

# **RPW-CODES**

RECEIVED

SEP 24 5 29 PM '01

POSTAL RATE COMMISSION  
OFFICE OF THE SECRETARY

## **Preface**

This is a Category 1 Library Reference. It updates and replaces USPS-LR-I-35/R2000-1, USPS-LR-I-36/2000-1, USPS-LR-I-38/2000-1, and USPS-LR-I-39/2000-1.

It provides documentation of the Computerized On-site Data Entry System for RPW, which is a subject of the testimony of witness Pafford, USPS-T-3.

**RPW-CODES**

**Computer System  
Documentation Description  
With  
Source Code on CD-ROM**

**USPS-LR-J-22/R2001-1**

# Table of Contents

This Library Reference, USPS-LR-J-22/R2001-1, updates USPS-LR-I-35/R2000-1,  
LR-I-36/R2000-1, LR-I-38/R2000-1, and LR-I-39/R2000-1.

|   |     |
|---|-----|
| Table of Contents . . . . .                                       | 2   |
| Introduction . . . . .  | 3   |
| Section 1            RPW CODES DATA ENTRY TECHNICAL DOCUMENTATION |     |
| Introduction . . . . .  | 8   |
| System Narratives. . . . .  | 9   |
| File Layouts. . . . .   | 16  |
| Program Flowchart . . . . .                                       | 67  |
| Section 2            RPW CODES BASE UNIT TECHNICAL DOCUMENTATION  |     |
| Introduction . . . . .  | 70  |
| System Narratives. . . . .  | 72  |
| File Layouts. . . . .   | 74  |
| Program Flowchart . . . . .                                       | 88  |
| Section 3            RPW CODES MAINFRAME TECHNICAL DOCUMENTATION  |     |
| Introduction . . . . .  | 91  |
| System Narratives . . . . .                                       | 92  |
| Program Flowcharts . . . . .                                      | 124 |
| JCLs . . . . .  | 144 |
| Section 4            CONTENTS OF CD-ROM WITH SOURCE CODE. . .     | 181 |

## **INTRODUCTION**

This Library Reference, USPS-LR-J-22/R2001-1, documents the RPW-CODES computer system. It combines all RPW-CODES System documentation and computer source code into this single library reference. Consequently, separate R2000-1 RPW-CODES library references such as USPS-LR-I-35, USPS-LR-I-36, USPS-LR-I-38, and USPS-LR-I-39 are being replaced by this consolidated approach.

The Revenue, Pieces, and Weight (RPW) program allows ~~to~~ collection of data on revenue, pieces, and weight of mail by class and subclass of mail, and fees by type of service, as part of the Postal Service's Cost Revenue Analysis System.

Sampling units selected for testing are randomly chosen from the ORFEO System on a quarterly basis. The design of the RPW system is to test both incoming and originating units.

Revenue, Pieces, and Weight data are required by the Postal Reorganization Act. Without such data, the Postal Rate Commission could not estimate unit costs and revenue by category, which is essential for developing fair and equitable postage rates. In addition, RPW data are used to develop the revenue unit measure underlying the CAG classification of post offices.

The Controller and the Planning Department use RPW data to advise senior management on budgeting and planning issues including productivity measurement. This data is also used to:

1. Plan the Postal Service budget, based on forecasts of the mail volume, workloads, and overall productivity.
2. Monitor productivity increases associated with automation programs
3. Assess deviations of actual volume from projected volume
4. Analyze other major postal service activities affecting costs and revenue.

This technical documentation for RPW application is based on the above application as it was in PQ3 2000.

The RPW-CODES system is deployed on three platforms: laptop, base unit, and mainframe.

Source code for RPW-CODES application in PQ3 2000 complements this technical documentation.

**DATA ENTRY:**

RPW Data Entry application is written in 'C++' and consists of the following database files:

**Look-up Files:**

BUCKET.1ST  
BUCKET.3RD  
BUCKET.4TH  
CLIMIT.DAT  
DBMCZIP.DAT  
INCOMING.DAT  
ORIGCOD.DAT  
ORIGINS.DAT  
ORIGREG.DAT  
RATES.DAT  
SSCATCOD.DAT  
AGENCY.FIL  
CATCODES  
BUCKET.INT  
STATE.TXT

**Sample files**

RpwSampl.dat  
RpwSampl.ndx

**Output Files**

SESSION.RPW  
DATAFILE.RPW  
EDITFILE.RPW

## **BASE UNIT**

RPW Base Unit application is written in 'Clipper' and consists of the following database files:

AUDITNEW.DBF  
CATCODE.DBF  
NEWRSMPL.DBF  
RPW.DBF  
RPWADMIN.DBF  
RPWAUDIT.DBF  
RPWNEW.DBF  
RSAMPLE.DBF  
SSCATCOD.DBF

## **MAINFRAME**

The RPW Mainframe application consists of the Source Programs, Control Files, and Proc files(JCLs).

The Control files are as follows:

HSD85001  
HSD85002  
HSD85003  
HSD85004  
HSD85005

The Source programs are as follows:

HSD830c3  
HSD840ca  
HSD841c1  
HSD850c5  
HSD860c3

The Proc (JCLs) programs are included in this documentation and are listed below.

HSD5000s  
HSD8300q  
HSD8350t  
HSD8500t

The CD with the source code for PQ3 2000 software release complements this technical documentation.

# **RPW-CODES**

## **Computer System Documentation Description**

### **Section 1**

#### **Data Entry System**

## **INTRODUCTION**

The CODES RPW Data Entry Sybsystem has been developed to facilitate the field (on-site) data collection of statistical data concerning volume, revenue, and weight of first class, priority, third class, and fourth class domestic and international mail. Tests are conducted according to a schedule, called the Sample Selection file, which is produced by the Mainframe CODES/RPW Domestic Interface Sybsystem. The Sample Selection file is downloaded to Base Units at Management Sectional Centers via SNA Exchange. RPW tests are performed by extracting specified mailpieces from the processing stream and entering their characteristics as prompted by the RPW data collection software. The software acts like an expert system to guide the user through the data entry process. The software is executed on the Base Unit IBM Ats or on Grid Systems portable computers. An electronic scale may be attached to the portable computers to aid in the collection of weight information. Data is transferred to the Base Units by either physical diskette copy or asynchronous transmission through software provided by Ascom. After processing by Base Unit systems it is uploaded on a weekly basis, to the mainframe at the National Information System Development Center in Raleigh, North Carolina.

The Data Entry section contains file narratives, file layouts, and flowcharts that illustrate RPW process flow.

Source code for PQ3 2000 complements this technical documentation.

## **RPW DATA ENTRY NARRATIVES**

**EDITFILE.RPW** - contains records deleted from the RPW data file.  
Format: Edit.file

**DATAFILE.RPW** - file, containing mailpiece and special service data records. This file is output by the Rpwentry.bas module.

**SESSION.RPW** - file which keeps track of test information, session numbers, the beginning and ending record numbers per session, and sample and total mailpiece weight if the session involved subsampling.

**AGENCY.FIL** - database of Federal Agency codes and corresponding agency name.

**ORIGINS.DAT** - table of mail category codes and all possable special service category codes for that mailpiece category.

**SSCATCOD.DAT** - table of special service category codes, locations in the rate table corresponding to these special services, and test containing the name of the special service or combination of services.

**BUCKET** files - are files for the catcod look-up.

**BUCKET.1ST** - file for the 1<sup>st</sup> class mail.

**BUCKET.3RD** - file for the standard mail.

**BUCKET.4TH** - file for the service package.

**CLIMIT.DAT** - files containing rate limits for some special mail categiries.

**INCOMING.DAT** - file for processing special service mail.

**ORIGCOD.DAT** - file for the COD mail.

**ORIGINS.DAT** - file for the insured mail.

**ORIGREG.DAT** - file for the registered mail.

RATES.DAT - file, that contains rate table.

BUCKET.INT - file for processing the international mail.

STATE.TXT - file that contains information about all US states.

RPWSAMPL.DAT - datafile.

RPWSAMPL.NDX - index for sample file.

**Functions available from the subfunctions menu:**

1. Input mailpiece data

Calls the data entry routine. This is the main routine found in the rpwentry.c module. All mailpiece and special service data can now be recorded. If a test is being sub-sampled the user can enter subsampling type and skip interval prior to inputting mailpiece data.

2. Review within a session

Review/delete RPW data records for the active session.

3. Abort a session

Place an 'A' in the last byte of the active session's data records. Does not write a session record. Return to main menu.

4. End session and save

Update the session file and return to the main menu.

### Rpwentr1.c and RDwentr2.c Modules

Rpwentr1.c and RDwentr2.c are the largest modules. All of the actual mailpiece data entry takes place in them. The main controlling function is called data entry(int \*routine number).

After turning off the cursor the data entry form screen displays. Data Entry consists of a switch statement inside of a loop. Substatus is an integer, which is the focus of the case statements inside the switch. It determines which routine to call next. Most of the routines called as a result of a particular case statement involve the display of a new data entry menu in the 'data entry' window in the upper right corner of the screen. After a menu choice is made, the appropriate subordinate routines are called, the substatus variable is reset, and a break or return statement forces a return to the case statement. A break statement forces processing to the top of the main loop, and then drops, according to the new value of substatus, to the appropriate case statement and corresponding next routine.

It is important to understand how several of the subsystems within Rpwentr\*.c work. As an example, the logic behind the input of a first class letter is as follows:

I. Sub\_status is initially 0, therefore case 0 is recognized by the main switch logic. Pop list1, the first data entry menu, is called.

1. Num input is called. This routine gets a keystroke.
2. The user chooses '1. First Class'.
3. 'First Class' is displayed in the mailpiece summary window.
4. Byte 0 of the five-byte buffer named 'class' is turned on by placing a '1' in it. This denotes first class.
5. Substatus is set to 1.
6. Control is returned to the main switch statement.

II. Since substatus is now 1, case I is recognized and Pop list2 is called.

1. Num input is again called to get a response from the user.
2. The user chooses '1. Single Piece'.

3. 'Single piece' is displayed in the mailpiece summary window, under 'First class'.
4. Byte 3 of the buffer named 'bucket1' is turned on by placing a '1' in it. This denotes single piece. See the following section concerning 'byte buckets'.
5. Substatus is set to 2.
6. Control is returned to the main switch statement.

III. The same sort of logic continues as the mailpiece is identified to the system. Further inputs required include subclass, number of pieces, weight, special services, indicia, whether private, government, or USPS, standard or not, automation compatible or not, revenue from enclosures or not, etc.

Along the way the category code, proper revenue, and any special services for the mailpiece are determined. How these things are accomplished is discussed below. When input is complete the mailpiece will have been summarized on the screen in the 'mailpiece summary' window. Special services will appear in the 'Special service summary' window. Proper revenue will be listed in the 'Revenue Summary' window. The category code will appear below the revenue summary window. The user can then choose to accept the mailpiece as shown, erase the screen and start over, escape back to the submenu by returning to the calling program (PWENTRY.C), or indicate that the actual revenue on the mailpiece differs from the suggested revenue. At this point he can input the 'other' revenue and the program will compute the revenue variance and display it in the revenue summary window.

#### How to determine the category code

Each class of mail, certificates of mailing, and international mail has an area of memory called a 'byte bucket' associated with it. The byte bucket for first class is 11 bytes long. Each byte position is significant. Letter, card, single piece, 5 digit presort, local or non-local, and standard or non-standard are examples of things represented by a particular byte location. As the mailpiece is 'built' on the screen the correct bytes in the bucket are turned on by placing a '1' in them. When the mailpiece is finished a unique representation of 1s and 0s has been created.

External bucket files, which are master lists of valid bucket combinations, have been read into arrays of structures at the beginning of the RPW.C module. For example, Bucket.lst contains all first class buckets. It

was read into the extern struct bucket1 \*bucket 1st array. The newly created bucket is compared sequentially with the first 11 bytes of the array structures until a match is found. The last four bytes of the matched array structure is the category code.

#### How to determine the correct rate

Rate lookups are base on category code. The file 'rates.dat' was read into a two-dimensional integer array at the beginning of the Rpwentry module. The routine void revenue (mt \*substatus) is called to perform the actual lookup. It is separated into sections based on mail class. Each section contains the proper formulas and rate table coordinates to calculate the revenue for the mailpiece. Some formulas are based on weight per piece and some (4th class parcels, for example) on zones. Revenue is stored in the extern long mt mail rate.

#### Determining Special Services

For most mailpieces there are a number of special services which can be purchased. These include certified, special handling, return receipt, special delivery, etc. Several tables have been developed to simplify the task of determining which special services are appropriate for a given mailpiece. They are incoming.dat (incoming mail tests), origreg.dat (originating registered test), origins.dat (originating insured test), and origcod.dat (originating COD test). These tables are read into arrays of structures at the top of the Rpw.c module. Each structure in the array is separated into 4 byte elements, each containing a category code. The first element is a normal mailpiece category code such as 1100, for first class letter. The remaining elements in the structure are special service category codes which may be associated with category 1100 mail. The logic to accomplish this table lookup is in a module entitled Rpwss.c. The routine get ss(int \*substatus, mt\_element) is called from the rpwentry.c module. According to the test class stored as tclass, each special service category code is read and a further validation is performed to select those special services which are allowed for the weight of the mailpiece. Those special service category codes passing this test are stored in an array called ssnum array. According to the number of special services allowed, a window size to display them is determined and memory is allocated to save that portion of the data entry screen. A blank window is popped. It is filled by doing a sscatcod array, finding the code, extracting the description of the special service from the array element, and displaying it in the window. The sscatcod array element number is stored in an array called ss\_element. A light bar is provided to enable the user to choose a special service. When the user presses the

carriage return the special service rate is calculated (see next section), the special service description, rate, and category code are stored in an array called ss fields array, the data entry screen is restored, the substatus variable is set, the special service rate is displayed in the revenue summary window, and control is returned to the rpwentry.c module.

#### Determining Special Service Revenue

Certain ss category codes represent combinations of special services, up to three. Each structure in the sscatcod array has 6 elements following the ss category code. These are row and column coordinates in the rate table for the three possible rates. If a row, column pair contains blanks then there must have been less than three special services in the combination. An ss category code might have only one rate lookup required. In this case the second and third row/column pair will be blank. The rate (or total if more than one rate) is stored in the extern variable ss rate and saved in the ss fields array mentioned above.

RPWLIC.C contains several routines, which are used by more than one of the other modules.

Rpwcomm.c is the module, which contains the code to communicate with the electronic scale. It is an interrupt driven routine written using the Greanleaf Comm Library. An interrogation string, 'SGW' is sent to the scale. If the scale has not responded in 5 seconds a timeout occurs and the user is prompted for the weight. If the scale responds properly it will send the weight of the piece(s), in ounces carried to 1 decimal place, followed by a carriage return/linefeed combination.

Rpwvideo.c contains all routines, which pertain to the handling of the screen. Video.h is the header file included in this module. It contains the structure video info, which is used to store attributes, cursor scan lines, and the display segment address for the monitor in use. One of the routines in this module is mit video(void), which looks at the monitor type and loads the video info structure accordingly. Other routines write strings and buffers to the screen at row, column locations with attribute, save portions of screens, pop windows and screens, restore screens, and handle the cursor.

Rpwss.c is the module containing the code to display the available special services for a particular mailpiece, get the user selection, determine the special service category code and rate, and then restore the data entry screen and return to the calling module (rpwentry.c). See the section on determining special services and rates.

Rpwrevnuc is the module, which performs the lookup for all rates in RATES.DAT. For first class and third class, where the rate structure is complicated by various sortation and entry level discounts, the rates are displayed in a 'pick list' in a window on the screen. For other categories, the rates are merely calculated and displayed.

#### Parent and Child Records

In the RPW datafile, slightly different record formats are used to distinguish basic mailpiece information from special service information. A 'parent' record contains information pertaining to the mailpiece itself, exclusive of special services like insurance, registered return receipt, special handling, etc. 'Child', or special service records immediately follow a parent record. There may be more than one. Child records are identical to their parent except that a Special Service Category Code is placed in the sscatcod field of the data record. The field called sspointer is used to relate the parent and child records. If a parent record has no children, this field is blank. If one or more child records exist then the ss\_pointer field in the parent record is '1' and the succeeding child records are numbered '2' through whatever. This linkage is very important in downstream processing. The special service category code will only appear in child records. These codes can be found in the file named Sscatcod.dat.

## File Layouts

```
char finance      [6];
char testid       [6];
char testdate     [5];
char testyear     [4];
} sel_sample;

DATAFILE.RPW
char finance      [6];
char testid       [6];
char testdate     [8];
char version      [2];
char time         [4];
char blowup       [4];
char category     [4];
char pieces        [4];
char revenue      [9];
char pounds        [3];
char ounces        [4];
char agency        [3];
char sscatcod     [4];
char zip          [3];
char zone          [1];
char shortpaid    [5];
char overpaid      [5];
char indicia       [1];
char auto_compat   [1];
char shape          [1];
char length        [3];
char width          [2];
char height         [2];
char girth          [2];
char sub_method    [1];
char ss_pointer    [2];
char session       [2];
char record         [4];
char user_id       [3];
char edit_flag      [1];
char pos_one_Terminal [1];
char cr            [1];
char lf            [1];
} data_record;           // 98 character record + cr/lf

SESSION.RPW
char finance      [6];
char testid       [6];
char date          [8];
char user_id       [3];
char begin_time   [4];
char end_time      [4];
char begin_rec     [4];
char end_rec       [4];
char total_wt      [8];
char sample_wt     [8];
```

```

    char valid_byte [1];
    char cr [1];
    char lf [1];
} session_record;      // 56 character record + cr/lf

agency.fil
    char agency_code [3];
    char agency_name [40];
};      // 43 character record + cr/lf

struct bucket1
{
    char bucket      [12];
    char catcode     [4];
};      // 16 character record + cr/lf
*/
}

struct bucket3
{
    char bucket      [23];
    char catcode     [4];
};      // 27 character record + cr/lf

struct bucket4
{
    char bucket      [13];
    char catcode     [4];
};      // 17 character record + cr/lf

struct bucketf
{
    char bucket      [8];
    char catcode     [4];
};      // 12 character record + cr/lf

struct bucketcrt      // certificates of mailing
{
    char bucket      [12];
    char catcode     [4];
};      // 16 character record + cr/lf
*/
/*
struct bucketc
{
    char catcode     [4];
    char text       [64];
    char cr         [1];
    char lf         [1];
};      // 68 character record + cr/lf
*/
struct incoming
{
    char catcode     [4];
    char sscat [36][4];
};      // 148 character record + cr/lf
*/

```

```

struct origcod
{
    char catcode [4];
    char sscat [9][4];
};      // 40 character record + cr/lf

struct origins
{
    char catcode [4];
    char sscat [15][4];
};      // 44 character record + cr/lf

struct origreg
{
    char catcode [4];
    char sscat [12][4];
};      // 52 character record + cr/lf
*/
}

struct
{
    char fld0 [4];
    char fld1 [4];
    char fld2 [4];
    char fld3 [4];
    char fld4 [4];
    char fld5 [4];
    char fld6 [4];
    char fld7 [4];
    char fld8 [4];
    char fld9 [4];
    char crlf [2];
} rate;      // 40 character record + cr/lf

struct sscatcod
{
    char catcode [4];
    char rate1_row [3];
    char rate1_col [1];
    char rate2_row [3];
    char rate2_col [1];
    char rate3_row [3];
    char rate3_col [1];
    char text [64];
};      // 80 character record + cr/lf

struct sample_rec
{
    char ste [3];
    char finance [6];
    char mainzip [5];
    char facilzip [5];
    char meptype [2];
    char hqcode [4];
    char filler1 [3];
    char bacode [2];
    char odisarea [3];
}

```

```

char testid      [6];
char testyear    [4];
char testmon     [2];
char testday     [2];
char strata      [3];
char facilnam   [30];
char cag         [1];
char mepdescr   [80];
char mepzip      [5];
char cell        [3];
char filler2     [3];
char mainoffc   [30];
char csd         [3];
char filler3     [45];
char cr          [1];
char lf          [1];
} ;      // 250 character record + cr/lf
*/
}

struct
{
    char zip      [3];
    char zones   [1998];
    char cr      [1];
    char lf      [1];
} zone;      // 2001 character record + cr/lf

struct ss_fields
{
    long ss_rate;
    char ss_category [4];
    long ss_shortpaid;
    char ss_text [64];
};      // 72 character record

struct index {      // sample selection file index
    char finance [6];
    char testid [6];
    char testyear [4];
    char testmon [2];
    char testday [2];
    long offset;
};      // 24 character record

struct rand_start {
    char testid [6];
    char letter_skip_info [4];    // [x] = number of rows in table
    char parcel_skip_info [5];
    char other_skip_info [6];
    char orig_skip_info [5];
    char override_skip_info [6];
    char container_range_skip_info [10];
    char container_pieces_skip_info [9];
};

struct dbmc {      // DBMC parcel zone table
    char pozip [3];

```

```
    char bmczip [3];
};      // 6 character record + cr/lf

struct blank_line /* jim */
{
    char line [81];
};
```

```

Function Caller/Called Summary Tree
*****
[ 86 RPWENTR2.C      1 :::
[ 142 RPWVIDEO.C    2 |->prtbuff
[ 120 RPWVIDEO.C    3 |->prtstr
[           | 4 |..strlen
[ 293 RPWENTR2.C    5 |->write_record
[           | 6 | |->handle_error2
[ 120 RPWVIDEO.C    7 | | |->prtstr { 3 }
[ 1235 RPWENTRY.C   8 | | | |->exit_routine
[ 140 CLIMIT.C      9 | | | | |->free_code_limit
[           | 10 | | | | |..free
[ 93 LOOKUP.C        11 | | | | |->close_zip_lookup
[ 83 STATECOD.C     12 | | | | | |->close_state_file
[           | 13 | | | | | |..free
[ 88 CITYZIP.C       14 | | | | | |->close_ziphelffile
[ 29 CITYZIP.C       15 | | | | | | |->ptr_ziphelffile
[           | 16 | | | | | | |..free hfree close
[ 212 RPWVIDEO.C    17 | | | | |->setcur
[           | 18 | | | | | |..int86
[ 78 RPWVIDEO.C     19 | | | | |->clear_scr
[           | 20 | | | | | |..int86
[ 211 ASIPORTS.H    21 | | | | |->exit
[           | 22 | | | | | |.._asiexit
[           | 23 | | | | | |..free _asiexit
[           | 24 | | | | | |..printf getch
[ 156 MAINRATE.C    25 | | | | |->edit_rate
[           | 26 | | | | | |..strcpy ltoa strlen
[ 723 RPWLIB.C       27 | | | | |->date_time
[           | 28 | | | | | |..time localtime strcpy asctime
[           | 29 | | | | | |..strcpy strcat fopen memset itoa strlen memcpy
[           | | | | |..memcmp
[ 83 RPWLIB.C        30 | | | | |->handle_error
[           | 31 | | | | |->clear_message
[           | | | |

```

```

316 RPWVIDEO.C      32 | | |->clr_box
120 RPWVIDEO.C      33 | | |->prtstr { 3 }
120 RPWVIDEO.C      34 | | ..printf strlen getch
1448 RPWENTR1.C    35 | |->clr_entry_scrn
316 RPWVIDEO.C    36 | | |->clr_box { 32 }
120 RPWVIDEO.C    37 | | |->prtstr { 3 }
120 RPWVIDEO.C    38 | | ..itoa memcpy
3332 RPWENTR2.C   39 | |->repeat
3332 RPWENTR2.C   40 | | ..memset
630 RPWENTR2.C    41 | |->display_record_count
120 RPWVIDEO.C    42 | | |->prtstr { 3 }
120 RPWVIDEO.C    43 | | ..itoa
60 RATEREV.C       44 | |->raterev
201 RPWVIDEO.C    45 | | |->cursor_off
201 RPWVIDEO.C    46 | | | ..int86
106 RATEREV.C      47 | | |->save_scrn
131 RATEREV.C      48 | | | |->handle_error3
159 RATEREV.C      49 | | | | |->box
176 RATEREV.C      50 | | | | | |->boxtop
1070 RATEREV.C     51 | | | | | | |->gotoxy
1070 RATEREV.C     52 | | | | | | | ..int86
1550 RATEREV.C     53 | | | | | | |->writeca
1550 RATEREV.C     54 | | | | | | | ..int86
1070 RATEREV.C     55 | | | | | | |->gotoxy { 51 }
1550 RATEREV.C     56 | | | | | | |->writeca { 53 }
1070 RATEREV.C     57 | | | | | | |->gotoxy { 51 }
1550 RATEREV.C     58 | | | | | | |->writeca { 53 }
120 RPWVIDEO.C     59 | | | | |->prtstr { 3 }
262 RPWVIDEO.C     60 | | | | |->shade_it
127 MAINRATE.C     61 | | | | |->mscgetch
127 MAINRATE.C     62 | | | | | ..int86
127 MAINRATE.C     63 | | | | ..strlen printf
127 MAINRATE.C     64 | | | | ..malloc
1180 RATEREV.C     65 | | |->prep_exit
163 RPWVIDEO.C     66 | | | |->pop_scrn
163 RPWVIDEO.C     67 | | | | ..free

```

```

1082 RATEREV.C      68 | |->load_catfile
131 RATEREV.C       69 | | |->handle_error3 { 48 }
                      70 | | |..._dos_open filelength malloc _dos_read
                      | | |..._dos_close
                      |
78 RPWVIDEO.C       71 | |->clear_scr { 19 }
                      |
1204 RATEREV.C     72 | |->process_menu
                      |
1149 RATEREV.C     73 | | |->menubak
159 RATEREV.C      74 | | | |->box { 49 }
1070 RATEREV.C     75 | | | |->gotoxy { 51 }
120 RPWVIDEO.C     76 | | | |->prtstr { 3 }
1550 RATEREV.C     77 | | | |->writeca { 53 }
                      78 | | | ..strlen
                      |
120 RPWVIDEO.C     79 | | |->prtstr { 3 }
                      |
403 RATEREV.C      80 | | |->getmenukey
                      81 | | | ..._bios_keybrd toupper
                      |
82 |.. :: strncmp strcpy ::::prtstr ::::clear_DE_box
|.. ::::strcpy ::::Disp_Text ::::num_input itoa
memcpy
                      |..atoi printf
                      |
83 RPWENTR1.C       83 ::

2668 RPWENTR1.C     84 |->usps_govt
                      |
149 RPWLlib.C       85 | |->clear_DE_box
316 RPWVIDEO.C     86 | | |->clr_box { 32 }
                      |
120 RPWVIDEO.C     87 | |->prtstr { 3 }
142 RPWVIDEO.C     88 | |->prtbuff { 2 }
                      |
484 RPWMENU.C      89 | |->Disp_Text
                      |
392 RPWMENU.C      90 | | |->Init_Text_Display
                      |
404 RPWMENU.C      91 | | | |->Search_Text_Array
404 RPWMENU.C      92 | | | | |->Init_Text_Display { RECURSVE }
                      93 | | | | ..min strlen memcmp
                      |
404 RPWMENU.C      94 | | |->Search_Text_Array { 91 }
120 RPWVIDEO.C     95 | | |->prtstr { 3 }
                      96 | | |..atoi
                      |
167 RPWLlib.C       97 | |->num_input
723 RPWLlib.C       98 | | |->date_time { 27 }
120 RPWVIDEO.C     99 | | |->prtstr { 3 }
113 RPWLlib.C      100 | | |->clear_message { 31 }
                      101 | | |..kbhit getch printf
                      |
630 RPWENTR2.C     102 | |->display_record_count { 41 }
83 RPWLlib.C        103 | |->handle_error { 30 }
                      |

```

```

452 RPWENTR1.C      104 | |->add_mailpiece
142 RPVIDEO.C       105 | | |->prtbuff { 2 }
120 RPVIDEO.C       106 | | |->prtstr { 3 }
120 RPVIDEO.C       107 | | |..strlen
120 RPVIDEO.C       108 | | |->print_cat
142 RPVIDEO.C       109 | | | |->prtbuff { 2 }
142 RPVIDEO.C       110 | | | |..strlen memcpy
120 RPVIDEO.C       111 | | | |..strcpy memcpy printf
120 RPWENTR1.C      112 | |->are_there_ss
149 RPWLIB.C        113 | |->clear_DE_box { 85 }
158 RPWLIB.C        114 | |->clear_SS_box
316 RPVIDEO.C       115 | | |->clr_box { 32 }
82 MAINRATE.C       116 | |->etoi
117 | | |..memcpy atoi
175 RPWTABLE.C      118 | |->table_search
45 RPWTABLE.C       119 | | |->Init_Table_Info
120 | | |..memcmp
57 RPWTABLE.C       121 | |->table_get
45 RPWTABLE.C       122 | | |->Init_Table_Info { 119 }
123 | | |..strncmp
120 RPVIDEO.C       124 | |->prtstr { 3 }
484 RPWMENU.C       125 | |->Disp_Text { 89 }
126 | | |..memcpy memcmp
189 RPWSS.C         127 |->get_ss
320 RPWSS.C         128 |->validate
129 | | |..memcpy memcmp
83 RPWLIB.C         130 |->handle_error { 30 }
293 RPWSS.C         131 |->save_wnd
387 RPVIDEO.C       132 |->create_box
120 RPVIDEO.C       133 |->prtstr { 3 }
316 RPVIDEO.C       134 |->clr_box { 32 }
262 RPVIDEO.C       135 |->shade_it { 60 }
504 RPWSS.C         136 |->get_choice
175 RPWTABLE.C      137 | | |->table_search { 118 }
57 RPWTABLE.C       138 | | |->table_get { 121 }
142 RPVIDEO.C       139 | | | |->prtbuff { 2 }
992 RPWSS.C         140 | | |->DisplayMultipleSelectMessage
120 RPVIDEO.C       141 | | | |->prtstr { 3 }
262 RPVIDEO.C       142 | | | |->shade_it { 60 }
143 | | | |..strcpy

```

```

[] 288 RPVIDEO.C      144 | |->restor_wnd
[]          |
[] 832 RPWLIB.C      145 | |->display_ss_rate
[] 156 MAINRATE.C    146 | | |->edit_rate { 25 }
[] 120 RPVIDEO.C     147 | | |->prtstr { 3 }
[]          |
[]          148 | |..memcpy atoi malloc free
[]          |
[] 1897 RPWENTR2.C   149 |->display_ss_text
[] 142 RPVIDEO.C     150 | |->prtbuff { 2 }
[]          |
[] 120 RPVIDEO.C     151 |->prtstr { 3 }
[] 630 RPWENTR2.C   152 |->display_record_count { 41 }
[]          |
[] 3048 RPWENTR1.C   153 |->other_revenue
[] 316 RPVIDEO.C     154 | |->clr_box { 32 }
[] 120 RPVIDEO.C     155 | |->prtstr { 3 }
[]          |
[] 600 RPWLIB.C      156 |->num_field
[] 212 RPVIDEO.C     157 | | |->setcur { 17 }
[] 120 RPVIDEO.C     158 | | |->prtstr { 3 }
[]          |
[] 224 RPVIDEO.C     159 | | |->disp_cur
[]          160 | | | ..int86
[]          |
[] 113 RPWLIB.C      161 | | |->clear_message { 31 }
[] 201 RPVIDEO.C     162 | | |->cursor_off { 45 }
[]          163 | | |..strlen getch atof isdigit printf
[]          |
[] 747 RPWLIB.C      164 | |->yes_no
[] 723 RPWLIB.C      165 | | |->date_time { 27 }
[] 120 RPVIDEO.C     166 | | |->prtstr { 3 }
[]          167 | | |..kbhit getch printf
[]          |
[] 2956 RPWENTR1.C   168 |->correct_paid
[] 156 MAINRATE.C    169 | | |->edit_rate { 25 }
[] 120 RPVIDEO.C     170 | | |->prtstr { 3 }
[] 83 RPWLIB.C       171 | | |->handle_error { 30 }
[]          172 | | |..memcpy
[]          |
[] 3239 RPWENTR1.C   173 |->get_variance
[] 82 MAINRATE.C     174 | | |->etoi { 116 }
[] 120 RPVIDEO.C     175 | | |->prtstr { 3 }
[] 747 RPWLIB.C      176 | | |->yes_no { 164 }
[] 1448 RPWENTR1.C   177 | | |->clr_entry_scrn { 35 }
[] 156 MAINRATE.C    178 | | |->edit_rate { 25 }
[]          |
[] 138 MAINRATE.C    179 | | |->trim
[]          180 | | |..strlen strcpy
[]          |
[] 60 RATEREV.C      181 | | |->raterev { 44 }
[] 83 RPWLIB.C       182 | | |->handle_error { 30 }
[]          183 | | |..memcmp
[]          |
[] 484 RPWMENU.C     184 |->Disp_Text { 89 }
[]          |

```

```

    | 2996 RPWENTR1.C      185 | |->not_correct_paid .
    | 156 MAINRATE.C      186 | | |->edit_rate { 25 }
    | 120 RPWVIDEO.C      187 | | |->prtstr { 3 }
    | 138 MAINRATE.C      188 | | |->trim { 179 }
    | 452 RPWENTR1.C      189 | | |->add_mailpiece { 104 }
    |
    | 60 RATEREV.C        190 | |->raterev { 44 }
    |                         191 | ..strcpy strchr atol is_Presort printf getch
    |                         | ..memcpy
    |
    | 3488 RPWENTR1.C      192 | |->get_agency
    | 83 RPWLIB.C          193 | | |->handle_error { 30 }
    | 293 RPWSS.C          194 | | |->save_wnd { 131 }
    | 387 RPWVIDEO.C       195 | | |->create_box { 132 }
    | 120 RPWVIDEO.C       196 | | |->prtstr { 3 }
    | 316 RPWVIDEO.C       197 | | |->clr_box { 32 }
    | 262 RPWVIDEO.C       198 | | |->shade_it { 60 }
    |
    | 3408 RPWENTR1.C      199 | |->disp_agency
    | 142 RPWVIDEO.C       200 | | |->prtbuff { 2 }
    |                         201 | | ..strncpy
    |
    | 3416 RPWENTR1.C      202 | |->get_agency_choice
    | 3408 RPWENTR1.C      203 | | |->disp_agency { 199 }
    | 142 RPWVIDEO.C       204 | | |->prtbuff { 2 }
    | 452 RPWENTR1.C       205 | | |->add_mailpiece { 104 }
    | 1448 RPWENTR1.C      206 | | |->clr_entry_scrn { 35 }
    | 60 RATEREV.C         207 | | |->raterev { 44 }
    |                         208 | | ..strcpy getch printf memcpy atoi
    |
    | 288 RPWVIDEO.C       209 | |->restor_wnd { 144 }
    |                         210 | ..malloc free
    |
    | 173 RPWENTR2.C       211 | |->final_ok
    |
    | 484 RPWENTR2.C       212 | |->get_zip
    | 149 RPWLIB.C          213 | | |->clear_DE_box { 85 }
    |                         214 | | ..strcpy memcpy
    |
    | 981 RPWENTR1.C       215 | |->pop_list6
    | 149 RPWLIB.C          216 | | |->clear_DE_box { 85 }
    |
    | 1546 RPWENTR1.C      217 | |->reset_buckets
    |                         218 | | ..memcpy
    |
    | 484 RPWMENU.C         219 | |->Disp_Text { 89 }
    | 167 RPWLIB.C          220 | | |->num_input { 97 }
    | 83 RPWLIB.C           221 | | |->handle_error { 30 }
    | 452 RPWENTR1.C         222 | | |->add_mailpiece { 104 }
    | 480 RPWENTR1.C         223 | | |->print_cat { 108 }
    |
    | 821 RPWENTR2.C        224 | |->where_to_go
    | 60 RATEREV.C          225 | | |->raterev { 44 }
    |
    | 645 RPWENTRY.C        226 | | |->piece_skip_override
    |
    | 473 RPWVIDEO.C        227 | | | |->sav_wnd

```

```

17 316 RPWVIDEO.C    228 | | | |->clr_box { 32 }
18 484 RPWMENU.C    229 | | | |->Disp_Text { 89 }

19 | | | |
20 852 RPWENTRY.C   230 | | | |->get_skip_fld
21 600 RPWLIB.C     231 | | | |->num_field { 156 }
22 120 RPWVIDEO.C   232 | | | |->prtstr { 3 }
23 747 RPWLIB.C     233 | | | |->yes_no { 164 }

24 | | | |
25 104 RPWLIB.C     234 | | | |->erase_error
26 113 RPWLIB.C     235 | | | |->clear_message { 31 }

27 | | | |
28 236 | | | |..strcpy atoi sprintf Handle_error
strcmp
29 | | | |
30 894 RPWENTRY.C   237 | | | |->display_start

31 | | | |
32 20 RANDOM.C      238 | | | |->get_random
33 | | | |
34 56 RANDOM.C      239 | | | |->ReadSeed
35 | | | |
36 81 RAN2.C        240 | | | |..nr_seed_scatter
37 | | | |
38 68 RAN2.C        241 | | | |->nr_seed_init
39 81 RAN2.C        242 | | | |->nr_seed_scatter { 240 }
40 243 | | | |..memset time

41 | | | |
42 244 | | | |..open read close
43 | | | |
44 30 RAN2.C        245 | | | |->ran2
45 | | | |
46 86 RANDOM.C      246 | | | |->WriteSeed
47 | | | |
48 94 RAN2.C        247 | | | |->nr_seed_gather
49 248 | | | |..open write close access unlink
50 | | | |
51 249 | | | |..modf fmod

52 | | | |
53 120 RPWVIDEO.C   250 | | | |->prtstr { 3 }
54 251 | | | |..sprintf

55 | | | |
56 120 RPWVIDEO.C   252 | | | |->prtstr { 3 }

57 | | | |
58 903 RPWENTRY.C   253 | | | |->AreYouSure
59 104 RPWLIB.C     254 | | | |->erase_error { 234 }
60 120 RPWVIDEO.C   255 | | | |->prtstr { 3 }
61 256 | | | |..getch printf

62 | | | |
63 288 RPWVIDEO.C   257 | | | |->restor_wnd { 144 }
64 258 | | | |..memset malloc printf free

65 | | | |
66 725 RPWENTRY.C   259 | | | |->container_skip_override
67 473 RPWVIDEO.C   260 | | | |->sav_wnd { 227 }
68 316 RPWVIDEO.C   261 | | | |->clr_box { 32 }
69 484 RPWMENU.C    262 | | | |->Disp_Text { 89 }
70 852 RPWENTRY.C   263 | | | |->get_skip_fld { 230 }
71 894 RPWENTRY.C   264 | | | |->display_start { 237 }
72 120 RPWVIDEO.C   265 | | | |->prtstr { 3 }

```

```

    □ 903 RPWENTRY.C      266 | | | |->AreYouSure { 253 }
    □ 288 RPWVIDEO.C     267 | | | |->restor_wnd { 144 }
    □ 268 | | | ..memset malloc printf free

    □
    □ 452 RPWENTR1.C     269 | | | |->add_mailpiece { 104 }
    □ 120 RPWVIDEO.C     270 | | | |->prtstr { 3 }

    □
    □ 433 RPWENTR1.C     271 | | | |->remove_mailpiece
    □ 452 RPWENTR1.C     272 | | | | |->add_mailpiece { 104 }

    □
    □ 316 RPWVIDEO.C     273 | | | |->clr_box { 32 }
    □ 630 RPWENTR2.C     274 | | | |->display_record_count { 41 }
    □ 275 | | | ..printf strncmp

    □
    □ 276 | | ..printf

    □
    □ 1041 RPWENTR1.C    277 |->pop_list7
    □ 316 RPWVIDEO.C     278 | |->clr_box { 32 }
    □ 484 RPWMENU.C      279 | |->Disp_Text { 89 }
    □ 167 RPWLIB.C       280 | |->num_input { 97 }
    □ 452 RPWENTR1.C     281 | |->add_mailpiece { 104 }
    □ 83 RPWLIB.C        282 | |->handle_error { 30 }
    □ 821 RPWENTR2.C     283 | |->where_to_go { 224 }
    □ 284 | | ..printf

    □
    □ 1085 RPWENTR1.C    285 |->letter_non
    □ 316 RPWVIDEO.C     286 | |->clr_box { 32 }
    □ 484 RPWMENU.C      287 | |->Disp_Text { 89 }
    □ 167 RPWLIB.C       288 | |->num_input { 97 }
    □ 452 RPWENTR1.C     289 | |->add_mailpiece { 104 }
    □ 821 RPWENTR2.C     290 | |->where_to_go { 224 }
    □ 291 | | ..printf

    □
    □ 744 RPWENTR2.C     292 |->get_enclos
    □ 149 RPWLIB.C       293 | |->clear_DE_box { 85 }

    □
    □ 3422 RPWENTR2.C    294 |->compress_ss_fields_array
    □ 295 | | ..memcpy

    □
    □ 484 RPWMENU.C      296 | |->Disp_Text { 89 }
    □ 167 RPWLIB.C       297 | |->num_input { 97 }
    □ 452 RPWENTR1.C     298 | |->add_mailpiece { 104 }
    □ 299 | | ..memcmp

    □
    □ 641 RPWENTR2.C     300 |->get_revenue_only
    □ 149 RPWLIB.C       301 | |->clear_DE_box { 85 }
    □ 113 RPWLIB.C       302 | |->clear_message { 31 }
    □ 484 RPWMENU.C      303 | |->Disp_Text { 89 }
    □ 600 RPWLIB.C       304 | |->num_field { 156 }
    □ 452 RPWENTR1.C     305 | |->add_mailpiece { 104 }
    □ 306 | | ..strcpy strchr atol

    □
    □ 1119 RPWENTR1.C    307 |->pop_list9
    □ 316 RPWVIDEO.C     308 | |->clr_box { 32 }
    □ 484 RPWMENU.C      309 | |->Disp_Text { 89 }
    □ 167 RPWLIB.C       310 | |->num_input { 97 }
    □ 83 RPWLIB.C        311 | |->handle_error { 30 }

```

```

| 452 RPWENTR1.C      312 | |->add_mailpiece { 104 }
| 821 RPWENTR2.C      313 | |->where_to_go { 224 }
|                           314 | ..memcpy printf
|
| 1122 RPWENTR2.C      315 |->get_machine
| 316 RPWVIDEO.C       316 | |->clr_box { 32 }
| 484 RPWMENU.C        317 | |->Disp_Text { 89 }
| 167 RPWLIB.C         318 | |->num_input { 97 }
| 452 RPWENTR1.C        319 | |->add_mailpiece { 104 }
| 821 RPWENTR2.C        320 | |->where_to_go { 224 }
|                           321 | ..printf
|
| 1160 RPWENTR2.C      322 |->get_local
| 316 RPWVIDEO.C       323 | |->clr_box { 32 }
| 484 RPWMENU.C        324 | |->Disp_Text { 89 }
| 167 RPWLIB.C         325 | |->num_input { 97 }
| 452 RPWENTR1.C        326 | |->add_mailpiece { 104 }
|                           327 | ..memset printf
|
| 1935 RPWENTR2.C      328 |->get_int_pp_type
| 316 RPWVIDEO.C       329 | |->clr_box { 32 }
| 484 RPWMENU.C        330 | |->Disp_Text { 89 }
| 167 RPWLIB.C         331 | |->num_input { 97 }
| 452 RPWENTR1.C        332 | |->add_mailpiece { 104 }
| 821 RPWENTR2.C        333 | |->where_to_go { 224 }
|                           334 | ..printf
|
| 1260 RPWENTR2.C      335 |->input_ins_fee
| 3422 RPWENTR2.C      336 | |->compress_ss_fields_array { 294 }
| 316 RPWVIDEO.C       337 | |->clr_box { 32 }
| 120 RPWVIDEO.C       338 | |->prtstr { 3 }
| 600 RPWLIB.C         339 | |->num_field { 156 }
| 452 RPWENTR1.C        340 | |->add_mailpiece { 104 }
| 433 RPWENTR1.C        341 | |->remove_mailpiece { 271 }
|
| 1366 RPWENTR2.C      342 | |->verify_ins_fee
| 83 RPWLIB.C          343 | |->handle_error { 30 }
| 747 RPWLIB.C          344 | |->yes_no { 164 }
| 113 RPWLIB.C          345 | |->clear_message { 31 }
|
| 1806 RPWENTR2.C      346 | |->get_ins_sscatcode
|                           347 | ..atoi memcpy
|
| 832 RPWLIB.C          348 | |->display_ss_rate { 145 }
| 60 RATEREV.C          349 | |->raterev { 44 }
|                           350 | ..memcmp strcpy strchr atol printf memcpy
|
| 1971 RPWENTR2.C      351 |->surf_or_air
| 316 RPWVIDEO.C       352 | |->clr_box { 32 }
| 484 RPWMENU.C        353 | |->Disp_Text { 89 }
| 167 RPWLIB.C         354 | |->num_input { 97 }
| 452 RPWENTR1.C        355 | |->add_mailpiece { 104 }
| 821 RPWENTR2.C        356 | |->where_to_go { 224 }
|                           357 | ..memcpy printf
|
| 2058 RPWENTR2.C      358 |->get_intl_fee
| 316 RPWVIDEO.C       359 | |->clr_box { 32 }

```

```

U 120 RPWVIDEO.C      360 | |->prtstr { 3 }
H 600 RPWLIB.C        361 | |->num_field { 156 }
O 452 RPWENTR1.C      362 | |->add_mailpiece { 104 }
U 156 MAINRATE.C      363 | |->edit_rate { 25 }
O 83 RPWLIB.C         364 | |->handle_error { 30 }
U
U 154 CLIMIT.C        365 | |->validate_limit
U
U 747 RPWLIB.C         366 | |..strncmp
U
U 104 RPWLIB.C         367 | |->yes_no { 164 }
U
U 1214 RPWENTR1.C     368 | |->erase_error { 234 }
U
U
U 316 RPWVIDEO.C       369 | |..strcpy strchr atol
U
U 484 RPWMENU.C        370 | |->pop_list10
U
U 167 RPWLIB.C          371 | |->clr_box { 32 }
U
U 452 RPWENTR1.C        372 | |->Disp_Text { 89 }
U
U 821 RPWENTR2.C        373 | |->num_input { 97 }
U
U
U 1423 RPWENTR2.C      374 | |->add_mailpiece { 104 }
U
U 3422 RPWENTR2.C      375 | |->where_to_go { 224 }
U
U
U 316 RPWVIDEO.C        376 | |..printf
U
U 120 RPWVIDEO.C        377 | |->input_cod_fee
U
U 600 RPWLIB.C          378 | |->compress_ss_fields_array { 294 }
U
U 433 RPWENTR1.C        379 | |->clr_box { 32 }
U
U
U 1487 RPWENTR2.C      380 | |->prtstr { 3 }
U
U 83 RPWLIB.C           381 | |->num_field { 156 }
U
U 832 RPWLIB.C          382 | |->remove_mailpiece { 271 }
U
U
U 1850 RPWENTR2.C      383 | |->verify_cod_fee
U
U 60 RATEREV.C          384 | |->handle_error { 30 }
U
U 832 RPWLIB.C          385 | |->display_ss_rate { 145 }
U
U
U 60 RATEREV.C          386 | |->get_cod_sscatcode
U
U
U 2117 RPWENTR2.C      387 | |..atoi memcpy
U
U
U 120 RPWVIDEO.C        388 | |->raterev { 44 }
U
U
U 600 RPWLIB.C          389 | |..memcmp strcpy strchr atol memcpy
U
U
U 1448 RPWENTR1.C      390 | |->alter_of_charges
U
U 83 RPWLIB.C           391 | |->prtstr { 3 }
U
U 138 MAINRATE.C        392 | |->num_field { 156 }
U
U 452 RPWENTR1.C        393 | |->clr_entry_scrn { 35 }
U
U 142 RPWVIDEO.C        394 | |->handle_error { 30 }
U
U 156 MAINRATE.C        395 | |->trim { 179 }
U
U 60 RATEREV.C          396 | |->add_mailpiece { 104 }
U
U
U 2221 RPWENTR2.C      397 | |->prtbuff { 2 }
U
U 316 RPWVIDEO.C        398 | |->edit_rate { 25 }
U
U 832 RPWLIB.C          399 | |->raterev { 44 }
U
U
U 2221 RPWENTR2.C      400 | |..strcpy memcpy atoi
U
U
U 316 RPWVIDEO.C        401 | |->intl_reg_mailpiece
U
U 484 RPWMENU.C          402 | |->clr_box { 32 }
U
U 167 RPWLIB.C           403 | |->Disp_Text { 89 }
U
U 452 RPWENTR1.C        404 | |->num_input { 97 }
U
U 821 RPWENTR2.C        405 | |->add_mailpiece { 104 }
U
U
U 2221 RPWENTR2.C      406 | |->where_to_go { 224 }
U
U
U 316 RPWVIDEO.C        407 | |..printf

```

```

| 2171 RPWENTR2.C      408 |->reg_ins_question
| 316 RPWVIDEO.C      409 |->clr_box { 32 }
| 120 RPWVIDEO.C      410 |->prtstr { 3 }
| 747 RPWLIB.C         411 |->yes_no { 164 }
| 452 RPWENTR1.C      412 |->add_mailpiece { 104 }
| 60 RATEREV.C         413 |->raterev { 44 }
| 484 RPWMENU.C        414 |..printf
|
| 1504 RPWENTR2.C      415 |->input_reg_fee
| 3422 RPWENTR2.C      416 |->compress_ss_fields_array { 294 }
| 316 RPWVIDEO.C      417 |->clr_box { 32 }
| 120 RPWVIDEO.C      418 |->prtstr { 3 }
| 600 RPWLIB.C         419 |->num_field { 156 }
| 433 RPWENTR1.C      420 |->remove_mailpiece { 271 }
| 484 RPWMENU.C        421 |..memcmp strcpy
|
| 506 RPWENTR1.C      422 |->pop_list1
| 149 RPWLIB.C         423 |->clear_DE_box { 85 }
| 484 RPWMENU.C        424 |->Disp_Text { 89 }
| 120 RPWVIDEO.C      425 |->prtstr { 3 }
| 630 RPWENTR2.C      426 |->display_record_count { 41 }
| 167 RPWLIB.C         427 |->num_input { 97 }
| 452 RPWENTR1.C      428 |->add_mailpiece { 104 }
| 83 RPWLIB.C          429 |->handle_error { 30 }
| 821 RPWENTR2.C      430 |->where_to_go { 224 }
| 484 RPWMENU.C        431 |..memcpy memset printf
|
| 2262 RPWENTR2.C      432 |->cert_type
| 316 RPWVIDEO.C      433 |->clr_box { 32 }
| 484 RPWMENU.C        434 |->Disp_Text { 89 }
| 167 RPWLIB.C         435 |->num_input { 97 }
| 452 RPWENTR1.C      436 |->add_mailpiece { 104 }
| 821 RPWENTR2.C      437 |->where_to_go { 224 }
| 484 RPWMENU.C        438 |..memcpy printf
|
| 2314 RPWENTR2.C      439 |->input_cert_fee
| 316 RPWVIDEO.C      440 |->clr_box { 32 }
| 120 RPWVIDEO.C      441 |->prtstr { 3 }
| 600 RPWLIB.C         442 |->num_field { 156 }
| 452 RPWENTR1.C      443 |->add_mailpiece { 104 }
| 83 RPWLIB.C          444 |->handle_error { 30 }
|
| 181 CLIMIT.C         445 |->validate_multiple
| 484 RPWMENU.C        446 |..strcmp
|
| 154 CLIMIT.C         447 |->validate_limit { 365 }
| 747 RPWLIB.C          448 |->yes_no { 164 }
| 104 RPWLIB.C          449 |->erase_error { 234 }
| 60 RATEREV.C          450 |->raterev { 44 }
| 156 MAINRATE.C        451 |->edit_rate { 25 }
| 484 RPWMENU.C          452 |..strcpy strchr atol
|
| 2412 RPWENTR2.C      453 |->final_ok_certs
| 120 RPWVIDEO.C      454 |->prtstr { 3 }
| 316 RPWVIDEO.C      455 |->clr_box { 32 }
| 484 RPWMENU.C        456 |->Disp_Text { 89 }

```

```

    | 167 RPWLIB.C      457 | |->num_input { 97 }
    | 293 RPWENTR2.C   458 | |->write_record { 5 }
    | 1448 RPWENTR1.C  459 | |->clr_entry_scrn { 35 }
    | 60 RATEREV.C     460 | |->raterrev { 44 }
    |                   461 | ..itoa printf
    |
    | 2456 RPWENTR2.C  462 | |->get_x_category
    | 316 RPWVIDEO.C   463 | |->clr_box { 32 }
    | 484 RPWMENU.C    464 | |->Disp_Text { 89 }
    | 167 RPWLIB.C     465 | |->num_input { 97 }
    | 120 RPWVIDEO.C   466 | |->prtstr { 3 }
    | 452 RPWENTR1.C   467 | |->add_mailpiece { 104 }
    | 480 RPWENTR1.C   468 | |->print_cat { 108 }
    | 821 RPWENTR2.C   469 | |->where_to_go { 224 }
    |                   470 | ..memcpy memset printf
    |
    | 2536 RPWENTR2.C  471 | |->x_cat_submenu
    | 316 RPWVIDEO.C   472 | |->clr_box { 32 }
    | 120 RPWVIDEO.C   473 | |->prtstr { 3 }
    | 484 RPWMENU.C    474 | |->Disp_Text { 89 }
    | 167 RPWLIB.C     475 | |->num_input { 97 }
    | 452 RPWENTR1.C   476 | |->add_mailpiece { 104 }
    | 480 RPWENTR1.C   477 | |->print_cat { 108 }
    | 821 RPWENTR2.C   478 | |->where_to_go { 224 }
    |                   479 | ..printf
    |
    | 2588 RPWENTR2.C  480 | |->get_post_due
    | 82 MAINRATE.C    481 | |->etoi { 116 }
    | 316 RPWVIDEO.C   482 | |->clr_box { 32 }
    | 120 RPWVIDEO.C   483 | |->prtstr { 3 }
    | 600 RPWLIB.C     484 | |->num_field { 156 }
    | 452 RPWENTR1.C   485 | |->add_mailpiece { 104 }
    | 83 RPWLIB.C      486 | |->handle_error { 30 }
    |
    | 859 RPWSS.C       487 | |->validate_fee
    | 83 RPWLIB.C      488 | | |->handle_error { 30 }
    |                   489 | | ..atoi
    |
    | 154 CLIMIT.C     490 | |->validate_limit { 365 }
    | 747 RPWLIB.C     491 | |->yes_no { 164 }
    | 104 RPWLIB.C     492 | |->erase_error { 234 }
    | 156 MAINRATE.C   493 | |->edit_rate { 25 }
    | 60 RATEREV.C     494 | |->raterrev { 44 }
    |                   495 | ..strcpy strchr atol
    |
    | 2698 RPWENTR2.C  496 | |->get_subsample_wt
    | 316 RPWVIDEO.C   497 | |->clr_box { 32 }
    | 484 RPWMENU.C    498 | |->Disp_Text { 89 }
    | 167 RPWLIB.C     499 | |->num_input { 97 }
    |
    | 2803 RPWENTR2.C  500 | |->get_sub_weight
    | 316 RPWVIDEO.C   501 | | |->clr_box { 32 }
    | 120 RPWVIDEO.C   502 | | |->prtstr { 3 }
    | 484 RPWMENU.C    503 | | |->Disp_Text { 89 }
    | 104 RPWLIB.C     504 | | |->erase_error { 234 }
    |
    | 31 RPWCOMM.C     505 | | |->ask_scale

```

```

[] 83 RPWLIB.C      506 | | | |->handle_error { 30 }
[]           507 | | | |..strcpy asiopen asigetc asiputs time
printf
[]           | | | |..asiquit isdigit atof itoa strlen fcvt
[]
[] 600 RPWLIB.C    508 | | | |->num_field { 156 }
[] 60 RATEREV.C   509 | | | |->raterev { 44 }
[] 83 RPWLIB.C    510 | | | |->handle_error { 30 }
[]           511 | | | |..floor fmod ltoa strcpy strlen fcvt getch
atol
[]           | | | |..atof printf
[]
[] 83 RPWLIB.C    512 | | | |->handle_error { 30 }
[] 747 RPWLIB.C   513 | | | |->yes_no { 164 }
[] 113 RPWLIB.C   514 | | | |->clear_message { 31 }
[] 60 RATEREV.C   515 | | | |->raterev { 44 }
[]           516 | | | |..atol atof printf
[]
[] 3194 RPWENTR2.C 517 |->get_indicia
[] 82 MAINRATE.C  518 | |->etoi { 116 }
[] 120 RPWVIDEO.C 519 | |->prtstr { 3 }
[]
[] 1656 RPWENTR2.C 520 |->post_weight_edits
[]
[] 3515 RPWENTR1.C 521 |->priority_question
[] 149 RPWLIB.C   522 | |->clear_DE_box { 85 }
[] 484 RPWMENU.C  523 | |->Disp_Text { 89 }
[] 167 RPWLIB.C   524 | |->num_input { 97 }
[] 452 RPWENTR1.C 525 | |->add_mailpiece { 104 }
[] 821 RPWENTR2.C 526 | |->where_to_go { 224 }
[]           527 | |..printf
[]
[] 3592 RPWENTR1.C 528 |->dbmc_question
[] 316 RPWVIDEO.C  529 | |->clr_box { 32 }
[] 484 RPWMENU.C  530 | |->Disp_Text { 89 }
[] 747 RPWLIB.C   531 | |->yes_no { 164 }
[] 104 RPWLIB.C   532 | |->erase_error { 234 }
[] 1448 RPWENTR1.C 533 | |->clr_entry_scrn { 35 }
[] 452 RPWENTR1.C  534 | |->add_mailpiece { 104 }
[]
[] 4037 RPWENTR1.C 535 | |->get_dbmc_zip
[] 175 RPWTABLE.C  536 | | |->table_search { 118 }
[] 57 RPWTABLE.C   537 | | |->table_get { 121 }
[] 82 MAINRATE.C  538 | | | |->etoi { 116 }
[]
[] 83 RPWLIB.C   539 | |->handle_error { 30 }
[] 60 RATEREV.C   540 | |->raterev { 44 }
[]           541 | |..itoa memcpy strlen memset printf
[]
[] 3659 RPWENTR1.C 542 |->parcel_size
[] 316 RPWVIDEO.C  543 | |->clr_box { 32 }
[] 484 RPWMENU.C  544 | |->Disp_Text { 89 }
[] 167 RPWLIB.C   545 | |->num_input { 97 }
[] 120 RPWVIDEO.C 546 | |->prtstr { 3 }
[] 630 RPWENTR2.C  547 | |->display_record_count { 41 }
[] 600 RPWLIB.C   548 | |->num_field { 156 }
[] 433 RPWENTR1.C  549 | |->remove_mailpiece { 271 }

```

```

    60 RATEREV.C      550 | |->raterev { 44 }
    83 RPWLIB.C      551 | |->handle_error { 30 }
   747 RPWLIB.C      552 | |->yes_no { 164 }
  1448 RPWENTR1.C    553 | |->clr_entry_scrn { 35 }
   113 RPWLIB.C      554 | |->clear_message { 31 }
   142 RPWVIDEO.C    555 | |->prtbuff { 2 }
   821 RPWENTR2.C    556 | |->where_to_go { 224 }
                           557 | ..memcpy strcpy memcmp printf atoi
                           |
  2022 RPWENTR2.C    558 |->canada_question
   316 RPWVIDEO.C    559 | |->clr_box { 32 }
  484 RPWMENU.C     560 | |->Disp_Text { 89 }
  167 RPWLIB.C      561 | |->num_input { 97 }
  452 RPWENTR1.C    562 | |->add_mailpiece { 104 }
   821 RPWENTR2.C    563 | |->where_to_go { 224 }
                           564 | ..printf
                           |
  2917 RPWENTR1.C    565 |->are_there_more_ss
   316 RPWVIDEO.C    566 | |->clr_box { 32 }
   120 RPWVIDEO.C    567 | |->prtstr { 3 }
   747 RPWLIB.C      568 | |->yes_no { 164 }
   104 RPWLIB.C      569 | |->erase_error { 234 }
                           570 | ..printf
                           |
  3439 RPWENTR2.C    571 |->ib_reply_submenu
   316 RPWVIDEO.C    572 | |->clr_box { 32 }
   484 RPWMENU.C    573 | |->Disp_Text { 89 }
   167 RPWLIB.C      574 | |->num_input { 97 }
   480 RPWENTR1.C    575 | |->print_cat { 108 }
   156 MAINRATE.C    576 | |->edit_rate { 25 }
   120 RPWVIDEO.C    577 | |->prtstr { 3 }
   821 RPWENTR2.C    578 | |->where_to_go { 224 }
                           579 | ..memcpy printf
                           |
   79 RPWSS.C        580 |->ss_menu
   149 RPWLIB.C      581 | |->clear_DE_box { 85 }
   120 RPWVIDEO.C    582 | |->prtstr { 3 }
                           |
  1403 RPWENTR1.C    583 |->presort_1st_menu
   316 RPWVIDEO.C    584 | |->clr_box { 32 }
   480 RPWENTR1.C    585 | |->print_cat { 108 }
   484 RPWMENU.C    586 | |->Disp_Text { 89 }
   167 RPWLIB.C      587 | |->num_input { 97 }
   452 RPWENTR1.C    588 | |->add_mailpiece { 104 }
   821 RPWENTR2.C    589 | |->where_to_go { 224 }
                           590 | ..printf
                           |
  2583 RPWENTR1.C    591 |->presort_gov_menu
   316 RPWVIDEO.C    592 | |->clr_box { 32 }
   120 RPWVIDEO.C    593 | |->prtstr { 3 }
   484 RPWMENU.C    594 | |->Disp_Text { 89 }
   167 RPWLIB.C      595 | |->num_input { 97 }
   452 RPWENTR1.C    596 | |->add_mailpiece { 104 }
   630 RPWENTR2.C    597 | |->display_record_count { 41 }
   480 RPWENTR1.C    598 | |->print_cat { 108 }
   821 RPWENTR2.C    599 | |->where_to_go { 224 }
                           600 | ..memcpy memset memcmp printf

```

```

1362 RPWENTR1.C      601 |->letter_card
316 RPVIDEO.C        602 | |->clr_box { 32 }
484 RPWMENU.C        603 | |->Disp_Text { 89 }
167 RPWLIB.C          604 | |->num_input { 97 }
452 RPWENTR1.C        605 | |->add_mailpiece { 104 }
480 RPWENTR1.C        606 | |->print_cat { 108 }
821 RPWENTR2.C        607 | |->where_to_go { 224 }
608 | ..printf
|
83 RPWLIB.C          609 |->handle_error { 30 }

429 RPWMENU.C        610 Disp_Error
392 RPWMENU.C        611 |->Init_Text_Display { 90 }
404 RPWMENU.C        612 |->Search_Text_Array { 91 }
83 RPWLIB.C          613 |->handle_error { 30 }
614 | ..sprintf
|
458 RPWMENU.C        615 Disp_MailPiece
392 RPWMENU.C        616 |->Init_Text_Display { 90 }
404 RPWMENU.C        617 |->Search_Text_Array { 91 }
452 RPWENTR1.C        618 |->add_mailpiece { 104 }
619 | ..sprintf
|
2438 RPWENTRY.C      620 _setargv
|
2441 RPWENTRY.C      621 _setenvp
|
323 REVIEW.C          622 catcomp
623 | ..memcmp
|
131 RPWLIB.C          624 clear_mailpiece
316 RPVIDEO.C         625 |->clr_box { 32 }
|
1357 RPREVNU.C        626 custom_rates
|
1624 RATEREV.C        627 display_mmcodes
316 RPVIDEO.C         628 |->clr_box { 32 }
|
1679 RATEREV.C        629 |->display_menu_bar
120 RPVIDEO.C          630 | |->prtstr { 3 }
|
120 RPVIDEO.C          631 |->prtstr { 3 }
632 | ..getch printf
|
1570 RATEREV.C        633 display_rules
316 RPVIDEO.C         634 |->clr_box { 32 }
120 RPVIDEO.C          635 |->prtstr { 3 }
636 | ..getch printf
|
2520 RPWENTR1.C        637 edit_rate?385?
638 | ..strcpy ltoa strlen memcpy
|
816 RPWLIB.C          639 etoi?493?
640 | ..memcpy atoi
|
107 CITYZIP.C          641 findcity

```

```

1 29 CITYZIP.C      642 |->ptr_ziphelpfile { 15 }
2                               643 |..strlen memcmp

3 1111 RPWREVNU.C    644 fourth_rates
4 82 MAINRATE.C      645 |->etoi { 116 }
5
6 820 RPWREVNU.C     646 |->parcel_rates
7 142 RPWVIDEO.C     647 | |->prtbuff { 2 }
8 120 RPWVIDEO.C     648 | |->prtstr { 3 }
9
10 104 RPWREVNU.C    649 | |->print_rate
11 156 MAINRATE.C    650 | | |->edit_rate { 25 }
12 138 MAINRATE.C    651 | | |->trim { 179 }
13 120 RPWVIDEO.C    652 | | |->prtstr { 3 }
14                               653 | | ..strlen
15
16 850 RPWLIB.C       654 | |->scroll_bar
17
18 343 RPWVIDEO.C    655 | | |->hilite
19
20 357 RPWVIDEO.C    656 | | |->grab_scr_char
21 82 MAINRATE.C      657 | | |->etoi { 116 }
22 60 RATEREV.C       658 | | |->raterrev { 44 }
23                               659 | | ..strcpy getch strchr atol
24
25 84 RPWREVNU.C     660 | |->rate_screen_reset
26 316 RPWVIDEO.C     661 | | |->clr_box { 32 }
27 120 RPWVIDEO.C     662 | | |->prtstr { 3 }
28 630 RPWENTR2.C     663 | | |->display_record_count { 41 }
29 113 RPWLIB.C       664 | | |->clear_message { 31 }
30
31 433 RPWENTR1.C     665 | |->remove_mailpiece { 271 }
32 480 RPWENTR1.C     666 | |->print_cat { 108 }
33 452 RPWENTR1.C     667 | |->add_mailpiece { 104 }
34
35 5 BPMRATE.C        668 |->BPM_SP_Rate
36
37 1293 RPWREVNU.C    669 |->Barcode_Display
38
39 1341 RPWREVNU.C    670 | |->Barcode_Header
40 120 RPWVIDEO.C     671 | | |->prtstr { 3 }
41
42 120 RPWVIDEO.C     672 | |->prtstr { 3 }
43 104 RPWREVNU.C     673 | |->print_rate { 649 }
44 850 RPWLIB.C       674 | |->scroll_bar { 654 }
45 452 RPWENTR1.C     675 | |->add_mailpiece { 104 }
46 84 RPWREVNU.C     676 | |->rate_screen_reset { 660 }
47
48 433 RPWENTR1.C     677 | |->remove_mailpiece { 271 }
49 480 RPWENTR1.C     678 | |->print_cat { 108 }
50
51 64 BPMRATE.C        679 |->BPM_CRT_Rate
52
53 39 BPMRATE.C        680 |->BPM_Bulk_Rate
54
55 39 BPMRATE.C        681 |->BPM_Bulk_Rate { 680 }
56 1341 RPWREVNU.C    682 |->Barcode_Header { 670 }

```

```

□ 120 RPVIDEO.C      683 |->prtstr { 3 }
□ 104 RPREVNU.C     684 |->print_rate { 649 }
□ 850 RPWLIB.C      685 |->scroll_bar { 654 }
□ 84 RPREVNU.C      686 |->rate_screen_reset { 660 }

□
□ 2348 RPWENTR1.C   687 get_auto
□ 316 RPVIDEO.C     688 |->clr_box { 32 }
□ 484 RPWMENU.C    689 |->Disp_Text { 89 }
□ 167 RPWLIB.C     690 |->num_input { 97 }
□ 452 RPWENTR1.C   691 |->add_mailpiece { 104 }
□ 821 RPWENTR2.C   692 |->where_to_go { 224 }
□ 693 |..printf

□
□ 1066 REVIEW.C     694 getkey?218?
□ 723 RPWLIB.C      695 |->date_time { 27 }
□ 120 RPVIDEO.C     696 |->prtstr { 3 }
□ 697 |.._bios_keybrd tolower

□
□ 60 SAMPSORT.C    698 handle_error?541?
□ 120 RPVIDEO.C     699 |->prtstr { 3 }
□ 262 RPVIDEO.C     700 |->shade_it { 60 }
□ 211 ASIPORTS.H    701 |->exit { 21 }
□ 702 |..strlen getch free fclose _asiexit

□
□ 1290 REVIEW.C     703 hash?223?

□
□ 3183 RPWENTR1.C   704 is_presort
□ 3223 RPWENTR1.C   705 |->match
□ 706 |..strncmp

□
□ 380 ASIPORTS.H    707 isalert
□ 708 |.._iswhat

□
□ 400 ASIPORTS.H    709 isbreak
□ 710 |.._isstat

□
□ 403 ASIPORTS.H    711 iscd
□ 712 |.._isstat

□
□ 407 ASIPORTS.H    713 ischgcd
□ 714 |.._isstat

□
□ 405 ASIPORTS.H    715 ischgcts
□ 716 |.._isstat

□
□ 406 ASIPORTS.H    717 ischgdsr
□ 718 |.._isstat

□
□ 408 ASIPORTS.H    719 ischgri
□ 720 |.._isstat

□
□ 401 ASIPORTS.H    721 iscts
□ 722 |.._isstat

□
□ 416 ASIPORTS.H    723 isctsblocked
□ 724 |.._iswhat

```

|   |                |                                   |
|---|----------------|-----------------------------------|
| □ | 402 ASIPORTS.H | 725 isdsr<br>726  .._isstat       |
| □ | 399 ASIPORTS.H | 727 isframerr<br>728  .._isstat   |
| □ | 390 ASIPORTS.H | 729 isigalert<br>730  .._iswhat   |
| □ | 393 ASIPORTS.H | 731 isigcd<br>732  .._iswhat      |
| □ | 391 ASIPORTS.H | 733 isigcts<br>734  .._iswhat     |
| □ | 392 ASIPORTS.H | 735 isigdsr<br>736  .._iswhat     |
| □ | 395 ASIPORTS.H | 737 isiglstat<br>738  .._iswhat   |
| □ | 394 ASIPORTS.H | 739 isigmstat<br>740  .._iswhat   |
| □ | 386 ASIPORTS.H | 741 islinerr<br>742  .._iswhat    |
| □ | 387 ASIPORTS.H | 743 ismodemerr<br>744  .._iswhat  |
| □ | 412 ASIPORTS.H | 745 isnod<br>746  .._iswhat       |
| □ | 410 ASIPORTS.H | 747 isncts<br>748  .._iswhat      |
| □ | 411 ASIPORTS.H | 749 isndsr<br>750  .._iswhat      |
| □ | 397 ASIPORTS.H | 751 isoverrun<br>752  .._isstat   |
| □ | 398 ASIPORTS.H | 753 isparityerr<br>754  .._isstat |
| □ | 404 ASIPORTS.H | 755 isri<br>756  .._isstat        |
| □ | 413 ASIPORTS.H | 757 isring<br>758  .._iswhat      |
| □ | 381 ASIPORTS.H | 759 isrxempty<br>760  .._iswhat   |
| □ | 382 ASIPORTS.H | 761 isrxfull<br>762  .._iswhat    |

```

□ 389 ASIPORTS.H    763 isrxintrunning
□          764 |.._iswhat
□
□ 383 ASIPORTS.H    765 isrxovflow
□          766 |.._iswhat
□
□ 384 ASIPORTS.H    767 istxempty
□          768 |.._iswhat
□
□ 385 ASIPORTS.H    769 istxfull
□          770 |.._iswhat
□
□ 388 ASIPORTS.H    771 istxintrunning
□          772 |.._iswhat
□
□ 414 ASIPORTS.H    773 isxmrxing
□          774 |.._iswhat
□
□ 415 ASIPORTS.H    775 isxoffblocked
□          776 |.._iswhat
□
□ 167 LOOKUP.C      777 item_lookup
□          |
□ 13 RPWWIN.C       778 |->window_init
□          779 | |..malloc
□          |
□ 44 RPWWIN.C       780 |->window_open
□ 473 RPWVIDEO.C    781 | |->sav_wnd { 227 }
□ 316 RPWVIDEO.C    782 | |->clr_box { 32 }
□ 387 RPWVIDEO.C    783 | |->create_box { 132 }
□          784 | |..malloc
□          |
□ 78 RPWWIN.C       785 |->window_write
□ 120 RPWVIDEO.C    786 | |->prtstr { 3 }
□ 142 RPWVIDEO.C    787 | |->prtbuff { 2 }
□ 316 RPWVIDEO.C    788 | |->clr_box { 32 }
□          789 | |..strlen
□          |
□ 284 LOOKUP.C      790 |->win_lookup_draw
□ 78 RPWWIN.C       791 | |->window_write { 785 }
□          |
□ 256 RPWWIN.C      792 | |->window_hilite
□ 343 RPWVIDEO.C    793 | | |->hilite { 655 }
□          |
□ 303 LOOKUP.C      794 |->win_new_row
□ 256 RPWWIN.C      795 | |->window_hilite { 792 }
□ 284 LOOKUP.C      796 | |->win_lookup_draw { 790 }
□          |
□ 264 LOOKUP.C      797 |->findany
□          798 | |..strlen memcmp
□          |
□ 65 RPWWIN.C       799 |->window_close
□ 288 RPWVIDEO.C    800 | |->restor_wnd { 144 }
□          801 | |..free
□          |
□          802 |..strncpy putchar

```

```

□ 1303 REVIEW.C      803 keycomp
□                                         804 |..memcmp
□
□ 1210 RPWENTR2.C    805 level_a_b
□ 316 RPVIDEO.C     806 |->clr_box { 32 }
□ 484 RPWMENU.C    807 |->Disp_Text { 89 }
□ 156 MAINRATE.C   808 |->edit_rate { 25 }
□ 120 RPVIDEO.C    809 |->prtstr { 3 }
□ 167 RPWLIB.C     810 |->num_input { 97 }
□ 452 RPWENTR1.C   811 |->add_mailpiece { 104 }
□                                         812 |..printf
□
□ 1047 RPWENTRY.C   813 load_rate_table?470?
□ 1219 RPWENTRY.C   814 |->handle_error2 { 6 }
□ 82 MAINRATE.C    815 |->etoi { 116 }
□                                         816 |..fopen fseek fclose fread
□
□ 1127 RPWENTRY.C   817 load_screen_buffer?472?
□ 1219 RPWENTRY.C   818 |->handle_error2 { 6 }
□                                         819 |..fopen fseek malloc fread fgets strlen fclose
□
□ 30 HASH.C          820 main
□                                         |
□ 69 HASH.C          821 |->hash
□                                         822 |..fopen fwrite fread fseek fcloseall
□
□ 91 PRERPW.C        823 main?101?
□                                         |
□ 308 PRERPW.C       824 |->clrscr
□                                         825 | |..int86
□
□ 319 PRERPW.C       826 |->CheckForDisk
□                                         827 | |..int86
□
□ 211 ASIPORTS.H     828 |->exit { 21 }
□
□ 126 PRERPW.C       829 |->CopyUpFiles
□                                         |
□ 287 PRERPW.C       830 | |->get_zone_files
□                                         831 | | |->exit { 21 }
□                                         832 | | |.._dos_findfirst printf kbhit getch strcpy
□                                         | | |.._dos_findnext
□
□ 171 PRERPW.C       833 | |->get_time_date_a
□ 211 ASIPORTS.H     834 | | |->exit { 21 }
□                                         835 | | |..strcpy strcat _dos_open printf kbhit getch
□                                         | | |.._dos_getftime _dos_close
□
□ 206 PRERPW.C       836 | |->get_time_date_c
□ 211 ASIPORTS.H     837 | | |->exit { 21 }
□                                         838 | | |.._dos_open _dos_getftime _dos_close printf
□                                         | | |..kbhit getch
□
□ 234 PRERPW.C       839 | |->compare_em
□ 211 ASIPORTS.H     840 | |->exit { 21 }
□                                         841 | |..printf strcpy system getch

```

```

842 | ..printf _dos_findfirst getch system

268 RPWENTRY.C 843 main?442?
| |
23 RPWVIDEO.C 844 |->init_video
| 845 | |..memcpy _dos_findfirst
|
318 CDATE.C 846 |->not_valid_version
201 RPWVIDEO.C 847 | |->cursor_off { 45 }
78 RPWVIDEO.C 848 | |->clear_scr { 19 }

271 CDATE.C 849 | |->get_version_days
| |
93 CDATE.C 850 | | | |->getdate
| |
179 CDATE.C 851 | | | | |->getjdate
| |
30 CDATE.C 852 | | | | | |->isLeapYear
853 | | | | | |..assert
| |
854 | | | | | |..assert
| |
855 | | | ..fopen fgets atoi strncpy sprintf fclose

484 RPWMENU.C 856 |->Disp_Text { 89 }

258 CDATE.C 857 |->get_curent_days
93 CDATE.C 858 | |->getdate { 850 }
859 | |..time localtime
| |
120 RPWVIDEO.C 860 | |->prtstr { 3 }
212 RPWVIDEO.C 861 | |->setcur { 17 }
862 | ..getch sprintf
| |
1863 RPWENTRY.C 863 |->load_message
484 RPWMENU.C 864 | |->Disp_Text { 89 }
262 RPWVIDEO.C 865 | |->shade_it { 60 }

93 CLIMIT.C 866 |->load_code_limit
| |
75 CLIMIT.C 867 | |->rec_count
868 | | ..ftell fgets fseek
| |
82 MAINRATE.C 869 | |->etoi { 116 }
870 | ..fopen puts calloc fgets strlen memcpy fclose
| |
1194 RPWENTRY.C 871 |->init_data_records
872 | |..memset
| |
179 MAINRATE.C 873 |->load_screen_buffer
874 | |..fopen handle_error2a fseek fgets strlen
fclose
| |
43 MAINRATE.C 875 |->load_rate_table
| |
98 MAINRATE.C 876 | |->handle_error2A

```

```

120 RPWVIDEO.C      877 |   | |->prtstr { 3 }
121 ASIPORTS.H      878 |   | |->exit { 21 }
122                   879 |   | ..printf getch puts
123
82 MAINRATE.C      880 |   | |->etoi { 116 }
124                   881 |   | ..fopen fseek fclose fread
125
77 RPWTABLE.C      882 |   |->table_load
126 45 RPWTABLE.C   883 |   |->Init_Table_Info { 119 }
127 1219 RPWENTRY.C 884 |   |->handle_error2 { 6 }
128
123 RPWTABLE.C      885 |   |->table_read
129                   886 |   | ..memchr lseek read memset memcpy min
130
131                   887 |   | ..open strcpy strcat calloc lseek close
132
57 RPWTABLE.C      888 |   |->table_get { 121 }
133
1106 RPWENTRY.C    889 |   |->update_ss_desc
134
1090 RPWENTRY.C    890 |   |->get_rate
135 82 MAINRATE.C   891 |   | |->etoi { 116 }
136
137                   892 |   | ..sprintf memcpy memcmp
138
47 LOOKUP.C         893 |   |->init_zip_lookup
139 29 CITYZIP.C     894 |   |->ptr_ziphelffile { 15 }
140
30 STATECOD.C       895 |   |->ptr_statefile
141
38 CITYZIP.C        896 |   |->init_ziphelffile
142 29 CITYZIP.C     897 |   | |->ptr_ziphelffile { 15 }
143
144                   898 |   | ..access strcpy open lseek read malloc
145
146 alloc
147                   |   | ..close free
148
35 STATECOD.C       899 |   |->init_state_file
149
89 STATECOD.C       900 |   | |->state_read
150
127 STATECOD.C      901 |   | | |->bytecopy
151                   902 |   | | ..memchr lseek read memset
152
130 CITYZIP.C        903 |   | |->CheckState
153 29 CITYZIP.C     904 |   | | |->ptr_ziphelffile { 15 }
154
155                   905 |   | | ..memcmp
156
157                   906 |   | ..open calloc lseek close
158
201 RPWVIDEO.C      907 |   |->cursor_off { 45 }
159 723 RPWLIB.C      908 |   |->date_time { 27 }
160
161
184 RPWVIDEO.C      909 |   |->pop_scrn2
162 120 RPWVIDEO.C   910 |   |->prtstr { 3 }
163 167 RPWLIB.C     911 |   |->num_input { 97 }
164
165
130 REVSSAMP.C      912 |   |->selsamp

```

```

372 RPWVIDEO.C    913 | |->clr_scrn
120 RPWVIDEO.C    914 | |->prtstr { 3 }
387 RPWVIDEO.C    915 | |->create_box { 132 }

530 REVAMP.C      916 | |->how_big_index
789 REVAMP.C      917 | | |->handle_an_error
120 RPWVIDEO.C    918 | | | |->prtstr { 3 }
120 RPWVIDEO.C    919 | | | ..printf getch

920 | | ..fopen filelength fileno

472 REVAMP.C      921 | |->allocate_mem
789 REVAMP.C      922 | | |->handle_an_error { 917 }
789 REVAMP.C      923 | | ..calloc

800 REVAMP.C      924 | |->do_exit
800 REVAMP.C      925 | | ..fcloseall free

547 REVAMP.C      926 | |->read_in_index
547 REVAMP.C      927 | | ..fread fclose

484 REVAMP.C      928 | |->read_in_recs
120 RPWVIDEO.C    929 | | |->prtstr { 3 }
789 REVAMP.C      930 | | |->handle_an_error { 917 }
789 REVAMP.C      931 | | ..fopen memcpy fseek fread memcmp fclose

316 RPWVIDEO.C    932 | |->clr_box { 32 }

70 REVAMP.C       933 | |->display_samp
120 RPWVIDEO.C    934 | | |->prtstr { 3 }
120 RPWVIDEO.C    935 | | ..strcpy memcmp memcpy

694 REVAMP.C      936 | |->get_date
120 RPWVIDEO.C    937 | | |->prtstr { 3 }
212 RPWVIDEO.C    938 | | |->setcur { 17 }
224 RPWVIDEO.C    939 | | |->disp_cur { 159 }
201 RPWVIDEO.C    940 | | |->cursor_off { 45 }
201 RPWVIDEO.C    941 | | ..strcpy getch printf memcpy isdigit

942 | | ..memcpy getch

74 REVIEW.C        943 |->review_data
723 RPWLIB.C       944 | |->date_time { 27 }

1261 REVIEW.C     945 | |->handle_error4
159 RATEREV.C     946 | | |->box { 49 }
1070 RATEREV.C    947 | | |->gotoxy { 51 }
1550 RATEREV.C    948 | | |->writeca { 53 }
120 RPWVIDEO.C    949 | | |->prtstr { 3 }
262 RPWVIDEO.C    950 | | |->shade_it { 60 }
127 MAINRATE.C    951 | | |->mscgetch { 61 }
127 MAINRATE.C    952 | | ..strlen printf

1180 RATEREV.C    953 | |->prep_exit { 65 }
131 RATEREV.C     954 | |->handle_error3 { 48 }

```

```

1162 REVIEW.C      955 |     |->get_session
131 RATEREV.C    956 |     |     |->handle_error3 { 48 }
957 |     |     |..strcpy strcat _dos_open filelength malloc
958 |     |     |.._dos_read

1899 REVIEW.C      958 |     |->size_data
959 |     |     |     |->erase_key
960 |     |     |     |->gotoxy { 51 }
961 |     |     |     |->writeca { 53 }

120 RPWVIDEO.C    962 |     |->prtstr { 3 }

78 RPWVIDEO.C     963 |     |->clear_scr { 19 }

1085 REVIEW.C     964 |     |->get_onesesn
131 RATEREV.C    965 |     |     |->handle_error3 { 48 }
120 RPWVIDEO.C    966 |     |     |->prtstr { 3 }
127 MAINRATE.C   967 |     |     |->mscgetch { 61 }
968 |     |     |..itoa malloc memcmp _dos_read free

1841 REVIEW.C     969 |     |->set_keys
1750 REVIEW.C     970 |     |->scrn_hndlr
1448 REVIEW.C     971 |     |     |->process_key
1361 REVIEW.C     972 |     |     |     |->movbar
1070 RATEREV.C   973 |     |     |     |->gotoxy { 51 }
1550 RATEREV.C   974 |     |     |     |->writeca { 53 }

445 REVIEW.C      975 |     |     |->delete_rec
120 RPWVIDEO.C   976 |     |     |->prtstr { 3 }

374 RATEREV.C     977 |     |     |     |->getkey
978 |     |     |     |.._bios_keybrd toupper

131 RATEREV.C     979 |     |     |->handle_error3 { 48 }
1180 RATEREV.C   980 |     |     |->prep_exit { 65 }

1954 REVIEW.C     981 |     |     |->size_idx
982 |     |     |     |..memcpy memcmp

1899 REVIEW.C     983 |     |     |->size_data { 958 }
78 RPWVIDEO.C    984 |     |     |->clear_scr { 19 }
127 MAINRATE.C   985 |     |     |->mscgetch { 61 }
1070 RATEREV.C   986 |     |     |->gotoxy { 51 }
1550 RATEREV.C   987 |     |     |->writeca { 53 }
1034 REVIEW.C     988 |     |     |->erase_key { 959 }

1311 REVIEW.C     989 |     |     |->key_info
120 RPWVIDEO.C   990 |     |     |->prtstr { 3 }

991 |     |     |..strcpy strcat open write lseek
|     |     |.._dos_write memcmp memcpy

```

```

    60 RATEREV.C      992 | | | |->raterev { 44 }
    1034 REVIEW.C    993 | | | |->erase_key { 959 }
    120 RPWVIDEO.C   994 | | | |->prtstr { 3 }
    995 | | | ..memcmp memcpy

    364 REVIEW.C     996 | | | |->databak

    1208 REVIEW.C    997 | | | |->get_samplerec
    131 RATEREV.C    998 | | | |->handle_error3 { 48 }
    999 | | | ..filelength malloc lseek free
_dos_read
    | | | ..memcpy lfind

    120 RPWVIDEO.C   1000 | | | |->prtstr { 3 }

    1683 REVIEW.C    1001 | | | |->prtfld
    142 RPWVIDEO.C   1002 | | | |->prtbuff { 2 }
    1070 RATEREV.C   1003 | | | |->gotoxy { 51 }
    1550 RATEREV.C   1004 | | | |->writeca { 53 }
    1005 | | | ..itoa strlen

    706 REVIEW.C     1006 | | | |->disp_blowup
    120 RPWVIDEO.C   1007 | | | |->prtstr { 3 }
    1683 REVIEW.C    1008 | | | |->prtfld { 1001 }
    1009 | | | ..memcmp

    120 RPWVIDEO.C   1010 | | | |->prtstr { 3 }

    1813 REVIEW.C    1011 | | | |->sessionbak
    120 RPWVIDEO.C   1012 | | | |->prtstr { 3 }
    1070 RATEREV.C   1013 | | | |->gotoxy { 51 }
    1550 RATEREV.C   1014 | | | |->writeca { 53 }

    1311 REVIEW.C    1015 | | | |->key_info { 989 }

    928 REVIEW.C     1016 | | | |->disp_sum
    1034 REVIEW.C    1017 | | | |->erase_key { 959 }
    120 RPWVIDEO.C   1018 | | | |->prtstr { 3 }
    1070 RATEREV.C   1019 | | | |->gotoxy { 51 }
    1550 RATEREV.C   1020 | | | |->writeca { 53 }

    331 REVIEW.C     1021 | | | |->cvto
    1022 | | | ..ltoa strlen

    1023 | | | ..memcpy memcmp atol atof floor fmod ltoa
    | | | ..strlen fcvt

    717 REVIEW.C     1024 | | | |->disp_cathdr
    69 HASH.C        1025 | | | |->hash { 821 }
    131 RATEREV.C   1026 | | | |->handle_error3 { 48 }

    116 MAINRATE.C   1027 | | | |->prtlin
    120 RPWVIDEO.C   1028 | | | |->prtstr { 3 }
    1029 | | | ..lseek _dos_read memcmp

    78 RPWVIDEO.C    1030 | | | |->clear_scr { 19 }

```

```

    ☐ 1708 REVIEW.C      1031 | | | ->revscrn
    ☐ 120 RPWVIDEO.C    1032 | | | | ->prtstr { 3 }
    ☐ 1070 RATEREV.C    1033 | | | | ->gotoxy { 51 }
    ☐ 1550 RATEREV.C    1034 | | | | ->writeca { 53 }

    ☐ 765 REVIEW.C      1035 | | | | ->disprecdta
    ☐ 1683 REVIEW.C     1036 | | | | | ->prtfld { 1001 }
    ☐ 120 RPWVIDEO.C    1037 | | | | | ->prtstr { 3 }
    ☐ 331 REVIEW.C      1038 | | | | | ->cvto { 1021 }

    ☐ 1191 REVIEW.C     1039 | | | | | ->get_spcat
    ☐
    ☐ 201 RPWTABLE.C    1040 | | | | | ->table_bsearch
    ☐ 45 RPWTABLE.C     1041 | | | | | | ->Init_Table_Info { 119 }
    ☐ 1042 | | | | | | ..bsearch

    ☐ 57 RPWTABLE.C     1043 | | | | | ->table_get { 121 }

    ☐ 116 MAINRATE.C    1044 | | | | | ->prtlin { 1027 }
    ☐ 1045 | | | | | ..memcpy atoi memcmp atol isdigit

    ☐ 876 REVIEW.C      1046 | | | | | ->disprecsen
    ☐ 120 RPWVIDEO.C    1047 | | | | | | ->prtstr { 3 }
    ☐ 1683 REVIEW.C     1048 | | | | | | ->prtfld { 1001 }
    ☐ 1049 | | | | | | ..itoa memcpy atoi strlen

    ☐ 1050 | | | | | ..itoa

    ☐ 374 RATEREV.C     1051 | | | | | ->getkey { 977 }

    ☐ 120 RPWVIDEO.C    1052 | | | | ->prtstr { 3 }

    ☐ 1138 REVIEW.C     1053 | | | | ->get_seq
    ☐
    ☐ 271 REVIEW.C      1054 | | | | ->boxdiv
    ☐
    ☐ 300 REVIEW.C      1055 | | | | | ->boxtop2
    ☐ 1070 RATEREV.C    1056 | | | | | | ->gotoxy { 51 }
    ☐ 1550 RATEREV.C    1057 | | | | | | ->writeca { 53 }

    ☐ 1070 RATEREV.C    1058 | | | | | ->gotoxy { 51 }
    ☐ 1550 RATEREV.C    1059 | | | | | ->writeca { 53 }

    ☐ 120 RPWVIDEO.C    1060 | | | | | ->prtstr { 3 }
    ☐ 1070 RATEREV.C    1061 | | | | | ->gotoxy { 51 }
    ☐ 1550 RATEREV.C    1062 | | | | | ->writeca { 53 }
    ☐ 374 RATEREV.C     1063 | | | | | ->getkey { 977 }

    ☐ 1043 REVIEW.C     1064 | | | | ->get_data
    ☐ 131 RATEREV.C     1065 | | | | | ->handle_error3 { 48 }
    ☐ 1066 | | | | | ..malloc lseek _dos_read

    ☐ 1994 REVIEW.C     1067 | | | | ->sort_data
    ☐
    ☐ 1324 REVIEW.C     1068 | | | | ->makcatkeys
    ☐ 131 RATEREV.C     1069 | | | | | ->handle_error3 { 48 }
    ☐ 1070 | | | | | ..malloc

```

```

1          | | |
1          | | |..qsort
1
1 1954 REVIEW.C 1071 | | |..qsort
1
1 159 RATEREV.C 1072 | |->size_idx { 981 }
1
1 127 MAINRATE.C 1073 | |->box { 49 }
1
1
1 1282 RPWENTRY.C 1074 | |->mscgetch { 61 }
1
1 212 RPWVIDEO.C 1075 | |..strcpy strcat access _dos_open malloc free
1
1
1 268 REVSSAMP.C 1076 | |->remote_exit_routine
1
1 372 RPWVIDEO.C 1077 | |->setcur { 17 }
1
1 78 RPWVIDEO.C 1078 | |->clear_scr { 19 }
1
1 211 ASIPORTS.H 1079 | |->exit { 21 }
1
1
1 268 REVSSAMP.C 1080 | |..free _asiexit
1
1
1 530 REVSSAMP.C 1081 | |->revssamp
1
1 472 REVSSAMP.C 1082 | |->clr_scrn { 913 }
1
1 800 REVSSAMP.C 1083 | |->how_big_index { 916 }
1
1 547 REVSSAMP.C 1084 | |->allocate_mem { 921 }
1
1 484 RPWMENU.C 1085 | |->do_exit { 924 }
1
1 484 RPWMENU.C 1086 | |->read_in_index { 926 }
1
1 262 RPWVIDEO.C 1087 | |->Disp_Text { 89 }
1
1 694 REVSSAMP.C 1088 | |->shade_it { 60 }
1
1 316 RPWVIDEO.C 1089 | |->get_date { 936 }
1
1 120 RPWVIDEO.C 1090 | |->clr_box { 32 }
1
1 484 REVSSAMP.C 1091 | |->prtstr { 3 }
1
1 484 REVSSAMP.C 1092 | |->read_in_recs { 928 }
1
1
1 556 REVSSAMP.C 1093 | |->display_data
1
1 316 RPWVIDEO.C 1094 | | |->clr_box { 32 }
1
1 120 RPWVIDEO.C 1095 | | |->prtstr { 3 }
1
1 142 RPWVIDEO.C 1096 | | |->prtbuff { 2 }
1
1 82 MAINRATE.C 1097 | | |->etoi { 116 }
1
1
1 809 REVSSAMP.C 1098 | | |->print_header
1
1 809 REVSSAMP.C 1099 | | |..fprintf fflush
1
1
1 809 REVSSAMP.C 1100 | | |..strcpy memcmp getch memcpy fprintf fflush
1
1 809 REVSSAMP.C 1100 | | |..itoa memset
1
1
1 212 RPWVIDEO.C 1101 | | |..memcpy getch printf fprintf fflush
1
1
1 212 RPWVIDEO.C 1102 | |->setcur { 17 }
1
1 1235 RPWENTRY.C 1103 | |->exit_routine { 8 }
1
1 60 RATEREV.C 1104 | |->raterev { 44 }
1
1
1 1325 RPWENTRY.C 1105 | |->header_input
1
1 723 RPWLIB.C 1106 | |->date_time { 27 }
1
1 184 RPWVIDEO.C 1107 | |->pop_scrn2 { 909 }
1
1 120 RPWVIDEO.C 1108 | |->prtstr { 3 }
1
1
1 282 RPWLIB.C 1109 | |->char_field
1
1 212 RPWVIDEO.C 1110 | | |->setcur { 17 }
1
1 224 RPWVIDEO.C 1111 | | |->disp_cur { 159 }
1
1 120 RPWVIDEO.C 1112 | | |->prtstr { 3 }
1
1 723 RPWLIB.C 1113 | | |->date_time { 27 }
1
1 201 RPWVIDEO.C 1114 | | |->cursor_off { 45 }
1
1 113 RPWLIB.C 1115 | | |->clear_message { 31 }

```

```

1116 | | | ..strlen kbhit getch printf isdigit isalpha
1117 | | | |->Corrected_Sample
1118 | | | |->get_curent_days { 857 }
1119 | | | |->getdate { 850 }
1120 | | | |->prtstr { 3 }
1121 | | | |->yes_no { 164 }
1122 | | | |->clr_box { 32 }
1123 | | | ..strncpy atoi

60 RATEREV.C 1124 |->raterev { 44 }

449 RPWLIB.C 1125 |->date_field
212 RPWVIDEO.C 1126 | |->setcur { 17 }
224 RPWVIDEO.C 1127 | |->disp_cur { 159 }
120 RPWVIDEO.C 1128 | |->prtstr { 3 }
723 RPWLIB.C 1129 | |->date_time { 27 }
113 RPWLIB.C 1130 | |->clear_message { 31 }
201 RPWVIDEO.C 1131 | |->cursor_off { 45 }
1132 | | ..strlen kbhit getch printf isdigit

1756 RPWENTRY.C 1133 |->date_check
1134 | | ..strchr atoi

1550 RPWENTRY.C 1135 |->get_sample_record
201 RPWVIDEO.C 1136 | |->cursor_off { 45 }
120 RPWVIDEO.C 1137 | |->prtstr { 3 }
723 RPWLIB.C 1138 | |->date_time { 27 }
184 RPWVIDEO.C 1139 | |->pop_scrn2 { 909 }
747 RPWLIB.C 1140 | |->yes_no { 164 }
316 RPWVIDEO.C 1141 | |->clr_box { 32 }
82 MAINRATE.C 1142 | |->etoi { 116 }

1703 RPWENTRY.C 1143 | |->display_sample
316 RPWVIDEO.C 1144 | | |->clr_box { 32 }
120 RPWVIDEO.C 1145 | | |->prtstr { 3 }
142 RPWVIDEO.C 1146 | | |->prtbuff { 2 }

1746 RPWENTRY.C 1147 | |->parse_descr
1148 | | | ..memcmp

568 RPWENTRY.C 1149 |->next_backup

553 RPWENTRY.C 1150 |->delete_backup1
1151 | | | ..strcpy _dos_findfirst remove
| | | | .._dos_findnext

517 RPWENTRY.C 1152 |->rename_backup
1153 | | | ..sprintf _dos_findfirst strcpy strlen
| | | | ..rename _dos_findnext
1154 | | | ..sprintf _dos_findfirst strlen

2416 RPWENTRY.C 1155 |->display_incoming_msg
473 RPWVIDEO.C 1156 | | |->sav_wnd { 227 }
316 RPWVIDEO.C 1157 | | |->clr_box { 32 }
387 RPWVIDEO.C 1158 | | |->create_box { 132 }

```

```

□ 484 RPWMENU.C    1159 | | | |->Disp_Text { 89 }
□ 288 RPWVIDEO.C  1160 | | | |->restor_wnd { 144 }
□
□ 2375 RPWENTRY.C 1161 | | | ..malloc getch free
□
□ 2375 RPWENTRY.C 1162 | | | |->originating_Question
□ 473 RPWVIDEO.C  1163 | | | |->sav_wnd { 227 }
□ 316 RPWVIDEO.C  1164 | | | |->clr_box { 32 }
□ 387 RPWVIDEO.C  1165 | | | |->create_box { 132 }
□ 120 RPWVIDEO.C  1166 | | | |->prtstr { 3 }
□ 747 RPWLIB.C   1167 | | | |->yes_no { 164 }
□ 288 RPWVIDEO.C 1168 | | | |->restor_wnd { 144 }
□
□ 2375 RPWENTRY.C 1169 | | | ..malloc printf free
□
□
□ 1926 RPWENTRY.C 1170 | | | ..memcpy fopen fseek fcloseall fread memcmp
□
□ 1889 RPWENTRY.C 1171 | | | |->get_session_number
□
□ 599 RPWENTRY.C  1172 | | | |->get_backup_session_number
□
□ 599 RPWENTRY.C  1173 | | | |->backup_filename
□ 82 MAINRATE.C   1174 | | | | ..sprintf
□
□ 1219 RPWENTRY.C 1175 | | | |->handle_error2 { 6 }
□ 82 MAINRATE.C   1176 | | | |->etoi { 116 }
□
□ 1219 RPWENTRY.C 1177 | | | ..fopen filelength fileno fseek fclose
fread
□
□ 1219 RPWENTRY.C 1178 | | | |->handle_error2 { 6 }
□ 82 MAINRATE.C   1179 | | | |->etoi { 116 }
□
□ 1219 RPWENTRY.C 1180 | | | ..strcat fopen filelength fileno fseek
fclose
□
□ 1838 RPWENTRY.C 1181 | | | ..fread
□
□ 1838 RPWENTRY.C 1182 | | | |->load_zone_structure
□ 131 RATEREV.C   1183 | | | |->handle_error3 { 48 }
□
□ 142 RPWVIDEO.C  1184 | | | ..strcpy fopen fseek fread
□
□ 1746 RPWENTRY.C 1185 | | | |->prtbuff { 2 }
□ 1746 RPWENTRY.C 1186 | | | |->parse_descr { 1147 }
□
□ 2169 RPWENTRY.C 1187 | | | |->current_session_question
□ 387 RPWVIDEO.C  1188 | | | |->create_box { 132 }
□ 120 RPWVIDEO.C  1189 | | | |->prtstr { 3 }
□ 747 RPWLIB.C   1190 | | | |->yes_no { 164 }
□
□ 936 RPWENTRY.C  1191 | | | ..printf
□
□ 316 RPWVIDEO.C  1192 | | | |->subsample_question
□ 387 RPWVIDEO.C  1193 | | | |->clr_box { 32 }
□ 120 RPWVIDEO.C  1194 | | | |->create_box { 132 }
□ 484 RPWMENU.C   1195 | | | |->prtstr { 3 }
□ 167 RPWLIB.C   1196 | | | |->Disp_Text { 89 }
□ 60 RATEREV.C   1197 | | | |->num_input { 97 }
□
□ 60 RATEREV.C   1198 | | | |->raterev { 44 }
□

```

```

□ 2199 RPWENTRY.C    1199 | |->get_counted_skip
□ 120 RPWVIDEO.C    1200 | | |->prtstr { 3 }
□ 387 RPWVIDEO.C    1201 | | |->create_box { 132 }
□ 316 RPWVIDEO.C    1202 | | |->clr_box { 32 }
□ 484 RPWMENU.C     1203 | | |->Disp_Text { 89 }
□
□ 2355 RPWENTRY.C    1204 | | |->get_start_info_element_num
□
□ 20 RANDOM.C        1205 | | |..memcmp
□
□ 1001 RPWLIB.C      1206 | | |->get_random { 238 }
□
□ 343 RPWVIDEO.C     1207 | | |->generic_scroll_bar
□ 343 RPWVIDEO.C     1208 | | |->hilite { 655 }
□
□ 357 RPWVIDEO.C     1209 | | |..getch
□
□ 2331 RPWENTRY.C    1210 | | |->grab_scr_char { 656 }
□ 316 RPWVIDEO.C     1211 | | |..itoa strlen strcmp
□
□ 449 RPWVIDEO.C     1212 | | |->get_container_skip
□ 316 RPWVIDEO.C     1213 | | |->clr_box { 32 }
□
□ 449 RPWVIDEO.C     1214 | | |->pop_wnd
□
□ 1105 RPWLIB.C      1215 | | |->split_scroll_bar
□ 120 RPWVIDEO.C     1216 | | | |->prtstr { 3 }
□ 343 RPWVIDEO.C     1217 | | | |->hilite { 655 }
□ 2355 RPWENTRY.C    1218 | | | |->get_start_info_element_num { 1204 }
□ 20 RANDOM.C        1219 | | | |->get_random { 238 }
□ 60 RATEREV.C       1220 | | | |->raterev { 44 }
□
□ 609 RPWENTRY.C     1221 | | | |..itoa getch
□
□ 609 RPWENTRY.C     1222 | | | ..printf getch
□
□ 316 RPWVIDEO.C     1223 |->scale_question
□ 316 RPWVIDEO.C     1224 | |->clr_box { 32 }
□ 120 RPWVIDEO.C     1225 | |->prtstr { 3 }
□ 387 RPWVIDEO.C     1226 | |->create_box { 132 }
□ 747 RPWLIB.C       1227 | |->yes_no { 164 }
□
□ 178 RPWENTR1.C     1228 | |..printf
□
□ 178 RPWENTR1.C     1229 |->data_entry
□ 201 RPWVIDEO.C     1230 | |->cursor_off { 45 }
□ 184 RPWVIDEO.C     1231 | |->pop_scrn2 { 909 }
□ 120 RPWVIDEO.C     1232 | |->prtstr { 3 }
□ 142 RPWVIDEO.C     1233 | |->prtbuff { 2 }
□ 1448 RPWENTR1.C    1234 | |->clr_entry_scrn { 35 }
□ 630 RPWENTR2.C     1235 | |->display_record_count { 41 }
□
□ 3964 RPWENTR1.C    1236 | |->get_orig_test_type
□ 316 RPWVIDEO.C     1237 | | |->clr_box { 32 }
□ 484 RPWMENU.C      1238 | | |->Disp_Text { 89 }
□ 167 RPWLIB.C       1239 | | |->num_input { 97 }
□ 60 RATEREV.C       1240 | | |->raterev { 44 }
□
□ 1269 RPWENTR1.C    1241 | | |..printf
□
□ 1269 RPWENTR1.C    1242 |->pop_list11
□ 316 RPWVIDEO.C     1243 | | |->clr_box { 32 }

```

```

□ 484 RPWMENU.C      1244 | | |->Disp_Text { 89 }
□ 167 RPWLIB.C       1245 | | |->num_input { 97 }
□ 120 RPWVIDEO.C    1246 | | |->prtstr { 3 }
□ 630 RPWENTR2.C    1247 | | |->display_record_count { 41 }
□ 452 RPWENTR1.C    1248 | | |->add_mailpiece { 104 }
□ 821 RPWENTR2.C    1249 | | |->where_to_go { 224 }
□
□
□ 1250 | | |..memcpy printf
□
□ 1323 RPWENTR1.C   1251 | | |->pop_list12
□ 316 RPWVIDEO.C    1252 | | |->clr_box { 32 }
□ 484 RPWMENU.C    1253 | | |->Disp_Text { 89 }
□ 167 RPWLIB.C     1254 | | |->num_input { 97 }
□ 120 RPWVIDEO.C    1255 | | |->prtstr { 3 }
□ 630 RPWENTR2.C    1256 | | |->display_record_count { 41 }
□ 452 RPWENTR1.C    1257 | | |->add_mailpiece { 104 }
□ 821 RPWENTR2.C    1258 | | |->where_to_go { 224 }
□
□ 1259 | | |..memcpy printf
□
□ 506 RPWENTR1.C    1260 | | |->pop_list1 { 422 }
□
□ 712 RPWENTR1.C    1261 | | |->pop_list2
□ 316 RPWVIDEO.C    1262 | | |->clr_box { 32 }
□ 484 RPWMENU.C    1263 | | |->Disp_Text { 89 }
□ 167 RPWLIB.C     1264 | | |->num_input { 97 }
□ 452 RPWENTR1.C    1265 | | |->add_mailpiece { 104 }
□ 821 RPWENTR2.C    1266 | | |->where_to_go { 224 }
□
□ 1267 | | |..printf
□
□ 770 RPWENTR1.C    1268 | | |->pop_list3
□ 316 RPWVIDEO.C    1269 | | |->clr_box { 32 }
□ 120 RPWVIDEO.C    1270 | | |->prtstr { 3 }
□ 484 RPWMENU.C    1271 | | |->Disp_Text { 89 }
□ 167 RPWLIB.C     1272 | | |->num_input { 97 }
□ 452 RPWENTR1.C    1273 | | |->add_mailpiece { 104 }
□ 83 RPWLIB.C       1274 | | |->handle_error { 30 }
□ 821 RPWENTR2.C    1275 | | |->where_to_go { 224 }
□
□ 1276 | | |..printf strncmp memset
□
□ 894 RPWENTR1.C    1277 | | |->pop_list4
□ 316 RPWVIDEO.C    1278 | | |->clr_box { 32 }
□ 484 RPWMENU.C    1279 | | |->Disp_Text { 89 }
□ 167 RPWLIB.C     1280 | | |->num_input { 97 }
□ 452 RPWENTR1.C    1281 | | |->add_mailpiece { 104 }
□ 821 RPWENTR2.C    1282 | | |->where_to_go { 224 }
□
□ 1283 | | |..printf
□
□ 944 RPWENTR1.C    1284 | | |->pop_list5
□ 316 RPWVIDEO.C    1285 | | |->clr_box { 32 }
□ 484 RPWMENU.C    1286 | | |->Disp_Text { 89 }
□ 167 RPWLIB.C     1287 | | |->num_input { 97 }
□ 452 RPWENTR1.C    1288 | | |->add_mailpiece { 104 }
□ 821 RPWENTR2.C    1289 | | |->where_to_go { 224 }
□
□ 1290 | | |..printf
□
□ 1565 RPWENTR1.C   1291 | | |->get_pieces
□ 149 RPWLIB.C      1292 | | |->clear_DE_box { 85 }
□ 113 RPWLIB.C      1293 | | |->clear_message { 31 }

```

```

120 RPVIDEO.C 1294 | | |->prtstr { 3 }
600 RPWLIB.C 1295 | | |->num_field { 156 }
452 RPWENTR1.C 1296 | | |->add_mailpiece { 104 }
433 RPWENTR1.C 1297 | | |->remove_mailpiece { 271 }
480 RPWENTR1.C 1298 | | |->print_cat { 108 }
1299 | | ..strcpy memcpy printf memcmp

1831 RPWENTR1.C 1300 | |->get_weight
316 RPVIDEO.C 1301 | |->clr_box { 32 }
484 RPWMENU.C 1302 | |->Disp_Text { 89 }
104 RPWLIB.C 1303 | |->erase_error { 234 }
31 RPWCOMM.C 1304 | |->ask_scale { 505 }

2038 RPWENTR1.C 1305 | |->prelim_wt_edits
1874 RPWENTR2.C 1306 | |->edit_wt_fields
1307 | | | ..atoi atof strcpy strchr strcat

83 RPWLIB.C 1308 | |->handle_error { 30 }
1309 | | ..memcpy atoi atof printf

452 RPWENTR1.C 1310 | |->add_mailpiece { 104 }
1311 | | ..memcpy strncmp strcpy getch atoi atof

2297 RPWENTR1.C 1312 | |->get_std
316 RPVIDEO.C 1313 | |->clr_box { 32 }
484 RPWMENU.C 1314 | |->Disp_Text { 89 }
167 RPWLIB.C 1315 | |->num_input { 97 }
452 RPWENTR1.C 1316 | |->add_mailpiece { 104 }

2392 RPWENTR1.C 1317 | |->get_cat_code
175 RPWTABLE.C 1318 | |->table_search { 118 }
57 RPWTABLE.C 1319 | |->table_get { 121 }
480 RPWENTR1.C 1320 | |->print_cat { 108 }
83 RPWLIB.C 1321 | |->handle_error { 30 }

124 RPWREVNU.C 1322 | |->revenue
120 RPVIDEO.C 1323 | |->prtstr { 3 }
149 RPWLIB.C 1324 | |->clear_DE_box { 85 }
113 RPWLIB.C 1325 | |->clear_message { 31 }

168 RPWREVNU.C 1326 | |->rate_1910
120 RPVIDEO.C 1327 | | |->prtstr { 3 }
104 RPWREVNU.C 1328 | | |->print_rate { 649 }
850 RPWLIB.C 1329 | | |->scroll_bar { 654 }
84 RPWREVNU.C 1330 | | |->rate_screen_reset { 660 }
433 RPWENTR1.C 1331 | | |->remove_mailpiece { 271 }

218 RPWREVNU.C 1332 | |->first_pre_rates
120 RPVIDEO.C 1333 | | |->prtstr { 3 }
104 RPWREVNU.C 1334 | | |->print_rate { 649 }
850 RPWLIB.C 1335 | | |->scroll_bar { 654 }
84 RPWREVNU.C 1336 | | |->rate_screen_reset { 660 }
452 RPWENTR1.C 1337 | | |->add_mailpiece { 104 }
433 RPWENTR1.C 1338 | | |->remove_mailpiece { 271 }
1339 | | ..memcpy

```

```

□ 489 RPWREVNU.C 1340 | | |->first_rates
□ 120 RPWVIDEO.C 1341 | | | |->prtstr { 3 }
□ 104 RPWREVNU.C 1342 | | | |->print_rate { 649 }
□ 850 RPWLIB.C 1343 | | | |->scroll_bar { 654 }
□ 84 RPWREVNU.C 1344 | | | |->rate_screen_reset { 660 }
□ 452 RPWENTR1.C 1345 | | | |->add_mailpiece { 104 }
□ 433 RPWENTR1.C 1346 | | | |->remove_mailpiece { 271 }
□ 1347 | | | ..memcpy
□
□
□ 1348 | | | ..memcmp
□
□ 1349 | | | ..memcpy
□
□ 2137 RPWENTRY.C 1350 |->abort_question
□ 387 RPWVIDEO.C 1351 | |->create_box { 132 }
□ 120 RPWVIDEO.C 1352 | |->prtstr { 3 }
□ 747 RPWLIB.C 1353 | |->yes_no { 164 }
□ 1354 | ..printf
□
□ 2045 RPWENTRY.C 1355 |->abort_session
□ 723 RPWLIB.C 1356 | |->date_time { 27 }
□
□ 1969 RPWENTRY.C 1357 | |->abort_backup_session
□ 599 RPWENTRY.C 1358 | | |->backup_filename { 1173 }
□ 83 RPWLIB.C 1359 | | |->handle_error { 30 }
□ 1360 | | | ..strcpy itoa strlen memcpy open lseek close
□ | | | ..write read
□
□ 83 RPWLIB.C 1361 | |->handle_error { 30 }
□ 82 MAINRATE.C 1362 | |->etoi { 116 }
□ 1363 | | ..memcpy strcpy itoa strlen strcat open lseek
□ | | | ..close write read
□
□ 1794 RPWENTRY.C 1364 |->end_session
□ 2355 RPWENTRY.C 1365 | |->get_start_info_element_num { 1204 }
□ 1366 | ..memcpy
□
□ 1367 | .._dos_findfirst strcpy memcpy printf system
□
□ 28 SAMPSORT.C 1368 main?536?
□ 23 RPWVIDEO.C 1369 |->init_video { 844 }
□ 120 RPWVIDEO.C 1370 |->prtstr { 3 }
□
□ 49 SAMPSORT.C 1371 |->get_num_recs
□ 83 RPWLIB.C 1372 | |->handle_error { 30 }
□ 1373 | ..fopen filelength fileno
□
□ 80 SAMPSORT.C 1374 |->load_array
□ 83 RPWLIB.C 1375 | |->handle_error { 30 }
□ 120 RPWVIDEO.C 1376 | |->prtstr { 3 }
□ 1377 | ..calloc fseek fread memcpy itoa fclose
□
□ 107 SAMPSORT.C 1378 |->sort_it
□ 120 RPWVIDEO.C 1379 | |->prtstr { 3 }
□
□ 156 SAMPSORT.C 1380 | |->buffcmp
□ 1381 | | ..memcpy strcmp

```

```

S 107 SAMPSSORT.C    1382 | |->sort_it { RECURSVE }
S | |
S | 1383 | ...memcpy
S |
S 144 SAMPSSORT.C    1384 |->output_index
S 120 RPWVIDEO.C    1385 | |->prtstr { 3 }
S 83 RPWLIB.C        1386 | |->handle_error { 30 }
S | 1387 | ...fopen fwrite fclose
S |
S | 1388 | ...printf free
S |
S 30 MAINRATE.C      1389 main?82?
S 23 RPWVIDEO.C      1390 |->init_video { 844 }
S 43 MAINRATE.C      1391 |->load_rate_table { 875 }
S 60 RATEREV.C       1392 |->raterev { 44 }
S 212 RPWVIDEO.C     1393 |->setcur { 17 }
S |
S 13 POSTRPW.C       1394 main?97?
S | 1395 | ...fopen filelength fileno fclose system
S |
S 1399 REVIEW.C      1396 mscgetch?227?
S | 1397 | ...int86
S |
S 1409 REVIEW.C      1398 prep_exit?228?
S 131 RATEREV.C      1399 |->handle_error3 { 48 }
S | 1400 | ...free _dos_close close
S |
S 773 RPWREVNU.C     1401 priority_rates
S 82 MAINRATE.C      1402 |->etoi { 116 }
S |
S 238 RPWVIDEO.C     1403 prt_substr
S |
S 1697 REVIEW.C      1404 prtlin?230?
S |
S 122 CDATE.C        1405 putdate
S | |
S 221 CDATE.C        1406 |->putjdate
S 30 CDATE.C         1407 | |->isLeapYear { 852 }
S | 1408 | ...assert
S |
S 30 CDATE.C         1409 |->isLeapYear { 852 }
S |
S 62 CDATE.C         1410 |->isValidDate
S 30 CDATE.C         1411 | |->isLeapYear { 852 }
S |
S | 1412 | ...assert
S |
S 1452 RATEREV.C     1413 screen1
S | |
S 298 RATEREV.C      1414 |->disply_scrn
S 120 RPWVIDEO.C     1415 | |->prtstr { 3 }
S |
S 1114 RATEREV.C     1416 | |->makscrnbar
S 120 RPWVIDEO.C     1417 | | |->prtstr { 3 }
S | 1418 | | ...strnset strncpy
S |
S 199 RATEREV.C      1419 |->cvt

```

```

11          1420 | ..itoa strlen
12
13 120 RPWVIDEO.C 1421 |->prtstr { 3 }
14
15 492 RATEREV.C 1422 |->getrates
16 199 RATEREV.C 1423 | |->cvt { 1419 }
17 120 RPWVIDEO.C 1424 | |->prtstr { 3 }
18
19 374 RATEREV.C 1425 |->getkey { 977 }
20
21 1465 RATEREV.C 1426 screen1a
22 298 RATEREV.C 1427 |->disply_scrn { 1414 }
23 492 RATEREV.C 1428 |->getrates { 1422 }
24 374 RATEREV.C 1429 |->getkey { 977 }
25
26 1475 RATEREV.C 1430 screen2
27
28 1298 RATEREV.C 1431 |->process_zn
29 78 RPWVIDEO.C 1432 | |->clear_scr { 19 }
30
31 312 RATEREV.C 1433 | |->disply_scrennz
32 120 RPWVIDEO.C 1434 | | |->prtstr { 3 }
33
34 120 RPWVIDEO.C 1435 | |->prtstr { 3 }
35 1070 RATEREV.C 1436 | |->gotoxy { 51 }
36 1550 RATEREV.C 1437 | |->writeca { 53 }
37 1114 RATEREV.C 1438 | |->makscrnbar { 1416 }
38
39 994 RATEREV.C 1439 | |->getratezn
40 120 RPWVIDEO.C 1440 | | |->prtstr { 3 }
41 199 RATEREV.C 1441 | | |->cvt { 1419 }
42
43 199 RATEREV.C 1442 | |->cvt { 1419 }
44
45 945 RATEREV.C 1443 | |->getbpmfsp
46 199 RATEREV.C 1444 | | |->cvt { 1419 }
47 120 RPWVIDEO.C 1445 | | |->prtstr { 3 }
48 5 BPMRATE.C 1446 | | |->BPM_SP_Rate { 668 }
49
50 970 RATEREV.C 1447 | |->getbpmbulk
51 199 RATEREV.C 1448 | | |->cvt { 1419 }
52 120 RPWVIDEO.C 1449 | | |->prtstr { 3 }
53 39 BPMRATE.C 1450 | | |->BPM_Bulk_Rate { 680 }
54
55 1023 RATEREV.C 1451 | |->getrateznbmcmc
56 199 RATEREV.C 1452 | | |->cvt { 1419 }
57 120 RPWVIDEO.C 1453 | | |->prtstr { 3 }
58
59 1483 RATEREV.C 1454 screen3
60
61 1246 RATEREV.C 1455 |->process_sp
62 78 RPWVIDEO.C 1456 | |->clear_scr { 19 }
63
64 353 RATEREV.C 1457 | |->disply_sp
65 298 RATEREV.C 1458 | | |->disply_scrn { 1414 }
66 492 RATEREV.C 1459 | | |->getrates { 1422 }
67 199 RATEREV.C 1460 | | |->cvt { 1419 }

```

```

    □ 120 RPWVIDEO.C      1461 | | |->prtstr { 3 }

    □ 233 RATEREV.C      1462 | | |->disply_class3
    □ 120 RPWVIDEO.C      1463 | | |->prtstr { 3 }
    □ 298 RATEREV.C      1464 | | |->disply_scrn { 1414 }

    □ 524 RATEREV.C      1465 | | |->getrates3bulklet
    □ 156 MAINRATE.C     1466 | | | |->edit_rate { 25 }
    □ 138 MAINRATE.C     1467 | | | |->trim { 179 }
    □ 120 RPWVIDEO.C     1468 | | | |->prtstr { 3 }
    □ 1469 | | | ..strlen

    □ 579 RATEREV.C      1470 | | |->getrates3bulkletnp
    □ 156 MAINRATE.C     1471 | | | |->edit_rate { 25 }
    □ 138 MAINRATE.C     1472 | | | |->trim { 179 }
    □ 120 RPWVIDEO.C     1473 | | | |->prtstr { 3 }
    □ 1474 | | | ..strlen

    □ 633 RATEREV.C      1475 | | |->getrates3bulknon
    □ 156 MAINRATE.C     1476 | | | |->edit_rate { 25 }
    □ 138 MAINRATE.C     1477 | | | |->trim { 179 }
    □ 120 RPWVIDEO.C     1478 | | | |->prtstr { 3 }
    □ 1479 | | | ..strlen strcpy memcpy

    □ 749 RATEREV.C      1480 | | |->getrates3bulknonnnp
    □ 156 MAINRATE.C     1481 | | | |->edit_rate { 25 }
    □ 138 MAINRATE.C     1482 | | | |->trim { 179 }
    □ 120 RPWVIDEO.C     1483 | | | |->prtstr { 3 }
    □ 1484 | | | ..strlen strcpy memcpy

    □ 505 RATEREV.C      1485 | | |->getrates3
    □ 199 RATEREV.C      1486 | | | |->cvt { 1419 }
    □ 120 RPWVIDEO.C     1487 | | | |->prtstr { 3 }

    □ 1488 | | | ..strlen

    □ 260 RATEREV.C      1489 | | |->disply_ct
    □ 78 RPWVIDEO.C      1490 | | | |->clear_scr { 19 }
    □ 120 RPWVIDEO.C     1491 | | | |->prtstr { 3 }

    □ 1194 RATEREV.C     1492 | | |->prntc
    □ 116 MAINRATE.C     1493 | | | |->prtlin { 1027 }
    □ 1494 | | | ..strlen

    □ 387 RATEREV.C      1495 | | |->getkeymult
    □ 1496 | | | .._bios_keybrd toupper

    □ 1491 RATEREV.C     1497 screen4
    □ 1246 RATEREV.C     1498 |->process_sp { 1455 }

    □ 1499 RATEREV.C     1499 screen5
    □ 1246 RATEREV.C     1500 |->process_sp { 1455 }

    □ 1507 RATEREV.C     1501 screen6
    □ 298 RATEREV.C      1502 |->disply_scrn { 1414 }

    □ 454 RATEREV.C      1503 |->getrate4sp

```

```

□ 199 RATEREV.C      1504 | |->cvt { 1419 }
□ 120 RPVIDEO.C     1505 | |->prtstr { 3 }
|
□ 374 RATEREV.C     1506 |->getkey { 977 }
□
□ 1518 RATEREV.C     1507 screen7
□ 1298 RATEREV.C     1508 |->process_zn { 1431 }
□
□ 1526 RATEREV.C     1509 screen7a
□ 1298 RATEREV.C     1510 |->process_zn { 1431 }
□
□ 1534 RATEREV.C     1511 screen8
□ 1298 RATEREV.C     1512 |->process_zn { 1431 }
□
□ 1542 RATEREV.C     1513 screen9
□ 1298 RATEREV.C     1514 |->process_zn { 1431 }
□
□ 95 RPVIDEO.C       1515 scroll_scr
□
□ 1892 REVIEW.C      1516 |..int86
□
□ 2005 REVIEW.C      1517 set_unitname
□
□ 2960 RPWENTR2.C    1518 spcomp
□
□ 316 RPVIDEO.C      1519 |..memcmp
□
□ 2960 RPWENTR2.C    1520 sub_wt_ok
□ 316 RPVIDEO.C      1521 |->clr_box { 32 }
□ 120 RPVIDEO.C      1522 |->prtstr { 3 }
|
□ 3152 RPWENTR2.C    1523 |->show_sub_weights
□ 484 RPWMENU.C      1524 | |->Disp_Text { 89 }
□ 120 RPVIDEO.C      1525 | |->prtstr { 3 }
□
□ 747 RPWLIB.C       1526 | ..ltoa gcvt strlen
|
□ 104 RPWLIB.C       1527 |->yes_no { 164 }
□ 83 RPWLIB.C        1528 |->erase_error { 234 }
□ 599 RPWENTRY.C     1529 |->handle_error { 30 }
□ 82 MAINRATE.C      1530 |->backup_filename { 1173 }
□
□ 222 RPWTABLE.C     1531 |->etoi { 116 }
□
□ 222 RPWTABLE.C     1532 |..memcpy fcvt strlen strcat open lseek close write
□
□ 810 RPWREVNU.C     1533 table_comp
□
□ 810 RPWREVNU.C     1534 |..memcmp
□
□ 2279 RPWENTR1.C    1535 third_rates
□
□ 485 RPWSS.C        1536 trim?381?
□
□ 485 RPWSS.C        1537 |..strlen strcpy
□
□ 157 CITYZIP.C       1538 Find_SS
□
□ 157 CITYZIP.C       1539 |..strlen strnicmp
□
□ 29 CITYZIP.C        1540 LoadOneState
□
□ 29 CITYZIP.C        1541 |->ptr_ziphelpfile { 15 }
□ 130 CITYZIP.C       1542 |->CheckState { 903 }
□
□ 1543 |..memcmp lseek _dos_read

```

```

    ]
    □ 890 RPWSS.C      1544 check_conj
    □ 82 MAINRATE.C   1545 |->etoi { 116 }

    ]
    □ 963 RPWSS.C      1546 check_for_dup
    □ 82 MAINRATE.C   1547 |->etoi { 116 }
    □ 890 RPWSS.C      1548 |->check_conj { 1544 }
    □ 83 RPWLIB.C      1549 |->handle_error { 30 }
    □
    □ 924 RPWSS.C      1550 |->check_no_conj

    ]
    □ 140 RPWLIB.C     1551 clear_category
    □ 120 RPWVIDEO.C   1552 |->prtstr { 3 }

    ]
    □ 122 RPWLIB.C     1553 clear_revenue
    □ 316 RPWVIDEO.C   1554 |->clr_box { 32 }

    ]
    □ 1608 RPWENTR2.C   1555 verify_reg_fee
    □

    □ 107 RPWWIN.C     1556 window_field
    □ 78 RPWWIN.C      1557 |->window_write { 785 }
    □ 212 RPWVIDEO.C   1558 |->setcur { 17 }
    □ 224 RPWVIDEO.C   1559 |->disp_cur { 159 }
    □ 201 RPWVIDEO.C   1560 |->cursor_off { 45 }
    □
    □ 1561 ..strlen get_key putchar

    ]
    □ 99 LOOKUP.C       1562 zip_lookup
    □ 29 CITYZIP.C      1563 |->ptr_ziphelpfile { 15 }
    □ 30 STATECOD.C     1564 |->ptr_statefile { 895 }
    □ 13 RPWWIN.C       1565 |->window_init { 778 }
    □ 44 RPWWIN.C       1566 |->window_open { 780 }
    □ 78 RPWWIN.C       1567 |->window_write { 785 }
    □ 107 RPWWIN.C      1568 |->window_field { 1556 }
    □ 157 CITYZIP.C     1569 |->LoadOneState { 1540 }
    □ 65 RPWWIN.C       1570 |->window_close { 799 }
    □
    □ 1571 ..sprintf get_key strcpy

```

#### USUMMARY Graphic TREEs Function-versus-line index

\*\*\*\*\*

```

    □ 1 :::
    □ 83 :::
    □ 253 AreYouSure
    □ 680 BPM_Bulk_Rate
    □ 679 BPM_CRT_Rate
    □ 668 BPM_SP_Rate
    □ 669 Barcode_Display
    □ 670 Barcode_Header
    □ 826 CheckForDisk
    □ 903 CheckState
    □ 829 CopyUpFiles
    □ 1117 Corrected_Sample
    □ 610 Disp_Error
    □ 615 Disp_MailPiece
    □ 89 Disp_Text
    □ 140 DisplayMultipleSelectMessage
    □ 1538 Find_SS
    □ 119 Init_Table_Info

```

↳ 90 Init\_Text\_Display  
↳ 1540 LoadOneState  
↳ 239 ReadSeed  
↳ 91 Search\_Text\_Array  
↳ 246 WriteSeed  
↳ 620 \_setargv  
↳ 621 \_setenvp  
□ 1357 abort\_backup\_session  
□ 1350 abort\_question  
□ 1355 abort\_session  
□ 104 add\_mailpiece  
□ 921 allocate\_mem  
□ 390 alter\_of\_charges  
□ 565 are\_there\_more\_ss  
□ 112 are\_there\_ss  
□ 505 ask\_scale  
□ 1173 backup\_filename  
□ 49 box  
□ 1054 boxdiv  
□ 50 boxtop  
□ 1055 boxtop2  
□ 1380 buffcmp  
□ 901 bytecopy  
□ 558 canada\_question  
□ 622 catcomp  
□ 432 cert\_type  
□ 1109 char\_field  
□ 1544 check\_conj  
□ 1546 check\_for\_dup  
□ 1550 check\_no\_conj  
□ 85 clear\_DE\_box  
□ 114 clear\_SS\_box  
□ 1551 clear\_category  
□ 624 clear\_mailpiece  
□ 31 clear\_message  
□ 1553 clear\_revenue  
□ 19 clear\_scr  
□ 12 close\_state\_file  
□ 11 close\_zip\_lookup  
□ 14 close\_ziphelpfile  
□ 32 clr\_box  
□ 35 clr\_entry\_scrn  
□ 913 clr\_scrn  
□ 824 clrsqr  
□ 839 compare\_em  
□ 294 compress\_ss\_fields\_array  
□ 259 container\_skip\_override  
□ 168 correct\_paid  
□ 132 create\_box  
□ 1187 current\_session\_question  
□ 45 cursor\_off  
□ 626 custom\_rates  
□ 1419 cvt  
□ 1021 cvto  
□ 1229 data\_entry  
□ 996 databak  
□ 1133 date\_check

□ 1125 date\_field  
□ 27 date\_time  
□ 528 dbmc\_question  
□ 1150 delete\_backup1  
□ 975 delete\_rec  
□ 199 disp\_agency  
□ 1006 disp\_blowup  
□ 1024 disp\_cathdr  
□ 159 disp\_cur  
□ 1016 disp\_sum  
□ 1093 display\_data  
□ 1155 display\_incoming\_msg  
□ 629 display\_menu\_bar  
□ 627 display\_mmcodes  
□ 41 display\_record\_count  
□ 633 display\_rules  
□ 933 display\_samp  
□ 1143 display\_sample  
□ 145 display\_ss\_rate  
□ 149 display\_ss\_text  
□ 237 display\_start  
□ 1462 disply\_class3  
□ 1489 disply\_ct  
□ 1414 disply\_scrn  
□ 1433 disply\_scrnzn  
□ 1457 disply\_sp  
□ 1035 disprecsdata  
□ 1046 disprecsesn  
□ 924 do\_exit  
□ 25 edit\_rate  
□ 637 edit\_rate?385?  
□ 1306 edit\_wt\_fields  
□ 1364 end\_session  
□ 234 erase\_error  
□ 959 erase\_key  
□ 116 etoi  
□ 639 etoi?493?  
□ 21 exit  
□ 8 exit\_routine  
□ 211 final\_ok  
□ 453 final\_ok\_certs  
□ 797 findany  
□ 641 findcity  
□ 1332 first\_pre\_rates  
□ 1340 first\_rates  
□ 644 fourth\_rates  
□ 9 free\_code\_limit  
□ 1207 generic\_scroll\_bar  
□ 192 get\_agency  
□ 202 get\_agency\_choice  
□ 687 get\_auto  
□ 1172 get\_backup\_session\_number  
□ 1317 get\_cat\_code  
□ 136 get\_choice  
□ 386 get\_cod\_sscatcode  
□ 1212 get\_container\_skip  
□ 1199 get\_counted\_skip

□ 857 get\_curent\_days  
□ 1064 get\_data  
□ 936 get\_date  
□ 535 get\_dbmc\_zip  
□ 292 get\_enclos  
□ 517 get\_indicia  
□ 346 get\_ins\_sscatcode  
□ 328 get\_int\_pp\_type  
□ 358 get\_intl\_fee  
□ 322 get\_local  
□ 315 get\_machine  
□ 1371 get\_num\_recs  
□ 964 get\_onesesn  
□ 1236 get\_orig\_test\_type  
□ 1291 get\_pieces  
□ 480 get\_post\_due  
□ 238 get\_random  
□ 890 get\_rate  
□ 300 get\_revenue\_only  
□ 1135 get\_sample\_record  
□ 997 get\_samplerec  
□ 1053 get\_seq  
□ 955 get\_session  
□ 1171 get\_session\_number  
□ 230 get\_skip\_fld  
□ 1039 get\_spcat  
□ 127 get\_ss  
□ 1204 get\_start\_info\_element\_num  
□ 1312 get\_std  
□ 500 get\_sub\_weight  
□ 496 get\_subsample\_wt  
□ 833 get\_time\_date\_a  
□ 836 get\_time\_date\_c  
□ 173 get\_variance  
□ 849 get\_version\_days  
□ 1300 get\_weight  
□ 462 get\_x\_category  
□ 212 get\_zip  
□ 830 get\_zone\_files  
□ 1447 getbpmbulk  
□ 1443 getbpmfsp  
□ 850 getdate  
□ 851 getjdate  
□ 977 getkey  
□ 694 getkey?218?  
□ 1495 getkeymult  
□ 80 getmenukey  
□ 1503 getrate4sp  
□ 1422 getrates  
□ 1485 getrates3  
□ 1465 getrates3bulklet  
□ 1470 getrates3bulkletnp  
□ 1475 getrates3bulknon  
□ 1480 getrates3bulknonnp  
□ 1439 getratezn  
□ 1451 getrateznbm  
□ 51 gotoxy

□ 656 grab\_scr\_char  
□ 917 handle\_an\_error  
□ 30 handle\_error  
□ 6 handle\_error2  
□ 876 handle\_error2A  
□ 48 handle\_error3  
□ 945 handle\_error4  
□ 698 handle\_error?541?  
□ 821 hash  
□ 703 hash?223?  
□ 1105 header\_input  
□ 655 hilite  
□ 916 how\_big\_index  
□ 571 ib\_reply\_submenu  
□ 871 init\_data\_records  
□ 899 init\_state\_file  
□ 844 init\_video  
□ 893 init\_zip\_lookup  
□ 896 init\_ziphelpfile  
□ 439 input\_cert\_fee  
□ 377 input\_cod\_fee  
□ 335 input\_ins\_fee  
□ 415 input\_reg\_fee  
□ 401 intl\_reg\_mailpiece  
□ 852 isLeapYear  
□ 1410 isValidDate  
□ 704 is\_presort  
□ 707 isalert  
□ 709 isbreak  
□ 711 iscd  
□ 713 ischgcd  
□ 715 ischgcts  
□ 717 ischgdsr  
□ 719 ischgri  
□ 721 iscts  
□ 723 isctsblocked  
□ 725 isdsr  
□ 727 isframerr  
□ 729 isigalert  
□ 731 isigcd  
□ 733 isigcts  
□ 735 isigdsr  
□ 737 isiglstat  
□ 739 isigmstat  
□ 741 islinerr  
□ 743 ismodemerr  
□ 745 isncd  
□ 747 isncts  
□ 749 isndsr  
□ 751 isoverrun  
□ 753 isparityerr  
□ 755 isri  
□ 757 isring  
□ 759 isrxempty  
□ 761 isrxfull  
□ 763 isrxintrunning  
□ 765 isrxovflow

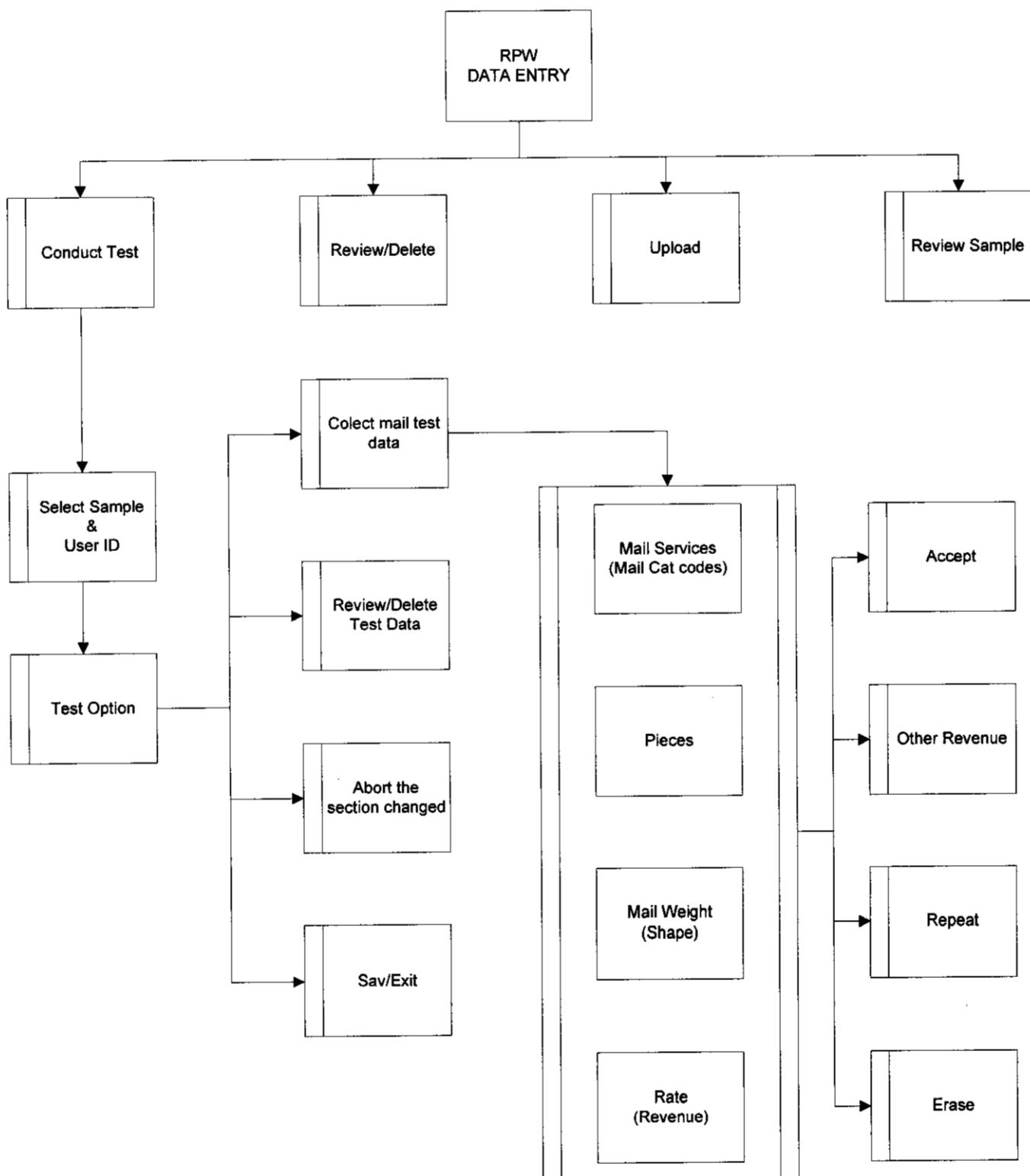
- 767 istxempty
- 769 istxfull
- 771 istxinrunning
- 773 isxmrxing
- 775 isxoffblocked
- 777 item\_lookup
- 989 key\_info
- 803 keycomp
- 601 letter\_card
- 285 letter\_non
- 805 level\_a\_b
- 1374 load\_array
- 68 load\_catfile
- 866 load\_code\_limit
- 863 load\_message
- 875 load\_rate\_table
- 813 load\_rate\_table?470?
- 873 load\_screen\_buffer
- 817 load\_screen\_buffer?472?
- 1182 load\_zone\_structure
- 820 main
- 823 main?101?
- 843 main?442?
- 1368 main?536?
- 1389 main?82?
- 1394 main?97?
- 1068 makcatkeys
- 1416 makscrnbar
- 705 match
- 73 menubak
- 972 movbar
- 61 mscgetch
- 1396 mscgetch?227?
- 1149 next\_backup
- 185 not\_correct\_paid
- 846 not\_valid\_version
- 247 nr\_seed\_gather
- 241 nr\_seed\_init
- 240 nr\_seed\_scatter
- 156 num\_field
- 97 num\_input
- 1162 originating\_Question
- 153 other\_revenue
- 1384 output\_index
- 646 parcel\_rates
- 542 parcel\_size
- 1147 parse\_descr
- 226 piece\_skip\_override
- 422 pop\_list1
- 370 pop\_list10
- 1242 pop\_list11
- 1251 pop\_list12
- 1261 pop\_list2
- 1268 pop\_list3
- 1277 pop\_list4
- 1284 pop\_list5
- 215 pop\_list6

277 pop\_list7  
307 pop\_list9  
66 pop\_scrn  
909 pop\_scrn2  
1214 pop\_wnd  
520 post\_weight\_edits  
1305 prelim\_wt\_edits  
65 prep\_exit  
1398 prep\_exit?228?  
583 presort\_1st\_menu  
591 presort\_gov\_menu  
108 print\_cat  
1098 print\_header  
649 print\_rate  
521 priority\_question  
1401 priority\_rates  
1492 prntc  
971 process\_key  
72 process\_menu  
1455 process\_sp  
1431 process\_zn  
1403 prt\_substr  
2 prtbuff  
1001 prtfld  
1027 prtlin  
1404 prtlin?230?  
3 prtstr  
895 ptr\_statefile  
15 ptr\_ziphelpfile  
1405 putdate  
1406 putjdate  
245 ran2  
1326 rate\_1910  
660 rate\_screen\_reset  
44 raterev  
926 read\_in\_index  
928 read\_in\_recs  
867 rec\_count  
408 reg\_ins\_question  
1076 remote\_exit\_routine  
271 remove\_mailpiece  
1152 rename\_backup  
39 repeat  
217 reset\_buckets  
144 restor\_wnd  
1322 revenue  
943 review\_data  
1081 revsamp  
1031 revscrn  
227 sav\_wnd  
47 save\_scrn  
131 save\_wnd  
1223 scale\_question  
1413 screen1  
1426 screen1a  
1430 screen2  
1454 screen3

| 1497 screen4  
C 1499 screen5  
C 1501 screen6  
U 1507 screen7  
C 1509 screen7a  
D 1511 screen8  
T 1513 screen9  
C 970 scrn\_hndlr  
C 654 scroll\_bar  
C 1515 scroll\_scr  
D 912 selsamp  
C 1011 sessionbak  
C 969 set\_keys  
C 1517 set\_unitname  
C 17 setcur  
C 60 shade\_it  
D 1523 show\_sub\_weights  
D 958 size\_data  
D 981 size\_idx  
D 1067 sort\_data  
C 1378 sort\_it  
D 1518 spcomp  
C 1215 split\_scroll\_bar  
C 580 ss\_menu  
C 900 state\_read  
D 1520 sub\_wt\_ok  
D 1192 subsample\_question  
D 351 surf\_or\_air  
C 1040 table\_bsearch  
C 1533 table\_comp  
D 121 table\_get  
D 882 table\_load  
D 885 table\_read  
D 118 table\_search  
D 1535 third\_rates  
D 179 trim  
D 1536 trim?381?  
D 889 update\_ss\_desc  
C 84 usps\_govt  
D 128 validate  
D 487 validate\_fee  
D 365 validate\_limit  
C 445 validate\_multiple  
D 383 verify\_cod\_fee  
D 342 verify\_ins\_fee  
D 1555 verify\_reg\_fee  
C 224 where\_to\_go  
D 790 win\_lookup\_draw  
D 794 win\_new\_row  
D 799 window\_close  
D 1556 window\_field  
D 792 window\_hilite  
D 778 window\_init  
D 780 window\_open  
D 785 window\_write  
D 5 write\_record  
C 53 writeca

471 x\_cat\_submenu  
 164 yes\_no  
 1562 zip\_lookup

# **RPW DATA ENTRY PROGRAM FLOWCHART**



# **RPW-CODES**

## **Computer System Documentation Description**

### **Section 2**

## **Base Unit System**

## **INTRODUCTION**

The CODES-RPW is a computer information system by which mail volume data, service analysis data and other mail characteristic data are collected, developed and presented in a variety of report formats for use by postal management. CODES-RPW is maintained on a network of IBM computers feeding a national host mainframe via SNA synchronous modems and a telecommunication package. Information from each sampled piece of mail is collected on a lap-held portable computer. On a daily basis, the collected data from the portable units are uploaded to a Base Unit computer at Management Sectional Centers (MSC) in two different ways:

1. The portable floppy diskette with sample data is inserted into the Base Unit to copy RPW data to the Base Unit.
2. The internal modem on the portable unit is used to transmit data over telephone lines to the Base Unit.

On a weekly basis, the Base Unit data are subsequently validated and uploaded to the USPS mainframe computer for RPW production.

The objectives of the CODES-RPW Base Unit program are:

**RPW Data Transfer** - to upload RPW data to the USPS mainframe computer; receive RPW (remote upload) data on the Base Unit from an associate office computer via telephone; or disk transfer data which are on a 3.5" diskette, generated by either a portable data entry computer or the Base Unit, to a second diskette on the Base Unit.

**RPW Data Management** - to review an RPW test data file, key a zero volume test, cancel a test, reschedule a test, or edit the header information in a test data file.

**RPW System Management** - to check RPW test status, to review a shortened version of the sample selection records, to review detailed sample selection records, review all changes made to the RPW test, and print RPW SSA reports.

**Prepare Program/Data Disk** - to prepare a diskette and transfer the RPW data entry program, remote batch file, remote script file, and the Permit communication file to it.

## Methods and Procedures Employed.

A software security package, PROTEC Security, on the Base Unit computer limits access to the data and software stored in the Base Unit. CODES-RPW Base Unit software cannot be accessed unless the correct procedures and password are used to access the Applications Menu. Various applications and utilities can be accessed on the Base Unit including: RPW, RPW, IOCS, Communications, Inventory, and several utilities. The CODES-RPW program is menu driven and the operator can browse through the various options, select the desired action, return to previous menus and quit the program by selecting numbered menu items.

The Base Unit CODES-RPW program is written in the 'Clipper' Programming language. 'Clipper' is a stand-alone application development system. After the program code is created, compiled, and linked, the executable (.EXE) file can be run on any PC/MS-DOS compatible computer. The Clipper compiler converts source code programs into object code files. The source code files are identified by the DOS file name extension 'PRG'. The object code files are identified by the DOS file name extension 'OBJ', and are linked with a linker, such as PLINK86 Plus, to produce an executable file, identified by the DOS file extension 'EXE'. The program can be run with only the executable file (and the associated input files) but any program changes require the source code, the Clipper program, a linker, and the FUNCky (version 1.5) third-party add-in function library. The CODES-RPW base unit computers are configured with two removable Bernoulli cartridge drives (C:\ and D:\).

The Base Unit section contains file narratives, file layouts, and the flowchart that illustrates RPW process flow.

Source code for PQ3 2000 complements this technical documentation.

## **RPW BASE UNIT NARRATIVES**

RPW Base Unit application is written in "Clipper" and consists of 9 database files which are as follows:

AUDITNEW.DBF  
CATCODE.DBF  
NEWRSMPL.DBF  
RPW.DBF  
RPWADMIN.DBF  
RPWAUDIT.DBF  
RPWNEW.DBF  
RSAMPLE.DBF  
SSCATCOD.DBF

The external sources of data used and produced within the CODES RPW base unit are as follows:

|               |  |
|---------------|--|
| AUTDITNEW.DBF | this is un updated file format for audit.dbf. When fields need to be added to or removed from audit, the desired format is delivered in auditnew (with no data). When end-of-quarter process runs, the format of audit is updated. |
| CATCODE.DBF   | this is reference table containing definitions for Category codes. These definitions are used on reports only  |
| NEWRSMPL.DBF  | downloaded from San Mateo mainframe computer.  |
| RPW.DBF       | RPW test data collected on remote units.   |
| RPWADMIN.DBF  | the sample selection download procedure creates this file.   |
| RPWAUDIT.DBF  | generated by the CODES-RPW program.  |
| RPWNEW.DBF    | this is an updated file format for rpw.dbf. When fields need to be added to or removed from rpw, the desired format is delivered in rpwnew (with no data). when the end-of-quarter process runs, the format of rpw is updated.     |
| RSAMPLE.DBF   | the renamed NEWOSMPL.DBF, downloaded from the San Mateo mainframe computer.  |

SSCATCOD.DBF      table of special service category codes, locations in the rate table corresponding to these special services, and test containing the name of the special service or combination of services.

Explanations of Data Modifications.

All modifications to the input RPW data file header information are documented and logged in the AUDIT.DBF file, which is an audit trail record of all aborted tests, changes, and edits performed on RPW test data. The RPW Test ID and Test Date header data may be edited, but the actual RPW test data cannot be changed.

## File Layouts

Section: Data Files

Category: XBASE Data Files

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\AUDITNEW.DBF

Field List:

| #  | Name       | Type      | Len | Dec | Offset |
|----|------------|-----------|-----|-----|--------|
| 1  | FINNUM     | Character | 6   | 0   | 0      |
| 2  | TESTID     | Character | 6   | 0   | 6      |
| 3  | TESTDATE   | Character | 8   | 0   | 12     |
| 4  | DEVERS     | Character | 2   | 0   | 20     |
| 5  | TIME       | Numeric   | 4   | 0   | 22     |
| 6  | FACTOR     | Character | 4   | 0   | 26     |
| 7  | CATCODE    | Character | 4   | 0   | 30     |
| 8  | PIECES     | Character | 4   | 0   | 34     |
| 9  | REVENUE    | Character | 9   | 0   | 38     |
| 10 | WEIGHT     | Character | 7   | 0   | 47     |
| 11 | AGPCODE    | Character | 3   | 0   | 54     |
| 12 | SCATCODE   | Character | 4   | 0   | 57     |
| 13 | OR_DES_ZIP | Character | 3   | 0   | 61     |
| 14 | ZONE       | Character | 1   | 0   | 64     |
| 15 | SHORT      | Character | 5   | 0   | 65     |
| 16 | OVER       | Character | 5   | 0   | 70     |
| 17 | INDICIA    | Character | 1   | 0   | 75     |
| 18 | AC         | Character | 1   | 0   | 76     |
| 19 | INDICATOR  | Character | 1   | 0   | 77     |
| 20 | LENGTH     | Character | 3   | 0   | 78     |
| 21 | WIDTH      | Character | 2   | 0   | 81     |
| 22 | HEIGHT     | Character | 2   | 0   | 83     |
| 23 | GIRTH      | Character | 2   | 0   | 85     |
| 24 | SSMETHOD   | Character | 1   | 0   | 87     |
| 25 | SSP        | Character | 2   | 0   | 88     |
| 26 | SESSION    | Numeric   | 2   | 0   | 90     |
| 27 | RECNUM     | Numeric   | 4   | 0   | 92     |
| 28 | USERID     | Character | 3   | 0   | 96     |
| 29 | FLAG       | Character | 1   | 0   | 99     |
| 30 | POS_TERM   | Character | 1   | 0   | 100    |

Record Information:

994 Bytes for the header  
102 Bytes per record  
30 Fields in all

Source:

AUDITNEW.DBF

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\CATCODE.DBF

Field List:

| # | Name | Type | Len | Dec | Offset |
|---|------|------|-----|-----|--------|
|---|------|------|-----|-----|--------|

|   |         |           |    |   |   |
|---|---------|-----------|----|---|---|
| 1 | CATCODE | Character | 4  | 0 | 0 |
| 2 | DESCRIP | Character | 70 | 0 | 4 |

Record Information:

02/10/00 Last updated  
 124 Records now in file  
 98 Bytes for the header  
 75 Bytes per record  
 9399 Bytes in file  
 2 Fields in all

Source:

CATCODE.DBF

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\NEWRSMPL.DBF

Field List:

| #  | Name     | Type      | Len | Dec | Offset |
|----|----------|-----------|-----|-----|--------|
| 1  | SITECODE | Character | 3   | 0   | 0      |
| 2  | FINNUM   | Character | 6   | 0   | 3      |
| 3  | MOZIP    | Character | 5   | 0   | 9      |
| 4  | POZIP    | Character | 5   | 0   | 14     |
| 5  | MEPTYPE  | Character | 2   | 0   | 19     |
| 6  | HQCODE   | Character | 4   | 0   | 21     |
| 7  | FILL1    | Character | 3   | 0   | 25     |
| 8  | BACODE   | Character | 2   | 0   | 28     |
| 9  | ODISAREA | Character | 3   | 0   | 30     |
| 10 | TESTID   | Character | 6   | 0   | 33     |
| 11 | TESTDATE | Character | 8   | 0   | 39     |
| 12 | STRATA   | Character | 3   | 0   | 47     |
| 13 | PONAME   | Character | 30  | 0   | 50     |
| 14 | CAG      | Character | 1   | 0   | 80     |
| 15 | MEPDESC  | Character | 80  | 0   | 81     |
| 16 | MEPZIP   | Character | 5   | 0   | 161    |
| 17 | SAMPCELL | Character | 3   | 0   | 166    |
| 18 | FILL2    | Character | 3   | 0   | 169    |
| 19 | MONAME   | Character | 30  | 0   | 172    |
| 20 | CSD      | Character | 3   | 0   | 202    |
| 21 | FILL3    | Character | 45  | 0   | 205    |

Record Information:

706 Bytes for the header  
 251 Bytes per record  
 21 Fields in all

Source:

NEWRSMPL.DBF

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\RPW.DBF

Field List:

| # | Name   | Type      | Len | Dec | Offset |
|---|--------|-----------|-----|-----|--------|
| 1 | FINNUM | Character | 6   | 0   | 0      |

|    |            |           |   |   |    |
|----|------------|-----------|---|---|----|
| 2  | TESTID     | Character | 6 | 0 | 6  |
| 3  | TESTDATE   | Character | 8 | 0 | 12 |
| 4  | DEVERS     | Character | 2 | 0 | 20 |
| 5  | TIME       | Numeric   | 4 | 0 | 22 |
| 6  | FACTOR     | Character | 4 | 0 | 26 |
| 7  | CATCODE    | Character | 4 | 0 | 30 |
| 8  | PIECES     | Character | 4 | 0 | 34 |
| 9  | REVENUE    | Character | 9 | 0 | 38 |
| 10 | WEIGHT     | Character | 7 | 0 | 47 |
| 11 | AGCODE     | Character | 3 | 0 | 54 |
| 12 | SCATCODE   | Character | 4 | 0 | 57 |
| 13 | OR_DES_ZIP | Character | 3 | 0 | 61 |
| 14 | ZONE       | Character | 1 | 0 | 64 |
| 15 | SHORT      | Character | 5 | 0 | 65 |
| 16 | OVER       | Character | 5 | 0 | 70 |
| 17 | INDICIA    | Character | 1 | 0 | 75 |
| 18 | AC         | Character | 1 | 0 | 76 |
| 19 | INDICATOR  | Character | 1 | 0 | 77 |
| 20 | LENGTH     | Character | 3 | 0 | 78 |
| 21 | WIDTH      | Character | 2 | 0 | 81 |
| 22 | HEIGHT     | Character | 2 | 0 | 83 |
| 23 | GIRTH      | Character | 2 | 0 | 85 |
| 24 | SSMETHOD   | Character | 1 | 0 | 87 |
| 25 | SSP        | Character | 2 | 0 | 88 |
| 26 | SESSION    | Numeric   | 2 | 0 | 90 |
| 27 | RECNUM     | Numeric   | 4 | 0 | 92 |
| 28 | USERID     | Character | 3 | 0 | 96 |
| 29 | FLAG       | Character | 1 | 0 | 99 |

#### Record Information:

962 Bytes for the header  
 101 Bytes per record  
 29 Fields in all

Source:  
 RPW.DBF

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\RPWADMIN.DBF

#### Field List:

| #  | Name     | Type      | Len | Dec | Offset |
|----|----------|-----------|-----|-----|--------|
| 1  | FINNUM   | Character | 6   | 0   | 0      |
| 2  | TESTID   | Character | 6   | 0   | 6      |
| 3  | TESTDATE | Character | 8   | 0   | 12     |
| 4  | TIME     | Numeric   | 4   | 0   | 20     |
| 5  | USERID   | Character | 3   | 0   | 24     |
| 6  | SESSION  | Numeric   | 2   | 0   | 27     |
| 7  | RECNUM   | Numeric   | 4   | 0   | 29     |
| 8  | TOTREC   | Numeric   | 5   | 0   | 33     |
| 9  | CHKIDATE | Date      | 8   | 0   | 38     |
| 10 | XMITDATE | Date      | 8   | 0   | 46     |
| 11 | RECHDATE | Date      | 8   | 0   | 54     |
| 12 | CATCODE  | Character | 4   | 0   | 62     |

#### Record Information:

418 Bytes for the header  
67 Bytes per record  
12 Fields in all

Source:  
RPWADMIN.DBF

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\RPWAUDIT.DBF

Field List:

| #  | Name       | Type      | Len | Dec | Offset |
|----|------------|-----------|-----|-----|--------|
| 1  | FINNUM     | Character | 6   | 0   | 0      |
| 2  | TESTID     | Character | 6   | 0   | 6      |
| 3  | TESTDATE   | Character | 8   | 0   | 12     |
| 4  | DEVERS     | Character | 2   | 0   | 20     |
| 5  | TIME       | Numeric   | 4   | 0   | 22     |
| 6  | FACTOR     | Character | 4   | 0   | 26     |
| 7  | CATCODE    | Character | 4   | 0   | 30     |
| 8  | PIECES     | Character | 4   | 0   | 34     |
| 9  | REVENUE    | Character | 9   | 0   | 38     |
| 10 | WEIGHT     | Character | 7   | 0   | 47     |
| 11 | AGC CODE   | Character | 3   | 0   | 54     |
| 12 | SCATCODE   | Character | 4   | 0   | 57     |
| 13 | OR DES ZIP | Character | 3   | 0   | 61     |
| 14 | ZONE       | Character | 1   | 0   | 64     |
| 15 | SHORT      | Character | 5   | 0   | 65     |
| 16 | OVER       | Character | 5   | 0   | 70     |
| 17 | INDICIA    | Character | 1   | 0   | 75     |
| 18 | AC         | Character | 1   | 0   | 76     |
| 19 | INDICATOR  | Character | 1   | 0   | 77     |
| 20 | LENGTH     | Character | 3   | 0   | 78     |
| 21 | WIDTH      | Character | 2   | 0   | 81     |
| 22 | HEIGHT     | Character | 2   | 0   | 83     |
| 23 | GIRTH      | Character | 2   | 0   | 85     |
| 24 | SSMETHOD   | Character | 1   | 0   | 87     |
| 25 | SSP        | Character | 2   | 0   | 88     |
| 26 | SESSION    | Numeric   | 2   | 0   | 90     |
| 27 | RECNUM     | Numeric   | 4   | 0   | 92     |
| 28 | USERID     | Character | 3   | 0   | 96     |
| 29 | FLAG       | Character | 1   | 0   | 99     |
| 30 | ORDER      | Numeric   | 2   | 0   | 100    |

Record Information:

994 Bytes for the header  
103 Bytes per record  
30 Fields in all

Source:  
RPWAUDIT.DBF

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\RPWNEW.DBF

Field List:

| # | Name | Type | Len | Dec | Offset |
|---|------|------|-----|-----|--------|
|---|------|------|-----|-----|--------|

|    |            |           |   |   |     |
|----|------------|-----------|---|---|-----|
| 1  | FINNUM     | Character | 6 | 0 | 0   |
| 2  | TESTID     | Character | 6 | 0 | 6   |
| 3  | TESTDATE   | Character | 8 | 0 | 12  |
| 4  | DEVERS     | Character | 2 | 0 | 20  |
| 5  | TIME       | Numeric   | 4 | 0 | 22  |
| 6  | FACTOR     | Character | 4 | 0 | 26  |
| 7  | CATCODE    | Character | 4 | 0 | 30  |
| 8  | PIECES     | Character | 4 | 0 | 34  |
| 9  | REVENUE    | Character | 9 | 0 | 38  |
| 10 | WEIGHT     | Character | 7 | 0 | 47  |
| 11 | AGCODE     | Character | 3 | 0 | 54  |
| 12 | SCATCODE   | Character | 4 | 0 | 57  |
| 13 | OR_DES_ZIP | Character | 3 | 0 | 61  |
| 14 | ZONE       | Character | 1 | 0 | 64  |
| 15 | SHORT      | Character | 5 | 0 | 65  |
| 16 | OVER       | Character | 5 | 0 | 70  |
| 17 | INDICIA    | Character | 1 | 0 | 75  |
| 18 | AC         | Character | 1 | 0 | 76  |
| 19 | INDICATOR  | Character | 1 | 0 | 77  |
| 20 | LENGTH     | Character | 3 | 0 | 78  |
| 21 | WIDTH      | Character | 2 | 0 | 81  |
| 22 | HEIGHT     | Character | 2 | 0 | 83  |
| 23 | GIRTH      | Character | 2 | 0 | 85  |
| 24 | SSMETHOD   | Character | 1 | 0 | 87  |
| 25 | SSP        | Character | 2 | 0 | 88  |
| 26 | SESSION    | Numeric   | 2 | 0 | 90  |
| 27 | RECNUM     | Numeric   | 4 | 0 | 92  |
| 28 | USERID     | Character | 3 | 0 | 96  |
| 29 | FLAG       | Character | 1 | 0 | 99  |
| 30 | POS_TERM   | Character | 1 | 0 | 100 |

#### Record Information:

994 Bytes for the header  
 102 Bytes per record  
 30 Fields in all

#### Source:

RPWNEW.DBF

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\RSAMPLE.DBF

#### Field List:

| #  | Name     | Type      | Len | Dec | Offset |
|----|----------|-----------|-----|-----|--------|
| 1  | SITECODE | Character | 3   | 0   | 0      |
| 2  | FINNUM   | Character | 6   | 0   | 3      |
| 3  | MOZIP    | Character | 5   | 0   | 9      |
| 4  | POZIP    | Character | 5   | 0   | 14     |
| 5  | MEPTYPE  | Character | 2   | 0   | 19     |
| 6  | HQCODE   | Character | 4   | 0   | 21     |
| 7  | FILL1    | Character | 3   | 0   | 25     |
| 8  | BACODE   | Character | 2   | 0   | 28     |
| 9  | ODISAREA | Character | 3   | 0   | 30     |
| 10 | TESTID   | Character | 6   | 0   | 33     |
| 11 | TESTDATE | Character | 8   | 0   | 39     |

|    |          |           |    |   |     |
|----|----------|-----------|----|---|-----|
| 12 | STRATA   | Character | 3  | 0 | 47  |
| 13 | PONAME   | Character | 30 | 0 | 50  |
| 14 | CAG      | Character | 1  | 0 | 80  |
| 15 | MEPDESC  | Character | 80 | 0 | 81  |
| 16 | MEPZIP   | Character | 5  | 0 | 161 |
| 17 | SAMPCELL | Character | 3  | 0 | 166 |
| 18 | FILL2    | Character | 3  | 0 | 169 |
| 19 | MONAME   | Character | 30 | 0 | 172 |
| 20 | CSD      | Character | 3  | 0 | 202 |
| 21 | FILL3    | Character | 45 | 0 | 205 |

Record Information:

706 Bytes for the header  
 251 Bytes per record  
 21 Fields in all

Source:

RSAMPLE.DBF

S:\RATECASE\2000Q3.1\BASELINE.BU\CODES\RPW\SSCATCOD.DBF

Field List:

| # | Name      | Type      | Len | Dec | Offset |
|---|-----------|-----------|-----|-----|--------|
| 1 | SSCATCODE | Character | 4   | 0   | 0      |
| 2 | RATEPTRS  | Character | 12  | 0   | 4      |
| 3 | DESCRIP   | Character | 64  | 0   | 16     |

Record Information:

02/10/00 Last updated  
 138 Records now in file  
 129 Bytes for the header  
 81 Bytes per record  
 11308 Bytes in file  
 3 Fields in all

Source:

SSCATCOD.DBF

Section: Source Code

Section: Reference

Application Tree for RPW Base Unit

```

Main()
    LoadVersion()                               RELEASE.PRG
2x
3   STOD()                                   FUNCLIB.PRG
1x
    AdvanceFiles()                            RELEASE.PRG
1x
3   CopyNewFiles()                           RELEASE.PRG
14x
3   3   ListEval()                           RELEASE.PRG
1x
3   DellList()                             RELEASE.PRG
10x
3   3   ListEval()+
1x
3   ExistFList()                           RELEASE.PRG
2x
3   ListEval()+
1x
    InitVersion()                           FUNCLIB.PRG
1x
ENDPQ()                                     RPWPROC.PRG
1x
3   YNResponse()                           RPWPROC.PRG
2x
3   STOD()+
3x
3   RPWSMPLAppend()                      RPWPROC.PRG
1x
3   AdvanceFiles()                           RELEASE.PRG
1x
    RPW2()                                 RPW2.PRG
2x
3   MASK()                                RPWPROC.PRG
1x
3   STOC()                                FUNCLIB.PRG
3x
3   STOD()+
3x
3   RPWPRNT()                            RPWPRNT.PRG
1x
3   3   Sess2()                            RPWPRNT2.PRG
1x
3   3   heading1()                           RPWPRNT.PRG
3x
3   3   3   hdrinfo1()                      RPWPRNT.PRG
1x
3   3   Cvnt_Indic()                      BASEREV.PRG
1x
3   3   total()                            RPWPRNT.PRG
3x
3   3   3   heading1()+
1x
3   3   summ2()                            RPWPRNT2.PRG
2x

```

```

      3   3       heading2()                               RPWPRNT2.PRG
1x
      3   3       3     hdrinfo2()                         RPWPRNT2.PRG
1x
      3   3       Sess2() +                            RPWPRNT2.PRG
1x
      3     RpwVolRpt()                            RPWVOLRP.PRG
1x
      3   3       OpenRefVolume()                      RPWVOLRP.PRG
1x
      3   3       OpenRpwFiles()                        RPWVOLRP.PRG
1x
      3   3       Check1CSPVolume()                   RPWVOLRP.PRG
1x
      3   3       Sum1CSPieces()                      RPWVOLRP.PRG
1x
      3   3       OutOfProportion()                  RPWVOLRP.PRG
1x
      3   3       RptOutOfProp()                      RPWVOLRP.PRG
1x
      3   3       3     PrintSubSamp()                RPWVOLRP.PRG
1x
      3   3       3     PrintInstructions()          RPWVOLRP.PRG
1x
      3   3       RptTestMatrix()                      RPWVOLRP.PRG
1x
      3     RMTUPLOAD()                            RPWPROC.PRG
1x
      3       YNResponse() +                         RPWPROC.PRG
1x
      MASK() +                                RPWPROC.PRG
1x
      Warning()                                RPWPROC.PRG
1x
      RPW1()                                  RPW1.PRG
1x
      RPW3()                                  RPW3.PRG
1x
      3     MASK() +                            RPWPROC.PRG
1x
      3     RPW31()                            RPW31.PRG
1x
      3   3     MASK() +                          RPWPROC.PRG
1x
      3   3     STOC() +                           FUNCLIB.PRG
1x
      3   3     RPWPRNT() +                      RPWPRNT.PRG
1x
      3   3     RpwVolRpt() +                    RPWVOLRP.PRG
2x
      3   3     RPWPrnt2() +                      RPWPRNT2.PRG
1x
      3   3   3     Sess2() +                            RPWPRNT2.PRG
1x
      3   3   3     heading2() +                      RPWPRNT2.PRG
3x

```

|    |   |   |                |              |
|----|---|---|----------------|--------------|
| 3  | 3 | 3 | Cvnt_Indic() + | BASEREV.PRG  |
| 1x | 3 | 3 | summ2() +      | RPWPRNT2.PRG |
| 2x | 3 | 3 | BaseRev()      | BASEREV.PRG  |
| 1x | 3 | 3 | DispCat()      | BASEREV.PRG  |
| 2x | 3 | 3 | Error()        | PROCLIB.PRG  |
| 2x | 3 | 3 | Beep()         | PROCLIB.PRG  |
| 1x | 3 | 3 | Ctr_pos()      | PROCLIB.PRG  |
| 1x | 3 | 3 | CatgScrn()     | BASEREV.PRG  |
| 1x | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG  |
| 3x | 3 | 3 | RecNoScrn()    | BASEREV.PRG  |
| 1x | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG  |
| 3x | 3 | 3 | SayRec()       | BASEREV.PRG  |
| 1x | 3 | 3 | StartPos()     | BASEREV.PRG  |
| 2x | 3 | 3 | STOC() +       | FUNCLIB.PRG  |
| 1x | 3 | 3 | Get_CatInfo()  | BASEREV.PRG  |
| 1x | 3 | 3 | ListRec()      | BASEREV.PRG  |
| 2x | 3 | 3 | Pcs_Count()    | BASEREV.PRG  |
| 1x | 3 | 3 | Rev_Piece()    | BASEREV.PRG  |
| 1x | 3 | 3 | Cvnt_LbOz()    | BASEREV.PRG  |
| 2x | 3 | 3 | Var_Pc()       | BASEREV.PRG  |
| 1x | 3 | 3 | Cvnt_Indic() + | BASEREV.PRG  |
| 1x | 3 | 3 | UpArrow()      | BASEREV.PRG  |
| 1x | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG  |
| 4x | 3 | 3 | Beep() +       | PROCLIB.PRG  |
| 1x | 3 | 3 | SayRec() +     | BASEREV.PRG  |
| 2x | 3 | 3 | StartPos() +   | BASEREV.PRG  |
| 2x | 3 | 3 | ListRec() +    | BASEREV.PRG  |
| 4x | 3 | 3 | Error() +      | PROCLIB.PRG  |
| 1x |   |   |                |              |

|    |   |                  |             |
|----|---|------------------|-------------|
| 3  | 3 | DnArrow()        | BASEREV.PRG |
| 1x | 3 | 3 Ctr_pos() +    | PROCLIB.PRG |
| 4x | 3 | 3 Beep() +       | PROCLIB.PRG |
| 1x | 3 | 3 SayRec() +     | BASEREV.PRG |
| 2x | 3 | 3 Get_NewRow()   | BASEREV.PRG |
| 2x | 3 | 3 3 StartPos() + | BASEREV.PRG |
| 3x | 3 | 3 StartPos() +   | BASEREV.PRG |
| 2x | 3 | 3 ListRec() +    | BASEREV.PRG |
| 4x | 3 | 3 Error() +      | PROCLIB.PRG |
| 1x | 3 | PgUp()           | BASEREV.PRG |
| 1x | 3 | 3 Ctr_pos() +    | PROCLIB.PRG |
| 1x | 3 | 3 Error() +      | PROCLIB.PRG |
| 2x | 3 | 3 SayRec() +     | BASEREV.PRG |
| 4x | 3 | PgDn()           | BASEREV.PRG |
| 1x | 3 | 3 Ctr_pos() +    | PROCLIB.PRG |
| 1x | 3 | 3 SayRec() +     | BASEREV.PRG |
| 4x | 3 | 3 Error() +      | PROCLIB.PRG |
| 2x | 3 | GetRec()         | BASEREV.PRG |
| 1x | 3 | 3 Pcs_Count() +  | BASEREV.PRG |
| 1x | 3 | 3 Rev_Piece() +  | BASEREV.PRG |
| 1x | 3 | 3 Cvnt_LbOz() +  | BASEREV.PRG |
| 2x | 3 | 3 Var_Pc() +     | BASEREV.PRG |
| 1x | 3 | 3 Cvnt_Indic() + | BASEREV.PRG |
| 1x | 3 | 3 Error() +      | PROCLIB.PRG |
| 2x | 3 | 3 Ctr_pos() +    | PROCLIB.PRG |
| 4x | 3 | 3 Confirm()      | PROCLIB.PRG |
| 1x | 3 | 3 3 Beep() +     | PROCLIB.PRG |
| 1x | 3 | 3 Ctr_pos() +    | PROCLIB.PRG |
| 1x |   |                  |             |

|    |   |   |              |             |
|----|---|---|--------------|-------------|
| 3  | 3 | 3 | SaveData()   | BASEREV.PRG |
| 1x | 3 | 3 | Ctr_pos() +  | PROCLIB.PRG |
| 1x | 3 | 3 | Loadrec()    | FUNCLIB.PRG |
| 2x | 3 | 3 | Putrec()     | FUNCLIB.PRG |
| 2x | 3 | 3 | SayRec() +   | BASEREV.PRG |
| 1x | 3 | 3 | StartPos() + | BASEREV.PRG |
| 1x | 3 | 3 | ListRec() +  | BASEREV.PRG |
| 1x | 3 | 3 | Edit_Pcs()   | BASEREV.PRG |
| 1x | 3 | 3 | Error() +    | PROCLIB.PRG |
| 3x | 3 | 3 | Ctr_pos() +  | PROCLIB.PRG |
| 2x | 3 | 3 | Confirm() +  | PROCLIB.PRG |
| 1x | 3 | 3 | Edit_Rev()   | BASEREV.PRG |
| 1x | 3 | 3 | Error() +    | PROCLIB.PRG |
| 5x | 3 | 3 | Ctr_pos() +  | PROCLIB.PRG |
| 2x | 3 | 3 | Calc_Pic()   | PROCLIB.PRG |
| 1x | 3 | 3 | Confirm() +  | PROCLIB.PRG |
| 7x | 3 | 3 | Edit_Lb()    | BASEREV.PRG |
| 1x | 3 | 3 | Error() +    | PROCLIB.PRG |
| 2x | 3 | 3 | Ctr_pos() +  | PROCLIB.PRG |
| 2x | 3 | 3 | Confirm() +  | PROCLIB.PRG |
| 1x | 3 | 3 | Edit_Oz()    | BASEREV.PRG |
| 1x | 3 | 3 | Error() +    | PROCLIB.PRG |
| 3x | 3 | 3 | Ctr_pos() +  | PROCLIB.PRG |
| 2x | 3 | 3 | Calc_Pic() + | PROCLIB.PRG |
| 1x | 3 | 3 | Confirm() +  | PROCLIB.PRG |
| 2x | 3 | 3 | Edit_Var()   | BASEREV.PRG |
| 1x | 3 | 3 | Error() +    | PROCLIB.PRG |
| 5x | 3 | 3 | Ctr_pos() +  | PROCLIB.PRG |
| 3x |   |   |              |             |

|    |   |   |   |                |             |
|----|---|---|---|----------------|-------------|
| 3  | 3 | 3 | 3 | Calc_Pic() +   | PROCLIB.PRG |
| 1x | 3 | 3 | 3 | Confirm() +    | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Beep() +       | PROCLIB.PRG |
| 1x | 3 | 3 | 3 | Edit_Agcode()  | BASEREV.PRG |
| 1x | 3 | 3 | 3 | Error() +      | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Cvnt_Indic() + | BASEREV.PRG |
| 1x | 3 | 3 | 3 | Edit_Indic()   | BASEREV.PRG |
| 1x | 3 | 3 | 3 | Error() +      | PROCLIB.PRG |
| 1x | 3 | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Cvnt_Indic() + | BASEREV.PRG |
| 2x | 3 | 3 | 3 | Edit_AC()      | BASEREV.PRG |
| 1x | 3 | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Edit_Length()  | BASEREV.PRG |
| 1x | 3 | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Confirm() +    | PROCLIB.PRG |
| 1x | 3 | 3 | 3 | Edit_Fld()     | BASEREV.PRG |
| 1x | 3 | 3 | 3 | Error() +      | PROCLIB.PRG |
| 1x | 3 | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Confirm() +    | PROCLIB.PRG |
| 1x | 3 | 3 | 3 | Edit_Girth()   | BASEREV.PRG |
| 1x | 3 | 3 | 3 | Error() +      | PROCLIB.PRG |
| 1x | 3 | 3 | 3 | Ctr_pos() +    | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Confirm() +    | PROCLIB.PRG |
| 1x | 3 | 3 | 3 | DelRecs()      | BASEREV.PRG |
| 1x | 3 | 3 | 3 | Confirm() +    | PROCLIB.PRG |
| 2x | 3 | 3 | 3 | Loadrec() +    | FUNCLIB.PRG |
| 2x | 3 | 3 | 3 | Putrec() +     | FUNCLIB.PRG |
| 4x |   |   |   |                |             |

|    |        |                |              |
|----|--------|----------------|--------------|
| 3  | 3      | Shring_Arr()   | PROCLIB.PRG  |
| 2x | 3      | SayRec() +     | BASEREV.PRG  |
| 2x | 3      | Get_NewRow() + | BASEREV.PRG  |
| 2x | 3      | StartPos() +   | BASEREV.PRG  |
| 2x | 3      | ListRec() +    | BASEREV.PRG  |
| 2x | 3      | Ctr_pos() +    | PROCLIB.PRG  |
| 2x | 3      | RPW32()        | RPW32.PRG    |
| 1x | 3      | STOC() +       | FUNCLIB.PRG  |
| 1x | 3      | RPW33()        | RPW33.PRG    |
| 1x | 3      | RPW34()        | RPW34.PRG    |
| 1x | 3      | STOD() +       | FUNCLIB.PRG  |
| 2x | 3      | RPW35()        | RPW35.PRG    |
| 1x | RPW4() |                | RPW4.PRG     |
| 1x | 3      | MASK() +       | RPWPROC.PRG  |
| 1x | 3      | RPW41()        | RPW4.PRG     |
| 1x | 3      | STOD() +       | FUNCLIB.PRG  |
| 1x | 3      | STOC() +       | FUNCLIB.PRG  |
| 1x | 3      | RPW42()        | RPW4.PRG     |
| 1x | 3      | MASK() +       | RPWPROC.PRG  |
| 1x | 3      | STOC() +       | FUNCLIB.PRG  |
| 1x | 3      | RPW43()        | RPW4.PRG     |
| 1x | 3      | MASK() +       | RPWPROC.PRG  |
| 2x | 3      | STOC() +       | FUNCLIB.PRG  |
| 3x | 3      | SAMPLEPT()     | RSMPLPRT.PRG |
| 1x | 3      | PITCH()        | RSMPLPRT.PRG |
| 2x | 3      | SAMPLEHEADER() | RSMPLPRT.PRG |
| 4x | 3      | MEPTypCode()   | FUNCLIB.PRG  |
| 1x | 3      | STOD() +       | FUNCLIB.PRG  |
| 1x |        |                |              |

```

      RPW44()
RPW4.PRG

1x      MASK()+
RPWPROC.PRG

1x      STOC()+
FUNCLIB.PRG

1x      RPW5()
RPW5.PRG

1x      ListInDir()
RELEASE.PRG

5x      ExistFList()+
RELEASE.PRG

5x      YNResponse()+
RPWPROC.PRG

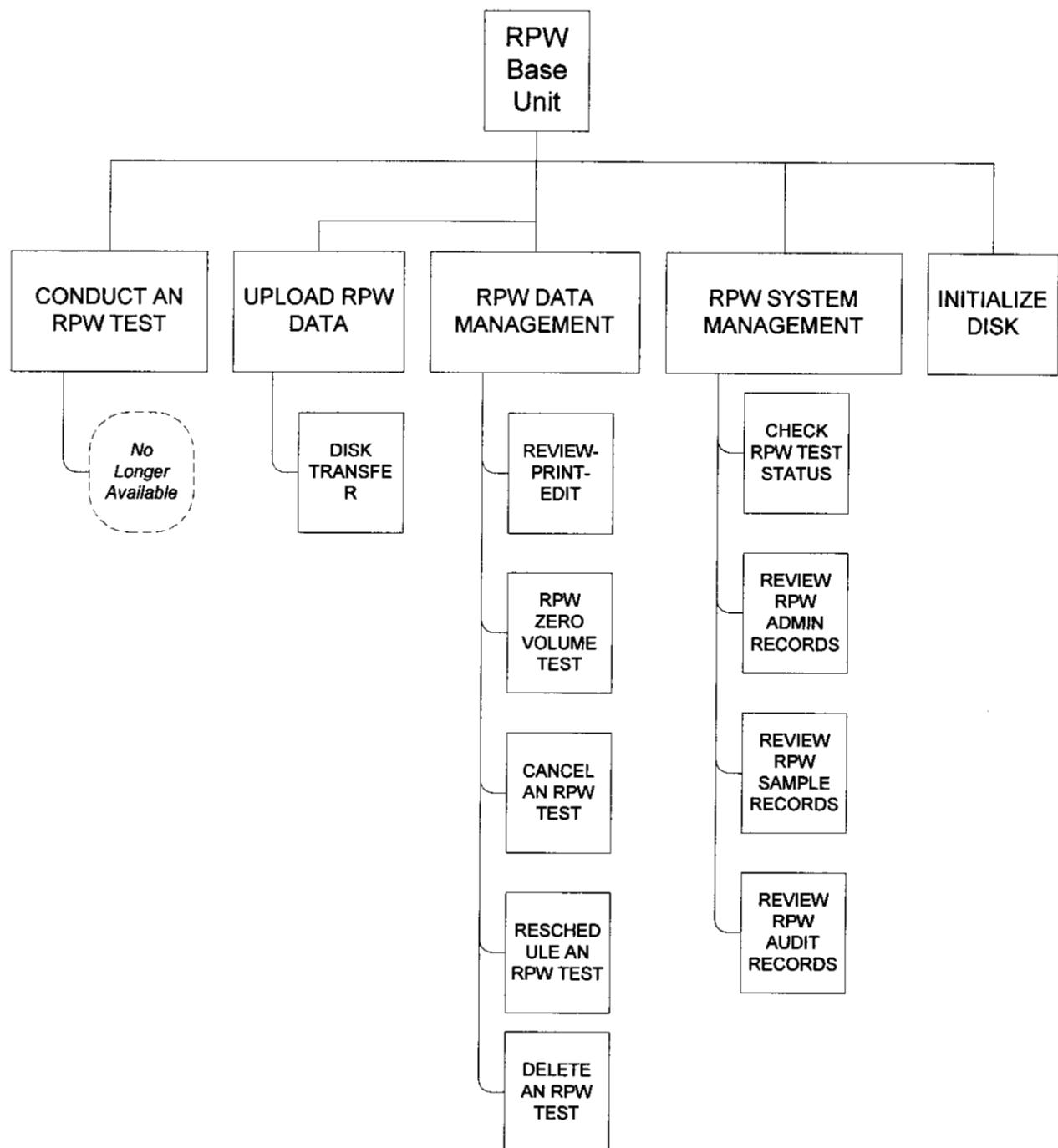
5x      AdvanceFiles()+
RELEASE.PRG      1x

```

Category: Project Summary  
 RPW Base Unit Summary

| Data Files       | Bytes | Lines | Comment | Bytes  | Lines | Code %  |
|------------------|-------|-------|---------|--------|-------|---------|
| XBASE Data Files | 0     | 0     | 0.00%   | 88215  | 9     | 100.00% |
| 1 category:      | 0     | 0     | 0.00%   | 88215  | 9     | 100.00% |
| <br>Source Code  |       |       |         |        |       |         |
| Declaration      | 540   | 16    | 26.57%  | 1492   | 39    | 73.43%  |
| Function         | 39267 | 1019  | 13.74%  | 246438 | 6710  | 86.26%  |
| Procedure        | 1987  | 68    | 30.67%  | 4492   | 147   | 69.33%  |
| Static Function  | 0     | 0     | 0.00%   | 22844  | 643   | 100.00% |
| 4 categories:    | 41794 | 1103  | 13.18%  | 275266 | 7539  | 86.82%  |
| <br>2 sections:  | 41794 | 1103  | 10.31%  | 363481 | 7548  | 89.69%  |

# **RPW BASE UNIT PROGRAM FLOWCHART**



# **RPW-CODES**

## **Computer System Documentation Description**

### **Section 3**

## **Mainframe**

## **INTRODUCTION**

The RPW Mainframe application is written in "Cobol".

Documentation for the RPW Mainframe application consists of the programs, divided into three main sections known as Control, Source, and Proc (JCLs) :

The Control files are as follows:

HSD85001  
HSD85002  
HSD85003  
HSD85004  
HSD85005

The Source programs are as follows:

HSD830c3  
HSD840ca  
HSD841c1  
HSD850c5  
HSD860c3

The Proc (JCLs) programs are included in this documentation and are listed below.

HSD5000s  
HSD8300q  
HSD8350t  
HSD8500t

The Mainframe Documentation consists of the following:

- Introduction
- System Narratives
- Program Flowcharts
- JCLs.

The CD with the source code for PQ3 2000 software release complements this technical documentation.

RPW MAINFRAME SYSTEM NARRATIVES

Upload RPW Data from Base Unit to Mainframe

JOB TITLE: Upload RPW data from base unit to mainframe

JOB NAME: RPWUPLD

WHEN RUN: As requested.

DESCRIPTION: This job will execute the program HSD805C0 which will upload RPW data from the base unit to the mainframe.

FILE STATUS: N/A

STEPS:

S01 - Execute COBOL program HSD805C0 to upload RPW data from base units to a VSAM file in the mainframe.

CONDITION: N/A

STEP INPUT: RPW Data

STEP OUTPUT: HSDHQV.VS805D01.RPWDATA VSAM file

RPW-DATA Transmit report

STEP PARAMETER: N/A

STEP SORT PARAMETER: N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY: Review error message. Correct problem and rerun this job. If problem cannot be corrected, skip this job, and produce a trouble report.

SETUP PROCEDURE: This job is setup by a CLIST program called 'RPWUPLD' which is executed from the base unit.

JOB RUN PROCEDURE: Check RPW-TRANSMIT-REPORT.

PRODUCTION HINTS:None

OUTPUT PROCEDURE:

PREVIOUS ERRORS:

CONSEQUENCES OF NOT RUNNING JOB: RPW data will not be uploaded to the mainframe.

RELATED JOBS and COMMENTS: N/A

**RPW Sample Selection Partitioning Process**

JOB TITLE: RPW Sample Selection Partitioning Process

JOB NAME:HSD5000Q

WHEN RUN:

DESCRIPTION: This job creates a sample file for RPW and ODIS

FILE STATUS: N/A

STEPS:

S01 - Execute SORT utility program to sort the MEPS master ZIP table file into a three-digit ZIP temporary file.

CONDITION:N/A

STEP INPUT:HSFRAN.PS001D01.ZIPTABLE.ORFEO

STEP OUTPUT:&&TEMP (Temporary Zip File)

STEP PARAMETER:

STEP SORT PARAMETER: HSDHQN.CODES.CONTROL(HSD50001)

SORT ORDER: Sorted by three-digit ZIP.

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S02 -Execute COBOL program HSD500C0 which creates the MSC reference table file for CODES project.

CONDITION:0

STEP INPUT:&&TEMP

STEP OUTPUT:&&TEMPI

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER:N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S04 - Execute the IDCAMS Utility program to delete old ODIS VSAM sample file and define a new ODIS VSAM sample file.

CONDITION: 0

STEP INPUT: HSDHQV.VS510D01.ODIS.SAMPLE

STEP OUTPUT: HSDHQV.VS510D01.ODIS.SAMPLE

STEP STEP PARAMETER: HSDHQN.CODES.CONTROL(HSD50002)

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S06 - Execute COBOL program HSD510C0 which loads the ODIS sample VSAM file with new sample data.

CONDITION:0

STEP INPUT: ZIPTABLE &&TEMP1

SAMPFILE HSB.HQ134D02.FY&FY.PQ0&QTR

SITEFILE HSDHQV.VS610D01.SITEFILE

FINFILE HSDHQN.PS001D01.ODIS.FINFILE

STEP OUTPUT: HSDHQV.VS510D01.ODIS.SAMPLE

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER:N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S08 - Execute IDCAMS utility program to delete old RPW VSAM file and define a new RPW sample VSAM file.

CONDITION:0

STEP INPUT:HSDHQV.VS800D01.RPW.SAMPLE  
STEP OUTPUT:HSDHQV.VS800D01.RPW.SAMPLE  
STEP PARAMETER:HSDHQN.CODES.CONTROL(HSD50003)  
STEP SORT PARAMETER:N/A  
SORT ORDER:N/A  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S10 - Execute SORT utility program to sort by FINMSTR-FIN  
CONDITION:0  
STEP INPUT:HSF.HQMAND07.SMPOFF&FY.PQ&QTR  
STEP OUTPUT:&&HSFSORT  
STEP PARAMETER:  
STEP SORT PARAMETER:HSDHQN.CODES.CONTROL(HSD50010)  
SORT ORDER:By FINMSTR-FIN  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S12 - Execute COBOL program HSD800C0 which loads the RPW sample  
VSAM file with new sample data.

CONDITION:0  
STEP INPUT:ZIPTABLE &&TEMP1  
SAMFILE HSF.HQ330T02.PQ0&QTR.FY&FY  
SITEFILE HSDHQV.VS610D01.SITEFILE  
FINFILEHSDHQN.PS001D01.ODIS.FINFILE  
FINMST&&HSFSORT  
STEP OUTPUT:HSDHQV.VS800D01.RPW.SAMPLE  
STEP PARAMETER:N/A  
STEP SORT PARAMETER:N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S14 - Execute COBOL program HSD522C0 in order to prepare the PDF-SITE code table for the 1<sup>st</sup> class service standard extract file processing.

CONDITION: 0

STEP INPUT: &&TEMP1

HSDHQN.PS001D01.SCFFILE

STEP OUTPUT: &&PDFSITE

STEP PARAMETER: N/A

STEP SORT PARAMETER:N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S16 - Execute SORT utility program to sort the PDF-SITE code table and eliminate duplicates.

CONDITION: 0

STEP INPUT: &&PDFSITE

STEP OUTPUT:&&PDFSITSR

STEP PARAMETER: N/A

STEP SORT PARAMETER:HSDHQN.CODES.CONTROL (HSD50009)

SORT ORDER: Sorted by PDF, Site

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S18 - Execute COBOL program HSD523C0 to extract 1<sup>st</sup> class Service standard data.

CONDITION:0

STEP INPUT: &&PDFSITSR HSBRAN.PS506T01.ODIS5061

STEP OUTPUT: &&STDXTRCT

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S20 - Execute COBOL program HSD524C0 to extract priority Service standard data and add to file &&STDXTRCT.

CONDITION: 0

STEP INPUT: &&TEMP1 HSDHQN.PS506D01.ODIS5063

STEP OUTPUT: &&STDXTRCT

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S22 - Execute SORT utility program to sort the combined (1<sup>st</sup> class and priority) service standard extract file.

CONDITION: 0

STEP INPUT: &&STDXTRCT

STEP OUTPUT: &&SSTDSORT

STEP PARAMETER:N/A

STEP SORT PARAMETER:HSDHQN.CODES.CONTROL(HSD50004)  
SORT ORDER: Sorted by SORT-CODE, CLASS, DESTINATION-  
SECT-CENTER-FAC, ORIGIN-SECT-CENTER-FAC  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S24 - Execute IDCAMS utility program to delete the old SERVSTD  
VSAM file and define a new SERVSTD VSAM file.

CONDITION:0  
STEP INPUT: HSDHQV.VS525D01.SERVSTD  
STEP OUTPUT:HSDHQV.VS525D01.SERVSTD  
STEP PARAMETER: HSDHQN.CODES.CONTROL (HSD50005)  
STEP SORT PARAMETER:N/A  
SORT ORDER: N/A  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S26 - Execute COBOL program HSD525C0 to load the service  
standard extract VSAM file with new service standard data.

CONDITION: 0  
STEP INPUT: &&SSTDSORT  
STEP OUTPUT:HSDHQV.VS525D01.SERVSTD  
STEP PARAMETER: N/A  
SORT ORDER: N/A  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S28 - Execute IDCAMS utility program to create sequential ODIS  
partitioned file from the VSAM file.

CONDITION:0  
STEP INPUT:HSDHQV.VS510D01.ODIS.SAMPLE  
STEP OUTPUT: &&ODISAMPL  
STEP PARAMETER:HSDHQN.CODES.CONTROL (HSD600U7)  
STEP SORT PARAMETER: N/A  
SORT ORDER: N/A  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S30 - Execute SORT utility program to extract fields and sort the ODIS sample file while eliminating duplicates.

CONDITION:0  
STEP INPUT: &&ODISAMPL  
STEP OUTPUT: &&SMPCSTRT  
STEP PARAMETER:  
STEP SORT PARAMETER:HSDHQN.CODES.CONTROL (HSD50006)  
SORT ORDER: Sorted by SAMPLE-CELL, STRATA-CODE SITE  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S32 - Execute COBOL program HSD527C0 to match ODIS sample extract and variables files

CONDITION:0  
STEP INPUT: &&SMPCSTRT HSF.HQ033D01.FY&FY.&QTR  
STEP OUTPUT: &&VRBLRAW  
STEP PARAMETER:N/A  
STEP SORT PARAMETER:N/A  
SORT ORDER:N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S34 - Execute SORT utility program to sort the variables file.

CONDITION:0

STEP INPUT: &&VRBLRAW

STEP OUTPUT: &&VRBLSRT

STEP PARAMETER:

STEP SORT PARAMETER: HSDHQN.CODES.CONTROL(HSD50007)

SORT ORDER: By SITE, SAMPLE-CELL, STRATA-CODE

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S36 - Execute IDCAMS utility program to delete the old variable VSAM file and define a new variable VSAM file.

CONDITION:0

STEP INPUT: HSDHQV.VS528D01.VARIABLE

STEP OUTPUT: HSDHQV.VS528D01.VARIABLE

STEP PARAMETER:HSDHQN.CODES.CONTROL(HSD50008)

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S38 - Execute COBOL program HSD528C0 to load variable VSAM file with new variable data.

CONDITION:0

STEP INPUT: &&VRBLSRT HSDHQV.VS610D01.SITEFILE

STEP OUTPUT: HSDHQV.VS528D01.VARIABLE

STEP PARAMETER:N/A

STEP SORT PARAMETER: N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S40 - Execute DSUTIL program to backup the ZIPTABLE ORFEO file.

CONDITION:0

STEP INPUT: HSFRAN.PS001D01.ZIPTABLE.ORFEO

STEP OUTPUT: HSDHQN.PS001D01.ZIPTABLE.ORFEO (+1)

STEP PARAMETER:N/A

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

SETUP PROCEDURE: This job is set up by a CLIST program called RPW. Select option one, 'submit ODIS/RPW sample selection partition job' and the FY, QTR, BEGQTR, and ENDQTR, information.

JOB RUN PROCEDURE: Check output JCL for correct condition codes.

PRODUCTION HINTS: Make sure to enter the correct parameters.

OUTPUT PROCEDURE: The output will go to SAR.

PREVIOUS ERRORS: Enter wrong parameters

CONSEQUENCES OF NOT RUNNING JOB: The RPW/ODIS sample file will not be generated for the current quarter.

### Creating the RPW Sample Selection Extract File

JOB TITLE:Creating the RPW Sample Selection Extract File

JOB NAME: HSD8300Q

WHEN RUN: At the beginning of each calendar quarter.

DESCRIPTION:This job creates the RPW Sample Selection Extract File for each calendar quarter.

FILE STATUS:N/A

STEPS:

S01 - Execute the sort utility program to sort the MEPS master ZIP table on three-digit zip into a temporary file.

CONDITION:N/A

STEP INPUT: HSFRAN.PS001D01.ZIPTABLE.ORFEO

STEP OUTPUT: &&TEMP (Temporary ZIP file)

STEP PARAMETER:N/A

STEP SORT PARAMETER:HSDHQN.CODES.CONTROL (HSD83001)

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S02 - Execute COBOL program HSD500C0. This program reads theZIP File sorted by three-digit ZIP Code and if it is numeric, writes the record to a temporary output file.

CONDITION:0

STEP INPUT: &TEMP (MSCTABLE)

STEP OUTPUT: &&TEMP1

STEP PARAMETER:N/A

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S03 - Execute SORT utility program to sort  
HSF.HQMAND07.SMPOFF&FY file by FINMSTR-FIN.

CONDITION:0

STEP INPUT: HSF.HQMAND07.SMPOFF&FY..PQ&QTR

STEP OUTPUT: &&HSFSORT

STEP PARAMETER:

STEP SORT PARAMETER:HSDHQN.CODES.CONTROL(HSD83002)

SORT ORDER: By FINMSTR-FIN

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S04 - Execute COBOL program HSD830C0 to process and sort sample  
records for the selected quarter, outputting a temporary  
extract file, an exception report, and a control report.

CONDITION:0

STEP INPUT:ZIPTABLE &&TEMP1,

SAMPLE HSF.HQ330T02.PQ0&QTR.FY&FY FINMAST &&HSFSORT

FINEXCP HSDHQN.PS001D01.ODIS.FINFILE

STEP OUTPUT:&&EXTRACT (Temporary Extract File)

Exception Report

Control Report

STEP PARAMETER: N/A

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S05 - Execute IDCAMS utility program to delete old RPW VSAM extract file and define a new RPW VSAM extract file.

CONDITION: 0

STEP INPUT: HSDHQV.VS830D01.EXTRACT (old)

STEP OUTPUT: HSDHQV.VS830D01.EXTRACT (An empty file)

STEP PARAMETER: HSDHQN.CODES.CONTROL(HSD83003)

STEP SORT PARAMETER:N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES: 0

S06 - Execute IDCAMS utility PROGRAM to load new RPW VSAMextract file created in step 04.

CONDITION: 0

STEP INPUT: &&EXTRACT (Temporary Extract File)

STEP OUTPUT:VSAM FILE: HSDHQV.VS830D01.EXTRACT

STEP PARAMETER: HSDHQN.CODES.CONTROL(HSD600U7)

STEP SORT PARAMETER: N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

SETUP PROCEDURE: This job is set up by a CLIST program called RPW. Select option '2' and enter the FY, QTR, BEGQTR, and END QTR information.

JOB RUN PROCEDURE: Check output JCL for the correct condition codes.

PRODUCTION HINTS:Make sure to enter the correct parameters.

OUTPUT PROCEDURE:The output will go to SAR.

PREVIOUS ERRORS: Wrong Parameters

CONSEQUENCES OF NOT RUNNING JOB: The RPW Extract file will not be generated for the current quarter.

RELATED JOBS and COMMENTS: N/A

## RPW Pre-Checkin Process

JOB TITLE: RPW Pre-Checkin Process

JOB NAME:HSD8350T

WHEN RUN: Weekly (Before Checkin)

DESCRIPTION: This program is executed on a weekly basis to determine which RPW sites did not transmit data for the processing week. One COBOL program in this process reads the RPW Master Data file, the RPW Extract file, and the RPW Site file to determine which RPW sites have not transmitted data. The other COBOL program is used to create two temporary files and a VSAM Site file.

FILE STATUS:

STEPS:

S01 - Execute IDCAMS utility program to repro the RPW master data file to a temporary sequential file.

CONDITION: N/A

STEP INPUT: HSDHQV.VS805D01.RPWDATA

STEP OUTPUT: &&PWDATA

STEP PARAMETER: HSDHQN.CODES.CONTROL (HSD600U7)

STEP SORT PARAMETER: N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S02 - Execute COBOL program HSD835C3 to determine which, if any, online RPW sites didn't transmit any data for the processing week. The program also writes two reports concerning RPW delinquencies.

CONDITION:0

STEP INPUT:

&&RPWDATAHSDHQV.VS610D01.SITEFILEHSDHQV.VS830D01.EXTRACT

STEP OUTPUT: Transmit Report Pre-Checkin Summary Report

STEP PARAMETER: N/A

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S03 - Execute sort utility program to sort RPW master data file into a temporary file.

CONDITION:0

STEP INPUT: &&RPWDATA

STEP OUTPUT: &&TEMP

STEP PARAMETER:

STEP SORT PARAMETER:HSDHQN.CODES.CONTROL(HSD83501)

SORT ORDER: Sorted by Finance \_No. and Test-ID

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S04 - Execute COBOL program HSD841C which is run prior to Checkin to ensure that all the sites have transmitted their RPW tests to the mainframe.

CONDITION: 0

STEP INPUT: &&TEMP HSDHQV.VS830D01.EXTRACT

HSDHQV.VS610D01.SITEFILE

STEP I/O&&EXTEMP &&RPWTEMP

STEP OUTPUT: RPW Pre-Checkin Delinquent test report

STEP PARAMETER: N/A

STEP SORT PARAMETER:N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

SETUP PROCEDURE: This job is run on request usually before the checkin process, called RPW. To run this job enter the week beginning date and week ending date information.

JOB RUN PROCEDURE: Check output JCL for correct condition codes.

Production Hints: Ensure that you enter the correct parameters.

OUTPUT PROCEDURE: The output will go to the SAR system.

PREVIOUS ERRORS: Entered wrong parameters.

Consequences of not Running Job: The RPW Pre-Checkin Summary report will not be generated.

RELATED JOBS and COMMENTS:N/A

## RPW CHECK-IN PROCESS

JOB TITLE: RPW CHECK-IN PROCESS

JOB NAME:HSD8500T

WHEN RUN: WEEKLY

DESCRIPTION: This process takes the RPW data and weekly RPW audit data which contains all data records transmitted the previous week. This data is then checked for data entry errors, general accuracy, and completeness.

FILE STATUS:

STEPS:

S01 - Execute IDCAMS utility program to copy weekly RPW data to temporary dataset.

CONDITION:N/A

STEP INPUT:HSDHQV.VS805D01.RPWDATA

STEP OUTPUT:HSDHQN.PS805D01.RPWDATA

STEP PARAMETER:HSDHQN.CODES.CONTROL(HSD600U7)

STEP SORT PARAMETER: N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S02 - Execute IDCAMS utility program to copy weekly RPW audit data to temporary dataset.

CONDITION:0

STEP INPUT:HSDHQV.VS805D01.RPWAUDIT

STEP OUTPUT:HSDHQN.PS805D01.RPWAUDIT

STEP PARAMETER:HSDHQN.CODES.CONTROL(HSD600U7)

STEP SORT PARAMETER:N/A

SORT ORDER: N/A

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S04 - Execute IDCAMS utility program to copy weekly data to tape and create a new VSAM file.

CONDITION:0

STEP INPUT:HSDHQN.PS001D01.BLANK150

HSDHQV.VS805D01.RPWDATA

STEP OUTPUT:HSDHQN.PS805T01.RPWDATA.RPW&FY&AP&WK

(TAPE)

HSDHQV.VS805D01.RPWDATA

(NEW EMPTY RPW VSAM FILE)

PARAMETER:HSDHQN.CODES.CONTROL(HSD85001)

SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S06 - Execute IDCAMS utility program to copy weekly audit data to tape and create a new VSAM file.

CONDITION:0

STEP INPUT:HSDHQN.PS001D01.BLANK150

HSDHQV.VS805D01.RPWAUDIT

STEP OUTPUT:HSDHQN.PS805T01.RPWAUDIT (TAPE)

HSDHQV.VS805D01.RPWAUDIT

NEW EMPTY RPW AUDIT VSAM FILE)

STEP PARAMETER:HSDHQN.CODES.CONTROL(HSD85002)

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S08 - Execute SORT program to sort weekly RPW data that was created in step '1'.

CONDITION:0

STEP INPUT:HSDHQN.PS805D01.RPWDATA

HSDHQN.PS840001.EDITFILE.

STEP OUTPUT:&&RPWDATA

STEP PARAMETER:

STEP SORT PARAMETER:HSDHQN.CODES.CONTROL(HSD85003)

SORT ORDER: Sorted by TEST-ID, MAIL-CODE RPW-PIECE TEST-  
DATE and VERSION

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S10 - Execute SORT utility program to sort weekly RPW audit data that was created in step 6.

CONDITION:0

STEP INPUT:HSDHQN.PS805D01.RPWAUDIT

STEP OUTPUT:&&AUDITRPW

STEP PARAMETER:

STEP SORT PARAMETER: HSDHQN.CODES.CONTROL(HSD85003)

SORT ORDER: Sorted by TEST-ID, MAIL-CODE RPW-PIECE TEST-  
DATE and VERSION

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S14 - Execute IDCAMS utility program to back up to tape the sample extract VSAM file prior to check-in run.

CONDITION:0

STEP INPUT:HSDHQV.VS830D01.EXTRACT

STEP OUTPUT:HSDHQN.PS830T01.EXTRACT1.RPW&FY&AP&WK  
(OUTPUT TO TAPE)

STEP PARAMETER:HSDHQN.CODES.CONTROL(HSD600U7)

STEP SORT PARAMETER: N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S16 - Execute COBOL program HSD840C0 which processes the RPW data uploaded from the base unit to the mainframe. This program validates and checks in RPW data and generates and validates output file. It also generates reports for feedback to headquarters.

CONDITION:0

STEP INPUT:HSDHQV.VS610D01.SITEFILE

&&RPWDATA

&&AUDITRPW

INPUT/OUTPUT: HSDHQV.VS830D01.EXTRACT

STEP OUTPUT:&&RPWOUT

&&AUDOUT

HSDHQN.PS840D01.EDITFILE.RPW&FY&AP&WK

HSDHQN.PS840D01.FUTUREWK.RPW&FY&AP&WK

&&RESCHED

&&ERROR

&&STATUS

&&HSD840P1 (REPORT SHOWING RESULTS;

tained When Processing RPW DATA)  
&&HSD840P2 (REPORT SHOWING THE RECORD,  
Level Errors Encounter)&&HSD840P3 REPORT SHOWING THE TEST,  
Level Errors Encountered)

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S18 - Execute IDCAMS utility program to back up sample extract  
VSAM file after check-in run.

CONDITION:0

STEP INPUT:HSDHQV.VS830D01.EXTRACT

STEP OUTPUT:HSDHQN.PS830T01.EXTRACT2.RPW&FY&AP&WK  
(TAPE)

STEP PARAMETER:HSDHQN.CODES.CONTROL (HSD600U7)

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

S20 - Execute COBOL program HSD850C to create the RPW status  
report file.

CONDITION:0

STEP INPUT:HSDHQV.VS610D01.SITEFILE

&&STATUS

STEP OUTPUT:&&HSD850P1 (SITE INFORMATION REPORT)

&&HSD850P2 (CONTROL REPORT OF RECORDS)

&&HSD850P3 (CONTROL REPORT OF TESTS)

&&HSD850P4 (SITE MSC STATUS REPORT)  
&&HSD850P5 (SUMMARY OF ACCEPT RECORD  
FILE)

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S22 - Execute COBOL program HSD860C to create the RPW  
reschedule report and merge it with the status report.

CONDITION:N/A

STEP INPUT:&&RESCHED

STEP OUTPUT:&&HSD860P1 (RESCHEDULE ACTION LIST  
REPORT)

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S24 - Execute the DSUTIL utility program to create an RPW output data tape.

CONDITION:0

STEP INPUT:&&RPWDATA

STEP OUTPUT:HSDHQN.PS840T01.CODES.RPW&FY.&AP.&WK  
(TAPE)

STEP PARAMETER:N/A

STEP SORT PARAMETER: N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S26 - Execute the DSUTIL utility program to create an RPW output data tape.

CONDITION: 0

STEP INPUT: &&RPWOUT

STEP OUTPUT: HSDHQN.PS840T02.CODESOUT.RPW&FY.&AP.&WK.  
(TAPE)

STEP PARAMETER: N/A

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S28 - Execute the DSUTIL utility program to create an RPW output reschedule data tape.

CONDITION:0

STEP INPUT:&&RESCHED

STEP OUTPUT:HSDHQN.PS840T03.RESCHED.RPW&FY.&AP.&WK.

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A  
SORT ORDER:  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S30 - Execute the DSUTIL utility program to create an RPW error output data tape.

CONDITION:0  
STEP INPUT:&&ERROR  
STEP OUTPUT:HSDHQN.PS840T04.ERROR.RPW&FY.&AP.&WK  
STEP PARAMETER:N/A  
STEP SORT PARAMETER:N/A  
SORT ORDER:  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S32 - Execute the DSUTIL utility program to create an RPW audit output data tape.

CONDITION:0  
STEP INPUT:&&AUDOUT  
STEP OUTPUT:HSDHQN.PS840T05.AUDIT.RPW&FY.&AP.&WK  
STEP PARAMETER:N/A  
STEP SORT PARAMETER:N/A  
SORT ORDER:  
ACCEPTABLE CONDITION CODES:0  
ERROR RECOVERY:

S36 - Execute the DSUTIL utility program to create an RPW status report tape.

CONDITION:N/A

STEP INPUT:HSDHQN.PS001D01.RPW.TEXT

&&HSD840P1

&&HSD840P2

&&HSD840P3

&&HSD850P1

&&HSD850P2

&&HSD850P3

&&HSD850P4

&&HSD850P5

&&HSD860P1

STEP OUTPUT:HSDHQN.PS850T01.STATRPT.RPW&FY.&AP.&WK

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S38 - Execute the DSUTIL utility program to create an RPW input audit data tape.

CONDITION:0

STEP INPUT:&&AUDITRPW

STEP OUTPUT:HSDHQN.PS840T01.AUDTDATA.RPW&FY.&AP.&WK

(TAPE)

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S40 - Execute IDCAMS utility program to create VSAM text file

CONDITION:0

STEP INPUT:HSDHQV.VS850D01.RPW.TEXT

STEP OUTPUT:HSDHQV.VS850D01.RPW.TEXT

STEP PARAMETER:DSN=HSDHQN.CODES.CONTROL(HSD85004)

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S42 - Execute COBOL program HSD686C0 to load text file VSAM from MSC status report dataset.

CONDITION:0

STEP INPUT:HSDHQV.VS610D01.SITEFILE

&&HSD850P4 SITE MSC STATUS REPORT

INPUT/OUTPUT: &&TEMP12 (TEXTSORT)

STEP OUTPUT:HSDHQV.VS850D01.RPW.TEXT

STEP PARAMETER:N/A

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S44 - Execute the IEFBR14 utility program to delete prior week's HQ text.

CONDITION:0

STEP INPUT:HSDHQN.PS001D01.RPW.TEXT

STEP OUTPUT:UNCATALOG PRIOR WEEK HEADQUARTERS TEXT  
FILE

STEP PARAMETER:

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S46 - Execute the IEBGENER utility program to create a copy of the current week's HQ text.

CONDITION:0

STEP INPUT:HSDHQN.PS850D01.HQTRS.TEXT

STEP OUTPUT:HSDHQN.PS001D01.RPW.TEXT

STEP PARAMETER:

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S48 - Execute the IEFBR14 utility program to delete RPW data and RPW audit files.

CONDITION:0

STEP INPUT:HSDHQN.PS805D01.RPWDATA

HSDHQN.PS805D01.RPWAUDIT

STEP OUTPUT:

STEP PARAMETER:N/A

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S50 - Execute IDCAMS utility program to repro next week's data to an RPW VSAM file to be processed the following week.

CONDITION:0

STEP INPUT:HSDHQN.PS840D01.FUTUREWK.RPW&FY&AP&WK

STEP OUTPUT:HSDHQV.VS805D01.RPWDATA

STEP PARAMETER:HSDHQN.CODES.CONTROL (HSD600U7)

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES: 0

ERROR RECOVERY:

S52 - Execute the IEBPTPCH utility program to route a copy of report to headquarters and the Northern VA service center.

CONDITION:0

STEP INPUT:HSDHQN.PS001D01.RPW.TEXT

&&HSD840P1

&&HSD840P2

&&HSD840P3

&&HSD850P1

&&HSD850P2

&&HSD850P3

&&HSD850P4

&&HSD850P5

&&HSD860P1

STEP OUTPUT:REPORT

STEP PARAMETER:HSDHQN.CODES.CONTROL (HSD85005)

STEP SORT PARAMETER:N/A

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S54 - Execute the IEBGENER utility program to move a backup copy of uploaded RPW data to tape.

CONDITION:0

STEP INPUT:HSDHQN.PS805D01.BACKUP.RPWDATA

STEP OUTPUT:HSDHQN.PS805T01.BACKUP.RPWDATA.RPW&FY&AP&WK

STEP PARAMETER:N/A

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

S56 - Execute the IEBGENER utility program to create a new backup file with a blank record.

CONDITION:0

STEP INPUT:HSDHQN.PS001D01.BLANK150

STEP OUTPUT:HSDHQN.PS805D01.BACKUP.RPWDATA

STEP PARAMETER:

STEP SORT PARAMETER:

SORT ORDER:

ACCEPTABLE CONDITION CODES:0

ERROR RECOVERY:

SETUP PROCEDURE: This job is run on a weekly basis. It is automatically submitted by the mainframe.

Parameters used are FY, AP, WK, ENDWEEK.

JOB RUN PROCEDURE: Check output JCL for correct condition code.

PRODUCTION HINTS: Ensure that you enter the correct

parameters.

OUTPUT PROCEDURE:The output will go to SAR system.

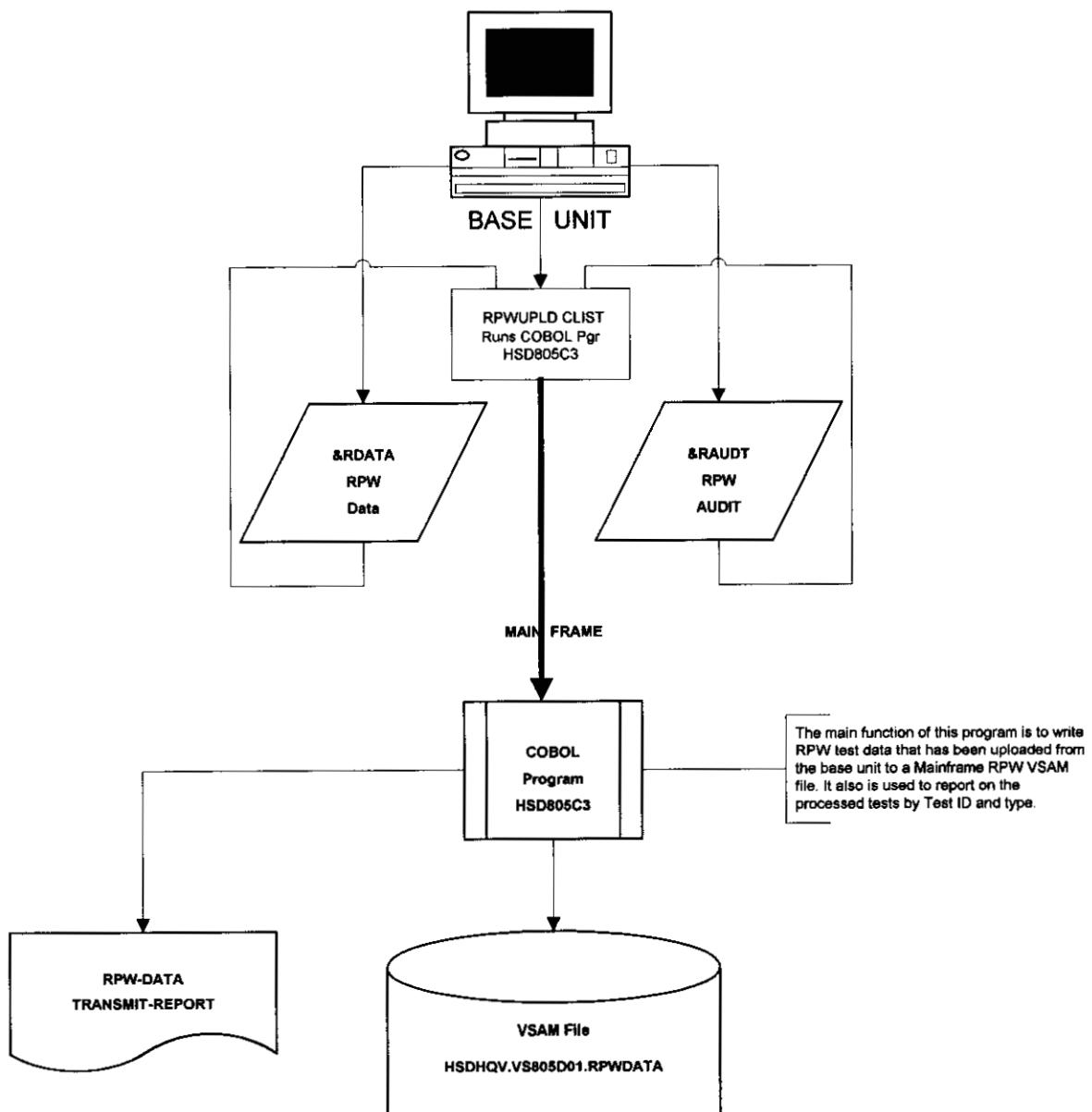
PREVIOUS ERRORS:Entered wrong parameters.

CONSEQUENCES OF NOT RUNNING JOB: The RPW data will not be checked in and output files will not be generated.

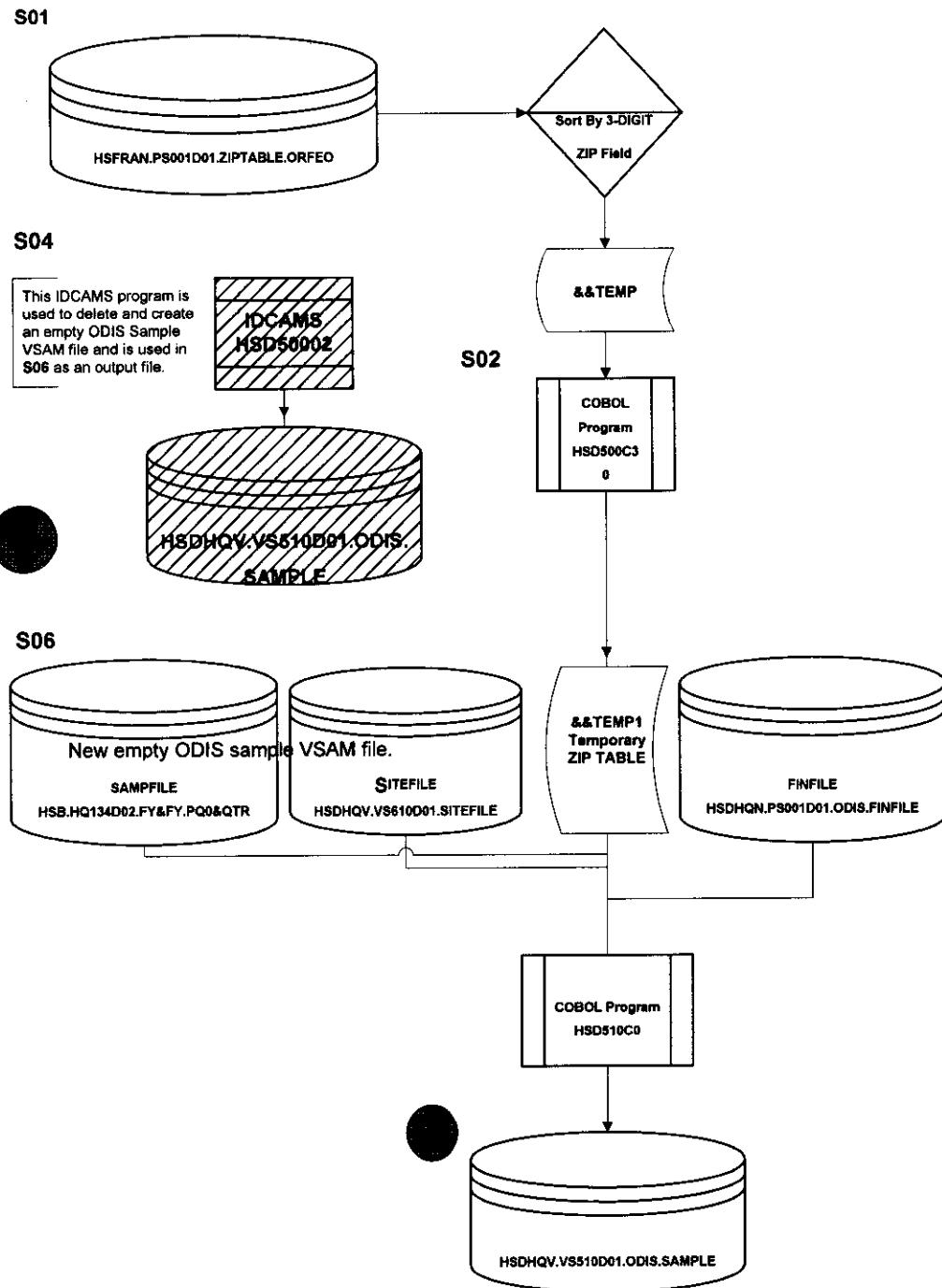
RELATED JOBS AND COMMENTS:

# **RPW Mainframe System Flowcharts**

## RPW Upload Process



**ODIS/RPW Sample Selection Partitioning Process**  
**(HSD5000Q)**

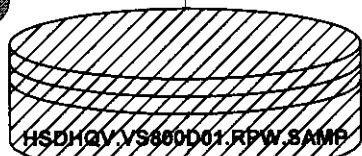


## S08

This IDCAMS program is used to delete and create old RPW VSAM files and is used in S12 as an output file.



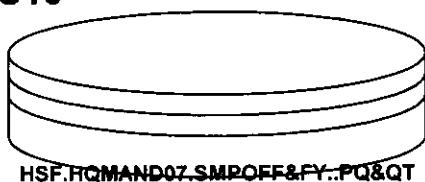
(B)



LE

New empty RPW sample  
file.

## S10

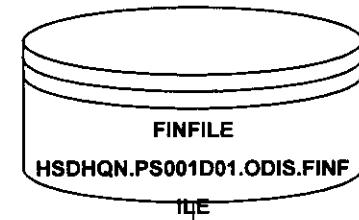
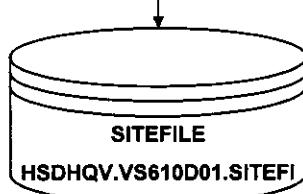
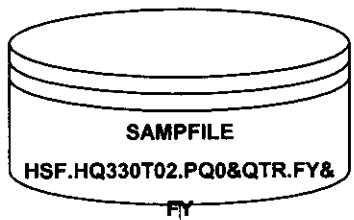


Sort BY  
FINMSTR-FIN

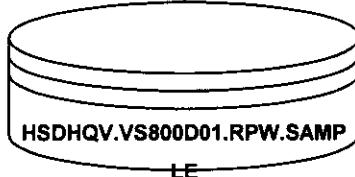
&&HSFSORT

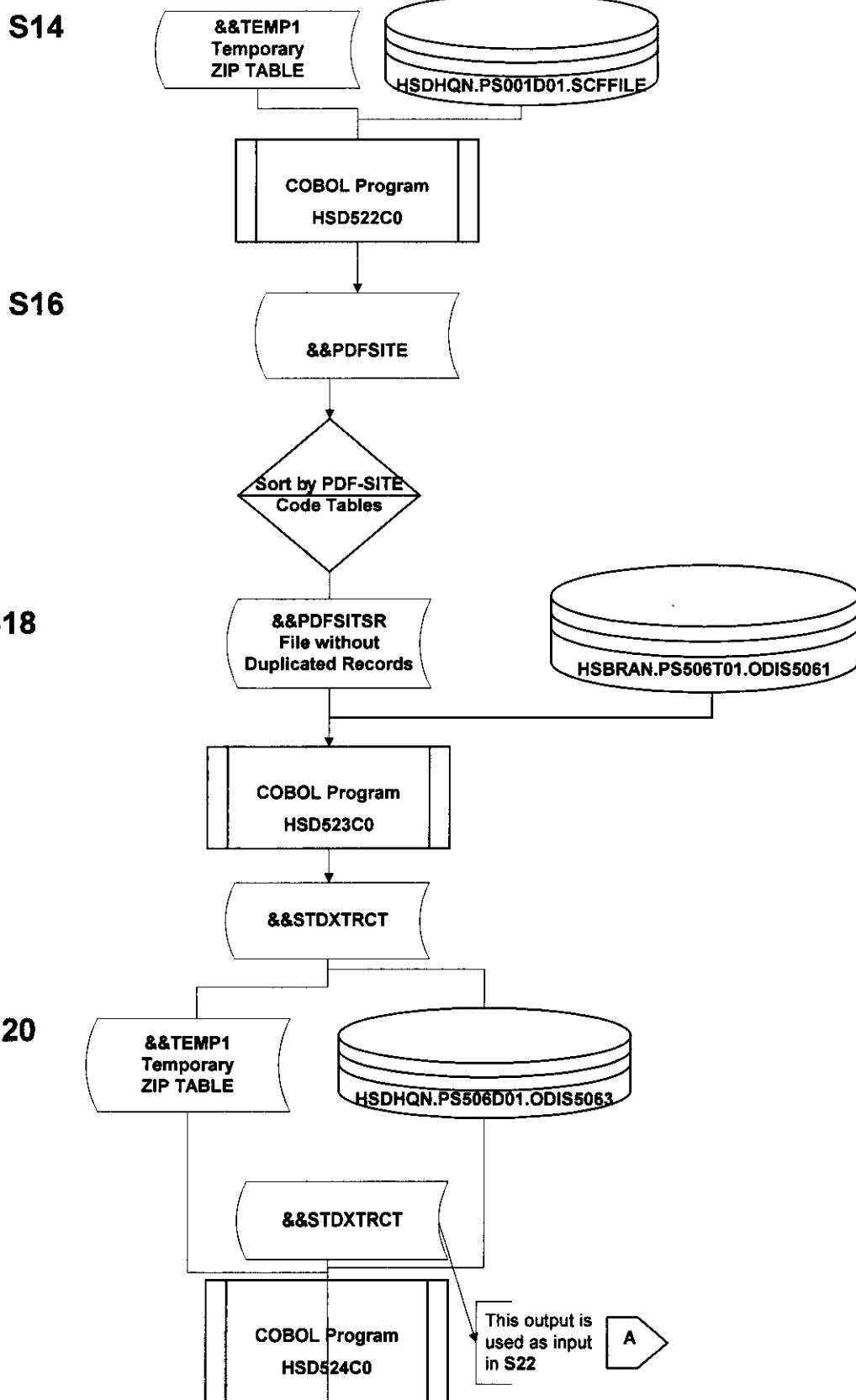
## S12

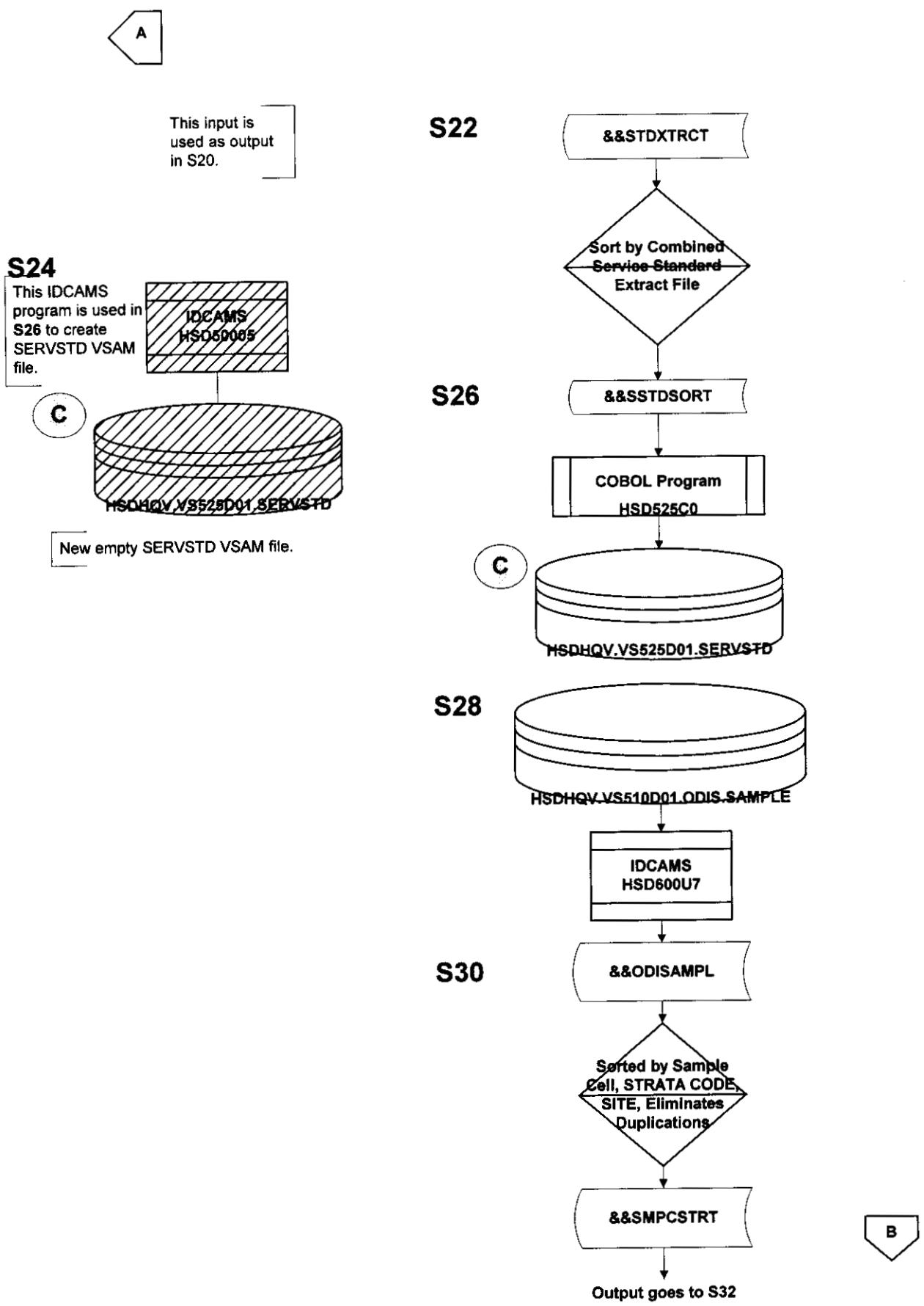
&&TEMP1  
Temporar  
y ZIP  
TABLE

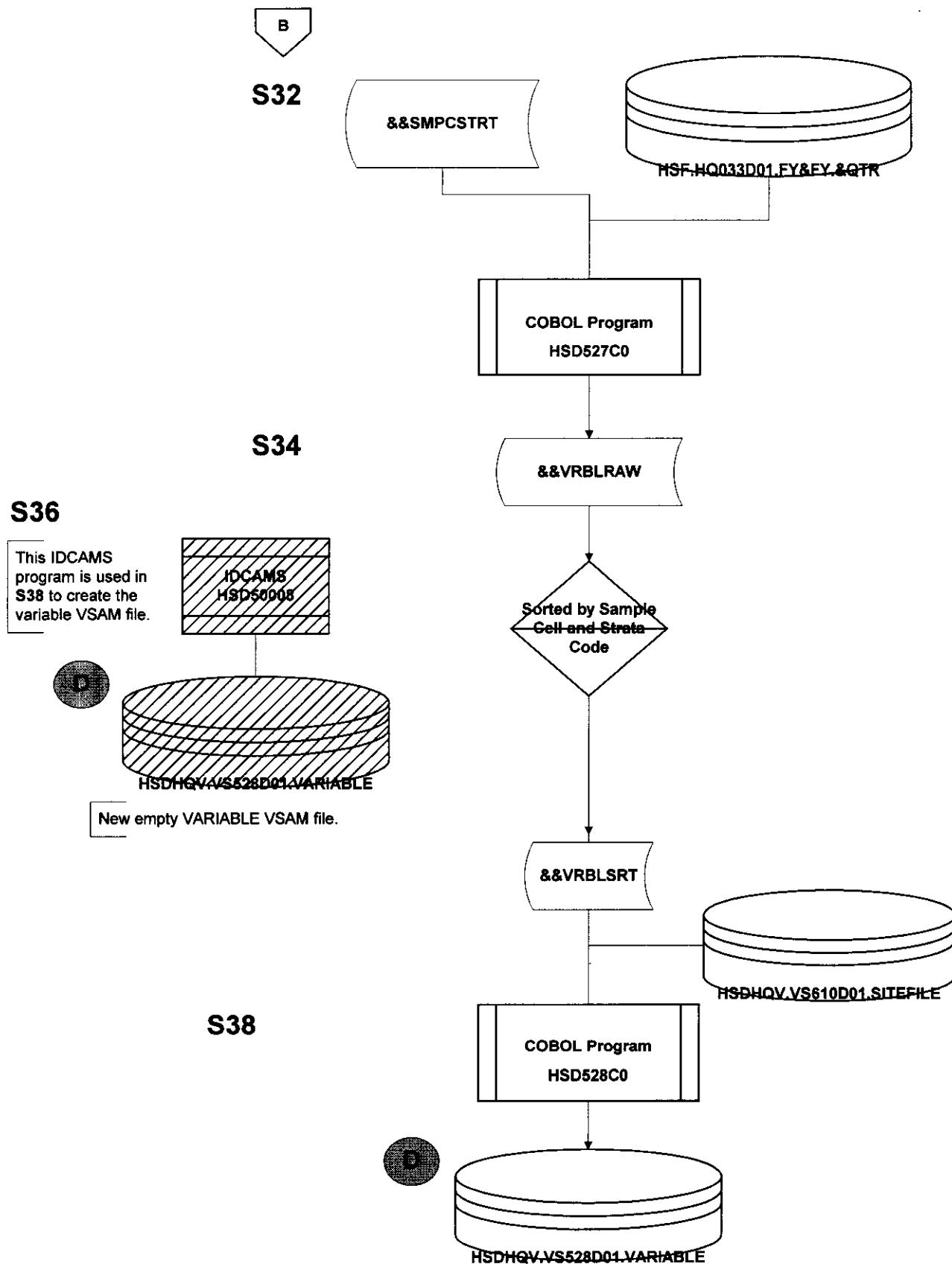


COBOL Program  
HSD800C0

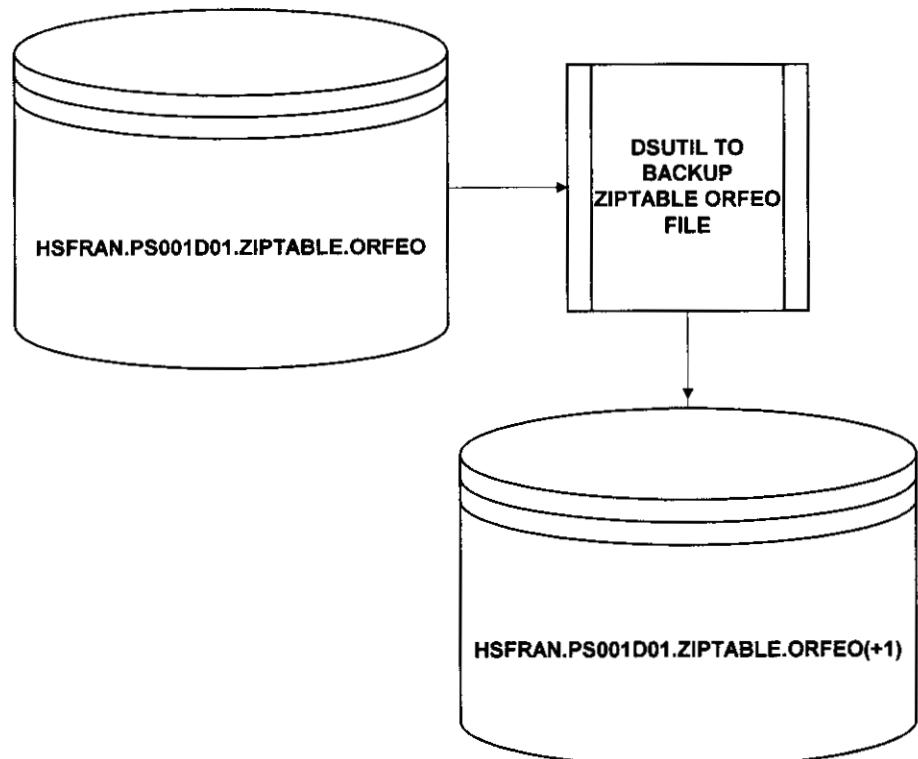






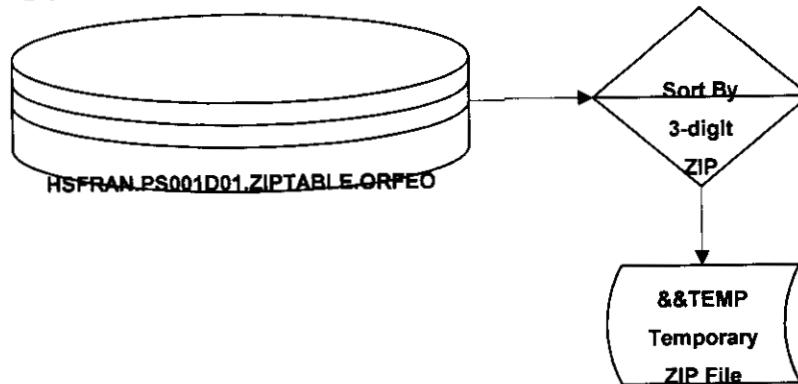


S40

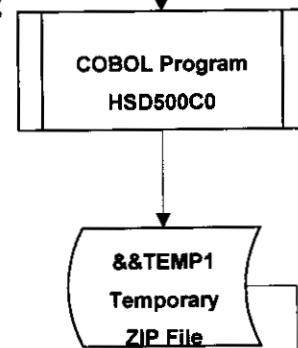


**RPW Sample Extract Process**  
**(HSD8300Q)**

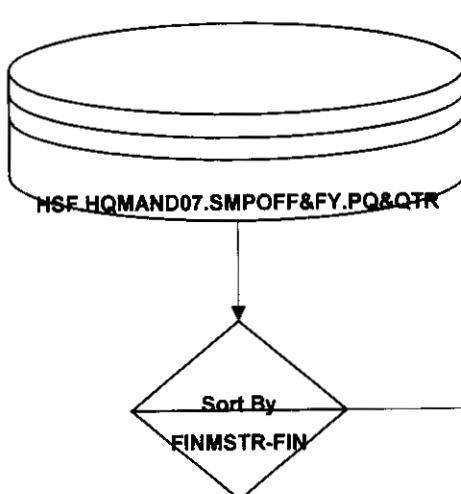
**S01**



**S02**



**S03**



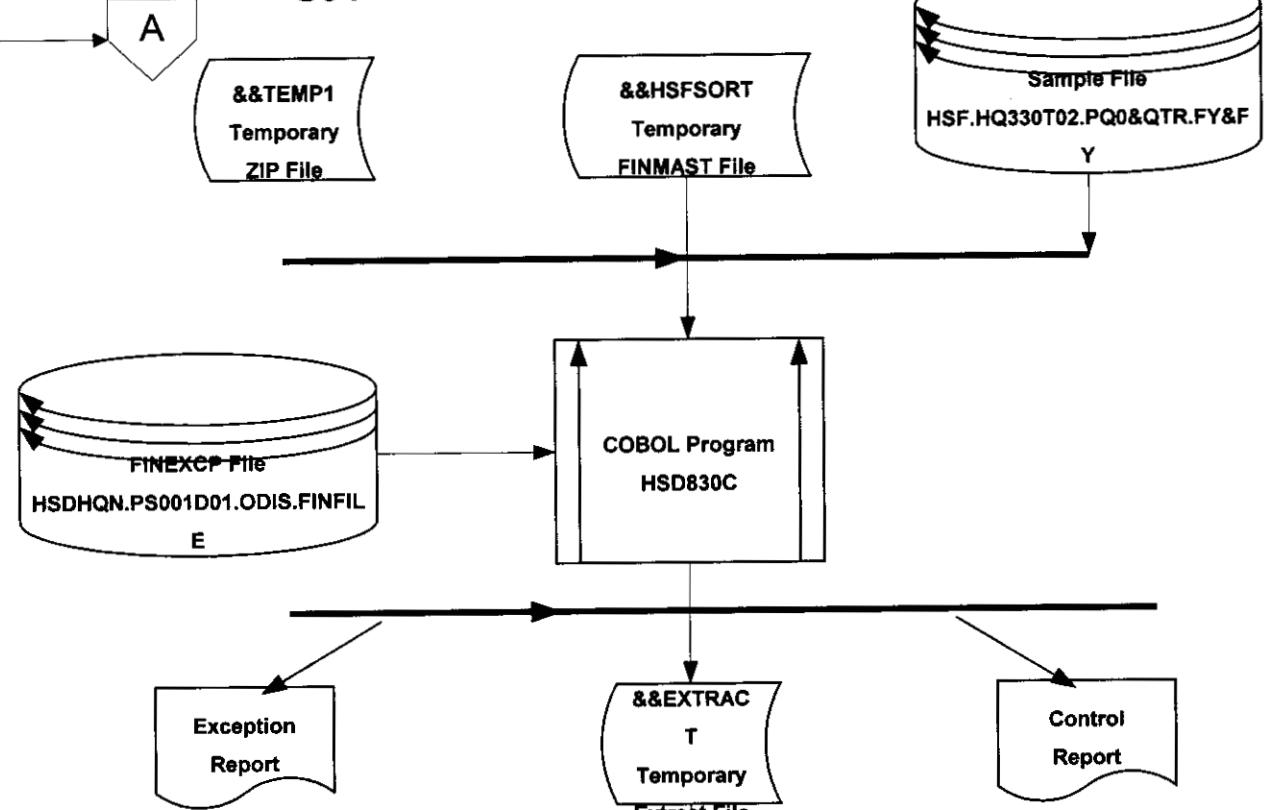
These two temporary files are used as input into S04

A

**RPW Sample Extract Process**

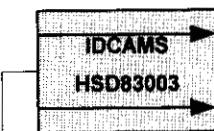
(HSD8300Q)

**S04**



**S05**

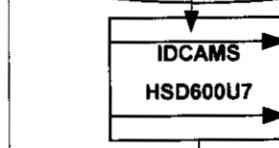
This IDCAMS program is used to delete the old RPW VSAM Extract file, and define a new one

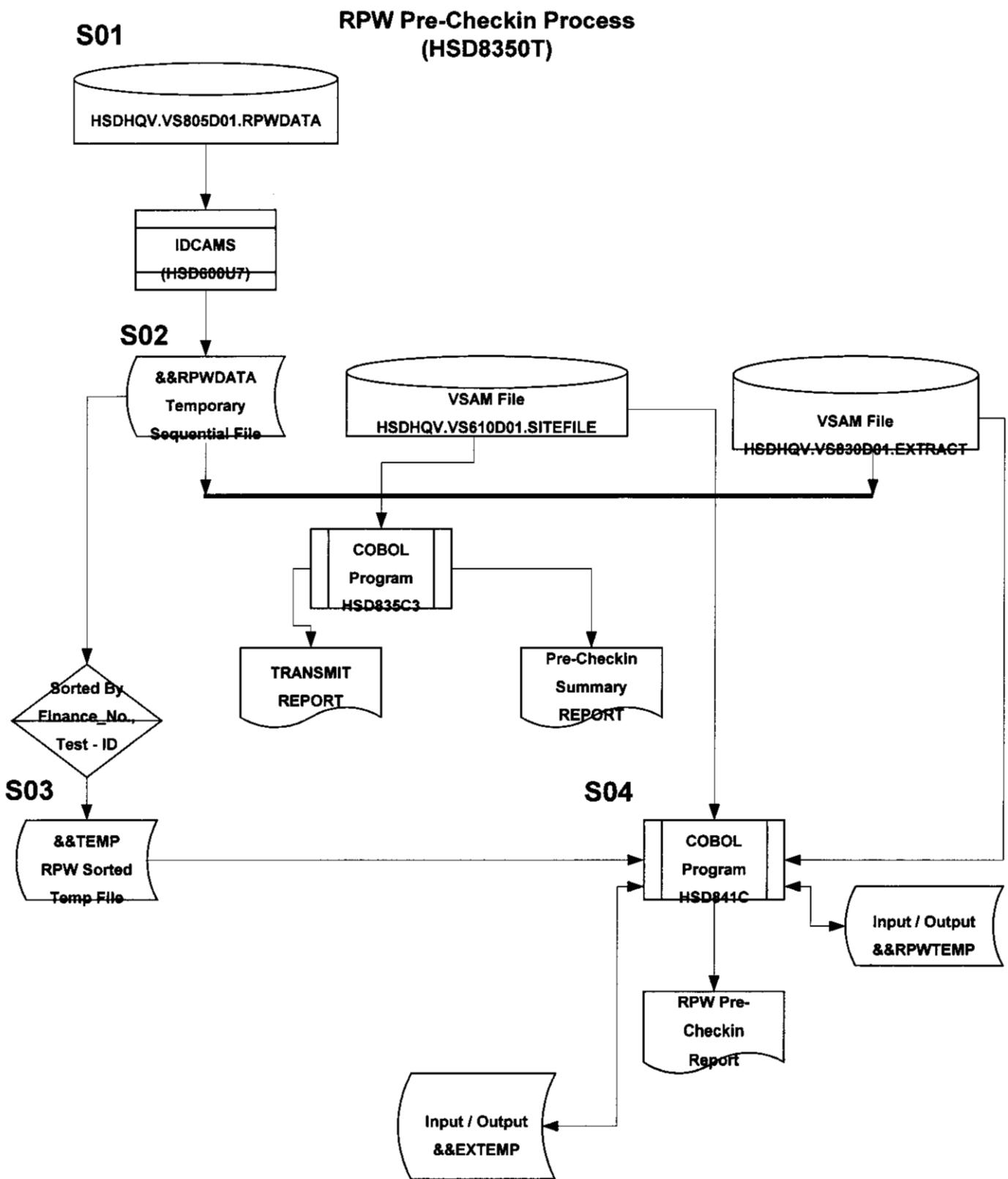


New empty RPW extract VSAM file.

**S06**

This IDCAMS program is used to create and load the new RPW VSAM Extract file

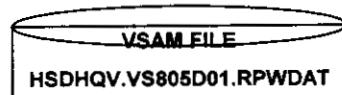




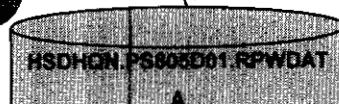
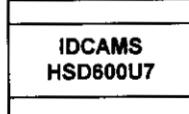
**RPW Check-in Process**

(HSD8500T)

**S01**

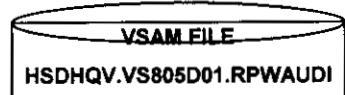


This IDCAMS program is used to move weekly RPW data to temporary dataset.

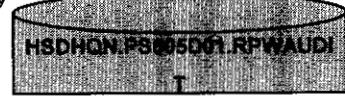
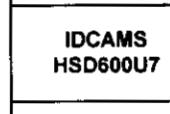


This file used in S08.

**S02**

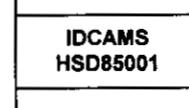
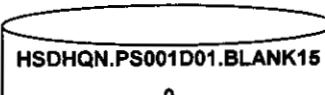
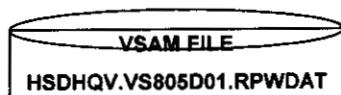


This IDCAMS program is used to move weekly RPW audit data to temporary dataset.

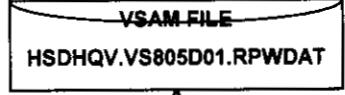
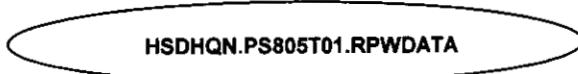


This file is used in S010.

**S04**



This IDCAMS program is used to move weekly data to tape and create a new VSAM FILE HSDHQV.VS805D01.RPWDAT A

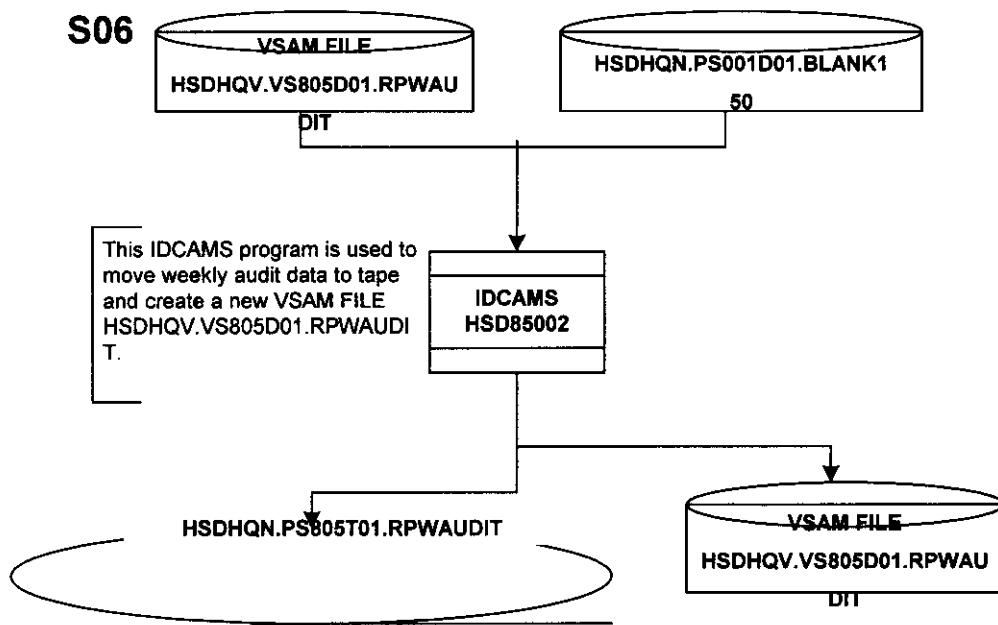


New empty RPW VSAM file.

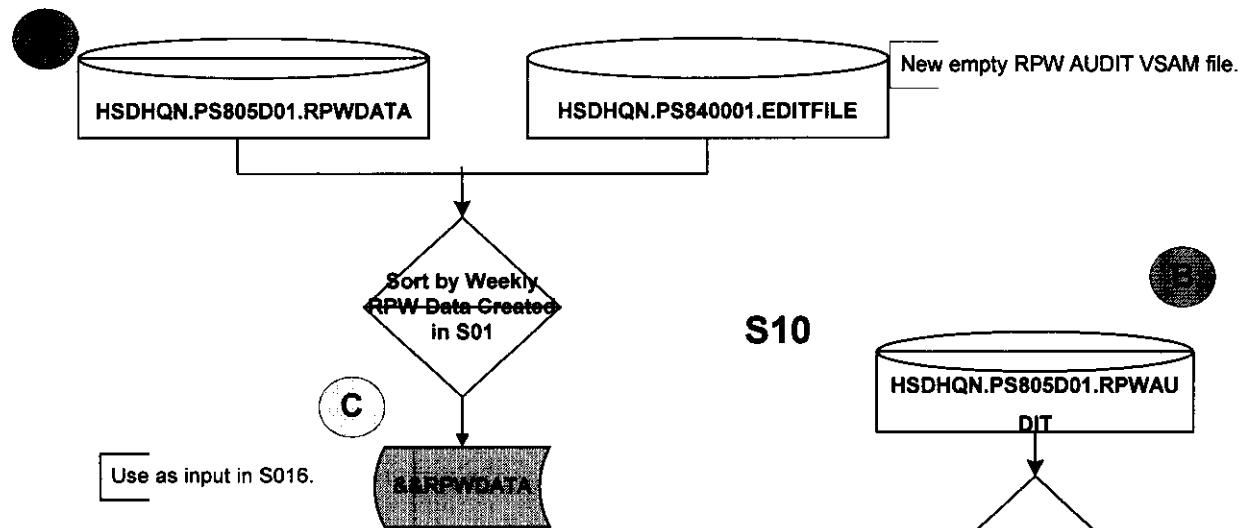
**S06**

S04

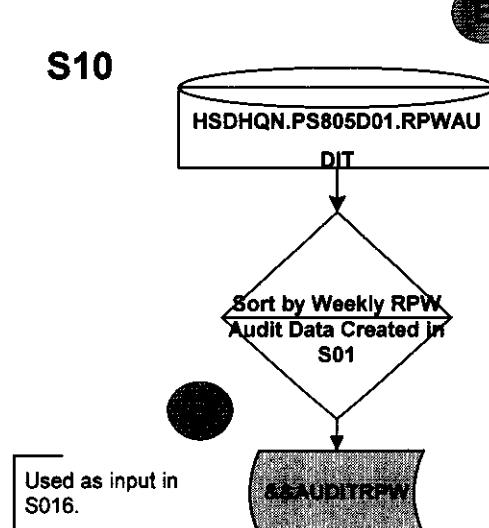
S06



S08



S10

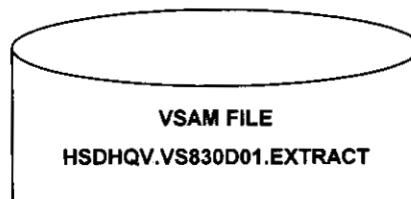


S14

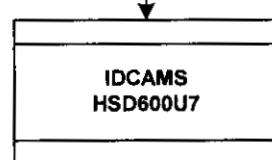
S10

S14

This VSAM file is used several times as input.

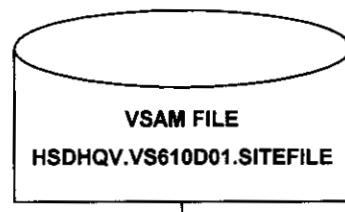
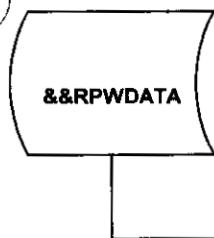


This IDCAMS program to backup is used to tape the sample extract VSAM file prior to check-in run.

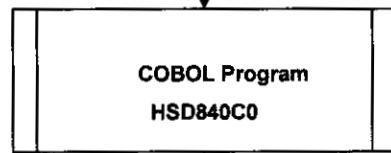


S16

C



These temporary files are connected to the input of S36

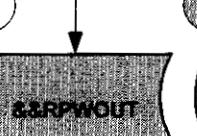
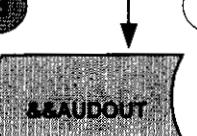


&&HSD840P1

&&HSD840P3

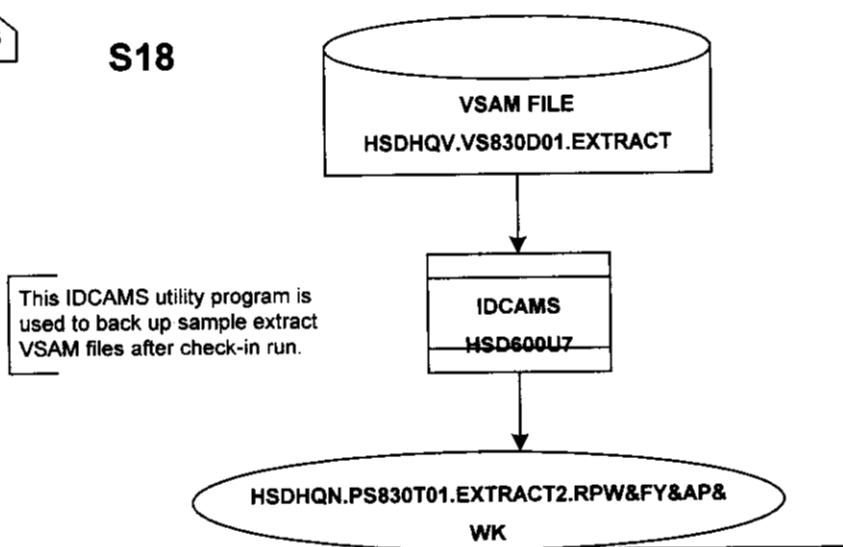
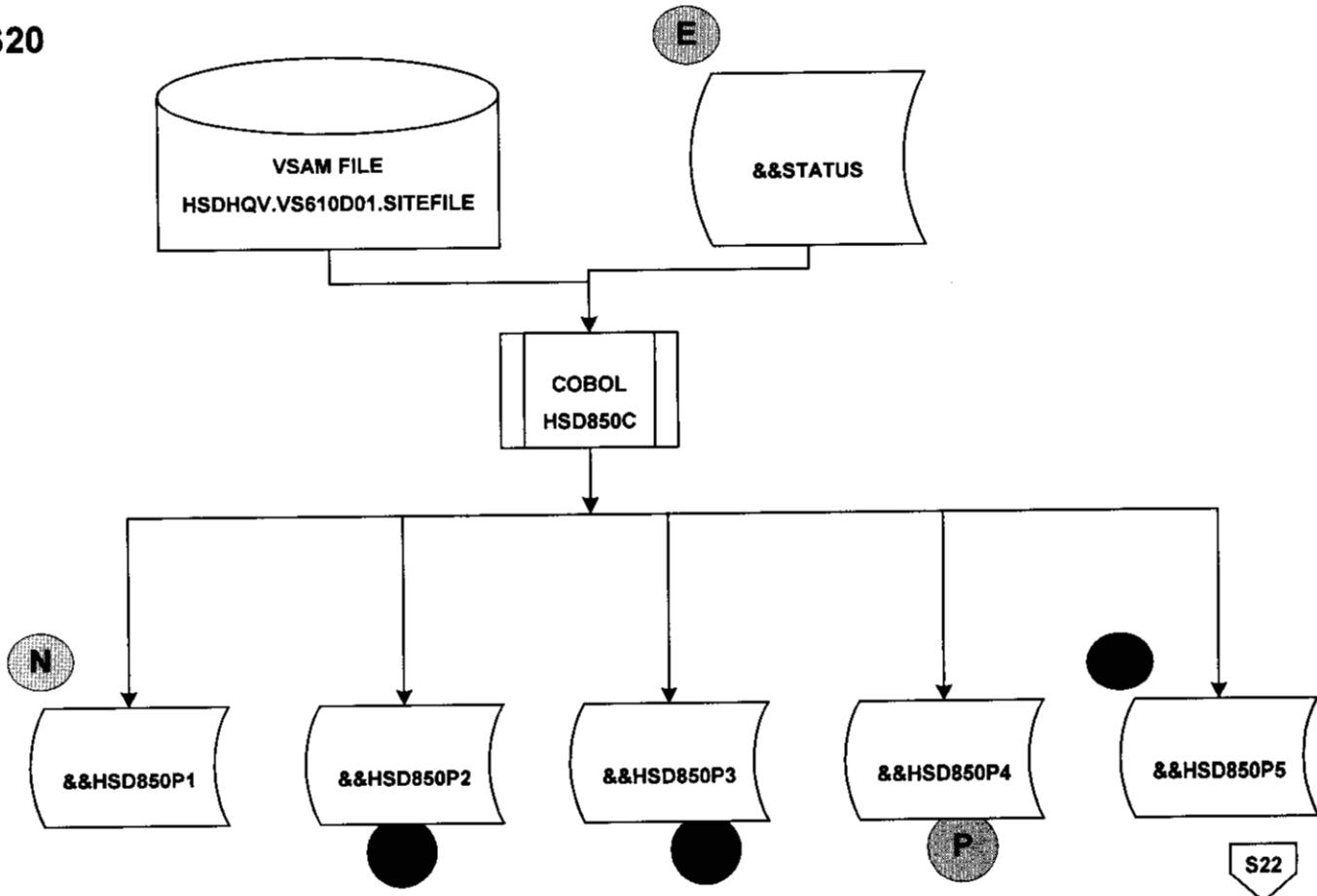
&&HSD840P2

HSDHQN.PS840D01 EDITFILE.RPW&FY&AP&WK



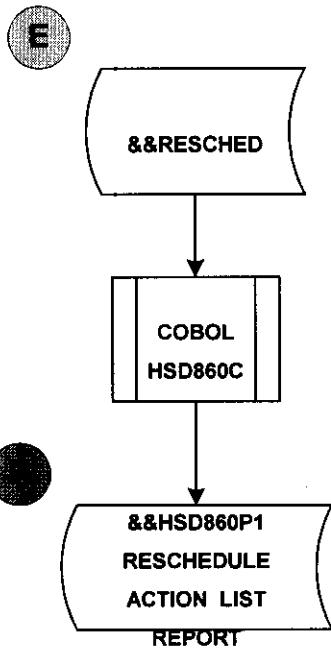
&&HSD840P1 REPORT SHOWING RESULTS

6

**S18****S20**

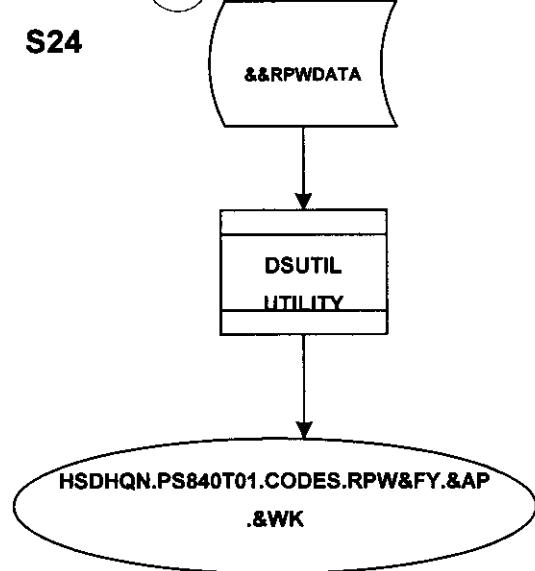
S20

S22



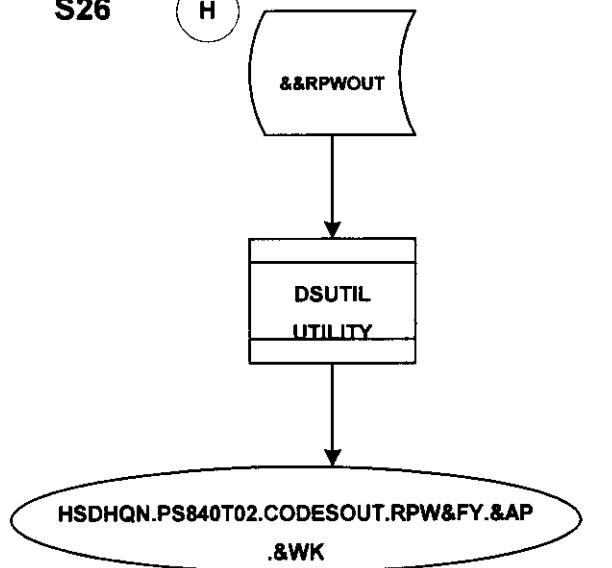
S24

C



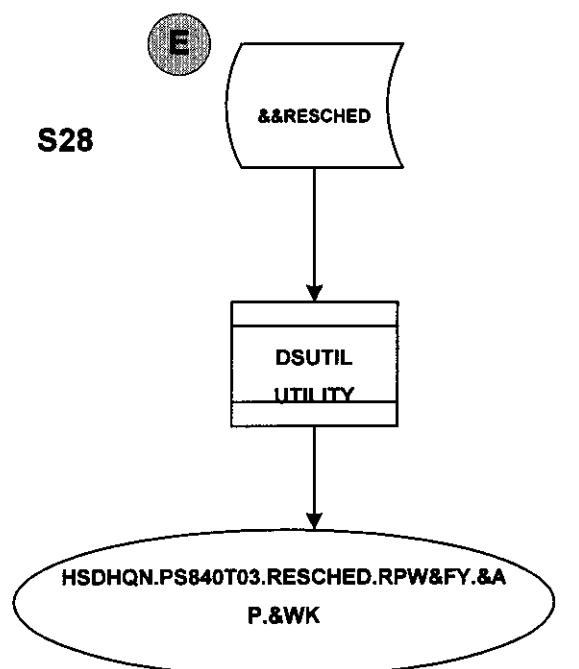
S26

H

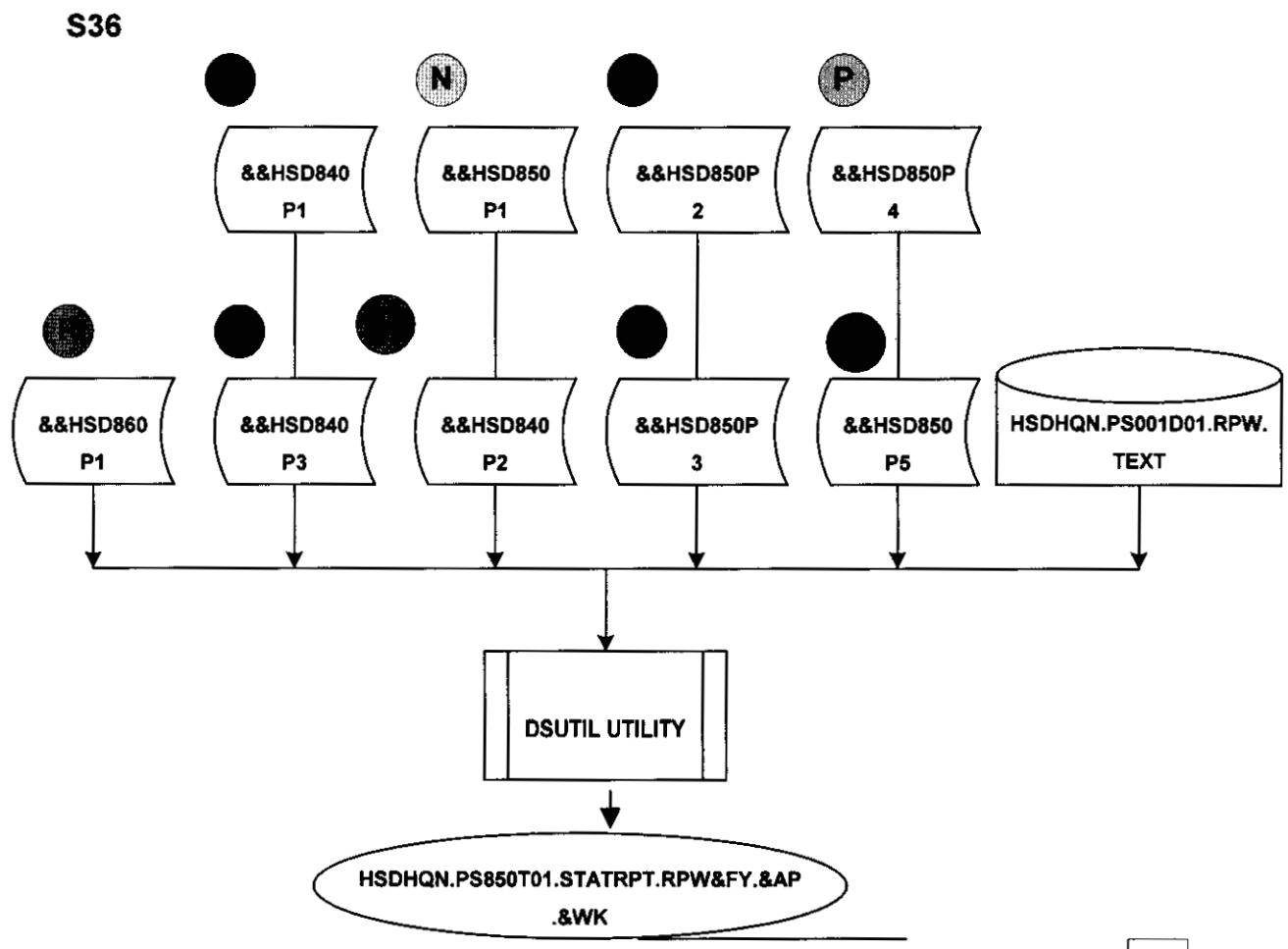
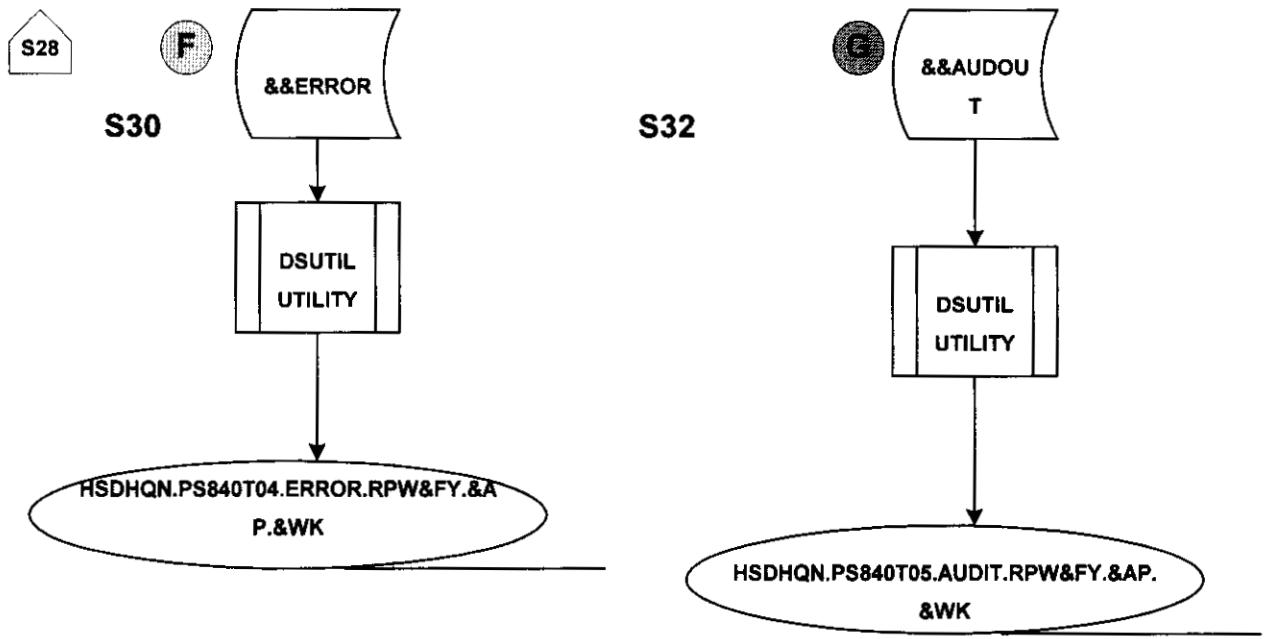


S28

E



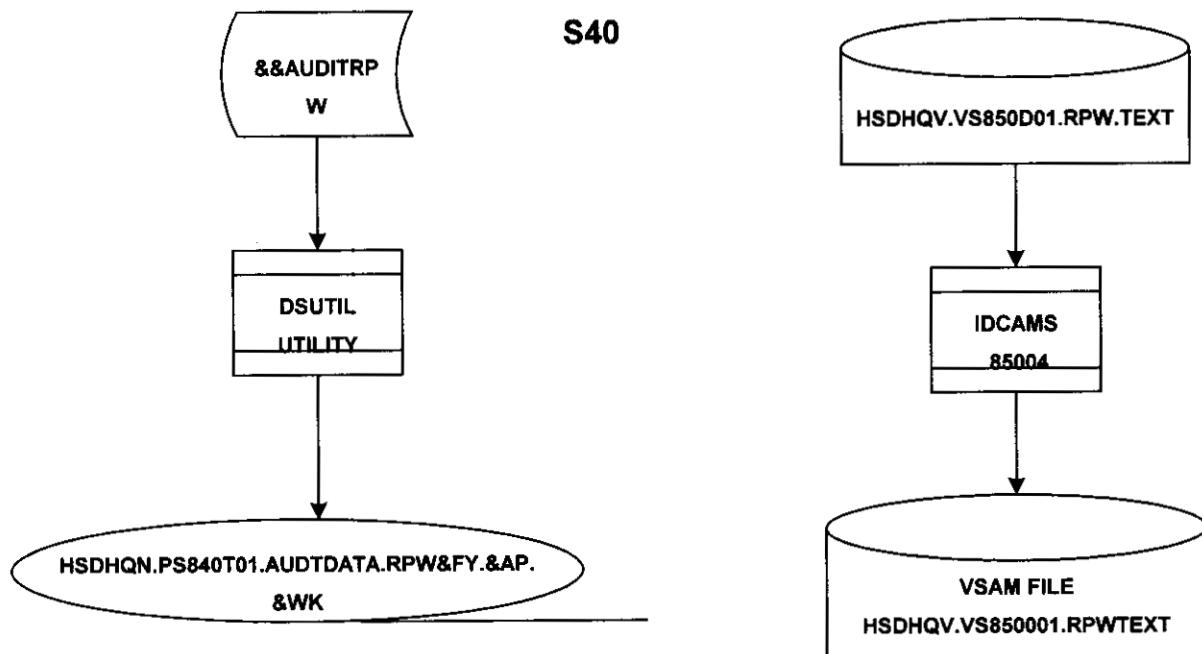
S30



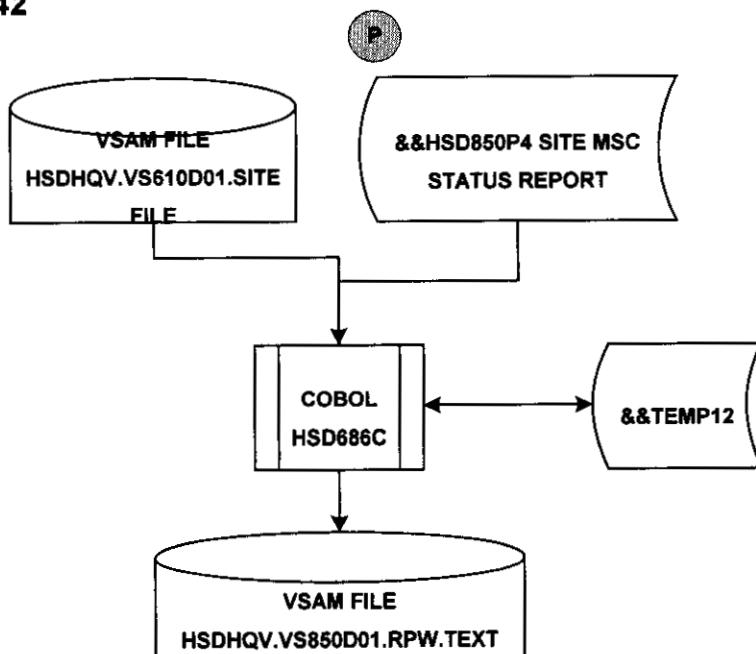
S36

S38

S40



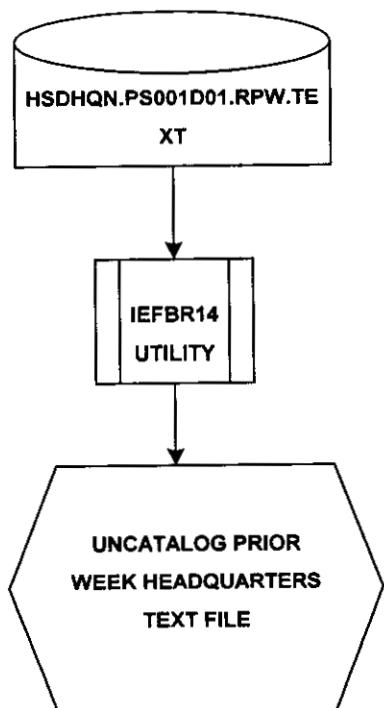
S42



S44

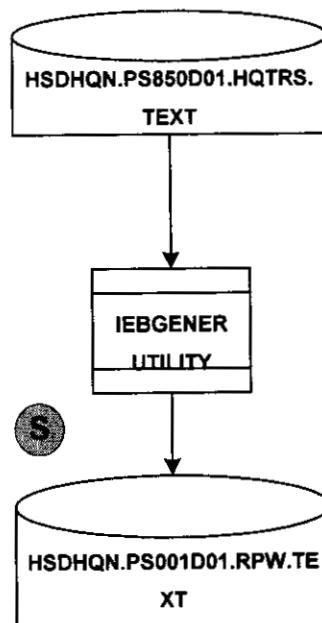
S42

S44



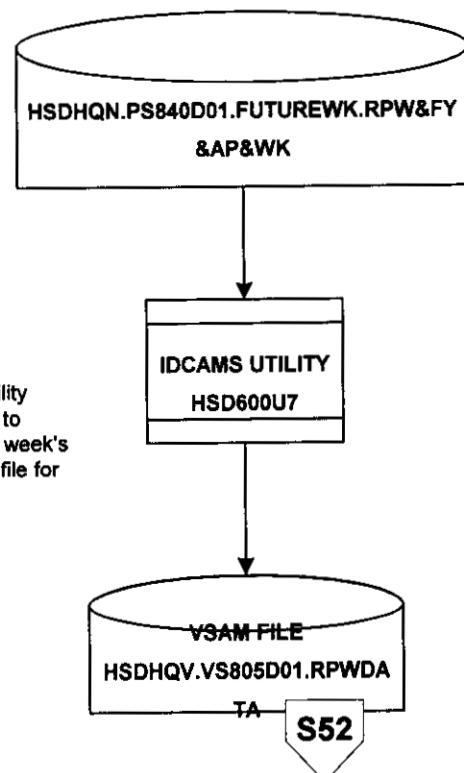
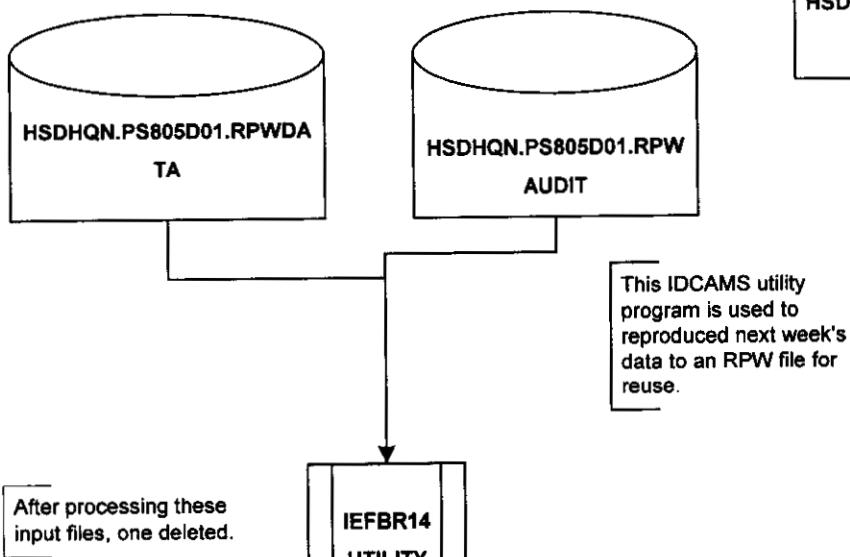
S46

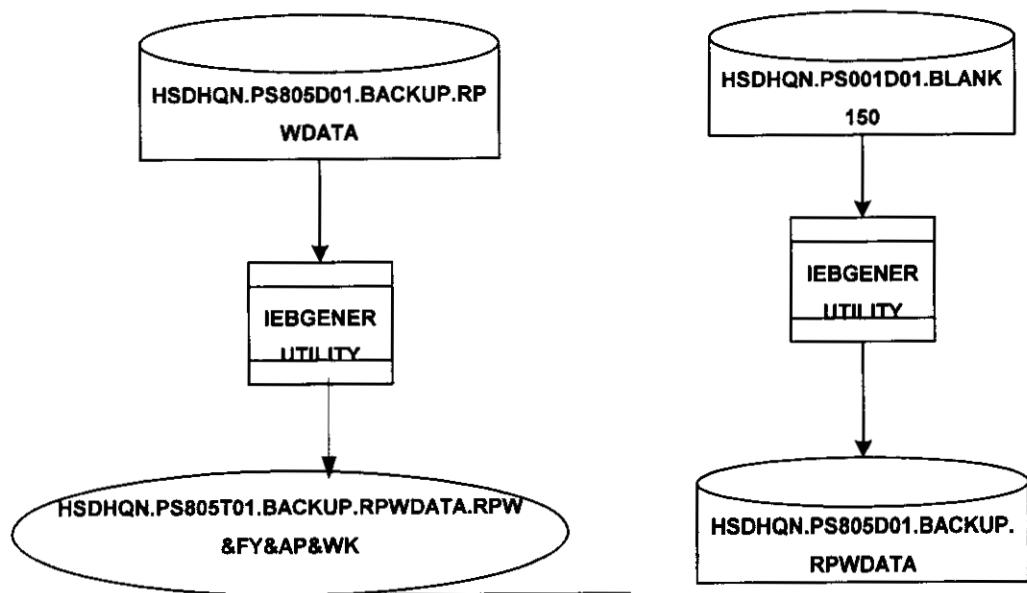
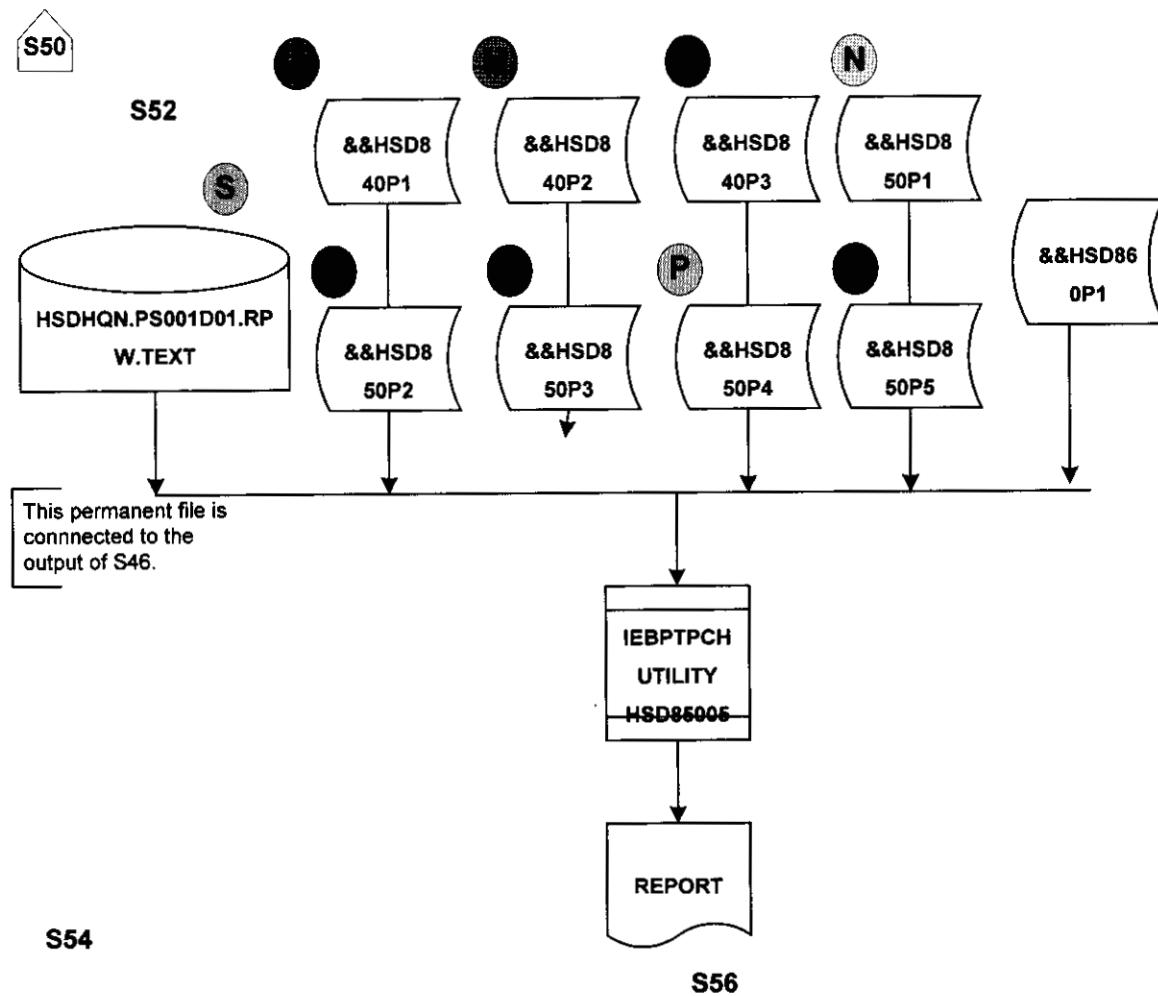
This permanent file is connected to the input of S52.



S50

S48





## JCL

```
//HSD5000Q PROC VER510=6,VER522=1,VER523=1,VER524=1,VER525=1,  
00010000  
//           VER527=1,VER528=1,VER800=2,FY=,QTR=,BQTR=,EQTR=  
00020000  
//*****  
00030000  
/*          SORT ZIP TABLE FILE               *  
00040000  
//*****  
00050000  
//S01      EXEC PGM=SORT  
00060000  
//OUT1     OUTPUT  DEST=R77,CLASS=H,COPIES=1  
00061000  
//OUT2     OUTPUT  DEST=U8058,CLASS=A,COPIES=1  
00062000  
//OUT3     OUTPUT  DEST=U281,CLASS=A,COPIES=1  
00063000  
//SYSOUT   DD    SYSOUT=*  
00070000  
//SYSPRINT DD    SYSOUT=*  
00080000  
//SYSUDUMP DD    SYSOUT=*  
00090000  
//SORTWK01 DD    UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)  
00100000  
//SORTWK02 DD    UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)  
00110000  
//SORTWK03 DD    UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)  
00120000  
//SORTIN    DD    DSN=HSFRAN.PS001D01.ZIPTABLE.ORFEO,DISP=SHR  
00130000  
//SORTOUT   DD    DSN=&&TEMP,  
00140000  
//           DISP=(NEW,PASS,DELETE),  
00150000  
//           UNIT=SYSDA,SPACE=(TRK,(5,1),RLSE),  
00160000  
//           DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)  
00170000  
//SYSIN     DD    DSN=HSDHQN.CODES.CONTROL(HSD50001),DISP=SHR  
00180000  
//*** SORT FIELDS=(1,3,A),FORMAT=CH  
00190000  
//  
00200000  
//*****  
00210000
```

```
/* MOVE SORTED ZIP INFORMATION INTO TEMPORARY FILE *  
00220000  
*****  
00230000  
//S02      EXEC PGM=HSD500C0,COND=(0,NE)  
00240000  
//SYSOUT   DD  SYSOUT=*  
00250000  
//SYSUDUMP DD  SYSOUT=*  
00260000  
//SYSPRINT DD  SYSOUT=*  
00270000  
//MSCTABLE DD  DSN=&&TEMP,DISP=(OLD,DELETE)  
00280000  
//TEMP     DD  DSN=&&TEMP1,  
00290000  
//          DISP=(NEW,PASS,DELETE),  
00300000  
//          UNIT=SYSDA,SPACE=(TRK,(5,1),RLSE),  
00310000  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)  
00320000  
/*  
00330000  
*****  
00340000  
/*      CREATE ODIS SAMPLE FILE VSAM *  
00350000  
*****  
00360000  
//S04      EXEC PGM=IDCAMS,COND=(0,NE)  
00370000  
//SYSPRINT DD  SYSOUT=*  
00380000  
//SYSUDUMP DD  SYSOUT=*  
00390000  
//SYSIN    DD  DSN=HSDHQN.CODES.CONTROL(HSD50002),DISP=SHR  
00400000  
/*  
00410000  
*****  
00420000  
/*      LOAD ODIS SAMPLE FILE VSAM WITH NEW SAMPLE DATA *  
00430000  
*****  
00440000  
//S06      EXEC PGM=HSD510C&VER510,REGION=4096K,COND=(0,NE),  
00450000  
//          PARM='&FY.&QTR.&BQTR.&EQTR'  
00460000  
//STEPLIB  DD  DSN=HSDHQN.CODES.LOAD,DISP=SHR  
00490000
```

```

//SYSOUT      DD   SYSOUT=*
00500000
//SYSPRINT    DD   SYSOUT=*
00510000
//SYSUDUMP    DD   SYSOUT=*
00520000
//HSD510P1   DD   SYSOUT=(,,,
00530000
//                  OUTPUT=(*.S01.OUT1,* .S01.OUT2,* .S01.OUT3)
00531000
//SORTWK01   DD   UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
00550000
//SORTWK02   DD   UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
00560000
//SORTWK03   DD   UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
00570000
//ZIPTABLE   DD   DSN=&&TEMP1,DISP=(OLD,PASS,DELETE)
00580000
//SAMPFILE   DD   DSN=HSF.HQ134T02.FY&FY.&QTR,DISP=OLD
00590000
//SITEFILE   DD   DSN=HSDHQV.VS610D01.SITEFILE,DISP=SHR
00610000
//FINFILE    DD   DSN=HSDHQN.PS001D01.ODIS.FINFILE,DISP=SHR
00620000
//OUTFILE    DD   DSN=HSDHQV.VS510D01.ODIS.SAMPLE,DISP=OLD
00630000
///*
00640000
//*****CREATE RPW SAMPLE FILE VSAM ****
00770000
//          CREATE RPW SAMPLE FILE VSAM *
00780000
//*****EXEC PGM=IDCAMS,COND=(0,NE) ****
00790000
//S08        EXEC PGM=IDCAMS,COND=(0,NE)
00800000
//SYSPRINT    DD   SYSOUT=*
00810000
//SYSUDUMP    DD   SYSOUT=*
00820000
//SYSIN      DD   DSN=HSDHQN.CODES.CONTROL(HSD50003),DISP=SHR
00830000
///*
00840000
//*****LOAD RPW SAMPLE FILE VSAM WITH NEW SAMPLE DATA ****
00850000
//          LOAD RPW SAMPLE FILE VSAM WITH NEW SAMPLE DATA *
00860000
//*****EXEC PGM=HSD800C&VER800,REGION=4096K,COND=(0,NE), ****
00870000
//S10        EXEC PGM=HSD800C&VER800,REGION=4096K,COND=(0,NE),
00880000

```

```

//          PARM='&FY.&QTR.&BQTR.&EQTR'
00890000
//STEPLIB DD DSN=HSDHQN.CODES.LOAD,DISP=SHR
00920000
//SYSOUT DD SYSOUT=*
00930000
//SYSPRINT DD SYSOUT=*
00940000
//SYSUDUMP DD SYSOUT=*
00950000
//HSD800P1 DD SYSOUT=(,,,
00960000
//          OUTPUT=(*.S01.OUT1,* .S01.OUT2,* .S01.OUT3)
00961000
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
00980000
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
00990000
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
01000000
//ZIPTABLE DD DSN=&&TEMP1,DISP=(OLD,PASS)
01010000
//SAMPFILE DD DSN=HSF.HQ330T02.PQ0&QTR.FY&FY,DISP=OLD
01020000
//SITEFILE DD DSN=HSDHQV.VS610D01.SITEFILE,DISP=SHR
01040000
//FINFILE DD DSN=HSDHQN.PS001D01.ODIS.FINFILE,DISP=SHR
01050000
//FINMST DD DSN=HSF.HQMAND07.SMPOFF&FY,DISP=SHR
01060000
//OUTFILE DD DSN=HSDHQV.VS800D01.RPW.SAMPLE,DISP=OLD
01070000
/*
01080000
*****+
01090000
//          PREPARE THE PDF-SITE CODE TABLE FOR THE 1ST      *
01100000
//          CLASS SERVICE STANDARD EXTRACT FILE PROCESSING.   *
01110000
*****+
01120000
//S12      EXEC PGM=HSD522C&VER522,REGION=4096K,COND=(0,NE)
01130000
//STEPLIB DD DSN=HSDHQN.CODES.LOAD,DISP=SHR
01160000
//SYSOUT DD SYSOUT=*
01170000
//SYSPRINT DD SYSOUT=*
01180000
//SYSUDUMP DD SYSOUT=*
01190000

```

```

//HSD522P1 DD SYSOUT=(, ),
01200000
// OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
01201000
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
01220000
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
01230000
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(40,40),RLSE)
01240000
//ZIPTABLE DD DSN=&&TEMP1,DISP=(OLD,PASS,DELETE)
01250000
//ODIS5031 DD DSN=HSB.PS001D01.SCFFILE,DISP=SHR
01260001
//PDFSITE DD DSN=&&PDFSITE,DISP=(NEW,PASS,DELETE),
01280000
// UNIT=SYSDA,SPACE=(CYL,(5,5),RLSE),
01290000
// DCB=(RECFM=FB,LRECL=6,BLKSIZE=600)
01300000
//*****eliminate duplicates and sort the PDF-SITE code table ****
01310000
//* ELIMINATE DUPLICATES AND SORT THE PDF-SITE CODE TABLE *
01320000
//*****eliminate duplicates and sort the PDF-SITE code table ****
01330000
//S14 EXEC PGM=SORT,COND=(0,NE)
01340000
//SYSOUT DD SYSOUT=*
01350000
//SYSPRINT DD SYSOUT=*
01360000
//SYSUDUMP DD SYSOUT=*
01370000
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
01380000
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
01390000
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
01400000
//SORTIN DD DSN=&&PDFSITE,DISP=(OLD,DELETE)
01410000
//SORTOUT DD DSN=&&PDFSITSR,DISP=(NEW,PASS,DELETE),
01420000
// UNIT=SYSDA,SPACE=(CYL,(5,5),RLSE),
01430000
// DCB=(RECFM=FB,LRECL=6,BLKSIZE=600)
01440000
//SYSIN DD DSN=HSDHQN.CODES.CONTROL(HSD50009),DISP=SHR
01450000
//*****eliminate duplicates and sort the PDF-SITE code table ****
01460000

```

```

/**SORT FIELDS=(1,3,A,           /* SORT BY:          PDF
01470000
/**             4,3,A),      /*                      SITE
01480000
/**             FORMAT=CH
01490000
/**SUM  FIELDS=NONE            /* ELIMINATE DUPLICATE RECORDS
01500000
*****                                                 *
01510000
/***
01520000
*****                                                 *
01530000
/*   EXTRACT SERVICE STANDARD DATA - 1ST CLASS      *
01540000
*****                                                 *
01550000
//S16      EXEC  PGM=HSD523C&VER523,REGION=4096K,COND=(0,NE)
01560000
//STEPLIB  DD   DSN=HSDHQN.CODES.LOAD,DISP=SHR
01590000
//SYSOUT   DD   SYSOUT=*
01600000
//SYSPRINT DD   SYSOUT=*
01610000
//SYSUDUMP DD   SYSOUT=*
01620000
//HSD523P1 DD   SYSOUT=(,,,
01630000
//                  OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
01631000
//SSTDFILE DD   DSN=HSBRAN.PS506T01.ODIS5061,DISP=OLD
01650000
//PDFSITE  DD   DSN=&&PDFSITSR,DISP=(OLD,DELETE)
01660000
//OUTFILE  DD   DSN=&&STDXTRCT,DISP=(NEW,PASS,DELETE),
01670000
//                  UNIT=SYSDA,SPACE=(CYL,(5,1),RLSE),
01680000
//                  DCB=(RECFM=FB,LRECL=11,BLKSIZE=6160)
01690000
/*
01700000
*****                                                 *
01710000
/*   EXTRACT SERVICE STANDARD DATA - PRIORITY      *
01720000
*****                                                 *
01730000
//S18      EXEC  PGM=HSD524C&VER524,REGION=4096K,COND=(0,NE)
01740000

```

```

//STEPLIB DD DSN=HSDHQN.CODES.LOAD,DISP=SHR
01770000
//SYSOUT DD SYSOUT=*
01780000
//SYSPRINT DD SYSOUT=*
01790000
//SYSUDUMP DD SYSOUT=*
01800000
//HSD524P1 DD SYSOUT=(,,,
01810000
//          OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
01811000
//SSTDFILE DD DSN=HSBRAN.PS506T01.ODIS5063,DISP=OLD
01830000
//ZIPTABLE DD DSN=&&TEMP1,DISP=(OLD,DELETE)
01840000
//OUTFILE DD DSN=&&STDXTRCT,DISP=(OLD,PASS)
01850000
///*
01860000
//*****
01870000
///* SORT THE COMBINED SERVICE STANDARD EXTRACT FILE
01880000
//*****
01890000
//S20      EXEC PGM=SORT,COND=(0,NE)
01900000
//SYSOUT DD SYSOUT=*
01910000
//SYSPRINT DD SYSOUT=*
01920000
//SYSUDUMP DD SYSOUT=*
01930000
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
01940000
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
01950000
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
01960000
//SORTIN DD DSN=&&STDXTRCT,DISP=(OLD,DELETE)
01970000
//SORTOUT DD DSN=&&SSTDTSORT,DISP=(NEW,PASS,DELETE),
01980000
//          UNIT=SYSDA,SPACE=(CYL,(5,1),RLSE),
01990000
//          DCB=(RECFM=FB,LRECL=11,BLKSIZE=6160)
02000000
//SYSIN DD DSN=HSDHQN.CODES.CONTROL(HSD50004),DISP=SHR
02010000
//*****
02020000

```

```
/*SORT FIELDS=(1,3,A,          /* SITE CODE
02030000
/*           4,1,A,          /* CLASS
02040000
/*           5,3,A,          /* DESTINATION SECT CENTER FAC (SCF)
02050000
/*           8,3,A),        /* ORIGIN      SECT CENTER FAC (SCF)
02060000
/*           FORMAT=CH
02070000
/*
02080000
*****CREATE SERVSTD FILE VSAM
02090000
/*           EXEC PGM=IDCAMS,COND=(0,NE)
02100000
*****EXEC PGM=HSD525C&VER525,REGION=4096K,COND=(0,NE)
02110000
//S22      DSN=HSDHQN.CODES.CONTROL(HSD50005),DISP=SHR
02120000
//SYSPRINT DD  SYSOUT=*
02130000
//SYSUDUMP DD  SYSOUT=*
02140000
//SYSIN     DD  DSN=HSDHQN.CODES.CONTROL(HSD50005),DISP=SHR
02150000
/*
02160000
*****LOAD SERVSTD EXTRCT VSAM FILE W/ NEW SERVSTD DATA
02170000
/*           EXEC PGM=HSD525C&VER525,REGION=4096K,COND=(0,NE)
02180000
*****EXEC PGM=HSD525P1,REGION=4096K,COND=(0,NE)
02190000
//S24      DSN=HSDHQN.CODES.LOAD,DISP=SHR
02200000
//STEPLIB  DD  DSN=HSDHQN.CODES.LOAD,DISP=SHR
02210000
//SYSOUT    DD  SYSOUT=*
02220000
//SYSPRINT DD  SYSOUT=*
02230000
//SYSUDUMP DD  SYSOUT=*
02240000
//HSD525P1 DD  SYSOUT=(,,),
02250000
//OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
02260000
//SSTDSORT DD  DSN=&&SSTDSORT,DISP=(OLD,DELETE)
02270000
//OUTFILE   DD  DSN=HSDHQV.VS525D01.SERVSTD,DISP=OLD
02280000
```

```

/*
02310000
/***** ****
02320000
/* CREATE SEQUENTIAL ODIS PARTITIONED FILE FROM THE VSAM FILE      *
02330000
/***** ****
02340000
//S26      EXEC PGM=IDCAMS,COND=(0,NE)
02350000
//SYSPRINT DD  SYSOUT=*
02360000
//SYSUDUMP DD  SYSOUT=*
02370000
//SYSUT1   DD  DSN=HSDHQV.VS510D01.ODIS.SAMPLE,DISP=OLD
02380000
//SYSUT2   DD  DSN=&&ODISAMPL,
02390000
//          DISP=(NEW,PASS),
02400000
//          SPACE=(CYL,(80,5),RLSE),
02410000
//          DCB=(RECFM=FB,LRECL=300,BLKSIZE=6000)
02420000
//SYSIN    DD  DSN=HSDHQN.CODES.CONTROL(HSD600U7),DISP=SHR
02430000
/*
02440000
/***** ****
02450000
/* EXTRACT FIELDS, ELIMINATE DUPLICATES AND SORT THE      *
02460000
/* ODIS SAMPLE FILE      *
02470000
/***** ****
02480000
//S28      EXEC PGM=SORT,COND=(0,NE)
02490000
//SYSOUT   DD  SYSOUT=*
02500000
//SYSPRINT DD  SYSOUT=*
02510000
//SYSUDUMP DD  SYSOUT=*
02520000
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
02530000
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
02540000
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
02550000
//SORTIN    DD  DSN=&&ODISAMPL,DISP=(OLD,DELETE)
02560000

```

```

//SORTOUT DD DSN=&&SMPCSTRT,DISP=(NEW,PASS),
02570000
// UNIT=SYSDA,SPACE=(CYL,(5,1),RLSE),
02580000
// DCB=(RECFM=FB,LRECL=9,BLKSIZE=6156)
02590000
//SYSIN DD DSN=HSDHQN.CODES.CONTROL(HSD50006),DISP=SHR
02600000
*****+
02610000
//** OMIT COND=(10,9,CH,EQ,C'99999999') /* ELIMINATE TRAILER RECS
02620000
//**INREC FIELDS=(176,3,          /* EXTRACT INPUT FIELDS: SAMPLE CELL
02630000
/**           57,3,          /* STRATA CODE
02640000
/**           1,3)          /* SITE
02650000
//**SORT FIELDS=(1,3,A,          /* SORT BY:             SAMPLE CELL
02660000
/**           4,3,A,          /* STRATA CODE
02670000
/**           7,3,A),         /* SITE
02680000
/**           FORMAT=CH
02690000
//**SUM FIELDS=NONE            /* ELIMINATE DUPLICATE RECORDS
02700000
*****+
02710000
/*
02720000
*****+
02730000
/**   MATCH ODIS SAMPLE EXTRACT AND VARIABLES FILES *
02740000
*****+
02750000
//S30      EXEC PGM=HSD527C&VER527,REGION=4096K,COND=(0,NE)
02760000
//STEPLIB DD DSN=HSDHQN.CODES.LOAD,DISP=SHR
02790000
//SYSOUT DD SYSOUT=*
02800000
//SYSPRINT DD SYSOUT=*
02810000
//SYSUDUMP DD SYSOUT=*
02820000
//HSD527P1 DD SYSOUT=(,,,
02830000
//           OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
02831000

```

```

//SMPCSTRT DD DSN=&&SMPCSTRT,DISP=(OLD,DELETE)
02850000
//VARIABLE DD DSN=HSF.HQ033D01.FY&FY.&QTR,DISP=OLD
02860000
//OUTFILE DD DSN=&&VRBLRAW,DISP=(NEW,PASS),
02870000
// UNIT=SYSDA,SPACE=(CYL,(5,1),RLSE),
02880000
// DCB=(RECFM=FB,LRECL=28,BLKSIZE=6160)
02890000
/*
02900000
***** 
02910000
/* SORT THE VARIABLES FILE ON THE SITE, SAMPLE CELL AND STRATA CODE*
02920000
***** 
02930000
//S32 EXEC PGM=SORT,COND=(0,NE)
02940000
//SYSOUT DD SYSOUT=*
02950000
//SYSPRINT DD SYSOUT=*
02960000
//SYSUDUMP DD SYSOUT=*
02970000
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
02980000
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
02990000
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
03000000
//SORTIN DD DSN=&&VRBLRAW,DISP=(OLD,DELETE)
03010000
//SORTOUT DD DSN=&&VRBLSRT,DISP=(NEW,PASS),
03020000
// UNIT=SYSDA,SPACE=(CYL,(5,1),RLSE),
03030000
// DCB=(RECFM=FB,LRECL=28,BLKSIZE=6160)
03040000
//SYSIN DD DSN=HSDHQN.CODES.CONTROL(HSD50007),DISP=SHR
03050000
***** 
03060000
/*SORT FIELDS=(1,3,A,          /* SORT BY:           SITE
03070000
/*          4,3,A,          /*                         SAMPLE CELL
03080000
/*          7,3,A),          /*                         STRATA CODE
03090000
/*          FORMAT=CH
03100000

```

```

//*****03110000*****
03110000
//*
03120000
//*****03130000*****
03130000
//**          CREATE VARIABLE FILE VSAM      *
03140000
//*****03150000*****
03150000
//S34      EXEC PGM=IDCAMS,COND=(0,NE)
03160000
//SYSPRINT DD  SYSOUT=*
03170000
//SYSUDUMP DD  SYSOUT=*
03180000
//SYSIN     DD  DSN=HSDHQN.CODES.CONTROL(HSD50008),DISP=SHR
03190000
//*
03200000
//*****03210000*****
03210000
//**          LOAD VARIABLE EXTRCT VSAM FILE W/ NEW VARIABLE DATA      *
03220000
//*****03230000*****
03230000
//S36      EXEC PGM=HSD528C&VER528,REGION=4096K,COND=(0,NE)
03240000
//STEPLIB   DD  DSN=HSDHQN.CODES.LOAD,DISP=SHR
03270000
//SYSOUT    DD  SYSOUT=*
03280000
//SYSPRINT DD  SYSOUT=*
03290000
//SYSUDUMP DD  SYSOUT=*
03300000
//HSD528P1 DD  SYSOUT=(,,,
03310000
//          OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
03311000
//VRBLSRT   DD  DSN=&&VRBLSRT,DISP=(OLD,DELETE)
03330000
//SITEFILE  DD  DSN=HSDHQV.VS610D01.SITEFILE,DISP=SHR
03340000
//OUTFILE   DD  DSN=HSDHQV.VS528D01.VARIABLE,DISP=OLD
03350000
//*
03360000
//*****03370000*****
03370000
//**          CLEAN UP PREVIOUS COPY OF THE 5031 ZIP CODE FILE      *
03380000

```

```
*****  
03390000  
//S38      EXEC  PGM=IEFBR14,COND=(0,NE)  
03400000  
//FILE     DD   DSN=HSDHQN.PS001D01.SCFFILE,UNIT=SYSDA,  
03410000  
//          DISP=(MOD,DELETE,DELETE),SPACE=(CYL,(1),RLSE)  
03420000  
/*  
03430000  
*****  
03440000  
/*   COPY THE 5031 ZIP CODE FILE TO ONE THAT IS ACCESSABLE TO      *  
03450000  
/*   TO THE FIELD      *  
03460000  
*****  
03470000  
//S40      EXEC  PGM=IEBGENER  
03480000  
//SYSPRINT DD SYSOUT=*  
03490000  
//SYSUT1   DD DSN=HSB.PS001D01.SCFFILE,DISP=SHR  
03500000  
//SYSUT2   DD DSN=HSDHQN.PS001D01.SCFFILE,VOL=SER=RAXZ04,  
03510000  
//          DISP=(NEW,CATLG,DELETE),UNIT=SYSDA,  
03520000  
//          SPACE=(CYL,(5,5),RLSE),  
03530000  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)  
03540000  
//SYSIN DD DUMMY  
03550000  
/*  
03560000
```

```
//HSD8300Q  PROC VERS=0
00080099
//*****
00081093
/*  PROC FOR CREATING THE RPW SAMPLE SELECTION EXTRACT FILE.      *
00082093
//*****
00084093
/*
00085093
//*****
00090067
/*  UPDATE THE CODES ZIP TABLE FILE FROM THE MEPS MASTER ZIP      *
00100074
/*  FILE.  BEGIN BY SORTING THE MASTER FILE ON THREE-DIGIT ZIP.   *
00101067
//*****
00102067
//S01      EXEC PGM=SORT
00103099
//OUT1      OUTPUT  DEST=U5704,CLASS=H,COPIES=1
00103199
//OUT2      OUTPUT  DEST=U8058,CLASS=A,COPIES=1
00103299
//OUT3      OUTPUT  DEST=U281,CLASS=A,COPIES=1
00103399
//SYSOUT    DD  SYSOUT=*
00104067
//SYSPRINT DD  SYSOUT=*
00105067
//SYSUDUMP DD  SYSOUT=*
00106067
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
00106167
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
00106267
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
00106367
//SORTIN    DD  DSN=HSFRAN.PS001D01.ZIPTABLE.ORFEO,DISP=SHR
00106467
//SORTOUT   DD  DSN=&&TEMP,
00106567
//          DISP=(NEW,PASS,DELETE),
00106667
//          UNIT=SYSDA,SPACE=(TRK,(5,1),RLSE),
00106767
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
00106867
//SYSIN    DD  DSN=HSDHQN.CODES.CONTROL(HSD83001),DISP=SHR
00106999
/*
00107067
```

```

*****00107167
/* MOVE SORTED ZIP INFORMATION INTO TEMPORARY FILE.
00107267
*****00107367
//S02      EXEC PGM=HSD500C0,COND=(0,NE)
00107499
//SYSOUT   DD  SYSOUT=*
00107567
//SYSUDUMP DD  SYSOUT=*
00107667
//SYSPRINT DD  SYSOUT=*
00107767
//MSCTABLE DD  DSN=&&TEMP,DISP=(OLD,DELETE)
00107867
//TEMP      DD  DSN=&&TEMP1,
00107967
//          DISP=(NEW,PASS,DELETE),
00108067
//          UNIT=SYSDA,SPACE=(TRK,(5,1),RLSE),
00108167
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
00108267
//**
00108367
*****00108467
/*      SORT HSF.HQMAND07.SMPOFF&FY FILE BY FINMSTR-FIN      *
00110599
*****00110699
//S03      EXEC PGM=SORT,COND=(0,NE)
00110799
//SYSOUT   DD  SYSOUT=*
00110899
//SYSPRINT DD  SYSOUT=*
00110999
//SYSDUMP  DD  SYSOUT=*
00111099
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(30,5),RLSE)
00111199
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(30,5),RLSE)
00111299
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(30,5),RLSE)
00111399
//SORTIN    DD  DSN=HSF.HQMAND07.SMPOFF&FY..PQ&QTR,DISP=SHR
00111599
//SORTOUT   DD  DSN=&&HSFSORT,
00111699
//          DISP=(NEW,PASS,DELETE),
00111799

```

```
//                      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6320),  
00111899  
//                      UNIT=SYSDA,SPACE=(TRK,(90,50),RLSE)  
00111999  
//SYSIN    DD DSN=HSDHQN.CODES.CONTROL(HSD83002),DISP=SHR  
00112099  
/*  
00112199  
*****  
00112293  
/* PROCESS AND SORT SAMPLE RECORDS FOR THE SELECTED QUARTER,      *  
00112393  
/* OUTPUTTING A TEMPORARY EXTRACT FILE.                          *  
00113093  
*****  
00120093  
//S04      EXEC PGM=HSD830C&VERS,REGION=4096K,TIME=10,COND=(0,NE),  
00130099  
//                      PARM='&FY&QTR&BQTR&EQTR'  
00140092  
//STEPLIB   DD DSN=HSDHQN.CODES.LOAD,DISP=SHR  
00150093  
//SYSUDUMP   DD SYSOUT=*  
00160019  
//SYSPRINT   DD SYSOUT=*  
00170020  
//SYSOUT     DD SYSOUT=*  
00180021  
//ZIPTABLE   DD DSN=&&TEMP1,DISP=(OLD,DELETE)  
00191099  
//SAMPLE     DD DSN=HSF.HQ330T02.PQ0&QTR.FY&FY,DISP=SHR  
00200099  
//FINMAST    DD DSN=&&HSFSORT,DISP=(OLD,DELETE)  
00210099  
//FINEXCP    DD DSN=HSDHQN.PS001D01.ODIS.FINFILE,DISP=SHR  
00220076  
//SORTWK01   DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)  
00230084  
//SORTWK02   DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)  
00231084  
//SORTWK03   DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)  
00232084  
//EXTRACT    DD DSN=&&EXTRACT,  
00241087  
//                      DISP=(NEW,PASS,DELETE),  
00242087  
//                      UNIT=SYSDA,SPACE=(TRK,(50,5),RLSE),  
00243089  
//                      DCB=(RECFM=FB,LRECL=132,BLKSIZE=23364)  
00244087  
//EXCPFILE   DD SYSOUT=(,),  
00250095
```

```
//          OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
00250199
//CNTLFILE DD SYSOUT=(,),
00260095
//          OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
00260199
/*
00261087
/* ****
00270092
/*  DELETE OLD RPW VSAM EXTRACT FILE, AND DEFINE NEW RPW VSAM
00280099
/*  EXTRACT FILE.
00281099
/* ****
00290092
//S05      EXEC PGM=IDCAMS,COND=(0,NE)
00300099
//SYSPRINT DD SYSOUT=*
00310087
//SYSDUMP  DD SYSOUT=*
00320087
//SYSIN    DD DSN=HSDHQN.CODES.CONTROL(HSD83003),DISP=SHR
00330099
/*
00340087
/* ****
00350087
/*  LOAD NEW RPW VSAM EXTRACT FILE. *
00360099
/* ****
00370087
//S06      EXEC PGM=IDCAMS,COND=(0,NE)
00380099
//SYSOUT   DD  SYSOUT=*
00390087
//SYSDUMP  DD  SYSOUT=*
00400087
//SYSPRINT DD  SYSOUT=*
00410087
//SYSUT1   DD  DSN=&&EXTRACT,DISP=(OLD,DELETE)
00420087
//SYSUT2   DD  DSN=HSDHQV.VS830D01.EXTRACT,DISP=OLD
00430093
//SYSIN    DD  DSN=HSDHQN.CODES.CONTROL(HSD600U7),DISP=SHR
00440093
/*
00450087
```

```

//HSD8350T  PROC  VER835=0,VER841=0
00010009
//*****
00011000
/*  PROC FOR RPW PRE-CHECK-IN. *
00012000
//*****
00013000
/*
00014000
//*****
00015000
/*  REPRO RPW MASTER DATA FILE TO A TEMPORARY SEQUENTIAL FILE.  *
00016000
//*****
00017000
//S01      EXEC PGM=IDCAMS
00018002
//OUT1      OUTPUT  DEST=U5704,CLASS=H,COPIES=1
00018103
//OUT2      OUTPUT  DEST=U8058,CLASS=H,COPIES=1
00018201
//OUT3      OUTPUT  DEST=U281,CLASS=A,COPIES=1
00018301
//SYSPRINT DD  SYSOUT=*
00019000
//SYSUDUMP DD  SYSOUT=*
00020000
//SYSUT1    DD  DSN=HSDHQV.VS805D01.RPWDATA,DISP=SHR
00030000
//SYSUT2    DD  DSN=&&PWDATA,
00031000
//          DISP=(NEW,PASS),
00032000
//          UNIT=SYSDA,SPACE=(CYL,(80,20),RLSE),
00033008
//          DCB=(RECFM=FB,LRECL=150,BLKSIZE=23400)
00034000
//SYSIN     DD  DSN=HSDHQN.CODES.CONTROL(HSD600U7),DISP=SHR
00035000
/*
00036000
//*****
00037000
/*  RUN PRE-CHECK-IN PROGRAM. *
00038000
//*****
00039000
//S02      EXEC PGM=HSD835C&VER835,TIME=10,
00040002
//          PARM='&BEEK.&EEEK'
00041000

```

```
//STEPLIB DD DSN=HSDHQN.CODES.LOAD,DISP=SHR
00042000
//SYSOUT DD SYSOUT=*
00043000
//SYSPRINT DD SYSOUT=*
00044000
//SYSDUMP DD SYSOUT=*
00045000
//MASTER DD DSN=&&RPWDATA,DISP=(OLD,PASS)
00046007
//SITEFILE DD DSN=HSDHQV.VS610D01.SITEFILE,DISP=SHR
00047000
//EXTRACT DD DSN=HSDHQV.VS830D01.EXTRACT,DISP=SHR
00048000
//REPORT1 DD SYSOUT=(,,,
00049101
//          OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
00049201
//REPORT2 DD SYSOUT=(,,,
00051001
//          OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
00052001
/*
00060000
//S03      EXEC PGM=SORT
00420006
//SYSOUT DD SYSOUT=*
00430004
//SYSPRINT DD SYSOUT=*
00440004
//SYSUDUMP DD SYSOUT=*
00450004
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(80,5),RLSE)
00460004
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(80,5),RLSE)
00470004
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(80,5),RLSE)
00480004
//SORTIN DD DSN=&&RPWDATA,DISP=(OLD,DELETE)
00490004
//SORTOUT DD DSN=&&TEMP,
00500004
//          DISP=(NEW,PASS,DELETE),
00510004
//          UNIT=SYSDA,SPACE=(CYL,(80,10),RLSE),
00520004
//          DCB=(RECFM=FB,LRECL=150,BLKSIZE=0)
00530004
//SYSIN DD DSN=HSDHQN.CODES.CONTROL(HSD83501),DISP=SHR
00540004
//**** SORT FIELDS=(1,12,A),FORMAT=CH
00550004
```

```
/*
00560004
//S04      EXEC PGM=HSD841C&VER841,TIME=10,PARM='&BEEK.&EEEK'
00570006
//STEPLIB  DD   DSN=HSDHQN.CODES.LOAD,DISP=SHR
00580004
//SYSOUT    DD   SYSOUT=*
00590004
//SYSUDUMP  DD   SYSOUT=*
00600004
//SYSPRINT  DD   SYSOUT=*
00610004
//HSD841P1  DD   SYSOUT=(,,,
00620004
//          OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
00622004
//RPWDATA   DD   DSN=&&TEMP,DISP=(OLD,DELETE)
00623007
//EXTRACT   DD   DSN=HSDHQV.VS830D01.EXTRACT,DISP=OLD
00630004
//EXTEMP    DD   DSN=&&EXTEMP,
00640004
//          DISP=(NEW,DELETE),
00650004
//          UNIT=SYSDA,SPACE=(TRK,(30,10),RLSE),
00660004
//          DCB=(RECFM=FB,LRECL=23,BLKSIZE=0)
00670004
//RPWTEMP   DD   DSN=&&RPWTEMP,
00680004
//          DISP=(NEW,DELETE),
00690004
//          UNIT=SYSDA,SPACE=(TRK,(30,10),RLSE),
00700004
//          DCB=(RECFM=FB,LRECL=23,BLKSIZE=0)
00710004
//SITEFILE  DD   DSN=HSDHQV.VS610D01.SITEFILE,DISP=SHR
00720004
//*
00740004
```

```

//HSD8500T PROC VER840=A,VER850=4,VER860=3,VER686=0,
00000114
//          FY=,AP=,WK=,EWEEK=
00000201
//*****00000
300
//** MOVE WEEKLY RPW DATA TO TEMPORARY DATASET
00000400
//*****00000
500
//S01      EXEC PGM=IDCAMS
00001000
//OUT1      OUTPUT  DEST=U8058,CLASS=H,COPIES=1
00002010
//OUT2      OUTPUT  DEST=HQ1,CLASS=X,COPIES=1
00003010
//OUT3      OUTPUT  DEST=U281,CLASS=A,COPIES=1
00003110
//SYSPRINT DD  SYSOUT=*
00003200
//SYSUT1    DD  DSN=HSDHQV.VS805D01.RPWDATA,DISP=OLD
00003300
//SYSUT2    DD  DSN=HSDHQN.PS805D01.RPWDATA,
00003400
//          DISP=(NEW,CATLG),
00003500
//          UNIT=SYSDA,SPACE=(CYL,(30,10),RLSE),
00003600
//          DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00003700
//SYSIN     DD  DSN=HSDHQN.CODES.CONTROL(HSD600U7),DISP=SHR
00003800
//*
00003900
//*****00004
000
//** MOVE WEEKLY RPW AUDIT DATA TO TEMPORARY DATASET
00004100
//*****00004
200
//S02      EXEC PGM=IDCAMS,COND=(0,NE)
00004300
//SYSPRINT DD  SYSOUT=*
00004400
//SYSUT1    DD  DSN=HSDHQV.VS805D01.RPWAUDIT,DISP=OLD
00004500
//SYSUT2    DD  DSN=HSDHQN.PS805D01.RPWAUDIT,
00004600
//          DISP=(NEW,CATLG),
00004700
//          UNIT=SYSDA,SPACE=(CYL,(12,5),RLSE),
00004800

```

```

//                      DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00004900
//SYSIN     DD DSN=HSDHQN.CODES.CONTROL(HSD600U7),DISP=SHR
00005000
//*
00005100
//*****00005
200
//** MOVE WEEKLY DATA TO TAPE AND CREATE NEW VSAM FILE
00005300
//*****00005
400
//S04      EXEC PGM=IDCAMS,COND=(0,NE)
00005500
//SYSPRINT DD SYSOUT=*
00005600
//SYSUT1    DD DSN=HSDHQV.VS805D01.RPWDATA,DISP=OLD
00005700
//SYSUT2    DD DSN=HSDHQN.PS805T01.RPWDATA.RPW&FY&AP&WK,
00005800
//                  DISP=(NEW,CATLG,DELETE),
00005900
//                  UNIT=TAPE,LABEL=RETPD=1825,
00006005
//                      DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00006100
//SYSUT3    DD DSN=HSDHQN.PS001D01.BLANK150,DISP=SHR
00006200
//SYSIN     DD DSN=HSDHQN.CODES.CONTROL(HSD85001),DISP=SHR
00006300
//*
00006400
//*****00006
500
//** MOVE WEEKLY AUDIT DATA TO TAPE AND CREATE NEW VSAM FILE
00006600
//*****00006
700
//S06      EXEC PGM=IDCAMS,COND=(0,NE)
00006800
//SYSPRINT DD SYSOUT=*
00006900
//SYSUT1    DD DSN=HSDHQV.VS805D01.RPWAUDIT,DISP=OLD
00007000
//SYSUT2    DD DSN=HSDHQN.PS805T01.RPWAUDIT.RPW&FY&AP&WK,
00007100
//                  DISP=(NEW,CATLG,DELETE),
00007200
//                  UNIT=TAPE,LABEL=RETPD=1825,
00007305
//                      DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00007400

```

```

//SYSUT3 DD DSN=HSDHQN.PS001D01.BLANK150,DISP=SHR
00007500
//SYSIN DD DSN=HSDHQN.CODES.CONTROL(HSD85002),DISP=SHR
00007600
/*
00007700
*****00007
800
/** SORT WEEKLY RPW DATA
00007900
*****00008
000
//S08 EXEC PGM=SORT,COND=(0,NE)
00008100
//SYSOUT DD SYSOUT=*
00008200
//SYSUDUMP DD SYSOUT=*
00008300
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(30,5))
00008400
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(30,5))
00008500
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(30,5))
00008600
//SORTIN DD DSN=HSDHQN.PS805D01.RPWDATA,DISP=OLD
00008700
//          DD &EDITDUM.DISP=OLD,
00008900
//          DSN=HSDHQN.PS840D01.EDITFILE.RPW&EDITDEF.
00009000
//SORTOUT DD DSN=&&RPWDATA,
00009100
//          DISP=(NEW,PASS,DELETE),
00009200
//          DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150),
00009300
//          UNIT=SYSDA,SPACE=(CYL,(30,10),RLSE)
00009400
//SYSIN DD DSN=HSDHQN.CODES.CONTROL(HSD85003),DISP=SHR
00009500
***** SORT FIELDS=(1,12,A,31,4,A,58,4,A,13,8,A,21,10,A,
00009604
*****           35,114,A),FORMAT=CH
00009704
/*
00009800
*****00009
900
/** SORT WEEKLY RPW AUDIT DATA
00010000
*****00010
100

```

```

//S10      EXEC PGM=SORT,COND=(0,NE)
00010200
//SYSOUT   DD  SYSOUT=*
00010300
//SYSUDUMP DD  SYSOUT=*
00010400
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(5,5))
00010500
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(5,5))
00010600
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(5,5))
00010700
//SORTIN    DD  DSN=HSDHQN.PS805D01.RPWAUDIT,DISP=OLD
00010800
//SORTOUT   DD  DSN=&&AUDITRPW,
00010900
//                      DISP=(NEW,PASS,DELETE),
00011000
//                      DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150),
00011100
//                      UNIT=SYSDA,SPACE=(CYL,(5,5),RLSE)
00011200
//SYSIN     DD  DSN=HSDHQN.CODES.CONTROL(HSD85003),DISP=SHR
00011300
//***** SORT FIELDS=(1,12,A,31,4,A,58,4,A,13,8,A,21,10,A,
00011404
//*****           35,114,A),FORMAT=CH
00011504
/*
00011600
*****00011
715
/* TEMPORARY STEP FOR YEAR 2000 CONVERSION
00011815
*****00011
915
//SSAA      EXEC PGM=FIXRESC,TIME=10
00012015
//STEPLIB   DD  DSN=H20471.LOAD,DISP=SHR
00012115
//SYSOUT   DD  SYSOUT=*
00012215
//SYSPRINT DD  SYSOUT=*
00012315
//INFILE   DD  DSN=&&RPWDATA,DISP=(OLD,PASS)
00012417
/*
00012515
*****00012
615
/* TEMPORARY STEP FOR YEAR 2000 CONVERSION
00012715

```

```

//*****00012
815
//SSBB      EXEC PGM=FIXRESC,TIME=10
00012915
//STEPLIB  DD  DSN=H20471.LOAD,DISP=SHR
00013015
//SYSOUT    DD  SYSOUT=*
00013115
//SYSPRINT  DD  SYSOUT=*
00013215
//INFILE    DD  DSN=&&AUDITRPW,DISP=(OLD,PASS)
00013317
/*
00013415
//*****00013
515
/* BACK UP SAMPLE EXTRACT VSAM FILE PRIOR TO CHECK-IN RUN
00013615
//*****00013
715
//S14      EXEC PGM=IDCAMS,COND=(0,NE)
00013815
//SYSPRINT  DD  SYSOUT=*
00013915
//SYSUDUMP  DD  SYSOUT=*
00014015
//SYSUT1    DD  DSN=HSDHQV.VS830D01.EXTRACT,DISP=OLD
00014115
//SYSUT2    DD  DSN=HSDHQN.PS830T01.EXTRACT1.RPW&FY&AP&WK,
00014215
//                  DISP=(NEW,CATLG,UNCATLG),
00014315
//                  UNIT=TAPE,LABEL=RETPD=1825,
00014415
//                  DCB=(RECFM=FB,LRECL=132,BLKSIZE=6204)
00014515
//SYSIN     DD  DSN=HSDHQN.CODES.CONTROL(HSD600U7),DISP=SHR
00014615
/*
00014715
//*****00014
815
/* RPW MAIN CHECKIN PROCESSING RUN
00014915
//*****00015
015
//S16      EXEC PGM=HSD840C&VER840,COND=(0,NE),
00015115
//                  PARM='&EWEEK.&AP.&WK.'
00015215
//STEPLIB  DD  DSN=HSDHQN.CODES.LOAD,DISP=SHR
00015315

```

```
//SYSOUT DD SYSOUT=*
00015415
//SYSUDUMP DD SYSOUT=*
00016015
//HSD840P1 DD DSN=&&HSD840P1,DISP=(NEW,PASS),
00220114
// UNIT=SYSDA,SPACE=(CYL,(5,5)),
00220214
// DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00220314
//HSD840P2 DD DSN=&&HSD840P2,DISP=(NEW,PASS),
00220614
// UNIT=SYSDA,SPACE=(CYL,(5,5)),
00220714
// DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00220814
//HSD840P3 DD DSN=&&HSD840P3,DISP=(NEW,PASS),
00221114
// UNIT=SYSDA,SPACE=(CYL,(5,5)),
00221214
// DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00221314
//MSCMAST DD DSN=HSDHQV.VS610D01.SITEFILE,DISP=SHR
00221500
//SAMPMAST DD DSN=HSDHQV.VS830D01.EXTRACT,DISP=OLD
00221600
//RPWIN DD DSN=&&RPWDATA,DISP=(OLD,PASS)
00221700
//AUDIN DD DSN=&&AUDITRPW,DISP=(OLD,PASS)
00221800
//RPWOUT DD DSN=&&RPWOUT,DISP=(NEW,PASS),
00221900
// UNIT=SYSDA,SPACE=(CYL,(30,10),RLSE),
00222000
// DCB=(RECFM=FB,LRECL=118,BLKSIZE=6136)
00222100
//AUDOUT DD DSN=&&AUDOUT,DISP=(NEW,PASS),
00222200
// UNIT=SYSDA,SPACE=(CYL,(5,5),RLSE),
00222300
// DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00222400
//EDITFILE DD DSN=HSDHQN.PS840D01.EDITFILE.RPW&FY&AP&WK,
00222500
// DISP=(NEW,CATLG,DELETE),
00222600
// UNIT=SYSDA,SPACE=(CYL,(12,4),RLSE),
00222700
// DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00222800
//FUTUREWK DD DSN=HSDHQN.PS840D01.FUTUREWK.RPW&FY&AP&WK,
00222900
```

```

//                      DISP=(NEW,CATLG,DELETE),
00223000
//                      UNIT=SYSDA,SPACE=(CYL,(12,4),RLSE),
00223100
//                      DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00223200
//RESCHED DD DSN=&&RESCHED,DISP=(NEW,PASS),
00223300
//                      UNIT=SYSDA,SPACE=(CYL,(1,1)),
00223400
//                      DCB=(RECFM=FB,LRECL=103,BLKSIZE=6180)
00223514
//ERROR DD DSN=&&ERROR,DISP=(NEW,PASS),
00223600
//                      UNIT=SYSDA,SPACE=(CYL,(5,1)),
00223700
//                      DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00223800
//STATUS DD DSN=&&STATUS,DISP=(NEW,PASS),
00223900
//                      UNIT=SYSDA,SPACE=(CYL,(5,1)),
00224000
//                      DCB=(RECFM=FB,LRECL=94,BLKSIZE=6110)
00224114
/*
00224200
*****00224
300
/* BACK UP SAMPLE EXTRACT VSAM FILE AFTER CHECK-IN RUN
00224400
*****00224
500
//S18      EXEC PGM=IDCAMS,COND=(0,NE)
00224600
//SYSPRINT DD SYSOUT=*
00224700
//SYSUDUMP DD SYSOUT=*
00224800
//SYSUT1    DD DSN=HSDHQV.VS830D01.EXTRACT,DISP=OLD
00224900
//SYSUT2    DD DSN=HSDHQN.PS830T01.EXTRACT2.RPW&FY&AP&WK,
00225000
//                      DISP=(NEW,CATLG,UNCATLG),
00225100
//                      UNIT=TAPE,LABEL=RETPD=1825,
00225200
//                      DCB=(RECFM=FB,LRECL=132,BLKSIZE=6204)
00225300
//SYSIN     DD DSN=HSDHQN.CODES.CONTROL(HSD600U7),DISP=SHR
00225400
/*
00225500

```

```

//*****00225
600
///* CREATE RPW STATUS REPORT FILE
00225700
//*****00225
800
//S20      EXEC PGM=HSD850C&VER850, PARM='&AP.&WK.',COND=(0,NE)
00225900
//STEPLIB DD DSN=HSDHQN.CODES.LOAD,DISP=SHR
00226002
//SYSOUT    DD  SYSOUT=*
00226100
//SYSUDUMP DD  SYSOUT=*
00226200
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
00226300
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
00226400
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
00226500
//MSCMAST   DD  DSN=HSDHQV.VS610D01.SITEFILE,DISP=SHR
00226600
//STATUS     DD  DSN=&&STATUS,DISP=(OLD,DELETE)
00226700
//HSD850P1  DD  DSN=&&HSD850P1,DISP=(NEW,PASS),
00226914
//                      UNIT=SYSDA,SPACE=(CYL,(5,5)),
00227014
//                      DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00227114
//HSD850P2  DD  DSN=&&HSD850P2,DISP=(NEW,PASS),
00227414
//                      UNIT=SYSDA,SPACE=(CYL,(5,5)),
00227514
//                      DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00227614
//HSD850P3  DD  DSN=&&HSD850P3,DISP=(NEW,PASS),
00227914
//                      UNIT=SYSDA,SPACE=(CYL,(5,5)),
00228014
//                      DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00228114
//HSD850P4  DD  DSN=&&HSD850P4,DISP=(NEW,PASS),
00228300
//                      UNIT=SYSDA,SPACE=(CYL,(5,5)),
00228400
//                      DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00228500
//HSD850P5  DD  DSN=&&HSD850P5,DISP=(NEW,PASS),
00228714
//                      UNIT=SYSDA,SPACE=(CYL,(5,5)),
00228814

```

```

//          DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00228914
//*
00229100
//*****00229
200
//** CREATE RPW RESCHEDULE REPORT AND MERGE WITH STATUS REPORT
00229300
//*****00229
400
//S22      EXEC PGM=HSD860C&VER860,PARM='&AP.&WK.',COND=(0,NE)
00229500
//STEPLIB DD DSN=HSDHQN.CODES.LOAD,DISP=SHR
00229602
//SYSOUT   DD  SYSOUT=*
00229700
//SYSUDUMP DD  SYSOUT=*
00229800
//RESCHED  DD  DSN=&&RESCHED,DISP=(OLD,PASS)
00229900
//HSD860P1 DD  DSN=&&HSD860P1,DISP=(NEW,PASS),
00230014
//          UNIT=SYSDA,SPACE=(CYL,(5,5)),
00230100
//          DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00230200
//*
00230300
//*****00230
400
//** CREATE RPW INPUT DATA TAPE
00230500
//*****00230
600
//S24      EXEC PGM=DSUTIL,COND=(0,NE)
00230700
//SYSPRINT DD  SYSOUT=*
00230800
//SYSUT1   DD  DSN=&&RPWDATA,DISP=(OLD,DELETE)
00230903
//SYSUT2   DD  DSN=HSDHQN.PS840T01.CODES.RPW&FY.&AP.&WK.,
00231000
//          DISP=(NEW,CATLG,DELETE),LABEL=RETPD=1825,
00231100
//          UNIT=TAPE,VOL=(PRIVATE,RETAIN),
00231200
//          DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00231300
//*
00231400
//*****00231
500

```

```

/* CREATE RPW OUTPUT DATA TAPE
00231600
*****00231
700
//S26      EXEC PGM=DSUTIL,COND=(0,NE)
00231800
//SYSPRINT DD  SYSOUT=*
00231900
//SYSUT1    DD  DSN=&&RPWOUT,DISP=(OLD,DELETE)
00232003
//SYSUT2    DD  DSN=HSDHQN.PS840T02.CODESOUT.RPW&FY.&AP.&WK.,
00232100
//                  DISP=(NEW,CATLG,DELETE),LABEL=(2,SL,,,RETPD=1825),
00232200
//                  UNIT=TAPE,VOL=(PRIVATE,RETAIN,,,REF=*.S24.SYSUT2),
00232300
//                  DCB=(RECFM=FB,LRECL=118,BLKSIZE=6136)
00232400
//*
00232500
*****00232
600
/* CREATE RPW OUTPUT RESCHEDULE DATA TAPE
00232700
*****00232
800
//S28      EXEC PGM=DSUTIL,COND=(0,NE)
00232900
//SYSPRINT DD  SYSOUT=*
00233000
//SYSUT1    DD  DSN=&&RESCHED,DISP=(OLD,DELETE)
00233103
//SYSUT2    DD  DSN=HSDHQN.PS840T03.RESCHED.RPW&FY.&AP.&WK.,
00233200
//                  DISP=(NEW,CATLG,DELETE),LABEL=(3,SL,,,RETPD=1825),
00233300
//                  UNIT=TAPE,VOL=(PRIVATE,RETAIN,,,REF=*.S24.SYSUT2),
00233400
//                  DCB=(RECFM=FB,LRECL=103,BLKSIZE=6180)
00233514
//*
00233600
*****00233
700
/* CREATE RPW OUTPUT ERROR DATA TAPE
00233800
*****00233
900
//S30      EXEC PGM=DSUTIL,COND=(0,NE)
00234000
//SYSPRINT DD  SYSOUT=*
00234100

```

```

//SYSUT1 DD DSN=&&ERROR,DISP=(OLD,DELETE)
00234203
//SYSUT2 DD DSN=HSDHQN.PS840T04.ERROR.RPW&FY.&AP.&WK.,
00234300
// DISP=(NEW,CATLG,DELETE),LABEL=(4,SL,,,RETPD=1825),
00234400
// UNIT=TAPE,VOL=(PRIVATE,RETAIN,,,REF=*.S24.SYSUT2),
00234500
// DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00234600
//*
00234700
//*****00234
800
///* CREATE RPW OUTPUT AUDIT DATA TAPE
00234900
//*****00235
000
//S32 EXEC PGM=DSUTIL,COND=(0,NE)
00235100
//SYSPRINT DD SYSOUT=*
00235200
//SYSUT1 DD DSN=&&AUDOUT,DISP=(OLD,DELETE)
00235303
//SYSUT2 DD DSN=HSDHQN.PS840T05.AUDIT.RPW&FY.&AP.&WK.,
00235400
// DISP=(NEW,CATLG,DELETE),LABEL=(5,SL,,,RETPD=1825),
00235500
// UNIT=TAPE,VOL=(PRIVATE,RETAIN,,,REF=*.S24.SYSUT2),
00235600
// DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00235700
//*
00235800
//*****00235
900
///* CREATE RPW STATUS REPORT TAPE
00236000
//*****00236
100
//S36 EXEC PGM=DSUTIL,COND=(0,NE)
00236200
//SYSPRINT DD SYSOUT=*
00236300
//SYSUT1 DD DSN=HSDHQN.PS001D01.RPW.TEXT,DISP=SHR
00236400
// DD DSN=&&HSD840P1,DISP=(OLD,PASS)
00236514
// DD DSN=&&HSD840P2,DISP=(OLD,PASS)
00236614
// DD DSN=&&HSD840P3,DISP=(OLD,PASS)
00236714

```

```

//          DD  DSN=&&HSD850P1,DISP=(OLD,PASS)
00236814
//          DD  DSN=&&HSD850P2,DISP=(OLD,PASS)
00236914
//          DD  DSN=&&HSD850P3,DISP=(OLD,PASS)
00237014
//          DD  DSN=&&HSD850P5,DISP=(OLD,PASS)
00237114
//          DD  DSN=&&HSD850P4,DISP=(OLD,PASS)
00237214
//          DD  DSN=&&HSD860P1,DISP=(OLD,PASS)
00237314
//SYSUT2    DD  DSN=HSDHQN.PS850T01.STATRPT.RPW&FY.&AP.&WK.,
00237414
//                      DISP=(NEW,CATLG,DELETE),LABEL=(6,SL,,,RETPD=1825),
00237514
//                      UNIT=TAPE,VOL=(PRIVATE,REF=*.S24.SYSUT2),
00237614
//                      DCB=(RECFM=FB,LRECL=133,BLKSIZE=13300)
00237714
///*
00237814
//*****00237
914
///* CREATE RPW INPUT AUDIT DATA TAPE
00238014
//*****00238
114
//S38      EXEC PGM=DSUTIL,COND=(0,NE)
00238214
//SYSPRINT DD  SYSOUT=*
00238314
//SYSUT1    DD  DSN=&&AUDITRPW,DISP=(OLD,DELETE)
00238414
//SYSUT2    DD  DSN=HSDHQN.PS840T01.AUDTDATA.RPW&FY.&AP.&WK.,
00238514
//                      DISP=(NEW,CATLG,DELETE),LABEL=(7,SL,,,RETPD=1825),
00238614
//                      UNIT=TAPE,VOL=(PRIVATE,RETAIN,,,REF=*.S24.SYSUT2),
00238714
//                      DCB=(RECFM=FB,LRECL=150,BLKSIZE=6150)
00238814
///*
00238914
//*****00239
014
///*          CREATE TEXT FILE VSAM
00239114
//*****00239
214
//S40      EXEC PGM=IDCAMS,COND=(0,NE)
00239314

```

```

//SYSPRINT DD SYSOUT=*
00239414
//SYSUDUMP DD SYSOUT=*
00239514
//SYSIN      DD DSN=HSDHQN.CODES.CONTROL(HSD85004),DISP=SHR
00239614
/*
00239714
*****00239
814
//** LOAD TEXT FILE VSAM FROM MSC STATUS REPORT DATASET
00239914
*****00240
014
//S42      EXEC PGM=HSD686C&VER686,COND=(0,NE)
00240114
//STEPLIB  DD DSN=HSDHQN.CODES.LOAD,DISP=SHR
00240214
//SYSOUT   DD SYSOUT=*
00240314
//SYSPRINT DD SYSOUT=*
00240414
//SYSUDUMP DD SYSOUT=*
00240514
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
00240614
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
00240714
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(30,30),RLSE)
00240814
//SITEFILE DD DSN=HSDHQV.VS610D01.SITEFILE,DISP=SHR
00240914
//REPORT    DD DSN=&&HSD850P4,DISP=(OLD,PASS)
00241014
//HQTEXT    DD DSN=HSDHQV.VS850D01.RPW.TEXT,DISP=OLD
00241114
//TEXTSORT  DD DSN=&&TEMP12,
00241214
//                  DISP=(NEW,DELETE),
00241314
//                  UNIT=SYSDA,SPACE=(CYL,(30,10),RLSE),
00241414
//                  DCB=(RECFM=FB,LRECL=142,BLKSIZE=6248)
00241516
/*
00241614
*****00241
714
//** UNATLG PRIOR WEEK HEADQUARTERS TEXT
00241814
*****00241
914

```

```

//S44      EXEC PGM=IEFBR14,COND=(0,NE)
00242014
//SYSPRINT DD  SYSOUT=*
00242114
//DD1      DD  DSN=HSDHQN.PS001D01.RPW.TEXT,DISP=(SHR,DELETE)
00242214
/*
00242314
//*****00242
414
//          CREATE COPY OF CURRENT WEEK HEADQUARTERS TEXT
00242514
//*****00242
614
//S46      EXEC PGM=IEBGENER,COND=(0,NE)
00242714
//SYSPRINT DD  SYSOUT=*
00242814
//SYSUDUMP DD  SYSOUT=*
00242914
//SYSUT1    DD  DSN=HSDHQN.PS850D01.HQTRS.TEXT,DISP=OLD
00243014
//SYSUT2    DD  DSN=HSDHQN.PS001D01.RPW.TEXT,
00243114
//          DISP=(NEW,CATLG,DELETE),
00243214
//          UNIT=SYSDA,SPACE=(TRK,(5,2),RLSE),
00243314
//          DCB=(RECFM=FB,LRECL=133,BLKSIZE=6118)
00243414
//SYSIN     DD  DUMMY
00243514
//*****00243
614
//          DELETE RPW DATA AND AUDIT FILES
00243714
//*****00243
814
//S48      EXEC PGM=IEFBR14,COND=(0,NE)
00243914
//SYSPRINT DD  SYSOUT=*
00244014
//DD1      DD  DSN=HSDHQN.PS805D01.RPWDATA,DISP=(OLD,DELETE)
00244114
//DD2      DD  DSN=HSDHQN.PS805D01.RPWAUDIT,DISP=(OLD,DELETE)
00244214
/*
00244314
//*****00244
414
//          REPRO NEXT WEEKS DATA TO RPW DATA FILE FOR REUSE
00244515

```

```

//*****00244
614
//S50      EXEC PGM=IDCAMS,COND=(0,NE)
00244715
//SYSPRINT DD  SYSOUT=*
00244814
//SYSUDUMP DD  SYSOUT=*
00244914
//SYSUT1    DD  DSN=HSDHQN.PS840D01.FUTUREWK.RPW&FY&AP&WK,DISP=OLD
00245014
//SYSUT2    DD  DSN=HSDHQV.VS805D01.RPWDATA,DISP=SHR
00245116
//SYSIN     DD  DSN=HSDHQN.CODES.CONTROL(HSD600U7),DISP=SHR
00245516
//*
00245614
//*****00245
714
//*          ROUTE COPY OF REPORT TO HEADQUARTERS AND NORTHERN VA
00245814
//*****00246
010
//S52      EXEC PGM=IEBPTPCH,COND=(0,NE)
03757710
//SYSOUT    DD  SYSOUT=*
03758010
//SYSUDUMP DD  SYSOUT=*
03758110
//SYSPRINT DD  SYSOUT=*
03758210
//SYSUT1    DD  DSN=HSDHQN.PS001D01.RPW.TEXT,DISP=SHR
03758310
//          DD  DSN=&&HSD840P1,DISP=(OLD,DELETE)
03758415
//          DD  DSN=&&HSD840P2,DISP=(OLD,DELETE)
03758515
//          DD  DSN=&&HSD840P3,DISP=(OLD,DELETE)
03758615
//          DD  DSN=&&HSD850P1,DISP=(OLD,DELETE)
03758715
//          DD  DSN=&&HSD850P2,DISP=(OLD,DELETE)
03758815
//          DD  DSN=&&HSD850P3,DISP=(OLD,DELETE)
03758915
//          DD  DSN=&&HSD850P5,DISP=(OLD,DELETE)
03759015
//          DD  DSN=&&HSD850P4,DISP=(OLD,DELETE)
03759115
//          DD  DSN=&&HSD860P1,DISP=(OLD,DELETE)
03759215
//SYSUT2    DD  SYSOUT=(,),DCB=LRECL=133,
03759410

```

```
//          OUTPUT=(*.S01.OUT1,*.S01.OUT2,*.S01.OUT3)
03759510
//SYSIN    DD  DSN=HSDHQN.CODES.CONTROL(HSD85005),DISP=SHR
03759610
//*
03759710
//*****03759
813
//          MOVE BACKUP COPY OF UPLOADED RPW DATA TO TAPE
03759913
//*****03760
013
//S54      EXEC PGM=IEBGENER,COND=(0,NE)
03790013
//SYSOUT   DD  SYSOUT=*
03800013
//SYSPRINT DD  SYSOUT=*
03810013
//SYSUT1   DD  DSN=HSDHQN.PS805D01.BACKUP.RPWDATA,
03820013
//          DISP=(OLD,DELETE,KEEP)
03830013
//SYSUT2   DD  DSN=HSDHQN.PS805T01.BACKUP.RPWDATA.RPW&FY&AP&WK,
03840013
//          DISP=(NEW,CATLG,UNCATLG),
03850013
//          UNIT=TAPE,LABEL=RETPD=1825,
03860013
//          DCB=(STDMODEL,RECFM=FB,LRECL=150,BLKSIZE=27900)
03870013
//SYSIN    DD  DUMMY
03880013
//*
03890013
//*****03891
013
//          CREATE NEW BACKUP DISK FILE WITH BLANK RECORD
03892013
//*****03893
013
//S56      EXEC PGM=IEBGENER,COND=(0,NE)
03900013
//SYSOUT   DD  SYSOUT=*
03910013
//SYSPRINT DD  SYSOUT=*
03920013
//SYSUT1   DD  DSN=HSDHQN.PS001D01.BLANK150,DISP=SHR
03930013
//SYSUT2   DD  DSN=HSDHQN.PS805D01.BACKUP.RPWDATA,
03940013
//          DISP=(NEW,CATLG,DELETE),
03950013
```

```
//          UNIT=SYSDA,SPACE=(CYL,(30,10),RLSE),  
03960013  
//          DCB=(RECFM=FB,LRECL=150,BLKSIZE=27900)  
03970013  
//SYSIN   DD  DUMMY  
03980013  
//  
03981013
```

# **RPW-CODES**

## **Computer System Documentation Description**

### **SECTION 4**

### **Contents of CD-ROM**

## CONTENTS OF CD-ROM WITH SOURCE CODE

This CD-ROM contains all programs referenced in the Library Reference USPS-LR-J-22. The files are organized into directories on the CD-ROM as described below.

### 1. RPW Data Entry Program:

All program files related to the RPW Data Entry application in Section 1 are in the *CD-ROM\RPW\LT* directory.

### 2. RPW Base Unit Programs:

All program files related to RPW the Base Unit application in Section 2 are contained in the *CD-ROM\RPW\BU* directory.

### 3. RPW Mainframe programs:

All Mainframe programs related to the RPW Mainframe application in Section 3 are contained in the *CD-ROM\RPW\MF* directory. Control files are saved in the *CD-ROM\RPW\MF\Control* directory, Source programs are in the *CD-ROM\RPW\MF\Source* directory, and JCLs are contained in the *CD-ROM\RPW\MF\Proc* directory.

### 4. RPW Computer System Documentation Description:.

RPW Computer System Documentation Description is contained in *CD-ROM\RPW\LR-J-22.doc*.