INTEGRATED INFORMATION
SUPPORT SYSTEM (IISS)
Volume VIII - User Interface Subsystem
Part 18 - Forms Driven Form Editor Product Specification

General Electric Company
Production Resources Consulting
One River Road
Schenectady, New York 12345

November 1985

Approved for public release; distribution is unlimited.

MATERIALS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AFB, OH 45433-6533

87 7 15 087
NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

DAVID L. JUDSON, PROJECT MANAGER
AFWAL/MLTC
WRIGHT PATTERSON AFB OH 45433

5 Aug 1976

FOR THE COMMANDER:

GERALD C. SHUMAKER, BRANCH CHIEF
AFWAL/MLTC
WRIGHT PATTERSON AFB OH 45433

7 Aug 86

"If your address has changed, or you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/MLTC, W-PAFB, OH 45433 to help us maintain a current mailing list."

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.
This specification establishes the detailed design of a computer program identified as the Forms Driven Form Editor (FDFE). The FDFE is a software tool for creating and initializing form definitions. The FDFE displays a series of screens which request information from the user and visually show the form under construction. Once a form has been completed, the FDFE stores the form definition constructs needed to recreate the form. The stored form can be selected and modified.
PREFACE

This product specification covers the work performed under Air Force Contract F33615-80-C-5155 (ICAM Project 6201). This contract is sponsored by the Materials Laboratory, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Gerald C. Shumaker, ICAM Program Manager, Manufacturing Technology Division, through Project Manager, Mr. David Judson. The Prime Contractor was Production Resources Consulting of the General Electric Company, Schenectady, New York, under the direction of Mr. Alan Rubenstein. The General Electric Project Manager was Mr. Myron Hurlbut of Industrial Automation Systems Department, Albany, New York.

Certain work aimed at improving Test Bed Technology has been performed by other contracts with Project 6201 performing integrating functions. This work consisted of enhancements to Test Bed software and establishment and operation of Test Bed hardware and communications for developers and other users. Documentation relating to the Test Bed from all of these contractors and projects have been integrated under Project 6201 for publication and treatment as an integrated set of documents. The particular contributors to each document are noted on the Report Documentation Page (DD1473). A listing and description of the entire project documentation system and how they are related is contained in document FTR620100001. Project Overview.

The subcontractors and their contributing activities were as follows:

TASK 4.2

<table>
<thead>
<tr>
<th>Subcontractors</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>Reviewer</td>
</tr>
<tr>
<td>D. Appleton Company (DACOM)</td>
<td>Responsible for IDEF support, state-of-the-art literature search</td>
</tr>
<tr>
<td>General Dynamics/ Ft. Worth</td>
<td>Responsible for factory view function and information models</td>
</tr>
<tr>
<td>Subcontractors</td>
<td>Role</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Illinois Institute of Technology</td>
<td>Responsible for factory view function research (IITRI) and information models of small and medium-size business</td>
</tr>
<tr>
<td>North American Rockwell</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Northrop Corporation</td>
<td>Responsible for factory view function and information models</td>
</tr>
<tr>
<td>Pritsker and Associates</td>
<td>Responsible for IDEF2 support</td>
</tr>
<tr>
<td>SofTech</td>
<td>Responsible for IDEF0 support</td>
</tr>
</tbody>
</table>

**TASKS 4.3 - 4.9 (TEST BED)**

<table>
<thead>
<tr>
<th>Subcontractors</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>Responsible for consultation on applications of the technology and on IBM computer technology.</td>
</tr>
<tr>
<td>Computer Technology Associates (CTA)</td>
<td>Assisted in the areas of communications systems, system design and integration methodology, and design of the Network Transaction Manager.</td>
</tr>
<tr>
<td>Control Data Corporation (CDC)</td>
<td>Responsible for the Common Data Model (CDM) implementation and part of the CDM design (shared with DACOM).</td>
</tr>
<tr>
<td>D. Appleton Company (DACOM)</td>
<td>Responsible for the overall CDM Subsystem design integration and test plan, as well as part of the design of the CDM (shared with CDC). DACOM also developed the Integration Methodology and did the schema mappings for the Application Subsystems.</td>
</tr>
</tbody>
</table>
Subcontractors

Digital Equipment Corporation (DEC)

Role
Consulting and support of the performance testing and on DEC software and computer systems operation.

McDonnell Douglas Automation Company (McAuto)

Responsible for the support and enhancements to the Network Transaction Manager Subsystem during 1984/1985 period.

On-Line Software International (OSI)

Responsible for programming the Communications Subsystem on the IBM and for consulting on the IBM.

Rath and Strong Systems Products (RSSP) (In 1985 became McCormack & Dodge)

Responsible for assistance in the implementation and use of the MRP II package (PIOS) that they supplied.

SofTech, Inc.

Responsible for the design and implementation of the Network Transaction Manager (NTM) in 1981/1984 period.

Software Performance Engineering (SPE)

Responsible for directing the work on performance evaluation and analysis.

Structural Dynamics Research Corporation (SDRC)

Responsible for the User Interface and Virtual Terminal Interface Subsystems.

Prime contractors under other projects who have contributed to Test Bed Technology, their contributing activities and responsible projects are as follows:

<table>
<thead>
<tr>
<th>Contractors</th>
<th>ICAM Project</th>
<th>Contributing Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>1701, 2201, 2202</td>
<td>Enhancements for IBM node use. Technology Transfer to Integrated Sheet Metal Center (ISM C)</td>
</tr>
<tr>
<td>Contractors</td>
<td>ICAM Project</td>
<td>Contributing Activities</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>--------------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>Control Data Corporation (CDC)</td>
<td>1502, 1701</td>
<td>IISS enhancements to Common Data Model Processor (CDMP)</td>
</tr>
<tr>
<td>D. Appleton Company (DACOM)</td>
<td>1502</td>
<td>IISS enhancements to Integration Methodology</td>
</tr>
<tr>
<td>General Electric</td>
<td>1502</td>
<td>Operation of the Test Bed and communications equipment.</td>
</tr>
<tr>
<td>Hughes Aircraft Company (HAC)</td>
<td>1701</td>
<td>Test Bed enhancements</td>
</tr>
<tr>
<td>Structural Dynamics Research Corp</td>
<td>1502, 1701,</td>
<td>IISS enhancements to User Interface/Virtual Terminal Interface (UI/VTI)</td>
</tr>
<tr>
<td>(SDRC)</td>
<td>1703</td>
<td></td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SECTION 1.0 SCOPE</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.1 Identification</td>
<td>1-1</td>
<td></td>
</tr>
<tr>
<td>1.2 Functional Summary</td>
<td>1-1</td>
<td></td>
</tr>
<tr>
<td><strong>SECTION 2.0 DOCUMENTS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.1 Reference Documents</td>
<td>2-1</td>
<td></td>
</tr>
<tr>
<td>2.2 Terms and Abbreviations</td>
<td>2-3</td>
<td></td>
</tr>
<tr>
<td><strong>SECTION 3.0 REQUIREMENTS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.1 Structural Description</td>
<td>3-1</td>
<td></td>
</tr>
<tr>
<td>3.1.1 Module Hierarchy</td>
<td>3-1</td>
<td></td>
</tr>
<tr>
<td>3.1.2 Module Descriptions</td>
<td>3-3</td>
<td></td>
</tr>
<tr>
<td>3.1.2.1 FDPE</td>
<td>3-3</td>
<td></td>
</tr>
<tr>
<td>3.1.2.2 PRSCMD</td>
<td>3-3</td>
<td></td>
</tr>
<tr>
<td>3.1.2.3 LISTIT</td>
<td>3-3</td>
<td></td>
</tr>
<tr>
<td>3.1.2.4 VIEW</td>
<td>3-4</td>
<td></td>
</tr>
<tr>
<td>3.1.2.5 FORMS LANGUAGE SOURCE ACCESS MODULES</td>
<td>3-4</td>
<td></td>
</tr>
<tr>
<td>3.1.2.6 EDTMOD</td>
<td>3-5</td>
<td></td>
</tr>
<tr>
<td>3.1.2.7 LISTFM</td>
<td>3-5</td>
<td></td>
</tr>
<tr>
<td>3.1.2.8 INSFRM</td>
<td>3-6</td>
<td></td>
</tr>
<tr>
<td>3.1.2.9 DRPFRM</td>
<td>3-6</td>
<td></td>
</tr>
<tr>
<td>3.1.2.10 EDTWHL</td>
<td>3-6</td>
<td></td>
</tr>
<tr>
<td>3.1.2.11 EDTFLD</td>
<td>3-6</td>
<td></td>
</tr>
<tr>
<td>3.1.2.12 LAYOUT</td>
<td>3-7</td>
<td></td>
</tr>
<tr>
<td>3.1.2.13 SCRMAN</td>
<td>3-7</td>
<td></td>
</tr>
<tr>
<td>3.1.2.14 CHGPOS</td>
<td>3-7</td>
<td></td>
</tr>
<tr>
<td>3.1.2.15 TRNSCR</td>
<td>3-7</td>
<td></td>
</tr>
<tr>
<td>3.1.2.16 TRNSTR</td>
<td>3-8</td>
<td></td>
</tr>
<tr>
<td>3.1.2.17 VALINP</td>
<td>3-8</td>
<td></td>
</tr>
<tr>
<td>3.1.2.18 GTNMFD</td>
<td>3-8</td>
<td></td>
</tr>
<tr>
<td>3.1.2.19 MODFLD</td>
<td>3-8</td>
<td></td>
</tr>
<tr>
<td>3.1.2.20 DELFLD</td>
<td>3-9</td>
<td></td>
</tr>
<tr>
<td>3.1.2.21 INSFLD</td>
<td>3-9</td>
<td></td>
</tr>
<tr>
<td>3.1.2.22 COPFRM</td>
<td>3-9</td>
<td></td>
</tr>
<tr>
<td>3.1.2.23 MODFRM</td>
<td>3-10</td>
<td></td>
</tr>
<tr>
<td>3.1.2.24 FLFMST</td>
<td>3-10</td>
<td></td>
</tr>
<tr>
<td>3.1.2.25 FLSTRC</td>
<td>3-10</td>
<td></td>
</tr>
<tr>
<td>3.1.2.26 FLWHST</td>
<td>3-10</td>
<td></td>
</tr>
<tr>
<td>3.1.2.27 GWHINP</td>
<td>3-11</td>
<td></td>
</tr>
<tr>
<td>3.1.2.28 GTCPFD</td>
<td>3-11</td>
<td></td>
</tr>
<tr>
<td>3.1.2.29 DRFWHL</td>
<td>3-11</td>
<td></td>
</tr>
<tr>
<td>3.1.2.30 MODWHL</td>
<td>3-11</td>
<td></td>
</tr>
<tr>
<td>3.1.2.31 INSWHL</td>
<td>3-12</td>
<td></td>
</tr>
</tbody>
</table>
3.2 Functional Flow ........................................ 3-12
3.3 Interfaces ............................................. 3-13
3.3.1 Application Interface ................................. 3-13
3.3.2 Forms Language Compiler ............................. 3-13
3.3.3 Operating System .................................... 3-14
3.4 Program Interrupts ...................................... 3-14
3.5 Timing and Sequencing Description .................... 3-14
3.6 Special Control Features ............................... 3-14
3.7 Storage Allocation ...................................... 3-14
3.7.1 Data Base Definition .................................. 3-14
3.7.1.1 File Descriptions ................................. 3-14
3.8 Object Code Creation .................................... 3-16
3.9 Adaptation Data ......................................... 3-17
3.10 Detailed Design Description ......................... 3-17
3.10.1 Main Program List .................................... 3-17
3.10.2 Module List ......................................... 3-19
3.10.3 External Routines List ................................ 3-25
3.10.4 Include File List ..................................... 3-28
3.10.5 Where Include File Used List ....................... 3-30
3.10.6 Where External Routine Used List ................. 3-45
3.10.7 Main Program Parts List ............................. 3-63
3.10.8 Module Documentation ............................... 3-74
3.10.9 Include File Description ............................ 3-253
3.10.10 Hierarchy Chart ..................................... 3-268
3.11 Program Listings Comments ............................ 3-379

SECTION 4.0 QUALITY ASSURANCE PROVISIONS ............... 4-1
4.1 Introduction and Definitions ................................ 4-1
4.2 Computer Programming and Test Evaluation .................. 4-1

FIGURES

3-1 FDFE Hierarchy Charts .................................. 3-2
3-2 FDFE Data Flow ......................................... 3-12
3-3 FDFE Interface Diagram .................................. 3-13
SECTION 1

SCOPE

1.1 Identification

This specification establishes the detailed design of a computer program identified as the Forms Driven Form Editor, hereinafter referred to as FDFE. The FDFE is one configuration item of the Integrated Information Support System (IISS) User Interface (UI).

1.2 Functional Summary

The FDFE is a software tool for creating and initializing form definitions. The FDFE displays a series of screens which request information from the user and visually show the form under construction. Once a form has been completed, the FDFE stores the form definition constructs needed to recreate the form. The stored form can be selected and modified.

The runtime UI or UIMS views the FDFE, which is part of the UIDS, as an application program which uses the Form Processor. Data to be selected or stored comes from or is passed to the Common Data Model (CDM) in the integrated implementation; otherwise, a file system is used. The FDFE also interacts with the Forms Language Compiler (FLAN) to translate between the forms language source and the compiled form definition.

The FDFE is a C program which makes extensive use of form language sources and compiled forms, performs interactive user input/output via the Form Processor (FP), and uses the FP to manage the compiled forms. The internal form data structure is the same as that used by the Form Processor.
SECTION 2

DOCUMENTS

2.1 Reference Documents


2.2 Terms and Abbreviations

American Standard Code for Information Interchange: (ASCII), the character set defined by ANSI X3.4 and used by most computer vendors.

Application Interface: (AI), subset of the IISS User Interface that consists of the callable routines that are linked with applications that use the Form Processor or Virtual Terminal. The AI enables applications to be hosted on computers other than the host of the User Interface.

Application Process: (AP), a cohesive unit of software that can be initiated as a unit to perform some function or functions.

Attribute: field characteristic such as blinking, highlighted, black, etc. and various other combinations. Background attributes are defined for forms or windows only. Foreground attributes are defined for items. Attributes may be permanent, i.e., they remain the same unless changed by the application program, or they may be temporary, i.e., they remain in effect until the window is redisplayed.

Device Drivers: (DD), software modules written to handle I/O for a specific kind of terminal. The modules map terminal specific commands and data to a neutral format. Device Drivers are part of the UI Virtual Terminal.

Display List: is similar to the open list, except that it contains only those forms that have been added to the screen and are currently displayed on the screen.

Extended Binary Coded Decimal Interchange Code: (EBCDIC), the character set used by a few computer vendors (notably IBM) instead of ASCII.

Field: two dimensional space on a terminal screen.

Form: structured view which may be imposed on windows or other forms. A form is composed of fields. These fields may be defined as forms, items, and windows.

Form Definition: (FD), forms definition language after compilation. It is read at runtime by the Form Processor.
Forms Definition Language: (FDL), the language in which electronic forms are defined.

Forms Driven Form Editor: (FDFE), subset of the FE which consists of a forms driven application used to create Form Definition files interactively.

Form Editor: (FE), subset of the IISS User Interface that is used to create definitions of forms. The FE consists of the Forms Driven Form Editor and the Forms Language Compiler.

Form Hierarchy: a graphic representation of the way in which forms, items and windows are related to their parent form.

Forms Language Compiler: (FLAN), subset of the FE that consists of a batch process that accepts a series of forms definition language statements and produces form definition files as output.

Form Processor: (FP), subset of the IISS User Interface that consists of a set of callable execution time routines available to an application program for form processing.

Form Processor Text Editor: (FPTE), subset of the Form Processor that consists of software modules that provide text editing capabilities to all users of applications that use the Form Processor.

IISS Function Screen: the first screen that is displayed after logon. It allows the user to specify the function he wants to access and the device type and device name on which he is working.

Integrated Information Support System: (IISS), a test computing environment used to investigate, demonstrate and test the concepts of information management and information integration in the context of Aerospace Manufacturing. The IISS addresses the problems of integration of data resident on heterogeneous data bases supported by heterogeneous computers interconnected via a Local Area Network.

Item: non-decomposable area of a form in which hard-coded descriptive text may be placed and the only defined areas where user data may be input/output.
Message: descriptive text which may be returned in the standard message line on the terminal screen. They are used to warn of errors or provide other user information.

Message Line: a line on the terminal screen that is used to display messages.

Network Transaction Manager: (NTM), IISS subsystem that performs the coordination, communication and housekeeping functions required to integrate the Application Processes and System Services resident on the various hosts into a cohesive system.

Open List: a list of all the forms that have been and are currently open for an application process.

Operating System: (OS), software supplied with a computer which allows it to supervise its own operations and manage access to hardware facilities such as memory and peripherals.

Page: instance of forms in windows that are created whenever a form is added to a window.

Paging and Scrolling: a method which allows a form to contain more data than can be displayed with provisions for viewing any portion of the data buffer.

Physical Device: a hardware terminal.

Qualified Name: the name of a form, item or window preceded by the hierarchy path so that it is uniquely identified.

Subform: a form that is used within another form.

User Data: data which is either input by the user or output by the application programs to items.

User Interface: (UI), IISS subsystem that controls the user's terminal and interfaces with the rest of the system. The UI consists of two major subsystems: the User Interface Development System (UIDS) and the User Interface Management System (UIMS).
User Interface Development System: (UIDS), collection of IISS User Interface subsystems that are used by applications programmers as they develop IISS applications. The UIDS includes the Form Editor and the Application Generator.

User Interface Management System: (UIMS), the runtime UI. It consists of the Form Processor, Virtual Terminal, Application Interface, the User Interface Services and the Text Editor.

User Interface Monitor: (UIM), part of the Form Processor that handles messaging between the NTM and the UI. It also provides authorization checks and initiates applications.

User Interface Services: (UIS), subset of the IISS User Interface that consists of a package of routines that aid users in controlling their environment. It includes message management, change password, and application definition services.

User Interface/Virtual Terminal Interface: (UI/VTI), another name for the User Interface.

Virtual Terminal: (VT), subset of the IISS User Interface that performs the interfacing between different terminals and the UI. This is done by defining a specific set of terminal features and protocols which must be supported by the UI software which constitutes the virtual terminal definition. Specific terminals are then mapped against the virtual terminal software by specific software modules written for each type of real terminal supported.

Window: dynamic area of a terminal screen on which predefined forms may be placed at run time.

Window Manager: a facility which allows the following to be manipulated: size and location of windows, the device on which an application is running, the position of a form within a window. It is part of the Form Processor.
SECTION 3
REQUIREMENTS

3.1 Structural Description

The general approach is to view the FDFE as a hierarchy of modules. The FDFE screens presented in the appendix are associated with only certain modules in the hierarchy based on the functionality being performed by the module.

3.1.1 Module Hierarchy

The following hierarchy chart shows the organization of the FDFE:
Figure 3-1  FDFE Hierarchy Charts
3.1.2 Module Descriptions

The following paragraphs describe the modules associated with each of the major sections of the FDFE.

3.1.2.1 **FDFE:**

This module is the main driver. It allows the user to choose among several file management options or to proceed to the edit task level. It controls the WRKTASK screen.

**Input Parameters:**
None

**Output Parameters:**
None

3.1.2.2 **PRSCMD:**

This module parses the command line for both the WRKTASK and EDITTASK screens to determine which other modules are to be called and what parameters are to be passed.

**Input Parameters:**
- Pointer to command line
- Array of parameter counts
- Array of pointers to valid commands

**Output Parameters:**
- Option chosen
- Pointer to parameters to be passed to the module which will execute the option.
- Number of parameters found in command line

3.1.2.3 **LISTIT:**

This module lists all of the Forms Definition Object or Forms Language Source in the user's specified forms language source or definition object libraries.

**Input Parameters:**
- Pointer to "FDL" or "FD" string

**Output Parameters:**
- Returns any error code or a NULL pointer if successful
3.1.2.4 **VIEW:**

This module displays a form just as it would appear on the screen when used by a program.

**Input Parameters:**
- Pointer to name of form to view

**Output Parameters:**
- Returns any error code or a NULL pointer if successful

3.1.2.5 **FORMS LANGUAGE SOURCE ACCESS MODULES:**

- **UNLINK:**
  - This module drops a particular Forms Language Source.
  
  **Input Parameters:**
  - Name of Forms Language Source
  
  **Output Parameters:**
  - Returns any error code or a NULL pointer if successful

- **COPY:**
  - This module copies a particular Forms Language Source to another Forms Language Source of specified name.
  
  **Input Parameters:**
  - Name of existing "from" Forms Language Source
  - Name of new or "to" Forms Language Source
  
  **Output Parameters:**
  - Returns any error code or a NULL pointer if successful

- **RENAME:**
  - This module renames a particular Forms Language Source to a specified name.
  
  **Input Parameters:**
  - Name of existing Forms Language Source
  - New name of Forms Language Source
  
  **Output Parameters:**
  - Returns any error code or a NULL pointer if successful
GETFLS:

This module retrieves a particular Forms Language Source and translates it into the internal data structure.

Input Parameters:
- Name of existing Forms Language Source

Output Parameters:
- Pointer to the "opened" form, the internal Forms Processor data structure

SAVFLS:

This module saves a particular Forms Language Source (fdl file) and a Forms Definition Object (fd file) after it translates the internal data structure into forms language syntax.

Input Parameters:
- Name of the form to be saved under flag indicating to write or not to write out the fd file
- Pointer to list of forms to be written out

Output Parameters:
- Returns any error code or a NULL pointer if successful

EDTMOD:

This module is the control module for all edit tasks. It controls the EDTTASK screen.

Input Parameters:
- New/old form flag
- Change/retrieve only flag

Output Parameters:
- Returns any error code or a NULL pointer if successful

LISTFM:

This module lists all forms in the Forms Language Source on which work is being done.

Input Parameters:
- None

Output Parameters:
- Returns any error codes or NULL pointer if successful
3.1.2.8 **INSFRM:**

This module inserts a new form into the Forms Language Source on which work is being done.

**Input Parameters:**
- name of form

**Output Parameters:**
- Returns any error code or NULL pointer if successful

3.1.2.9 **DRPFRM:**

This module deletes a form from the Forms Language Source on which work is being done.

**Input Parameters:**
- Name of form

**Output Parameters:**
- Returns any error codes or NULL pointer if successful

3.1.2.10 **EDTWHL:**

This module allows the user to edit an entire form at once. It controls the presentation of the FORM EDIT screen.

**Input Parameters:**
- Read-only flag
- Name of form

**Output Parameters:**
- Returns any error code or NULL pointer if successful

3.1.2.11 **EDTFLD:**

This module allows the user to edit all fields of a form one at a time. It controls the presentation of the FIELD EDIT screen. It is also called from LAYOUT.

**Input Parameters:**
- Read-only flag
- col cursor position if coming from layout mode
- row cursor position if coming from layout mode
- pointer to internal form structure
- edit mode

**Output Parameters:**
- Returns any error code or NULL pointer if successful
3.1.2.12 **LAYOUT:**

This module allows the user to edit an entire form as it would appear when used (with regards to the location and size of fields) on one screen. The other needed information is filled in on the LAYOUT DESCRIPTION screen.

**Input Parameters:**
- Pointer to internal form structure
- Read-only flag

**Output Parameters:**
- Returns any error code or NULL pointer if successful

3.1.2.13 **SCRMAN:**

This module controls the first stage of layout edit mode - it manages the screen using the following three modules to translate internal structure to screen layout and vice versa.

**Input Parameters:**
- Read-only flag
- Pointer to internal form structure

**Output Parameters:**
- Row position returned from GETCUR
- Col position returned from GETCUR

3.1.2.14 **CHGPOS:**

This module allows the user to change the location of a field in layout mode by indicating the "from" and "to" locations on the screen.

**Input Parameters:**
- Pointer to internal form structure

**Output Parameters:**
- Returns any error code or NULL pointer if successful

3.1.2.15 **TRNSCR:**

This module translates the layout screen format to internal structure.

**Input Parameters:**
- Pointer to internal form structure

**Output Parameters:**
- Returns any error code or NULL pointer if successful
3.1.2.16 **TRNSTR:**

This module translates the internal structure to the layout screen format.

**Input Parameters:**
- Pointer to internal form structure
- Read only flag

**Output Parameters:**
- None

3.1.2.17 **VALINP:**

This module performs validation checks on fields. The objects to be validated are the values input on the FIELD EDIT and FORM EDIT screens.

**Input Parameters:**
- Pointer to form to be validated
- Pointer to field to be validated
- Flag indicating type of validation

**Output Parameters:**
- Returns TRUE if validation okay else returns FALSE

3.1.2.18 **GTMFMD:**

This module retrieves fields from the internal structure.

**Input Parameters:**
- Pointer to 1st field in internal structure
- Name of field to find

**Output Parameters:**
- Pointer to field in the internal structure or NULL if could not find field

3.1.2.19 **MODFLD:**

This module modifies a field in the internal structure.

**Input Parameters:**
- Pointer to parent of field
- Pointer to pointer of field being modified
- Pointer to screen changed information
- Pointer to screen help info
Pointer to screen value info
Pointer to screen item info

Output Parameters:
  Returns any error code or NULL pointer if successful

3.1.2.20  DELFLD:

This module deletes a field from the internal structure.
This is the same function as that used by the Form Processor.

3.1.2.21  INSFLD:

This module inserts a field into the internal structure.

Input Parameters:
  Address of pointer to field being inserted
  Address where next field pointer will be inserted
  Address where previous field pointer will be inserted
  Pointer to parent of field
  Pointer to screen field information
  Pointer to screen help info
  Pointer to screen value info
  Pointer to screen item info
  Recursion level

Output Parameters:
  Returns any error code or NULL pointer if successful

3.1.2.22  COPFRM:

This module copies a Forms Language Source file into an alternate internal data structure and gets the pointer to the specified form.

Input Parameters:
  Name of Forms Language Source file to copy from
  Name of form to copy

Output Parameters:
  Sets global variables:
    Copyfls, name of Forms Language Source just copied
    Copyfrm, name of form just copied
    Altbuf, beginning of list containing all forms of Forms Language Source
    Altfrm, pointer to form user wishes to copy
3.1.2.23 MODFRM:

This module updates information about the form.

Input Parameters:
- Pointer to form internal structure
- Pointer to screen form information

Output Parameters:
- Returns any error code or NULL pointer if successful

3.1.2.24 FLPFMST:

This module translates an FPD field structure for a form into the screen information structure.

Input Parameters:
- Pointer to screen structure
- Pointer to fpd form field

Output Parameters:
- None

3.1.2.25 FLSTRC:

This module translates the FPD field structure to screen information structure for items, windows, and forms.

Input Parameters:
- Pointer to fpd field
- Pointer to screen field structure
- Pointer to screen item help structure
- Pointer to screen item value structure
- Pointer to screen field domain check structure

Output Parameters:
- Fills appropriate screen area with field information

3.1.2.26 FLWHST:

This module fills in the output screen FORM EDIT, and associates each field line on the output screen with the field's internal structure.

Input Parameters:
- Pointer to form internal structure on which editing is to occur.
Output Parameters:
Fills output screen FORM EDIT area with form and field info and creates an external array of pointers

3.1.2.27 GWHINP:
This module gets all input for the FORM EDIT screen for the fields on the form being edited.

Input Parameters:
Pointer to form internal structure
Output Parameters:
PF key provided by OISCR
Returns any error code or NULL pointer if successful

3.1.2.28 GTCPFD:
This module gets the field at the located cursor position.

Input Parameters:
Pointer to form internal structure
Row cursor position
Col cursor position
Output Parameters:
Pointer to field at that location else NULL if no field found

3.1.2.29 DRPWHL:
This module deletes all fields marked by the user on the FORM EDIT screen.

Input Parameters:
Pointer to internal form structure
Output Parameters:
PF key received by OISCR

3.1.2.30 MODWHL:
This module modifies existing fields as input by the user on the FORM EDIT screen.

Input Parameters:
Pointer to parent of field
Pointer to field being modified
Pointer to input screen structure
Pointer to help line on screen
3.1.2.31 INSWHL:

This module inserts all fields that have been entered on the FORM EDIT screen.

Input Parameters:
- Pointer to form internal structure

Output Parameters:
- Returns any error code or NULL pointer if successful

3.2 Functional Flow

Figure 3-2 is a data flow for the Forms Driven Form Editor.
3.3 Interfaces

The FDFE interfaces directly with users as an IISS application. Physical terminals are assumed to have video display, a textual keyboard, four cursor positioning keys or key sequences, a help key or key sequence, an entry key, and four other keys to be used by the FDFE for special processing. The FDFE must interface with the AI, FLAN, and the operating system.

![Figure 3-3 FDFE Interface Diagram](image)

3.3.1 Application Interface

The FDFE interacts with users by calling appropriate routines of the Application Interface (AI). This interface creates messages which are sent to the Form Processor which moves information describing interactive terminal input and output and provides a link to users of the FDFE through the Virtual Terminal.

3.3.2 Forms Language Compiler

The FDFE uses the Forms Language Compiler (FLAN) to convert forms language source into the Form Processor internal forms.
structure. The FDFE also invokes the FLAN when the form under construction is to be stored.

3.3.3 Operating System

The FDFE stores form language source files (fdl files on the VAX) and compiled form definitions (fd files on the VAX). Form language source files may subsequently be compiled and displayed. The storage of the fdl files and fd files is system dependent. The VAX implementation uses the logicals IISSSLIB (for fdl files) to store/retrieve the files in/from the appropriate directory.

3.4 Program Interrupts

This section does not apply to the detailed design of the Forms Driven Form Editor.

3.5 Timing and Sequencing Description

The data flow diagram in section 3.2 and the detail design description in section 3.10 contain the procedural information for sequencing and control logic.

3.6 Special Control Features

The detailed design of the FDFE does not include any special control features as defined in the ICAM Documentation Standards manual.

3.7 Storage Allocation

The Forms Driven Form Editor executable is 351 blocks.

3.7.1 Data Base Definition

3.7.1.1 File Descriptions

1. FILE NAME: name.FDL - Form Definition Language file.

PURPOSE: This file contains the language description of one or more forms. Compiling this file produces Form Definition files.

DECLARATION:

char line [132];
2. FILE NAME: name.TMP - Temporary FDL file.

PURPOSE: This file contains the language description of one or more forms. It is created during an editing session from the FDPE's internal data structure. If this file compiles successfully, it is renamed to an FDL file, otherwise it remains so that the user can correct errors and recompile it.

DECLARATION:

char line [132];

3. FILE NAME: formname.FD - Form Definition file. The name of this file is dependent upon the form it describes.

PURPOSE: This file contains information about the structure and attributes of a form that is used at run time by the Form Processor.

DECLARATION:

typedef struct /* version number record */
{
    char rectyp; /* '1' */
    int vernum; /* current version number (2) */
    char linefeed; /* */
} VERREC;

typedef struct /* form record */
{
    char form_name[10]; /* form name */
    char background[10]; /* background name */
    short row; /* starting row */
    short col; /* starting col */
    short width; /* width */
    short depth; /* depth */
    short n_txtflds; /* number of text fields */
    short n_datflds; /* number of data fields */
    short s_txtbuf; /* size of the text buffer */
    short s_defbuf; /* size of the default buffer */
    char linefeed; /* */
} FRMREC;
typedef struct /* text record */ {
    short row; /* starting row */
    short col; /* starting col */
    short len; /* total length */
    char linefeed;
} TXTREC;

typedef struct /* field record */ {
    char fld_name[10]; /* field name */
    char fld_type; /* field type (F, I, W, A) */
    short row; /* starting row */
    short col; /* starting col */
    short width; /* field width */
    short depth; /* field depth */
    int min_value; /* minimum value (if any) */
    int max_value; /* maximum value (if any) */
    char helpline[80]; /* help text */
    char disp_att[10]; /* display attribute */
    short n_formats; /* number of formats */
    char format[12][10]; /* format strings */
    short n_arydefs; /* number of dimensions */
    struct /* dimension specification */ {
        char dir; /* repeat direction (H, V) */
        short cnt; /* actual repeat count */
        short sp; /* number of spaces between repetitions */
        short dsp_size; /* display repeat count */
    } array_def[3];
    char linefeed;
} FLDREC;

3.8 Object Code Creation

The FDFE routines were compiled using a C compiler developed by Interactive Software under VAX/VMS.
3.9 Adaptation Data

The C source modules for the FDFE can be compiled using any UNIX version 7 compatible C compiler. The files FPDINI.H and GETFLS.C contain file names for the Form Definition and Form Definition Language files which may not port to systems other than VAX/VMS.

3.10 Detailed Design Description

3.10.1 Main Program List

The following is a list of all "Main Programs" which are modules that are not called by any other module being documented here. These modules are either program entry points or, if they are hooked into another set of programs via subroutine calls, they are the points the external programs can call and therefore enter through. To differentiate between the two types of entry points, look at the individual Module Documentation (section 3.10.8) and look at Module Type for each of the Main Program modules listed. Note whether the routine is a Program, Subroutine, or Function. If it is a Program, it is truly a main program entry point. If not, then it is merely called by other programs not being documented here.
<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHKFLD</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td>CHKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td>CSTASH</td>
<td>CHARACTER STASH</td>
</tr>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
</tr>
<tr>
<td>MAKINT</td>
<td>MAKE EXPRESSION INTO AN INTEGER</td>
</tr>
<tr>
<td>MAKSTR</td>
<td>MAKE EXPRESSION INTO A STRING</td>
</tr>
<tr>
<td>MKPOS</td>
<td>MAKE POSITION NODE</td>
</tr>
<tr>
<td>VALINP/CCKVAL</td>
<td>CHECK VALUE</td>
</tr>
</tbody>
</table>
3.10.2 Module List

The following is a list of all the modules being documented here along with their purpose. Each module has a unique name, no matter what language it was written in.
<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDCHK</td>
<td>ADD POSITION TO CHECK LIST</td>
</tr>
<tr>
<td>ADDEXT</td>
<td>ADD EXTENSION TO FILE_NAME</td>
</tr>
<tr>
<td>CHKARY</td>
<td>CHECK ARRAY</td>
</tr>
<tr>
<td>CHKFLD</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td>CHKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td>CHKPRM</td>
<td>CHECK PARAMETER</td>
</tr>
<tr>
<td>CPYFRM</td>
<td>COPY FORM</td>
</tr>
<tr>
<td>CSTASH</td>
<td>CHARACTER STASH</td>
</tr>
<tr>
<td>DRPFRM</td>
<td>DROP FORM</td>
</tr>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>ERROR</td>
<td>ISSUE ERROR MESSAGE</td>
</tr>
<tr>
<td>EXPAND</td>
<td>EXPAND AN ARRAY</td>
</tr>
<tr>
<td>EXPAND/FIXFRM</td>
<td>FIX UP A FORM</td>
</tr>
<tr>
<td>FATAL</td>
<td>ISSUE FATAL ERROR MESSAGE</td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
</tr>
<tr>
<td>FIFDST</td>
<td>FILL IN FIELD STRUCTURE</td>
</tr>
</tbody>
</table>
FORMS DRIVEN FORM EDITOR Module List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLANCI</td>
<td>FLAN CALLABLE INTERFACE</td>
</tr>
<tr>
<td>FLDTYP</td>
<td>FIELD TYPE</td>
</tr>
<tr>
<td>FLFMST</td>
<td>FIELD TO FORM STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td>FLSTRC</td>
<td>FIELD STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td>FLWHST</td>
<td>FILL WHOLE STRUCTURE</td>
</tr>
<tr>
<td>FNDATT</td>
<td>FIND ATTRIBUTE</td>
</tr>
<tr>
<td>FREBUF</td>
<td>FREE BUFFER</td>
</tr>
<tr>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td>GETFLS/TREEEXP</td>
<td>TREE EXPRESSION</td>
</tr>
<tr>
<td>GETLEN</td>
<td>GET LENGTH</td>
</tr>
<tr>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
</tr>
<tr>
<td>GFLDPT</td>
<td>GET FIELD POINTER</td>
</tr>
<tr>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
</tr>
<tr>
<td>GNXTFD</td>
<td>GET NEXT FIELD</td>
</tr>
<tr>
<td>GNXTFD/NXTFLD</td>
<td>NEXT FIELD</td>
</tr>
<tr>
<td>GTCPFD</td>
<td>GET USING CURSOR POSITION FIELD</td>
</tr>
<tr>
<td>GTFDTX</td>
<td>GET FIELD TEXT</td>
</tr>
<tr>
<td>GTFDTX/GTXINF</td>
<td>GET TEXT INFORMATION</td>
</tr>
<tr>
<td>GTNMFD</td>
<td>GET NAMED FIELD</td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td>INSFLD</td>
<td>INSERT FIELD</td>
</tr>
</tbody>
</table>

3-21
## FORMS DRIVEN FORM EDITOR Module List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>INSFRM</td>
<td>INSERT FORM</td>
</tr>
<tr>
<td>INSWHL</td>
<td>INSERT WHOLE</td>
</tr>
<tr>
<td>LAYOUT</td>
<td>LAYOUT MODE</td>
</tr>
<tr>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td>MAKINT</td>
<td>MAKE EXPRESSION INTO AN INTEGER</td>
</tr>
<tr>
<td>MAKSTR</td>
<td>MAKE EXPRESSION INTO A STRING</td>
</tr>
<tr>
<td>MKPOS</td>
<td>MAKE POSITION NODE</td>
</tr>
<tr>
<td>MODFLD</td>
<td>MODIFY FIELD</td>
</tr>
<tr>
<td>MODFRM</td>
<td>MODIFY FORM</td>
</tr>
<tr>
<td>MODFRM/FRETXT</td>
<td>FREE TEXT</td>
</tr>
<tr>
<td>MODWHL</td>
<td>MODIFY WHOLE</td>
</tr>
<tr>
<td>MYALLOC</td>
<td>MY MALLOC</td>
</tr>
<tr>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
</tr>
<tr>
<td>PREC</td>
<td>PRECEDENCE</td>
</tr>
<tr>
<td>PRSCMD</td>
<td>PARSE COMMAND</td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
</tr>
<tr>
<td>SCRMAN/CHGPOS</td>
<td>CHANG POSITION</td>
</tr>
<tr>
<td>SCRMAN/GETROW</td>
<td>GET ROW</td>
</tr>
<tr>
<td>Module Name</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------</td>
</tr>
<tr>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
</tr>
<tr>
<td>TRNSCR/FLCST</td>
<td>FILL LOCATION STRUCTURE</td>
</tr>
<tr>
<td>TRNSCR/FRLCST</td>
<td>FREE LOCATION STRUCTURES</td>
</tr>
<tr>
<td>TRNSCR/GTFMPMT</td>
<td>GET FORM PROMPT INFORMATION</td>
</tr>
<tr>
<td>TRNSCR/GTPINF</td>
<td>GET PROMPT INFORMATION</td>
</tr>
<tr>
<td>TRNSCR/LDPMINF</td>
<td>LOAD PROMPT INFORMATION</td>
</tr>
<tr>
<td>TRNSCR/MTPMPMT</td>
<td>MATCH PROMPT WITH FIELD</td>
</tr>
<tr>
<td>TRNSCR/PARSCRN</td>
<td>PARSE SCREEN DATA</td>
</tr>
<tr>
<td>TRNSCR/SPSYMB</td>
<td>SPECIAL SYMBOL CHECK</td>
</tr>
<tr>
<td>TRNSTR</td>
<td>TRANSLATE STRUCTURE TO SCREEN</td>
</tr>
<tr>
<td>TRNSTR/FLFLD</td>
<td>FILL FIELD</td>
</tr>
<tr>
<td>TRNSTR/FLPRMPT</td>
<td>FILL PROMPT</td>
</tr>
<tr>
<td>TRNSTR/GARINF</td>
<td>GET ARRAY INFORMATION</td>
</tr>
<tr>
<td>VALINP</td>
<td>VALIDATE INPUT</td>
</tr>
<tr>
<td>VALINP/CCKFLD</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td>VALINP/CCKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td>VALINP/CCKHLP</td>
<td>CHECK HELP</td>
</tr>
<tr>
<td>VALINP/CCKITM</td>
<td>CHECK ITEM</td>
</tr>
<tr>
<td>VALINP/CCKNAM</td>
<td>CHECK NAME</td>
</tr>
<tr>
<td>VALINP/CCKPRM</td>
<td>CHECK PROMPT</td>
</tr>
<tr>
<td>VALINP/CCKRSV</td>
<td>CHECK FOR RESERVED WORD</td>
</tr>
</tbody>
</table>
## FORMS DRIVEN FORM EDITOR Module List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALINP/CCKVAL</td>
<td>CHECK VALUE</td>
</tr>
<tr>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
<tr>
<td>WARNING</td>
<td>ISSUE WARNING MESSAGE</td>
</tr>
<tr>
<td>WRTEXP</td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td>WRTFDL</td>
<td>WRITE FDL FILE</td>
</tr>
<tr>
<td>WRTFDL/ARYREF</td>
<td>ARRAY REFERENCE</td>
</tr>
</tbody>
</table>
3.10.3 External Routines List

The following is a list of all routines or functions not documented here that are called by modules that are documented here. The first caller, in alphabetical order, is listed as well. The specification in which any module is documented may be found in the Module Documentation Index (Document Number CM 620100001). See section 3.10.6 for a list of the modules that call each of these external routines.
# FORMS DRIVEN FORM EDITOR External Routines List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>First User</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABORT</td>
<td>VALINP</td>
</tr>
<tr>
<td>ABS</td>
<td>SCRMAN/CHGPOS</td>
</tr>
<tr>
<td>ACCESS</td>
<td>FDFE</td>
</tr>
<tr>
<td>ADDFRM</td>
<td>FDFE</td>
</tr>
<tr>
<td>ATOI</td>
<td>VALINP/CCKITM</td>
</tr>
<tr>
<td>BLEN</td>
<td>WRTFDL</td>
</tr>
<tr>
<td>CLSFRM</td>
<td>VIEW</td>
</tr>
<tr>
<td>COPFLD</td>
<td>EXPAND</td>
</tr>
<tr>
<td>COPY</td>
<td>FDFE</td>
</tr>
<tr>
<td>DELFLD</td>
<td>EXPAND/FIXFRM</td>
</tr>
<tr>
<td>ESCPY</td>
<td>VALINP/CCKITM</td>
</tr>
<tr>
<td>FCLOSE</td>
<td>GETFLS</td>
</tr>
<tr>
<td>FEOF</td>
<td>LISTIT</td>
</tr>
<tr>
<td>FERROR</td>
<td>PRCFIL</td>
</tr>
<tr>
<td>FGETS</td>
<td>PRCFIL</td>
</tr>
<tr>
<td>FOPEN</td>
<td>SAVFLS</td>
</tr>
<tr>
<td>FPRINTF</td>
<td>WRTFDL</td>
</tr>
<tr>
<td>FREE</td>
<td>TRNSCR/FRLCST</td>
</tr>
<tr>
<td>GATDEF</td>
<td>FIFDST</td>
</tr>
<tr>
<td>GDATA</td>
<td>GWHINP</td>
</tr>
<tr>
<td>GETCUR</td>
<td>SCRMAN</td>
</tr>
<tr>
<td>GWINDO</td>
<td>VIEW</td>
</tr>
<tr>
<td>INITIAL</td>
<td>FDFE/MAIN</td>
</tr>
<tr>
<td>INITFP</td>
<td>FDFE/MAIN</td>
</tr>
<tr>
<td>ISALPHA</td>
<td>VALINP/CCKFLD</td>
</tr>
<tr>
<td>ISSPACE</td>
<td>TRNSCR/GTPINF</td>
</tr>
<tr>
<td>MAKFLD</td>
<td>INSFLD</td>
</tr>
<tr>
<td>MALLOC</td>
<td>MYALLOC</td>
</tr>
<tr>
<td>MATOI</td>
<td>VALINP/CCKFLD</td>
</tr>
<tr>
<td>MAX</td>
<td>CHKFLD</td>
</tr>
<tr>
<td>MEMCMP</td>
<td>VALINP</td>
</tr>
<tr>
<td>MEMCPY</td>
<td>GWHINP</td>
</tr>
<tr>
<td>MEMDT</td>
<td>GITMD</td>
</tr>
<tr>
<td>MEMSET</td>
<td>PRSCMD</td>
</tr>
<tr>
<td>MIN</td>
<td>VALINP/CCKFLD</td>
</tr>
<tr>
<td>MITOA</td>
<td>TRNSCR</td>
</tr>
<tr>
<td>MKTEMP</td>
<td>LISTIT</td>
</tr>
<tr>
<td>OISCR</td>
<td>SCRMAN</td>
</tr>
<tr>
<td>PDATA</td>
<td>LISTIT</td>
</tr>
<tr>
<td>PMSGLC</td>
<td>SCRMAN</td>
</tr>
<tr>
<td>PMSGLS</td>
<td>LISTFM</td>
</tr>
</tbody>
</table>
# FORMS DRIVEN FORM EDITOR External Routines List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>First User</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUTATT</td>
<td>EDTFLD</td>
</tr>
<tr>
<td>PUTCUR</td>
<td>PUTERR</td>
</tr>
<tr>
<td>RENAME</td>
<td>SAVFLS</td>
</tr>
<tr>
<td>REWIND</td>
<td>SAVFLS</td>
</tr>
<tr>
<td>RMVPAG</td>
<td>LISTIT</td>
</tr>
<tr>
<td>RSVATT</td>
<td>EXPAND/FIXFRM</td>
</tr>
<tr>
<td>SPRINTF</td>
<td>VALINP/CCKNAM</td>
</tr>
<tr>
<td>STRASN</td>
<td>CHKARY</td>
</tr>
<tr>
<td>STRCAT</td>
<td>SAVFLS</td>
</tr>
<tr>
<td>STRCHR</td>
<td>VALINP/CCKFRM</td>
</tr>
<tr>
<td>STRCMP</td>
<td>VALINP/CCKFLD</td>
</tr>
<tr>
<td>STRCPY</td>
<td>LISTIT</td>
</tr>
<tr>
<td>STRLEN</td>
<td>FDFE</td>
</tr>
<tr>
<td>STRNCMP</td>
<td>PRSCMD</td>
</tr>
<tr>
<td>STRNCPY</td>
<td>PRSCMD</td>
</tr>
<tr>
<td>STRRCHR</td>
<td>SCRMAN/GETROW</td>
</tr>
<tr>
<td>STRSPN</td>
<td>VALINP/CCKHLP</td>
</tr>
<tr>
<td>STRUPC</td>
<td>VALINP/CCKHLP</td>
</tr>
<tr>
<td>SYSMSG</td>
<td>VIEW</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>LISTIT</td>
</tr>
<tr>
<td>TERMFP</td>
<td>FDFE/MAIN</td>
</tr>
<tr>
<td>TRMNAT</td>
<td>FDFE/MAIN</td>
</tr>
<tr>
<td>UNLINK</td>
<td>FDFE</td>
</tr>
<tr>
<td>WRTFRM</td>
<td>SAVFLS</td>
</tr>
<tr>
<td>YYPARSE</td>
<td>FLANCI</td>
</tr>
</tbody>
</table>

---

3-27
3.10.4 Include File List

The following is a list of all include files called in by modules being documented here. Each include file has a unique name regardless of the language being used. The purpose of each include file is listed as well. A more complete description of each include file is given in section 3.10.9. The purpose listed is the one that is in the source code of the include file.

A purpose of "**** PURPOSE NOT FOUND BY STRIPPER ****" indicates that a purpose statement was not written into the include file itself. The most common reason for this is that the include file comes from system libraries that were not developed by the project, such as 'C' libraries that are provided with the 'C' compiler.

See section 3.10.6 for a set of lists which show all the modules which call in each of these include files.
# FORMS DRIVEN FORM EDITOR Include File List

<table>
<thead>
<tr>
<th>File Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTYPE</td>
<td>**** PURPOSE NOT FOUND BY STRIPPER ****</td>
</tr>
<tr>
<td>FDFE</td>
<td>FDFE DATA STRUCTURES</td>
</tr>
<tr>
<td>FDFEFM</td>
<td>FDFE FORM DEFINITIONS</td>
</tr>
<tr>
<td>FDFEINI</td>
<td>FDFE INITIALIZATIONS</td>
</tr>
<tr>
<td>FPCODE</td>
<td>FORM PROCESSOR RETURN CODES</td>
</tr>
<tr>
<td>FPD</td>
<td>FORM PROCESSOR DATA</td>
</tr>
<tr>
<td>FPDINI</td>
<td>FPD INITIALIZATION</td>
</tr>
<tr>
<td>FPPARM</td>
<td>FORM PROCESSOR PARAMETERS</td>
</tr>
<tr>
<td>NTM</td>
<td>NTM INTERFACE INCLUDE FILE</td>
</tr>
<tr>
<td>RW</td>
<td>REPORT WRITER DEFINITIONS</td>
</tr>
<tr>
<td>STDIO</td>
<td>**** PURPOSE NOT FOUND BY STRIPPER ****</td>
</tr>
<tr>
<td>STDTyp</td>
<td>STANDARD TYPE DEFINITIONS</td>
</tr>
</tbody>
</table>
3.10.5 Where Include File Used List

The following lists each include file from 3.10.4 and all the modules documented in this specification which include them. The purpose of each module is listed as well.
# FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include</th>
<th>Module</th>
<th>Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Name</td>
<td>Purpose</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## CType
- **EDTFLD**: Edit Field
- **GITMD**: Get Item Data and Insert in Structure
- **PUTERR**: Put Error
- **TRNSCR**: Translate Screen to Structure
- **TRNSCR/FL**: Fill Location Structure
- **TRNSCR/FR**: Free Location Structures
- **TRNSCR/GT**: Get Form Prompt Information
- **TRNSCR/LD**: Load Prompt Information
- **TRNSCR/MT**: Match Prompt with Field
- **TRNSCR/PA**: Parse Screen Data
- **TRNSCR/SP**: Special Symbol Check
- **VALINP**: Validate Input
- **VALINP/CC**: Check Field
- **VALINP/CC**: Check Form
- **VALINP/CC**: Check Help
- **VALINP/CC**: Check Item
- **VALINP/CC**: Check Name
- **VALINP/CC**: Check Prompt
- **VALINP/CC**: Check for Reserved Word
- **VALINP/CC**: Check Value

## FDFE
- **ADDEXT**: Add Extension to File_Name
- **CHKPRM**: Check Parameter
- **CPYFRM**: Copy Form
- **DRPFRM**: Drop Form
- **DRPWHL**: Drop Whole
- **EDTFLD**: Edit Field
- **EDTMOD**: Edit Mode
- **EDTWHL**: Edit Whole
- **FDFE**: Forms Driven Form Editor
- **FDFE/MAIN**: Main Module for Forms Driven Forms Editor (FDFE)
- **FLFMST**: Field to Form Structure Translation
- **FLSTRC**: Field Structure Translation
FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLWHST</td>
<td>FILL WHOLE STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>FREBUF</td>
<td>FREE BUFFER</td>
<td></td>
</tr>
<tr>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
<td></td>
</tr>
<tr>
<td>GETFLS/TR</td>
<td>TREE EXPRESSION</td>
<td></td>
</tr>
<tr>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
<td></td>
</tr>
<tr>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>GNXTFD</td>
<td>GET NEXT FIELD</td>
<td></td>
</tr>
<tr>
<td>GNXTFD/NX</td>
<td>NEXT FIELD</td>
<td></td>
</tr>
<tr>
<td>GTFDTX</td>
<td>GET FIELD TEXT</td>
<td></td>
</tr>
<tr>
<td>GTFDTX/GT</td>
<td>GET TEXT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>GTHMFDF</td>
<td>GET NAMED FIELD</td>
<td></td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
<td></td>
</tr>
<tr>
<td>INSFLD</td>
<td>INSERT FIELD</td>
<td></td>
</tr>
<tr>
<td>INSFRM</td>
<td>INSERT FORM</td>
<td></td>
</tr>
<tr>
<td>INSWHLL</td>
<td>INSERT WHOLE</td>
<td></td>
</tr>
<tr>
<td>LAYOUT</td>
<td>LAYOUT MODE</td>
<td></td>
</tr>
<tr>
<td>LISTFM</td>
<td>LIST FORMS</td>
<td></td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
<td></td>
</tr>
<tr>
<td>MODFLD</td>
<td>MODIFY FIELD</td>
<td></td>
</tr>
<tr>
<td>MODFRM</td>
<td>MODIFY FORM</td>
<td></td>
</tr>
<tr>
<td>MODFRM/FR</td>
<td>FREE TEXT</td>
<td></td>
</tr>
<tr>
<td>MODWHL</td>
<td>MODIFY WHOLE</td>
<td></td>
</tr>
<tr>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
<td></td>
</tr>
<tr>
<td>PRSCMD</td>
<td>PARSE COMMAND</td>
<td></td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/CH</td>
<td>SCREEN MANAGER</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/GE</td>
<td>GET ROW</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/FL</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/FR</td>
<td>FILL LOCATION STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/GT</td>
<td>FILL LOCATION STRUCTURES</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/GT</td>
<td>GET FORM PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/GT</td>
<td>GET PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/LD</td>
<td>LOAD PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/MT</td>
<td>MATCH PROMPT WITH FIELD</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/PA</td>
<td>PARSE SCREEN DATA</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/SP</td>
<td>SPECIAL SYMBOL CHECK</td>
<td></td>
</tr>
<tr>
<td>TRNSTR</td>
<td>TRANSLATE STRUCTURE TO SCREEN</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/FL</td>
<td>FILL FIELD</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/FL</td>
<td>FILL PROMPT</td>
<td></td>
</tr>
</tbody>
</table>

3-32
### FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TRNSTR/GA</td>
<td>GET ARRAY INFORMATION</td>
</tr>
<tr>
<td></td>
<td>VALINP</td>
<td>VALIDATE INPUT</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK HELP</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK ITEM</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK NAME</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK PROMPT</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK FOR RESERVED WORD</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK VALUE</td>
</tr>
<tr>
<td></td>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
<tr>
<td></td>
<td>ADDEXT</td>
<td>ADD EXTENSION TO FILE_NAME</td>
</tr>
<tr>
<td></td>
<td>CHKPRM</td>
<td>CHECK PARAMETER</td>
</tr>
<tr>
<td></td>
<td>CPYFRM</td>
<td>COPY FORM</td>
</tr>
<tr>
<td></td>
<td>DRPFRM</td>
<td>DROP FORM</td>
</tr>
<tr>
<td></td>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td></td>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td></td>
<td>EDTMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td></td>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td></td>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td></td>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
</tr>
<tr>
<td></td>
<td>FLMFST</td>
<td>FIELD TO FORM STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td></td>
<td>FLSSTRC</td>
<td>FIELD STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td></td>
<td>FLWHST</td>
<td>FILL WHOLE STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>FREBUF</td>
<td>FREE BUFFER</td>
</tr>
<tr>
<td></td>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td></td>
<td>GETFLS/TR</td>
<td>TREE EXPRESSION</td>
</tr>
<tr>
<td></td>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
</tr>
<tr>
<td></td>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>GNXTFD</td>
<td>GET NEXT FIELD</td>
</tr>
<tr>
<td></td>
<td>GNXTFD/NX</td>
<td>NEXT FIELD</td>
</tr>
<tr>
<td></td>
<td>GTFDTX</td>
<td>GET FIELD TEXT</td>
</tr>
<tr>
<td></td>
<td>GTFDTX/GT</td>
<td>GET TEXT INFORMATION</td>
</tr>
<tr>
<td></td>
<td>GTHMFDF</td>
<td>GET NAMED FIELD</td>
</tr>
</tbody>
</table>
## FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td></td>
<td>INSFLD</td>
<td>INSERT FIELD</td>
</tr>
<tr>
<td></td>
<td>INSF RM</td>
<td>INSERT FORM</td>
</tr>
<tr>
<td></td>
<td>INSWHL</td>
<td>INSERT WHOLE</td>
</tr>
<tr>
<td></td>
<td>LAYOUT</td>
<td>LAYOUT MODE</td>
</tr>
<tr>
<td></td>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td></td>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td></td>
<td>MODF LD</td>
<td>MODIFY FIELD</td>
</tr>
<tr>
<td></td>
<td>MODF RM</td>
<td>MODIFY FORM</td>
</tr>
<tr>
<td></td>
<td>MODF RM/FR</td>
<td>FREE TEXT</td>
</tr>
<tr>
<td></td>
<td>MODW HL</td>
<td>MODIFY WHOLE</td>
</tr>
<tr>
<td></td>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
</tr>
<tr>
<td></td>
<td>PRSC MD</td>
<td>PARSE COMMAND</td>
</tr>
<tr>
<td></td>
<td>PUTERR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td></td>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
</tr>
<tr>
<td></td>
<td>SCRMAN/CH</td>
<td>CHANG POSITION</td>
</tr>
<tr>
<td></td>
<td>SCRMAN/GE</td>
<td>GET ROW</td>
</tr>
<tr>
<td></td>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>TRN SCR/FL</td>
<td>FILL LOCATION STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>TRN SCR/FR</td>
<td>FREE LOCATION STRUCTURES</td>
</tr>
<tr>
<td></td>
<td>TRN SCR/GT</td>
<td>GET FORM PROMPT INFORMATION</td>
</tr>
<tr>
<td></td>
<td>TRN SCR/GT</td>
<td>GET PROMPT INFORMATION</td>
</tr>
<tr>
<td></td>
<td>TRN SCR/LD</td>
<td>LOAD PROMPT INFORMATION</td>
</tr>
<tr>
<td></td>
<td>TRN SCR/MT</td>
<td>MATCH PROMPT WITH FIELD</td>
</tr>
<tr>
<td></td>
<td>TRN SCR/PA</td>
<td>PARSE SCREEN DATA</td>
</tr>
<tr>
<td></td>
<td>TRN SCR/SP</td>
<td>SPECIAL SYMBOL CHECK</td>
</tr>
<tr>
<td></td>
<td>TRN ST R</td>
<td>TRANSLATE STRUCTURE TO SCREEN</td>
</tr>
<tr>
<td></td>
<td>TRN STR/FL</td>
<td>FILL FIELD</td>
</tr>
<tr>
<td></td>
<td>TRN STR/FL</td>
<td>FILL PROMPT</td>
</tr>
<tr>
<td></td>
<td>TRN STR/GA</td>
<td>GET ARRAY INFORMATION</td>
</tr>
<tr>
<td></td>
<td>VAL INP</td>
<td>VALIDATE INPUT</td>
</tr>
<tr>
<td></td>
<td>VAL INP/CC</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td></td>
<td>VAL INP/CC</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td></td>
<td>VAL INP/CC</td>
<td>CHECK HELP</td>
</tr>
<tr>
<td></td>
<td>VAL INP/CC</td>
<td>CHECK ITEM</td>
</tr>
<tr>
<td></td>
<td>VAL INP/CC</td>
<td>CHECK NAME</td>
</tr>
<tr>
<td></td>
<td>VAL INP/CC</td>
<td>CHECK PROMPT</td>
</tr>
<tr>
<td></td>
<td>VAL INP/CC</td>
<td>CHECK FOR RESERVED WORD</td>
</tr>
<tr>
<td></td>
<td>VAL INP/CC</td>
<td>CHECK VALUE</td>
</tr>
<tr>
<td></td>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
</tbody>
</table>
FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FDFEINI

<table>
<thead>
<tr>
<th>Module</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
</tr>
</tbody>
</table>

FPCODE

<table>
<thead>
<tr>
<th>Module</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDCHK</td>
<td>ADD POSITION TO CHECK LIST</td>
</tr>
<tr>
<td>CHKARY</td>
<td>CHECK ARRAY</td>
</tr>
<tr>
<td>CHKFLD</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td>CHKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td>CPYFRM</td>
<td>COPY FORM</td>
</tr>
<tr>
<td>CSTASH</td>
<td>CHARACTER STASH</td>
</tr>
<tr>
<td>DRPFRM</td>
<td>DROP FORM</td>
</tr>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>EXPAND</td>
<td>EXPAND AN ARRAY</td>
</tr>
<tr>
<td>EXPAND/FI</td>
<td>FIX UP A FORM</td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
</tr>
<tr>
<td>FLANCI</td>
<td>FLAN CALLABLE INTERFACE</td>
</tr>
<tr>
<td>FLDTYP</td>
<td>FIELD TYPE</td>
</tr>
<tr>
<td>FNDATT</td>
<td>FIND ATTRIBUTE</td>
</tr>
<tr>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td>GETFLS/TR</td>
<td>TREE EXPRESSION</td>
</tr>
<tr>
<td>GF DINP</td>
<td>GET FIELD INPUT</td>
</tr>
<tr>
<td>GFLDPT</td>
<td>GET FIELD POINTER</td>
</tr>
<tr>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
</tr>
<tr>
<td>GNXTFD</td>
<td>GET NEXT FIELD</td>
</tr>
<tr>
<td>GNXTFD/NX</td>
<td>NEXT FIELD</td>
</tr>
<tr>
<td>GFDFTX</td>
<td>GET FIELD TEXT</td>
</tr>
</tbody>
</table>
### FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>GTFDTX/GT</td>
<td>GET TEXT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
<td></td>
</tr>
<tr>
<td>INSFLD</td>
<td>INSERT FIELD</td>
<td></td>
</tr>
<tr>
<td>INSRFM</td>
<td>INSERT FORM</td>
<td></td>
</tr>
<tr>
<td>INSWHL</td>
<td>INSERT WHOLE</td>
<td></td>
</tr>
<tr>
<td>LAYOUT</td>
<td>LAYOUT MODE</td>
<td></td>
</tr>
<tr>
<td>LISTFM</td>
<td>LIST FORMS</td>
<td></td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
<td></td>
</tr>
<tr>
<td>MAKINT</td>
<td>MAKE EXPRESSION INTO AN INTEGER</td>
<td></td>
</tr>
<tr>
<td>MAKSTR</td>
<td>MAKE EXPRESSION INTO A STRING</td>
<td></td>
</tr>
<tr>
<td>MKPOS</td>
<td>MAKE POSITION NODE</td>
<td></td>
</tr>
<tr>
<td>MODFLD</td>
<td>MODIFY FIELD</td>
<td></td>
</tr>
<tr>
<td>MODFRM</td>
<td>MODIFY FORM</td>
<td></td>
</tr>
<tr>
<td>MODFRM/FR</td>
<td>FREE TEXT</td>
<td></td>
</tr>
<tr>
<td>MODWHL</td>
<td>MODIFY WHOLE</td>
<td></td>
</tr>
<tr>
<td>MYALLOC</td>
<td>MY MALLOC</td>
<td></td>
</tr>
<tr>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
<td></td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
<td></td>
</tr>
<tr>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
<td></td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/CH</td>
<td>CHANGE POSITION</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/GE</td>
<td>GET ROW</td>
<td></td>
</tr>
<tr>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/FL</td>
<td>FILL LOCATION STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/FR</td>
<td>FREE LOCATION STRUCTURES</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/GT</td>
<td>GET FORM PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/MT</td>
<td>MATCH PROMPT WITH FIELD</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/PA</td>
<td>PARSE SCREEN DATA</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/SP</td>
<td>SPECIAL SYMBOL CHECK</td>
<td></td>
</tr>
<tr>
<td>TRNSTR</td>
<td>TRANSLATE STRUCTURE TO SCREEN</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/FL</td>
<td>FILL FIELD</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/FL</td>
<td>FILL PROMPT</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/GA</td>
<td>GET ARRAY INFORMATION</td>
<td></td>
</tr>
<tr>
<td>VALINP</td>
<td>VALIDATE INPUT</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK FIELD</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK FORM</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK HELP</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK ITEM</td>
<td></td>
</tr>
</tbody>
</table>
FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include</th>
<th>Module</th>
<th>Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Name</td>
<td>Purpose</td>
</tr>
<tr>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK NAME</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK PROMPT</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK FOR RESERVED WORD</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK VALUE</td>
</tr>
<tr>
<td></td>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
<tr>
<td></td>
<td>WRTEXP</td>
<td>WRITE EXPRESSION</td>
</tr>
</tbody>
</table>

FPD

<table>
<thead>
<tr>
<th>Module</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDCHK</td>
<td>ADD POSITION TO CHECK LIST</td>
</tr>
<tr>
<td>ADDEXT</td>
<td>ADD EXTENSION TO FILE_NAME</td>
</tr>
<tr>
<td>CHKARY</td>
<td>CHECK ARRAY</td>
</tr>
<tr>
<td>CHKFLD</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td>CHKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td>CHKPRM</td>
<td>CHECK PARAMETER</td>
</tr>
<tr>
<td>CPYFRM</td>
<td>COPY FORM</td>
</tr>
<tr>
<td>CSTASH</td>
<td>CHARACTER STASH</td>
</tr>
<tr>
<td>DRPFRM</td>
<td>DROP FORM</td>
</tr>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>EXPAND</td>
<td>EXPAND AN ARRAY</td>
</tr>
<tr>
<td>EXPAND/FI</td>
<td>FIX UP A FORM</td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
</tr>
<tr>
<td>FIFDST</td>
<td>FILL IN FIELD STRUCTURE</td>
</tr>
<tr>
<td>FLANCI</td>
<td>FLAN CALLABLE INTERFACE</td>
</tr>
<tr>
<td>FLDTYPEP</td>
<td>FIELD TYPE</td>
</tr>
<tr>
<td>FLFMST</td>
<td>FIELD TO FORM STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td>FLSTRC</td>
<td>FIELD STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td>FLWHST</td>
<td>FILL WHOLE STRUCTURE</td>
</tr>
<tr>
<td>FNDATT</td>
<td>FIND ATTRIBUTE</td>
</tr>
<tr>
<td>FREBUF</td>
<td>FREE BUFFER</td>
</tr>
<tr>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td>GETFLS/TR</td>
<td>TREE EXPRESSION</td>
</tr>
<tr>
<td>GFDFINP</td>
<td>GET FIELD INPUT</td>
</tr>
</tbody>
</table>

3-37
FORMS DRIVEN FORM EDITOR Where-include-file-used List

Include File Module Name Module Purpose

GFLDPT GET FIELD POINTER
GITMD GET ITEM DATA AND INSERT IN STRUCTURE
GNXTFD GET NEXT FIELD
GNXTFD/NX NEXT FIELD
GTCPFD GET USING CURSOR POSITION FIELD
GTFDXT GET FIELD TEXT
GTFDXT/GT GET TEXT INFORMATION
GTNMFD GET NAMED FIELD
GWHIMP GET WHOLE INPUT
INSFLD INSERT FIELD
INSFRM INSERT FORM
INSWHL INSERT WHOLE
LAYOUT LAYOUT MODE
LISTFM LIST FORMS
LISTIT LIST IT
MAKINT MAKE EXPRESSION INTO AN INTEGER
MAKSTR MAKE EXPRESSION INTO A STRING
MKPOS MAKE POSITION NODE
MODFLD MODIFY FIELD
MODFRM MODIFY FORM
MODFRM/FR FREE TEXT
MODWHL MODIFY WHOLE
MYALLOC MY MALLOC
PRCFIL PROCESS TEMPORARY FILE
PREC PRECEDENCE
PRSCMD PARSE COMMAND
PUTERR PUT ERROR
SAVFLS SAVE FDL SOURCE
SCRMAN SCREEN MANAGER
SCRMAN/CH CHANG POSITION
SCRMAN/GE GET ROW
TRNSCR TRANSLATE SCREEN TO STRUCTURE
TRNSCR/FL FILL LOCATION STRUCTURE
TRNSCR/FR FREE LOCATION STRUCTURES
TRNSCR/GT GET FORM PROMPT INFORMATION
TRNSCR/GT GET PROMPT INFORMATION
TRNSCR/LD LOAD PROMPT INFORMATION
TRNSCR/MT MATCH PROMPT WITH FIELD
TRNSCR/PA PARSE SCREEN DATA
TRNSCR/SP SPECIAL SYMBOL CHECK

3-38
FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include</th>
<th>Module</th>
<th>Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Name</td>
<td>Purpose</td>
</tr>
<tr>
<td>--------</td>
<td>----------</td>
<td>-----------------</td>
</tr>
</tbody>
</table>

- **TRNSTR** TRANSLATE STRUCTURE TO SCREEN
- **TRNSTR/FL** FILL FIELD
- **TRNSTR/FL** FILL PROMPT
- **TRNSTR/GA** GET ARRAY INFORMATION
- **VALINP** VALIDATE INPUT
- **VALINP/CC** CHECK FIELD
- **VALINP/CC** CHECK FORM
- **VALINP/CC** CHECK HELP
- **VALINP/CC** CHECK ITEM
- **VALINP/CC** CHECK NAME
- **VALINP/CC** CHECK PROMPT
- **VALINP/CC** CHECK FOR RESERVED WORD
- **VALINP/CC** CHECK VALUE
- **VIEW** VIEW A FORM
- **WRTEXP** WRITE EXPRESSION
- **WRTFDL** WRITE FDL FILE
- **WRTFDL/AR** ARRAY REFERENCE

---

**FPDINI**

- **FDFE** FORMS DRIVEN FORM EDITOR
- **FDFE/MAIN** MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)

---

**FPPARM**

- **DRPWHL** DROP WHOLE
- **EDTFLD** EDIT FIELD
- **EDTMOD** EDIT MODE
- **EDTWHL** EDIT WHOLE
- **FDFE** FORMS DRIVEN FORM EDITOR
- **FDFE/MAIN** MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
- **GFDINP** GET FIELD INPUT
- **GITMD** GET ITEM DATA AND INSERT IN STRUCTURE
- **GWHINP** GET WHOLE INPUT
### FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>INSFLD</td>
<td>INSERT FIELD</td>
</tr>
<tr>
<td></td>
<td>INSFRM</td>
<td>INSERT FORM</td>
</tr>
<tr>
<td></td>
<td>LAYOUT</td>
<td>LAYOUT MODE</td>
</tr>
<tr>
<td></td>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td></td>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td></td>
<td>PUTERR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td></td>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
</tr>
<tr>
<td></td>
<td>SCRMAN/CH</td>
<td>CHANG POSITION</td>
</tr>
<tr>
<td></td>
<td>SCRMAN/GE</td>
<td>GET ROW</td>
</tr>
<tr>
<td></td>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/FL</td>
<td>FILL LOCATION STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/FR</td>
<td>FREE LOCATION STRUCTURES</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/GT</td>
<td>GET FORM PROMPT INFORMATION</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/GT</td>
<td>GET PROMPT INFORMATION</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/LD</td>
<td>LOAD PROMPT INFORMATION</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/MT</td>
<td>MATCH PROMPT WITH FIELD</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/PA</td>
<td>PARSE SCREEN DATA</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/SP</td>
<td>SPECIAL SYMBOL CHECK</td>
</tr>
<tr>
<td></td>
<td>VALINP</td>
<td>VALIDATE INPUT</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK HELP</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK ITEM</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK NAME</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK PROMPT</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK FOR RESERVED WORD</td>
</tr>
<tr>
<td></td>
<td>VALINP/CC</td>
<td>CHECK VALUE</td>
</tr>
<tr>
<td></td>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
</tbody>
</table>

**NTM**

- **FDFE**: FORMS DRIVEN FORM EDITOR
- **FDFE/MAIN**: MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
### FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include</th>
<th>Module</th>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>RW</td>
<td>ADDCHK</td>
<td>ADD</td>
<td>POSITION TO CHECK LIST</td>
</tr>
<tr>
<td></td>
<td>CHKARY</td>
<td>CHECK</td>
<td>ARRAY</td>
</tr>
<tr>
<td></td>
<td>CHKFLD</td>
<td>CHECK</td>
<td>FIELD</td>
</tr>
<tr>
<td></td>
<td>CHKFRM</td>
<td>CHECK</td>
<td>FORM</td>
</tr>
<tr>
<td></td>
<td>CSTASH</td>
<td>CHARACTER</td>
<td>STASH</td>
</tr>
<tr>
<td></td>
<td>FLANCI</td>
<td>FLAN</td>
<td>CALLABLE INTERFACE</td>
</tr>
<tr>
<td></td>
<td>FLDTYP</td>
<td>FIELD</td>
<td>TYPE</td>
</tr>
<tr>
<td></td>
<td>FNDATT</td>
<td>FIND</td>
<td>ATTRIBUTE</td>
</tr>
<tr>
<td></td>
<td>GFLDPT</td>
<td>GET</td>
<td>FIELD POINTER</td>
</tr>
<tr>
<td></td>
<td>MAKINT</td>
<td>MAKE</td>
<td>EXPRESSION INTO AN INTEGER</td>
</tr>
<tr>
<td></td>
<td>MAKSTR</td>
<td>MAKE</td>
<td>EXPRESSION INTO A STRING</td>
</tr>
<tr>
<td></td>
<td>MKPOS</td>
<td>MAKE</td>
<td>POSITION NODE</td>
</tr>
<tr>
<td></td>
<td>MYALLOC</td>
<td>MY</td>
<td>MALLOC</td>
</tr>
<tr>
<td></td>
<td>WRTEXP</td>
<td>WRITE</td>
<td>EXPRESSION</td>
</tr>
</tbody>
</table>

### STDIO

<table>
<thead>
<tr>
<th>ADDCHK</th>
<th>ADD POSITION TO CHECK LIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHKARY</td>
<td>CHECK ARRAY</td>
</tr>
<tr>
<td>CHKFLD</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td>CHKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td>CSTASH</td>
<td>CHARACTER STASH</td>
</tr>
<tr>
<td>FLANCI</td>
<td>FLAN CALLABLE INTERFACE</td>
</tr>
<tr>
<td>FLDTYP</td>
<td>FIELD TYPE</td>
</tr>
<tr>
<td>FNDATT</td>
<td>FIND ATTRIBUTE</td>
</tr>
<tr>
<td>GFLDPT</td>
<td>GET FIELD POINTER</td>
</tr>
<tr>
<td>MAKINT</td>
<td>MAKE EXPRESSION INTO AN INTEGER</td>
</tr>
<tr>
<td>MAKSTR</td>
<td>MAKE EXPRESSION INTO A STRING</td>
</tr>
<tr>
<td>MKPOS</td>
<td>MAKE POSITION NODE</td>
</tr>
<tr>
<td>MYALLOC</td>
<td>MY MALLOC</td>
</tr>
<tr>
<td>WRTEXP</td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
</tr>
<tr>
<td>PREC</td>
<td>PRECEDENCE</td>
</tr>
</tbody>
</table>

3-41
FORMS DRIVEN FORM EDITOR Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAVFLS</td>
<td></td>
<td>SAVE FDL SOURCE</td>
</tr>
<tr>
<td>WRTEXP</td>
<td></td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td>WRTFDL</td>
<td></td>
<td>WRITE FDL FILE</td>
</tr>
<tr>
<td>WRTFDL/AR</td>
<td></td>
<td>ARRAY REFERENCE</td>
</tr>
</tbody>
</table>

STDTYP

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDCHK</td>
<td>ADD POSITION TO CHECK LIST</td>
</tr>
<tr>
<td>ADEXTEXT</td>
<td>ADD EXTENSION TO FILE_NAME</td>
</tr>
<tr>
<td>CHKARY</td>
<td>CHECK ARRAY</td>
</tr>
<tr>
<td>CHKFLD</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td>CHKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td>CHKPRM</td>
<td>CHECK PARAMETER</td>
</tr>
<tr>
<td>CPYFRM</td>
<td>COPY FORM</td>
</tr>
<tr>
<td>CSTASH</td>
<td>CHARACTER STASH</td>
</tr>
<tr>
<td>DRPFRM</td>
<td>DROP FORM</td>
</tr>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>ERROR</td>
<td>ISSUE ERROR MESSAGE</td>
</tr>
<tr>
<td>EXPAND</td>
<td>EXPAND AN ARRAY</td>
</tr>
<tr>
<td>EXPAND/FI</td>
<td>FIX UP A FORM</td>
</tr>
<tr>
<td>FATAL</td>
<td>ISSUE FATAL ERROR MESSAGE</td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
</tr>
<tr>
<td>FIFDST</td>
<td>FILL IN FIELD STRUCTURE</td>
</tr>
<tr>
<td>FLANCI</td>
<td>FLAN CALLABLE INTERFACE</td>
</tr>
<tr>
<td>FLDTYP</td>
<td>FIELD TYPE</td>
</tr>
<tr>
<td>FLFMST</td>
<td>FIELD TO FORM STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td>FLSTRC</td>
<td>FIELD STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td>FLWHST</td>
<td>FILL WHOLE STRUCTURE</td>
</tr>
<tr>
<td>FNDATT</td>
<td>FIND ATTRIBUTE</td>
</tr>
<tr>
<td>FREBUF</td>
<td>FREE BUFFER</td>
</tr>
<tr>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td>GETFLS/TR</td>
<td>TREE EXPRESSION</td>
</tr>
<tr>
<td>GETLEN</td>
<td>GET LENGTH</td>
</tr>
</tbody>
</table>
FORMS DRIVEN FORM EDITOR Where-included-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
<td></td>
</tr>
<tr>
<td>GFLDPT</td>
<td>GET FIELD POINTER</td>
<td></td>
</tr>
<tr>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>GNXTFD</td>
<td>GET NEXT FIELD</td>
<td></td>
</tr>
<tr>
<td>GNXTFD/NX</td>
<td>NEXT FIELD</td>
<td></td>
</tr>
<tr>
<td>GTCPFD</td>
<td>GET USING CURSOR POSITION FIELD</td>
<td></td>
</tr>
<tr>
<td>GTFDTX</td>
<td>GET FIELD TEXT</td>
<td></td>
</tr>
<tr>
<td>GTFDTX/GT</td>
<td>GET TEXT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>GTNMF</td>
<td>GET NAMED FIELD</td>
<td></td>
</tr>
<tr>
<td>GWWHNP</td>
<td>GET WHOLE INPUT</td>
<td></td>
</tr>
<tr>
<td>INSFLD</td>
<td>INSERT FIELD</td>
<td></td>
</tr>
<tr>
<td>INSFRM</td>
<td>INSERT FORM</td>
<td></td>
</tr>
<tr>
<td>INSWHL</td>
<td>INSERT WHOLE</td>
<td></td>
</tr>
<tr>
<td>LAYOUT</td>
<td>LAYOUT MODE</td>
<td></td>
</tr>
<tr>
<td>LISTFM</td>
<td>LIST FORMS</td>
<td></td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
<td></td>
</tr>
<tr>
<td>MAKINT</td>
<td>MAKE EXPRESSION INTO AN INTEGER</td>
<td></td>
</tr>
<tr>
<td>MAKSTR</td>
<td>MAKE EXPRESSION INTO A STRING</td>
<td></td>
</tr>
<tr>
<td>MKPOS</td>
<td>MAKE POSITION MODE</td>
<td></td>
</tr>
<tr>
<td>MODFLD</td>
<td>MODIFY FIELD</td>
<td></td>
</tr>
<tr>
<td>MODFRM</td>
<td>MODIFY FORM</td>
<td></td>
</tr>
<tr>
<td>MODFRM/FR</td>
<td>FREE TEXT</td>
<td></td>
</tr>
<tr>
<td>MODWHL</td>
<td>MODIFY WHOLE</td>
<td></td>
</tr>
<tr>
<td>MYALLOC</td>
<td>MY MALLOC</td>
<td></td>
</tr>
<tr>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
<td></td>
</tr>
<tr>
<td>PREC</td>
<td>PRECEDENCE</td>
<td></td>
</tr>
<tr>
<td>PRSCHD</td>
<td>PARSE COMMAND</td>
<td></td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
<td></td>
</tr>
<tr>
<td>SAVFILS</td>
<td>SAVE FDL SOURCE</td>
<td></td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/CH</td>
<td>CHANGE POSITION</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/GE</td>
<td>GET ROW</td>
<td></td>
</tr>
<tr>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/FL</td>
<td>FILL LOCATION STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/FR</td>
<td>FREE LOCATION STRUCTURES</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/GT</td>
<td>GET FORM PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/PA</td>
<td>PARSE SCREEN DATA</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/MT</td>
<td>MATCH PROMPT WITH FIELD</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/LD</td>
<td>LOAD PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>Include File</td>
<td>Module Name</td>
<td>Module Purpose</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>TRNSCR/SP</td>
<td>SPECIAL SYMBOL CHECK</td>
<td></td>
</tr>
<tr>
<td>TRNSTR</td>
<td>TRANSLATE STRUCTURE TO SCREEN</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/FL</td>
<td>FILL FIELD</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/FL</td>
<td>FILL PROMPT</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/GA</td>
<td>GET ARRAY INFORMATION</td>
<td></td>
</tr>
<tr>
<td>VALINP</td>
<td>VALIDATE INPUT</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK FIELD</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK FORM</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK HELP</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK ITEM</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK NAME</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK PROMPT</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK FOR RESERVED WORD</td>
<td></td>
</tr>
<tr>
<td>VALINP/CC</td>
<td>CHECK VALUE</td>
<td></td>
</tr>
<tr>
<td>VIEW</td>
<td>VIEW A FORM</td>
<td></td>
</tr>
<tr>
<td>WARNING</td>
<td>ISSUE WARNING MESSAGE</td>
<td></td>
</tr>
<tr>
<td>WRTEXP</td>
<td>WRITE EXPRESSION</td>
<td></td>
</tr>
<tr>
<td>WRTFDL</td>
<td>WRITE FDL FILE</td>
<td></td>
</tr>
<tr>
<td>WRTFDL/AR</td>
<td>ARRAY REFERENCE</td>
<td></td>
</tr>
</tbody>
</table>
3.10.6 Where External Routine Used List

The following lists each external function or routine listed in 3.10.3 and all the documented modules which call it. The purpose of each module is listed as well.
### FORMS DRIVEN FORM EDITOR

**Where-external-routine-used List**

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABORT</td>
<td>VALINP</td>
<td>VALIDATE INPUT</td>
</tr>
<tr>
<td>ABS</td>
<td>CHKARY</td>
<td>CHECK ARRAY</td>
</tr>
<tr>
<td></td>
<td>CHKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td></td>
<td>CHKFRM</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td></td>
<td>EXPAND</td>
<td>EXPAND AN ARRAY</td>
</tr>
<tr>
<td></td>
<td>SCRMAN/CHGCHANG</td>
<td>POSITION</td>
</tr>
<tr>
<td></td>
<td>TRNSTR/GARGET</td>
<td>ARRAY INFORMATION</td>
</tr>
<tr>
<td></td>
<td>WRTFDL/ARYARRAY</td>
<td>REFERENCE</td>
</tr>
<tr>
<td>ACCESS</td>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>ADDFRM</td>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td></td>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td></td>
<td>EDTMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td></td>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td></td>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td></td>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td></td>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td></td>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td></td>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
</tr>
<tr>
<td></td>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
<tr>
<td>ATOI</td>
<td>VALINP/CCKCHECK</td>
<td>FIELD</td>
</tr>
<tr>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>ITEM</td>
</tr>
</tbody>
</table>

3-46
## FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**BLEN**
- CHKFLD: CHECK FIELD
- FLSTRC: FIELD STRUCTURE TRANSLATION
- GETFLS: GET FDL SOURCE FILE
- GITMD: GET ITEM DATA AND INSERT IN STRUCTURE
- WRTFDL: WRITE FDL FILE

**CLSFRM**
- VIEW: VIEW A FORM

**COPFLD**
- EXPAND: EXPAND AN ARRAY
- EXPAND/FIXFIX: UP A FORM

**COPY**
- FDFE: FORMS DRIVEN FORM EDITOR

**DELFLD**
- DRFRM: DROP FORM
- DRPWHL: DROP WHOLE
- EDTFLD: EDIT FIELD
- EXPAND/FIXFIX: UP A FORM
- FLANCI: FLAN CALLABLE INTERFACE
- FREBUF: FREE BUFFER
- MODFLD: MODIFY FIELD
- SAVFLS: SAVE FDL SOURCE
- TRNSCR: TRANSLATE SCREEN TO STRUCTURE
FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>------</td>
<td>------</td>
<td>-----------</td>
<td>-------</td>
</tr>
</tbody>
</table>

ESCPY
ADDEXT  ADD EXTENSION TO FILE_NAME
CHKPRM  CHECK PARAMETER
CPYFRM  COPY FORM
FIFDST  FILL IN FIELD STRUCTURE
GETFLS  GET FDL SOURCE FILE
GITMD  GET ITEM DATA AND INSERT IN STRUCTURE
GTFDTX  GET FIELD TEXT
GTNMFD  GET NAMED FIELD
INSFLD  INSERT FIELD
INSFRM  INSERT FORM
MODFRM  MODIFY FORM
SAVFLS  SAVE FDL SOURCE
SCRMAN/GETGET ROW
VALINP/CCKCHECK FIELD
VALINP/CCKCHECK FORM
VALINP/CCKCHECK HELP
VALINP/CCKCHECK ITEM
VALINP/CCKCHECK FOR RESERVED WORD
WRTFDL  WRITE FDL FILE

FCLOSE
FDFE  FORMS DRIVEN FORM EDITOR
GETFLS  GET FDL SOURCE FILE
LISTIT  LIST IT
SAVFLS  SAVE FDL SOURCE

FEOF
LISTIT  LIST IT

FERROR
# FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>Name</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------</td>
<td>--------</td>
<td>------------------------------</td>
</tr>
</tbody>
</table>

1. **PCFIL**  
   Process Temporary File

2. **FGETS**  
   PRCFIL Process Temporary File

3. **FOPEN**  
   FDFE FORMS DRIVEN FORM EDITOR  
   GETFLS GET FDL SOURCE FILE  
   LISTIT LIST IT  
   SAVFLS SAVE FDL SOURCE

4. **PRINTF**  
   WRTFDL WRITE FDL FILE

5. **FREE**  
   CHKFLD CHECK FIELD  
   CHKFRM CHECK FORM  
   GITMD GET ITEM DATA AND INSERT IN STRUCTURE  
   GTFDTX GET FIELD TEXT  
   MODFLD MODIFY FIELD  
   MODFRM/FREFREE TEXT  
   TRNSCR/FRLFREE LOCATION STRUCTURES  
   WRTEXP WRITE EXPRESSION

6. **GATDEF**  
   FIFDST FILL IN FIELD STRUCTURE  
   INSFLD INSERT FIELD  
   INSFRM INSERT FORM
FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**GDATA**

<table>
<thead>
<tr>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
<td></td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
<td></td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
<td></td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
<td></td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
<td></td>
</tr>
<tr>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
<td></td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
<td></td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
<td></td>
</tr>
</tbody>
</table>

**GETCUR**

<table>
<thead>
<tr>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
<td></td>
</tr>
<tr>
<td>SCRMAN</td>
<td>CHGCHANG POSITION</td>
<td></td>
</tr>
</tbody>
</table>

**GWINDO**

<table>
<thead>
<tr>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIEW</td>
<td>VIEW A FORM</td>
<td></td>
</tr>
</tbody>
</table>

**INITAL**

<table>
<thead>
<tr>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
<td></td>
</tr>
</tbody>
</table>

**INITFP**

<table>
<thead>
<tr>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
<td></td>
</tr>
</tbody>
</table>

**ISALPHA**

<table>
<thead>
<tr>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALINP</td>
<td>VALINP/CKCHECK FIELD</td>
<td>3-50</td>
</tr>
<tr>
<td>System</td>
<td>Module</td>
<td>Module Name</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCKCHECK</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISSPACE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSCR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSCR/GTPGET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSCR/PARPARSE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAKFLD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>INSFLD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>INSFRM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MALLOC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GETFLS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GITMD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GTFDTX</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODFRM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MYALLOC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSCR/FLCFILL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSCR/GTPGET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MATOI</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GITMD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GTFDTX/GTXGET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>INSFLD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODFLD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODFRM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCRMAN/GETGET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCKCHECK</td>
</tr>
</tbody>
</table>

3-51
<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module Name</td>
<td>Module Purpose</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td></td>
</tr>
<tr>
<td>CHKFLD</td>
<td>CHECK FIELD</td>
<td></td>
</tr>
<tr>
<td>CHKFRM</td>
<td>CHECK FORM</td>
<td></td>
</tr>
<tr>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
<td></td>
</tr>
<tr>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>INSFLD</td>
<td>INSERT FIELD</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/CHGCHANG</td>
<td>POSITION</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/GTPGET</td>
<td>PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td>TRNSCR/MTCHATCH</td>
<td>PROMPT WITH FIELD</td>
<td></td>
</tr>
<tr>
<td>TRNSTR/FLFFILL</td>
<td>FIELD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MEMCMP</td>
<td></td>
</tr>
<tr>
<td>CHKPRM</td>
<td>CHECK PARAMETER</td>
<td></td>
</tr>
<tr>
<td>CPYFRM</td>
<td>COPY FORM</td>
<td></td>
</tr>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
<td></td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
<td></td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
<td></td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
<td></td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
<td></td>
</tr>
<tr>
<td>FDFE/MAIN</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
<td></td>
</tr>
<tr>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
<td></td>
</tr>
<tr>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>GNXTFD</td>
<td>GET NEXT FIELD</td>
<td></td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
<td></td>
</tr>
<tr>
<td>INSWHL</td>
<td>INSERT WHOLE</td>
<td></td>
</tr>
<tr>
<td>LAYOUT</td>
<td>LAYOUT MODE</td>
<td></td>
</tr>
<tr>
<td>LISTFM</td>
<td>LIST FORMS</td>
<td></td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
<td></td>
</tr>
<tr>
<td>MODWHL</td>
<td>MODIFY WHOLE</td>
<td></td>
</tr>
<tr>
<td>PRSCMD</td>
<td>PARSE COMMAND</td>
<td></td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
<td></td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/CHGCHANG</td>
<td>POSITION</td>
<td></td>
</tr>
<tr>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
<td></td>
</tr>
<tr>
<td>VALIDATE</td>
<td>INPUT</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKCHECK</td>
<td>FIELD</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKCHECK</td>
<td>FORM</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKCHECK</td>
<td>HELP</td>
<td></td>
</tr>
</tbody>
</table>

3-52
### FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>ITEM</td>
</tr>
<tr>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>NAME</td>
</tr>
<tr>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>PROMPT</td>
</tr>
<tr>
<td></td>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
</tbody>
</table>

**MEMCPY**

- **CHKFLD**: CHECK FIELD
- **CPIFRM**: COPY FORM
- **EDTFLD**: EDIT FIELD
- **EDTWHL**: EDIT WHOLE
- **FDFE**: FORMS DRIVEN FORM EDITOR
- **FLSTRC**: FIELD STRUCTURE TRANSLATION
- **GETFLS**: GET FDL SOURCE FILE
- **GITMD**: GET ITEM DATA AND INSERT IN STRUCTURE
- **GWHINP**: GET WHOLE INPUT
- **LISTFM**: LIST FORMS
- **TRNSCR**: TRANSLATE SCREEN TO STRUCTURE
- **TRNSCR/LDPLOAD**: PROMPT INFORMATION
- **WRTEXP**: WRITE EXPRESSION

**MEMDGT**

- **GITMD**: GET ITEM DATA AND INSERT IN STRUCTURE

**MEMSET**

- **CHKFLD**: CHECK FIELD
- **DRPWHL**: DROP WHOLE
- **EDTFLD**: EDIT FIELD
- **EDTMOD**: EDIT MODE
- **EDTWHL**: EDIT WHOLE
- **FDFE**: FORMS DRIVEN FORM EDITOR
- **FLFMST**: FIELD TO FORM STRUCTURE TRANSLATION
- **FLSTRC**: FIELD STRUCTURE TRANSLATION
- **FLWHST**: FILL WHOLE STRUCTURE
- **GETFLS**: GET FDL SOURCE FILE
### System Module Module Name Purpose

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODFLD</td>
<td>MODIFY FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRSCMD</td>
<td>PARSE COMMAND</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSTR</td>
<td>TRANSLATE STRUCTURE TO SCREEN</td>
</tr>
</tbody>
</table>

### MIN

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN</td>
<td>STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>SCRMAN/CHGCHANG</td>
<td>POSITION</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRNSCR/GTFGET</td>
<td>PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRNSCR/PARPARSE</td>
<td>SCREEN DATA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRNSTR/FLFFILL</td>
<td>FIELD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRNSTR/FLPFILL</td>
<td>PROMPT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>FIELD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>FORM</td>
<td></td>
</tr>
</tbody>
</table>

### MITOA

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRNSCR/GTFGET</td>
<td>FORM PROMPT INFORMATION</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRNSCR/LDPLOAD</td>
<td>PROMPT INFORMATION</td>
<td></td>
</tr>
</tbody>
</table>

### MKTEMP

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LISTIT</td>
<td>LIST IT</td>
<td></td>
</tr>
</tbody>
</table>

### OISCR

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EDTMOD</td>
<td>EDIT MODE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
<td></td>
</tr>
</tbody>
</table>
FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
</tr>
<tr>
<td></td>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td></td>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td></td>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td></td>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
</tr>
<tr>
<td></td>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
</tbody>
</table>

PDATA

<table>
<thead>
<tr>
<th>Module</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
</tr>
</tbody>
</table>

PMSGLC

<table>
<thead>
<tr>
<th>Module</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
</tr>
</tbody>
</table>

PMSGLS

<table>
<thead>
<tr>
<th>Module</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>ERROR</td>
<td>ISSUE ERROR MESSAGE</td>
</tr>
<tr>
<td>FATAL</td>
<td>ISSUE FATAL ERROR MESSAGE</td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>INSFLD</td>
<td>INSERT FIELD</td>
</tr>
<tr>
<td>INSWHL</td>
<td>INSERT WHOLE</td>
</tr>
</tbody>
</table>
FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>LISTFM</td>
<td>LIST FORMS</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
<td>LIST IT</td>
</tr>
<tr>
<td>MODFLD</td>
<td>MODIFY FIELD</td>
<td>MODIFY FIELD</td>
</tr>
<tr>
<td>MODWHL</td>
<td>MODIFY WHOLE</td>
<td>MODIFY WHOLE</td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
<td>SCREEN MANAGER</td>
</tr>
<tr>
<td>SCRMAN/CHGCHANG</td>
<td>POSITION</td>
<td>POSITION</td>
</tr>
<tr>
<td>TRNSTR/FLFFILL</td>
<td>FIELD</td>
<td>TRNSTR/FLFFILL FIELD</td>
</tr>
<tr>
<td>VIEW</td>
<td>VIEW A FORM</td>
<td>VIEW A FORM</td>
</tr>
<tr>
<td>WARNING</td>
<td>ISSUE WARNING MESSAGE</td>
<td>ISSUE WARNING MESSAGE</td>
</tr>
<tr>
<td>PUTATT</td>
<td>DROP WHOLE</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td>PUTCUR</td>
<td>EDIT FIELD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td>PUTERR</td>
<td>PUT ERROR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td>RENAME</td>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
<td>SAVE FDL SOURCE</td>
</tr>
<tr>
<td>REWIND</td>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
</tr>
</tbody>
</table>
FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**RMVPAG**

<table>
<thead>
<tr>
<th>Module</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTMOD</td>
<td>EDIT MODE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td>SCRMAN</td>
<td>SCREEN MANAGER</td>
</tr>
<tr>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
</tbody>
</table>

**RSVATT**

EXPAND EXPAND AN ARRAY
EXPAND/FIXFIX UP A FORM

**SPRINTF**

<table>
<thead>
<tr>
<th>Module</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDEXT</td>
<td>ADD EXTENSION TO FILE_NAME</td>
</tr>
<tr>
<td>DRPWHL</td>
<td>DROP WHOLE</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td>ERROR</td>
<td>ISSUE ERROR MESSAGE</td>
</tr>
<tr>
<td>FATAL</td>
<td>ISSUE FATAL ERROR MESSAGE</td>
</tr>
<tr>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td>FLMST</td>
<td>FIELD TO FORM STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td>FLSTRC</td>
<td>FIELD STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td>GETFLS/TRETREE EXPRESSION</td>
<td></td>
</tr>
<tr>
<td>GFDINP</td>
<td>GET FIELD INPUT</td>
</tr>
<tr>
<td>GWHINP</td>
<td>GET WHOLE INPUT</td>
</tr>
<tr>
<td>INSWHL</td>
<td>INSERT WHOLE</td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td>MODWHL</td>
<td>MODIFY WHOLE</td>
</tr>
</tbody>
</table>
### FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSTR/FLFFILL FIELD</td>
<td>VALIDATE INPUT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCCKCHECK FLD</td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WARNING</td>
<td>ISSUE WARNING MESSAGE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WRTEXP</td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td>STRASN</td>
<td></td>
<td>STRASN</td>
<td>CHECK ARRAY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CHKARY</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td>STRGAT</td>
<td></td>
<td>STRGAT</td>
<td>ADD EXTENSION TO FILE_NAME</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ADDEXT</td>
<td>GET FDL SOURCE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GETFLS</td>
<td>GET FDL SOURCE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GETFLS/TRETREE EXPRESSION</td>
<td>LIST IT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PUTERR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WRTFDL/ARYARRAY REFERENCE</td>
<td>WRITE EXPRESSION</td>
</tr>
</tbody>
</table>

### STRCHR

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCCKCHECK FLD</td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCCKCHECK FORM</td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCCKCHECK HELP</td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCCKCHECK ITEM</td>
<td>WRITE EXPRESSION</td>
</tr>
</tbody>
</table>
## FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>STRCMP</td>
<td>EXPAND/FIXFIX UP A FORM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FNDATT</td>
<td>FIND ATTRIBUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GFLDPT</td>
<td>GET FIELD POINTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GTNMFD</td>
<td>GET NAMED FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>HELP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>FOR RESERVED WORD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WRTFDL</td>
<td>WRITE FDL FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STRCPY</td>
<td>CSTASH CHARACTER STASH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FLSTRC</td>
<td>FIELD STRUCTURE TRANSLATION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PUTERR</td>
<td>PUT ERROR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STRLEN</td>
<td>CHKFLD CHECK FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CHKFrm</td>
<td>CHECK FORM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CSTASH CHARACTER STASH</td>
<td>CHKFLD CHECK FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EDTFLD</td>
<td>EDIT FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EDTWHL</td>
<td>EDIT WHOLE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ERROR</td>
<td>ISSUE ERROR MESSAGE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FATAL</td>
<td>ISSUE FATAL ERROR MESSAGE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FDFE</td>
<td>FORMS DRIVEN FORM EDITOR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GTNMFD</td>
<td>GET NAMED FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LISTFM</td>
<td>LIST FORMS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRSCMD</td>
<td>PARSE COMMAND</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCRMAN/GETGET</td>
<td>ROW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNSCR</td>
<td>TRANSLATE SCREEN TO STRUCTURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>HELP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WARNING</td>
<td>ISSUE WARNING MESSAGE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WRTEXP</td>
<td>WRITE EXPRESSION</td>
</tr>
<tr>
<td>System Module</td>
<td>Module Name</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>STRNCPY</td>
<td>PRSCMD</td>
<td>PARSE COMMAND</td>
</tr>
<tr>
<td></td>
<td>STRRNCPY</td>
<td>PRSCMD</td>
<td>PARSE COMMAND</td>
</tr>
<tr>
<td></td>
<td>STRRCHR</td>
<td>ADDEXT</td>
<td>ADD EXTENSION TO FILE_NAME</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRCFIL</td>
<td>PROCESS TEMPORARY FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCRMAN/GETGET</td>
<td>GET ROW</td>
</tr>
<tr>
<td></td>
<td>STRSPN</td>
<td>VALINP/CCKCHECK</td>
<td>FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALINP/CCKCHECK</td>
<td>HELP</td>
</tr>
<tr>
<td></td>
<td>STRUPC</td>
<td>VALINP/CCKCHECK</td>
<td>HELP</td>
</tr>
<tr>
<td></td>
<td>SYSMSG</td>
<td>CHKFLD</td>
<td>CHECK FIELD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GETFLS</td>
<td>GET FDL SOURCE FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GITMD</td>
<td>GET ITEM DATA AND INSERT IN STRUCTURE</td>
</tr>
</tbody>
</table>
FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>GTFDTX</td>
<td>GET FIELD TEXT</td>
</tr>
<tr>
<td>LISTIT</td>
<td>LIST IT</td>
</tr>
<tr>
<td>MODFRM</td>
<td>MODIFY FORM</td>
</tr>
<tr>
<td>SAVFLS</td>
<td>SAVE FDL SOURCE</td>
</tr>
<tr>
<td>VIEW</td>
<td>VIEW A FORM</td>
</tr>
</tbody>
</table>

SYSTEM

LISTIT LIST IT

TERMFP

FDFE/MAIN MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)

TRMNAT

FDFE/MAIN MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)

UNLINK

FDFE FORMS DRIVEN FORM EDITOR
LISTIT LIST IT

WRTFRM

SAVFLS SAVE FDL SOURCE

YYPARSE

FLANCI FLAN CALLABLE INTERFACE
### FORMS DRIVEN FORM EDITOR Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3-62
3.10.7 Main Program Parts List

The following lists each Main Program listed in 3.10.1 and all the modules which are called either by that module itself or by any of the documented modules which it calls. It is possible for a non-main module to be listed more than once if it is called by multiple modules. The called modules, in this case known as program parts, are marked as to whether they are documented here. If so, the phrase "well-defined module" appears by the module name, if not it is an "external routine". The Purpose of the Main Program module is listed as well.
### FORMS DRIVEN FORM EDITOR Main Program Parts List

<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHKFLD</td>
<td>Purpose--&gt;CHECK FIELD</td>
<td></td>
</tr>
<tr>
<td>BLEN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ERROR</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FATAL</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FNDATT</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FREE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MALLOC</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MAX</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MEMCPY</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MEMSET</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MYALLOC</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>PHSGLS</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>SPRINTF</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>STRCMP</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>STRLEN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>SYSMSG</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>WRTEXP</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>Module</td>
<td>Purpose</td>
<td>Type</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>ABORT</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ABS</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ACCESS</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ADDEXT</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>ADDFRM</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ATOI</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>BLEN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CHKPRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>CLSFRM</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>COPY</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CPYFRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DELFLD</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>DRPFRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DRPWHL</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>EDTFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>EDTMOD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>EDTWHL</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>ESCPY</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FCLOSE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FDFE</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FEOF</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FERROR</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FGETS</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FIFDST</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FLANCI</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FLMST</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FLSTRC</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FLWHST</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FNDATT</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FOPEN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>PRINTF</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FREEBUF</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FREE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>GATDEF</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>GDATA</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>GETCUR</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>GETFLS</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GETFLS/TREEEXP</td>
<td></td>
<td>Well-defined module</td>
</tr>
<tr>
<td>Main Pgm Name</td>
<td>Module Name</td>
<td>Module Type</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>GETLEN</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GFDINP</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GITMD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GNXTFD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GNXTFD/NXTFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GTFPRD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GTFDTP</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GTFDTP/GTXINF</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GTNMF</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GWHINP</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GWINDO</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>INITAL</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>INITFP</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>INSFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>INSFRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>INSWHI</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>ISALPHA</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ISSPACE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>LAYOUT</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>LISTFM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>LISTIT</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>MAKFLD</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MALLOC</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MATOI</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MAX</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MEMCMP</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MEMCPY</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MEMDGT</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MEMSET</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MIN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MITOA</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MKTEMP</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>MODFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>MODFRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>MODFRM/FRETXT</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>MODWHL</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>OISCR</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>PDATA</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>PMSGSLC</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>PMSGLS</td>
<td>External routine</td>
<td></td>
</tr>
</tbody>
</table>
# FORMS DRIVEN FORM EDITOR Main Program Parts List

<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDFE/MAIN</td>
<td>Purpose</td>
<td>MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)</td>
</tr>
<tr>
<td>ABORT</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ABS</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ACCESS</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ADDEXT</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>ADDFRM</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ATOI</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>BLEN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CHKPRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>CLSFRM</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>COPY</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CPYFRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DELFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DRPFRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>DRPWHL</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>EDTFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>EDTMOD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>EDTWHL</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>ESCPY</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FCLOSE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FDFE</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FEOF</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FERROR</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FGETS</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FIFDST</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FLANCI</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FLFMST</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FLSTRC</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FLWHST</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FNDATT</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FOPEN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FPRINTF</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>FREBUF</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>FREE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>GATDEF</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>GDATA</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>GETCUR</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>GETFLS</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>GETFLS/TREEXP</td>
<td>Well-defined module</td>
<td></td>
</tr>
</tbody>
</table>

3-67
<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GETLEN</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GFDINP</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GITMD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GNXTFD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GNXTFD/NXTFLD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GTCPFD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GTFDTX</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GTFDTX/GTXINF</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GTNMFD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GWHINP</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>GWINDO</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>INITIAL</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>INITFP</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>INSFLD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>INSFRM</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>INSWHL</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>ISALPHA</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>ISSPACE</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>LAYOUT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>LISTFM</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>LISTIT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>MAKFLD</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MALLOC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MATOI</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MAX</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MEMCMP</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MEMCPY</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MEMDGST</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MEMSET</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MIN</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MITOA</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MKTEMP</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MODFLD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>MODFRM</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>MODFRM/FRETXT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>MODWHL</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>OISCR</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PDATA</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PMSGLC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PMSGLS</td>
<td>External routine</td>
</tr>
</tbody>
</table>
## FORMS DRIVEN FORM EDITOR Main Program Parts List

<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PRCFIL</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PREC</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PRSCMD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PUTATT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PUTCUR</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>PUTERR</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>RENAME</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>REWIND</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>RMVPAG</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SAVFLS</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>SCRMAN</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>SCRMAN/CHGPOS</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>SCRMAN/GETROW</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>SPRINTF</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRCAT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRCHR</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRCMP</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRCPY</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRLEN</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRNCPY</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRRCHR</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRSPN</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRUPC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SYMSG</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SYSTEM</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TERMFP</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TRNMAT</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>TRNSCR</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/FLGOST</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/FRLCST</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/GTFMPMT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/GTPINF</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/LDPMINF</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/MTPCHMPT</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/PARSRCRN</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSCR/SPSYMB</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSTR</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSTR/FLFLD</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>TRNSTR/FLPRMPT</td>
<td>Well-defined module</td>
</tr>
</tbody>
</table>
## FORMS DRIVEN FORM EDITOR Main Program Parts List

<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRNSTR/GARINF</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>UNLINK</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>VALINP</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKFLD</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKFRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKHLP</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKITM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKNAM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKPRM</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>VALINP/CCKRSV</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>VIEW</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>WRTFDL</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>WRTFDL/ARYREF</td>
<td>Well-defined module</td>
<td></td>
</tr>
<tr>
<td>WRTFRM</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>YYPARSE</td>
<td>External routine</td>
<td></td>
</tr>
</tbody>
</table>

3-70
```
<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAKINT</td>
<td>FATAL</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>MALLOC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MYALLOC</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PMSGLS</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SPRINTF</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRLEN</td>
<td>External routine</td>
</tr>
</tbody>
</table>
```

Purpose: MAKE EXPRESSION INTO AN INTEGER
<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MAKSTR</td>
<td>Purpose: MAKE EXPRESSION INTO A STRING</td>
</tr>
<tr>
<td></td>
<td>FATAL</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>MALLOC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MYALLOC</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PMSGLS</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SPRINTF</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRLEN</td>
<td>External routine</td>
</tr>
</tbody>
</table>
### FORMS DRIVEN FORM EDITOR Main Program Parts List

<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>MKPOS</td>
<td>FATAL</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>MALLOC</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>MYALLOC</td>
<td>Well-defined module</td>
</tr>
<tr>
<td></td>
<td>PMSGLS</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>SPRINTF</td>
<td>External routine</td>
</tr>
<tr>
<td></td>
<td>STRLEN</td>
<td>External routine</td>
</tr>
</tbody>
</table>

**Purpose:** MAKE POSITION NODE
3.10.8 Module Documentation

The following documentation describes information which is specific to each individual module being documented in this specification as listed in section 3.10.2. It provides a compact way of getting information that would be otherwise buried within each module's source code.

The specific items in this module documentation have the following meanings:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME:</td>
<td>Name of program Module.</td>
</tr>
<tr>
<td>PURPOSE:</td>
<td>Purpose of Module as detailed in the source code.</td>
</tr>
<tr>
<td>LANGUAGE:</td>
<td>Programming language source code is written in.</td>
</tr>
<tr>
<td></td>
<td>The choices are:</td>
</tr>
<tr>
<td></td>
<td>VAX-11 FORTRAN</td>
</tr>
<tr>
<td></td>
<td>C (I/S-1 Workbench 'C')</td>
</tr>
<tr>
<td></td>
<td>VAX-11 COBOL</td>
</tr>
<tr>
<td>MODULE TYPE:</td>
<td>Whether a Program, Subroutine, or Function.</td>
</tr>
<tr>
<td>SOURCE FILE:</td>
<td>Name of Source File from file specification.</td>
</tr>
<tr>
<td>SOURCE FILE TYPE:</td>
<td>Source File Extension from file specification.</td>
</tr>
<tr>
<td>HOST:</td>
<td>Whether this is a host-dependent routine (VAX or IBM) or blank if host-independent.</td>
</tr>
<tr>
<td>SUBSYSTEM:</td>
<td>IISS sub-system this file resides in.</td>
</tr>
<tr>
<td>SUBDIRECTORY:</td>
<td>Sub-directory of that subsystem in which this file resides.</td>
</tr>
<tr>
<td>DOCUMENTATION GROUP:</td>
<td>Name of documentation group of which this source file is a member.</td>
</tr>
<tr>
<td>DESCRIPTION:</td>
<td>A description of the module as obtained</td>
</tr>
</tbody>
</table>
from the source code.

ARGUMENTS: The arguments with which this routine is called if it is a Subroutine or a Function.

INCLUDE FILES: A list of all the files that are included into this module as well as their purposes.

ROUTINES CALLED: Subroutines or Functions, either documented or external, called by this module, if any.

CALLED DIRECTLY BY: The documented routines which call this module, if any.

USED IN MAIN PROGRAM(S): The documented Main Programs which contain this module in their parts list according to the list in section 3.10.7.

The Module Documentation is arranged alphabetically according to Module Name.
NAME: ADDCHK
PURPOSE: ADD POSITION TO CHECK LIST
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS
VOID ADDCHK(POSPTR)
POSPTR;

DESCRIPTION
ADDS THE SPECIFIED POSITION TO THE OVERLAP CHECK LIST

ARGUMENTS:

POSPTR = POS *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

FLDTYPE - FIELD TYPE
ERROR - ISSUE ERROR MESSAGE

CALLED DIRECTLY BY:

3-76
CHKFRM - CHECK FORM

USED IN MAIN PROGRAM(S):

CHKFRM - CHECK FORM
NAME: ADDEXT
PURPOSE: ADD EXTENSION TO FILE_NAME
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: ADDEXT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE
DESCRIPTION:

SYNOPSIS

VOID ADDEXT(NAME, FLG, FILE_NAME)
ENAME NAME;
INT FLG;
CHAR FILE_NAME[];

INPUTS/OUTPUTS:

INPUTS:
ENAME NAME: NAME OF FILE WITHOUT EXTENSION.
INT FLG: TYPE OF EXTENSION AND DIRECTORY TO ADD.

OUTPUTS:
CHAR FILE_NAME[]: NEW NAME WITH EXTENSION ADDED.

DESCRIPTION
CONCATENATES THE SPECIFIED EXTENSION ONTO THE NAME AND
RETURNS IT IN FILE_NAME.

ARGUMENTS:

NAME = CHAR []
FLG = INT
FILE_NAME = CHAR [ ]

INCLUDE FILES:
---------------------
STDTYP         - STANDARD TYPE DEFINITIONS
FPD          - FORM PROCESSOR DATA
FDFEFM         - FDFE FORM DEFINITIONS
FDFE         - FDFE DATA STRUCTURES

ROUTINES CALLED:
---------------------
ESCPY
STRRCHR
STRCAT
SPRINTF

CALLED DIRECTLY BY:
---------------------
FDFE         - FORMS DRIVEN FORM EDITOR

USED IN MAIN PROGRAM(S):
---------------------
FDFE/MAIN        - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: CHKARY
PURPOSE: CHECK ARRAY
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS
VOID CHKARY(ARYPTR)
    FIELD *ARYPTR;

DESCRIPTION
GENERATES POSITIONS FOR EACH ELEMENT OF AN ARRAY FOR OVERLAP CHECKING

ARGUMENTS:

ARYPTR = FIELD *

INCLUDE FILES:

STDTTYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

MYALLOC - MY MALLOC
ABS
STRASN
PS 620144402
1 November 1985

CALLED DIRECTLY BY:

-----------------------------
CHKFRM - CHECK FORM

USED IN MAIN PROGRAM(S):

-----------------------------
CHKFRM - CHECK FORM
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: CHKFLD
PURPOSE: CHECK FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:
SYNOPSIS
VOID CHKFLD()

DESCRIPTION
CHECKS THE CURRENT FIELD FOR COMPLETENESS AND CONSISTENCY

INCLUDE FILES:
STDTPY - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:
FNDATT - FIND ATTRIBUTE
ERROR - ISSUE ERROR MESSAGE
MEMSET
MAX
FREE
WRTEXP - WRITE EXPRESSION
BLEN
MEMCPY
SYSMSG
MYALLOC - MY MALLOC
STRLEN

3-82
NAME: CHKFRM
PURPOSE: CHECK FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *( )
SOURCE FILE: FLANS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

VOID CHKFRM()

DESCRIPTION

CHECKS THE CURRENT FORM FOR COMPLETENESS AND CONSISTENCY

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
STDOI - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

WARNING - ISSUE WARNING MESSAGE
ADDCHK - ADD POSITION TO CHECK LIST
CHKARY - CHECK ARRAY
ABS
STRLEN
FREE
FLDTYPE - FIELD TYPE
ERROR - ISSUE ERROR MESSAGE
GFLDPT - GET FIELD POINTER
ABS
MAX
PS 620144402
1 November 1985

STRASN
FNDATT - FIND ATTRIBUTE
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: CHKPRM
PURPOSE: CHECK PARAMETER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: CHKPRM
SOURCE FILE TYPE: C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE
DESCRIPTION:

SYNOPSIS

BOOL CHKPRM(PARAMTR, PARSIZ, RTPARM);
    CHAR PARAMTR[];
    INT PARSIZ;
    CHAR RTPARM[];

INPUTS/OUTPUTS:

INPUTS:
    PARAMTR - CHAR STRING (NON NULL TERMINATED) CONTAINING THE PARAMETER
    BEING CHECKED FOR NOT BEING ENTERED.
    PARSIZE - LENGTH OF CHARACTER STRING CONTAINING PARAMETER

OUTPUTS:
    RTPARM - NULL TERMINATED STRING CONTAINING PARAMETER
    RETURNS A BOOL FALSE IF PARAMETER IS A BLANK

DESCRIPTION
    CHECKS TO MAKE SURE THAT PARAMETER HAS BEEN ENTERED BY USER AND COPIES IT INTO A NULL TERMINATED STRING PROVIDED BY CALLER

ARGUMENTS:

3-85
PARAMTR = CHAR []
PARSZ = INT
RTPARM = CHAR []

INCLUDE FILES:
-------------------
STDTYPE  - STANDARD TYPE DEFINITIONS
FPD      - FORM PROCESSOR DATA
FDFEFM   - FDFE FORM DEFINITIONS
FDFE     - FDFE DATA STRUCTURES

ROUTINES CALLED:
------------------
MEMCMP
ESCPY

CALLED DIRECTLY BY:
-------------------
EDTMOD    - EDIT MODE
FDFE      - FORMS DRIVEN FORM EDITOR

USED IN MAIN PROGRAM(S):
-------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: CPYFRM
PURPOSE: COPY FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *( )
SOURCE FILE: CPYFRM
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
CHAR *CPYFRM(COPFLS ,COPFRM)
  ENAME COPFLS;
  ENAME COPFRM;

INPUTS/OUTPUTS:

INPUTS:
  COPFLS - NON NULL TERMINATED STRING CONTAINING THE NAME
           OF FLS FILE
           FROM WHICH USER WISHES TO COPY FORM.
  COPFLS - NON NULL TERMINATED STRING CONTAINING THE NAME
           OF THE FORM.

OUTPUTS:
  THIS ROUTINE FILLS (AND IS THE ONLY ROUTINE WHICH
  WRITES TO THESE)
  VARIABLES) SEVERAL GLOBAL VARIABLES:
  ALTBUF - BEGINNING OF LINK LIST CONTAINING ALL FORMS
           ETC. OF FLS.
  ALTFRM - POINTER TO FORM WHICH THE USER WHISH TO
           COPY INTO THE
           SCREEN DATA AREA.
  COPYFLS - NAME OF FLS JUST COPIED - USED TO SAVE ON
           CALLS TO
           GETFLS.
COPYFRM - NAME OF FORM JUST COPIED - USED TO SAVE ON CALLS TO GTNMFD.

DESCRIPTION
THIS ROUTINE IS USED TO COPY AN FLS FILE INTO AN ALTERNATE INTERNAL DATA STRUCTURE AND GETS THE POINTER TO THE FORM INTERESTED IN. IS USED BY WHLEDT AND FLDEDT TO RETRIEVE DATA CONCERNING A FIELD(S) ON A FORM NOT NOW BEING WORKED ON IN AN FLS THAT IS NOT BEING WORKED ON, FOR POSSIBLE INSERTION BY THE USER.

ARGUMENTS:

-------
COPFLS = ENAME
COPFRM = ENAME

INCLUDE FILES:

--------
STDTYP  - STANDARD TYPE DEFINITIONS
FPCODE  - FORM PROCESSOR RETURN CODES
FPD     - FORM PROCESSOR DATA
FDFEFM  - FDDE FORM DEFINITIONS
FDDE    - FDDE DATA STRUCTURES

ROUTINES CALLED:

----------------
MEMCMP
STRNCMP
ESCPY
GETFLS - GET FDL SOURCE FILE
GTNMFD  - GET NAMED FIELD
MEMCPY

CALLED DIRECTLY BY:

---------------------
EDTFLD  - EDIT FIELD
EDTWHL  - EDIT WHOLE

USED IN MAIN PROGRAM(S):

3-88
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: CSTASH
PURPOSE: CHARACTER STASH
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:
-------------
SYNOPSIS
CHAR *CSTASH(S)
CHAR *S;

DESCRIPTION
SAVES THE SPECIFIED CHARACTER STRING AND RETURNS A
POINTER TO IT

ARGUMENTS:
-----------
S = CHAR *

INCLUDE FILES:
---------------
STDTP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FP CODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:
-----------------
STRCPY
STRLEN
MYALLOC = MY MALLOC
FORMS DRIVEN FORM EDITOR Module Documentation

<table>
<thead>
<tr>
<th>NAME</th>
<th>DRPFRM</th>
</tr>
</thead>
<tbody>
<tr>
<td>PURPOSE</td>
<td>DROP FORM</td>
</tr>
<tr>
<td>LANGUAGE</td>
<td>C</td>
</tr>
<tr>
<td>MODULE TYPE</td>
<td>FUNCTION</td>
</tr>
<tr>
<td>FUNCTION TYPE</td>
<td>CHAR * ()</td>
</tr>
<tr>
<td>SOURCE FILE</td>
<td>DRPFRM</td>
</tr>
<tr>
<td>SOURCE FILE TYPE</td>
<td>.C</td>
</tr>
<tr>
<td>HOST</td>
<td></td>
</tr>
<tr>
<td>SUBSYSTEM</td>
<td>UI</td>
</tr>
<tr>
<td>SUBDIRECTORY</td>
<td>FDFE</td>
</tr>
<tr>
<td>DOCUMENTATION GROUP</td>
<td>FDFE</td>
</tr>
</tbody>
</table>

DESCRIPTION:

SYNOPSIS

```
CHAR *DRPFRM(FRMNAM)
ENAME FRMNAM;
```

INPUTS/OUTPUTS:

INPUTS:
FRMNAM - NON NULL TERMINATED STRING CONTAINING THE NAME OF FORM TO BE DELETED.

OUTPUTS:
RETURNS A NULL IF FOUND THE FORM AND DELETED IT AND RETURNS ERROR CODE IF COULD NOT FIND FORM TO DELETE.

DESCRIPTION
DELETES A FORM NAMED BY USER FROM INTERNAL DATA STRUCTURE - CALLS DELFLD TO DO ACTUAL WORK OF DELETEING.

ARGUMENTS:

```
FRMNAM  =  ENAME
```
INCLUDE FILES:

- STDTYPE - STANDARD TYPE DEFINITIONS
- FPCODE - FORM PROCESSOR RETURN CODES
- FPD - FORM PROCESSOR DATA
- FDFEFH - FDFE FORM DEFINITIONS
- FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

- GTNMFD - GET NAMED FIELD
- DELFDL

CALLED DIRECTLY BY:

- EDTMOD - EDIT MODE

USED IN MAIN PROGRAM(S):

- FDFE/MIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: DRPWHL
PURPOSE: DROP WHOLE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: DRPWHL
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *DRPWHL(FRMPNT,PFKEY)
FIELD *FRMPNT;
INT *PFKEY;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM FROM WHICH FIELDS MARKED WILL BE DROPPED

OUTPUTS:
PFKEY - RETURNS TO CALL THE PFKEY RECEIVED FROM OISCR.

DESCRIPTION
DROPS ALL FEILDS MARKED BY USER

ARGUMENTS:

FRMPNT = FIELD *
PFKEY = INT *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS

3-93
FP CODE  - FORM PROCESSOR RETURN CODES
FP ARM   - FORM PROCESSOR PARAMETERS
FP D    - FORM PROCESSOR DATA
FP DFM   - FDFE FORM DEFINITIONS
FP F    - FDFE DATA STRUCTURES

ROUTINES CALLED:
------------------
ADDFRM
MEMCMP
DELFLD
GDATA
PMSGLS
RMVPAG
OISCR
MEMSET
FLSTRC   - FIELD STRUCTURE TRANSLATION
SPRINTF
PDATA
PUTATT

CALLED DIRECTLY BY:
------------------
EDTWHL   - EDIT WHOLE

USED IN MAIN PROGRAM(S):
-----------------------
FDFE/MAIN  - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: EDTFLD
PURPOSE: EDIT FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *
SOURCE FILE: EDTFLD
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *EDTFLD(FRMPNT, RDONLY, MODE, ROW, COL)
FIELD *FRMPNT;
BOOL RDONLY;
BOOL MODE;
INT ROW;
INT COL;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM WORKING ON
RDONLY - FLAG INDICATING WHETHER READ ONLY IS ALLOWED OR NOT
MODE - FLAG INDICATING WHICH MODE NOW IN (LAYOUT OR EDIT FIELD)
ROW - CURSOR POSITION COMING FROM LAYOUT (WILL BOTH BE 0 IF
COL IN FIELD EDIT MODE.

OUTPUTS:
IN EDIT FIELD MODE:
RETURNS A NULL IF QUITING
RETURNS AN ERROR CODE IF ABNORMALLY TERMINATED
IN LAYOUT MODE:
RETURNS A NULL IF GOING TO LAYOUT AGAIN

3-95
Returning a exit of DFE if quitting returns an error code if abnormally terminated.

Description
This is the driver module for field edit mode and the detailed edit mode for layout. It allows user to insert, modify and delete fields one at a time as well as modifying form information (prompt and background information) and to copy field data from another form in another FLS file into screen data area.

Arguments:

- **FRMFNT** - field
- **RDONLY** - bool
- **MODE** - bool
- **ROW** - int
- **COL** - int

Include Files:

- **STDIO** - """" PURPOSE NOT FOUND BY STRIPPER """
- **STDTYPE** - STANDARD TYPE DEFINITIONS
- **CTYPE** - """" PURPOSE NOT FOUND BY STRIPPER """
- **FPPARM** - FORM PROCESSOR PARAMETERS
- **FPCODE** - FORM PROCESSOR RETURN CODES
- **FPD** - FORM PROCESSOR DATA
- **FDFEFM** - FDFE FORM DEFINITIONS
- **FDFE** - FDFE DATA STRUCTURES

Routines Called:

- **ADDFRM**
- **MEMCMP**
- **RMVFAG**
- **GDATA**
- **PHSGLD**
- **VALINP** - VALIDATE INPUT
- **PHSGLC**
- **CPYFRM** - COPY FORM
- **GNXTFD** - GET NEXT FIELD
- **GFDFINP** - GET FIELD INPUT
- **MODFLD** - MODIFY FIELD

3-96
INSFLD - INSERT FIELD
MODFRM - MODIFY FORM
PUTCUR
OISCR
PUTATT
SPRINTF
MEMSET
MEMCPY
STRLEN
FLSTRC - FIELD STRUCTURE TRANSLATION
DELFLD
PDATA
GTCPFD - GET USING CURSOR POSITION FIELD
FLFMST - FIELD TO FORM STRUCTURE TRANSLATION

CALLED DIRECTLY BY:
----------
EDTMOD - EDIT MODE
LAYOUT - LAYOUT MODE

USED IN MAIN PROGRAM(S):
----------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: EDTMOD
PURPOSE: EDIT MODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: EDTMOD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *EDTMOD(NEWOLD,EDTFLAG)
BOOL NEWOLD;
INT EDTFLAG;

INPUTS/OUTPUTS:

INPUTS:
NEWOLD - FLAG INDICATING WHETHER FORM IS NEW OR AN OLD FORM
EDTFLAG - FLAG INDICATING TYPE OF EDITING CHANGE/SELECT

OUTPUTS:
RETURNS NULL IF QUIT KEY PRESSED
RETURNS EXITFDFE IF EXIT OPTION CHOSE
RETURNS ERROR CODE IF ABNORMALLY TERMINATED

DESCRIPTION
MAIN DRIVER MODULE FOR EDIT MODE. IN ADDITION TO ALLOWING
USER TO CHOOSE EDIT MODE(WHOLE,FIELD,LAYOUT). USER CAN VIEW
FORM. GET LIST OF FORMS IN FLS AND SAVE (AND COMPILE) SOURCE

ARGUMENTS:

3-98
NEWOLD = BOOL
EDTFLAG = INT

INCLUDE FILES:

STDTPP - STANDARD TYPE DEFINITIONS
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

ADDFRM
MEMCMP
FREBUF - FREE BUFFER
PRSCMD - PARSE COMMAND
CHEKFRM - CHECK PARAMETER
MEMSET
GDATA
PMSGLS
RMVPAG
SAVFLS - SAVE FDL SOURCE
DRPFRM - DROP FORM
INSFRM - INSERT FORM
LAYOUT - LAYOUT MODE
EDTFLD - EDIT FIELD
EDTWHL - EDIT WHOLE
GTNMFD - GET NAMED FIELD
LISTFM - LIST FORMS
OISCR
PUTATT
PDATA

CALLED DIRECTLY BY:

FDFE - FORMS DRIVEN FORM EDITOR

USED IN MAIN PROGRAM(S):

FDFE/MND - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)

3-99
FORMS DRIVEN FORM EDITOR Module Documentation

NAME:          EDTWHL
PURPOSE:       EDIT WHOLE
LANGUAGE:      C
MODULE TYPE:   FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE:   EDTWHL
SOURCE FILE TYPE: .C
HOST:          
SUBSYSTEM:     UI
SUBDIRECTORY:  FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
-------------

SYNOPSIS
CHAR *EDTWHL(FRMPNT,RDONLY)
FIELD *FRMPNT;
BOOL RDONLY;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTS TO FORM WORKING ON
RDONLY - FLAG INDICATING WHETHER OR NOT IN READ ONLY MODE

OUTPUTS:
RETURNS A ERROR CODE IF EXIT ABNORMALLY
RETURNS A NULL IF EXIT NORMALLY

DESCRIPTION
THIS MODULE IS THE MAIN DRIVER MODULE FOR WHOLE EDT MODE

ARGUMENTS:
-------------
FRMPNT =     FIELD *
RDONLY =     BOOL

INCLUDE FILES:

3-100
STDTYP - STANDARD TYPE DEFINITIONS
FPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MEMSET
CPYFRM - COPY FORM
FLWHST - FILL WHOLE STRUCTURE
DRPWHL - DROP WHOLE
ADDFRM
MEMCMP
GDATA
PMSGLS
RMVPAG
GWHINP - GET WHOLE INPUT
PMSGLC
MODFRM - MODIFY FORM
MODWHL - MODIFY WHOLE
INSWHL - INSERT WHOLE
VALINP - VALIDATE INPUT
OISCR
PUTATT
SPRINTF
MEMCPY
STRENC
FLFMST - FIELD TO FORM STRUCTURE TRANSLATION
PDATA

CALLED DIRECTLY BY:

EDTMOD - EDIT MODE

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: ERROR
PURPOSE: ISSUE ERROR MESSAGE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLANERR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

VOID ERROR(S, A, B, C, D, E, F)

DESCRIPTION

PRINTS AN ERROR MESSAGE ON STDERR AND INCREMENTS THE NUMBER OF ERRORS

ARGUMENTS:

S = CHAR *
A = CHAR *
B = CHAR *
C = CHAR *
D = CHAR *
E = CHAR *
F = CHAR *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS

ROUTINES CALLED:

PMSGLS
STRLEN
PRINTF
PS 620144402
1 November 1985

CALLED DIRECTLY BY:

 CHKFLD - CHECK FIELD
 CHKFRM - CHECK FORM
 ADDCHK - ADD POSITION TO CHECK LIST

USED IN MAIN PROGRAM(S):

 CHKFLD - CHECK FIELD
 CHKFRM - CHECK FORM
NAME: EXPAND
PURPOSE: EXPAND AN ARRAY
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *( )
SOURCE FILE: EXPAND
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *EXPAND( FDP, USELST)
FIELD *FDP;
FIELD **USELST;

INPUTS:
FIELD *FDP; ** THE FORM YOU WISH EXPANDED **
FIELD **USELST; ** WHERE TO LOOK FOR EXPANDING SUBFORMS

DESCRIPTION

THIS GUY IS RESPONSIBLE FOR EXPANDING AN ARRAY WHICH WAS PARTIALLY CONSTRUCTED BY FLAN. IT TAKES A POINTER TO THE FORM TO BE EXPANDED AND A POINTER TO THE POINTER TO THE LIST FROM WHICH SUBFORMS MAY BE TAKEN. IF A SUBFORM IS NOT FOUND THE FIELD'S DISPLAY ATTRIBUTE IS SET TO INPUT. THE CASE WHERE BOTH A FIELD AND THE SUBFORM HAVE PROMPTS IS RESOLVED BE CREATING A SPECIAL FIELD TO HOLD THE FIELD'S PROMPTS. USELST MUST BE A POINTER TO A POINTER BECAUSE DELFLD IS USED AND THAT'S WHAT IT NEEDS.

3-104
ARGUMENTS:

---

FDP = FIELD *
USELST = FIELD **

INCLUDE FILES:

---

STDTYP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

---

RSVATT
FNDATT - FIND ATTRIBUTE
ABS
COPFLD
EXPAND/FIXFRM - FIX UP A FORM

CALLED DIRECTLY BY:

---

EXPAND/FIXFRM - FIX UP A FORM
FORMS DRIVEN FORM EDITOR Module Documentation

NAME:        EXPAND/FIXFRM
PURPOSE:     FIX UP A FORM
LANGUAGE:    C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: EXPAND
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM:   UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
FIXES A SUBFORM BY LOCATING IT AND ATTACHING IT IN PLACE AND
EXPANDING IT IF REQUIRED.
NOTE: FIELDS WITH PROMPTS AND SUBFORMS WITH PROMPTS CAUSE A
SPECIAL FIELD TO BE CREATED.

ARGUMENTS:
NDP = FIELD **
USELST = FIELD **

INCLUDE FILES:
STDTYP - STANDARD TYPE DEFINITIONS
FPD  - FORM PROCESSOR DATA
FP CODE  - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:
EXPAND - EXPAND AN ARRAY
DELF LD
RSVATT
FNDATT - FIND ATTRIBUTE
COPFLD
STR CMP

CALLED DIRECTLY BY:
3-106
EXPAND - EXPAND AN ARRAY
NAME: FATAL
PURPOSE: ISSUE FATAL ERROR MESSAGE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLANERR
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
SYNOPSIS:
VOID FATAL(S, A, B, C, D, E, F)

DESCRIPTION
PRINTS A FATAL MESSAGE ON STDERR AND EXITS

ARGUMENTS:
S = CHAR *
A = CHAR *
B = CHAR *
C = CHAR *
D = CHAR *
E = CHAR *
F = CHAR *

INCLUDE FILES:
STDTYP - STANDARD TYPE DEFINITIONS

ROUTINES CALLED:
SPRINTF
STRLEN
PMSGLS
PS 620144402
1 November 1985

CALLED DIRECTLY BY:
-----------------------
MYALLOC - MY MALLOC

USED IN MAIN PROGRAM(S):
------------------------
CHKFLD - CHECK FIELD
CHKFRM - CHECK FORM
CSTASH - CHARACTER STASH
MAKINT - MAKE EXPRESSION INTO AN INTEGER
MAKSTR - MAKE EXPRESSION INTO A STRING
MKPOS - MAKE POSITION NODE
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: FDFE
PURPOSE: FORMS DRIVEN FORM EDITOR
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FDFE
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

LANGUAGE: C

DESCRIPTION:

DESCRIPTION
MAIN INCLUDE FILE FOR FDFE

INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS
FPPARM - FORM PROCESSOR PARAMETERS
FPFCD - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEP - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES
FDFEINI - FDFE INITIALIZATIONS
FPDINI - FPD INITIALIZATION
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

RMVPAG - VIEW A FORM
RENAME
COPY
GETFLS - GET FDL SOURCE FILE

3-110
EDTHMOD - EDIT MODE
UNLINK
FCLOSE
FOPEN
STRLEN
MEMCPY
ACCESS
ADDEXT - ADD EXTENSION TO FILE_NAME
LISTIT - LIST IT
SPRINTF
CHKPRM - CHECK PARAMETER
PRSCMD - PARSE COMMAND
GDATA
OISCR
PMSGLS
PUTATT
MEMCMP
ADDFRM
MEMSET

CALLED DIRECTLY BY:
---------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: FDFE/MAIN
PURPOSE: MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)

LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT()
SOURCE FILE: FDFE
SOURCE FILE TYPE: .C

HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
FDFE()

INPUTS/OUTPUTS:
NONE

INPUTS:

OUTPUTS:

DESCRIPTION
MAIN DRIVER MODULE FOR THE FORMS DRIVEN FORMS EDITOR.
IT ALLOWS
USER TO CHOOSE AMONG SEVERAL FILE MANAGEMENT OPTIONS
AS WELL AS
EDIT MODES.

INCLUDE FILES:

------------------
STDTP - STANDARD TYPE DEFINITIONS
FPPARM - FORM PROCESSOR PARAMETERS
FPDCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA

3-112
FDFEFM - FDGE FORM DEFINITIONS
FDFE - FDGE DATA STRUCTURES
FDFEINI - FDGE INITIALIZATIONS
FPDINI - FPD INITIALIZATION
NTM - NTM INTERFACE INCLUDE FILE

ROUTINES CALLED:

INITAL
MEMCMP
INITFP
FDFE - FORMS DRIVEN FORM EDITOR
termfp
TRMNAT
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: FIFDST
PURPOSE: FILL IN FIELD STRUCTURE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FIFDST
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *FIFDST(FLDPNT,FLDNAM,ROW, COL, WDTH, DPTH, FLDTYP, ATTNAM)
FIELD *FLDPNT:
ENAME FLDNAM:
INT ROW:
INT COL:
INT WDTH:
INT DPTH:
CHAR FLDTYP[1]:
ENAME ATTNAM:

INPUTS/OUTPUTS:

INPUTS:

FLDNAM - NAME OF FIELD CONCERNED
ROW - ROW OF FIELD CONCERNED
COL - COL OF FIELD CONCERNED
WDTH - WIDTH OF FIELD CONCERNED
DPTH - DEPTH OF FIELD CONCERNED
FLDTYP - TYPE OF FIELD CONCERNED
ATTNAM - ATTRIBUTE NAME OF FIELD CONCERNED

OUTPUTS:

NONE - BUT MODIFIES INTERNAL DATA STRUCTURE

DESCRIPTION
THIS MODULE FILL IN THE FILD INFORMATION PASSED IT BY CALLER.

ARGUMENTS:

ARGUMENTS:
FLDPNT = FIELD
FLDNAM = ENAME
ROW = INT
COL = INT
WDTH = INT
DPTH = INT
FLDTYP = CHAR [1]
ATTNAM = ENAME

INCLUDE FILES:

INCLUDE FILES:
STDTYP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA

ROUTINES CALLED:

ROUTINES CALLED:
ESCPY
GATDEF

CALLED DIRECTLY BY:

CALLED DIRECTLY BY:
MODFLD - MODIFY FIELD
MODFRM - MODIFY FORM

USED IN MAIN PROGRAM(S):

USED IN MAIN PROGRAM(S):
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: FLANCI
PURPOSE: FLAN CALLABLE INTERFACE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR (*)
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS
CHAR *FLANCI(FPTR)
FILE *FPTR;

INPUTS:
FPTR - FILE TO BE COMPILED

DESCRIPTION
COMPILES THE SPECIFIED FILE INTO THE LOCAL OPEN LIST.

ARGUMENTS:

FPTR = FILE *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

YYPARSE
DELFLD
CALLED DIRECTLY BY:

---------------------
GETFLS   - GET FDL SOURCE FILE
SAVFLS   - SAVE FDL SOURCE

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MAIN  - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: FLDTYP
PURPOSE: FIELD TYPE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

-------
SYNOPSIS
CHAR *FLDTYP(C)
CHAR C;

DESCRIPTION
RETURNS A STRING OF THE SPECIFIED FIELD TYPE

ARGUMENTS:
-------
C = CHAR

INCLUDE FILES:
-------
STDTYP - STANDARD TYPE DEFINITIONS
STDCIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FFPCODE - FORM PROCESSOR RETURN CODES

called directly by:
-------
CHKFRM - CHECK FORM
ADDCHK - ADD POSITION TO CHECK LIST

used in main program(s):
-------

3-118
PS 6?0144402
1 November 1985

CHKFRM - CHECK FORM

3-119
NAME: FLMFST
PURPOSE: FIELD TO FORM STRUCTURE TRANSLATION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLMFST
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

VOID FLMFST(DP, FP)
FIELD *DP;
STRUCT FRMINFO *FP;

INPUTS:
FIELD *DP; ** INPUT FPD FORM FIELD **
STRUCT FRMINFO *FP; ** OUTPUT DISPLAY FORM **

DESCRIPTION
TRANSLATE FPD FIELD (DP) TO A FRMINFO (FP).

ARGUMENTS:

DP = FIELD *
FP = STRUCT FRMINFO *

INCLUDE FILES:

STDTYPE - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

3-120
SPRINTF
MEMSET

CALLED DIRECTLY BY:
EDTFLD - EDIT FIELD
EDTWHL - EDIT WHOLE
TRNSCR - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: FLSTRC
PURPOSE: FIELD STRUCTURE TRANSLATION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID()
SOURCE FILE: FLSTRC
SOURCE FILE TYPE:.C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
VOID FLSTRC(DP, COM, HELP, VAL, CHKS)
    FIELD *DP;
    STRUCT COMINFO *COM;
    STRUCT ITMHELP *HELP;
    STRUCT ITMVAL *VAL;
    STRUCT ITMCHKS *CHKS;

INPUTS:
FIELD *DP; ** INPUT FIELD TO BE TRANSLATED **
STRUCT COMINFO *COM; ** OUTPUT COMINFO/FLDINFO **
STRUCT ITMHELP *HELP; ** OUTPUT ITMHELP **
STRUCT ITMVAL *VAL; ** OUTPUT ITMVAL **
STRUCT ITMCHKS *CHKS; ** OUTPUT ITMCHKS **

DESCRIPTION
TRANSLATES AN FPD FIELD TO COMINFO/FLDINFO(COM)
STRUCTURE, ITMHELP(HELP)
STRUCTURE, ITMVAL(VAL) STRUCTURE, AND ITMCHKS(CHKS)
STRUCTURE.

ARGUMENTS:

DP = FIELD *
COM = STRUCT COMINFO *
HELP = STRUCT ITMHELP *
VAL = STRUCT ITMVAL *
CHKS = STRUCT ITMCHKS *

3-122
INCLUDE FILES:

- STDYP - STANDARD TYPE DEFINITIONS
- FPD - FORM PROCESSOR DATA
- FDFEFM - FDFE FORM DEFINITIONS
- FDFFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

- SPRINTF
- MEMSET
- MEMCPY
- BLEN
- STRCPY

CALLED DIRECTLY BY:

- DRPWHL - DROP WHOLE
- EDTFLD - EDIT FIELD
- FLWHST - FILL WHOLE STRUCTURE
- TRNSCR - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: FLWHST
PURPOSE: FILL WHOLE STRUCTURE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLWHST
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

---

SYNOPSIS
VOID FLWHST(FRMPNT)
FIELD *FRMPNT;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM FROM WHICH INFORMATION WILL RETRIEVED AND AND PUT IN SCREEN DATA AREA.

OUTPUTS:
NONE - BUT DOES FILL EXTERNAL SCREEN DATA AREA AND EXTERNAL ARAY OF POINTERS.

DESCRIPTION
THIS MODULE CALLS FLSTRC FOR EACH FIELD IN FORM AND STORES POINTER TO FIELD IN PNTARY WHICH IS A GLOBAL ARRAY USEDED TO ASSOCIATE EACH FIELD ON SCREEN WITH ITS INTERNAL STRUCTURE.

ARGUMENTS:

--------
FRMPNT - FIELD *
INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FDPEFM - FDFE FORM DEFINITIONS
FDPE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MEMSET
FLSTRC - FIELD STRUCTURE TRANSLATION

CALLED DIRECTLY BY:

EDTWHL - EDIT WHOLE

USED IN MAIN PROGRAM(S):

FDPE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: FNDATT
PURPOSE: FIND ATTRIBUTE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: ATTMAP * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS

ATTMAP *FNDATT(S)
CHAR *S;

DESCRIPTION

RETURNS A POINTER TO THE SPECIFIED ATTRIBUTE IN THE ATTRIBUTE MAP

ARGUMENTS:

S = CHAR *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

STRCMP

CALLED DIRECTLY BY:

3-126
EXPAND/FIXFRM - FIX UP A FORM
EXPAND - EXPAND AN ARRAY
VALINP/CCKFRM - CHECK FORM
VALINP/CCKFLD - CHECK FIELD
CHKFLD - CHECK FIELD
CHKFRM - CHECK FORM

USED IN MAIN PROGRAM(S):
CHKFLD - CHECK FIELD
CHKFRM - CHECK FORM
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: FREBUF
PURPOSE: FREE BUFFER
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FREBUF
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

VOID FREBUF(BUFPNT)
    FIELD **BUFPNT;

INPUTS/OUTPUTS:

INPUTS:
    BUFPNT - ADDRESS OF BUFFER TO BE DELETED

OUTPUTS:
    NONE

DESCRIPTION
    THIS MODULE CALLS DELFLD FOR ALL FIELDS IN A BUFFER(ALTBUF
    OR WRKBUF).

ARGUMENTS:

    BUFPNT - FIELD **

INCLUDE FILES:

    STDTYPE - STANDARD TYPE DEFINITIONS
    FPD - FORM PROCESSOR DATA
FDFEFM        - FDGE FORM DEFINITIONS
FDFE        - FDGE DATA STRUCTURES

ROUTINES CALLED:
---------------
DEFLFD

CALLED DIRECTLY BY:
-------------------
EDTMOH        - EDIT MODE
GETFLS        - GET FDL SOURCE FILE

USED IN MAIN PROGRAM(S):
-----------------------
FDGE/MAIN     - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDGE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: GETFLS
PURPOSE: GET FDL SOURCE FILE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: GETFLS
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *GETFLS(FRMNAM, PDP)

CHAR FRMNAM[];
FIELD **PDP;

INPUTS:
CHAR FRMNAM[]; ** THE NAME OF THE FILE CONTAINING THE
FDL SOURCE
THERE IS NO FILE EXTENSION **

OUTPUTS:
FIELD **PDP; ** A POINTER WHICH IS TO BE SET TO THE
'OPENED' FORMS

DESCRIPTION
GIVEN A FORM NAME OPENS THE FDL SOURCE FILE AND CALLS
FLANCI TO CREATE
AN FPD STRUCTURE.

ARGUMENTS:

-------------
FRMNAM - CHAR []
PDP - FIELD **

INCLUDE FILES:

-------------
STDTYP - STANDARD TYPE DEFINITIONS

3-130
STDIO - ***** PURPOSE NOT FOUND BY STRIPPER *****
FPD - FORM PROCESSOR DATA
FP CODE - FORM PROCESSOR RETURN CODES
FDDEFM - FDGE FORM DEFINITIONS
FDGE - FDGE DATA STRUCTURES

ROUTINES CALLED:
---------------------
FREBUF - FREE BUFFER
ESCPY
SPRINTF
STRRCHR
STRCAT
FOPEN
SYSMSG
BLEN
MEMSET
MEMCPY
FLANCI - FLAN CALLABLE INTERFACE
FCLOSE
GETFLS/TREEXP - TREE EXPRESSION:
MAX
STREN
MALLOC

CALLED DIRECTLY BY:
---------------------
CPYFRM - COPY FORM
FDGE - FORMS DRIVEN FORM EDITOR

USED IN MAIN PROGRAM(S):
--------------------------
FDGE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDGE)
NAME: GETFLS/TREEXP
PURPOSE: TREE EXPRESSION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: GETFLS
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

-------------
STATIC VOID TREEXP(STR, EP)
    CHAR STR[];
    ENODE *EP;

creates an algebraic expression given its syntactic tree representation.
this procedure is also used by GETFLS when fetching source forms.

ARGUMENTS:

-------------
STR = CHAR []
EP = ENODE *

INCLUDE FILES:

-------------
STDYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

-------------
PREC - PRECEDENCE
PRINTF

3-132
GETFLS/TREEEXP - TREE EXPRESSION
STRCAT

CALLED DIRECTLY BY:
----------------------
GETFLS/TREEEXP - TREE EXPRESSION
GETFLS     - GET FDL SOURCE FILE

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MAIN     - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: GETLEN
PURPOSE: GET LENGTH
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: GETLEN
SOURCE FILE TYPE: .C
HOST: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

    INT GETLEN(BUF,BUFLEN)
    CHAR       BUF[];
    REGISTER INT BUFLEN;

INPUTS/OUTPUTS:

INPUTS:
    BUF   - NON NULL TERMINATED STRING THE 'NON BLANK'
            LENGTH OF WHICH
            IS RETURNED
    BUFLEN - ACTUAL LENGTH OF BUFFER

OUTPUTS:
    LENGTH (NON BLANK) OF STRING IS RETURNED

DESCRIPTION

    THIS MODULE RETURNS THE "NON BLANK" LENGTH OF A NON
    NULL TERMINATED
    STRING.

ARGUMENTS:

    BUF   =       CHAR []
    BUFLEN =      INT
INCLUDE FILES:

STDTPY - STANDARD TYPE DEFINITIONS

CALLED DIRECTLY BY:

GITMD - GET ITEM DATA AND INSERT IN STRUCTURE
GTFDXT/GTXINF - GET TEXT INFORMATION
MODFRM - MODIFY FORM

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME:   GFDINP
PURPOSE: GET FIELD INPUT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: GFDINP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
CHAR *GFDINP(PFKEY, FRMPNT, FLDPNT)
FIELD   *FRMPNT;
FIELD   *FLDPNT;
INT     *PFKEY;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - FORM WORKING ON
FLDPNT - FIELD WORKING ON

OUTPUTS:
PFKEY - RETURN TO CALLER PFKE RECEIVED FROM OISCR
RETRUNS NULL IF NOMALLY TERMINATED AND AN ERROR CODE IF
ABNORMALLY TERMINATED

DESCRIPTION
THIS MODULE GETS INPUT UPPERCASES FIELDS WHICH MUST BE
UPPER CASE
AND CALLS VALINP IF PFKEY IS ENTER AND IF TASK = INSERT
OR MODIFY.

ARGUMENTS:

PFKEY =     INT *
FRMPNT = FIELD *
FLDPNT = FIELD *

INCLUDE FILES:
---------------------
STDTYP       - STANDARD TYPE DEFINITIONS
FPCODE       - FORM PROCESSOR RETURN CODES
FPFARM       - FORM PROCESSOR PARAMETERS
FPD          - FORM PROCESSOR DATA
FDFEFM       - FDFE FORM DEFINITIONS
FDFE         - FDFE DATA STRUCTURES

ROUTINES CALLED:
---------------------
PDATA
MEMCMP
VALINP       - VALIDATE INPUT
GDATA
OISCR
SPRINTF

CALLED DIRECTLY BY:
---------------------
EDTFLD       - EDIT FIELD

USED IN MAIN PROGRAM(S):
---------------------
FDFE/MAIN     - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: GFLDPT
PURPOSE: GET FIELD POINTER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: FIELD *( )
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS
FIELD *GFLDPT(FLDPTR, S)
FIELD *FLD PTR;
CHAR *S;

DESCRIPTION
RETURN A POINTER TO THE NAMED FIELD ON THE SPECIFIED FORM.

ARGUMENTS:

FLD PTR = FIELD *
S = CHAR *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
STDOUT - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FP CODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

STRCMP

CALLED DIRECTLY BY:

3-138
PS 620144402
1 November 1985

CHKFRM - CHECK FORM

USED IN MAIN PROGRAM(S):

CHKFRM - CHECK FORM
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: GITMD
PURPOSE: GET ITEM DATA AND INSERT IN STRUCTURE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: GITMD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *GITMD (FLDPNT,ITMHLP,ITMVAL,ITMCHKS)
FIELD *FLDPNT;
CHAR *ITMHLP;
CHAR *ITMVAL;
STRUCT ITMCHKS *ITMCHKS;

INPUTS/OUTPUTS:

INPUTS:
FLDPNT - FIELD IN WHICH PUTTING INFORMATION RECEIVED FROM USER
ITMHLP - HELP LINE INPUT BY USER
ITMVAL - ITEM DEFAULT VALUE INPUT BY USER
ITMCHKS - CHECKING DESIRED BY USER

OUTPUTS:
RETURNS ERROR CODE IF ABNORMAL TERMINATION
RETURNS NULL IF NORMAL TERMINATION

DESCRIPTION
THIS MODULE INSERTS DESIRED CHANGES INTO ITEM DATA STRUCTURE OF INTERNAL DATA STRUCTURE

ARGUMENTS:
**FLDPNT =** FIELD *
**ITMHELP =** CHAR *
**ITMVAL =** CHAR *
**ITMCHKS =** STRUCT ITMCHKS *

**INCLUDE FILES:**

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STDIO</td>
<td>&quot;**** PURPOSE NOT FOUND BY STRIPPER ****&quot;</td>
</tr>
<tr>
<td>STDTyp</td>
<td>STANDARD TYPE DEFINITIONS</td>
</tr>
<tr>
<td>CTYPe</td>
<td>&quot;**** PURPOSE NOT FOUND BY STRIPPER ****&quot;</td>
</tr>
<tr>
<td>FPPARM</td>
<td>FORM PROCESSOR PARAMETERS</td>
</tr>
<tr>
<td>FPcode</td>
<td>FORM PROCESSOR RETURN CODES</td>
</tr>
<tr>
<td>FPD</td>
<td>FORM PROCESSOR DATA</td>
</tr>
<tr>
<td>FDFEFM</td>
<td>FDFE FORM DEFINITIONS</td>
</tr>
<tr>
<td>FDFE</td>
<td>FDFE DATA STRUCTURES</td>
</tr>
</tbody>
</table>

**Routines Called:**

- FREE
- MAX
- GETLEN - GET LENGTH
- MALLOC
- SYMSMG
- ESCPY
- MEMDGT
- MATOI
- BLEN
- MEMSET
- MEMCPY
- MIN
- MEMCMP

**Called Directly By:**

- INSFLD - INSERT FIELD
- MODFLD - MODIFY FIELD

**Used in Main Program(s):**

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: GNXTFD
PURPOSE: GET NEXT FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: FIELD *( )
SOURCE FILE: GNXTFD
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE
DESCRIPTION:

SYNOPSIS
FIELD *GNXTFD(FRMPNT, FLDPNT, PFKEY)
FIELD *FRMPNT;
FIELD *FLDPNT;
INT PFKEY;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM WORKING ON
FLDPNT - CURRENT FIELD POINTER
PFKEY - PFKEY PASSED BY CALLER FROM OISCR

OUTPUTS:
RETURNS - POINTER TO NEXT FIELD IF FOUND
RETURNS - NULL IF NO FIELD FOUND

DESCRIPTION
THIS MODULE GETS THE NEXT FIELD TO BE VIEWED OR WORKED ON USING USER INPUT

ARGUMENTS:
-------------------
FRMPNT = FIELD *
FLDPNT = FIELD *
PFKEY = INT

INCLUDE FILES:

STDYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MEMCMP
GTNMFD - GET NAMED FIELD
GNXTFD/NXTFLD - NEXT FIELD

CALLED DIRECTLY BY:

EDTFLD - EDIT FIELD

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: GNXTFD/NXTFLD
PURPOSE: NEXT FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: FIELD *( )
SOURCE FILE: GNXTFD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

STATIC FIELD *NXTFLD(FLDPNT, TYP, DRCTN)
    FIELD *FLDPNT;
    CHAR TYP;
    CHAR DRCTN;

INPUTS/OUTPUTS:

INPUTS:
    FLPNT - CURRENT FIELD POINTER
    TYP - TYPE OF FIELD INTERESTED IN
    DRCTN - DIRECTION WANT TO SEARCH

OUTPUTS:
    RETURNS - POINTER TO NEXT FIELD IF FOUND
    RETURNS - NULL IF NO FIELD FOUND

DESCRIPTION

THIS MODULE GETS THE NEXT FIELD OF A SPECIFIED TYPE
SEARCHING IN A
SPECIFIED DIRECTION TO BE VIEWED OR WORKED ON.

ARGUMENTS:

FLDPNT = FIELD *
TYP = CHAR
DRCTN = CHAR

3-144
INCLUDE FILES:

- STDTYP - STANDARD TYPE DEFINITIONS
- FPCODE - FORM PROCESSOR RETURN CODES
- FPD - FORM PROCESSOR DATA
- FDFFEFM - FDFE FORM DEFINITIONS
- FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

- GNXTFD/NXTFLD - NEXT FIELD

CALLED DIRECTLY BY:

- GNXTFD/NXTFLD - NEXT FIELD
- GNXTFD - GET NEXT FIELD

USED IN MAIN PROGRAM(S):

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: GTCPFD
PURPOSE: GET USING CURSOR POSITION FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: FIELD *
SOURCE FILE: GTCPFD
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
FIELD *GTCPFD(FRMPNT,ROW,COL)
FIELD *FRMPNT;
INT ROW;
INT COL;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM WORKING ON
ROW - CURSOR POSITION OF DESIRED FIELD
COL

OUTPUTS:
RETURNS A POINTER TO FIELD AT INDICATED LOCATION IF ONE
RETURNS A NULL IF NO FIELD FOUND

DESCRIPTION
THIS MODULE RETURNS A POINTER TO FIELD AT ROW AND COL
GIVEN BY CALLER

ARGUMENTS:

FRMPNT = FIELD *
ROW = INT
COL = INT
INCLUDE FILES:

-------------------
STDTYP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA

CALLED DIRECTLY BY:

-------------------
EDTFLD - EDIT FIELD
SCRMAN/CHGPOS - CHANG POSITION
TRNSCR - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

-------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: GTFDTX
PURPOSE: GET FIELD TEXT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *()
SOURCE FILE: GTFDTX
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
CHAR *GTFDTX(FLDPNT,COMINFO)
FIELD *FLDPNT;
STRUCT COMINFO *COMINFO;

INPUTS/OUTPUTS:

INPUTS:
FLDPNT - POINTER TO FIELD WORKING ON
COMINFO - POINTER TO STRUCTURE CONTAINING USER INPUT

OUTPUTS:
RETURNS NULL IF TERMINATED NORMALLY
RETURNS ERROR CODE IF TERMINATED ABNORMALLY

DESCRIPTION
THIS MODULE GETS TEXT FIELDS AND INSERTS THEM INTO INTERNAL STRUCTURE
OF FIELD CONCERNED.

ARGUMENTS:

-------------
FLDPNT = FIELD *
COMINFO = STRUCT COMINFO *

INCLUDE FILES:
-------------

3-148
STDTYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:
---------------
FREE
GTFDXT/GTXINF - GET TEXT INFORMATION
MALLOC
SYSMSG
ESCPY

CALLED DIRECTLY BY:
---------------------
INSFLD - INSERT FIELD
MODFLD - MODIFY FIELD

USED IN MAIN PROGRAM(S):
-------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: GTFDTX/GTXINF
PURPOSE: GET TEXT INFORMATION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: GTFDTX
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

-------------
SYNOPSIS

STATIC VOID GTXINF(COMINFO, PROW, PCOL, PLNGTH)
    STRUCT COMINFO *COMINFO;
    INT *PROW, *PCOL, *PLNGTH;

INPUTS/OUTPUTS:

INPUTS:
    COMINFO - POINTER TO STRUCTURE CONTAINING USER INPUT
OUTPUTS:
    PROW - PROMPT (OR TEXT) ROW
    PCOL - PROMPT (OR TEXT) COL
    PLNGTH - PROMPT (OR TEXT) LENGTH

DESCRIPTION

THIS MODULE GETS TEXT FIELD INFORMATION AND PASSES IT BACK TO CALLER

ARGUMENTS:

--------------
COMINFO = STRUCT COMINFO *
PROW = INT *
PCOL = INT *
PLNGTH = INT *

INCLUDE FILES:

3-150
STDTYP  - STANDARD TYPE DEFINITIONS
FPCODE  - FORM PROCESSOR RETURN CODES
FPD     - FORM PROCESSOR DATA
FDFEFM  - FDFE FORM DEFINITIONS
FDFE    - FDFE DATA STRUCTURES

ROUTINES CALLED:
-----------------
GETLEN   - GET LENGTH
MATOI

CALLED DIRECTLY BY:
-------------------
GTFDTX   - GET FIELD TEXT

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MAP   - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: GTNMFD
PURPOSE: GET NAMED FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: FIELD *( )
SOURCE FILE: GTNMFD
SOURCE FILE TYPE: .C
HOST: UI
SUBSYSTEM: FDFE
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
FIELD *GTNMFD(FLDPNT,FLDNAM)
   FIELD *FLDPNT;
   ENAME FLDNAM;

INPUTS/OUTPUTS:

INPUTS:
   FLDPNT - POINTER TO FIRST FIELD IN LIST
   FLDNAM - NON NULL TERMINATED STRING WITH FIELD NAME WANTED

OUTPUTS:
   RETURNS A POINTER TO FIELD WITH NAME GIVEN IF FOUND
   RETURNS NULL IF NO FIELD FOUND WITH NAME GIVEN

DESCRIPTION
   THIS MODULE RETURNS A POINTER TO FIELD STRUCTURE WITH
   GIVEN NAME
   OR A NULL IF COULD NOT FIND SUCH A FIELD.

ARGUMENTS:

   FLDPNT = FIELD *
   FLDNAM = ENAME
INCLUDE FILES:

- STDTYPE - STANDARD TYPE DEFINITIONS
- FPD - FORM PROCESSOR DATA
- FDFEFM - FDFE FORM DEFINITIONS
- FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

- ESCPY
- STRLEN
- STRCMP

CALLED DIRECTLY BY:

- CPYFRM - COPY FORM
- DRPFRM - DROP FORM
- EDTMOD - EDIT MODE
- GNXTFD - GET NEXT FIELD
- INSFRM - INSERT FORM

USED IN MAIN PROGRAM(S):

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: GWHINP
PURPOSE: GET WHOLE INPUT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *( )
SOURCE FILE: GWHINP
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

```
CHAR *GWHINP(FRMPNT,PFKEY)
FIELD *FRMPNT;
INT *PFKEY;
```

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM BEING WORKED ON

OUTPUTS:
PFKEY - PFKEY PASSED BACK TO CALLER FROM OISCR
RETURNS NULL IF NORMAL TERMINATION
RETURNS ERROR CODE IF ABNORMAL TERMINATION

DESCRIPTION

this module gets all input for all screens calling VALINP
and FLWHST for all screens.

ARGUMENTS:

```
PFKEY = INT *
FRMPNT = FIELD *
```

INCLUDE FILES:

3-154
STDTYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

Routines Called:

- ADDFRM
- MEMCMP
- MEMCPY
- RMVPAG
- VALIMP - VALIDATE INPUT
- PUTCUR
- OISCR
- SPRINTF
- GDATA
- PUTATT
- PDATA

Called Directly By:

- EDTWHL - EDIT WHOLE

Used in Main Program(s):

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: INSFLD
PURPOSE: INSERT FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *( )
SOURCE FILE: INSFLD
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *INSFLD(FLDPNT,NXTFLD,PRVFLD,PARPNT,COMINFO,ITMHLP,ITMVAL,ITMCHK S,LEVEL)

FIELD **FLDPNT;
FIELD **NXTFLD;
FIELD **PRVFLD;
FIELD *PARPNT;
STRUCT COMINFO *COMINFO;
CHAR *ITMHLP;
CHAR *ITMVAL;
STRUCT ITMCHKS *ITMCHKS;
INT LEVEL;

INPUTS/OUTPUTS:

INPUTS:
FLDPNT - ADDRESS OF POINTER TO FIELD BEING INSERTED
PRVFLD - ADDRESS WHERE PREVIOUS FIELD POINTER WILL BE INSERTED
NXTFLD - ADDRESS WHERE NEXT FIELD POINTER WILL BE INSERTED
PARPNT - POINTER TO PARENT OF FIELD
COMINFO - POINTER TO STRUCTURE CONTAINING INPUT FOR GENERAL FIELD
INFORMATION OBTAINED FROM USER
ITMHLPL - POINTER TO CHAR STRING CONTAINING HELP LINE
ITMVAL - POINTER TO CHAR STRING CONTAINING DEFAULT VALUE FOR ITEM
ITMCHKS - POINTER TO STRUCTURE CONTAINING USER INPUT FOR ITM CHECKS
LEVEL - LEVEL OF OF RECURSION

OUTPUTS:
RETURNS NULL IF NORMAL TERMINATION
RETURNS ERROR CODE IF ABNORMAL TERMINATION

DESCRIPTION
THIS MODULE INSERTS FIELD INTO DATA STRUCTURE - CALLS MAKFLD TO
MAKE THE ACTUAL INSERTION THEN INSERTING PARTICULAR INFORMATION
FOR FIELD(WINDOW.ITEM.FORM.OR ARRAY)

ARGUMENTS:
--------
FLDPNT - FIELD **
NXTFLD - FIELD **
PRVFLD - FIELD **
PARPNT - FIELD *
COMINFO - STRUCT COMINFO *
ITMHLPL - CHAR *
ITMVAL - CHAR *
ITMCHKS - STRUCT ITMCHKS *
LEVEL - INT

INCLUDE FILES:
---------
STDYP - STANDARD TYPE DEFINITIONS
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFFEM - FDFFE FORM DEFINITIONS
FDFE - FDFFE DATA STRUCTURES

ROUTINES CALLED
----------
MATOII
INSFLD - INSERT FIELD
PMSGLS
ESCPY

3-157
GATDEF
MAKFLD
MAX
GTFDTX   - GET FIELD TEXT
GITMD    - GET ITEM DATA AND INSERT IN STRUCTURE

CALLED DIRECTLY BY:

- EDTFLD   - EDIT FIELD
- INSFLD   - INSERT FIELD
- INSWHL   - INSERT WHOLE
- MODFLD   - MODIFY FIELD
- TRNSCR   - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: INSFRM
PURPOSE: INSERT FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: INSFRM
SOURCE FILE TYPE: .C
HOST: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *INSFRM(FRMNAM)
ENAME FRMNAM;

INPUTS/OUTPUTS:

INPUTS:
FRMNAM - NON NULL TERMINATED STRING CONTAINING NAME OF FRM TO BE INSERTED.

OUTPUTS:
RETURNS A NULL IF SUCCESSFUL
RETURNS AN ERROR CODE IF FORM ALREADY EXIST OR ABNORMALLY TERMINATED.

DESCRIPTION
THIS MODULE INSERTS TOP LEVEL FORM FILLING IN DEFAULT VALUES

ARGUMENTS:

FRMNAM = ENAME

INCLUDE FILES:
STDTYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

GTWMFD - GET NAMED FIELD
ESCPY
GATDEF
MAKFLD

CALLED DIRECTLY BY:

EDTMOD - EDIT MODE

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: INSWHL
PURPOSE: INSERT WHOLE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: INSWHL
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *INSWHL(FRMPNT)
    FIELD *FRMPNT;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM WORKING ON

OUTPUTS:
RETURNS NULL IF NORMAL TERMINATION
RETURNS ERROR CODE IF ABNORMALLY TERMINATED

DESCRIPTION
THIS MODUL INSERTS ALL FIELDS USER ENTERED ON WHOLE EDIT.
IT USES THE GLOBAL DATA AREA.

ARGUMENTS:

-------------
FRMPNT = FIELD *

INCLUDE FILES:

-------------
STDTYP - STANDARD TYPE DEFINITIONS
FPCODE  -  FORM PROCESSOR RETURN CODES
FPD   -  FORM PROCESSOR DATA
FDFEFM  -  FDFE FORM DEFINITIONS
FDSE  -  FDFE DATA STRUCTURES

ROUTINES CALLED:
-------------------
MEMCMP
SPRINTF
PMSGLS
INSFLD  -  INSERT FIELD

CALLED DIRECTLY BY:
-------------------
EDTWHL  -  EDIT WHOLE

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MAIN  -  MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: LAYOUT
PURPOSE: LAYOUT MODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: LAYOUT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
------------

SYNOPSIS
CHAR *LAYOUT(FRMPNT,RDONLY)
FIELD *FRMPNT;
BOOL RDONLY;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM BEING WORKED ON
RDONLY - FALG INDICATING WHETHER IN READ ONLY MODE OR NOT

OUTPUTS:
RETURNS NULL IF NORMAL TERMINATION
RETURNS ERROR CODE IF ABNORMAL TERMINATION

DESCRIPTION
THIS MODULE IS THE MAIN SWITCHER FOR LAYOUT MODE
BETWEEN ACTUAL
LAYOUT

ARGUMENTS:
-----------
FRMPNT = FIELD *
RDONLY = BOOL

3-163
INCLUDE FILES:

- STDTYP - STANDARD TYPE DEFINITIONS
- FPPARM - FORM PROCESSOR PARAMETERS
- FPCODE - FORM PROCESSOR RETURN CODES
- FPD - FORM PROCESSOR DATA
- FDFEFM - FDFE FORM DEFINITIONS
- FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

- SCRMAN - SCREEN MANAGER
- MEMCMP
- EDTFLD - EDIT FIELD

CALLED DIRECTLY BY:

- EDTMOD - EDIT MODE

USED IN MAIN PROGRAM(S):

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: LISTFM
PURPOSE: LIST FORMS
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: LISTFM
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
CHAR *LISTFM()

INPUTS/OUTPUTS:

INPUTS:
NONE

OUTPUTS:
RETURNS NULL IF NORMAL TERMINATION
RETURNS ERROR CODE IF ABNORMAL TERMINATION

DESCRIPTION
THIS MODULE LIST ALL FORMS IN CURRENT FLS FILE ON SCREEN

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
FPARM - FORM PROCESSOR PARAMETERS
FP CODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES
ROUTINES CALLED:
----------------
MEMSET
MEMCPY
ADDFRM
MEMCMP
RMVPAG
PMSGLS
OISCR
PDATA
STRLEN

CALLED DIRECTLY BY:
-------------------
EDTMod       - EDIT MODE

USED IN MAIN PROGRAM(S):
------------------------
FDFe/Main      - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFe)
NAME: LISTIT
PURPOSE: LIST IT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *( )
SOURCE FILE: LISTIT
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
-----------------

SYNOPSIS
CHAR *LISTIT(TYPE)
BOOL TYPE;

INPUTS/OUTPUTS:

INPUTS:
CHAR *TYPE POINTER TO THE CHARACTER STRING "FDL"
OR "FD"

OUTPUTS:
RETURNS SYSERR IF THE SYSTEM FUNCTION FAILS
OTHERWISE RETURNS A NULL POINTER.

DESCRIPTION
USES THE SYSTEM FUNCTION TO DO A DIRECTORY COMMAND FOR
*.FDL OR *.FD
FILES AND REDIRECTS THE OUTPUT INTO A TEMPORARY FILE.
THEN CALLS PRCFIL
TO READ THE FILE AND FILL IN THE FILENAMES ON THE
SCREEN.

ARGUMENTS:
-----------

3-167
TYPE = BOOL

INCLUDE FILES:

--------
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTP - STANDARD TYPE DEFINITIONS
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFFM - FDFE FORM DEFINITIONS
FDFF - FDFE DATA STRUCTURES

ROUTINES CALLED:

----------------
MKTEMP
STRCPY
STRCAT
SPRINTF
SYSMSG
PRCFIL - PROCESS TEMPORARY FILE
PDATA
ADDFRM
MENCMP
FEOF
FCLOSE
UNLINK
RMVPAG
PMSGLS
OISCR
FOPEN
SYSTEM

CALLED DIRECTLY BY:

---------------------
FDFF - FORMS DRIVEN FORM EDITOR

USED IN MAIN PROGRAM(S):

---------------------
FDFF/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFF)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: MAKINT
PURPOSE: MAKE EXPRESSION INTO AN INTEGER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: ENODE * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST: UI
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS
ENODE *MAKINT(EP)
ENODE *EP:

DESCRIPTION
CONVERT THE SPECIFIED EXPRESSION TO INTEGER AND RETURN
POINTER TO NEW
EXPRESSION.

ARGUMENTS:

EP = ENODE *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

MYALLOC - MY MALLOC
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: MAKSTR
PURPOSE: MAKE EXPRESSION INTO A STRING
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: ENODE * ( )
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

---------------
SYNOPSIS
ENODE *MAKSTR(EP)
ENODE *EP:

DESCRIPTION
CONVERT THE SPECIFIED EXPRESSION TO STRING AND RETURN
POINTER TO NEW
EXPRESSION.

ARGUMENTS:
---------------
EP = ENODE *

INCLUDE FILES:
---------------
STDTYPE  - STANDARD TYPE DEFINITIONS
STDIO    - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD      - FORM PROCESSOR DATA
RW       - REPORT WRITER DEFINITIONS
FPCODE   - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:
---------------
MYALLOC   - MY MALLOC
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: MKPOS
PURPOSE: MAKE POSITION NODE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: POS * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS
POS *MKPOS(HPOS, HMIN, HLOC, HREF, VPOS, VMIN, VLOC, VREF)
INT HPOS, HMIN, HLOC;
CHAR *HREF;
INT VPOS, VMIN, VLOC;
CHAR *VREF;

DESCRIPTION
CREATES THE SPECIFIED POSITION NODE AND ADDS IT TO THE LIST. HPOS AND
VPOS ARE THE REFERENCE POINTS ON THE CURRENT FIELD. HMIN
AND VMIN ARE THE LOCATION RELATIVE TO THE REFERENCE FIELD. HLOC AND VLOC
ARE THE REFERENCE POINTS ON THE REFERENCE FIELD. AND HREF AND VREF ARE THE
REFERENCE FIELDS.

ARGUMENTS:

HPOS - INT
HMIN - INT
HLOC - INT
HREF - CHAR *
VPOS - INT
VMIN - INT
VLOC - INT
VREF - CHAR *

3-171
INCLUDE FILES:

- STDTYP - STANDARD TYPE DEFINITIONS
- STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
- FPD - FORM PROCESSOR DATA
- RW - REPORT WRITER DEFINITIONS
- FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

- MYALLOC - MY MALLOC
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: MODFLD
PURPOSE: MODIFY FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: MODFLD
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *MODFLD(PARPNT,FLDPNT,COMINFO,ITMHLP,ITMVAL,ITMCHKS)
FIELD "PARPNT;
FIELD "FLDPNT;
STRUCT COMINFO "COMINFO;
CHAR "ITMHLP;
CHAR "ITMVAL;
STRUCT ITMCHKS "ITMCHKS.

INPUTS/OUTPUTS:

INPUTS:

PARPNT - POINTER TO PARENT FIELD
FLDPNT - ADDRESS OF POINTER TO FIELD TO BE MODIFIED
COMINFO - POINTER TO STRUCTURE CONTAINING USER INPUT
FOR GENERAL FIELD INFORMATION
ITMHLP - POINTER TO CHAR STRING CONTAINING HELP LINE INPUT BY USER
ITMVAL - POINTER TO DEFAULT VALUE FOR ITEMS INPUT BY USER
ITMCHKS - POINTER TO STRUCTURE CONTAINING ITEM VALIDATION CHECKS INPUT BY USER

OUTPUTS: 3 173
RETURNS NULL IF NORMAL TERMINATION
RETURNS ERROR CODE IF ABNORMAL TERMINATION

DESCRIPTION
THIS MODULE MODIFIES EXISTING FIELD IN ACCORDENCE WITH USER INPUT

ARGUMENTS:

------------
PARPNT =  FIELD 
FLDPNT =  FIELD 
COMINFO =  STRUCT COMINFO 
ITMHLP =  CHAR 
ITMVAL =  CHAR 
ITMCHKS =  STRUCT ITMCHKS

INCLUDE FILES:

-------------
STDTYP  -  STANDARD TYPE DEFINITIONS
FPCODE  -  FORM PROCESSOR RETURN CODES
FPD     -  FORM PROCESSOR DATA
FDFEFM  -  FDFE FORM DEFINITIONS
FDFE    -  FDFE DATA STRUCTURES

ROUTINES CALLED:

------------
PMSGLS
DELFLD
INSFLD  -  INSERT FIELD
FREE
MATO1
FIFDST  -  FILL IN FIELD STRUCTURE
GTFDFTX -  GET FIELD TEXT
GITMD   -  GET ITEM DATA AND INSERT IN STRUCTURE
MEMSET

CALLED DIRECTLY BY

-----------
EDLFLD  -  EDIT FIELD
MODWHL  -  MODIFY WHOLE
TRNSCR  -  TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S)
PS 620144402
1 November 1985

FDFA/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
INTEGRATED INFORMATION SUPPORT SYSTEM (I ISS) VOLUME B
USER INTERFACE SUBS. (U) GENERAL ELECTRIC CO
SCHEMECTADY NY PRODUCTION RESOURCES CONS.
UNCLASSIFIED C MORENC ET AL. 01 NOV 85 PS-620144482  F/G 11/5 NL
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: MODFRM
PURPOSE: MODIFY FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: MODFRM
SOURCE FILE TYPE: .C
HOST: UI
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
CHAR *MODFRM(FRMPNT,FRMINFO)
FIELD *FRMPNT;
STRUCT FRMINFO *FRMINFO;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM'S INTERNAL STRUCTURE
FRMINFO - ADDRESS OF USER INPUT

OUTPUTS:
RETURNS A NULL IF SUCCESS AND POINTER TO ERROR CODE STRING IF ERROR

DESCRIPTION
THIS MODULE MODIFIES INTERNAL STRUCTURE FOR FORM POINTED TO BY FRMPNT USING USER INPUT POINTED TO BY FRMINFO.

ARGUMENTS:

FRMPNT = FIELD *
FRMINFO = STRUCT FRMINFO *

INCLUDE FILES:

3-176
STDTYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:
---------------------
MATOI
FIFDST - FILL IN FIELD STRUCTURE
MODFRM/FRETXT - FREE TEXT
SYSMSG
ESCPY
GETLEN - GET LENGTH
MALLOC

CALLED DIRECTLY BY:
---------------------
EDTFLD - EDIT FIELD
EDTWHL - EDIT WHOLE
TRNSCR - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: MODFRM/FRETXT
PURPOSE: FREE TEXT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: MODFRM
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

STATIC VOID FRETXT(FLDPNT)
FIELD *FLDPNT;

INPUTS/OUTPUTS:

INPUTS:
FLDPNT - POINTER TO FIELD'S INTERNAL STRUCTURE

OUTPUTS:
NONE

DESCRIPTION

THIS MODULE FREES UP TEXT BUFFER'S FOR THE FIELD
POINTED TO BY FLDPNT.

ARGUMENTS:

FLDPNT - FIELD *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

3-178
ROUTINES CALLED:
-----------------
FREE

CALLED DIRECTLY BY:
---------------------
MODFRM    - MODIFY FORM

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MAIN    - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: MODWHL
PURPOSE: MODIFY WHOLE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: MODWHL
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *MODWHL(FRMPNT)
FIELD *FRMPNT:

INPUTS/OUTPUTS:

INPUTS:
FRMPNT – POINTER TO FORM BEING WORKED ON

OUTPUTS:
RETURNS A NULL IF NORMAL TERMINATION
RETURNS A ERROR CODE IF ABNORMAL TERMINATION

DESCRIPTION
THIS MODULE MODIFIES ALL FIELDS ON A FORM BY CALLING MODFLD FOR EACH
FIELD ON FORM

ARGUMENTS:

FRMPNT – FIELD *

INCLUDE FILES:

STDTYP – STANDARD TYPE DEFINITIONS

3-180
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDPEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:
---------------------
MODFLD - MODIFY FIELD
MEMCMP
SPRINTF
PMSGLS

CALLED DIRECTLY BY:
---------------------
EDTWHL - EDIT WHOLE

USED IN MAIN PROGRAM(S):
---------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: MYALLOC
PURPOSE: MY MALLOC
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: FLANSP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FE
DOCUMENTATION GROUP: FDFE/FLAN

DESCRIPTION:

SYNOPSIS
CHAR *MYALLOC(SIZE)
UNSIGNED SIZE;

DESCRIPTION
ALLOCATE THE SPECIFIED MEMORY IF POSSIBLE, ELSE ISSUE
FATAL ERROR

ARGUMENTS:

SIZE = UNSIGNED

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
STDIO - PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
RW - REPORT WRITER DEFINITIONS
FPSCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

FATAL - ISSUE FATAL ERROR MESSAGE
MALLOC

CALLED DIRECTLY BY:

3-182
--

CHKFLD - CHECK FIELD
CHKARY - CHECK ARRAY
CSTASH - CHARACTER STASH
WRTEXP - WRITE EXPRESSION
MKPOS - MAKE POSITION NODE
MAKINT - MAKE EXPRESSION INTO AN INTEGER
MAKSTR - MAKE EXPRESSION INTO A STRING

USED IN MAIN PROGRAM(S):
--

CHKFLD - CHECK FIELD
CHKFRM - CHECK FORM
CSTASH - CHARACTER STASH
MAKINT - MAKE EXPRESSION INTO AN INTEGER
MAKSTR - MAKE EXPRESSION INTO A STRING
MKPOS - MAKE POSITION NODE

3-183
NAME: PRCFIL
PURPOSE: PROCESS TEMPORARY FILE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: PRCFIL
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *PRCFIL(FILE_PTR)
FILE *FILE_PTR;

INPUTS/OUTPUTS:

INPUTS:
FILE *FILE_PTR FILE POINTER TO THE TEMPORARY FILE CONTAINING THE DIRECTORY LIST.

OUTPUTS:

DESCRIPTION
PROCESSES THE TEMPORARY FILE FINDING ALL THE FILE NAMES AND PLACES EACH ONE INTO THE SCREEN.

ARGUMENTS:

--------
FILE_PTR = FILE *

INCLUDE FILES:

3-184
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
STDTP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDIF - FDFE DATA STRUCTURES

ROUTINES CALLED:
-------------------
STRCHR
FGETS
STRRCHR
SPRINTF
MEMSET
FERROR

CALLED DIRECTLY BY:
-------------------
LISTIT - LIST IT

USED IN MAIN PROGRAM(S):
-------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: PREC
PURPOSE: PRECEDENCE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: WRTFDL
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDPE
DOCUMENTATION GROUP: FDPE

DESCRIPTION:
-----------------
INT PREC(EP)
   ENODE *EP;

RETURNS THE PRECEDENCE OF AN EXPRESSION. THIS ROUTINE IS
USED IN
TREEXP TO DETERMINE IF AN EXPRESSION REQUIRES PARENTHESIS.

ARGUMENTS:
-------------
EP = ENODE *

INCLUDE FILES:
-----------------
STDTYPE - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA

ROUTINES CALLED:
-----------------
PREC - PRECEDENCE

CALLED DIRECTLY BY:
---------------------
GETFLS/TREEXP - TREE EXPRESSION
PREC - PRECEDENCE

3-186
USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: PRSCMD
PURPOSE: PARSE COMMAND
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT ()
SOURCE FILE: PRSCMD
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

----------

SYNOPSIS

INT PRSCMD(CMD,PARSIZ,CMDNAM,PARAMTR,NUMPARM)
CHAR CMD[];
INT PARSIZ[MAXPARM];
CHAR *CMDNAM[];
CHAR *PARAMTR[MAXPARM];
INT *NUMPARM;

INPUTS/OUTPUTS:

INPUTS:

CMD - COMMAND LINE
PARSIZ - ARRAY OF PARAMETER SIZES
CMDNAM - ARRAY OF CHAR POINTERS TO LEGITIMATE COMMANDS

OUTPUTS:

PARAMTR - PARAMETERS ARE RETURN TO THIS ARRAY
NUMPARM - NUMBER OF PARAMETERS FOUND
RETURNS INT VALUE CORRESPONDING TO OPTION CHosen

DESCRIPTION

THIS MODULE PARSES THE COMMAND LINE AND RETURNS INT NUM
CORRESPONDING TO
OPTION AND PUTS IN 'NUMPARM' THE NUMBER OF PARAMETERS
FOUND AND IN
PARAMTR THE ACTUAL PARAMETERS FOUND.
ARGUMENTS:

CMD = CHAR [ ]
PARSIZ = INT [ MAXPARM ]
CMDNAM = CHAR * [ ]
PARAMTR = CHAR * [ MAXPARM ]
NUMPARM = INT *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

STRNCPY
MEMCMP
MEMSET
STRNCMP
STRLEN

CALLED DIRECTLY BY:

EDTMOD - EDIT MODE
FDFE - FORMS DRIVEN FORM EDITOR

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: PUTERR
PURPOSE: PUT ERROR
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

BOOL PUTERR(PATH, EXT, STR)
  CHAR PATH[], EXT[], STR[];

SET THE ERROR ATTRIBUTE FOR THE INVALID FIELD AND PUT THE
CURSOR THERE.
PUTERR(PATH, EXTENSION, EXPLANATION)

ARGUMENTS:

PATH = CHAR []
EXT = CHAR []
STR = CHAR []

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

STRCPY
STRCAT

3-190
PUTATT
MEMCMP
PMSGLC
PMSGLS
PUTCUR

CALLED DIRECTLY BY:

VALINP/CCKFRM - CHECK FORM
VALINP/CCKFLD - CHECK FIELD
VALINP/CCKHLP - CHECK HELP
VALINP/CCKITM - CHECK ITEM
VALINP/CCKNAM - CHECK NAME

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: SAVFLS
PURPOSE: SAVE FDL SOURCE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: SAVFLS
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
CHAR *SAVFLS(FRMNAM, SAVFD, DP)
  CHAR FRMNAM[];
  BOOL SAVFD;
  FIELD *DP;

INPUTS:
CHAR FRMNAM[]; ** THE NAME THE FORM IS THE BE SAVED UNDER
BOOL SAVFD; ** IF TRUE THE .FD FILES ARE ALSO WRITTEN **
FIELD *DP; ** POINTER TO LIST OF FORMS TO BE WRITTEN OUT

DESCRIPTION
A LIST OF FORMS POINTED TO BY DP IS TO BE WRITTEN OUT AS .FDL SOURCE.
THE FORMS ARE FIRST WRITTEN TO A TEMPORARY FILE WHICH IS FLANED TO CHECK FOR ERRORS IN VALUE EXPRESSIONS AND OVERLAPPING FIELDS. IF THERE ARE NO ERRORS THE FILE IS RENAMED. IF THE SAVFD FLAG IS TRUE THE .FD FILES ARE ALSO WRITTEN.

ARGUMENTS:

---
FRMMAM = CHAR []
SAVFD = BOOL
DP = FIELD

INCLUDE FILES:

---------------------
STDTYP - STANDARD TYPE DEFINITIONS
STUDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES

ROUTINES CALLED:

---------------------
ESCPY
SPRINTF
STRCPY
STRRCHR
STRCAT
FOPEN
SYSMSG
FLANCI - FLAN CALLABLE INTERFACE
WRTFDL - WRITE FDL FILE
REWIND
FCLOSE
RENAME
WRTFRM
DELFLD

CALLED DIRECTLY BY:

---------------------
EDTMOD - EDIT MODE

USED IN MAIN PROGRAM(S):

---------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: SCRMAN
PURPOSE: SCREEN MANAGER
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR (* ()
SOURCE FILE: SCRMAN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *SCRMAN(FRMPNT, RDONLY, ROW, COL)
FIELD *FRMPNT;
BOOL RDONLY;
INT *ROW;
INT *COL;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM WORKING ON
RDONLY - FLAG INDICATING WHETHER IN READ ONLY MODE OR NOT

OUTPUTS:
ROW - ROW RECEIVED FROM GETCUR ROUTINE
COL - COL RECEIVED FROM GETCUR ROUTINE

DESCRIPTION

THIS MODULE MANAGES THE SCREEN LAYOUT MODE: CALLING Routines TO TRANSLATE INTERNAL STRUCTURE TO SCREEN LAYOUT FORMAT AND VICE VERSA HANDLING ALL INPUT THROUGH AN OISR CHECKING THE VALUE OF PFKEY RETURNED

3-194
BY OISCR AND ERRR CODES RETURNED BY TRANLATE Routines TO DETERMINE WHAT ACTION SHOULD BE TAKEN.

ARGUMENTS:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>FRMPT</td>
<td>FIELD</td>
</tr>
<tr>
<td>RDOMLY</td>
<td>BOOL</td>
</tr>
<tr>
<td>ROW</td>
<td>INT</td>
</tr>
<tr>
<td>COL</td>
<td>INT</td>
</tr>
</tbody>
</table>

INCLUDED FILES:

- STDTYP - STANDARD TYPE DEFINITIONS
- FPPARM - FORM PROCESSOR PARAMETERS
- FPCODE - FORM PROCESSOR RETURN CODES
- FPD    - FORM PROCESSOR DATA
- FDFEFM - FDFE FORM DEFINITIONS
- FDFE   - FDFE DATA STRUCTURES

ROUTINES CALLED:

- SCRMAN/CHGPOS - CHANG POSITION
- SCRMAN/GETROW - GET ROW
- TRNSCR    - TRANSLATE SCREEN TO STRUCTURE
- ADDFRM    - 
- MEMCMP    - 
- RMVPAG    - 
- GETCUR    - 
- GDATA     - 
- OISCR     - 
- TRNSTR    - TRANSLATE STRUCTURE TO SCREEN
- PMSGLS    - 
- PMSGGLC   - 
- PDATA     - 

CALLED DIRECTLY BY:

- LAYOUT    - LAYOUT MODE

USED IN MAIN PROGRAM(S):

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: SCRMAN/CHGPOS
PURPOSE: CHANG POSITION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *()
SOURCE FILE: SCRMAN
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

STATIC CHAR *CHGPOS(FRMPNT)
FIELD *FRMPNT;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM WORKING ON

OUTPUTS:
STANDARD FORM PROCESSOR RETURN CODE

DESCRIPTION

THIS MODUE ALTERS THE INTERNAL DATA STRUCTURE OF FIELD MARKED SO THAT ITS NEW ROW = ROW OBTAINED FROM GETCUR AND THE COL = COL OBTAINED FROM GETCUR + 1. IT SENDS THE USER APPROPRIATE ERROR MESSAGES IF HE/SHE MAKES AN ERROR (NOT MARKING A FIELD TO BE MOVED, FOR EXAMPLE).

ARGUMENTS:

FRMPNT - FIELD *
INCLUDE FILES:

- STDYP - STANDARD TYPE DEFINITIONS
- FPPARM - FORM PROCESSOR PARAMETERS
- FPCODE - FORM PROCESSOR RETURN CODES
- FPD - FORM PROCESSOR DATA
- FDFEFM - FDFE FORM DEFINITIONS
- FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

- SCRM/GETROW - GET ROW
- GTCPFD - GET USING CURSOR POSITION FIELD
- MAX
- MIN
- ABS
- TRNSTR - TRANSLATE STRUCTURE TO SCREEN
- PMSGLS
- GETCUR
- MEMCMP

CALLED DIRECTLY BY:

- SCRM - SCREEN MANAGER

USED IN MAIN PROGRAM(S):

- FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
PS 620144402
1 November 1985

FORMS DRIVEN FORM EDITOR Module Documentation

NAME: SCRMAN/GETROW
PURPOSE: GET ROW
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: INT()
SOURCE FILE: SCRMAN
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
-----------------
SYNOPSIS

STATIC INT GETROW(FQ_NAM)
   PATH   FQ_NAM;

INPUTS/OUTPUTS:

INPUTS:
   FQ-NAM - FULLY QUALIFIED NAME RETURNED FROM GETCUR

OUTPUTS:
   RETURNS ROW OR 0 IF NOT ARRAY

DESCRIPTION

THIS MODULE RETURNS ROW BASED ON ARRAY INDEX

ARGUMENTS:
-----------------
FQ_NAM = PATH

INCLUDE FILES:
-----------------
STDTYP - STANDARD TYPE DEFINITIONS
FPPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS

3-198
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:
------------------
ESCPY
STRRCHR
NATOI
STRLEN

CALLED DIRECTLY BY:
---------------------
SCRMAN - SCREEN MANAGER
SCRMAN/CHGPOS - CHANG POSITION

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MMAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSCR
PURPOSE: TRANSLATE SCREEN TO STRUCTURE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: .c
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *TRNSCR(FRMPNT)
FIELD *FRMPNT;

INPUTS/OUTPUTS:

INPUTS:
FRMPNT - POINTER TO FORM BEING WORKED ON

OUTPUTS:
RETURN NULL IF NO ERRORS
RETURN ERROR CODE IF NO EITHER USER(OVERLAP) ERRORS OR SYSTEM ERRORS

DESCRIPTION

THIS MODULE VALIDATES AND TRANSLATES USER INPUT AT LAYOUT MODE TO INTERNAL DATA STRUCTURE.

ARGUMENTS:

---------------
FRMPNT = FIELD *

INCLUDE FILES:

------------
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FP CODE - FORM PROCESSOR RETURN CODES
FPARM - FORM PROCESSOR PARAMETERS
FPD - FORM PROCESSOR DATA
FDPEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MEMSET
TRNSCR/LDPMINF - LOAD PROMPT INFORMATION
INSFLD - INSERT FIELD
MODFLD - MODIFY FIELD
TRNSCR/FLCST - FILL LOCATION STRUCTURE
TRNSCR/PARSCRN - PARSE SCREEN DATA
MEMCMP
ISSPACE
TRNSCR/SPSYMB - SPECIAL SYMBOL CHECK
TRNSCR/GTFMPMT - GET FORM PROMPT INFORMATION
TRNSCR/GTPINF - GET PROMPT INFORMATION
TRNSCR/MTCHPMT - MATCH PROMPT WITH FIELD
TRNSCR/FRLCST - FREE LOCATION STRUCTURES
MODFRM - MODIFY FORM
DELPFLD
GTCPFD - GET USING CURSOR POSITION FIELD
FLSTRC - FIELD STRUCTURE TRANSLATION
MITOA
SPRINTF
MEMCPY
STRLEN
FLFMST - FIELD TO FORM STRUCTURE TRANSLATION

CALLED DIRECTLY BY:

SCRMAN - SCREEN MANAGER

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSCR/FLCST
PURPOSE: FILL LOCATION STRUCTURE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

STATIC CHAR *FLCST(ROW, COL, DPTH, WDTH)
REGISTER INT ROW, COL, DPTH, WDTH;

INPUTS/OUTPUTS:

INPUTS:
ROW - ROW WHERE FIELD IS LOCATED
COL - COL WHERE FIELD IS LOCATED
DPTH - DEPTH OF FIELD LOCATED
WDTH - DEPTH OF FIELD LOCATED

OUTPUTS:
RETURN NULL IF NO ALLOCATION ERRORS
RETURN ALCERR IF ALOCATION ERROR.

DESCRIPTION

THIS MODULE CREATES AND FILLS FIELD LOCATION STRUCTURE.

ARGUMENTS:

ROW = INT
COL = INT
DPTH = INT
WDTH = INT
INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
FPD  - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MALLOC

CALLED DIRECTLY BY:

TRNSCR/PARSCRN - PARSE SCREEN DATA
TRNSCR   - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: TRNSCR/FRLCST
PURPOSE: FREE LOCATION STRUCTURES
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
STATIC VOID FRLCST();

INPUTS/OUTPUTS:

INPUTS:
NONE

OUTPUTS:
NONE

DESCRIPTION
THIS MODULE FREES UP ALL DATA LOCATION STRUCTURES
ALLOCATED BY TRNSCR
AND ITS DEPENDENT STATIC SUB MODULES (ALL FLDLOC AND
PMTLOC STRUCTURES).

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

3-204
Routines called:
-----------------
FREE

Called directly by:
-------------------
TRNSCR - TRANSLATE SCREEN TO STRUCTURE

Used in main program(s):
------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSCR/GTFMPMT
PURPOSE: GET FORM PROMPT INFORMATION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *( )
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: . C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
------------
SYNOPSIS
STATIC CHAR *GTFMPMT(ROW, COL, N)
REGISTER INT ROW, COL, N;

INPUTS/OUTPUTS:

INPUTS:
ROW - ROW OF PROMPT ASSOCIATED WITH FORM (ON SCREEN ARRAY)
COL - COL OF PROMPT ASSOCIATED WITH FORM (ON SCREEN ARRAY)
N - NUMBER OF THE PROMPT ASSOCIATED WITH FORM

OUTPUTS:
NONE

DESCRIPTION
THIS MODULE GETS FORM PROMPT INFO FOR PROMPT AND PUTS IT INTO COMMON DATA STRUCTURE WHICH INSFRM AND MODFRM USE

ARGUMENTS:
-----------
ROW = INT
COL = INT
N = INT
INCLUDE FILES:

- STDTYPE - STANDARD TYPE DEFINITIONS
- CTYP" - "*** PURPOSE NOT FOUND BY STRIPPER "***
- FPCODE - FORM PROCESSOR RETURN CODES
- FPPARM - FORM PROCESSOR PARAMETERS
- FPD - FORM PROCESSOR DATA
- FDFEFM - FDFE FORM DEFINITIONS
- FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

- MITGA

CALLED DIRECTLY BY:

- TRNSCR - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

- FDFE/Main - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSCR/GTPINF
PURPOSE: GET PROMPT INFORMATION
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR * ()
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
-----------
SYNOPSIS
STATIC CHAR *GTPINF(ROW, COL, DIRFLG, RCOL, FSTFLG, ERRFLG)
REGISTER INT ROW, COL, DIRFLG;
INT *RCOL;
BOOL *FSTFLG, *ERRFLG:

INPUTS/OUTPUTS:

INPUTS:
ROW - ROW WHERE PROMPT SYMBOL WAS FOUND
COL - COL WHERE PROMPT SYMBOL WAS FOUND
DIRFLG - TYPE OF PROMPT (DIRECTION OF ASSOCIATION)
FSTFLG - FIRST PROMPT IN COL FLAG

OUTPUTS:
RCOL - LOCATION TO RESUME SEARCH THOUGH SCREEN ARRAY.
FSTFLG - FIRST PROMPT IN COL FLAG
ERRFLG - SET TO TRUE IF PROMPT IS OPEN ENDED

DESCRIPTION
THIS MODULE DETERMINES LOCATION OF PROMPT IF IT CAN AND CREATE A
PROMPT LOCATION STRUCTURE WITH LOCATION AND CONTENT OF
PROMPT.
ARGUMENTS:

- **ROW** = INT
- **COL** = INT
- **DIRFLG** = INT
- **RCOL** = INT
- **FSTFLG** = BOOL
- **ERRFLG** = BOOL

INCLUDE FILES:

- **STDTYPE** - STANDARD TYPE DEFINITIONS
- **CTYPE** - **** PURPOSE NOT FOUND BY STRIPPER ****
- **FPCODE** - FORM PROCESSOR RETURN CODES
- **FPPARM** - FORM PROCESSOR PARAMETERS
- **FPD** - FORM PROCESSOR DATA
- **FDFEFM** - FDFE FORM DEFINITIONS
- **FDFE** - FDFE DATA STRUCTURES

ROUTINES CALLED:

- **MAX**
- **MIN**
- **MALLOC**
- **ISSPACE**
- **TRANSCHR/SPSYM** - SPECIAL SYMBOL CHECK

CALLED DIRECTLY BY:

- **TRANSCHR** - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

- **FDFE/MAIN** - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME:       TRNSCR/LDPMINF
PURPOSE:    LOAD PROMPT INFORMATION
LANGUAGE:   C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: .C
HOST:       
SUBSYSTEM:  UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

-------------------
SYNOPSIS
STATIC VOID LDPMINF(FLCPNT)
STRUCT FLDLOC *FLCPNT:

INPUTS/OUTPUTS:

INPUTS:
   FLCNTP - POINTER TO FIELD LOCATION STRUCTURE WHOSE
            PROMPT IS BEING WORKED ON

OUTPUTS:

DESCRIPTION
   THIS MODULE LOADS PROMPT INFO OF LAYOUT FIELD INTO COMMON
   DATA STRUCTURE USED BY AND PASSED TO INSFLD(MODFLD) FIELD ETC.

ARGUMENTS:

-------------------
FLCPNT = STRUCT FLDLOC *

INCLUDE FILES:

-------------------
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPCODE - FORM PROCESSOR RETURN CODES

3-210
FPPARM - FORM PROCESSOR PARAMETERS
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MEMCPY
MTOA

CALLED DIRECTLY BY:

TRNSCR - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSCR/MTCHPMT
PURPOSE: MATCH PROMPT WITH FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

-------------
SYNOPSIS
STATIC BOOL MTCHPMT()

INPUTS/OUTPUTS:

INPUTS: NONE

OUTPUTS:
RETURNS MATCHED/NOT_MATCHED FLAG

DESCRIPTION
THIS MODULE MATCHES PROMPTS WITH FIELDS. IF PROMPT IS UNMATCHED OR AMBIGUOUSLY MATCHED RETURN FAILURE ELSE PUT POINTER TO PROMPT LOCATION FIELD INTO FIELD LOCATION FIELD PROMT POINTER.

INCLUDE FILES:

-------------
STDTYPE   - STANDARD TYPE DEFINITIONS
CTYPE     - PURPOSE NOT FOUND BY STRIPPER ****
FPFPCODE  - FORM PROCESSOR RETURN CODES
FPFPARM   - FORM PROCESSOR PARAMETERS
FPD       - FORM PROCESSOR DATA
FDFEFM    - FDFE FORM DEFINITIONS

3-212
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:
---------------

MAX

CALLED DIRECTLY BY:
-------------------

TRNSCR  - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):
------------------------

FDFE/MAIN  - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSCR/PARSCRN
PURPOSE: PARSE SCREEN DATA
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR *()
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

----------

SYNOPSIS

STATIC CHAR *PARSCRN(ROW, COL, MAXCOL)
    REGISTER INT ROW, COL, MAXCOL;

INPUTS/OUTPUTS:

INPUTS:
    ROW - ROW IN WHICH BEGINNING FIELD SYMBOL OCCURRED
    COL - COL IN WHICH BEGINNING FIELD SYMBOL OCCURRED
    MAXCOL - MAX COL IN WHICH ENDING FIELD SYMBOL CAN BE FOUND WITHOUT
              AN OVERLAP ERROR OCCURRING.

OUTPUTS:
    RETURN NULL IF NO ERRORS
    RETURN ERROR CODE IF EITHER OVERLAP ERRORS OR SYSTEM ERRORS

DESCRIPTION

THIS MODULE PARSES SCREEN TO DETERMINE THE LOCATION OF
FIELDS
WHILE CHECKING FOR OVERLAPPING OF OTHER FIELDS OR PROMPTS.

ARGUMENTS:

----------

    ROW = INT
    COL = INT
MAXCOL = INT

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPCODE - FORM PROCESSOR RETURN CODES
FPARM - FORM PROCESSOR PARAMETERS
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

TRNSCR/SPSYMB - SPECIAL SYMBOL CHECK
ISSPACE
TRNSCR/PARSCRN - PARSE SCREEN DATA
TRNSCR/FLCST - FILL LOCATION STRUCTURE
MIN

CALLED DIRECTLY BY:

TRNSCR/PARSCRN - PARSE SCREEN DATA
TRNSCR - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSCR/SPSYMB
PURPOSE: SPECIAL SYMBOL CHECK
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: TRNSCR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

STATIC BOOL SPSYMB(SCRCHR)
CHAR SCRCHR;

INPUTS/OUTPUTS:

INPUTS:
SCRCHR - CHARACTER TO BE CHECKED

OUTPUTS:
RETURNS TRUE/FALSE FLAG

DESCRIPTION

THIS MODULE RETURNS TRUE IF CHAR A SPECIAL SYMBOL ELSE
RETURNS FALSE

ARGUMENTS:

SCRCHR - CHAR

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS

3-216
FPD - FORM PROCESSOR DATA
FDFE FM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

CALLED DIRECTLY BY:

-----------------------
TRNSCR/GTPINF - GET PROMPT INFORMATION
TRNSCR/PARSCRN - PARSE SCREEN DATA
TRNSCR - TRANSLATE SCREEN TO STRUCTURE

USED IN MAIN PROGRAM(S):

-----------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMSEDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSTR
PURPOSE: TRANSLATE STRUCTURE TO SCREEN
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TRNSTR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

VOID TRNSTR(FRMPNT,FLDPNT,RDONLY)
  FIELD *FRMPNT;
  FIELD *FLDPNT;
  BOOL RDONLY;

INPUTS/OUTPUTS:

INPUTS:
  FRMPNT - POINTER TO FORM BEING WORKED ON
  FLDPNT - POINTER TO FIELD BEING WORKED ON IF NULL DO ENTIRE FORM
  RDONLY - FLAG INDICATING WHETHER IN READ ONLY MODE OR NOT

OUTPUTS:
  NONE

DESCRIPTION
  THIS MODULE TRANSLATES INTERNAL DATA STRUCTURE INTO SCREEN LAYOUT

ARGUMENTS:

  FRMPNT = FIELD *

3-218
FLDPNT = FIELD
RONLY = BOOL

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD  - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE  - FDFE DATA STRUCTURES

ROUTINES CALLED:

MEMSET
TRNSTR/FLPRMPT - FILL PROMPT
TRNSTR/FLFLD - FILL FIELD

CALLED DIRECTLY BY:

SCRMAN - SCREEN MANAGER
SCRMAN/CHGPOS - CHANG POSITION

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSTR/FLFLD
PURPOSE: FILL FIELD
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TRNSTR
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
STATIC VOID FLFLD(FLDPNT,RDONLY)
FIELD *FLDPNT;
BOOL RDONLY;

INPUTS/OUTPUTS:

INPUTS:
FLDPNT - POINTER TO FIELD BEING WORKED ON
RDONLY - FLAG INDICATING WHETHER IN READ ONLY MODE OR NOT

OUTPUTS:
NONE

DESCRIPTION
THIS MODULE TRANSLATES INTERNAL DATA STRUCTURE OF ONE FIELD INTO SCREEN
LAYOUT FORMAT.

ARGUMENTS:

----------------
FLDPNT = FIELD *
RDONLY = BOOL

INCLUDE FILES:

3-220
STDTYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

TRNSTR/FLPRMPT - FILL PROMPT
PMSGLS
PRINTF
MIN
MAX
TRNSTR/GARINF - GET ARRAY INFORMATION

CALLED DIRECTLY BY:

TRNSTR - TRANSLATE STRUCTURE TO SCREEN

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)

3-221
NAME: TRNSTR/FLPRMPT
PURPOSE: FILL PROMPT
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TRNSTR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

STATIC VOID FLPRMPT(FLFPNT, TXTPNT, PRMSYM)
FIELD *FLDPNT;
TEXT *TXTPNT;
CHAR PRMSYM;

INPUTS/OUTPUTS:

INPUTS:
FLDPNT - POINTER TO FIELD ON WHICH IS FOUND TEXT (PROMPT)
TXTPNT - POINTER TO TEXT (PROMPT) BEING WORKED ON
PRMSYM - PROMT SYMBOL (LAYOUT PROMPT SYMBOLS)

OUTPUTS:
NONE

DESCRIPTION

THIS MODULE FILLS SCREEN STRUCTURE WITH PROMPT STRING FROM TEXT
STRUCTURE AT THE APPROPRIATE ROW AND COL WITH THE
APPROPRIATE SYMBOL PASSED DOWN IN PRMSYM.

ARGUMENTS:

FLDPNT = FIELD *
TXTPNT = TEXT *
PRMSYM = CHAR
INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MIN

CALLED DIRECTLY BY:

TRNSTR/FLFLD - FILL FIELD
TRNSTR - TRANSLATE STRUCTURE TO SCREEN

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: TRNSTR/GARINF
PURPOSE: GET ARRAY INFORMATION
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: TRNSTR
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

STATIC VOID GARINF(FLDPNT, WDTH, DPTH, ARWDTH, ARDPTH)
FIELD
INT *FLDPNT;
*WDTH, *DPTH, *ARWDTH, *ARDPTH;

INPUTS/OUTPUTS:

INPUTS:
FLDPNT - POINTER TO ARRAY BEING WORKED ON
RDONLY - FLAG INDICATING WHETHER IN READ ONLY MODE OR NOT

OUTPUTS:
WDTH - WIDTH OF FIRST MEMBER OF ARRAY
DPTH - DEPTH OF FIRST MEMBER OF ARRAY
ARWDTH - WIDTH OF WHOLE OF ARRAY
ARDPTH - DEPTH OF WHOLE OF ARRAY

DESCRIPTION
THIS MODULE OBTAINS ROW, COL, WIDTH, DEPTH, OF FIRST MEMBER OF ARRAY AND THE WIDTH AND DEPTH OF ENTIRE ARRAY AND RETURNS THEM TO CALLER

ARGUMENTS:

FLDPNT - FIELD
WDTH = INT *
DPTH = INT *
ARWDTH = INT *
ARDPTH = INT *

INCLUDE FILES:

STDTP - STANDARD TYPE DEFINITIONS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

ABS

CALLED DIRECTLY BY:

TRNSTR/FLFLD - FILL FIELD

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: VALINP
PURPOSE: VALIDATE INPUT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

BOOL VALINP(FDP, DP, EDTTYP, BLNKSTOP)

FIELD *FDP;
FIELD *DP;
INT EDTTYP;
BOOL BLNKSTOP;

INPUTS:
FIELD *FDP; ** POINTER TO FORM CONTAINING FIELD TO BE VALIDATED
FIELD *DP; ** POINTER TO FIELD TO BE VALIDATED **
INT EDTTYP; ** INDICATES THE COMBINATION OF THINGS TO VALIDATE
BOOL BLNKSTOP; ** TRUE IF CHECKING IS TO STOP ON THE FIRST BLANK FIELD

DESCRIPTION
PERFORMS VALIDATION CHECKS ON FIELDS. FDP AND DP INDICATE THE FORM AND FIELD TO BE VALIDATED. EDTTYP INDICATES THE COMBINATION OF THINGS TO BE VALIDATED. THE OBJECTS TO BE VALIDATED ARE CONTAINED IN GLOBAL DATA. IF THE OBJECTS PASS THE VALIDATION CHECKS VALINP RETURNS A TRUE.

ARGUMENTS:

3-226
FDP = FIELD
DP = FIELD
EDTTYP = INT
BLNKSTOP = BOOL

INCLUDE FILES:

STDTYPE - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MEMCMP
SPRINTF
VALINP/CCKITM - CHECK ITEM
ABORT
VALINP/CCKFRM - CHECK FORM
VALINP/CCKHLIP - CHECK HELP
VALINP/CCKFLD - CHECK FIELD
VALINP/CCKNAM - CHECK NAME

CALLED DIRECTLY BY:

EDTFLD - EDIT FIELD
EDTWHL - EDIT WHOLE
GFDINP - GET FIELD INPUT
GWHINP - GET WHOLE INPUT

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: VALINP/CCKFLD
PURPOSE: CHECK FIELD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

------------------------
STATIC BOOL CCKFLD(PATH, ARPATH, FLDPTR, DP, FDP)
PATH PATH[], ARPATH[];
STRUCT COMINFO *FLDPTR:
FIELD *DP, *FDP;

VALIDATE A FIELD.
CCKFLD (FIELD PATH, ARRAY PATH, POINTER TO DISPLAY, POINTER TO FPD FIELD)

ARGUMENTS:

------------------------
PATH = PATH []
ARPATH = PATH []
FLDPTR = STRUCT COMINFO *
DP = FIELD *
FDP = FIELD *

INCLUDE FILES:

------------------------
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FPARM - FORM PROCESSOR PARAMETERS
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

3-228
ROUTINES CALLED:

MATOI
VALINP/CCKPRM - CHECK PROMPT
MIN
ATOI
SPRINTF
MEMCMP
FNDATT - FIND ATTRIBUTE
STRCHR
STRCMP
VALINP/CCKRSV - CHECK FOR RESERVED WORD
STRLEN
STRSPN
PUTERR - PUT ERROR
ISALPHA
ESCPY

CALLED DIRECTLY BY:

VALINP - VALIDATE INPUT

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: VALINP/CCKFRM
PURPOSE: CHECK FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST: 
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
----------
STATIC BOOL CCKFRM(PATH, FRMPTR)
   CHAR PATH[];
   STRUCT FRMINFO *FRMPTR;

VALIDATE A FORM.
CCKFRM (PATH TO FORM, POINTER TO DISPLAY FORM)

ARGUMENTS:
----------
PATH = CHAR []
FRMPTR = STRUCT FRMINFO *

INCLUDE FILES:
----------
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:
----------
SPRINTF
STRCHR
MIN

3-230
MEMCMP
PUTERR - PUT ERROR
FNDATT - FIND ATTRIBUTE
ESCPY

CALLED DIRECTLY BY:
---------------------
VALINP - VALIDATE INPUT

USED IN MAIN PROGRAM(S):
------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: VALINP/CCKHLP
PURPOSE: CHECK HELP
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

-----------------  
STATIC BOOL CCKHLP(PATH, HLPPTR)  
PATH PATH;  
STRUCT ITMHELP *HLPPTR;  

VALIDATE HELP.  
CCKHLP(PATH TO FIELD, POINTER TO DISPLAY FOR HELP)

ARGUMENTS:

-----------  
PATH = PATH  
HLPPTR = STRUCT ITMHELP *

INCLUDE FILES:

-----------------  
STDTYP - STANDARD TYPE DEFINITIONS  
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****  
FPD - FORM PROCESSOR DATA  
FPCODE - FORM PROCESSOR RETURN CODES  
FPFARM - FORM PROCESSOR PARAMETERS  
FDFEFM - FDFE FORM DEFINITIONS  
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

-----------------  
STRCMP  
STRSPN  
ISALPHA

3-232
STRNCMP
PUTERR - PUT ERROR
STRLEN
STRUPC
ESCPY
MEMCMP
STRCHR

Called directly by:

---------------
VALINP - VALIDATE INPUT

Used in main program(s):

---------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: VALINP/CCKITM
PURPOSE: CHECK ITEM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

---------------------
STATIC BOOL CCKITM(PATH, ITMPTR)
PATH PATH;
STRUCT ITMCHCK *ITMPTR;

VALIDATE ITEM DOMAIN TYPE STUFF.
CCKITM(FIELD PATH, POINTER TO DISPLAY)

ARGUMENTS:

-------------
PATH = PATH
ITMPTR = STRUCT ITMCHCK *

INCLUDE FILES:

-------------
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FP CODE - FORM PROCESSOR RETURN CODES
FPARM - FORM PROCESSOR PARAMETERS
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

-------------
ATOI
ESCPY
MEMCMP

3-234
PUTERR - PUT ERROR
STRCHR

CALLED DIRECTLY BY:

-------------------

VALINP - VALIDATE INPUT

USED IN MAIN PROGRAM(S):

----------------------

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: VALINP/CCKNAM
PURPOSE: CHECK NAME
LANGUAGE: G
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

STATIC BOOL CCKNAM(FDP, BLNKSTOP)
FIELD *FDP;
BOOL BLNKSTOP;

CHECK FOR DUPLICATE NAMES ON WHLEDT

ARGUMENTS:

FDP = FIELD *
BLNKSTOP = BOOL

INCLUDE FILES:

STDYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FPPARM - FORM PROCESSOR PARAMETERS
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

MEMCMP
PUTERR - PUT ERROR
SPRINTF
PS 620144402
1 November 1985

CALLED DIRECTLY BY:

-----------------------------
VALINP - VALIDATE INPUT

USED IN MAIN PROGRAM(S):

-----------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: VALINP/CCKPRM
PURPOSE: CHECK PROMPT
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

-----------
STATIC BOOL CCKPRM(POS) 
CHAR POS[ ];

CHECK THE FIELD PROMPT LOCATION FOR A LEGAL POSITION.
CCKPRM(STRING)

ARGUMENTS:

-----------
POS = CHAR []

INCLUDE FILES:

-----------
STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - ***** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FPFARM - FORM PROCESSOR PARAMETERS
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES

ROUTINES CALLED:

-----------
MEMCMP

CALLED DIRECTLY BY:

----------------
VALINP/CCKFLD - CHECK FIELD

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: VALINP/CCKRSV
PURPOSE: CHECK FOR RESERVED WORD
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

-----------

STATIC BOOL CCKRSV(FLDNAM)
    CHAR FLDNAM[];

CHECK THE FIELD NAME AGAINST THE RESERVED WORD LIST.
CCKRSV(STRING)

ARGUMENTS:

----------

FLDNAM = CHAR []

INCLUDE FILES:

-----------

STDTYPE   - STANDARD TYPE DEFINITIONS
CTYPE     - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD       - FORM PROCESSOR DATA
FPCODE    - FORM PROCESSOR RETURN CODES
FPARM     - FORM PROCESSOR PARAMETERS
FDFEFM    - FDFE FORM DEFINITIONS
FDFE      - FDFE DATA STRUCTURES

ROUTINES CALLED:

----------

ESCPY
STRCMP

CALLED DIRECTLY BY:

3-240
VALINP/CCKFLD - CHECK FIELD

USED IN MAIN PROGRAM(S):

------------------------
FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
NAME: VALINP/CCKVAL
PURPOSE: CHECK VALUE
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: BOOL ()
SOURCE FILE: VALINP
SOURCE FILE TYPE: .C
HOST: SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

-----------

STATIC BOOL CCKVAL(PATH, VALPTR)
    PATH PATH;
    STRUCT ITMVAL *VALPTR;

VALIDATE AN EXPRESSION (ACTUALLY DONE BY FLAN).
CCKVAL(FIELD TO PATH, POINTER TO DISPLAY FOR VALUE)

ARGUMENTS:

-----------

PATH = PATH
VALPTR = STRUCT ITMVAL *

INCLUDE FILES:

-----------

STDTYP - STANDARD TYPE DEFINITIONS
CTYPE - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
FPCODE - FORM PROCESSOR RETURN CODES
FPARM - FORM PROCESSOR PARAMETERS
FDFEFM - FDFE FORM DEFINITIONS
FDFE - FDFE DATA STRUCTURES
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: VIEW
PURPOSE: VIEW A FORM
LANGUAGE: C
MODULE TYPE: FUNCTION
FUNCTION TYPE: CHAR (*)
SOURCE FILE: VIEW
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

CHAR *VIEW(FORMNAME)
NAME FORMNAME;

INPUTS/OUTPUTS:

INPUTS:
NAME FORMNAME NAME OF THE FORM TO BE VIEWED.

OUTPUTS:

DESCRIPTION
RETURNS STANDARD FORM PROCESSOR ERROR STRINGS OR IF SUCCESSFUL, A NULL POINTER.

ARGUMENTS:

-----
FORMNAME - NAME

INCLUDE FILES:

---------
STDTYP - STANDARD TYPE DEFINITIONS

3-243
FPARM - FORM PROCESSOR PARAMETERS
FPCODE - FORM PROCESSOR RETURN CODES
FPD - FORM PROCESSOR DATA
FDFFFM - FDFF FORM DEFINITIONS
FDFF - FDFF DATA STRUCTURES

ROUTINES CALLED:
-------------------
ADDFRM
MEMCMP
SYSMSG
CLSF RM
RMVPA G
GWPDNO
PMSGLS
OISGR

CALLED DIRECTLY BY:
-------------------
FDFF - FORMS DRIVEN FORM EDITOR

USED IN MAIN PROGRAM(S):
-----------------------
FDFF/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFF)
NAME: WARNING
PURPOSE: ISSUE WARNING MESSAGE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID ()
SOURCE FILE: FLANERR
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS
VOID WARNING(S, A, B, C, D, E, F)

DESCRIPTION
PRINTS A WARNING MESSAGE ON STDERR

ARGUMENTS:

S = CHAR *
A = CHAR *
B = CHAR *
C = CHAR *
D = CHAR *
E = CHAR *
F = CHAR *

INCLUDE FILES:

STDTYP - STANDARD TYPE DEFINITIONS

ROUTINES CALLED:

PMSGLS
STRLEN
PRINTF
CALLED DIRECTLY BY:

chkfrm - check form

USED IN MAIN PROGRAM(S):

chkfrm - check form
FORMS DRIVEN FORM EDITOR Module Documentation

**NAME:** WRTEXP  
**PURPOSE:** WRITE EXPRESSION  
**LANGUAGE:** C  
**MODULE TYPE:** FUNCTION  
**FUNCTION TYPE:** CHAR * ()  
**SOURCE FILE:** FLANSP  
**SOURCE FILE TYPE:** .C  
**HOST:**  
**SUBSYSTEM:** UI  
**SUBDIRECTORY:** FE  
**DOCUMENTATION GROUP:** FDFE/FLAN

**DESCRIPTION:**

---

**SYNOPSIS**

CHAR *WRTEXP(EP)  
ENODE *EP:

**INPUTS:**  
EP - EXPRESSION TO WRITE

**OUTPUTS:**  
RETURNS A POINTER TO THE WRITTEN EXPRESSION OR NULL FOR ERROR

**DESCRIPTION**  
RETURNS A POINTER TO THE CHARACTER STRING REPRESENTING THE GIVEN EXPRESSION, OR NULL IF AN ERROR IS DETECTED.

**ARGUMENTS:**

---

EP = ENODE *

**INCLUDE FILES:**

---

STDTYP - STANDARD TYPE DEFINITIONS  
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****  
FPD - FORM PROCESSOR DATA  
RW - REPORT WRITER DEFINITIONS  
FPCODE - FORM PROCESSOR RETURN CODES
ROUTINES CALLED:

--------------
FREE
WRTEXP      - WRITE EXPRESSION
MEMCPY
MYALLOC     - MY MALLOC
STRLEN
SPRINTF

CALLED DIRECTLY BY:

--------------
CHKFLD      - CHECK FIELD
WRTEXP      - WRITE EXPRESSION

USED IN MAIN PROGRAM(S):

--------------
CHKFLD      - CHECK FIELD
NAME: WRTFDL

PURPOSE: WRITE FDL FILE

LANGUAGE: C

MODULE TYPE: SUBROUTINE

FUNCTION TYPE: VOID ()

SOURCE FILE: WRTFDL

SOURCE FILE TYPE: .C

HOST:

SUBSYSTEM: UI

SUBDIRECTORY: FDFE

DOCUMENTATION GROUP: FDFE

DESCRIPTION:

SYNOPSIS

VOID WRTFDL(FRMPTR, FILPTR)
    FIELD *FRMPTR;
    FILE *FILPTR;

INPUTS:

FIELD *FRMPTR; ** THE LIST OF FORMS FOR WHICH .FDL IS TO BE WRITTEN
FILE *FILPTR; ** THE FILE TO WHICH THE FORMS ARE TO BE WRITTEN

DESCRIPTION

GIVEN A POINTER TO A LIST OF FORMS (FRMPTR) AND A FILE POINTER (FILPTR),
THE SOURCE LANGUAGE REPRESENTATION OF THE FPD LIST IS CREATED.

ARGUMENTS:

FRMPTR = FIELD *
FILPTR = FILE *

INCLUDE FILES:

STDTYPE - STANDARD TYPE DEFINITIONS
STDIO - **** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA
ROUTINES CALLED:

FPRINTF
BLEN
ESCPY
STRCMP
STRNCMP
WRTFDL/ARYREF - ARRAY REFERENCE

CALLED DIRECTLY BY:

SAVFLS - SAVE FDL SOURCE

USED IN MAIN PROGRAM(S):

FDFE/MAN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
FORMS DRIVEN FORM EDITOR Module Documentation

NAME: WRTFDL/ARYREF
PURPOSE: ARRAY REFERENCE
LANGUAGE: C
MODULE TYPE: SUBROUTINE
FUNCTION TYPE: VOID()
SOURCE FILE: WRTFDL
SOURCE FILE TYPE: .C
HOST:
SUBSYSTEM: UI
SUBDIRECTORY: FDFE
DOCUMENTATION GROUP: FDFE

DESCRIPTION:
-----------
STATIC VOID ARYREF(ARYDEF, DP)
  CHAR ARYDEF[];
  FIELD *DP;

CREATES THE FORMS LANGUAGE REPRESENTATION OF AN ARRAY
SPECIFICATION
FROM THE UID.CURFPD-

ARGUMENTS:
-----------
ARYDEF = CHAR []
DP = FIELD *

INCLUDE FILES:
---------------
STDTYP - STANDARD TYPE DEFINITIONS
STDIO - "**** PURPOSE NOT FOUND BY STRIPPER ****
FPD - FORM PROCESSOR DATA

ROUTINES CALLED:
------------------
ABS
SPRINTF
STRCAT

CALLED DIRECTLY BY:

3-251
WRTFDL - WRITE FDL FILE

USED IN MAIN PROGRAM(S):

FDFE/MAIN - MAIN MODULE FOR FORMS DRIVEN FORMS EDITOR (FDFE)
3.10.9 Include File Descriptions

The following list contains a purpose and description of each include file listed in 3.10.4 as specified in the source code. The language it is written in is also given.
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: FDFE
PURPOSE: FDFE DATA STRUCTURES
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
MAIN INCLUDE FILE FOR FDFE
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: FDFEFM
PURPOSE: FDFE FORM DEFINITIONS
LANGUAGE: C

DESCRIPTION:
----------
FILE NAME: FDFEINI
PURPOSE:  FDFE INITIALIZATIONS
LANGUAGE:  C

DESCRIPTION:

DESCRIPTION
INITIALIZING INCLUDE FILE FOR FDFE
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: FFFV2
PURPOSE: FORM FILE FORMAT - VERSION 2
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
RECORD LAYOUTS FOR THE BINARY FORM DEFINITION FILE
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: FLAN
PURPOSE: FLAN INTERNAL STRUCTURES
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
  AUXILIARY DATA STRUCTURES USED BY FLAN.
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: FPCODE
PURPOSE: FORM PROCESSOR RETURN CODES
LANGUAGE: C

DESCRIPTION:

------------
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: FPD
PURPOSE: FORM PROCESSOR DATA
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
DATA DEFINITIONS FOR ALL FORM PROCESSOR (INCLUDING MONITOR) DATA.
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: FPDINI
PURPOSE: FPD INITIALIZATION
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
INITIALIZED VERSION OF UID FOR INCLUSION IN MAIN PROGRAM.
FILE NAME: FPPARM
PURPOSE: FORM PROCESSOR PARAMETERS
LANGUAGE: C

DESCRIPTION:

DESCRIPTION: THESE DATA DEFINITIONS ARE USED IN THE FORM PROCESSOR ROUTINES.
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: NTM
PURPOSE: NTM INTERFACE INCLUDE FILE
LANGUAGE: C

DESCRIPTION:

DESCRIPTION
INCLUDE FILE FOR NTM INTERFACE
FORMS DRIVEN FORM EDITOR Include File Description

FILE NAME: RW
PURPOSE: REPORT WRITER DEFINITIONS
LANGUAGE: C

DESCRIPTION:

--------------

DESCRIPTION
FILE NAME: STDTYP
PURPOSE: STANDARD TYPE DEFINITIONS
LANGUAGE: C

DESCRIPTION:

DESCRIPTION

THIS FILE ENSURES THAT THE FOLLOWING STANDARD TYPES ARE AVAILABLE:

FLOAT - SINGLE PRECISION FLOAT
DOUBLE - DOUBLE PRECISION FLOAT

LONG - 32 BIT (OR LARGER) SIGNED INTEGER
LBITS - 32 BITS (OR MORE) FOR BIT MANIPULATION

INT - NATURAL SIZE SIGNED INTEGER
UNSIGNED - NATURAL SIZE UNSIGNED INTEGER
BOOL - NATURAL SIZE LOGICAL (ZERO / NON-ZERO ONLY)

SHORT - 16 BIT (OR LARGER) SIGNED INTEGER
USHORT - 16 BIT (OR LARGER) UNSIGNED INTEGER
BITS - 16 BITS (OR MORE) FOR BIT MANIPULATION

CHAR - SINGLE MACHINE CHARACTER (REAL CHARACTERS ALWAYS POSITIVE)
TINY - 8 BIT (OR LARGER) SIGNED INTEGER
UTINY - 8 BIT (OR LARGER) UNSIGNED INTEGER
TBITS - 8 BITS (OR MORE) FOR BIT MANIPULATION
TBOOL - 8 BIT (OR LARGER) LOGICAL (ZERO / NON-ZERO ONLY)

METACHAR - 16 BIT (OR LARGER) AUGMENTED CHARACTER (SIGNED)

VOID - FUNCTION THAT RETURNS NO VALUE

FORTRAN - STORAGE CLASS FOR FOREIGN (NON-C) ROUTINES OR C ROUTINES WHICH ARE CALLABLE FROM FOREIGN ROUTINES

SINCE NOT ALL COMPILERS SUPPORT USHORT, TINY, AND UTINY, THE FUNCTIONS
USHORT(), TINY(), AND UTINY() SHOULD BE USED WHENEVER REFERENCING THEM.

IN ADDITION, THE FOLLOWING UTILITY MACROS ARE DEFINED:

LURSHIFT(N, B) - UNSIGNED LONG RIGHT SHIFT
MAX(A, B) - MAXIMUM OF A AND B
MIN(A, B) - MINIMUM OF A AND B
ABS(A) - ABSOLUTE VALUE OF A
STRASN(A, B) - TRANSPORTABLE A - B FOR STRUCTURES
NULL - NULL POINTER VALUE (0)
TRUE - 1
FALSE - 0
SUCCESS - EXIT(SUCCESS) INDICATES SUCCESSFUL COMPLETION
FAILURE - EXIT(Failure) INDICATES ERRORS

THE FOLLOWING SYMBOLS SHOULD BE DEFINED BASED ON THE COMPILER BEING USED:
USHORT - COMPILER SUPPORTS UNSIGNED SHORT
TINY - COMPILER TREATS CHAR AS SIGNED
UTINY - CHAR IS SIGNED AND COMPILER SUPPORTS UNSIGNED CHAR
VOID - COMPILER SUPPORTS VOID
FORTRAN - COMPILER SUPPORTS FORTRAN
STRASN - DEFINE APPROPRIATE MACRO
SUCCESS - DEFINE APPROPRIATE VALUE IF NOT 0
FAILURE - DEFINE APPROPRIATE VALUE IF NOT 1
3.10.10 Hierarchy Chart

The following hierarchy charts show the relationships between all of the modules mentioned in the above documentation. A module may call a subroutine several times within its code, but the call will only be shown once as a single relationship on this hierarchy chart. All modules shown at the top of the first page are considered Main Programs as described in section 3.10.1 above.

There is an internal paging scheme as marked by the numbers in the upper right corner of each page. An index after the last page of the chart shows where a routine and its calls are first defined. If a routine has no page reference, it either makes no calls or is an external routine. A continuation box on the end of a tree limb shows where the tree continues on the page numbered mentioned. A number in a box with a routine name points to the page where the routine is further defined within the hierarchy tree. If there is no number in a box, the routine either makes no calls or is an external routine.
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A
PS 620144402
1 November 1985

5

---

FDFE/MAIN
---

INITIAL | MEMCMP | INITFP | FDFE | TERMFP | TRMNAT
---

RMVPAG | VIEW | RENAME | COPY | GETFLS | (CONT)
---

11 | 12+ | 13+
WARNING

<table>
<thead>
<tr>
<th>WARNING</th>
</tr>
</thead>
<tbody>
<tr>
<td>---------</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>---------</td>
</tr>
</tbody>
</table>

PS 620144402
1 November 1985
PS 620144402
1 November 1985

+--------+
| WRTEXP |
+--------+

+--------+ +--------+ +--------+ +--------+ +--------+ +--------+
| FREE   | | WRTEXP | | MEMCPY | | MYALLOC | | STRLEN  | |PRINTF  |
+--------+ +--------+ +--------+ +--------+ +--------+ +--------+

3-283
PS 620144402
1 November 1985

19

+--------+
| GETFLS |
+--------+

+--------+
| (CONT) |
+--------+

+--------+ +--------+ +--------+ +--------+ +--------+ +--------+
| FOPEN  | SYSMSG  | BLN   | MEMSET | (CONT) |
| 12      | 12       | 12    | 12     | 12     |

3-287
PS 620144402
1 November 1985

+-----+
| ADDEXT |
+-----+

+-----------------------+
+--+--+
+---+---+ +--+---+ +---+---+
IESCPY ISTRRCHR ISTRCAT ISPRINTF

+-------+
|        |
|        |
+-------+

3-293
IPRSCMD

ISTRNCPY | IMEMCMP | IMEMSET | STRNCMP | STRLEN

+-+---+-  +--+-+  +----+-  +-----+  +-----+
PS 620144402
1 November 1985

34

+-------+
| LAYOUT |
+-------+

+--------+
| SCRMAN | MEMCMP | EDTFLD |
+--------+--------+--------+

+--------+
| SCRMAN/CHGPOS | SCRMAN/GETROW | TRNSCR | (CONT) |
+--------+--------+--------+--------+

+--------+
| ESCPY | STRCHR | NATOI | STRLEN |
+--------+--------+--------+--------+

3-302
PS 620144402
1 November 1985

49

+---------+
| SAVFLS |
+---------+

| (CONT) | REWIND | FCLOSE | RENAME | WRTFRM | DELFLD |
+--------+--------+--------+--------+--------+--------+

3-317
PS 620144402
1 November 1985

61

--------+
| SCRMAN1 |
+--------+

| (CONT) | PMSGLC | PDATA |
+--------+--------+--------+

+---52+ +-----+ +-----+
### Table 1

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDTFLD</td>
<td>I</td>
</tr>
<tr>
<td>(CONT)</td>
<td>GFDINP</td>
</tr>
<tr>
<td>MODFLD</td>
<td>INSFLD</td>
</tr>
<tr>
<td>MODFRM</td>
<td>(CONT)</td>
</tr>
<tr>
<td>53</td>
<td>74</td>
</tr>
<tr>
<td>75</td>
<td>76</td>
</tr>
<tr>
<td>77</td>
<td></td>
</tr>
</tbody>
</table>

### Table 2

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMSGLS</td>
<td>DELFLD</td>
</tr>
<tr>
<td>INSFLD</td>
<td>FREE</td>
</tr>
<tr>
<td>MATOI</td>
<td>(CONT)</td>
</tr>
<tr>
<td>75</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td></td>
</tr>
</tbody>
</table>

---

3-334
---+
| MODWHL |
---+

| MODFLD | MEMCMP | SPRINTF | PMSGLS |
---+       +----+      +----+      +----|
| 66+     +----+      +----+      +----|

3-349
PS 620144402
1 November 1985

98

+---------------------+
| VALINP/CCKFLD |
+---------------------+

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| MATOI | VALINP/CCKPRM | MIN | ATOI | (CONT) |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| MEMCMP |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
PS 620144402
1 November 1985

105

---
I VALIMP/CCKFLD
------

---
I (CONT)
----
SPRINTF
MENCMP
FNDATT
STRCHR
( CONT )
---98+-
---+-
-+-
-+-
-+-
-+-
-+-
-+-
3-373
<table>
<thead>
<tr>
<th>ABORT</th>
<th>GETCUR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS</td>
<td>GETFLS</td>
</tr>
<tr>
<td>ACCESS</td>
<td>GETFLS/TREEXP</td>
</tr>
<tr>
<td>ADDCHK</td>
<td>GETLEN</td>
</tr>
<tr>
<td>ADDEXT</td>
<td>GFDINP</td>
</tr>
<tr>
<td>ADDFRM</td>
<td>GFLDPT</td>
</tr>
<tr>
<td>ATOI</td>
<td>GITMD</td>
</tr>
<tr>
<td>BLEN</td>
<td>GNXTFD</td>
</tr>
<tr>
<td>CHKARY</td>
<td>GNXTFD/NXTFLD</td>
</tr>
<tr>
<td>CHKFLD</td>
<td>GTCPFD</td>
</tr>
<tr>
<td>CHKFRM</td>
<td>GTFDtx</td>
</tr>
<tr>
<td>CHKPRM</td>
<td>GTFDtx/GTXINF</td>
</tr>
<tr>
<td>CLSPFRM</td>
<td>GTNMFd</td>
</tr>
<tr>
<td>COPFLD</td>
<td>GWHINP</td>
</tr>
<tr>
<td>COPY</td>
<td>GWINDO</td>
</tr>
<tr>
<td>CPYFRM</td>
<td>INITIAL</td>
</tr>
<tr>
<td>CSTASH</td>
<td>INITFP</td>
</tr>
<tr>
<td>DELFLD</td>
<td>INSFld</td>
</tr>
<tr>
<td>DRPFRM</td>
<td>INSFRM</td>
</tr>
<tr>
<td>DRPWHL</td>
<td>INSWHL</td>
</tr>
<tr>
<td>EDTFLD</td>
<td>ISALPHA</td>
</tr>
<tr>
<td>EDTMOD</td>
<td>ISSPACE</td>
</tr>
<tr>
<td>EDTWHL</td>
<td>LAYOUT</td>
</tr>
<tr>
<td>ERROR</td>
<td>LISTFM</td>
</tr>
<tr>
<td>ESCPY</td>
<td>LISTIT</td>
</tr>
<tr>
<td>EXPAND</td>
<td>LAYOUT</td>
</tr>
<tr>
<td>EXPAND/FIXFRM</td>
<td>LAYOUT</td>
</tr>
<tr>
<td>FATAL</td>
<td>LAYOUT</td>
</tr>
<tr>
<td>FCLOSE</td>
<td>LAYOUT</td>
</tr>
<tr>
<td>FDFE</td>
<td>LAYOUT</td>
</tr>
<tr>
<td>FDFE/MAIN</td>
<td>LAYOUT</td>
</tr>
<tr>
<td>FEOF</td>
<td>MAX</td>
</tr>
<tr>
<td>FERROR</td>
<td>MAX</td>
</tr>
<tr>
<td>FGETS</td>
<td>MAX</td>
</tr>
<tr>
<td>FIFDST</td>
<td>MEMCMP</td>
</tr>
<tr>
<td>FLANCi</td>
<td>MEMCPY</td>
</tr>
<tr>
<td>FLDTYP</td>
<td>MEMDGT</td>
</tr>
<tr>
<td>FLFMST</td>
<td>MEMSET</td>
</tr>
<tr>
<td>FLSTRC</td>
<td>MIN</td>
</tr>
<tr>
<td>FLWHRST</td>
<td>MITOA</td>
</tr>
<tr>
<td>FMDATT</td>
<td>MKPOS</td>
</tr>
<tr>
<td>FOPEN</td>
<td>MKPOS</td>
</tr>
<tr>
<td>FPRINTF</td>
<td>MKPOS</td>
</tr>
<tr>
<td>FREBUF</td>
<td>MKPOS</td>
</tr>
<tr>
<td>FREE</td>
<td>MKPOS</td>
</tr>
<tr>
<td>GATDEF</td>
<td>MKPOS</td>
</tr>
<tr>
<td>GDATA</td>
<td>MKPOS</td>
</tr>
</tbody>
</table>

3-377
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMSGLS</td>
<td>VALNP/CCKFRM</td>
<td>86</td>
</tr>
<tr>
<td>PROFIL</td>
<td>VALNP/CCKHLP</td>
<td>73</td>
</tr>
<tr>
<td>PREC</td>
<td>VALNP/CCKITM</td>
<td>72</td>
</tr>
<tr>
<td>PRSCMD</td>
<td>VALNP/CCKNAM</td>
<td>87</td>
</tr>
<tr>
<td>PUTATT</td>
<td>VALNP/CCKPRM</td>
<td>98</td>
</tr>
<tr>
<td>PUTCUR</td>
<td>VALNP/CCKRSV</td>
<td>107</td>
</tr>
<tr>
<td>PUTERR</td>
<td>VIEW</td>
<td>11</td>
</tr>
<tr>
<td>RENAME</td>
<td>WARNING</td>
<td>8</td>
</tr>
<tr>
<td>REWIND</td>
<td>WRTEXP</td>
<td>15</td>
</tr>
<tr>
<td>RMVPAG</td>
<td>WRTFDL</td>
<td>48</td>
</tr>
<tr>
<td>RSVATT</td>
<td>WRTFDL/ARYREF</td>
<td>58</td>
</tr>
<tr>
<td>SAVFLS</td>
<td>WRTFRM</td>
<td></td>
</tr>
<tr>
<td>SCRMAN</td>
<td>YYPARSE</td>
<td></td>
</tr>
<tr>
<td>SCRMAN/CHGPOS</td>
<td></td>
<td>59</td>
</tr>
<tr>
<td>SCRMAN/GETROW</td>
<td></td>
<td>34</td>
</tr>
<tr>
<td>SPRINTF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRASN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRCAT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRCHR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRCMP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRCPY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRLEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRNCHP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRNPC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRRCHR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRSPN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRUPC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYMSMG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYSTEM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TERMFP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRMNAT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRNSCR</td>
<td></td>
<td>40</td>
</tr>
<tr>
<td>TRNSCR/F'~ST</td>
<td></td>
<td>51</td>
</tr>
<tr>
<td>TRNSCR/FRLCST</td>
<td></td>
<td>84</td>
</tr>
<tr>
<td>TRNSCR/GRMPMT</td>
<td></td>
<td>70</td>
</tr>
<tr>
<td>TRNSCR/GTPINF</td>
<td></td>
<td>83</td>
</tr>
<tr>
<td>TRNSCR/LDPMINF</td>
<td></td>
<td>40</td>
</tr>
<tr>
<td>TRNSCR/MTCHPMT</td>
<td></td>
<td>84</td>
</tr>
<tr>
<td>TRNSCR/PARSCRN</td>
<td></td>
<td>59</td>
</tr>
<tr>
<td>TRNSCR/SPSYMB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRNSTR</td>
<td></td>
<td>52</td>
</tr>
<tr>
<td>TRNSTR/FLFLD</td>
<td></td>
<td>62</td>
</tr>
<tr>
<td>TRNSTR/FLPRMPT</td>
<td></td>
<td>52</td>
</tr>
<tr>
<td>TRNSTR/GARINF</td>
<td></td>
<td>71</td>
</tr>
<tr>
<td>UNLINK</td>
<td>VALNP</td>
<td>63</td>
</tr>
<tr>
<td>VALNP/CCKFLD</td>
<td></td>
<td>98</td>
</tr>
</tbody>
</table>
3.11 Program Listings Comments

This information is contained in the Module Descriptions in section 3.10.
SECTION 4
QUALITY ASSURANCE PROVISIONS

4.1 Introduction and Definitions

"Testing" is a systematic process that may be preplanned and explicitly stated. Test techniques and procedures may be defined in advance, and a sequence of test steps may be specified. "Debugging" is the process of isolation and correction of the cause of an error.

"Antibugging" is defined as the philosophy of writing programs in such a way as to make bugs less likely to occur and when they do occur, to make them more noticeable to the programmer and the user. In other words, as much error checking as is practical and possible in each routine should be performed.

4.2 Computer Programming Test and Evaluation

The quality assurance provisions for test consists of the normal testing techniques that are accomplished during the construction process. They consist of design and code walk-throughs, unit testing, and integration testing. These tests are performed by the design team. Structured design, design walk-through and the incorporation of "antibugging" facilitate this testing by exposing and addressing problem areas before they become coded "bugs."
END
0-87
DTIC