Network Control System Project Block III Software

S. E. Friesema, J. Blackstock, T. Gee, N. Hammerwold,
A. Irvine, N. Larson, and J. Williams
DSN Data Systems Development Section

The Network Control System (NCS) Project is responsible for the software implementation of the Network Operations Control Center (NOCC) Block III system. This involves the programming for a twenty-one computer distributed network (including on-line spares). This system is presently in a three-phase development plan. Phases I and II are currently designed and coding is in progress. Phase III requirements are being reviewed with respect to Phase I and II capabilities and NCS manpower availability. NOCC operational testing for Phases I and II will begin in September 1975, with final transfer to DSN operations Feb. 1, 1976. NOCC Phase III is tentatively scheduled for transfer to DSN operations on 1 July, 1976

I. Introduction

The Network Control System (NCS) software development is divided into three major implementations: Block I, Block II, and Block III. This report will summarily describe the software implementation elements and status for NCS Block III.

The NCS Block III is an 18-computer distributed system. There are additionally three controller and preprocessor computers utilized as part of larger computer subsystems. The NCS Block III System is being implemented to:

(1) Receive and provide accountability for high-speed data blocks (HSDBs) transmitted by Deep Space Stations (DSSs) to JPL.

(2) Transmit and display command data utilized in the configuration and control of DSSs.

(3) Generate, format, and transmit to selected DSSs:

(a) Sequence of Events (SOE) files

(b) Seven-day schedules

(c) Radio-metric predict data

(d) Telemetry predict data

(e) Test data

(f) Data recall control messages

(4) Provide printer dump of selected data being transmitted by the DSSs to JPL and from NCS to

the DSSs. This includes wideband data being received in either 1200- or 2400-bit block size.

(5) Provide sufficient HSDB analysis to assure meaningful displays for operational support in the areas of telemetry, tracking, command, and monitor.

(6) Provide operational status of the NCS to a level that will readily identify major system operational problems.

(7) Provide flight projects with an intermediate data record (IDR) tape. This tape should contain a merged file of the maximum number of useable HSD/wideband data (WBD) blocks recoverable by the DSN.

(8) Provide fast subsystem recovery by means of on-line spare processors.

The NCS Block III functional configuration is shown by the simplified block diagram in Fig. 1. The following paragraphs describe briefly the software for each of the major functional subsystems. The only exception to this is in the real-time monitor (RTM) area. Rather than presenting a summary of each monitoring application, the general software structure for the RTMs as a group is presented. For a more detailed description of the NCS hardware configuration see Ref. 1.

II. NCS Communication Log Processor and Network Control Processor

Control of the communication log processor (CLP)/network control processor (NCP) subsystems is accomplished through hardware-generated interrupts and operator control inputs (OCI) (Figs. 2, 3). At the completion of each data block transfer, either input or output, an interrupt is generated. This interrupt causes control to be transferred to the proper input/output (I/O) handler, either comm buffer handler, star switch controller (SSC) handler, terminet handler, magnetic tape (MT) label printer handler, MT handler, or disk handler. The I/O handler performs the proper processing, and if necessary, places the data block on queue to one of the internal subroutines or to another I/O handler. Input of an OCI by the operator may cause a change in the control of the program, such as a request to cease reading an input data line or cease outputting over a particular SSC path.

All input data blocks are received by the comm buffer handler or the SSC handler. They are then queued to the routing module. The routing module compares the information contained in the GCF header with the data contained in the proper routing table to obtain an output line number, or numbers. The data block, along with the output line numbers, is then queued to the proper handler, or handlers, for output. When an OCI is input by the operator it is queued to the OCI processor module, which takes the necessary action to complete the request.

The interface between the CLP and NCP subsystems is the HSD traffic, which has originated outside either subsystem and is routed by both to the proper destination. In only one case is a control message generated by one subsystem with the other subsystem as the destination. This is the case where the backup CLP has been changed to the prime CLP by operator input. In this case the CLP sends a control message, in HSDB format, to the NCP for it to become the prime NCP.

III. NCS Display Processor

The display subsystem operates in the Network Operations Control Area (NOCA) of Bldg. 230. Its purpose is to provide a centrally controlled facility for displaying data for the Network Operations Control Center (NOCC), and for receiving and processing operator control inputs to direct NOCC operation. Additionally, the display subsystem accepts bulk data, which it prints, or transmits to the Network Data Processing Area (NDPA) in Bldg. 202 (Fig. 4).

A. Displays

Up to 46 digital TV (DTV) display devices are driven at the NOCA. Of these, up to 31 are slave units, and up to 15 are master units with numeric keyboard for display channel selection. The DTV screen is logically divided into quadrants, each quadrant individually assignable to any defined display format. In addition to the DTVs, there are three CRT display devices, each with a display capability of two pages, where a CRT page equals a DTV quadrant in terms of display capacity. A subset of the display subsystem software resides in each other minicomputer subsystem of NOCC that has a local cathode-ray tube/keyboard (CRT/KB) display capability as backup.

B. Operator Control Inputs

OCI's are entered from the CRT/KB alphanumeric keyboard. The OCIs are validated at NOCA, and illegal entries are rejected with a response. OCI prompting is available to assist the operator. OCI's that pass NOCA validation are encoded for wideband data line (WBDL) transmission to Network Data Processing Area (NDPA). A subset of the OCI software resides in each other minicomputer subsystem of the NCS that has a local CRT/KB display as backup.

C. Bulk Data

Prestored bulk data can be entered from the card reader, and acted upon as prestored OCIs, or as data for transmission to the NDPA. Additionally, bulk data received from the NDPA can be printed at the NOCA. The Network Operations Control Area also prints a log of system events, OCIs, responses, and system status on separate printers.

IV. NCS Real-Time Monitors

In order to support all of the software functions required of a real-time monitor (RTM) operating within the NOCC, we have identified the areas of commonality within each RTM and assigned the responsibility of interface/design and implementation to people who are not programming each RTM applications module. This approach frees the applications programmers of common supporting activities and eliminates multiple designs of the same logical function. The end result is a series of stand-alone products which have been modularly developed (Fig. 5).

All data arriving at an RTM consists of two types: (1) DSS originated or (2) Intra-NOCC data (a single exception to this is Ground Communication Facility (GCF) monitor-originated status which is sent to the monitor RTM for status processing). The GCF handler is responsible for bringing the data, in the form of 1200 or 2400 bit data blocks, into I/O buffers allocated by the application. Deep Space Station-originated data are queued directly for application processing while intra-NOCC data are queued for further analysis by the preprocessor routine. The preprocessor uses information contained in the GCF header to route the data to the proper routine by queueing the data onto one of several queues which have been defined by the application. The application module performs the required analysis of DSS-originated data, using the services of the high-speed data block (HSDB) reformatter to extract and realign the data as received for ease of processing. It then saves the extracted data in its subsystem control and data tables for further use and for subsequent display. Incoming data are also used to update the system performance record (SPR)/network performance record (NPR) (monitor RTM only), which is written to disk by the SPR control processor when full. Both the application and the HSDB validation and accountability routines add to the GAP list when block sequence anomalies occur. The GAP list is also maintained on disk by the SPR control processor. At operator request or when the SPR/NPR or GAP list on the disk is nearly full, the SPR despool processor will be advised to begin transferral of the data to the NC support subsystem. This is done by acquiring a 2400-bit I/O buffer and reading the accumulated SPR/NPR or GAP list data from the disk into the buffer and queueing the buffer for transmission, via the GCF handler, when it is full.

OCI's entered locally are validated and reformatted by the operator communications processor, and are queued for processing by the application. When the application has completed its processing, the response indicated by the application is returned to the CRT/KB terminal from which the OCI was issued to complete the processing.

Asynchronous to other processing, the display preparation processor can be preparing a new display in response to a request from the display processor or preparing an update to an active format which is being displayed locally or remotely in the NOCA. To do this, the routine draws upon the data contained in the display data table and the format control table employing the services of the applications display address resolution routine to access the data contained in the subsystem control and data tables. If the data is to be displayed remotely, the accumulated data is packed into 2400-bit I/O buffers and queued for transmission to the display computer via the GCF handler.

V. NCS Support Processor

The Network Control Support Subsystem (NCSS) is required to perform the non-real time functions of five subsystems as well as its own tasks. The non-real time programs common to those subsystems can be generally categorized as:

(1) File extraction and HSD output

(2) SPR/NPR data extraction and recording

(3) Standards and limits generation

(4) Predicts generation

(5) System analysis

The NCSS program shall operate in a multi-use environment in that it will have the capability to operate in several different modes as it services the needs of the different subsystems. (Fig. 6).

(1) Batch mode. In this mode the NCSS will operate in a non-real time computational manner. Such func-

tions as sequence of events and tracking predictions shall be performed in this mode.

(2) Demand-responsive mode. The NCSS shall be capable of a dialog with an end-user. Such functions as standards and limits (S&L) update shall be performed in this mode.

(3) Near-real time mode. The NCSS shall be capable of responding to specific stimuli and provide controller responses under clock impulse. Such functions as the transmittal of HSDBs shall be performed in this mode.

(4) Development mode. The NCSS shall provide resources to develop and test programs for the entire NCS in a limited mode.

(5) Test mode. The NCSS shall be able to operate in a limited test mode.

The NCSS program shall operate asynchronously with respect to the NCS except at those times when it is called upon to act in a near-real time or demand-responsive mode. This means that events represented by messages from all data paths, including peripherals, shall be scheduled. Action shall be deferred until such time as the proper resources can be applied to servicing such events. Events shall be acted upon at some time later than their time of entry into the environment of the NCSS.

VI. NCS Data Records Processor

The functions of the data records processor (DRP) (Fig. 7) are primarily the following:

(1) The recall of the data from a DSS based upon a gap list received either from operator entry at the DRP (manual mode) or from the Support Computer (automatic mode).

(2) The generation of either an intermediate data record (IDR) or a fill data tape by combining a network data log (NDL) and a recall data tape.

(3) The replay of an NDL to a real-time monitor (RTM).

The transmission of data between the DRP and the Deep Space Stations, Support processor or an RTM will utilize the star handler and the buffer queueing/dequeueing common software. Operator inputs (OCIs) to initialize and/or control the functions within the DRP will utilize the OCI handler common software package. Any status or alarm messages generated by the above functions will be output to the CRT/KB display and will interface with the display handler, also a common software module.

The three prime functions run independentally of each other. Function (1) is activated upon receipt of an OCI or a gap list message from the support processor. Its output is the recall data tape. Function (2) takes this tape along with the NDL from the communications log processor (CLP) and combines them to produce either an IDR or a fill data tape. The last function causes an NDL to be replayed to an RTM at a selectable rate.

VII. NCS Test and Training Processor

The test and training Subsystem (TTS) provides test support to all of the subsystems within the Network Control System (NCS) and to the Deep Space Station (Fig. 8).

High-speed data blocks (HSDBs) or wide-band data blocks (WBDBs) will be created from card input. These blocks will be output through the star switch controller (SSC) by the GCF handler at timed intervals. These blocks will also be saved on magnetic tape or disk if requested. The created test blocks may be modified at any time by OCI or card input.

HSD or WBD blocks destined for other systems may also be switched to the TTS by the SSC. These blocks will be put into the HSDB queue by the GCF handler. They will then be written onto magnetic tape for later test transmission.

HSD or WBD blocks may also be input from previously recorded magnetic tapes. These blocks will be interrogated and the ones specified (by OCI) will be output through the SSC by the GCF handler. The TTS will process up to six streams defined by one or all of the following parameters: source code, destination code, spacecraft number, user-dependent type code, and data-dependent type code.

VIII. NCS Block III Software Status

The NCS Block III software development is divided into three phases:

(1) Phase I Development, complete Sept. 1, 1975. This phase includes the basic capabilities to receive, log, recall, and merge DSS data blocks.

(2) Phase II Development, complete Dec. 1, 1975. This phase meets all the basic capabilities listed in the introduction.

(3) Phase III Development, complete approximately July 1, 1976. Refinements to allow more efficient control as well as increased status visibility will be

incorporated in this phase. Lower priority functions and approved operational change requests will be implemented as manpower and operations permit.

Presently a large portion of the systems software development is being consumed in the new interface (hardware and software) integration. Due to the relatively large number of newly developed interface assemblies as well as new vendor-developed handlers, the system integration has been delayed. It is anticipated that problems will be corrected and the scheduling will be met for NCS Block III.

Reference

1. Petrie, R. G., "NCS Minicomputer Systems Status Report," in *The Deep Space Network Progress Report 42-22*, pp. 152–159. Jet Propulsion Laboratory, Pasadena, Calif., Aug. 15, 1974.

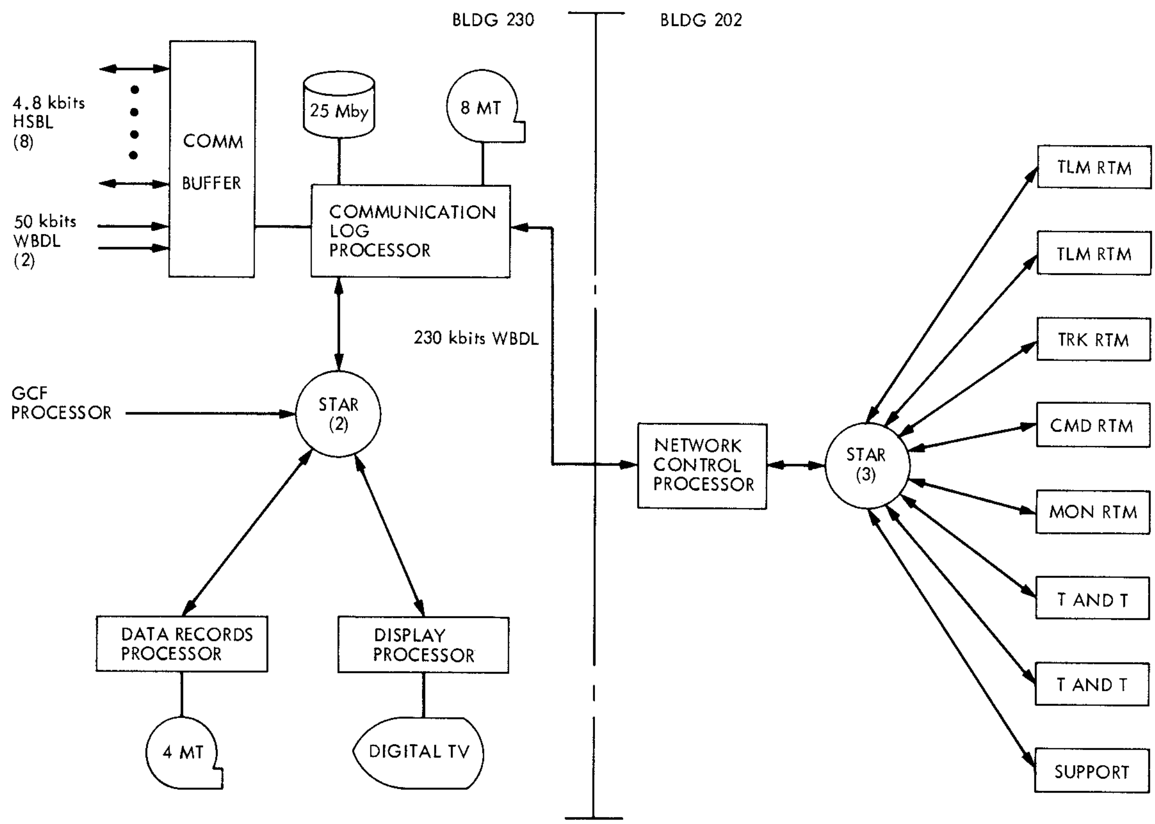

Fig. 1. Network Control System Block III configuration

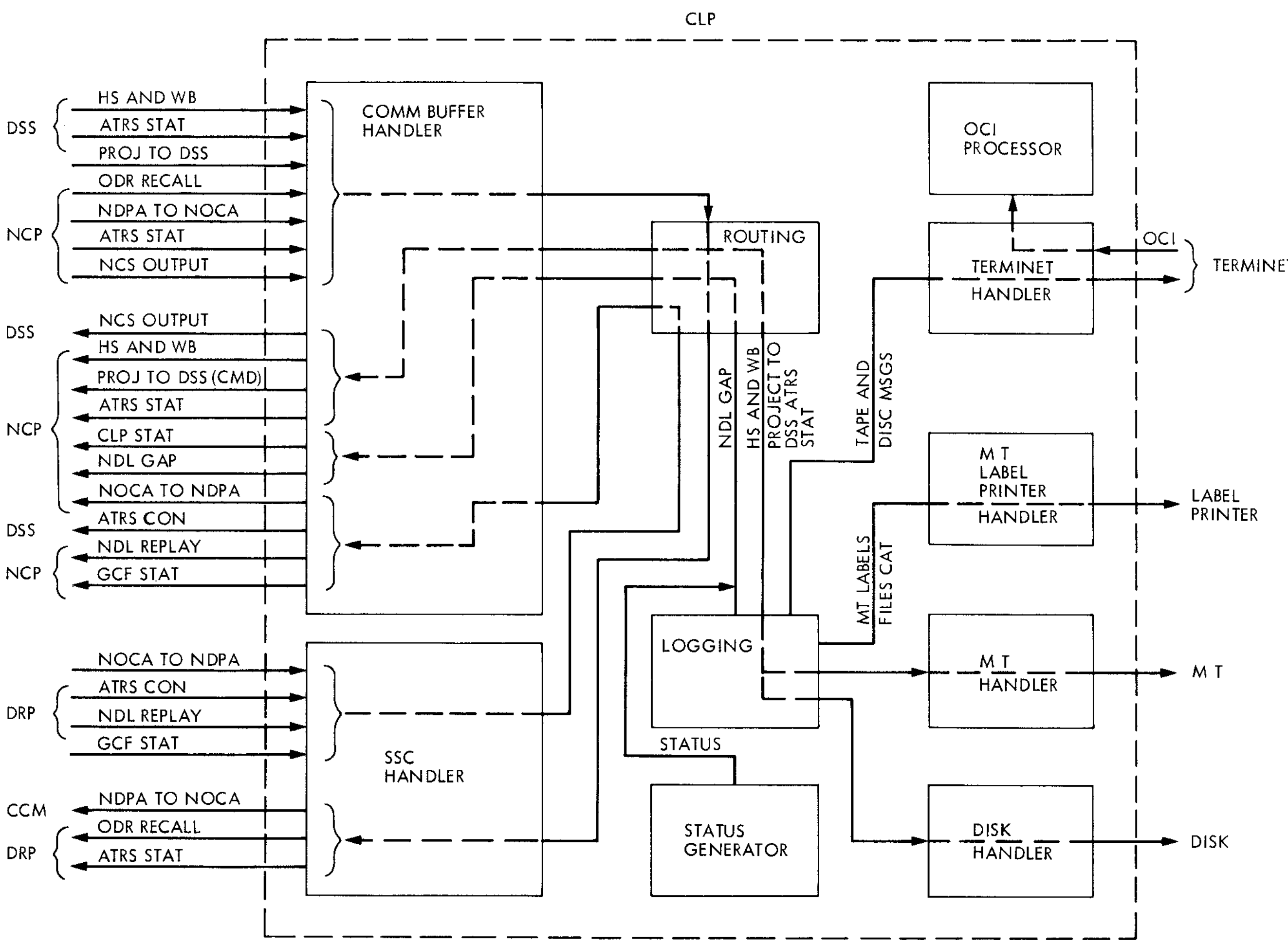

Fig. 2. Communication log processor software module diagram

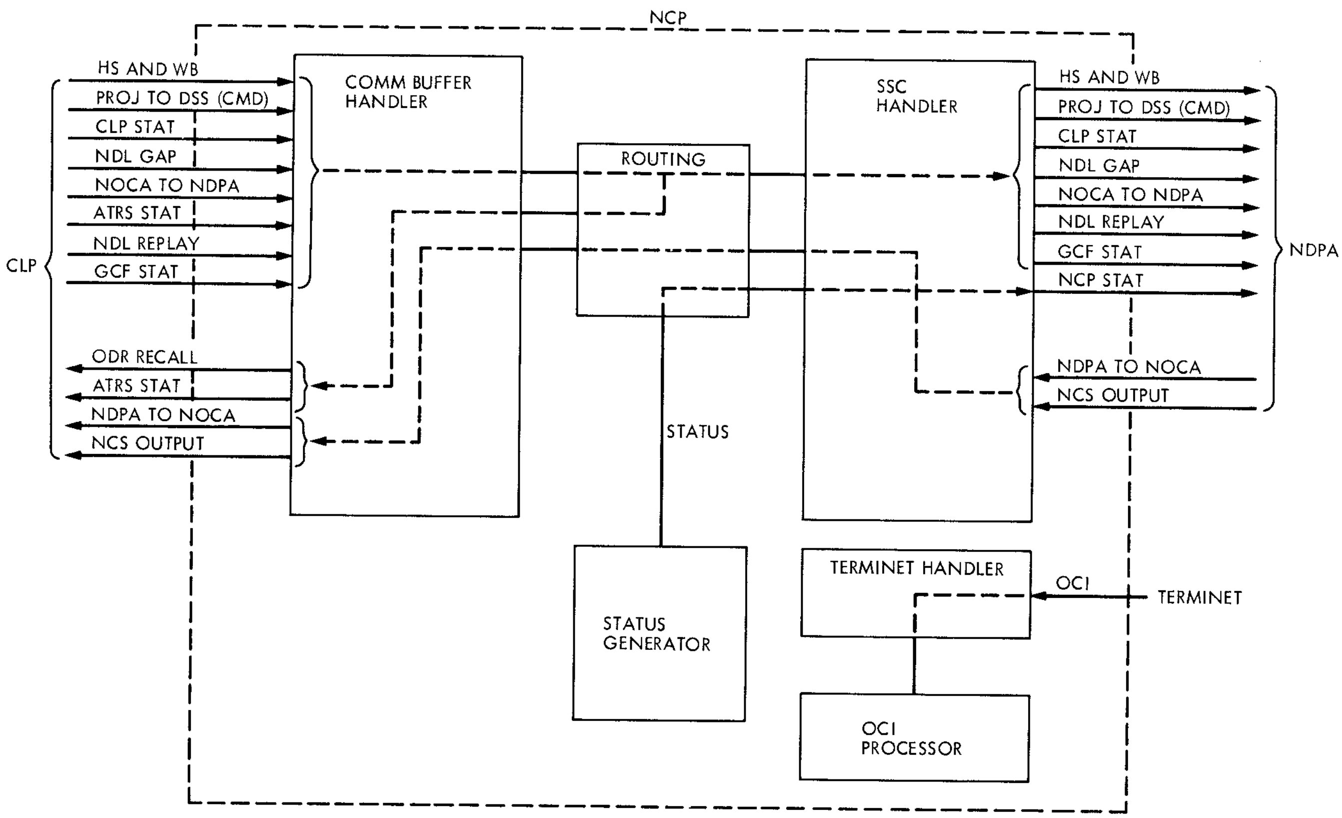

Fig. 3. Network Control Processor software diagram

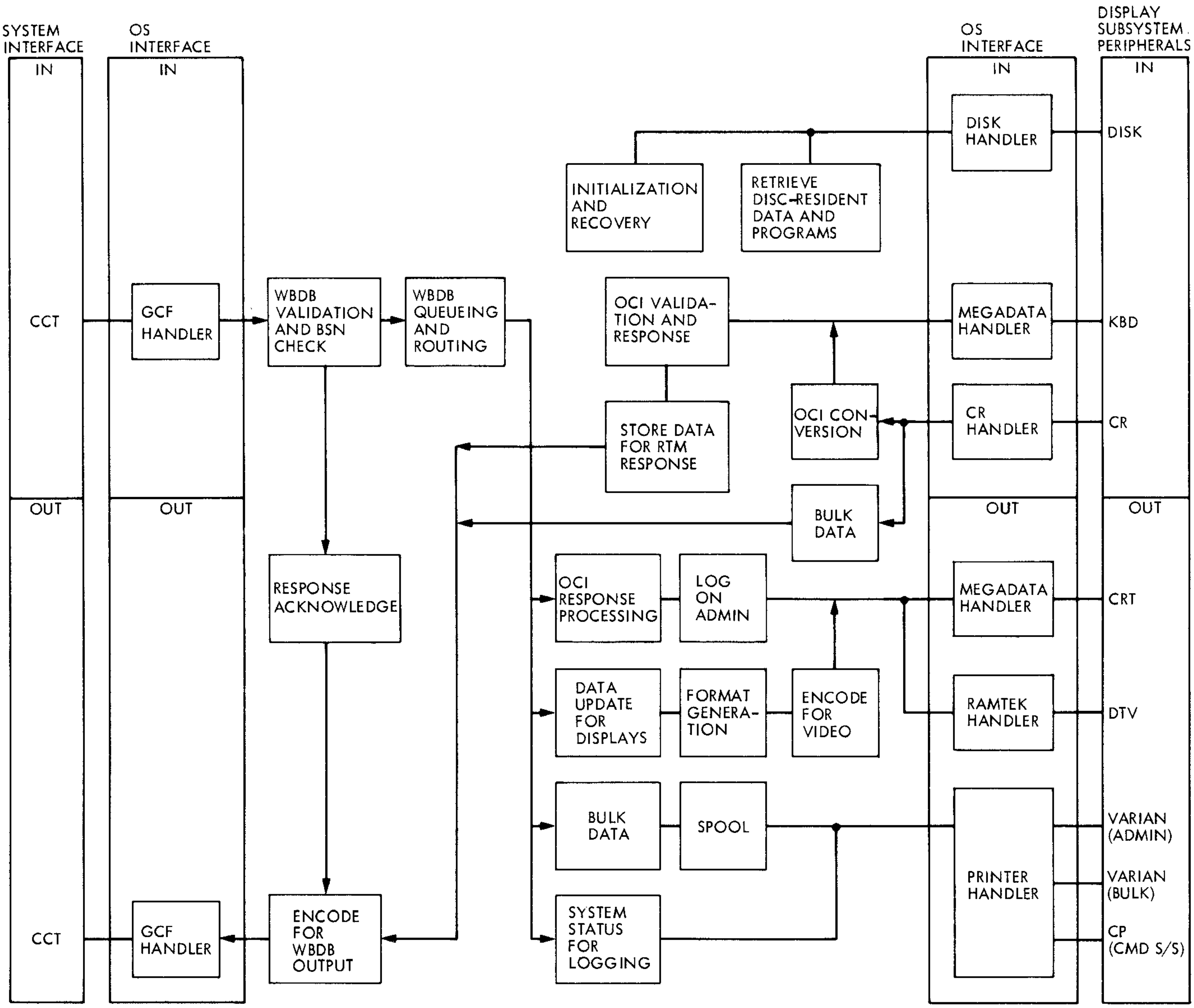

Fig. 4. Display subsystem software module diagram

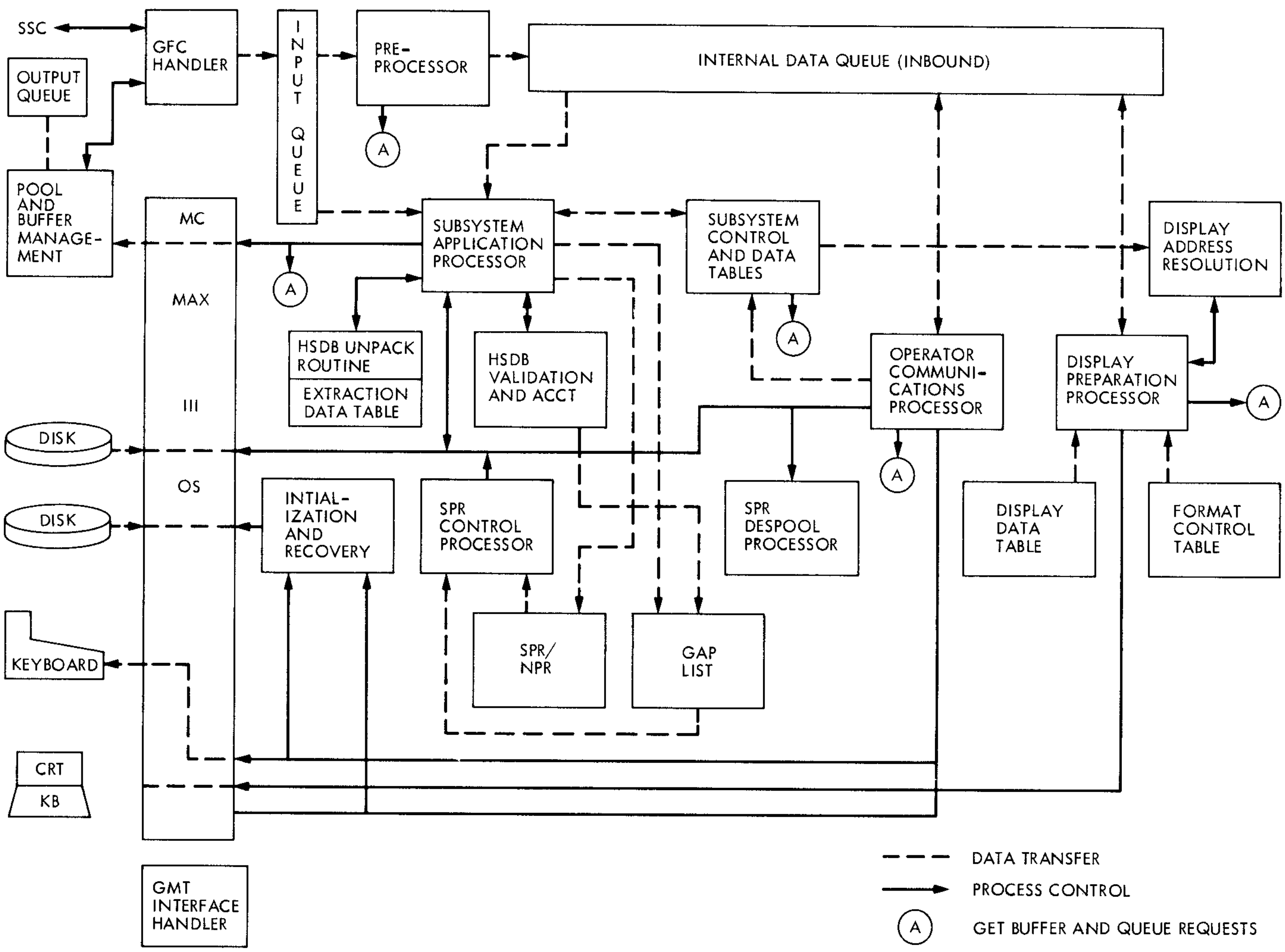

Fig. 5. Typical real-time monitor software module diagram

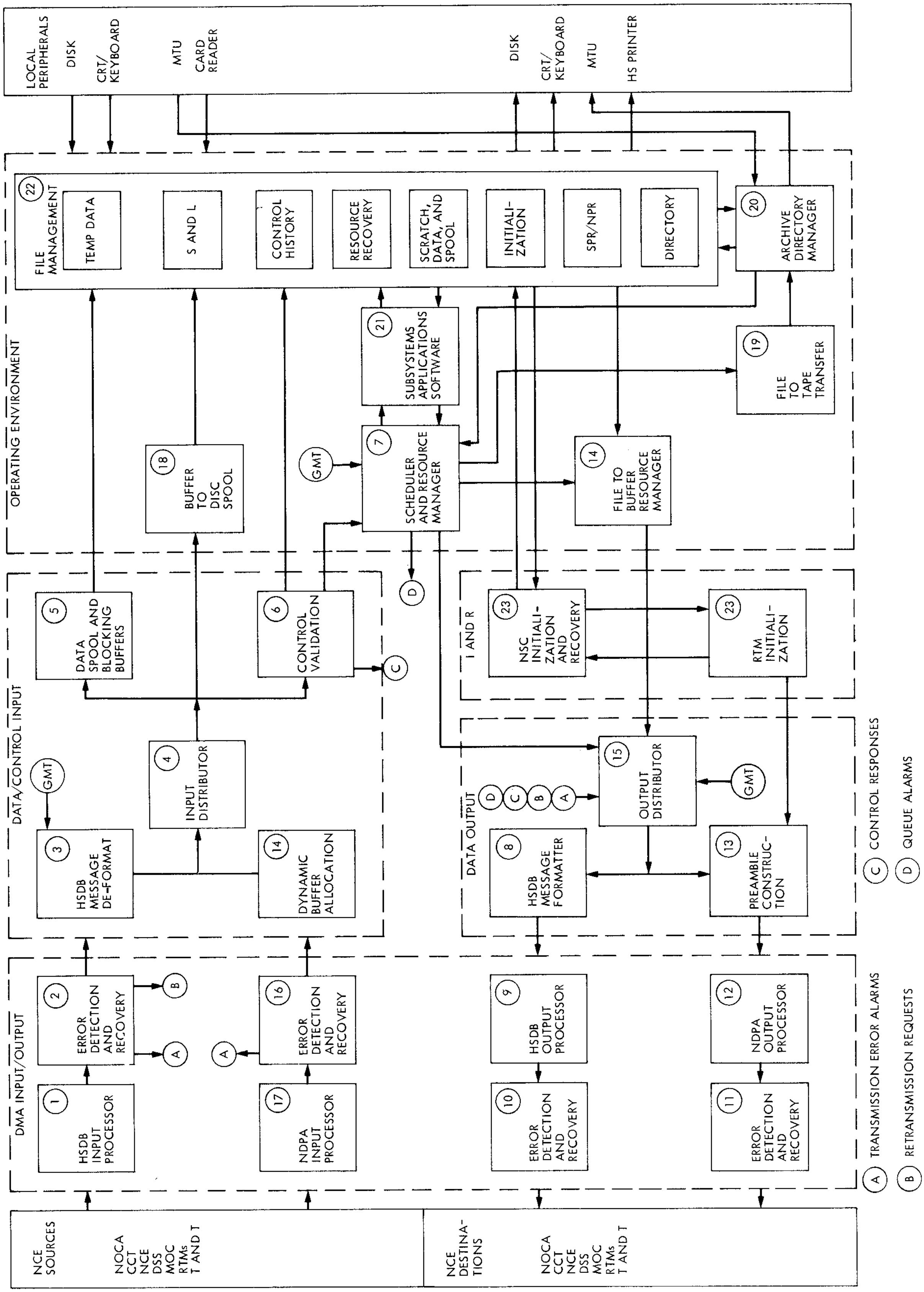

Fig. 6. Support subsystem software module diagram

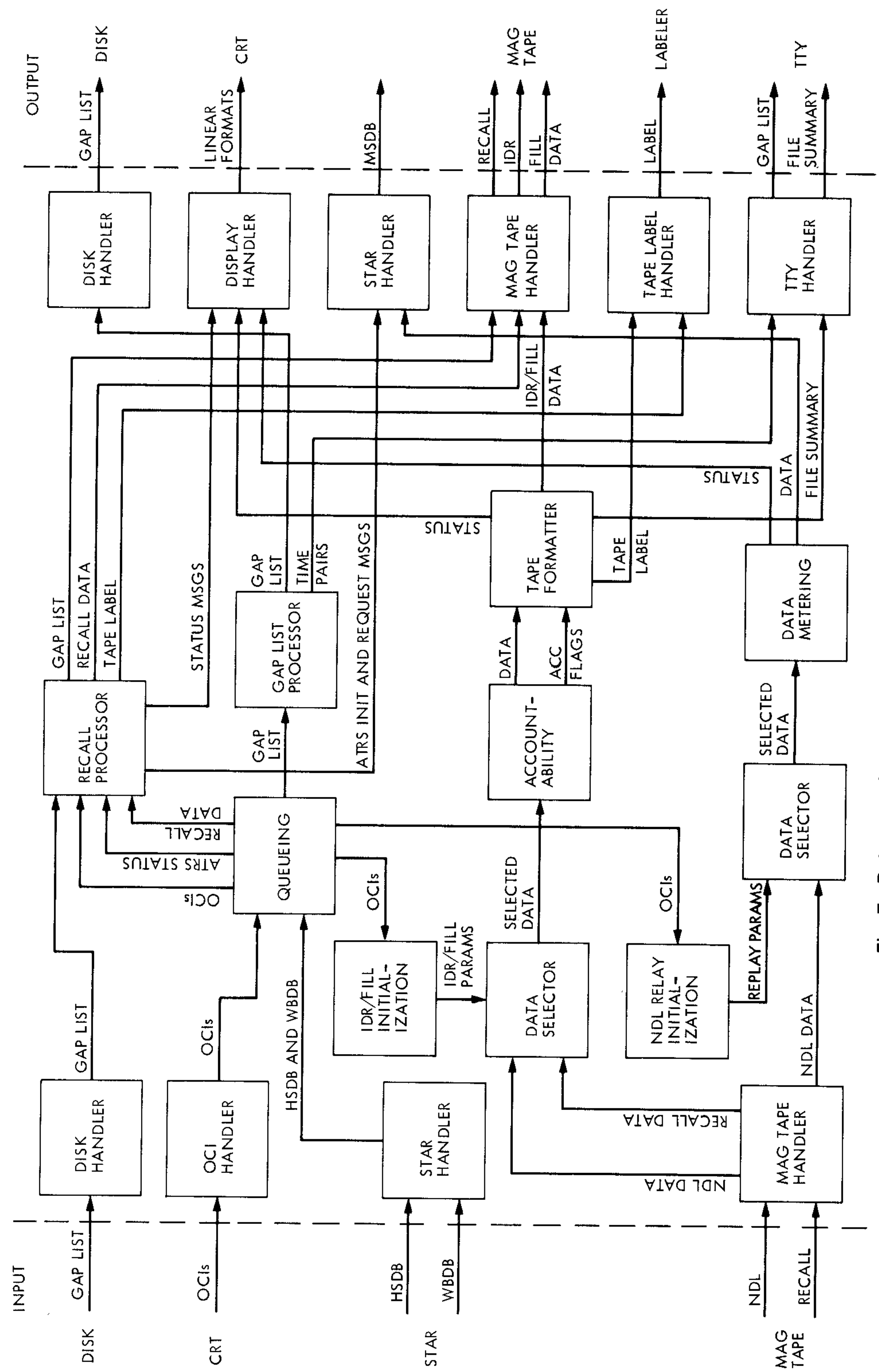

Fig. 7. Data records processor software module diagram

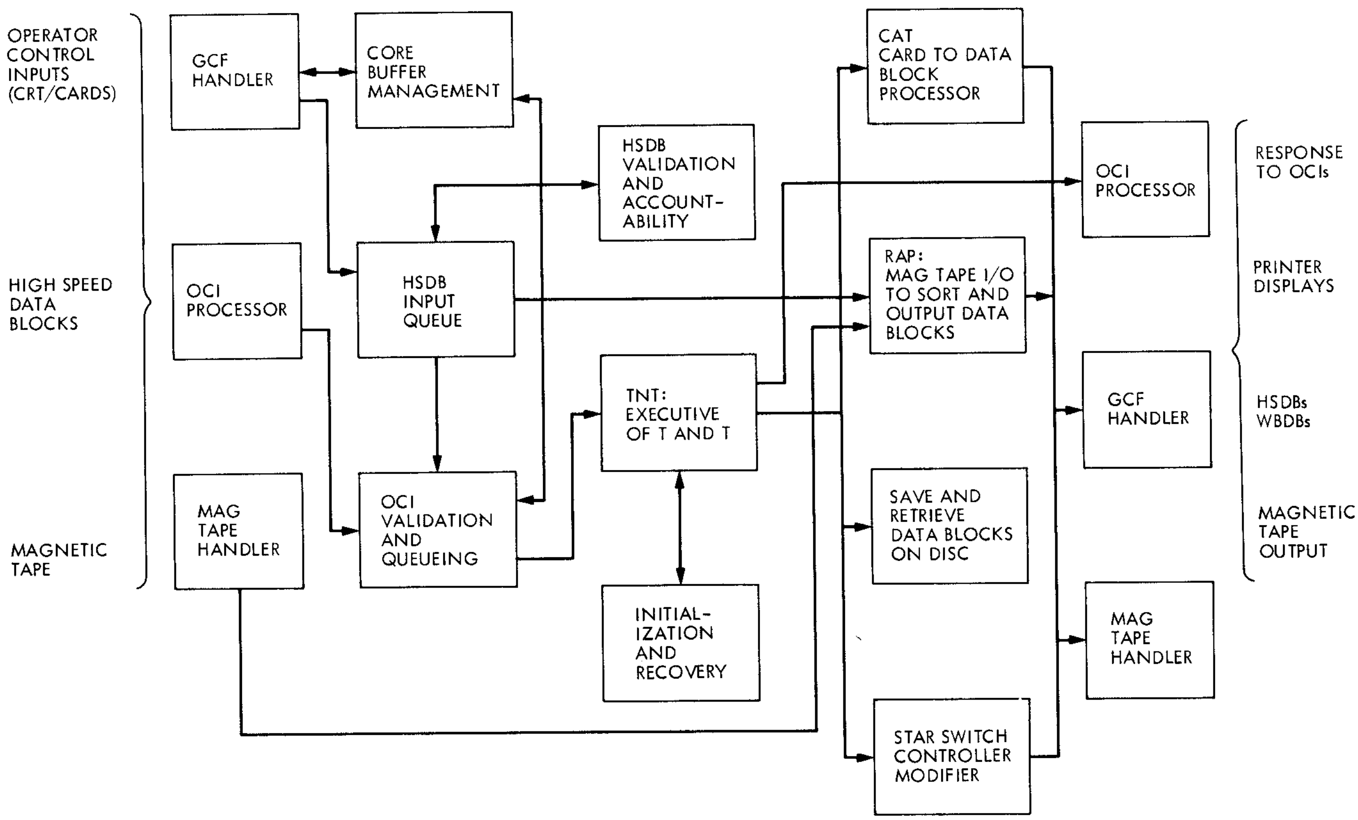

Fig. 8. Test and training subsystem software module diagram