# INTERCOMM

## GENERALIZED FRONT END FACILITY

SDA
71 5th Avenue
New York, New York 10003

## LICENSE: INTERCOMM TELEPROCESSING MONITOR

# Generalized Front End Facility

## Publishing History

| Publication | Date | Remarks |
|---|---|---|
| First Edition | August 1975 | This manual corresponds to Intercomm Release 6.2. |
| Second Edition | December 1982 | Updates and Revisions corresponding to Intercomm Release 9.0. |

# PREFACE

Intercomm is a state-of-the-art teleprocessing monitor system of SDA, executing on the IBM System 360/370 family of computers and operating under the control of IBM Operating Systems (MFT, MVT, VS1, MVS). Intercomm monitors the transmission of messages to and from terminals, concurrent message processing, centralized access to I/O files, and the routine utility operations of editing input messages and formatting output messages, as required.

The Generalized Front End Interface (GFE) is offered as a Special Feature to the basic Intercomm system. It provides the capability to support nonstandard Front End configurations via a convenient and standarized facility. Support for unusual Front End configurations can be implemented with minimal effort on the part of the user. The GFE facility is also required as an interface to the Intercomm Extended TCAM Front End support.

This document describes the GFE Interface and details the specifications for implementation. As a reference document, only specifications for the GFE facility are presented. The reader is referred to the following Intercomm publications in conjunction with the use of this document:

- BTAM Terminal Support Guide
- Basic System Macros
- System Control Commands
- TCAM Support Users Guide

A User Review Form is included at the back of this manual. We welcome recommendations, suggestions and reactions to this or any Intercomm publication.

# INTERCOMM PUBLICATIONS

## GENERAL INFORMATION MANUALS

Concepts and Facilities

Planning Guide

## APPLICATION PROGRAMMERS MANUALS

Assembler Language Programmers Guide

COBOL Programmers Guide

PL/1 Programmers Guide

## SYSTEM PROGRAMMERS MANUALS

Basic System Macros

BTAM Terminal Support Guide

Installation Guide

Messages and Codes

Operating Reference Manual

System Control Commands

## CUSTOMER INFORMATION MANUALS

Customer Education Course Catalog

Technical Information Bulletins

User Contributed Program Description

## FEATURE IMPLEMENTATION MANUALS

Amigos Users Guide

Autogen Facility

ASMF Users Guide

DBMS Users Guide

Data Entry Installation Guide

Data Entry Terminal Operators Guide

Dynamic Data Queuing Facility

Dynamic File Allocation

Extended Security System

File Recovery Users Guide

Generalized Front End Facility

Message Mapping Utilities

Model System Generator

Multiregion Support Facility

Page Facility

Remote Job Entry (OS)

Store/Fetch Facility

SNA Terminal Support Guide

TCAM Support Users Guide

Utilities Users Guide

# TABLE OF CONTENTS

# LIST OF ILLUSTRATIONS

# Chapter 1

# INTRODUCTION

Intercomm's Generalized Front End (GFE) Interface provides the capability to support nonstandard Front End configurations with a minimal amount of effort on the part of the user. It provides an interface between the Intercomm BTAM Front End and user-written routines which perform the actual I/O and related functions required to access the nonstandard Front End device(s). All of the logic related to control of the teleprocessing environment is performed by Intercomm modules except the specifics related to the user's hardware. The effect, essentially, is to use the user's access method instead of BTAM.

Typical uses of GFE are:

- Interface to nonsupported programmable terminals
- Interface to nonsupported hardware Front End devices
- Interface to nonsupported minicomputers
- Interface to TSO devices
- Interface to unit record devices
- Interface to TCAM terminals: this support is the Intercomm-supplied Extended TCAM Support, a special facility of the BTAM Front End.

With GFE, the user has a convenient and standardized facility for implementing support for unusual Front End configurations. GFE may be used exclusively or in combination with standard supported devices operating over normal BTAM lines. Terminals operating under GFE may take advantage of the Front End facilities available to standard BTAM supported terminals (such as 3270 AID Processing, TPUP, Lock, Backspace Correction, etc.). See the BTAM Terminal Support Guide for implementation of all BTAM Front End facilities, and System Control Commands for implementation of network control commands.

## Chapter 2

## OPERATIONAL OVERVIEW

The Intercomm GFE Interface consists of logic within the Intercomm BTAM Front End to call user routines to perform functions normally supplied by Front End logic for supported devices. (See Figure 1.) Each type of nonsupported device controlled by user routines is effectively a line group plus lines and terminals to Intercomm.

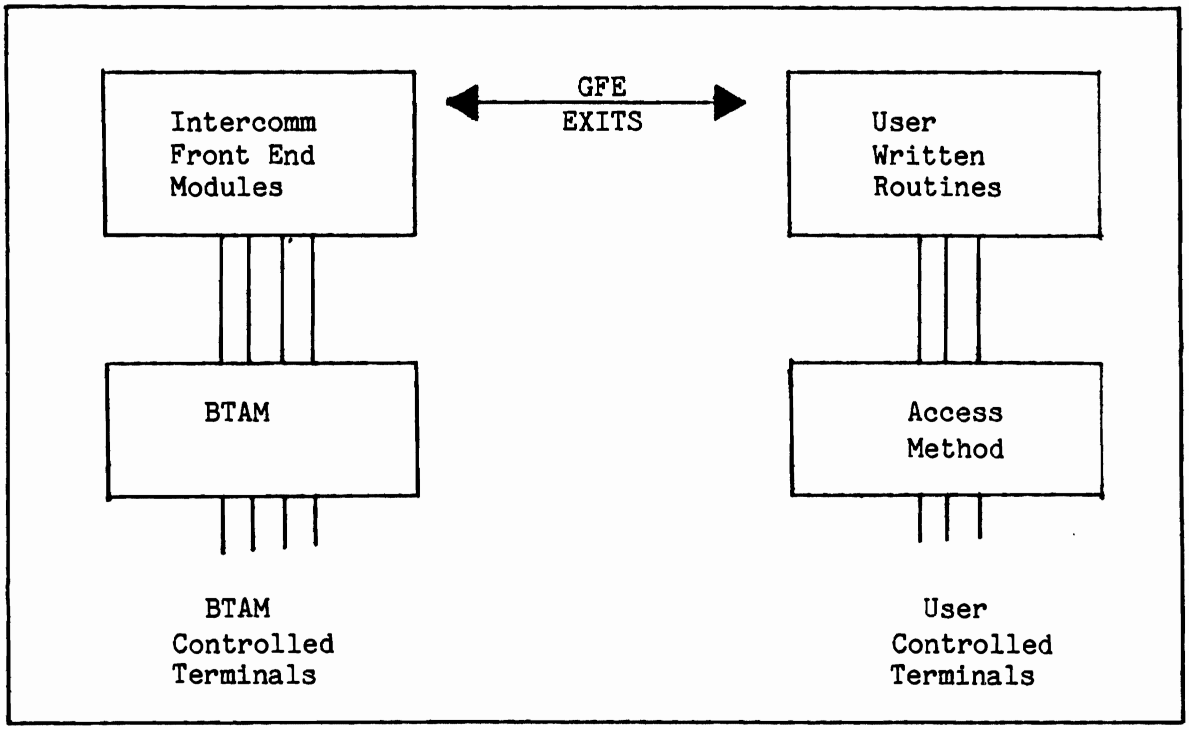

Intercomm Front End Modules
GFE EXITS
User Written Routines
BTAM
Access Method
BTAM Controlled Terminals
User Controlled Terminals

Figure 1. GFE Operational Overview

The standard Intercomm BTAM Front End Network Table defines the operation of each "GFE-device." A LINEGRP macro generates a DCB for the GFE device(s); a BLINE macro generates a DECB; a BTERM macro specifies a device type that the user routines are controlling. Front End processing of input messages will expect control character sequences within message text; processing of output messages may add control characters to the message text passed to the user routines.

Device types which may be simulated by user routines are listed in Figure 2.

| Device Type | BDEVICE TERMTYP= |
|---|---|
| IBM 2260 Remote | IBM2260 |
| IBM 2740 model 1 or 2 | IBM27401 or IBM27402 |
| IBM 3270 Local or Remote | IBM327L or IBM3270 |
| IBM 1050 | IBM1050 |
| Leased Teletype | TELETYPE |

Figure 2. GFE Device Types.

Note that these are merely device types. Any physical device may be used, as long as input and output data formats appear to Intercomm to be the same as those of one of the above devices. All messages must be in EBCDIC character format, unless a translate table is specified by the BLINE macro.

The user routines for particular GFE device types are defined by a GFE macro within the standard BTAM Front End Network Table. This macro generates a vector table specifying entry points to user routines for specific functions and additional A-Type or V-Type addresses if required (for example, for user tables or control blocks).

The user routines are called by GFE interface logic within the BTAM Front End for the functions tabulated in Figure 3. A parameter list is used for communication between the GFE interface and user routines; a return code is used to inform GFE of the success or failure of user routine functions.

| Function | Calling Program | Normal BTAM Front End Function |
|---|---|---|
| Table Verification | BTVERIFY | Validate Network Configuration Table Coding |
| Initialize a Line Group | BTAMLINE<br>TPUMSG | OPEN Lines (DCB) defined by LINEGRP macro. |
| Initialize a Line | BTAMLINE | Dispatch (schedule) Input Line Handler (BLHIN) for a line in the Line Group. |
| Input | BLHIN | Issue READ for Line from Input Line Handler. |
| Output | BLHOT | Issue WRITE for Line from Output Line Handler. |
| Reset | BTSEARCH | Cancel Outstanding Read Operation (i.e., a wraplist poll function). |
| Start/Stop Terminal | TPUMSG | Activate/deactivate operation of a particular device. |
| Start/Stop Line | TPUMSG | Activate/deactivate operation of a particular line. |
| Terminate a Line Group | BTAMCLSE<br>TPUMSG | CLOSE Lines (DCB) defined by LINEGRP macro. |

Figure 3. User Routines and Related BTAM Front End Functions

# Chapter 3

## INTERCOMM-SUPPORTED GFE FRONT ENDS

Although GFE is designed to interface with user-provided routines, some Intercomm Front End support is provided by way of GFE. In these cases, GFE will call modules which are provided and supported by Intercomm. One such Intercomm-supported Front End provided is the Extended TCAM Support. This facility allows the user to define TCAM process and destination queues as lines to the Intercomm BTAM Front End. This approach simplifies coding of the TCAM Message Control Program in that no Intercomm-oriented logic need be performed therein (message header construction, etc). Further, the BTAM Front End control commands are then available to the TCAM User.

Refer to the TCAM Support Users Guide for additional detail.

Chapter 4

# INTERCOMM BTAM FRONT END CONCEPTS

The GFE user should be familiar with the Intercomm Front End environment, including Front End table structure and Front End modules. An overview is presented in this chapter. For further details, refer to the BTAM Terminal Support Guide and Basic System Macros.

There are four principal macros used to define the BTAM/GFE Front End Network Table. They are:

| Macro | Defines | Related BTAM<br>Control Block | Comments |
|---|---|---|---|
| LINEGRP | Line Group | DCB | One per ddname |
| BLINE | Line | DECB | One or more per LINEGRP |
| BTERM | Terminal | - | One or more per BLINE |
| BDEVICE | Device Type | - | Referenced by BTERM |

Figure 4. BTAM Front End Macros for Tables

A LINEGRP macro generates a DCB and contains pointers to the first and last BLINEs of the LINEGRP. Each BLINE is therefore associated with exactly one line group and is followed by one or more BTERMs. Finally, each BTERM references a BDEVICE, which defines the terminal's device characteristics and various Intercomm processing options.

Figure 5 illustrates the relationship between the macros and associated control blocks.

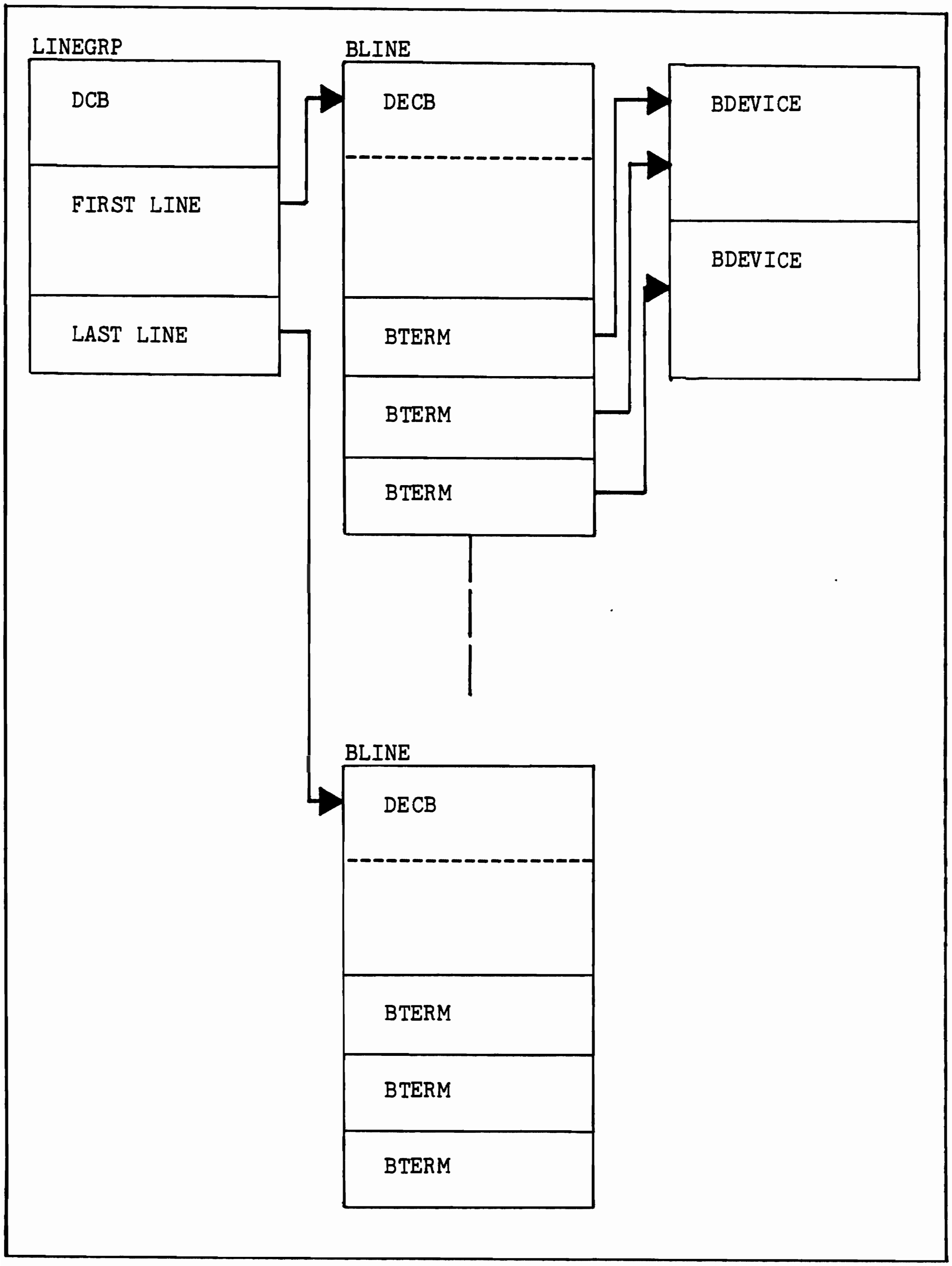


Figure 5. BTAM Front End Macros and Control Blocks

Dsects for each of these control blocks are provided on the Intercomm release tape (SYMREL), and are:

| Control Block | DSECT |
|---|---|
| LINEGRP (DCB) | LGDSECT (IHADCB) |
| BLINE (DECB) | PLNDSECT (IECTDECB) |
| BTERM | PTRDSECT |
| BDEVICE | DEVTABL |

Figure 6. DSECTs for BTAM Front End Macros

The modules which use these tables to control teleprocessing activities and their chief functions are:

- BTAMLINE - Performs Front End initialization.
    - Opens each LINEGRP's DCBs.
    - Dispatches a line handler to perform teleprocessing I/O for each BLINE of an opened line group.
- BTVERIFY - A startup routine which performs validity and consistency checks on the Front End tables.
- BLHIN and BLHOT - The most widely used line handler. BLHIN reads terminal input, BLHOT writes terminal output. BLHIN and BLHOT are logically a single line handler, although they are physically two modules. There is one BLHIN/BLHOT thread created for each BLINE of an opened line group.
- BTSEARCH - A service routine which:
    - Locates input verbs.
    - Identifies where input should be routed.
    - Issues BTAM RESETPL to free up a line which has queued output.

- TPUMSG - A routine which processes Front End network control commands, such as TPUP/TDWN, STLN/SPLN, and STLG/SPLG.

- BTAMCLSE - A routine (Csect in BTAMLINE) which shuts down the Front End during an Intercomm closedown.

These, very briefly, are the "building blocks" of the part of Intercomm's BTAM Front End which supports TP communication and TP control. The remainder of the Front End consists primarily of queuing control routines and a variety of other specialized line handlers.

# Chapter 5

## GFE MODULE DEVELOPMENT GUIDELINES

Each user routine of a set of modules composing an access method for a device type is called to perform a particular function normally provided by BTAM Front End logic as previously listed. The entry points are either defined uniquely by the GFE macro, or the GFE macro defines one entry point for a user routine which in turn identifies functions to be performed by a code passed by the GFE interface at the time of the call.

User modules must be designed with the awareness that they will operate as:

- Intercomm system routines

- Intercomm Front End routines

- GFE routines

Each of these considerations is discussed more fully below.

An Intercomm system routine is a module which is considered a part of the Intercomm Control Program, as distinguished from a subsystem which is treated as an application module. Standards for the design and coding of an Intercomm system routine are listed below. This list is provided as a guideline only, and is not necessarily comprehensive.

- Multithreading considerations

  1. Use the INTWAIT or DISPATCH macro for explicit WAIT and TIME functions.

  2. Avoid, or at least minimize, operating system operations which imply a WAIT (such as PUT or ENQ). Most of these can be performed by an Intercomm-provided macro (for example, DISPATCH, INTWAIT, INTENQ/INTDEQ) which functions in such a way as to overlap other Intercomm work with the embedded WAIT. For example, substitute WRITE/DISPATCH/CHECK for PUT.

  3. Use the Intercomm STORAGE and STORFREE macros to acquire and release storage (not GETMAIN/FREEMAIN).

  4. Modules which are multithreaded must be coded as reentrant, and use standard Assembler Language linkage conventions.

- Resource Responsibility

  System routines operate with a zero thread number. Intercomm's automatic resource purging is provided only for subsystems, which operate with a nonzero thread number. Therefore, it is entirely the responsibility of the system routine to pair off the following macros which request and release resources:

  STORAGE/STORFREE
  SELECT/RELEASE
  INTENQ/INTDEQ

As Intercomm BTAM Front End routines, the GFE routines will be called to perform specific device-dependent functions. Recall that GFE devices are specified as specific device types via the normal Front End Table specification (see Figure 2) and as such are subject to device-dependent logic. For example, the user may define a GFE device as a CRT. Front End logic will then send one output 'screen' at a time, requiring an input before sending a second output message. The GFE user should be familiar with the standard Front End logic for the device type to be simulated (see BTAM Terminal Support Guide).

Each GFE Front End function must be completed prior to returning to the calling program. In other words, the user may not schedule execution of the function to be performed via the DISPATCH macro and then return to GFE before that operation completes.

GFE routines must not alter any Intercomm Front End tables. GFE is designed as a logical extension to Intercomm's existing BTAM Front End. The same macros are used for GFE as for any BTAM device. The user's GFE routines are treated as Front End system subroutines which are expected to perform certain predefined functions.

In order to properly use GFE, the user must understand how to code the Front End tables to define a GFE line group. He must also understand, for each user entry point called by GFE:

1. Why the entry point is receiving control.

2. What information is available

   - From Intercomm-defined sources.

   - From user-defined sources.

3. What information must be returned upon completion.

The following chapters provide that information.

# Chapter 6

## BTAM FRONT END TABLES FOR GFE DEVICES

A GFE line group is defined within the normal Intercomm BTAM table structure. The macros defining standard supported devices are used to define GFE devices as well:

- LINEGRP: generates a DCB.
- BLINE: generates a DECB plus extension for Intercomm control fields for each line.
- BTERM: generates Intercomm control fields for each terminal.
- BDEVICE: generates Intercomm control fields for each terminal type.

Refer to the BTAM Terminal Support Guide for a detailed description of Front End tables, and Basic System Macros for detailed coding specifications for all macros. See Figure 6 for a listing of the Intercomm Dsects associated with these macros.

A GFE macro is required for each set of user routines composing an access method for GFE devices. The GFE macro generates a vector table which specifies the user's entry point(s) and any additional pointers required by the user. Either one entry point may be used for all functions (ENTRY parameter) or multiple entry points may be specified (VERIFY, STARTUP, ...). A routine code is passed as a parameter to identify the function required when one entry point is used. Following the entry points, additional pointers required by the user can be generated through the use of the ADCONS (for A-type addresses) and VCONS (for V-type addresses) parameters.

Up to 11 individual functions may be defined as summarized previously in Figure 3. If no entry point is defined for a particular function it is assumed that no user action was required at that time and a default return code will be assumed. If no OPEN function is specified, the line group will not be started.

Three other operands complete the specification of a GFE device:

1. LINEGRP macro, UNIT=GFE operand
2. BLINE macro, UNIT=GFE operand
3. BLINE macro, GFEVECT=label-of-GFE-macro

Figure 7 illustrates GFE Front End tables; Figures 8, 9, and 10 illustrate typical coding of GFE macros.

```
NETWORK   CSECT

*NETWORK CONFIGURATION TABLE
LG1       LINEGRP    UNIT=GFE,.....

LG2       LINEGRP    UNIT=GFE,.....

          BLINE      UNIT=GFE,GFEVECT=GFEDEV1,LGNAME=LG1,.....

          BTERM      ....
          BTERM         Terminal Definitions
           .
           .
           .

          BLINE      UNIT=GFE,GFEVECT=GFEDEV2,LGNAME=LG2,.....

          BTERM      ....
           .            Terminal Definitions
           .
          PMISTOP       (required)
          BDEVICE    ....
           .            Device Type Definitions
           .
*GFE MACRO, ONE ENTRY POINT, "ACCESS METHOD 1"

GFEDEV1   GFE        ENTRY=AMTYPE1

*GFE MACRO, MULTIPLE ENTRY POINTS, "ACCESS METHOD 2"

GFEDEV2   GFE        STARTUP=AM2START,CLOSE=AM2CLOSE,OPEN=AM2OPEN,   X
                READ=AM2READ,WRITE=AM2WRITE,RESET=AM2RESET

           .
           .
NETWORK   CSECT
          ENTRY      BTVTBEND
BTVTBEND  EQU        *
          END
```

Figure 7. Front End Network Table Structure With GFE

## Coding the GFE Macro

The GFE macro is referenced by a BLINE macro to define the user's Generalized Front End entry points and any additional addresses which may be needed to support the user's nonstandard Front End.

The forms of the GFE macro are:

| | | |
|---|---|---|
| [symbol] | GFE | if only one entry point is used:<br>ENTRY=single-entry-point<br>[,ADCONS=(adcon1,adcon2,...)]<br>[,VCONS=(vcon1,vcon2,...)] |
| [symbol] | GFE | if multiple entry points are used:<br>[VERIFY=verify-entry]<br>[,OPEN=open-entry]<br>[,STARTUP=startup-entry]<br>[,READ=read-entry]<br>[,WRITE=write-entry]<br>[,RESET=reset-entry]<br>[,UP=TPUP-entry]<br>[,DOWN=TDWN-entry]<br>[,STRTLNE=STLN-entry]<br>[,STOPLNE=SPLN-entry]<br>[,CLOSE=closedown-entry]<br>[,ADCONS=(adcon1,adcon2,...)]<br>[,VCONS=(vcon1,vcon2,...)] |
| [symbol] | GFE | if Intercomm-supported GFE Front End used:<br>TYPE=GFE-type-specification |

The parameters are described below.

ENTRY
specifies a single entry point to be called for all GFE functions. If this parameter is specified, all other entry point specifications are ignored.

VERIFY
specifies an entry point called during start-up to perform GFE table verification.

OPEN
specifies an entry point (called during startup or when a STLG command is being processed) to perform initialization for the GFE line group. If this parameter is omitted, the GFE line group is not activated.

STARTUP
specifies an entry point called during startup to perform initialization for a GFE line and determine whether to dispatch this particular line.

READ
specifies an entry point called to perform an input operation.

WRITE
specifies an entry point called to perform an output operation.

RESET
specifies an entry point called to cancel an outstanding READ operation.

UP
specifies an entry point called when a TPUP command is being processed.

DOWN
specifies an entry point called when a TDWN command is being processed.

STARTLNE
specifies an entry point called when a STLN command is being processed.

STOPLNE
specifies an entry point called when a SPLN command is being processed.

CLOSE
specifies an entry point called (during closedown or when a SPLG command is being processed) to perform termination functions for the GFE line group.

TYPE

specifies that a GFE Front End is of a type which is supported by Intercomm and identifies that Front End. At present, the only valid specification is TYPE=TCAM, which identifies Extended TCAM Support.

ADCONS

specifies one or more A-type addresses which will be generated following the list of entry points. These are entirely user-determined.

VCONS

specifies one or more V-type addresses which will be generated following the ADCONS (if any). These are entirely user-determined.

```
  *
  *       EXAMPLE OF A GFE MACRO DEFINING A TAPE I/O MODULE
  *       A SINGLE ENTRY POINT IS USED AND THE MODULE WILL USE THE
  *       FILE HANDLER TO PERFORM TAPE READS AND WRITES.
  *
  TAPEMOD GFE   ENTRY=TAPEMOD

 +*  GENERALIZED FRONT END INTERFACE VECTOR TABLE

 +TAPEMOD DC    A(0)     .        START OF USER ENTRIES
 +        DC    AL1(0)      .        GFE TYPE CODE
 +        DC    B'10000000'                               FLAGS
 +        DC    V(TAPEMOD)  .         SINGLE ENTRY POINT USED
```

Figure 8. GFE Macro for a Tape Device

```
 *
 *       EXAMPLE OF A GFE MACRO DEFINING SUPPORT FOR A NON-STANDARD
 *       TELEPROCESSING TERMINAL
 *
 NSDEVICE GFE   OPEN=NSOPCL,CLOSE=NSOPCL,  ENTRY FOR OPEN/CLOSE        @
                UP=NSUPDN,DOWN=NSUPDN,     ENTRY FOR TPUP/TDWN         @
                READ=NSRDWR,WRITE=NSRDWR,  ENTRY FOR READ/WRITE        @
                VERIFY=NSVERIFY,           ENTRY TO VERIFY USER TABLES @
                ADCONS=(NSDCB,NSLINE)      POINTERS TO USER'S DCB,DECB

+*  GENERALIZED FRONT END INTERFACE VECTOR TABLE

+NSDEVICE DC    A(GFE0002) .        START OF USER ENTRIES
+         DC    AL1(0)     .        GFE TYPE CODE
+         DC    B'00000000'                                FLAGS

+         DC    V(NSVERIFY)  .        GFE ENTRY POINT
+         DC    A(0) .              ENTRY UNUSED
+         DC    V(NSOPCL) .         GFE ENTRY POINT
+         DC    V(NSRDWR) .         GFE ENTRY POINT
+         DC    V(NSRDWR) .         GFE ENTRY POINT
+         DC    A(0) .              ENTRY UNUSED
+         DC    V(NSUPDN) .         GFE ENTRY POINT
+         DC    V(NSUPDN) .         GFE ENTRY POINT
+         DC    A(0) .              ENTRY UNUSED
+         DC    A(0) .              ENTRY UNUSED
+         DC    V(NSOPCL) .         GFE ENTRY POINT
+GFE0002  DC    OF .                USER SPECIFIED ADDRESSES
+         DC    A(NSDCB)      .     USER GFE ADCON
+         DC    A(NSLINE)     .     USER GFE ADCON
          PRINT NOGEN

 *        DCB FOR NON-STANDARD TP
 NSDCB    DCB   DDNAME=NSDD,DSORG=CX,MACRF=(R,W)

 *        DECB FOR NON-STANDARD READS/WRITES
          READ  NSLINE,TI,NSDCB,MF=L
```

Figure 9. GFE Macro for a Terminal Device

```
 *
 *        EXAMPLE OF A GFE MACRO DEFINING EXTENDED TCAM SUPPORT
 *
          GFE   TYPE=TCAM,ADCONS=(TCAMREAD,TCAMIN,TCAMWRIT,TCAMOUT)

+*  GENERALIZED FRONT END INTERFACE VECTOR TABLE

+         DC    A(GFE0008) .         START OF USER ENTRIES
+         DC    AL1(1)     .         GFE TYPE CODE
+         DC    B'00000000'                                FLAGS

+         DC    V(TCAMVER) .          GFE ENTRY POINT
+         DC    A(0) .               ENTRY UNUSED
+         DC    V(TCAMOPEN) .          GFE ENTRY POINT
+         DC    V(TCAMREAD) .          GFE ENTRY POINT
+         DC    V(TCAMWRIT) .          GFE ENTRY POINT
+         DC    A(0) .               ENTRY UNUSED
+         DC    A(0) .               ENTRY UNUSED
+         DC    A(0) .               ENTRY UNUSED
+         DC    V(TCAMOPEN) .          GFE ENTRY POINT
+         DC    V(TCAMCLOS) .          GFE ENTRY POINT
+         DC    V(TCAMCLOS) .          GFE ENTRY POINT
+GFE0008  DS    OF .                 USER SPECIFIED ADDRESSES
+         DC    A(TCAMREAD)     .    USER GFE ADCON
+         DC    A(TCAMIN)       .    USER GFE ADCON
+         DC    A(TCAMWRIT)     .    USER GFE ADCON
+         DC    A(TCAMOUT)      .    USER GFE ADCON

          PRINT NOGEN
 *        TCAM INPUT DCB
 TCAMIN   DCB   DDNAME=TCAMIN,BLKSIZE=1000,OPTCD=WCU,RECFM=V,DSORG=PS, @
                MACRF=R

 *        TCAM OUTPUT DCB
 TCAMOUT  DCB   DDNAME=TCAMOUT,OPTCD=WCU,RECFM=V,DSORG=PS,             @
                MACRF=W

 *        TCAM INPUT DECB
          READ  TCAMREAD,SF,TCAMIN,MF=L

 *        TCAM OUTPUT DECB
          WRITE TCAMWRIT,SF,TCAMOUT,MF=L
```

Figure 10.   GFE Macro for Extended TCAM Support

# Chapter 7

# CODING GFE ROUTINES

Once the GFE device(s) are defined with respect to the BTAM Front End tables via the LINEGRP, BLINE, BTERM, BDEVICE and GFE macros, it remains to code the actual routines composing the access method for a particular device type.

This chapter presents a discussion of

1. Message formats

2. GFE DSECTS

3. Linkage conventions to user GFE routines

4. GFE routine usage

The parameter list passed to the various user entry points includes pointers to a BLINE or LINEGRP and in some cases a BTERM macro. These control blocks are passed to the user entry point for reference; their usage is discussed below. However, they may not be modified by the user's routines. Moreover, the DCB within the LINEGRP and the DECB within the BLINE are simulated control blocks which are reserved for Intercomm's internal use. If the user requires a DCB, DECB or other additional tables, they must be independently created. Two keywords of the GFE macro (ADCONS, VCONS) are provided whose expected usage is to generate user pointers to any required tables and/or control blocks.

## MESSAGE FORMATS

Input and output messages are formatted as a standard variable length logical record, that is, message text preceded by a fullword Record Descriptor Word (RDW), the first two bytes providing the length of the entire area. Input and output is expected to be in EBCDIC since GFE will not perform translation. However, a translate table may be referenced by the BLINE macro, TRSTBL parameter, if the user wishes to perform translation in the user READ and WRITE routines. End of transmission characters for the device type are required on input, and will be provided on output depending on the device-type definition of the terminal.

Line-control characters are defined via the BDEVICE macro (STCHAR and CTCHAR parameters), or they may be inserted by the user in the WRITE routine.

If output messages must be modified, two approaches are possible.

1. The output record RDW is built by GFE within the output message header. The address passed to the user points to the RDW. The user may save and restore up to 36 bytes preceding the RDW for use when writing the output message.

2. The user may obtain an area, reformat the output and write from the new area. The user must then free the area obtained.

In either case the 36 bytes preceding the RDW (first 36 bytes of the Intercomm message header) must be returned to GFE unaltered.

All input is treated as a full message by default. However, segmented message processing may be handled by communicating the segment type to GFE in the GFEQPR field of the parameter list. The valid settings of this field and their meanings are as follows:

| GFEQPR | Segment Type |
|---|---|
| X'00' | First or Header |
| X'01' | Intermediate or Detail |
| X'02' | Full Message |
| X'03' | Last or Final |

Similarly, on output, GFEQPR provides the segment type of the message to the user. If the user cannot interleave segments to one terminal with output for other terminals, he can specify to continue with the same terminal by placing a code of X'04' in GFECODE1 (byte 1 of the return code field). In this way all segments of an output message will be provided to the user before going on to the next terminal having output. If, for some reason, an expected segment does not arrive, the user will be given control with an output message address of 0 and may take some form of corrective action.

## GFE DSECTS

A GFEDSECT macro is provided to generate DSECTs or in-line fields for referencing the GFE vector table (created by the GFE macro) and the GFE parameter list (passed to user's entry points).

The form of the GFEDSECT macro is:

| | | |
|---|---|---|
| [symbol] | GFEDSECT | [PARMS={INLINE}]<br>[ {DSECT} ]<br><br>[,VECTORS={INLINE}]<br>[ {DSECT }] |

PARMS

INLINE will create the fields in the GFE parameter list. DSECT will generate the parameter list preceded by a GFEPARMS DSECT statement.

VECTORS

INLINE will create the fields in the GFE vector table. DSECT will create the vector table preceded by a GFEDSECT DSECT statement.

The Dsects are illustrated in Figures 11 and 12.

```
                USING *,2
 *
 *              SAMPLE GFE MACROS
 *
 *

 *
 *              LISTING OF DSECT FOR GFE PARAMETER LIST
 *
          GFEDSECT PARMS=DSECT
+GFEPARMS DSECT                 GFE PARAMETER LIST DSECT
+GFEUSERV DS    F               ADDRESS OF USER'S GFE ADDRESSES
+GFECODES DS    OF              RETURN CODE AREA
+GFECODE1 DS    C               BYTE 1
+GFENXTID EQU   0               NORMAL OUTPUT SEQUENCE
+GFETIDUP EQU   4               SAME TERMINAL IF UP
+GFETIDWN EQU   8               SAME TERMINAL IF UP OR DOWN
+GFECODE2 DS    C               BYTE 2
+GFECODE3 DS    C               BYTE 3
+GFECODE4 DS    C               BYTE 4
+GFENORM  EQU   0               SUCCESSFUL
+GFEABNRM EQU   4               UNSUCCESSFUL
+GFETDOWN EQU   4               PUT TERMINAL DOWN
+GFELDOWN EQU   8               STOPLINE RETURN CODE
+GFENEGRS EQU   12              NEGATIVE RESPONSE TO READ
+GFELG    DS    OF              LINEGRP ADDRESS
+GFEQPR   DS    OC              QPR FOR MESSAGE SEGMENTING.
+GFEBLINE DS    F               BLINE ADDRESS
+GFEBTERM DS    F               BTERM ADDRESS
+GFEINPUT DS    OF              INPUT ADDRESS
+GFEOUTPT DS    F               OUTPUT ADDRESS
+GFEPARML EQU   *-GFEPARMS      PARAMETER LIST LENGTH
```

Figure 11. GFEDSECT Macro for GFE Parameters Dsect

```
*
*         LISTING OF DSECT FOR GFE VECTOR TABLE
*
          GFEDSECT VECTORS=DSECT
+GFEDSECT DSECT        GFE VECTOR TABLE DSECT
+GFEUSER  DS     F     POINTER TO USER ADDRESSES
+GFETYPE  DS     C     TYPE OF GFE BEING USED
+GFETCAM  EQU    1     GFE IS TCAM
+GFEFLAG1 DS     B     FLAG BYTE
+GFEONEEP EQU    X'80' SINGLE ENTRY POINT USED
+*                     OTHERWISE, ONE ENTRY PER FUNCTION CODE
+         DS     H     RESERVED
+GFENTRYS DS     OF    START OF ENTRY POINT(S)
+*        GFE FUNCTION CODES
+GFEVERFY EQU    0     CODES
+GFESTART EQU    1     DEFINING
+GFEOPEN  EQU    2     FUNCTIONS
+GFEREAD  EQU    3     TO
+GFEWRITE EQU    4     BE
+GFERESET EQU    5     PERFORMED
+GFEUP    EQU    6     BY
+GFEDOWN  EQU    7     USER
+GFESTLN  EQU    8     ENTRY
+GFESPLN  EQU    9     POINT
+GFECLOSE EQU    10
+         DS     OF    USER'S GFE FIELDS BEGIN HERE
```

Figure 12. GFEDSECT Macro for GFE Vector Table

## LINKAGE CONVENTIONS

The register usage at linkage to each entry point is as follows:

```
R13 = save area address
R14 = return point
R15 = user entry point address
R0  = entry's function code
R1  = parameter list address
```

The function code in register 0 identifies what action the user's entry point is expected to perform. The parameter list contains pointers to the appropriate data and control blocks plus other codes and indicators. It is used to pass information from GFE to the user as well as to pass information from the user back to GFE. In short, it is the chief interface between GFE and the user.

The remainder of this section describes, for each defined user GFE function, what module invokes it, what action is expected to be performed, what information is provided to the user in the parameter list, and what must be returned to GFE in the parameter list. Upon return from the user, GFE will check the completion status and, depending on the results, different actions may be taken by the Intercomm Front End as discussed below.

In the case where a single entry point is used (ENTRY=), the same user entry point receives control for all GFE functions. At that entry point, the user determines which action to perform by examining the function code passed in register 0. After this, the logical processing done is the same as if an individual entry point had been specified. A similar approach can be taken if more than one entry point is specified, but a specific entry point may be shared by more than one function.

The GFE parameter list is from one to five words, depending on the function called. For all functions, the usage of words 1 and 2 of the parameter list remains constant:

WORD 1 Address of the user extension to the GFE vector table. This user extension was created by coding ADCONS= and/or VCONS=. If both were coded, the ADCONS are generated first.

WORD 2 Return code area, consisting of four bytes, initialized to binary zeros. The user's return codes are placed here. These will be referred to below as CODE 1, CODE 2, CODE 3, CODE 4. CODE 2 and CODE 3 are currently unused and must remain zeros.

For simplicity, therefore, only the usage of CODE 1, CODE 4, WORD 3, WORD 4 and WORD 5 will be defined below. Figure 11 describes the Dsect for the parameter list (see also GFEDSECT macro).

An asterisk (*) next to a return code indicates the default assumed by GFE when there is no user entry point defined for a particular function.

Appendix A provides a summary chart of parameters for each function.

## INITIALIZATION ENTRY POINTS

### The VERIFY Entry Point

Module calling GFE:

BTVERIFY Function Code: 0

Parameters Passed:

WORD 3 - LINEGRP address
WORD 4 - Not used
WORD 5 - Not used

User Action:

Perform consistency checking only on the user's own tables, if any.

Information Returned:

| CODE 4 | Meaning | BTVERIFY Action |
|---|---|---|
| 0* | Successful | Continue Normally |
| 4 | Unsuccessful | ABEND 599 |

## The STARTUP Entry Point

NOTE: STARTUP entry is called after OPEN entry.

Module Calling GFE:

BTAMLINE Function Code: 1

Parameters Passed:

WORD 3 - BLINE address
WORD 4 - Not Used
WORD 5 - Not Used

User Action:

Decide whether a line handler should be started for this BLINE.

Information Returned:

| CODE 4 | Meaning and BTAMLINE Action |
|---|---|
| 0* | Start line handler |
| 4 | Do not start line handler |

## The OPEN Entry Point

The OPEN entry is called for each GFE line group before STARTUP entry at system initialization. OPEN entry is also called when a STLG command is entered for a GFE line group.

Modules calling GFE:

| | | |
|---|---|---|
| BTAMLINE | Function Code: 2 | (startup) |
| TPUMSG | Function Code: 2 | (STLG command) |

Parameters Passed:

WORD 3 - LINEGRP address
WORD 4 - Not Used
WORD 5 - Not Used

User Action:

Major initialization logic should be performed here, for example, initialize tables, open DCB, etc.

Information Returned:

| Code 4 | Meaning | BTAMLINE/TPUMSG Action |
|---|---|---|
| 0 | Initialization successful | Flag line group active. (Proceed to STARTUP entry.) |
| 4* | Initialization unsuccessful | Flag all lines inactive. ( Do not call STARTUP entry.) |

NOTE: If no OPEN entry is provided, the default CODE 4 value is 4. Therefore, OPEN is required to permit further processing for a line group.

## I/O ACCESS ENTRY POINTS

### READ, WRITE and RESET Logic

When designing the READ, WRITE and RESET entry points, the user should understand how BLHIN and BLHOT interact with each other. Figures 13 and 14 illustrate the conceptual flow of BLHIN and BLHOT with respect to these functions. This should be thoroughly understood by the GFE user. In particular, notice:

- that output has priority over input.
- the use of the RESET function.

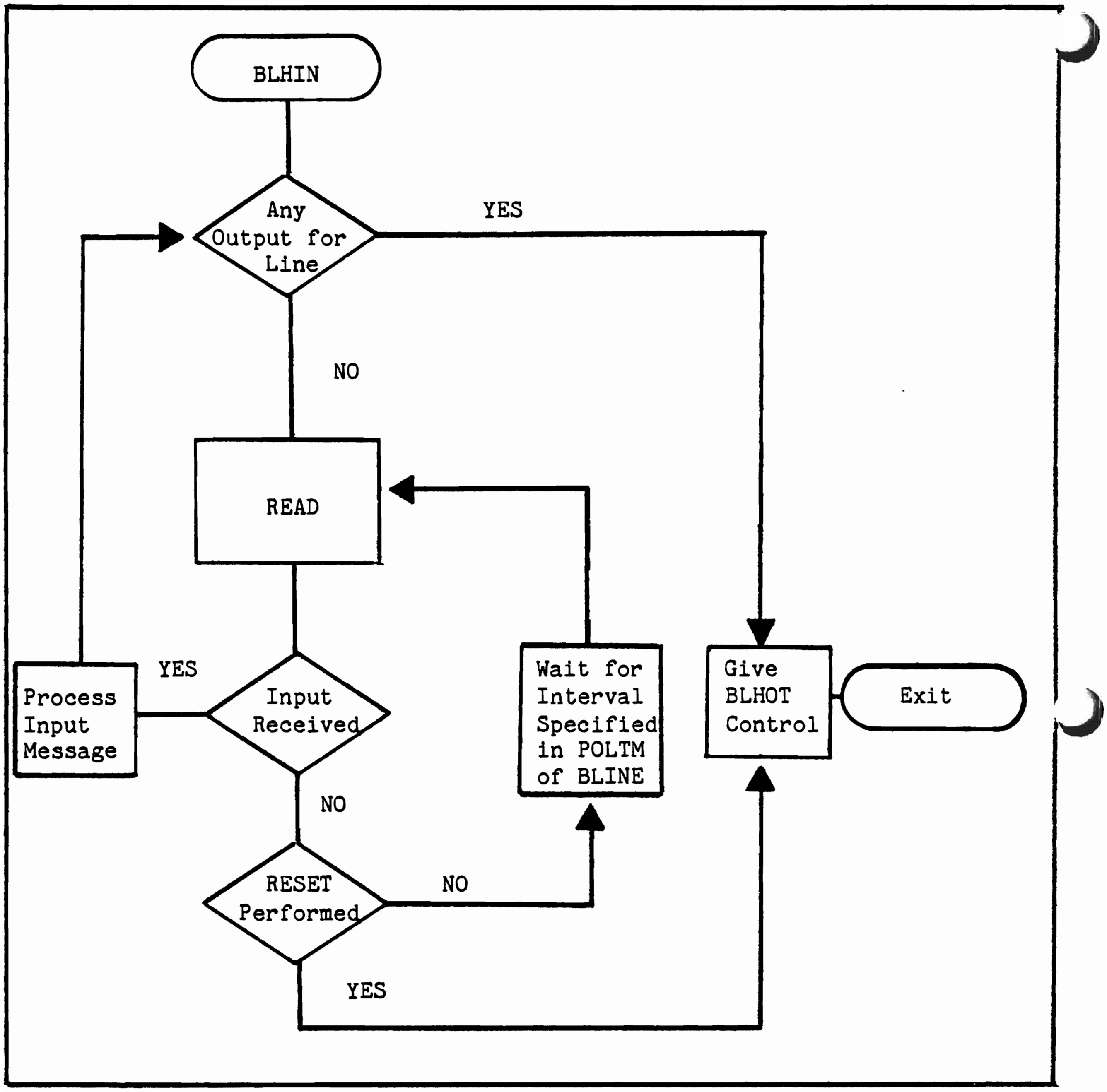
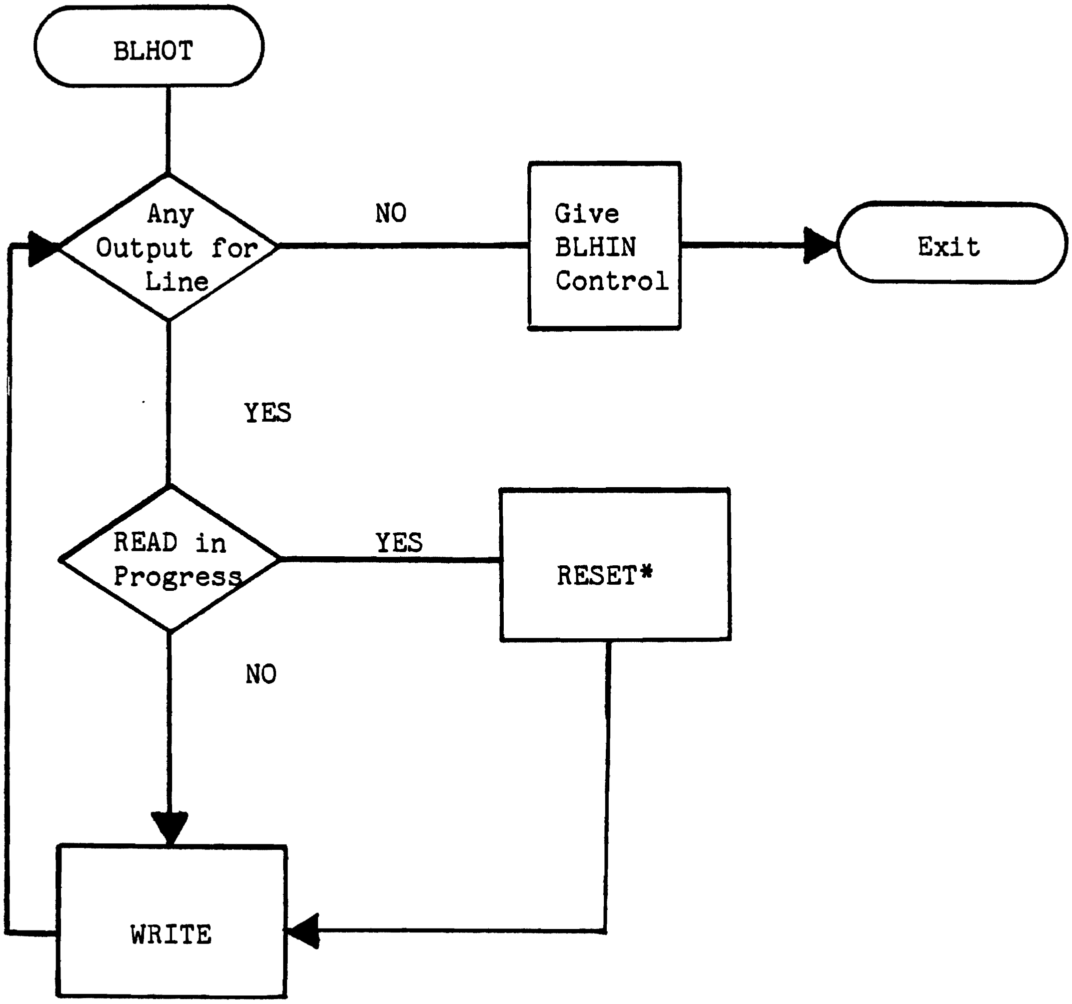


Figure 13. Conceptual Flow of BLHIN



*This requests that the READ operation be cancelled if data transfer has not yet begun.

Figure 14. Conceptual Flow of BLHOT

## The READ Entry Point

Input Segment Processing:

Normally, input is assumed to be a full message. If the user wishes to have input flagged as segments of a message, the GFEQPR field, a one-byte field in the parameter list, must be used to indicate the segment type (see Figure 11). Segment type will be placed in the MSGHQPR field of the input message. The segment types are defined as follows:

X'00' - Header or first segment
X'02' - Full message (default segment type)
X'01' - Detail or intermediate segment
X'03' - Trailer or final segment

Input segments will be queued by Intercomm as they are received. Therefore, the GFE user who wishes to use input segments must be careful to prevent interleaving of input segments from a particular BTERM with segments or full messages from the same or another BTERM.

Module calling GFE:

BLHIN Function Code: 3

Parameters Passed:

WORD 3 - BLINE address
WORD 4 - 0 - Reserved for BTERM address
WORD 5 - 0 - Reserved for MSG address

User Action:

Obtain a single input message by whatever mechanism is being used and format it correctly for GFE input. (See "Message Formats").

Information Returned:

| CODE 4 | Meaning | BLHIN Action |
|---|---|---|
| 0 | Input received | Process input |
| 4 | Error | Put terminal down |
| 8 | Error | Put line down |
| 12* | No Input | Send output, if any. If none, wait for time interval specified in POLTM, then call READ again. |

GFEQPR - Segment code. See "user action" above. If not initialized, this field will default to X'02', indicating a full message is being returned.

WORD 4 - BTERM address. This must be returned when CODE 4 = 0 or 4.

WORD 5 - Input area address. This must be returned when CODE 4 = 0.

## The WRITE Entry Point

Output Segment Processing:

If output segments are being sent, the user can check the GFEQPR segment type to see whether the output area contains a first, intermediate or final segment. Normally, only one output segment will be presented per BTERM before going to the next BTERM's output. However, the GFE user may stipulate that Intercomm not proceed to the next BTERM, but present output from the same BTERM again. This is done by returning a X'04' in CODE 1. Thus, the user may force all output segments for a given message to be presented with no intervening output for other BTERMs. In the event that an expected output segment does not get queued for the affected BTERM after a reasonable delay, the user's WRITE entry point will be called with WORD 4 (the output address) set to 0. Therefore, users implementing this method of processing must check the output area address to see if it is zero, so that an "expected segment missing" condition may be detected and appropriate action taken.

Module calling GFE:

BLHOT Function Code: 4

Parameters Passed:

GFEQPR - Segment type. Taken from MSGHQPR field of Intercomm message header. See "Message Formats" for a definition of segment types.

WORD 3 - BLINE address

WORD 4 - BTERM address

WORD 5 - Output area address or 0

User Action:

Transmit the output message by whatever mechanism is being used. Output area format is discussed in "Message Formats."

Information Returned:

| CODE 1 | Meaning |
|---|---|
| 0* | Normal output dequeuing sequence |
| 4 | Present next output for same BTERM<br>(cannot be used if CRT=YES was coded for the BTERM) |

| CODE 4 | Meaning | BLHOT Action |
|---|---|---|
| 0* | WRITE successful | Continue as defined in CODE 1 |
| 4 | Error | Put terminal down |
| 8 | Error | Put line down |

The RESET Entry Point

Module calling GFE:

BTSEARCH Function Code: 5

Parameters Passed:

WORD 3 - BLINE address
WORD 4 - Not used
WORD 5 - Not used

User Action:

Cancel an incomplete read operation if data transfer has not yet begun. This is necessary when output is ready to be sent but the line is unavailable because it is awating input. This entry point is logically equivalent to a BTAM RESETPL macro. The use of this function depends entirely upon the method of data transfer being used. It is available, though may not be needed. If used, it will include some means of communicating to the user's READ routine that a read should be cancelled.

Information Returned:

| CODE 4 | Meaning |
|---|---|
| 0* | Normal return from the RESET function. When the READ function is interrupted, BLHIN will receive control and will post BLHOT to allow a WRITE. |
| 4 | GFEINTFC will dispatch a routine which will post BLHOT to allow a WRITE. The WRITE function may then receive control before the READ function completes. A GFE user who requires this facility is responsible for overlapping the READ and WRITE logic. |

## DEVICE CONTROL ENTRY POINTS

### The UP and DOWN Entry Points

Module calling GFE:

TPUMSG Function Code: UP - 6, DOWN - 7

Parameter list:

WORD 3 - BLINE address<br>
WORD 4 - BTERM address<br>
WORD 5 - not used

User Action:

These entries are used to notify the user that a TPUP or TDWN (UP or DOWN, respectively) command was entered for the BTERM referenced in the parameter list. The user performs whatever processing may be needed at the time a particular terminal is activated or deactivated.

NOTE: The initial status of the terminal is defined via the BTERM macro, TPUP parameter.

Information Returned:

None.

## The STRTLNE and STOPLNE Entry Points

Module calling GFE:

TPUMSG Function Code: STRTLNE-8, STOPLNE-9

Parameter List:

WORD 3 - BLINE address<br>
WORD 4 - BTERM address<br>
WORD 5 - not used

User Action:

> These entry points are used to notify the user that a STLN or SPLN (STRTLNE or STOPLNE, respectively) command was entered for the line referenced in the parameter list. The user can perform processing related to activating or deactivating that line. It is important to realize that this will cause the user's UP or DOWN entry, respectively, to be given control once for each terminal on the line. The STRTLNE entry provides the user the ability to set a return code determining whether the line can in fact be started.

Information Returned:

| Function | CODE 4 | TPUMSG Action |
|---|---|---|
| STRTLNE | 0* | Start the line handler |
| | 4 | Do not start line handler |
| STOPLNE | None | |

## The CLOSE Entry Point

The CLOSE entry is called at system closedown for each GFE line group after outstanding messages are written (if possible). The CLOSE entry is also called if a SPLG command is issued for a GFE line group.

Module calling GFE:

TPUMSG Function Code: 10 (SPLG command)
BTAMCLSE Function Code: 10 (system closedown)
NOTE: BTAMCLSE is a Csect in BTAMLINE

Parameter List:

WORD 3 - LINEGRP address
WORD 4 - BTERM address
WORD 5 - Not used

User Action:

Perform any termination processing required for the GFE line group. Close DCBs, free resources, etc.

Information Returned:

None.

# Chapter 8

## GFE IMPLEMENTATION

In addition to coding tables and user routines as described in previous sections, the following steps are required for implementation of GFE:

1. Update the member SETENV as follows:

```
&GFE   SETB  1                    GENERALIZED FRONT END IN USE
```

   and set any device type globals used by GFE terminals, if not already initialized in SETENV (that is, &IBM3270, &IBM2260, etc.).

2. Reassemble and linkedit:

   - BLHIN
   - BLHOT
   - BTAMLINE
   - BTSEARCH
   - BTVERIFY
   - TPUMSG
   - Network Table

3. Generate a linkedit deck using the ICOMLINK macro with GFE=YES in addition to any other required specification for Intercomm features (generates an INCLUDE for GFEINTFC).

4. Add INCLUDE cards for any required user GFE routines and tables.

5. Verify the presence of INCLUDE cards for all application-oriented tables, user routines, and Intercomm routines.

6. Linkedit

7. Execute

   There are no messages, snaps, or abends associated with GFE.

Additional installation specifications for Extended TCAM Support via GFE are described in the TCAM Support Users Guide.

# Appendix A

## SUMMARY OF GFE INTERFACE PARAMETERS

| Entry<br>Point | Purpose of<br>Entry Point | Code<br>(R0) | Caller | Parameters<br>Wn=WORD n | Return Codes (Word 2, Byte 4)<br>*default return if no entry point |
|---|---|---|---|---|---|
| VERIFY | Validate Tables | 0 | BTVERIFY | W3=LINEGRP | * 0 - Tables Valid<br>4 - Tables not Valid - Abend 599 |
| STARTUP | Start a Line | 1 | BTAMLINE | W3=BLINE | * 0 - Dispatch BLHIN<br>4 - Do not Dispatch BLHIN |
| OPEN | Start Linegrp | 2 | BTAMLINE<br>TPUMSG | W3=LINEGRP | 0 - Linegrp started<br>* 4 - Linegrp not started |
| READ | Read Input | 3 | BLHIN | W3=BLINE<br>W4=BTERM<br>W5=Input message<br>W4 returned with 0, 4<br>W5 returned with 0 | 0 - Input received<br>4 - Error - Put terminal down<br>8 - Error - Put line down<br>*12 - Negative Response |
| WRITE | Write Output | 4 | BLHOT | W3=BLINE<br>W4=BTERM<br>W5=Output message | * 0 - Successful write - See BYTE 1<br>4 - Error - Put terminal down<br>8 - Error - Put line down<br>BYTE 1<br>* 0 - Dequeue from next BTERM<br>4 - Dequeue from same BTERM |
| RESET | Cancel a Read | 5 | BTSEARCH | W3=BLINE | * 0 - Read cancelled, do write<br>4 - Overlap read with write |
| UP | Put Terminal Up | 6 | TPUMSG | W3=BLINE<br>W4=BTERM | ---<br>--- |
| DOWN | Put Terminal Down | 7 | TPUMSG | W3=BLINE<br>W4=BTERM | ---<br>--- |
| STRTLNE | Start Line | 8 | TPUMSG | W3=BLINE<br>W4=BTERM | * 0 - Line Started<br>4 - Line not Started |
| STOPLNE | Stop Line | 9 | TPUMSG | W3=BLINE<br>W4=BTERM | ---<br>--- |
| CLOSE | Close Linegrp | 10 | BTAMCLSE<br>TPUMSG | W3=LINEGRP<br>W4=BTERM | ---<br>--- |

# INDEX

Page

Page

Intercomm GENERALIZED FRONT END FACILITY SDA

# USERS REVIEW FORM

Name ______________________________
Position ____________________________
Company ____________________________

We would appreciate your thoughts on the usefulness and readability of this publication. Please write your comments below. Continue if necessary on the back of this form.

MAIL TO: Manager, Publications Department
SDA Products
71 5th Avenue
New York, New York 10003