control is returned to the caller with one of the following hexadecimal return codes in general register 15.

00 -- normal completion.

04 -- RELEASES failed, job terminated.

08 -- invalid parameter, job terminated.

0C -- missing parameter and no default value, job terminated.

10 -- VOLID match in CHAPVT, job terminated.

14 -- invalid RUNMODE parameter, job terminated.

18 -- volume can not be mounted, job terminated.

1C -- PATCLEAR canceled by attention interruption.

20 -- CZUBC failed, job terminated.

24 -- error in reading PAT, job terminated.

28 -- error in writing PAT, job terminated.

Operation: PATCLEAR first determines the validity of the command parameters. Invalid parameters, attention interruptions, and VOLIDs for mounted public volumes cause control to be returned to the caller with an appropriate return code for the error. Now, if a conversational user requested a nonconversational RUNMODE, CZUBC is entered with a parameter list pointing to command strings for SECURE and PATCLEAR commands. CZUBC will create and execute a background task which will, in turn, enter PATCLEAR for processing at CZUF01.

On return from CZUBC, or if the request is to be processed in this same task, MTREQ causes the volume to be mounted; PATCLEAR reads the symbolic device address (SDA) from CHASDA. VAMINIT reads the PAT into storage. PATCLEAR translates all bytes, except those containing X'C0' and X'7F' configurations, into all zeros (X'C0' and X'7F' are left as is). The relocation entries are also zeroed. A PGOUT SVC writes the PAT back to the disk; FREEMAIN releases the device. Any messages are now issued, via PROMPT, and control is returned with the appropriate code in register 15.

Error Conditions: On exit, a return code in general register 15 may indicate one of these errors:

- Error return from a macro or called module.
- Missing or invalid parameter.
- Specified RUNMODE is not allowed this user.
- Requested VOLID refers to a mounted public volume.

System Control Block Usage:
Interrupt Storage Area (CHAISA)

Public Volume Table (CHAPVT)

Symbolic Device Allocation Table (CHASDA)

Task Common (CHATCM)

DMPRST Processor (CZUFA)

CZUFA dumps/restores data on direct access volumes. (See Chart ZC.)

Entry Points:

CZUFA1 -- Macro entry

CZUFA2 -- Command entry

CZUFAB -- BPKD macro label

CZUFAD -- Disk/tape access CSECT

Modules Called:

CLOSE (CZCLB) -- Closes tape DCB.

DDEF (CZAEA) -- Defines the tape.

FREEMAIN (CZCGA3) -- Returns buffer space.

GETMAIN (CZCGA2) -- Obtains buffer space.

GDV (CZASDX) -- Gets a default value for a parameter.

MTREQ (CZCAA) -- Mounts a volume.

OBTAIN/RETAIN (CZCFO) -- Reads/Writes volume labels.

OPEN (CZCLA) -- Opens tape DCB.

PRMPT (CZATJ) -- Issues a message to the user.

READ/WRITE (CZCRAS) -- Reads /Writes to tape.

RELEASES (CZCAD) -- Releases disk device.

RELEASE (CZAFJ) -- Releases tape device.

TSU Module (CZUBC) -- Initiates a background task.

XWTO (CZABQ) -- Writes messages to operator.

Exits: Control is returned to the caller with a completion message, and one of the following hexadecimal return codes, in general register 15.

00 -- normal completion
04 -- parameter missing
08 -- error in a parameter
0C -- duplicate volume serial numbers
10 -- tape to tape DMPRST attempted
14 -- 2314-2311 DMPRST attempted
18 -- CZUBC failed
1C -- attention interruption
20 -- error in GETMAIN
24 -- error in DDEF or MTREQ
28 -- device has invalid format
2C -- I/O error (tape SYNAD or disk PGOUT)
30 -- no available page or more than 96 relocations

Operation: On entry, DMPRST makes repeated calls to the GETVAL subroutine for all parameters; it validates the parameters; and, it sets up a DDEF command for tape; or it calls MTREQ for disk mounting. When all parameters have been verified, DMPRST passes control to CZUBC if the request is to be processed in a new background task, or to CZUFAD, a CSECT within a DMPRST, if the request is to be processed in this task. CZUBC creates a VISAM data set from the command string passed by DMPRST, executes that command string to create a background task that will reenter DMPRST for processing, and return control of the current task to DMPRST for final cleanup and return operations. On return from CZUFAD or CZUBC, DMPRST prompts with a completion message, loads the return code in register 14, and returns control to the user.

CZUFAD, the disk/tape I/O CSECT of CZUFA, sets up I/O parameters and calls GETMAIN for buffer space; it distributes buffer space for read/write operations involving tape volumes; and, it defines any required tape devices. One of three processing paths is now chosen, depending on

the requested operation: restore from tape; dump to tape; disk to disk operation.

• For a restore operation, from tape to disk, DMPRST performs the following:

1. Opens the tape data set for a restore operation.
2. Reads the PAT, IPL, and label.
3. Calls MTREQ to mount the disk.
4. Obtains the volume label.
5. Puts the PAT SDA and RPN in PGOUT list.
6. Reads all PAT pages from the disk into buffers.
7. Validates the PAT and resolves any relocations.
8. Clears from PAT error pages; writes X'FF's on the PAT.
9. Marks and relocates error pages.
10. Relocates the PAT if the to-volume RPN = a new PAT page.
11. Reads data pages into two buffers and writes them to disk.
12. Issues completion message.
13. Releases disk; closes and releases tape.
14. Frees buffer space and returns.

• For a dump operation, from a disk to tape, DMPRST performs the following:

1. Opens the tape data set for a dump operation.
2. Calls MTREQ to mount the disk.
3. Reads the volume label.
4. Puts the PAT SDA and RPN in PGOUT list.
5. Calls SETXP to read all PAT pages from disk.
6. Resolves any relocations and puts error flags on cylinder 199.
7. Writes PAT, IPL, labels, and dummy IPL text.
8. Writes data pages to tape from two buffers.

9. Issues completion message.
 10. Releases disk; closes and releases tape.
 11. Frees buffer space and returns.
- For a disk to disk operation, DMPRST performs the following:
 1. Calls MTREQ to mount both disks.
 2. Reads volume labels.
 3. Calls SETXP to read all PAT pages.
 4. Resolves relocations.
 5. Puts X'FF's in the new PAT.
 6. Marks and relocates error pages in new PAT.
 7. Moves PAT if TOVOL RPN = a new page.
 8. Writes the label.
 9. Calls PGOUT to write all pages to disk.
 10. Issues completion messages.
 11. Releases disk devices.
 12. Frees buffer space and returns.

Error Conditions: DMPRST returns control with an error message and error return code for any of the following error conditions:

- Missing or invalid parameter.
- Duplicate volume serial numbers.
- Tape to tape, or 2311-2314 operation requested.
- Failure in a called module or macro.
- I/O error.
- No available page or more than 96 relocations.

System Control Block Usage:
Data Control Block (CHADCB)

Data Event Control Block (CHADEC)

Interrupt Storage Area (CHAISA)

Symbolic Device Allocation Table (CHASDA)

Task Common (CHATCM)

Task Data Definition Table (CHATDT)

Time Shared Utility Module (CZUBC)

The Time Shared Utility Module initiates background tasks for the PATCLEAR and DUMP/RESTORE modules. (See Chart ZD.)

Entry Points: CZUBC1 -- Main entry.

Modules Called:

- CLOSE (CZCLBC) -- Closes a DCB.
- DDEF (CZAEA) -- Defines a VISAM data set.
- ERASE (CZAEJ) -- Erases a data set.
- LOCATE (CZCFL) -- Locates a non-unique dsname.
- OPEN (CZCLA) -- Opens a DCB.
- RELEASE (CZAFJ) -- Releases JFCB storage.

Exits: Control returns to the appropriate command processor (PATCLEAR or DMPRST) with one of the following return codes in general register 15:

- 00 -- normal return.
- 04 -- RELEASE failed, job terminated.
- 08 -- unique dsname was not generated, job terminated.
- 0C -- DDEF failed, job terminated.
- 10 -- attention interruption, job terminated.
- 14 -- input parameter count invalid, job terminated.
- 18 -- invalid parameter, job terminated.
- 1C -- I/O error during PUT operation to VISAM data set, job terminated.

Operation: On entry, this module obtains the userid, charge number, and password from task common (CHATCM); it moves these to an area for the LOGON command. Next, after establishing a retry limit of 100, it calls GENDDN to generate unique dd/ds names. GENDDN functions as follows:

1. Obtains the value of TCMDN.
2. Increments and restores this value.
3. Converts this value to a five-byte EBCDIC value.
4. Appends a three-byte prefix (TSU).
5. Passes the eight-byte value to the mainline processor.

The mainline routine then moves the five-byte numeric portion of the generated name to FULQLNAM, for LOCATE, and makes a series of calls to the following subroutines:

1. CHKDSN -- Calls LOCATE to make sure that the generated dsname is unique.
2. DDEF10 -- Performs the DDEF for the VISAM data set.
3. OPENDCB -- Checks for an attention interruption.
4. BLDREC -- Builds a VISAM data set from command parameters.
5. CLOSEDCB -- Closes the VISAM data set.
6. RELEASE -- Releases the JFCB storage.

The mainline routine now executes an OBEY macro instruction to execute the VISAM data set, loads the appropriate return code in general register 15, returns control to the caller. If release operations fail, the data set is erased prior to return.

The CHKDSN subroutine calls LOCATE to find the generated dsname. When LOCATE fails to find the name, and therefore the name is unique, the numeric portion is put in the dsname and control is returned to the mainline processor. If the name is not unique, GENAME is repeatedly called to generate a unique dsname. When this fails, control returns to the caller with an error code in register 15.

The DDEF10 subroutine calls DDEF for a VISAM data set and, if the DDEF is successful, returns control to the mainline routine. When DDEF returns an error code because the ddname is not unique, DDEF10 attempts to generate a unique name, via GENAME. Now, if a unique name is generated, DDEF10 reenters itself to issue a new DDEF. If a unique name is not generated, or if DDEF indicates any other error condition, control returns to the caller with an error code.

The OPENDCB subroutine uses ATPOL to test for any pending attention interruptions and, if one exists, sets an appropriate return code, releases the JFCB storage, and, returns control to the caller. If no attention interruption is pending, OPENDCB calls OPEN to open the DCB, and returns control to the mainline routine.

The BLDREC subroutine first builds a LOGON command as a VISAM record; it uses PUT to write this record to a VISAM data set. The command string, passed to CZUBC from PATCLEAR or DMPRST, is written as a VISAM record on the data set. Now RET and LOGOFF

commands are written, as VISAM records, at the end of the data set. Control returns to the mainline routine.

The CLOSEDCB subroutine calls CLOSE to close the DCB and it returns control to the mainline routine.

The RELEASE subroutine calls REL to release the JFCB storage and returns control to the mainline routine.

Error Conditions: Error codes are returned to the caller for any of the following errors:

- Failure in any called module or macro instruction.
- Failure to generate unique ds/dd names.
- Invalid parameter or parameter count.
- I/O error during the PUT operations.

System Control Block Usage:
Data Control Block (CHADCB)

Interrupt Storage Area (CHAISA)

Task Common (CHATCM)

PATFIX Routine (CZUPF)

The PATFIX routine checks the Page Assignment Tables for each volume specified in the VOLDEF parameter. The routine then builds various diagnostic reports which reflect any errors found in this initial PAT scan, builds duplicate PAT tables from the scan of DSCB and data pages on each volume, and allows the user to replace an incorrect PAT with the newly created duplicate PAT. (See Chart ZE.)

Entry Points: CZUPF1 -- Main entry.

Modules Called:

CHECKDS (CZAAC) -- Validates all dsnames.

CLOSE (CZCIB) -- Closes report data set DCB.

DDEF (CZAEA) -- Defines an MSAM printer.

FINDDS (CZAEC) -- Gets the JFCB for the report data set.

FINISH (CZCMH) -- Ends MSAM processing.

FREEMAIN (CZCHA3) -- Releases virtual storage.

GETMAIN (CZCGA2) -- Gets storage for internal tables.

LOCKPAT (CZCEJ) -- Locks Page Assignment Tables in public storage when FIX is on.

MTREQ (CZCAA) -- Mounts private volumes.
OPEN (CZCLA) -- Opens report data set DCB.
PAUSE (CZCAC) -- Dismounts and mounts private volumes on the same device.
PRMPT (CZATJ1) -- Writes messages to terminal.
RELEAS (CZCAD) -- Releases private volumes.
VAMINIT (CZCEQ) -- Initializes SDAT entries for private volumes.

Exits: Control is returned via type-1 linkage. If an error occurs, a diagnostic message is issued and control returns via type-1 linkage.

Operation: On entry at CZUPF1 the input parameters are checked. Then internal tables are set up to monitor information on PATs that are scanned.

The PATs are first scanned for correct entries: the first four entries, and cylinder 199, must be flagged as C0; the 7F and FF entries must be correctly located. Relocation errors are found, and the PAT page error report is created.

Next, each DSCB page on the volume is scanned, and a report on each data set is created. This report reflects invalid format-E dsnames and invalid checksums. During this scan, a duplicate PAT is built for each specified volume. Then, if there are shared pages, these are scanned to determine which data sets claim them.

When the duplicate PATs are built for the volume, the routine compares the old and new PATs; unreferenced format-F DSCBs are found as are allocated but unreferenced data pages and DSCB page-slot-count errors. A PAT comparison report is built to reflect these errors; a data set profile report is also created.

After these scans have been completed, the user may replace the old PATs with the newly created versions; he may also correct checksum errors, zero unreferenced format-F DSCBs, and zero unused data page DSCB slots.

Error Conditions: This routine writes an error message on SYSOUT for each error found. These messages can be found in System Messages.

System Control Block Usage:

Format E DSCB (CHADSE)

Format F DSCB (CHADSF)

Interrupt Storage Area (CHAISA)

Public Volume Table (CHAPVT)

Symbolic Device Allocation Table (CHASDA)

Task Common (CHATCM)

Task Data Definition Table (CHATDT)

User Table (CHAUSE)

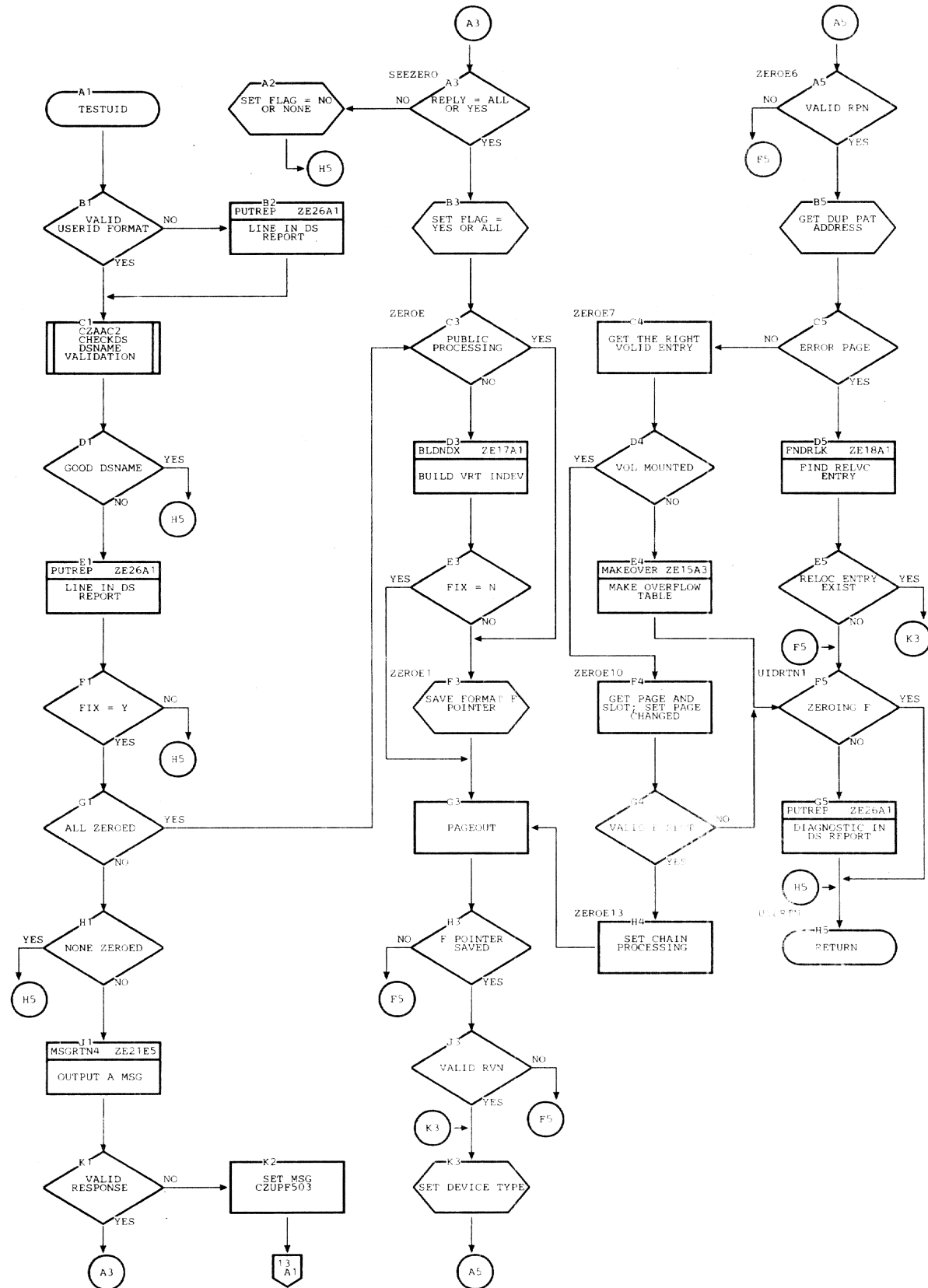
Program Logic Manual

GY28-2047-4

Operator Task and Bulk I/O

Flowcharts on pages 51-230 were not scanned.

Chart ZE. PATFIX Routine - CZUPF (Page 28 of 28)



APPENDIX B: EXTERNAL CALLED MODULES REFERENCED TO PLMS

Module		ACCESS METHODS	SYSTEM SERVICE	SUPERVISOR	COMMAND SYSTEM	TASK MONITOR
ID	Name					
CEAMC	CRTSI			X		
CEAP7	AWAIT			X		
CEAQ5	VSEND			X		
CEARQ	TWAIT			X		
CEAS2	XTRSYS			X		
CEAS6	REDTIM			X		
CEAT2	SCRISI			X		
CEA32	SETSIS			X		
CZAAB3	GATWR				X	
CZAACB	CHKNM				X	
CZAAC1	NEXTPAR				X	
CZAAC2	CHEKDS				X	
CZAAD2	MSGWR				X	
CZAC3	ALFNUM				X	
CZACP1	ABEND				X	
CZACS1	PAIR				X	
CZAEA4	DDEF				X	
CZAEB1	FINDJFCB				X	
CZAEC1	FINDDS				X	
CZAEI2	CATALOG				X	
CZAEJ3	ERASE				X	
CZAFJ3	RELEASE				X	
CZAFM2	LOGON				X	
CZAFN1	LOGOFF				X	
CZASA1	OBEG				X	
CZASDX	GDV				X	
CZATC	GATE				X	
CZATD2	VMTI-2				X	
CZATJ	PROMPT				X	
CZAYE1	SARD				X	
CZAYF1	EXHIBIT				X	
CZCAA	MTREQ		X			
CZCAC	PAUSE		X			
CZCAD	RELEAS		X			
CZCEJ	LOCKPAT		X			
CZCEQ	VAMINIT		X			
CZCFA1	ADDCAT		X			
CZCFI1	INDEX		X			
CZCFL1	LOCATE		X			
CZCFO	OBTAIN/ RETAIN		X			
CZCFX	CATFLUSH		X			
CZCGA2	GETMAIN		X			
CZCHA3	FREEMAIN		X			
CZCJDA	DIR					X
CZCJIA	INTINQ					X
CZCJQS	QLE					X
CZCJSA	SIR					X
CZCLA	OPEN	X				
CZCLBC	CLOSE	X				
CZCMA	GETBUF	X				
CZCMD1	SETUR	X				
CZCMH1	FINISH	X				
CZCNA	FREEBUF	X				
CZCOS	PUT	X				
CZCOT	SETL	X				
CZCPB	GET	X				
CZCPC	VISAM SETL	X				
CZCPD	ESETL	X				
CZCPE	READ/WRITE	X				
CZCPH	DELREC	X				
CZCRA	READ	X				
CZCRC	CHECK	X				

APPENDIX C: MODULE DIRECTORY

Entry Point	Routine Name	Module	CSECT	PSECT	PLM Refs	
					Sect	Chart
CZABA1	General Control Processor	CZABA	CZABAV	CZABAR	2	DD
CZABA2	EXECUTE Subprocessor	CZABA	CZABAV	CZABAR	2	DD
CZABA3	CANCEL Subprocessor	CZABA	CZABAV	CZABAR	2	DD
CZABA4	BULKIO Subprocessor	CZABA	CZABAV	CZABAR	2	DD
CZABA5	BACK Subprocessor	CZABA	CZABAV	CZABAR	2	DD
CZABA6	Timer Interrupt	CZABA	CZABAV	CZABAR	2	DD
CZABA7	LOGOFF Subprocessor	CZABA	CZABAV	CZABAR	2	DD
CZABA8	DIRECT Subprocessor	CZABA	CZABAV	CZABAR	2	DD
CZABA9	SHUTDOWN Subprocessor	CZABA	CZABAV	CZABAR	2	DD
CZABAB	Batch Monitor Processor	CZABA	CZABAV	CZABAR	2	DD
CZABAC	EODAD	CZABA	CZABAV	CZABAR	2	DD
CZABAD	SYNAD	CZABA	CZABAV	CZABAR	2	DD
CZABAE	Entry from Batch Wait	CZABA	CZABAV	CZABAR	2	DD
CZABAF	MOCP Interrupt Flag	CZABA	CZABAV	CZABAR	2	DD
CZABAQ	EXHIBIT BWQ DCB	CZABA	CZABAV	CZABAR	2	DD
CZABD1	DATASET Routine	CZABD	CZABDC	CZABDR	3	EE
CZABD2	RT command Routine	CZABD	CZABDC2	CZABDR	3	EE
CZABD3	PR macro Routine	CZABD	CZABDC	CZABDR	3	EE
CZABD4	PRINT command Routine	CZABD	CZABDC	CZABDR	3	EE
CZABD6	PUNCH command Routine	CZABD	CZABDC2	CZABDR	3	EE
CZABD7	PU macro Routine	CZABD	CZABDC2	CZABDR	3	EE
CZABD8	WT command Routine	CZABD	CZABDC2	CZABDR	3	EE
CZABD9	WT macro Routine	CZABD	CZABDC2	CZABDR	3	EE
CZABDA	RJE SYNAD exit	CZABD	CZABDC2	CZABDR	3	EE
CZABDB	RJE EODAD exit	CZABD	CZABD2	CZABDR	3	EE
CZABF1	RT Processor	CZABF	CZABFV	CZABFR	5	UU
CZABF2	RT Tape Data Set EODAD	CZABF	CZABFV	CZABFR	5	UU
CZABF3	RT Tape Data Set SYNAD	CZABF	CZABFV	CZABFR	5	UU
CZABF4	RT VAM Data Set SYNAD	CZABF	CZABFV	CZABFR	5	UU
CZABG1	PRINT Processor	CZABG	CZABGV	CZABGR	5	WW

Entry Point	Routine Name	Module	CSECT	PSECT	PLM Refs	
					Sect	Chart
CZABG2	PRINT Input Data Set EODAD	CZABG	CZABGV	CZABGR	5	WW
CZABG3	PRINT Input Data Set SYNAD	CZABG	CZABGV	CZABGR	5	WW
CZABG4	PRINT Output Data Set SYNAD	CZABG	CZABGV	CZABGR	5	WW
CZABH1	PUNCH Processor	CZABH	CZABHV	CZABHR	5	XX
CZABH2	PUNCH Input Data Set EODAD	CZABH	CZABHV	CZABHR	5	XX
CZABH3	Input Data Set SYNAD	CZABH	CZABHV	CZABHR	5	XX
CZABH4	Output Data Set SYNAD	CZABH	CZABHV	CZABHR	5	XX
CZABI1	WT Processor	CZABI	CZABIV	CZABIR	5	YY
CZABI2	WT Input Data Set EODAD	CZABI	CZABIV	CZABIR	5	YY
CZABI3	Input Data Set SYNAD	CZABI	CZABIV	CZABIR	5	YY
CZABI4	Output Data Set SYNAD	CZABI	CZABIV	CZABIR	5	YY
CZABK1	ASNBD Command Processor	CZABK	CZABKC	CZABKR	5	PP
CZABK2	ASNBD from MOHR at STARTUP	CZABK	CZABKC	CZABKR	5	PP
CZABL1	TWAIT in Batch Monitor	CZABA	CZABL	none	2	DD
CZABL2	Batch Wait Interrupt Flag	CZABA	CZABL	none	2	DD
CZABQ1	XWTO Processor	CZABQ	CZABQV	CZABQR	6	ZA
CZABX1	LABEL Command Processor	CZABX	CZABXV	CZABXR	5	RS
CZABXX	BPKD Entry	CZABX	CZABXV	CZABXR	5	RS
CZACA1	MOCF Main Entry	CZACA	CZACAV	CZACAR	2	BB
CZACA2	SYSLOG DCB	CZACA	CZACAS	CZACAR	2	BB
CZACA3	REPLY Processor	CZACA	CZACAW	CZACAR	2	BB
CZACA4	REPLY Prompter	CZACA	CZACAV	CZACAR	2	BB
CZACA5	PUT-IN-LOG Routine	CZACA	CZACAY	CZACAR	2	BB
CZACB2	MOHR	CZACB	CZACBC	CZACBR	2	AA
CZACB3	MOHR	CZACB	CZACBC	CZACBR	2	AA
CZACF1	MSG Processor	CZACF	CZACFV	CZACFR	5	SS
CZACF2	BCST Processor	CZACF	CZACFV	CZACFR	5	SS
CZACF3	MSG/BCST SYNAD	CZACF	CZACFV	CZACFR	5	SS
CZACF4	MSG/BCST EODAD	CZACF	CZACFV	CZACFR	5	SS
CZACG1	FORCE Processor	CZACG	CZACGC	CZACGR	5	QQ
CZACG2	FORCE SYNAD	CZACG	CZACGC	CZACGR	5	QQ
CZACN1	SHUTDOWN Processor	CZACN	CZACNV	CZACNR	5	VV
CZACN2	SHUTDOWN EODAD	CZACN	CZACNV	CZACNR	5	VV

Entry Point	Routine Name	Module	CSECT	PSECT	PLM Refs	
					Sect	Chart
CZACN3	Batch Monitor SHUTDOWN entry	CZACN	CZACNV	CZACNR	5	VV
CZAWA1	ABEND Recovery	CZAWA	CZAWAC	none	4	OO
CZAWC1	FORTTRAN Data Set Conversion	CZAWC	CZAWCP	none	4	KL
CZAWM1	Message Handler (and common messages)	CZAWM	CZAWMC	none	4	NN
CZAWM2	Uncommon messages	CZAWM	CZAWMC2	none	4	NN
CZAWN1	Write Message to ACK	CZAWN	CZAWNC	none	4	OA
CZAWN2	Write ACK Messages to RJE	CZAWN	CZAWNC	none	4	OA
CZAWN3	EODAD	CZAWN	CZAWNC	none	4	OA
CZAWN4	SYNAD	CZAWN	CZAWNC	none	4	OA
CZAWS1	Initialization	CZAWS	CZAWSC	none	4	FF
CZAWS2	ASNBD Message	CZAWS	CZAWSC	none	4	FF
CZAWS3	Add ABENDED Device	CZAWS	CZAWSC	none	4	FF
CZAWS4	ABEND on CZAWS	CZAWS	CZAWSC	none	4	FF
CZAWT1	Control Main Entry	CZAWT	CZAWTC	none	4	GG
CZAWT2	Control ICB/SIR	CZAWT	CZAWTC	none	4	GG
CZAWU1	Attend Alert	CZAWU	CZAWUC	none	4	II
CZAWU2	Attend Alert EODAD	CZAWU	CZAWUC	none	4	II
CZAWU3	Attend Alert SYNAD	CZAWU	CZAWUC	none	4	II
CZAWU4	Attend Alert ABEND	CZAWU	CZAWUC	none	4	II
CZAWV1	Input Closeout	CZAWV	CZAWVC	none	4	LL
CZAWW1	Master Service	CZAWW	CZAWWC	none	4	HH
CZAWW2	Continue after ABEND	CZAWW	CZAWWC	none	4	HH
CZAWW6	Return from CZAWU	CZAWW	CZAWWC	none	4	HH
CZAWW9	Return from CZAWY/CZAWZ	CZAWW	CZAWWC	none	4	HH
CZAWWA	Error return from CZAWU	CZAWW	CZAWWC	none	4	HH
CZAWX1	Asynchronous Interrupts	CZAWX	CZAWXC	none	4	JJ
CZAWX2	Initialization Service	CZAWX	CZAWXC	none	4	JJ
CZAWX3	Delete ICB	CZAWX	CZAWXC	none	4	JJ
CZAWX4	Return to Task Monitor	CZAWX	CZAWXC	none	4	JJ
CZAWY1	Output Service	CZAWY	CZAWYC	none	4	MM
CZAWY2	VAM SYNAD	CZAWY	CZAWYC2	none	4	MM
CZAWY3	VAM EODAD	CZAWY	CZAWYC2	none	4	MM

<u>Entry Point</u>	<u>Routine Name</u>	<u>Module</u>	<u>CSECT</u>	<u>PSECT</u>	<u>PLM Refs</u>	
					<u>Sect</u>	<u>Chart</u>
CZAWY4	Output Cleanup	CZAWY	CZAWYC2	none	4	IM
CZAWY5	MCB Area	CZAWY	CZAWYC2	none	4	MM
CZAWY6	VAM ddname hold area	CZAWY	CZAWYC	none	4	MM
CZAWY7	VAM JFCB pointer	CZAWY	CZAWYC	none	4	MM
CZAWZ1	Input Service	CZAWZ	CZAWZC	none	4	KK
CZAWZ2	VISAM PUT SYNAD	CZAWZ	CZAWZC	none	4	KK
CZAWZ3	RJE Disconnect	CZAWZ	CZAWZC	none	4	KK
CZCM04	DROP Processor	CZCM0	CZCM0Z	CZCM0P	5	RR
CZCM07	HOLD Processor	CZCM0	CZCM0Z	CZCM0P	5	RR
CZCM0E	I/O Completion	CZCM0	CZCM0Z	CZCM0P	5	RR
CZUBC1	TSU MODULE	CZUBC	CZUBCC	CZUBCP	7	ZD
CZUFA1	DMPRST	CZUFA	CZUFAC	CZUFAP	7	ZC
CZUFA2	DMPRST	CZUFA	CZUFAC	CZUFAP	7	ZC
CZUFAB	DMPRST	CZUFA	CZUFAC	CZUFAP	7	ZC
CZUFAD	DMPRST	CZUFA	CZUFAD	CZUFAP	7	ZC
CZUFO1	PATCLEAR	CZUFO	CZUFOC	CZUFOP	7	ZB
CZUFO2	PATCLEAR	CZUFO	CZUFOC	CZUFOP	7	ZB
CZUFO3	PATCLEAR	CZUFO	CZUFOC	CZUFOP	7	ZB
CZUPF1	PATFIX	CZUPF	CZUPFC	CZUPFP	7	ZE

APPENDIX D: ACTIVE REMOTE TABLE

The Active Remote Table (ART) is created, using GETMAIN, by the Batch Monitor Initialization routine at CZABA1. Each ART entry (16 bytes) contains the userid, BSN, CANCEL flag, and taskid for a BACK job. BACK jobs are assigned a BSN from 1 to 255, and are queued on the ART; normal batch jobs are queued on the BWQ, with a BSN greater than 255. The following figure illustrates the Active Remote Table.

<-----8 bytes----->	<-----4 bytes----->	<--1 byte-->	<--1 byte-->	<---2 bytes--->
USERID	BSN	UNUSED	CANCEL FLAG	TASKID

APPENDIX E: OPERATOR TASK AND BULKIO MODULES USING RCR MACROS

Module ID	RCR Macro Used	Purpose
CZABC	UPDATE	To update accounting fields in user table entry.
	RATION	To request resources for the Background task,
	VACATE	To remove allocated resources if BACK request not accepted by Batch Monitor.
CZABF	RATION	To account for # records read from tape for the user.
CZABG	RATION	To account for # records printed from user's private volume.
CZABH	RATION	To account for # records punched from user's private volume.
CZABI	RATION	To account for # records written on tape for the user.
CZAWA	RATION	To account for # records read or printed/punched for the user by the BULKIO task if not done by CZAWV or CZAWY.
	CLOSE	If ration is done by CZAWA or if CLOSE was not done by CZAWX, CZAWV, or CZAWY.
CZAWU	OPEN	To validate userid and batch privileges for printing or punching by the BULKIO task.
CZAWV	RATION	To account for # records read for the user by the BULKIO task.
	CLOSE	To update user table and close user table entry.
CZAWX	OPEN	To validate userid and batch privileges for reading cards by the BULKIO task.
	CLOSE	To update user table and close user table entry if user not allowed card read privileges.
CZAWY	RATION	To account for # records printed or punched by the BULKIO task.
	CLOSE	To update user table and close user table entry.

INDEX

%ENDDS card 22

- Abnormal task termination 1
- ABORT codes (CZAWY) 25
- ACK (see Acknowledgment Data Set)
- Acknowledgment Data Set (ACK) 16,22,26
 - Member Validation 26
 - Messages 26
 - Printing 18
- Action Message 6
- Activate BULKIO task 4
- Active Remote Task table (ART) 9
- Active table entry 15
- Add a label 34
- Arrows (Action Message) 6
- ART (Active Remote Task) table 9
- ASCII code 25,36,38,41,42
- ASNBD 16
- ASNBD Routine (CZABK) 30
 - Entry points 30-31
 - Modules called 31
 - Exit 31
 - Operation 31
 - Error conditions 31
 - System control block usage 31
 - Flowchart 151-155
- ATPOL 49
- Attention Interruptions (TSU) 46,47,49

- BACK command 1
- Background subprocessor 10
- Background Task creation 45,47
- Batch initiation 30
- Batch Monitor (CZABA) 7
 - General operations 7
 - Entry points 8-9
 - Modules called 9
 - Exit 9
 - Operation 9-11
 - Error conditions 11
 - System control block usage 11
 - Flowchart 57-73
- Batch Monitor taskid 14
- Batch Monitor wait routine 11
- Batch sequence number 7
- BCST (see MSG/BCST) 34
- Binary serial day 4
- BLDREC subroutine 49
- BPRD (TSU) 45
- BsN (batch sequence number) 7
- Build a VISAM record 49
- Build an output data set 22
- BULTIN (TSU) 45
- Bulk input/output commands 1
- BULKCOMM table (CHABCT) 15
- BULKIO ACK Data Set Handler (CZAWN) 26-27
 - Entry Points 26
 - Modules called 26
 - Exit 26
 - Operation 26-27
 - System control block usage 27
 - Flowchart 132-135
- BULKIO Abend Recovery (CZAWA) 27
 - Entry points 27
 - Modules called 27
 - Subroutines 27
 - Exit 27
 - Operation 27
 - Flowchart 136-150
- BULKIO Attend Alert (CZAWU) 18-19
 - Entry points 18
 - Modules called 18
 - Exit 18
 - Operation 19
 - Flowchart 98-104
- BULKIO Control Routine (CZAWT) 17
 - Entry points 17
 - Modules called 17
 - Exit 17
 - Operation 17
 - System control block usage 17
 - Flowchart 96
- BULKIO Initialization Routine (CZAWS) 16-17
 - Entry points 16
 - Modules called 16
 - Exit 16
 - Operation 16-17
 - System control block usage 17
 - Flowchart 89-95
- BULKIO Input Closeout (CZAWV) 23-24
 - Entry points 23
 - Modules called 23
 - Exit 23
 - Operation 24
 - System control block usage 24
 - Flowchart 120-122
- BULKIO Input Service (CZAWZ) 21-23
 - Entry points 21
 - Modules called 21-22
 - Exit 22
 - Operation 22-23
 - System control block usage 23
 - Flowchart 109-118
- BULKIO Input Start (CZAWX) 19-21
 - Entry points 20
 - Modules called 20
 - Exit 20
 - Operation 20-21
 - System control block usage 21
 - Flowchart 105-108
- BULKIO Master Service (CZAWW) 17-18
 - Entry points 18
 - Modules called 18
 - Exit 18
 - Operation 18
 - System control block usage 18
 - Flowchart 97
- BULKIO Message Handler (CZAWM) 26
 - Entry points 26
 - Modules called 26
 - Exit 26

- Operation 26
- System control block usage 26
- Flowchart 131
- BULKIO Output Service (CZAWY) 24-26
 - Entry points 24
 - Modules called 24
 - Exit 24
 - Operation 24-25
 - ABORT codes 25
 - System control block usage 25
 - Flowchart 123-130
- BULKIO Preprocessor (CZABD) 13-14
 - General operations 13
 - Entry points 13
 - Modules called 13-14
 - Executive routines 13
 - Operation 14
 - Error conditions 14
 - System control block usage 14
 - Flowchart 73-88
- BULKIO Preprocessor taskid 14
- BULKIO recovery procedures 27
- BULKIO task 15

- Card-read cleanup operations 23
- Card-reader deletion/reassignment 27
- Catalog interlocks 29
- CAWRLOG (CZACA5) 7
- CHABCT (BULKCOMM) table 15,18
- CHAOPH (Operator Header) 6
- CHKDSN subroutine 49
- Close a card reader 24
- CLOSEDCB subroutine 49
- CNPAR 14
- Combine-reader-and-punch option 19
- Command acceptance message 14
- Command Analyzer and Executor 1,30
- Command routines 30
- Command string 46,47
- Configuration message 4
- Conversational batch request 2
- Conversational task termination 32
- Creation of Operator task 1
- Cycling the BULKIO task 15
- CZABA (see Batch Monitor) 7-12
- CZABD (see BULKIO Preprocessor) 13-14
- CZABF (see RT) 38-39
- CZABG (see PRINT) 35-37
- CZABH (see PUNCH) 37-38
- CZABI (see WT) 40-42
- CZABK (see ASNBD) 30-31
- CZABQ (see XWTO) 43-44
- CZABX (see LABEL) 34
- CZACA (see Main Operator Communications Program) 6-7
- CZACB (see Main Operator's Housekeeping Routine) 3-6
- CZACF (see MSG/BCST) 34-35
- CZACG (see FORCE) 32
- CZACN (see SHUTDOWN) 39-40
- CZAWA (see BULKIO ABEND Recovery) 27
- CZAWC (see FORTRAN Data Set Conversion) 23
- CZAWM (see BULKIO Message Handler) 26
- CZAWN (see BULKIO ACK Data Set Handler) 26-27
- CZAWS (see BULKIO Initialization Routine) 16-17

- CZAWT (see BULKIO Control Routine) 17
- CZAWU (see BULKIO Attend Alert) 18-19
- CZAWV (see BULKIO Input Closeou) 23-24
- CZAWW (see BULKIO Master Service) 17-18
- CZAWX (see BULKIO Input Start) 19-21
- CZAWY (see BULKIO Output Service) 24-26
- CZAWZ (see BULKIO Input Service) 21-23
- CZCM0 (see HOLD/DROP) 32-33
- CZUBC (see TSU Module) 48-49
- CZUFA (see DMPRST) 46-48
- CZUFO (see PATCLEAR) 45-46
- CZUPF (see PATFIX) 49-50

- DATASET card 13,22
- DDEF10 subroutine 49
- Defaulted parameter 14
- Delete a label 34
- Device availability 31
- DIRECT command (CZABA8) 9,11
- Disk/tape I/O CSECT (CZUFAD) 47
- Disk to disk dump 48
- Disk to tape dump 47-48
- DMPRST processor (CZUFA) 46-48
 - Entry points 46
 - Modules called 46
 - Exits 47
 - Operation 47-48
 - Error conditions 48
 - System control block usage 48
 - Flowchart 193-200
- DROP (see HOLD/DROP) 32-33
- Dump operation
 - Disk to tape 47-48
 - Disk to disk 48
- DUMP/RESTORE (see also DMPRST) 46

- Edit control-character validation 24
- Edit handling 36
- Erase unused SYSIN and SYSOUT 10
- EXECUTE command 1
- Execute subprocessor 10
- Execute a VISAM data set 49
- Express Batch 2
- Express Batch extended processing 23
- Express Batch processing 20,21
- Express mode flag 24

- FORCE routine (CZACG) 32
 - Entry points 33
 - Modules called 33
 - Exit 33
 - Operation 33
 - Error conditions 33
 - System control block usage 33
 - Flowchart 155
- FORTAN control character 42
- FORTAN Data Set Conversion (CZAWC) 23
 - Entry points 23
 - Modules called 23
 - Exits 23
 - Operation 23
 - System control block usage 23
 - Flowchart 119

GENDDN subroutine 48
 Generate unique dd/ds names 48
 GETVAL subroutine 47

High priority message 6
 HOLD/DROP routine (CZCM0) 32-33
 Entry points 32
 Modules called 32
 Exit 32
 Operation 32-33
 System control block usage 33
 Flowchart 155

Independent batch operation 30
 Initial page positioning 41
 Initial print line 36
 Initialization for Operator's task 3
 Initialization of SYSLOG data set 4
 Initialization Recovery (WAINIT) 29
 Entry points 29
 Modules called 29
 Subroutines 29
 Operation 29
 Initialization processor 9,10
 Initialize Batch Monitor 4
 Initiating nonconversational tasks 7
 INITL 14
 Input job assignment 15
 Installation-written command procedures 19
 Interrupt-driven BULKIO 17
 Interruption, attention 46,47,49

JFCB storage, release 49
 Job 15

LABEL Routine (CZABX) 34
 Entry points 34
 Modules called 34
 Exit 34
 Operation 34
 Error conditions 34
 System control block usage 34
 Flowchart 161-164
 Line data set creation 39
 LOGOFF subprocessor 11

Main Operator 3
 Main Operator Control 1
 Main Operator Communications Program
 (MOCP-CZACA) 6-7
 Entry points 6
 Modules called 6
 Subroutines 6
 Exit 6
 Operation 6-7
 Error conditions 7
 System control block usage 7
 Flowchart 53-56
 Main Operator Housekeeping Routine
 (MOHR-CZACB) 3-6
 Entry points 3
 Modules called 3
 Exit 4
 Operation 4-6
 Error conditions 6
 System control block usage 6
 Flowchart 52
 Machine code, control
 characters 25,36,38,42
 Macro Instruction handling 43
 Message Control Block (MCB) 6,44
 Message Length (ASNBD) 16
 Message pieces 26
 MOCP (see Main Operator Communications
 Program) 6-7
 MOHR (see Main Operator Housekeeping
 Routine) 3-6
 MSAM inhibit option 19
 MSAM inhibit recovery (WABE) 27-28
 Entry points 27
 Modules called 27
 Subroutines 27-28
 MSAM Output Recovery (WAMO) 28-29
 Entry points 28
 Modules called 28
 Subroutines 28
 Operation 28-29
 MSAM recovery 28
 MSG/BCST routine (CZACF) 34-35
 Entry points 34
 Modules called 34-35
 Exit 35
 Operation 35
 Error conditions 35
 System control block usage 35
 Flowchart 165-166

New background task creation 45
 Nonconversational operations 1
 Nonconversational RUNMODE 46
 Normal task termination 1
 Not-ready-to-ready 15

OPENDCB subroutine 49
 OBEY 45
 Operator Action Message 6
 Operator header (CHAOPH) 6
 Operator replies 6
 Operator task and Batch Monitor 3
 Operator task general operation 3
 Output accounting 25
 Output job assignment 15
 Output of undesired VAM records 19
 Overdue replies 6,7

Packing SYSINS 23
 Page Assignment Table (PAT) 45
 Page handling 25
 Page numbering 25
 Parameter processing routines 13,14
 PAT 45
 PATCLEAR Processor (CZUFO) 45-46
 Entry points 45
 Modules called 45-46
 Exits 46
 Operation 46
 Error conditions 46
 System control block usage 46

Flowcharts 190-192
 PCS 'SET' statements 19
 Pending attention interruptions 49
 PR macro instruction 43
 PRINT routine (CZABG) 35-37
 Entry points 35
 Modules called 35-36
 Exit 36
 Operation 36
 Error conditions 36-37
 System control block usage 37
 Flowchart 174-180
 Printer spacing 19
 Printing ACK 18
 Private BWQ entries 9
 Private device entries 10
 PU macro instruction 43
 Public BWQ entries 9
 Public unit record I/O 15
 Public VAM I/O 19
 PUNCH routine (CZABH) 37-38
 Entry points 37
 Modules called 37
 Exit 37
 Operation 38
 Error conditions 38
 System control block usage 38
 Flowchart 171-173
 PUT record in VISAM data set 49

 RCR OPEN 19,21
 RCR RATION 24,36,38,39,41
 Read-cards job 19,20
 Recovery procedures (BULKIO) 27
 Release a card reader 24
 Release JFCB storage 49
 RELEASE subroutine 49
 Remote Job Entry (RJE) 15,19,22
 Devices 26,28,30-31
 Line 21
 Recovery 27
 S-entry (CZAWS5) 16,18,20
 Station ID 18,22-23,26
 REPLY elapsed time exceeds maximum 7
 Reply queue 6
 Reply to Operator task 6
 Restore BULKIO task 28
 RESTORE operation 47
 Retry limit 48
 RJE (see Remote Job Entry)
 RJELC (SVC 232) 18
 RQE 6,7
 RT routine (CZABF) 38-39
 Entry points 38
 Modules called 38-39
 Exit 39
 Operation 39
 Error conditions 39
 System control block usage 39
 Flowchart 167-169
 RUNMODE parameter 46

 S-entry 16-18,20,23
 Search TDT chain 28
 SECURE command 46
 Separate background task 46,47

 SHUTDOWN routine (CZACN) 39-40
 Entry points 39
 Modules called 39-40
 Exit 40
 Operation 40
 Error conditions 40
 System control block usage 40
 Flowchart 170
 Shutdown message 40
 Shutdown subprocessor 11
 Special purpose ABEND and DDEF 16
 Station ID (RJE) 18,22-23,26
 STOPLOG 6
 Sub-task processing 2
 SVC 232 (RJELC) 18
 SYSIN data set 21,34
 SYSIN object code (BULKI/O) 1
 SYSIN processing 20-21
 SYSOUT listing 1
 System shutdown 39-40
 System public disk storage I/O 15

 Tape labeling 34
 Task 001 7,31
 Task 002 15
 Task initiation 1
 Task termination 1
 Termination of TSS operation 39-40
 Termination of conversational task 32
 Third-level TSI availability 10
 Timer-driven BULKIO 17
 Time in microseconds 4
 Time Shared Dump/Restore (CZUFA) 46-48
 (see also DMPRST)
 Time Shared Utility Module (CZCBC) 48-49
 (see also TSU Module)
 Time Shared Utility Programs 45
 DMPRST (CZUFO) 46-48
 PATCLEAR (CZUFA) 45-46
 TSU Module (CZUBC) 48-49
 Timer interrupt processor 10
 Timing statistics (BULKIO) 17
 Timing the BULKIO task 17
 TWAIT processor 11
 TSU Module (CZUBC) 48-49
 Entry points 48
 Modules called 48
 Exits 48
 Operation 48-49
 Error conditions 49
 System control block usage 49
 Flowchart 201-203

 Unique dd/ds name 48
 Unit record assignment 30

 Valid ACK member 26
 VAM data set removal 27
 VAM Input Recovery routine (WABV) 28
 Entry points 28
 Modules called 28
 Subroutines 28
 VAM Output Recovery routine (WAVO) 29
 Entry points 29
 Modules called 29

Subroutines 29
 Operation 29
 VAM recovery 28-29
 VSENDR macro instruction 43

WABE (see MSAM Input Recovery) 27
 WABF 27
 WABN 27
 WABV (see VAM Input Recovery) 28
 WABVEX 27
 WALNIT (see Initialization Recovery) 29
 WAMO (see MSAM Output Recovery) 28
 WAVO (see VAM Output Recovery) 29
 WT routine (CZABI) 40-41
 Entry points 40-41
 Modules called 41
 Exit 41
 Operation 41

Error conditions 41-42
 System control block usage 42
 Flowchart 181-187
 WT macro instruction 43
 WTL macro instruction 6,43
 WTO macro instruction 6,43
 WTOA macro instruction 6,43
 WTOR macro instruction 6,43

XWTO routine (CZABQ) 43-44
 Entry points 44
 Modules called 44
 Exit 44
 Operation 44
 Error conditions 44
 System control block usage 44
 Flowchart 188-189



International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
[U.S.A. only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]