# Standard POS Device Configuration Manual

BASE24®

# Contents

**The ACI Standard POS Message *continued***

## SPDH Module Configuration Considerations *continued*

## Configuration Files Maintenance and Set Up *continued*

**Configuration Files Maintenance and Set Up *continued***

**Timeout Reversals *continued***

# Preface

The ***BASE24-pos Standard POS Device Configuration Manual*** is designed to give readers a thorough understanding of the BASE24 Standard POS Device Handler (SPDH) module. This manual is designed to provide a general overview of the SPDH module and how it is configured. It provides information on topics important in gaining a basic understanding of the SPDH module, such as supported features and terminals. However, this manual is also helpful to readers who require a detailed explanation of the SPDH module and its functionality. It provides detailed information including how to configure the SPDH module, the files used with the SPDH module, and the transaction flows used with the SPDH module.

## Audience

This manual is intended as a reference for financial institution personnel responsible for updating and maintaining ACI standard POS devices for use with BASE24-pos.

## Prerequisites

A knowledge of BASE24-pos is encouraged for the customer who is reading this manual. This manual is designed to be used in conjunction with the ***ACI Standard POS Device Message Specifications Manual***, which defines the format of the ACI standard POS message. If more information about BASE24-pos is desired, readers should refer to the ***BASE24-pos Transaction Processing Manual***.

## Additional Documentation

The BASE24 documentation set is arranged so that each BASE24 manual presents a topic or group of related topics in detail. When one BASE24 manual presents a topic that has already been covered in detail in another BASE24 manual, the topic is summarized and the reader is directed to the other manual for additional information. Information has been arranged in this manner to be more efficient for

readers that do not need the additional detail and at the same time provide the source for readers that require the additional information. This manual contains references to the following BASE24 publications:

- The ***BASE24 CRT Access Manual*** contains information on accessing and printing BASE24-pos files maintenance screen.
- The ***BASE24 Base Files Maintenance Manual*** discusses the base files used with the SPDH module.
- The ***BASE24 Device Control Manual*** contains additional information on the EMT Control Command screen.
- The ***BASE24 Logical Network Configuration File Manual*** discusses in detail the assigns and params used in configuring the SPDH module.
- The ***BASE24 Event Message Reference Manual*** discusses in detail how to access event messages generated by the SPDH module.
- The ***BASE24 Text Command Reference Manual*** discusses in detail text commands initiated from the network control facility.
- The ***BASE24 Tokens Manual*** provides information about tokens used with the BASE24-pos Standard Internal Message.
- The ***BASE24 Transaction Security Manual*** discusses in detail PINs, message authentication codes (MACs), and dynamic key management used to provide transaction security for the SPDH module.
- The ***BASE24 Transaction Security Services Processing Guide*** describes how the Transaction Security Services process is integrated into a BASE24 system. It describes how BASE24 hardware security data is converted and maintained in data files, how BASE24 processes are configured to use the Transaction Security Services process, and the implementation requirements for using the process.
- The ***BASE24-pos Standard POS Device Support Manual*** documents the release 5.3 native message format used by the SPDH module.
- The ***BASE24-pos Address Verification Manual*** provides more information on the BASE24-pos add-on Address Verification module.
- The ***BASE24-pos Transaction Processing Manual*** discusses in detail transaction processing used in configuring the SPDH module as well as the BASE24-pos Standard Internal Message (PSTM).
- The ***BASE24-pos Files Maintenance Manual*** discusses the BASE24-pos files used with the SPDH module.

- The ***BASE24-pos EMV Support Manual*** discusses configuration and processing options for the BASE24-pos Europay, MasterCard, and Visa (EMV) add-on product.
- The ***BASE24-pos Multiple Currency Support Manual*** discusses configuration and processing options for the BASE24-pos Multiple Currency add-on product.
- The ***ACI Standard POS Device Message Specifications Manual*** contains detailed descriptions of ACI standard POS message data elements.

This manual also contains references to the ***NET24-XPNET Network Control Operations Guide,*** which discusses network control facilities, and to the ***NET24-XPNET Network Control Command Reference Manual,*** which discusses the process, station, line and device declarations used in configuring the SPDH module.

In addition, the ANSI X9.19-1986 standard, ***Financial Institution Retail Message Authentication,*** describes the standard used for the support of message authentication codes (MACs).

## Software

This manual documents standard processing as of its publication date. Software that is not current and custom software modifications (CSMs) may result in processing that differs from the material presented in this manual. The customer is responsible for identifying and noting these changes.

## Manual Summary

The following is a summary of the contents of this manual.

"Conventions Used in This Manual" follows this preface and describes notation and documentation conventions necessary to understand the information in the manual.

**Section 1, "Introduction,"** provides an overview of the SPDH module and its functionality. It also defines the responsibilities that ACI, the customer, and the vendor have when using a terminal connected to the SPDH module. This section includes information explaining how the SPDH module can be integrated with BASE24-pos. This section also discusses the role of the SPDH module during transaction processing.

**Section 2, "Terminal Configuration,"** explains the options available to customers configuring their POS terminals. This section explains the files that must be set up in order to tailor SPDH configuration options to meet the needs of a particular customer. This section includes descriptions of the Logical Network Configuration File (LCONF) assigns and params used by the SPDH module.

**Section 3, "BASE24–pos Standard POS Device Handler Module,"** contains information about the SPDH module. It includes an overview of module functions, a description of each file used by the SPDH module during processing, a description of the steps the SPDH module performs during initialization and warmbooting, and a list of the text commands recognized by the SPDH module.

**Section 4, "The ACI Standard POS Message,"** contains information on how standard message header fields are set or interpreted by the SPDH module, how optional data fields are mapped to and from the POS Standard Internal Message (PSTM), BASE24 files, or BASE24 tokens, and how transaction codes are mapped to and from BASE24-pos internal transaction codes. This section is designed to be used in conjunction with the detailed message formats described in the ***ACI Standard POS Device Message Specifications Manual***.

**Section 5, "Download Data,"** contains information about the role of the SPDH module when downloading configuration data to terminals for use during processing. It includes information about the records that can be fully or partially downloaded to terminals from the ACI Standard Device Configuration File (ACNF), the BASE24-pos Terminal Data files (PTD), and the POS Retailer Definition File (PRDF).

**Section 6, "SPDH Module Configuration Considerations,"** discusses topics the customer must consider when deciding on how to configure the SPDH module. It includes message sequencing options, message authentication code (MAC) information, draft capture options, transaction accumulation totals information, handshaking transaction support, and BASE24-mail support. In addition, section 6 contains information on how customers can configure receipt formats. It includes the standard header fields that are typically printed on receipts, the parameters that need to be set in order for the SPDH module to determine the type of response to return to the terminal, and the parameters to determine the exact text of the response.

**Section 7, "Transaction Message Flows,"** contains transaction flows between the terminal and the SPDH module. This section includes BASE24-mail transaction flows.

**Section 8, "Configuration Files Maintenance and Set Up,"** contains the files used in configuring the SPDH module. BASE24-pos file maintenance manipulates these files through the use of formatted screens. This section shows the formatted screens used by the SPDH module and describes each field shown on these screens. It also includes the procedures an operator can take to configure each screen to meet specific requirements.

**Appendix A, "Track Data Processing,"** describes how the SPDH module formats Track 1 and Track 2 information in the BASE24-pos Standard Internal Message (PSTM).

**Appendix B, "Timeout Reversals,"** describes how the SPDH module processes a reversal caused by a timeout in a controller-based environment. The scenarios in this appendix also apply to terminals that are capable of generating reversals.

**Appendix C, "BASE24-pos Mobile Top-Up Extension**," describes how the BASE24-pos Mobile Top-Up Device Handler extension enables cardholders to replenish, or *top up*, their mobile telephone accounts at the point-of-sale (POS) device.

## Publication Identification

Three entries appearing at the bottom of each page uniquely identify this BASE24 publication. The publication number (for example, PS-DP140-02 for the ***BASE24–pos Standard POS Device Configuration Manual***) appears on every page to assist readers in identifying the manual from which a page of information was printed. The publication date (for example, Nov-2007 for November, 2007) indicates the issue of the manual. The software release information (for example, R6.0v7 for release 6.0, version 8) specifies the software that the manual describes. This information matches the document information on the copyright page of the manual.

*ACI Worldwide, Inc.*

# Conventions Used in This Manual

This section explains how fields and unlabeled fields on screens, blank characters, and optional data fields and subfields in the ACI standard POS message are documented in this manual.

## Field Descriptions

Each field appearing on a file maintenance screen is listed by name and then described. Field descriptions briefly summarize the contents, purpose, and permissible values, as shown in the following example taken from the ACI Standard Device Configuration File (ACNF).

***Example***

**TERMINAL GROUP —** The terminal group combined with the RECORD TYPE and RECORD ID is the primary key into the ACNF. It identifies the terminal or group of terminals to which the configuration information set up in the ACNF applies. The configuration information is used by all terminals that have a value corresponding to this field in the TERMINAL GROUP field on POS Terminal Data files (PTD) screen 1.

| | |
|---|---|
| Field Length: | 4 alphanumeric characters |
| Required Field: | Yes |
| Default Value: | 0000 |
| Data Name: | ACNF.KEY-S.GRP |

***Explanation***

Each field description is completed by one or more of the following items of information:

| Item | Description |
|---|---|
| Example | Illustrates a possible entry for the field to further clarify the value or values that can be entered. |

| Item | Description |
|---|---|
| Field Length | Specifies the size of the field and the type of characters that can be entered. This length refers to the field and valid values for the file maintenance screen, not the field in the DDLs. Possible values are alphabetic, alphanumeric, hexadecimal, and numeric. The term *alphanumeric* includes all alphabetic, numeric, and special characters that can be entered from a keyboard without using a control sequence. The term *hexadecimal* includes all numbers and the letters A through F.<br><br>When a field value cannot be modified by the operator, the field length is System-protected. |
| Occurs | Indicates the number of times the field can be displayed on the screen. This information is provided only when the field can be displayed multiple times. |
| Required Field | Specifies whether a value has to be entered in the field. Possible values are Yes and No. Some fields are required only under certain conditions. In this case, the entry is Yes, followed by the conditions that determine when the field is required. |
| Default Value | Specifies the value that is automatically placed in the field when the screen is first displayed or when the **F8** key is pressed to clear the screen. |
| Data Name | Provides the DDL name associated with the field appearing on the screen. Data names are included in the documentation to assist in communicating screen and field issues to your technical staff. Note that screen data is not always stored in the BASE24 database as it appears on the screen or as it is described in the field description. If you need information on how screen data is actually stored, consult the DDLs. |

## Unlabeled Fields on Screens

Angle brackets (< >) indicate an unlabeled field on a screen. An unlabeled field is a field that is present on a screen but is not preceded by an identifying literal label. For the purposes of documenting the field, a label has been assigned and appears inside the angle brackets. A multiple line unlabeled field is displayed as a shaded area and also has a label in angle brackets.

In the field descriptions for the screen, the unlabeled field appears according to its place on the screen and is identified by the same label.

Unlabeled fields are not included in the index by field name; they appear by subject in the main index.

## Blank Characters

Field descriptions for internal and external messages, tokens, files maintenance screens, and reports often include lists of valid values. When the value contains a blank character, this manual uses the symbol ␣ to indicate the blank character.

## Optional Data Field and Subfield Description Format

Optional data fields and optional data subfields are described in section 4 using a standard format. Each field and subfield description begins with a title structured as shown below.

**FID x — Field Name**
**SFID x — Subfield Name**

| | |
|---|---|
| FID *x* | The field identifier for this field, where *x* is the alphabetic (A–Z, a–z) or numeric (0–9) character that identifies the field. |
| Field Name | A text description of the data associated with the FID. |
| SFID *x* | The subfield identifier for this subfield, where *x* is the alphabetic (A–Z, a–z) character that identifies the subfield. |
| Subfield Name | A text description of the data associated with the SFID. |

The FID/SFID title is followed by these standard labeled fields.

**Request:** Indicates whether the FID/SFID can be included in request messages. *Optional* indicates that the customer can configure the FID/SFID to be included in request messages. *Not available* indicates that the FID/SFID cannot be included in request messages. An indication of whether the request FID/SFID length is fixed or variable is also included.

**Response:** Indicates whether the FID/SFID can be included in response messages. *Optional* indicates that the customer can configure the FID/SFID to be included in response messages. *Not available* indicates that the FID/SFID cannot be included in responses. FIDs/SFIDs that are echoed from the request are also noted. However, even FIDs/SFIDs that are noted as being echoed in responses are not included in responses at all, unless they are configured to be included at the host. An indication of whether the response FID/SFID length is fixed or variable is also included.

**Internal Field:** The name of the POS Standard Internal Message (PSTM) field, token, or file field to which this FID/SFID is mapped or associated with internally.

The standard labeled fields are followed by a description of the FID/SFID and its use. The description includes the valid values associated with the FID/SFID, if applicable, and if the data associated with the FID/SFID consists of a group of fields, these group fields are described as well.

# 1: Introduction

This section contains introductory-level information describing the BASE24 Standard POS Device Handler (SPDH) module and its functionality. Its main purpose is to provide readers with the background information they require as a basis for understanding the SPDH module. This section introduces the main topics that are important when using the SPDH module and provides general overviews for each topic identified.

This section is intended for readers who require only an overview of the SPDH module and its functionality. It provides information on topics important in gaining a basic understanding of the SPDH module, such as supported features and terminals. However, this section is also helpful to readers who require a detailed explanation of the SPDH module and its functionality. Section 1 lays the groundwork to prepare these readers for information covered in the rest of the manual by providing readers with previews of topics discussed in greater detail in subsequent sections. In addition, it provides comprehensive lists of the terminals supported by the SPDH module, the transactions supported by the SPDH module, and the files used by the SPDH module. Section 1 also outlines the responsibilities of ACI, the customer, and the vendor when implementing the SPDH module.

# BASE24 Standard POS Device Handler Module Overview

The BASE24 Standard POS Device Handler (SPDH) module was engineered and developed by ACI Worldwide Inc. (ACI), an international leader in the Electronic Funds Transfer (EFT)/Point-of-Sale (POS) industry, in response to changing standards encountered in the POS industry. These changing standards have resulted in the need for an increase in transaction support, configuration flexibility, functionality, and awareness of retailer requirements. ACI's answer to these inefficiencies is the SPDH module.

The SPDH module operates in conjunction with the BASE24-pos and BASE24-mail products. BASE24-pos is a point-of-sale transaction processing system. BASE24-mail is an electronic mail transfer system. The SPDH module provides unprecedented transaction support, configuration options, and functionality, thus addressing the shortcomings in the POS industry. In addition, the SPDH module allows retailers to select the functionality they require for a particular business application.

# Supported Terminals

Multiple terminal vendors have worked with the BASE24 Standard POS Device Handler (SPDH) specifications as set forth in this manual. Processors and retailers have used the SPDH specifications in developing the terminal functionality that meets their POS EFT requirements.

In addition, the SPDH module is flexible enough to communicate with any sophisticated third-generation POS terminal having the capability of sending messages to and receiving messages from the SPDH module using the message format supported by the SPDH module. Customers can use any POS terminal that conforms to the SPDH requirements set forth in this manual.

# Supported Protocols

The SPDH module can be configured to use the following communication protocols:

- ACISTD
- Asynchronous 3270
- Bisynchronous 3270
- POS-2 DATAPAC 3101/VISA End-to-End
- SNA LU Type 0
- TCP/IP
- Visa
- Visa 1
- Visa 2
- X.21
- X.25

The protocol selected by the customer depends on the line communications environment established between the SPDH module and its terminals. Line communications environments for the SPDH module can be dial, leased, X.21, or X.25. For example, the Bisynchronous 3270 protocol is used to accommodate leased bisynchronous communications with the SPDH module, whereas the Asynchronous 3270 protocol is used to accommodate leased asynchronous communications with the SPDH module. In contrast, the ACISTD protocol is required when the SPDH module is used in dial, X.21, or X.25 environments.

Authorized ACI personnel assist customers at installation or when a system is configured in determining the type of protocol best suited for the customer's particular requirements. For detailed information on protocols and their usage, refer to the ***NET24-XPNET Network Control Command Reference Manual***.

# Supported Features

An enhanced set of features is available for use with the SPDH module. Customers can select and configure the features necessary to meet their processing requirements. This subsection provides a list of these features, followed by a summary of each feature. The following features are supported by the SPDH module:

- Expanded transaction set
- Multiple terminal vendor support
- Configurable messages
- Multiple language support
- Configurable receipts
- Download options
- Draft capture options
- American Express data collection
- BASE24-pos settlement and cutover support
- Terminal balancing
- BASE24 transaction security
- Derived unique key per transaction (DUKPT) support
- American Express card security codes (CSCs)
- Message sequencing
- Event message generation
- BASE24-mail support
- PS2000 support
- Track 1 and Track 2 processing
- Address verification
- Port usage tracking
- Maximum returns and adjustments
- Stored value cards
- Balance inquiry service
- Electronic check authorization

- Electronic benefits transfer (EBT) support
- Europay, MasterCard, and Visa (EMV) chip card support
- Multiple currency support
- Mobile Top-Up support
- Contactless Transaction Support
- Dynamic Card Verification
- Healthcare/Transit Auto-Substantiation Transactions
- Healthcare Eligibility Inquiry Transactions
- Visa Card Level Results

## Expanded Transaction Set

The SPDH module can handle processing for a total of 33 different POS transactions sent from the terminal, including both financial and administrative transactions. Customer flexibility and functionality are greatly increased because the SPDH module supports a wide variety of transaction types. Some of the transactions supported by the SPDH module include check guarantee, check verification, purchase with cash back, preauthorization purchase, preauthorization purchase completion, and request mail. A complete list of the transaction set supported by the SPDH module is included later in this section.

## Multiple Terminal Vendor Support

Customers can use any POS terminal that conforms to the SPDH requirements set forth in this manual. The SPDH module is flexible enough to communicate with any sophisticated third-generation POS terminal having the capability of sending messages to and receiving messages from the SPDH module using the message format supported by the SPDH module.

## Configurable Messages

The SPDH message (also referred to as the ACI standard POS message) format allows customers to configure message requests and responses. The SPDH module can read requests from the terminal that include any number of optional

data fields. These optional data fields can be in the request in any order. The data fields included in responses sent to the terminal can also be configured by customers.

Customers can select a different message format for each transaction type, if they want. The request message and the response message for each transaction type can consist of different fields. The fields included in the request message can be in any order the customer chooses.

All SPDH messages consist of a standard header and optional data fields. The standard header is mandatory and can stand alone (i.e., Logon, Logoff, Handshake, and Totals transactions). However, it can be followed by any number of optional data fields. Optional data fields are subject to the requirements of the transaction. This means that certain transactions may require the presence of certain data fields in their messages. The only other restrictions on the number of optional data fields that can be included in the message are the protocol limitations, the size of the terminal read buffer, and the size of the SPDH read buffer. The SPDH read buffer is configurable to a maximum of 4,088 bytes, including all data communication characters associated with the message.

Optional data fields are identified in the SPDH message using field identifier (FID) characters. Some optional data fields can contain subfields, which are identified using subfield identifier (SFID) characters. The SPDH module maps data contained in optional data fields and subfields to and from BASE24-pos Standard Internal Message (PSTM) fields, tokens appended to the PSTM, or BASE24-pos files. This manual contains detailed information on optional data fields and subfields, FIDs, and SFIDs. The PSTM is described in the ***BASE24-pos Transaction Processing Manual*** and tokens are described in the ***BASE24 Tokens Manual***. BASE24-pos files are described in the ***BASE24-pos Files Maintenance Manual***.

## Multiple Language Support

Customers have the option of formatting display responses to a terminal in one of three different languages. The terminal may determine which language to use based on information contained in its request to the SPDH module. The SPDH module uses information contained in the terminal request to access language tables configured by the customer in the ACI Standard Device Response File (ARSP) and to format its response, or it uses information contained in the POS Terminal Data Static File—general data (PTDS1) to determine the language.

If no language is specified in the request, terminal responses are displayed in the default language set up in the LANGUAGE ID field on POS Terminal Data files (PTD) screen 1. If a language table is specified in the request, the information in the ARSP tables override the default value set up in the PTDS1. In short, the language table information must be requested by the terminal in order to be used. Otherwise, the default value in the PTDS1 is used. Setting up the multiple language support option is explained in section 8. For more information on the PTD, refer to the ***BASE24-pos Files Maintenance Manual***.

**Note:** Data entered or accessed from PTD screens is stored in different files, depending on whether the data is dynamic or static. Dynamic data, such as terminal totals, is stored in the POS Terminal Data Dynamic File—general data (PTDD1). Static data, such as the default language, is stored in the POS Terminal Data Static File—general data (PTDS1). Last message token data is optionally stored in the POS Terminal Data Dynamic File—scratch pad (PTDD2). Throughout this manual, the phrase "BASE24-pos Terminal Data files (PTD)," refers to all three of the above files.

## Configurable Receipts

The SPDH module does not format receipts. Customers must determine if receipts are necessary and then supply the optional data using the ACI Standard Device Configuration File (ACNF), ACI Standard Device Response File (ARSP), and Response Code Description File (RCDF). It is the customer's responsibility to configure each response in the terminal configuration data to include sufficient data for the terminal to format and print a receipt. The SPDH module has the capability of returning a response determined by the customer in up to three different languages as well as returning a 48-character response. This is set up in the ARSP and the RCDF.

## Download Options

The SPDH module supports both full and partial downloads. The terminal can receive full configuration file information or single records. Whether a full or partial download is requested is determined at the terminal level. Download data is flexible, allowing customers the ability to send more than 1,000 bytes of discretionary processing data to the terminal. In addition, the download data can pertain to as many terminals as the customer selects. A separate download record is **not** required for each terminal.

Data elements included in downloads to POS devices are identified by download field identifiers (DIDs) in the ACI standard POS message.

## Draft Capture Options

The SPDH module provides an efficient method of draft capture with third-generation terminals by performing authorization and draft capture in a single transaction. If a transaction is not completed correctly, the operator who entered the transaction sends an adjustment transaction.

The SPDH module also supports transactions that require paper follow-up. This method consists of authorizing a transaction online and then sending a paper voucher/draft follow-up. This method requires merchants to submit the paper voucher/draft follow-up in order to receive payment.

## American Express Data Collection

The SPDH module provides customers with the option of performing data capture on transactions originating from American Express cardholders. Generally, American Express has defined categories into which transactions being processed are placed. For each of these categories, American Express requires different data to be sent from the device. To ensure the data they require is captured by the device and sent to BASE24, American Express has set forth standard industry formats for each category. In order to perform the collection of data on transactions originating from an American Express cardholder, customers must use a Device Handler flexible enough to accept and recognize the American Express standard industry formats. The SPDH module is able to recognize these standard industry formats and, subsequently, capture and process transactions using them.

For the American Express standard industry formats, refer to the ***ACI Standard POS Device Message Specifications Manual***.

## BASE24-pos Settlement and Cutover

BASE24-pos provides an efficient method of settlement and cutover. *Settlement*, in terms of BASE24-pos, is actually a reconciliation effort whereby the transaction activity for the current day is closed out and reported, and the system is shown to be in balance. The term *cutover* refers to the point in time where transactions stop being logged to the current day's business and start being logged to the next day's business. It refers to the closing of the current posting day and the opening of the next.

BASE24-pos settlement differs from the settlement carried out by participating BASE24-pos financial entities (i.e., retailers, retailer sponsors, cardholders, card issuers, interchanges) in that BASE24-pos does not move funds to settle accounts. It does, however, provide the support that allows these financial entities to initiate the movement of funds.

An extensive set of reports provides financial entities with the information they require to settle their accounts. BASE24-pos also provides the POS Transaction Log File (PTLF), which keeps a record of each transaction processed by BASE24-pos during its defined reporting day. In addition, BASE24-pos provides an automated method to cut over terminals, retailers, institutions, interchanges, and the BASE24-pos network to a new posting day. For more information on BASE24-pos settlement, cutover, and reporting, refer to the ***BASE24-pos Settlement and Reporting Manual***.

## Terminal Balancing

The SPDH module enables merchants to balance their terminals with BASE24-pos. The SPDH module retains terminal totals and enables the retailer to request totals information from BASE24-pos in order to balance their totals and receipts against BASE24-pos.

The SPDH module maintains totals for each terminal at a merchant site. Additionally, the SPDH module can maintain a set of totals that combines all of the terminals at a merchant site. Terminals in a department store or a pay-at-the-pump service station are examples of site configurations.

The SPDH module provides at least three methods for merchants to balance terminals with BASE24-pos. Merchants have the option of performing subtotal request transactions to determine if BASE24-pos and the terminal have accumulated the same totals. Merchants then have the option of entering adjustment transactions, if necessary, before closing the batch.

In addition, merchants also can receive a report a day later, or contact the service provider to receive terminal balancing information. The BASE24-pos report that contains terminal balancing information is the Retailer Reconciliation Report (B24POS35). This report is used to balance transaction activity at the terminal batch level and can be used by customers to determine whether their terminals balance with BASE24-pos.

Another method allows merchants to balance terminal and BASE24-pos totals after each transaction is performed. This method can be accomplished by configuring the ACI Standard Device Configuration File (ACNF) to return subtotals with each response message.

## BASE24 Transaction Security

The SPDH module uses the Transaction Security Services process to offer various ways of ensuring that transactions are transmitted and secured correctly. These include options that ensure every transaction is being processed only once and options that allow customers to select from various PIN encryption methods.

The SPDH module supports the use of message authentication codes (MACs). MACs are used to verify whether data has been accidentally or fraudulently altered. Any data sent between the terminal and the SPDH module can be verified through the use of the MACs. The details necessary to use this standard have been developed by ACI to use with BASE24-pos and the SPDH module. The customer can decide whether to use MAC verification with a particular transaction by configuring MAC parameters in the ACNF.

The SPDH module uses the Transaction Security Services (TSS) process to provide data encryption for FID J (Available Balance). TSS is used for full message encryption and configurable message encryption.

The SPDH module supports dynamic key management, which is the automatic replacement of working keys for a terminal based on one or more thresholds, for the following working keys:

- The PIN communications key (KPE), which is returned in FID M.
- The MAC communications key (KMAC), which is returned in FID H.
- The data encryption communications key (KME), which is returned in FID I.

The transaction acquirer can determine which PIN encryption, message authentication, and data encryption keys are to be used in a transaction. The SPDH module includes key indexes in requests to the Transaction Security Services process, and can return the indexes to interchanges that require them.

The ACI standard POS message supports full message encryption and configurable message encryption. Full message encryption allows the institution to encrypt all optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. Configurable message encryption allows the institution to encrypt specific optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. The

specific optional data fields are configured to be encrypted in the ACI Standard Device Configuration File (ACNF) request and response field maps. Full message encryption and configurable message encryption are enabled using the DATA ENCRYPTION TYPE field in the POS Terminal Data File.

Transaction security features are described in greater detail in section 6 and setting up the options associated with these features is explained in section 8. For more information on BASE24 transaction security, refer to the ***BASE24 Transaction Security Manual*** and the ***BASE24 Transaction Security Services Processing Guide.***

## Derived Unique Key per Transaction (DUKPT) Support

The SPDH module provides support for POS devices that derive a unique key for each transaction processed. Each unique key is derived from a base derivation key loaded into the terminal's security module (TSM) when it is initialized. The same base derivation key is loaded into a database at the host. When a POS device derives a unique key for each transaction, the PIN block and MAC in the request message are encrypted under the derived key and the request message includes a key serial number field in the message. The key serial number consists of a host derivation key identifier, a host derivation key subordinate identifier, the TSM identifier, and a transaction counter. When the host receives a PIN block or MAC encrypted under a derived key, it uses the key serial number passed in the message and the terminal's base derivation key stored in the host database to derive the same unique transaction key, decrypt the PIN block, and verify the PIN or MAC.

## American Express Card Security Codes (CSCs)

The ACI standard POS message contains data that enables the SPDH module to verify American Express card security codes. The three types of card security codes (CSCs) are as follows:

- Three-digit CSC located on the signature panel
- Four-digit CSC located on the front of the card
- Five-digit CSC located on the magnetic stripe

For more information about CSC processing performed by the SPDH module, refer to section 6.

## Message Sequencing

The SPDH module supports message sequencing, which is a method of ensuring that every message being sent from the terminal is being received only once and in the correct order. There are two options available for message sequencing. One option calls for the SPDH module to check the transmission number, which is a field in the ACI standard POS message header, in order to detect and drop duplicate requests. The other option consists of comparing message sequence numbers to positively identify the sequencing of a request. If message sequence numbers are included in the message, the SPDH module can ensure that no messages have been lost or transmitted out of sequence. The message sequence number method is more reliable, but requires an additional sequence number field to be included in the message itself. For more information on the sequence number field, refer to section 4.

## Event Message Generation

The SPDH module generates event messages that keep operators aware of activity occurring in the system. Event messages can be used to assist operators with their daily tasks or to notify operators of problems occurring in the system. Event messages provide operators with insight as to how the system is running.

In addition, ACI has developed a method that allows customers to view event message documentation electronically. This online event message documentation provides customers with greater flexibility by allowing them to print out event message documentation as needed at their location or to electronically manipulate the data to suit their needs. The online event message documentation provides the message number, severity, and text for each event message generated by the SPDH module. In addition, it also provides the cause of each message and the recommended action to take.

Accessing the documentation for event messages generated by the SPDH module is discussed in section 4. For a complete explanation of online event message documentation, refer to the ***BASE24 Event Message Reference Manual***.

## BASE24-mail Support

The SPDH module is fully compatible with BASE24-mail release 6.0. BASE24-mail allows host processors to send informational messages to POS terminals throughout a network. Mail messages can also be transmitted from POS terminals to the host processor. BASE24-mail uses the technology of the

HP NonStop computer to provide reliable, fault-tolerant service 24 hours a day, seven days a week. Integration of the SPDH module with BASE24-mail is discussed in section 4.

## PS2000 Support

The SPDH module is compatible with Visa Payment Service 2000 (PS2000). PS2000 is a Visa program that ensures that the portion of the transaction that is authorized is the same as the portion that is cleared and settled. All phases of the transaction are linked, allowing the institution to reduce back office expenses. The SPDH module supports the PS2000 program for the passenger transport, hotel, automobile rental, direct marketing, and fuel market segments.

## Track 1 and Track 2 Support

The SPDH module formats both Track 1 and Track 2 information from the native message into the BASE24-pos Standard Transaction Message (PSTM). The SPDH module fills the Track 2 Data field of the PSTM regardless of the type of track data available in the native message. When Track 1 data is available, the SPDH module adds the Track 1 token to the PSTM. For more information on how the SPDH module formats track data, see appendix A.

## Address Verification

The SPDH module is compatible with the BASE24-pos add-on Address Verification module. Address verification assists merchants in controlling losses that can result from credit card fraud during transactions where the cardholder cannot be readily identified. The SPDH module places address verification information in the PSTM. See section 4 for information about the data fields used to format address verification data in the PSTM. For more information on the BASE24-pos add-on Address Verification module, refer to the ***BASE24-pos Address Verification Manual.***

## Port Usage Tracking

The SPDH module can track station or port usage. Port usage tracking gives an institution the ability to analyze, control, and balance network service usage. The SPDH module retrieves the station ID from the network message header and

places it in the Station ID token. The token is appended to the PSTM and can be configured to be logged to the POS Transaction Log File (PTLF). Refer to the ***BASE24 Tokens Manual*** for more information on the Station ID token.

## Maximum Returns and Adjustments

The SPDH module can be configured to prevent a clerk from accidentally exceeding the return and adjustment limits defined in the PTDS1. When a return or adjustment limit is exceeded, the SPDH module can decline the transaction, refer the transaction, or continue processing based on an LCONF param. For more information about checking return and adjustment limits, refer to section 3.

## Stored Value Cards

The SPDH module provides the following transactions for support of stored value cards:

- Additional Card Activation
- Card Activation
- Full Redemption
- Replenishment

## Balance Inquiry Service

The SPDH module can be configured to return balances on Visa Prepaid purchase requests and balance inquiry requests. The purchase request function returns balance information with purchase, purchase with cash back, and preauthorization purchase transactions. The balance inquiry function enables cardholders to request account balance or available credit amounts prior to initiating a purchase transaction.

## Electronic Check Authorization

This add-on product provides POS issuers (or drawee banks) with the ability to verify funds availability for check verification and check guarantee transactions based on a transit routing and account number rather than a card number. The customer's account balance is impacted (reduced) for check guarantee transactions. This functionality allows retailers to authorize checks electronically.

## Electronic Benefits Transfer

This add-on product provides POS acquirers with the ability to acquire and route EBT transactions to an EBT processor or gateway service provider. This application supports PIN-based transactions and two additional account types: food stamps and cash benefit.

## Europay, MasterCard, and Visa (EMV) Transaction Support

The BASE24-pos EMV add-on product allows the SPDH to process the following transactions entered at a POS device using a Europay, MasterCard, and Visa (EMV) card:

- Normal Purchase
- Preauthorization Purchase
- Preauthorization Purchase Completion
- Mail or Telephone Order
- Merchandise Return
- Cash Advance
- Card Verification
- Balance Inquiry
- Offline Log-only
- Purchase with Cash Back
- Purchase Adjustment
- Merchandise Return Adjustment
- Cash Advance Adjustment
- Cash Back Adjustment

The SPDH module supports the FIDs, subFIDs, and tokens used to process these transactions. This manual contains detailed information on FIDs and subFIDs. The ***BASE24-pos EMV Support Manual*** describes the SPDH module configuration and processing requirements for EMV transactions. The ***BASE24 Tokens Manual*** describes the tokens used to process these transactions.

## Multiple Currency Support

The BASE24-pos Multiple Currency add-on product allows the SPDH module to process transactions in a multiple currency environment where accounts can be specified in different currencies and the currency used in POS devices can vary from one device to another. The following transactions are supported for multiple currency processing:

- Normal Purchase
- Preauthorization Purchase
- Preauthorization Purchase Completion
- Mail or Telephone Order
- Merchandise Return
- Cash Advance
- Card Verification
- Purchase with Cash Back
- Check Verification
- Check Guarantee
- Purchase Adjustment
- Merchandise Return Adjustment
- Cash Advance Adjustment
- Cash Back Adjustment

The SPDH module supports the FIDs and tokens used to process these transactions in a multiple currency environment. This manual contains detailed information on FIDs. The ***BASE24-pos Multiple Currency Support Manual*** describes the SPDH module configuration and processing requirements for multiple currency transactions. The ***BASE24 Tokens Manual*** describes the tokens used to process these transactions.

# Contactless Transactions

The SPDH module allows for identifying contactless transactions from terminals. Contactless transactions are transactions initiated without physical contact between the card and terminal. They can include contactless magnetic stripe transactions and contactless chip card transactions.

Contactless transaction are identified by a value of 07 (contactless chip card transaction) or 91 (contactless magnetic stripe transaction) in the FID 6 subFID E (POS Entry Mode) field in the transaction request from the terminal.

The SPDH moves the contents from FID 6 subFID E (POS Entry Mode) into the PT-SRV-ENTRY-MDE field in the 0200 request PSTM.

## Terminal Requirements

To identify a transaction as contactless, the terminal must include a value of 07 (contactless chip card transaction) or 91 (contactless magnetic stripe transaction) in the FID 6 subFID E (POS Entry Mode) field in the transaction request.

EMV terminal capabilities are determined from Processing Flag 2 field in the standard message header. If the terminal is capable of contactless EMV transactions, the Processing Flag 2 field must be set to 5 also.

Note: Contactless transaction processing within BASE24-pos is an add-on product. Without this add-on product, contact processing will not take place. For a description of this processing, refer to the ***BASE24-pos Transaction Processing Manual***.

## Configuration Requirements

The terminal's ACNF response field map record must be configured to include FID 6 subFID E (POS Entry Mode) in requests from the terminal.

The terminal's input capabilities need to be set to 3 (contactless chip - EMV) or 4 (contactless magnetic stripe) in the TERMINAL INPUT CAPABILITIES field on PTD screen 19.

# Dynamic Card Verification

The SPDH module is compatible with BASE24 Dynamic Card Verification, a type of card verification performed by BASE24 on contactless magnetic stripe transactions. BASE24 Dynamic Card Verification supports Mastercard's CVC3 card verification value (used for PayPass card verification) and Visa's dCVV card verification value (used for Visa Wave card verification).

**Note:** Dynamic Card Verification is a BASE24 add-on product that is used exclusively for contactless magnetic stripe transactions. It cannot be used with contactless chip card transactions, which are handled through EMV verification. Dynamic Card Verification is also the only type of BASE24 card verification supported for contactless magnetic stripe transactions. For more information on Dynamic Card Verification, refer to the ***BASE24 Transaction Security Services Processing Guide***.

## Terminal Requirements

For contactless magnetic stripe transactions, the applicable card verification values for Dynamic Card Verification are included in the Issuer Discretionary Data fields of the Track 1 and Track 2 Data. To support Dynamic Card Verification, the terminal must include the following in the track information it sends:

- Card verification value (the CVC3 or dCVV)
- Application Transaction Counter (ATC) – a transaction counter maintained by the application on the integrated chip card.
- Unpredictable Number (UN) – a number generated by the terminal and used to create uniqueness in the transaction (required by MasterCard only).

In addition, the track location and length of this card verification information must match that defined to BASE24 in the Card Prefix File.

# Healthcare/Transit Auto-Substantiation Transactions

The SPDH module supports healthcare/transit auto-substantiation transactions. Auto-Substantiation is the process of verifying that purchase transactions are for expenses permitted under Internal Revenue Service regulations for Flexible Spending Accounts (FSAs) and Healthcare Reimbursement Arrangements (HRAs). Healthcare auto-substantiation transactions enable employers and their

third-party healthcare service providers to approve qualified medical expenses at the point-of-sale for purchases made with FSA and HRA payment cards at participating retailers.

Additionally, participating retailers that sell transit fare media, such as commuter passes, parking passes, and mass transit vouchers and tickets, can also use auto-substantiation transactions to approve purchases made with FSA payment cards.

Partial authorizations are supported for auto-substantiation transactions. Similar to other prepaid or pre-funded cards, it is difficult for consumers to spend the exact amount available in FSA or HRA accounts. Partial authorizations allow consumers to make purchases that exceed the account balance, with the remainder of the purchase paid by other means (credit card, cash, etc.)

Healthcare/transit auto-substantiation transactions are handled as purchases. They are identified as healthcare/transit auto-substantiation transactions by the presence of FID 6 subFID l (Healthcare/Transit Data).

The SPDH accepts FID 6 subFID l (Healthcare/Transit Data) in transaction requests from the terminal, moves the information to the Healthcare/Transit token (CV), and adds the token to the 0200 request PSTM.

The SPDH returns subFID l to the terminal when the 0210 response PSTM contains the Healthcare/Transit token (CV) and when auto-substantiation data is present in the token.

## Terminal Requirements

The terminal must set the following fields when formatting a healthcare/transit auto-substantiation transaction request.

| | |
|---|---|
| FID B (Amount 1) | Must be set to the full transaction amount. |
| FID 1 (PS2000 Data) | The Market Specific Data ID field must be set to M for healthcare (medical) or T for transit. |

| | |
|---|---|
| FID 6 subFID l (Healthcare/Transit Data) | Fields in the first Additional Amount entry must be set as follows:<br>• Account Type – Set to 00 (unspecified).<br>• Amount Type – Set to 4S (amount healthcare) or 4T (amount transit).<br>• Currency Code – Set to the ISO currency code of the amount.<br>• Amount Sign – Set to C (credit, positive balance).<br>• Amount – Set to the amount of the qualified healthcare or transit purchase (right-justified and zero-filled on the left).<br>Although the Additional Amount table can contain from 1 to 6 entries, only entry 1 should be provided in the request. Entries 2 through 6 should not be included. |

The terminal will receive FID B (Amount 1) and FID 6 subFID l (Healthcare/ Transit Data) in the healthcare/transit auto-substantiation transaction response. If the issuer approves the healthcare/transit auto-substantiation transaction for the full amount, FID B will contain the original requested amount.

If the Issuer approves the healthcare/transit auto-substantiation transaction for a partial amount (partial authorization), the response fields will be set as follows:

| | |
|---|---|
| FID B (Amount 1) | Set to the approved partial amount (less than the original requested amount). |
| FID 6 subFID l (Healthcare/Transit Data) | Fields in the first Additional Amount table entry will be set as follows:<br>• Amount Type – Set to 57 (original amount).<br>• Amount – Set to the original requested amount.<br>Entries 2 through 6 of the Additional Amount table will not be included in the response. |

Entries 2 through 6 of the Additional Amount table in FID 6 subFID l (Healthcare/ Transit Data) will not be included in the response.

### Configuration Requirements

The terminal's ACNF response field map record should be configured so that FID B (Amount 1) and FID 6 (Product subFIDs) are returned to the terminal in transaction responses.

## Healthcare Eligibility Inquiry Transactions

The SPDH module supports healthcare eligibility inquiry transactions, which allow healthcare providers (doctors, dentists, hospitals, etc.) to immediately determine the healthcare insurance coverage of FSA or HRA payment card holders.

Healthcare eligibility inquiry transactions are handled as balance inquiries. They are identified as healthcare eligibility inquiry transactions by the presence of FID 6 subFID m (Healthcare Service Data) in the terminal request.

The SPDH moves the healthcare service data received in subFID m to the Healthcare Service token (CW) and adds the token to the 0200 request PSTM. Similarly, the SPDH returns subFID m to the terminal when the 0210 response PSTM contains the Healthcare Service token (CW) and when healthcare service data is present in the token.

### Terminal Requirements

The terminal must set the following fields when formatting a healthcare eligibility inquiry transaction request.

| | |
|---|---|
| Transaction Code (Message Header) | Must be set to 07 (balance inquiry). |
| FID 6 subFID U (Transaction Subtype Data) | The Transaction Subtype field must be set to C002 (healthcare eligibility inquiry).<br>The Acquirer Processing Code and Issuer Processing Code fields must be set to blanks. |

| | |
|---|---|
| FID 6 subFID m (Healthcare Service Data) | The Provider ID must be set to the medical license number of the healthcare provider.<br>The Type Code must be set to the HIPAA code for the healthcare service.<br>The rest of the subfield must be set to blanks. |

The terminal may request eligibility information for up to five healthcare services within a single request message. Each entry in the subFID m Service table represents a single healthcare service.

The terminal will receive FID 6 subFID l (Healthcare/Transit Data) and FID 6 subFID m (Healthcare Service Data) in the healthcare eligibility inquiry transaction response. SubFID l will contain from 1 to 6 entries in the Additional Amount table.

The Amount Type field in each entry will contain a value of 3S (amount co-payment). SubFID m will contain from 1 to 5 entries in the Service table.

### Configuration Requirements

The terminal's ACNF response field map record should be configured so that FID 6 (Product subFIDs) is returned to the terminal in transaction responses.

## Visa Card Level Results

The SPDH module can pass Visa card level results to the terminal if they are present in transaction responses. Visa card level results are codes identifying the participation (rewards) programs in which the card involved in a transaction is enrolled (e.g., Visa Traditional, Visa Traditional Rewards, Visa Signature, or Visa Infinite).

Card level results codes are carried in the Card Level Product ID field in the Switch Common Data token (BY). If the 0210 response PSTM contains the Switch Common Data token and the token's Card Level Product ID field is not blank, the SPDH will move the information to FID 6 subFID k (Card Level Results) and send it to the terminal.

## Configuration Requirements

To receive card level results, the terminal's ACNF response field map record must be configured so that FID 6 (Product subFIDs) is returned to the terminal in transaction responses.

# SPDH Transaction Support

All of the transactions supported by the SPDH module must originate with a request from the terminal. A response is sent from the SPDH module for each request received unless the SPDH module is unable to read the POS Terminal Data Dynamic File—general data (PTDD1) and POS Terminal Data Static File—general data (PTDS1) records that correspond to the terminal sending the request. In this case, no response is sent and the terminal times out. Message requests and responses are explained in more detail later.

The SPDH module supports both financial and administrative transactions. Financial transactions are monetary transactions performed on behalf of a cardholder, and adjustments to these transactions. Administrative transactions are monetary and non-monetary transactions performed in support of financial transactions, typically by supervisory personnel. A complete list of transactions supported by the SPDH module is shown in the following tables.

## Financial Transactions

The following table describes the financial transactions supported by the SPDH module.

| Financial Transactions | |
|---|---|
| **Transaction** | **Description** |
| Additional Card Activation | Enables customers to purchase an additional stored value card that is associated with an existing stored value card account. |
| Balance Inquiry | Online inquiry into the balance of a customer's account. This transaction can be initiated using a credit or a debit card. Both magnetic stripes and ICCs are supported for this transaction. |
| Card Activation | Activates a stored value card account when a customer purchases a stored value card. |
| Card Verification | Online check to see if a customer's credit card is valid. This transaction can be initiated using a credit card. Both magnetic stripes and ICCs are supported for this transaction. |

| Financial Transactions | |
|---|---|
| **Transaction** | **Description** |
| Cash Advance | Cash disbursement from a customer's account at the point of sale. This transaction can be initiated using a credit or a debit card. Both magnetic stripes and ICCs are supported for this transaction. |
| Cash Advance Adjustment | Corrects an error in a previously-completed cash advance transaction. Both magnetic stripes and ICCs are supported for this transaction. |
| Cash Back Adjustment | Corrects an error in a previously-completed purchase with cash back transaction. Adjustments can be made only to the cash back amount. Both magnetic stripes and ICCs are supported for this transaction. |
| Check Guarantee | Electronically reserves funds in a customer's checking account to cover a check written by the customer. BASE24-pos supports this transaction in pass through mode. This means BASE24-pos provides a link between the transaction and destination by simply passing the transaction to the destination without performing authorization.<br>**Note:** When using electronic check authorization, the transaction is authorized and the cardholder account is impacted. |
| Check Verification | Online check verification or authorization to see if a customer's check is tied to a valid checking account. BASE24-pos supports this transaction in pass through mode. This means BASE24-pos provides a link between the transaction and destination by simply passing the transaction to the destination without performing authorization. |
| Full Redemption | Enables customers to redeem the remaining value from a stored value card. This transaction reduces the balance of the stored value card account to zero. |

| Financial Transactions | |
|---|---|
| **Transaction** | **Description** |
| Mail or Telephone Order | Purchase of goods or services where the cardholder is not present at the point of sale (e.g., through the mail or by telephone). This transaction can be initiated using a credit or a debit card. Both magnetic stripes and ICCs are supported for this transaction. |
| Merchandise Return | Return of merchandise to a retailer for refund. The refund is electronically credited to the customer account. This transaction can be initiated using a credit or a debit card. Both magnetic stripes and ICCs are supported for this transaction. |
| Merchandise Return Adjustment | Corrects an error in a previously-completed merchandise return transaction. Both magnetic stripes and ICCs are supported for this transaction. |
| Normal Purchase | Purchase of goods or services at the point of sale using a credit or a debit card. Both magnetic stripes and ICCs are supported for this transaction. |
| Preauthorization Purchase | Online check of a customer's account to determine if funds are available for an intended purchase. The amount preauthorized is placed on hold in the customer's account until the purchase is completed, a timer expires, or the hold is removed. This transaction can be initiated using a credit or a debit card. Both magnetic stripes and ICCs are supported for this transaction. |
| Preauthorization Purchase Completion | Completion of a purchase that was preauthorized. The actual amount of the purchase is entered using this transaction and the hold for the amount entered in the preauthorization purchase transaction is released. The transaction is then completed using the correct amount. This transaction can be initiated using a credit or a debit card. Both magnetic stripes and ICCs are supported for this transaction. |

| Financial Transactions | |
|---|---|
| **Transaction** | **Description** |
| Purchase Adjustment | Corrects an error in a previously-completed purchase transaction. Both magnetic stripes and ICCs are supported for this transaction. |
| Purchase with Cash Back | Combination purchase and cash withdrawal at the point of sale. The requested transaction amount is higher than the price of the purchase. The customer receives the difference in cash. This transaction is used only with debit cards. Both magnetic stripes and ICCs are supported for this transaction. |
| Replenishment | Enables customers to add funds to stored value card. This transaction increases the balance of a stored value card account by the replenishment amount. |

# Administrative Transactions

The following table describes the administrative transactions supported by the SPDH module.

| Administrative Transactions | |
|---|---|
| **Transaction** | **Description** |
| Batch Subtotals | Obtains the count and amount of draft capture transactions that have been entered at a point-of-sale terminal since the last close batch request transaction or since the totals were last cleared from the PTDD1.<br><br>These totals include only draft capture transaction activity unless a card type is set up for authorization only processing, in which case the totals contain authorization only totals. However, the system uses only draft capture totals for terminal reconciliation. Authorization only activity is not included in terminal reconciliation.<br><br>This transaction can be used when balancing a terminal to determine whether the system totals are in agreement with the actual totals at the terminal. This transaction can be entered at any time, is not logged, and does not affect PTDD1 totals. |

| Administrative Transactions | |
|---|---|
| **Transaction** | **Description** |
| Close Batch | Ends the current batch and opens a new batch to which data is logged and accumulated. When a close batch transaction is received, the system creates a batch totals record from the terminal totals in the PTDD1 and writes it to the PTLF, if batch totals are configured to be logged. It then clears the batch totals in the PTDD1.<br>If the terminal can send its own batch totals, the system also compares the terminal totals to the PTDD1 totals, and, if the debits, credits, and adjustments do not match exactly, writes the terminal totals to the PTLF immediately following the PTDD1 totals, if terminal totals are configured to be logged.<br>**Note:** When a batch is closed, the system automatically writes a service totals record to the PTLF (prior to the batch totals record), if service totals are configured to be logged, and clears the service totals in the PTDD1.<br>The system also automatically writes a clerk totals record to the PTLF (prior to a batch totals record), if clerk totals are configured to be logged, and clears the clerk totals in the PTDD1. |

| Administrative Transactions | |
|---|---|
| **Transaction** | **Description** |
| Close Day | Ends the current terminal day and begins a new day. When a close day transaction is received, the system creates a daily totals record from the terminal totals in the PTDD1 and writes it to the PTLF, if terminal totals are configured to be logged. It then clears the terminal daily totals in the PTDD1 that have accumulated since the last close day transaction or since the totals were last cleared.<br><br>If the terminal can send its own daily totals, the system also compares the terminal totals to the PTDD1 totals, and, if the debits, credits, and adjustments do not match exactly, writes the terminal totals to the PTLF, if daily totals are configured to be logged, immediately following the PTDD1 totals.<br><br>**Note:** Close day transactions, when entered, must be immediately preceded by a close shift transaction. If not, the system denies the close day transaction. Close day transactions should be entered only once during a reporting day in order to maintain reporting integrity. |

| Administrative Transactions | |
|---|---|
| **Transaction** | **Description** |
| Close Shift | Ends the current shift and opens a new shift to which data is logged and accumulated. When a close shift transaction is received, the system creates a shift totals record from the terminal totals in the PTDD1 and writes it to the PTLF, if shift totals are configured to be logged. It then clears the shift totals in the PTDD1.<br>If the terminal can send its own shift totals, the system also compares the terminal totals to the PTDD1 totals, and, if the debits, credits, and adjustments do not match exactly, writes the terminal totals to the PTLF immediately following the PTDD1 totals, if terminal totals are configured to be logged.<br>**Note:** Close shift transactions, when entered, must be immediately preceded by a close batch transaction. If not, the system denies the close shift transaction. |
| Day Subtotals | Obtains the count and amount of draft capture transactions that have been entered at a point-of-sale terminal since the last close day request transaction or since the totals were cleared.<br>These totals include only draft capture transaction activity unless a card type is set up for authorization only processing, in which case the totals contain authorization only totals. However, the system uses only draft capture totals for terminal reconciliation. Authorization only activity is not included in terminal reconciliation.<br>This transaction can be used when balancing a terminal to determine whether the system totals are in agreement with the actual totals at the terminal. This transaction can be entered at any time, is not logged, and does not affect PTDD1 totals. |
| Download | Causes a full- or a partial-download to be sent to the terminal. |

| Administrative Transactions | |
|---|---|
| **Transaction** | **Description** |
| Employee Subtotals | Obtains the count and amount of transactions that have been entered at a point-of-sale terminal by a specific employee since the totals were last cleared from the PTDD1. This includes clerk check amounts and counts. Employee Subtotals Request also indicates which totals requests to return in response to a clerk totals request.<br><br>This transaction requests PTDD1 clerk totals. It can be used when balancing a terminal to determine whether the system totals are in agreement with the actual totals at the terminal. This transaction can be entered at any time, is not logged, and does not affect PTDD1 totals. |
| Handshake | Verifies the status of the communications link and, optionally, the terminal communication and authentication keys. |
| Logoff | Clears the clerk ID in the PTDD1. |
| Logon | Changes the clerk ID in the POS Terminal Data Dynamic File—general data (PTDD1) and accumulates totals for the new employee. It also automatically performs an implied logoff of the previous employee, if required. |
| Mail Delivered | Notifies the Mail process that mail in a read mail request was delivered to the terminal. |
| Read Mail | Obtains a mail message intended for the point-of-sale terminal. |
| Send Mail | Sends a mail message from the point-of-sale terminal to another destination. |

| Administrative Transactions | |
|---|---|
| **Transaction** | **Description** |
| Shift Subtotals | Obtains the count and amount of draft capture transactions that have been entered at a point-of-sale terminal since the last close shift request transaction or since the totals were last cleared.<br>These totals include only draft capture transaction activity unless a card type is set up for authorization only processing, in which case the totals contain authorization only totals. However, the system uses only draft capture totals for terminal reconciliation. Authorization only activity is not included in terminal reconciliation.<br>This transaction can be used when balancing a terminal to determine whether the system totals are in agreement with the actual totals at the terminal. This transaction can be entered at any time, is not logged, and does not affect PTDD1 totals. |

# SPDH File Usage

The following table provides a comprehensive list of all files used by the SPDH module. The SPDH module uses both Base files and BASE24-pos files and, therefore, the table indicates whether each file is a Base file or a BASE24-pos file.

| Files Used by the SPDH Module | |
|---|---|
| **File Name** | **File Type** |
| ACI Standard Device Configuration File (ACNF) | BASE24-pos |
| ACI Standard Device Response File (ARSP) | BASE24-pos |
| Acquirer Processing Code File (APCF) | Base |
| Institution Definition File (IDF) | Base |
| Logical Network Configuration File (LCONF) | Base |
| Log Message Configuration File (LMCF) | Base |
| POS Retailer Definition File (PRDF) | BASE24-pos |
| BASE24-pos Terminal Data files (PTD) | BASE24-pos |
| POS Transaction Log File (PTLF) | BASE24-pos |
| Response Code Description File (RCDF) | BASE24-pos |

# Integration With BASE24-pos

The SPDH module can be integrated with BASE24-pos. BASE24-pos is a complete point-of-sale authorization and switching system that uses the technology of the HP NonStop computer to provide reliable, fault-tolerant service 24 hours a day, seven days a week.

BASE24-pos offers routing of transactions based on card type and retailer floor and ceiling limits. If the primary destination for authorization is not available, BASE24-pos allows the transaction to be sent to up to two additional authorization centers before a decision is made about approving, denying, or referring the transaction. Destinations to which transactions can be routed include host computers and interchange networks.

In addition, BASE24-pos offers a comprehensive set of authorization options on the HP NonStop computer. These include local and national negative identification, positive identification, and positive identification with account balances. Another authorization option for credit card accounts, parametric authorization, enables institutions to define additional criteria on which to base the authorization of transaction requests that exceed standard cardholder limits.

BASE24-pos also includes online control of all network communications elements and POS terminals, as well as a comprehensive set of daily and periodic reports for service providers and individual retailers.

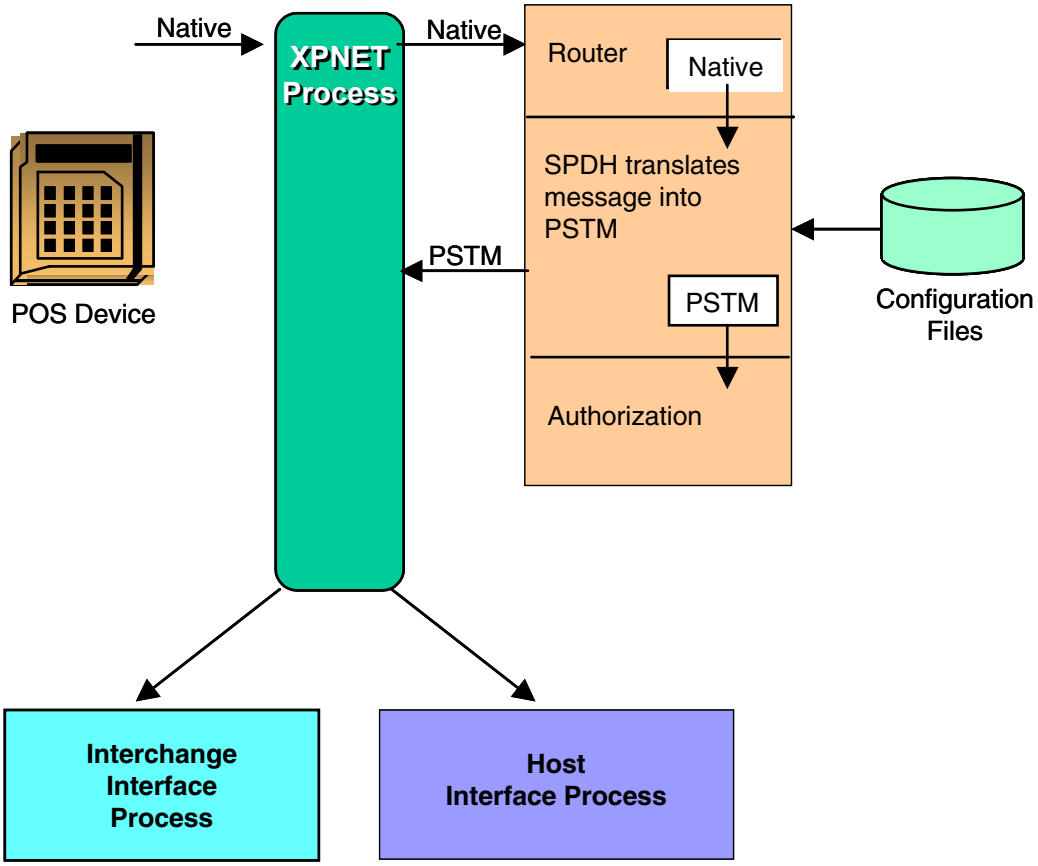
## Transaction Flow

When a POS terminal sends a message to BASE24, it is first received by the XPNET process, the transaction manager, which then sends the message to the bound Device Handler/Router/Authorization process.

The SPDH module is bound into a Router/Authorization process. Any messages coming into the bound process from a POS device are first received by the Router module which then routes the message to the Device Handler module. Messages coming into the bound process from any other destination are also received first by the Router module, but are routed instead to the Authorization module to be logged to the POS Transaction Log File (PTLF).

# Tying the SPDH Module to BASE24-pos

The SPDH module uses terminal configuration files within the BASE24 database to determine the format and content of messages coming from and going to the POS device. This information allows the SPDH module to translate messages from the native language used by the POS device into the BASE24-pos Standard Internal Message (PSTM), which is understood by all the other BASE24-pos processes. When BASE24-pos is sending a message to the POS device, the SPDH module translates the message from the PSTM format to the native language used by the POS device. The following diagram illustrates the SPDH module link. Note that the SPDH module is bound with a BASE24-pos Router/Authorization module.



# SPDH Releases and Versions

The SPDH module is available in two releases: 5.3 and 6.0. These releases of the SPDH module correspond to release 5.3 and release 6.0 of BASE24-pos software. Customers should note that release 6.0 of the SPDH module is designed to accommodate terminals that have firmware designed for release 6.0 of the SPDH module.

Release 6.0 of the SPDH module supports the release 6.0 native message format. While release 6.0 of the SPDH module is documented in this manual, release 5.3 of the SPDH module is documented in the ***BASE24-pos Standard POS Device Support Manual***.

# Implementation Responsibilities

Implementing the SPDH module requires the cooperation of three separate parties in the POS environment. These three parties are ACI, the terminal vendor, and the terminal owner, who is referred to as the customer. Each of these parties has certain responsibilities when implementing the SPDH module. However, the customer is primarily responsible for selecting the options available with the SPDH module that are applicable to the particular POS environment in which the terminals will be operating. Responsibilities are outlined below.

| Responsible Party | Task |
|---|---|
| Customer/ACI | License the BASE24 Standard POS Device Handler (SPDH) module. The customer contracts with ACI to license the SPDH module software. |
| Customer/ACI | Receive the ***BASE24-pos Standard POS Device Configuration Manual*** and ***the ACI Standard POS Device Message Specifications Manual***. The customer orders and receives the ***BASE24-pos Standard POS Device Configuration Manual*** and the ***ACI Standard POS Device Message Specifications Manual*** from ACI. |
| Customer | Review functionality of the SPDH module. The customer reads the manuals and understands the SPDH module options. |
| Customer | Develop the specification. The customer outlines the features that will be supported by the customer's BASE24-pos Standard POS devices, as follows:<br>• Include transaction set.<br>• Include message formats for requests and responses.<br>• Include standard header values and a mechanism for changing values.<br>• Include PIN encryption requirements.<br>• Determine fields to be downloaded.<br>• Develop operator procedures for completing downloads. |

| Responsible Party | Task |
|---|---|
| Customer/ACI | Determine protocol requirements. The customer must decide the type of line to be used in communication with the terminal and whether the line will be leased, dial, or network. |
| Customer/ACI | Review the specification with ACI. ACI reviews the specification developed by the customer to verify that the configuration decisions are complete and that the functionality desired is feasible. |
| Customer | Choose terminal vendor. The customer chooses a third-generation terminal vendor who meets the customer's requirements and who can develop a terminal that meets ACI's specifications. |
| Customer/ACI/Vendor | The customer verifies that the vendor has a current copy of the ***ACI Standard POS Device Message Specifications Manual***. |
| Customer/Vendor | Develop operational procedures and guidelines for the terminal, as follows:<br>• Determine operational functions for the terminal.<br>• Determine terminal screen prompt sequence.<br>• Determine keyboard configuration.<br>• Determine printer functions.<br>• Determine PIN pad functions.<br>• Determine card reader functions. |
| Customer | Develop draft capture procedures, as follows:<br>• Include storage capacity of terminal.<br>• Consider batch, shift, and end-of-day processing.<br>• Include mechanism for storing paper drafts and dealing with dispute claims.<br>• Determine draft capture profile in BASE24-pos. |

| Responsible Party | Task |
|---|---|
| Customer | Develop merchant reconciliation procedures. The customer must determine how merchants are to reconcile with BASE24-pos. Options include an online transaction, a batch procedure, or a report. |
| Customer/Vendor | Develop receipt requirements. The customer and the vendor must determine how receipts are produced and the information to be included on the receipts. |
| Customer/Vendor/ACI | Develop implementation plan for terminal, as follows:<br>• Include vendor programming.<br>• Include vendor test plan with ACI.<br>• Include vendor delivery to customer. |
| Customer/Vendor/ACI | Complete customer testing. The customer tests the terminals and the SPDH module, along with the vendor and ACI. |
| Customer | Review terminals and software. The customer must accept the final product. |
| Customer | Implement terminals at merchant location. The customer implements terminals at merchant sites and ensures that they are fully functional. |
| Customer | Review procedures and make appropriate changes. The customer reviews the product and its procedures and makes any identified changes. |

*ACI Worldwide, Inc.*

# 2: Terminal Configuration

The BASE24 Standard POS Device Handler (SPDH) module enables customers to configure their POS terminals to meet their exact specifications.

Options supported by the SPDH module are based on the way the base, SPDH module, and associated BASE24-pos files are set up. These files consist of the following:

- ACI Standard Device Configuration File (ACNF)
- ACI Standard Device Response File (ARSP)
- Acquirer Processing Code File (APCF)
- Institution Definition File (IDF)
- Logical Network Configuration File (LCONF)
- Log Message Configuration File (LMCF)
- POS Retailer Definition File (PRDF)
- BASE24-pos Terminal Data files (PTD)
- POS Transaction Log File (PTLF)
- Response Code Description File (RCDF)
- SPDH Names File (SPDHNAMS)
- TSPD Names File (TSPDNAMS)

# SPDH Configuration Files

This section describes the files used to configure the SPDH module. The descriptions explain the role each file plays in determining the options supported by the SPDH module.

## ACI Standard Device Configuration File

The ACI Standard Device Configuration File (ACNF) is central to the configuration of the SPDH module. It specifies processing parameters, the fields to be included in requests and responses for every transaction type, the fields to be encrypted within the messages, the fields to be verified through message authentication codes (MACs) within the messages, information for dynamic key management, and all the information downloaded to the terminal.

This file is accessed through BASE24 files maintenance. Instructions on setting up and accessing the ACNF can be found in section 8.

The ACNF is a key-sequenced file, with a primary key consisting of three fields: the TERMINAL GROUP, RECORD TYPE, and RECORD ID.

The value in the TERMINAL GROUP field should correspond to a value in the TERM GROUP field on screen 1 of the POS Terminal Data files (PTD). The SPDH module uses the TERM GROUP field on PTD screen 1 to determine the ACNF records associated with the terminal. Therefore, ACNF records are applicable to an entire group of terminals. The number of terminals within one group is determined by the customer when they set up records in the BASE24-pos Terminal Data files (PTD).

The value in the RECORD TYPE field indicates whether the record contains processing data, field map data, or download data. These record types are discussed in the following paragraphs. The value in the RECORD ID field identifies the record number within the record type.

### Processing Data

One processing data record exists for each terminal group defined in the ACNF. This record consists of fields and flags that allow customers to set up general processing parameters used by the SPDH module when handling messages. From this record, customers can configure several timeout options including the amount of time the SPDH module waits for responses from the Authorization and Mail

processes. Customers can also determine the maximum length of responses from the SPDH module to the terminal and whether the SPDH module is to log user data from the POS Standard Internal Message (PSTM) to the POS Transaction Log File (PTLF) with every financial transaction. In addition, customers can determine if totals returned to the terminal in responses are draft capture totals or if the totals include all totals.

Customers can use the ACNF to set up dynamic key management thresholds for PIN, message authentication code (MAC), and data encryption keys.

Several BASE24-mail options can also be set using this record. Customers can determine whether unsolicited mail messages are sent to the terminal, whether a terminal is allowed to perform implicit closes, and the data processing centers to which mail should be sent.

The SPDH module reads this data for a terminal into memory upon receiving the first message from the terminal. When the SPDH module receives a WARMBOOT message, the data in memory is deleted and this record is reread when the SPDH module receives the first message from the terminal.

## Field Map Data

Two field map data records exist for each terminal group included in the ACNF. The first record defines the fields that are required in requests for each transaction type, the fields within these request messages that are required to be encrypted, and the fields within these request messages to be verified with MACs. The second record defines the fields to be included in responses for each transaction type, the fields within these response messages to be encrypted, and the fields within these response messages to be verified with MACs.

Changes to field map data can be made online through BASE24 files maintenance. However, the customer and vendor must coordinate on this data because the field maps do not determine the data sent in requests. The purpose of the field map data in the ACNF is to allow the SPDH module to know what to expect from the terminal and how to format the responses to the terminal. Terminals do not receive the field map data. It is the responsibility of the terminal software to determine the data in requests.

The SPDH module reads this data for a terminal into memory upon receiving the first message from the terminal. When the SPDH module receives a WARMBOOT message, the data in memory is deleted and this record is reread when the SPDH module receives the first message from the terminal.

Although the fields within the requests and responses can be determined by the customer, some are required for certain transactions. Information concerning the fields required for specific transactions is included in section 3. Using MACs is described later in this section.

### Download Data

There are 12 download records for each terminal group included in the ACNF.

Records 00, 01, and 02 contain a total of 26 40-byte fields that can consist of any data the customer wants to download to the terminal or any information the terminal requires.

Records 03, 04, 05, 07, 08, 09, 10, and 11 contain card prefix range processing parameters. Examples of the processing parameters include whether cards within a specified card prefix range use draft capture processing, whether cards within a specified card prefix range require PINs, and authorization telephone numbers for each card prefix range specified. Up to 30 card prefix ranges can be defined in this set of records.

Record 06 contains flags that indicate the fields from the BASE24-pos Terminal Data files (PTD) that can be downloaded to the terminal.

Downloads can be initiated by terminals using a network control facility. Additional information about downloads is included in section 5. For more information on network control facilities, refer to the ***NET24-XPNET Network Control Operations Guide***.

## ACI Standard Device Response File

The ACI Standard Device Response File (ARSP) contains a table of terminal response displays in up to three different languages. The SPDH module uses this file to develop terminal response displays that describe the results of transaction processing to the terminal. Each response can contain a maximum of 48 bytes.

The SPDH module reads this data for a terminal into memory upon receiving the first message from the terminal. When the SPDH module receives a WARMBOOT message, the data in memory is deleted and this record is reread when the SPDH module receives the first message from the terminal.

The ARSP is a key-sequenced file with the primary key consisting of two fields: the TERMINAL GROUP and RECORD NUMBER. The value in the TERMINAL GROUP field should correspond to a value in the TERM GROUP field on screen 1 of the POS Terminal Data files (PTD). If no terminal group is specified on PTD screen 1, a terminal group default of **** can be used. If a customer wants to have all terminals or a select few associated with a set of ARSP records, the TERMINAL GROUP field in the ARSP should be left blank and the default terminal group can be used. In this case, the SPDH module reads the ARSP for the default records. The SPDH module uses the TERM GROUP field on PTD screen 1 to determine the ARSP records associated with the terminal. Therefore, ARSP records are applicable to an entire group of terminals. The number of terminals within one group is determined by the customer when they set up POS Terminal Data file records.

The ARSP contains a maximum of five records for each terminal group. These records consist of a response code map record, three display text records, and a transaction description table record. The response code map record allows for up to 1000 response codes to be mapped to descriptions defined in the display text records. Since there are three display text records, responses for the terminal can be in one of three languages. The language to be used is determined in the request from the terminal. Language parameters are set up in the LANGUAGE ID field on screen 1 of the PTD. The transaction description table record allows customers to determine the wording to be used to describe each transaction in the transaction set.

This file is accessed through BASE24 files maintenance. Instructions on setting up and accessing the ARSP are found in section 8.

## Acquirer Processing Code File

The Acquirer Processing Code File (APCF) contains one record for each combination of acquirer transaction profile, message category, and ISO processing code that is supported by BASE24-pos acquirer endpoint in the system. Each acquirer transaction profile defines a set of transactions supported for an individual acquiring terminal or group of terminals. For BASE24-pos, retailer and administrative card transaction profiles are also used to determine whether an administrative card is required to perform a transaction and whether a transaction is allowed for a specific administrative card.

When processing a transaction from a POS device, the SPDH module accesses the APCF extended memory table to determine whether a transaction is allowed from that device. The SPDH module accesses the APCF directly when downloading the transactions allowed table to a POS device.

# Institution Definition File

The Institution Definition File (IDF) contains one record for each institution in the network. The SPDH module uses the IDF to obtain the token retrieval option, if it is not set at the terminal level in the BASE24-pos Terminal Data files (PTD), and to obtain the default acquirer transaction profile, if it is not set at the retailer level in the PRDF or at the terminal level in the BASE24-pos Terminal Data files (PTD). The token retrieval option indicates whether tokens are included in reversal messages, and if so, from where the token data is retrieved. The acquirer transaction profile specifies a set of allowed transaction processing codes and is used to determine whether an acquired transaction is allowed at a terminal.

The SPDH module accesses information in the IDF using the IDF extended memory table.

# Logical Network Configuration File

One Logical Network Configuration File (LCONF) exists for each logical network. It contains the assigns and params associated with the network. For basic information on assigns and params, as well as more information on the LCONF, refer to the ***BASE24 Logical Network Configuration File Manual***.

## Assigns

The SPDH module requires several assigns in the LCONF in order to perform processing. These assigns are as follows and are in addition to any assigns required by the Router/Authorization module bound to the SPDH module.

**APCF —** The fully file qualified name of the Acquirer Processing Code File (APCF).

**POS-SPDH-ACNF —** Contains the name of the ACI Standard Device Configuration File (ACNF).

**POS-SPDH-ARSP —** Contains the name of the ACI Standard Device Response File (ARSP).

**POS-PTD-DYN-GNRL —** The fully qualified file name of the general BASE24-pos Terminal Data Dynamic File—general data (PTDD1).

**POS-PTD-DYN-SCRATCH-PAD —** The fully qualified file name of the second BASE24-pos Terminal Data Dynamic File—scratch pad (PTDD2).

**POS-PTD-STATIC-GNRL —** The fully qualified file name of the BASE24-pos Terminal Data Static File—general data (PTDS1).

**POS-DH-PTD-EXTMEM-SWAPVOL —** The fully qualified name of the POS Terminal Data File extended memory swap volume.

**RCDFEMT —** The fully qualified file name of the Response Code Description File (RCDF) extended memory table. A separate utility program builds this extended memory table from the RCDF for use by the SPDH module.

## Params

The SPDH module uses several params during processing. The following params are in addition to any params required by the Router/Authorization module bound to the SPDH module.

**POS-DH-CONS-MAC-ERR-LMT —** Contains the maximum number of consecutive message authentication code (MAC) errors allowed before the SPDH module takes corrective action.

**POS-DH-DUKPT-UPDATE-METHOD —** Specifies whether the SPDH module can automatically update the PIN ENCRYPTION TYPE field on BASE24-pos Terminal Data files (PTD) screen 7 to a value of 07 (DUKPT) when the Key Serial Number (KSN) and Descriptor field (SubFID T of FID 6) is received in a message for the first time and a Derivation Key File (KEYD) record exists for the SPDH terminal.

**POS-DH-ISO-RESP-CDE-FRMT —** Controls which International Organization for Standardization (ISO) response code format is used in FID X of the POS standard message. If used, the two-digit value in the SPDHNAMS File must be left-justified with a trailing blank.

**POS-DH-KEYD-FROM-DISK —** Specifies whether the SPDH module reads the Derivation Key File (KEYD) from disk or from memory.

**POS-DH-KSN-APPRV-OPT —** Controls the action taken by the SPDH if the static data received in the Key Serial Number (KSN) and Descriptor field (FID 6, subFID T) of a request message does not match the KSN static data maintained in PTD device dependent data.

**POS-DH-KSN-CNTR-APPRV-OPT —** Controls the action taken by the SPDH based upon the the transaction counter for two transactions from the same terminal.

**POS-DH-KSN-DESCR —** Specifies the default key serial number (KSN) descriptor required by the Thales e-Security hardware security module (HSM) to perform a translate command.

**POS-DH-KSN-MAX-DIFFERENCE —** Indicates the largest allowable difference in the transaction counter between two transactions from the same terminal.

**POS-DH-KSN-SAVE-OPT —** Specifies whether to save and restore the Key Serial Number (KSN) from the previous transaction in an out-of-sequence situation.

**POS-DH-LMT-EXCEED-DISP —** Specifies the action the SPDH module takes when the return or adjustment limits are exceeded.

**POS-DH-MAX-CNFGS —** Indicates the maximum number of ACI Standard Device Configuration File (ACNF) records to be stored in memory by the SPDH module at one time.

**POS-DH-MAX-TERMS —** The maximum number of POS Terminal Data file records that each SPDH module can hold in extended memory at one time. Thus, this param indicates the maximum number of terminals each SPDH module can service.

**POS-DH-MAX-TERMS-IN-DYN-TBL —** The maximum number of POS Terminal Data Dynamic File (PTDD1 and PTDD2) records that an SPDH module can store in extended memory at one time. Thus, this param indicates the amount of extended memory each SPDH module must reserve for PTDD1 and PTDD2 records and the maximum number of transactions each SPDH module can process at one time.

**POS-DH-MAX-TERMS-IN-STATIC-TBL —** The maximum number of POS Terminal Data Static File—general data (PTDS1) records that an SPDH module can store in extended memory at one time. Thus, this param indicates the amount of extended memory each SPDH module must reserve for PTDS1 records and the maximum number of transactions each SPDH module can process at one time.

**POS-DH-PTD-STATIC-REPL —** Controls whether the POS Terminal Data Static File (PTDS1) requires a warmboot after file maintenance is performed.

**POS-LOG-MAC-ERR —** Controls whether the SPDH module logs a response to the PTLF when an invalid MAC request occurs.

**POS-PASSTHRU-ACQ-SEQ-NUM-CHK —** Indicates whether the acquirer of a merchant link financial transaction is to check sequence numbers.

**POS-PASSTHRU-PROCESS-TYPE —** Indicates whether a POS device is an acceptor or an acquirer of merchant link financial transactions.

**POS-DISCR-DATA-PSTM-SUPPRS —** Indicates whether discretionary data carried in the BASE24-pos Standard Internal Message (PSTM) 0210 response is suppressed.

**POS-RETRY-TIMER —** Indicates the number of seconds the SPDH module waits before retrying a message to the device after a message failure has occurred. REVERSE-BAL-INQ

**POS-SPLIT-TXN-RTG-DEST —** Specifies the destination of the Transaction Context Manager (TCM). The TCM is required for split transaction routing used with the SPDH Device Handler Mobile Top-Up extension. The Mobile Top-Up extension accommodates the possibility of multiple authorization destinations. The SPDH Device Handler module routes transactions with a mobile top-up transaction subtype to the TCM destination specified in this parameter.

## Log Message Configuration File (LMCF)

As with all BASE24 application processes, the SPDH module can access the LMCF to generate modified event messages.

## POS Retailer Definition File

The POS Retailer Definition File (PRDF) contains one record for each retailer in the network. The PRDF is used by the SPDH module in validating POS transactions and defining authorization and routing parameters. The SPDH module also retrieves the name of the merchant and posting date when this information is to be downloaded to the terminal. The customer determines if this information is to be downloaded to the terminal by setting flags in the ACNF. For more information on the PRDF, refer to the ***BASE24-pos Files Maintenance Manual***.

## POS Terminal Data Files

The BASE24-pos Terminal Data files (PTD) contain one record for each POS terminal in the network, including terminals that conform to the SPDH module. The BASE24-pos Terminal Data files (PTD) consist of the following three files:

- The POS Terminal Data Dynamic File—general data (PTDD1) contains dynamic data such as terminal totals that changes during transaction processing.
- The POS Terminal Data Static File—general data (PTDS1) contains static terminal data that generally does not change during the course of transaction processing.

  Generally, when file maintenance is performed to the static data in the PTDS1, a warmboot is required for this data to be used in processing. However, a parameter in the Logical Network Configuration File (LCONF) can override this requirement. When the POS-DH-PTD-STATIC-REPL parameter is set to Y, and a change to static data in the PTDS1 is made, the latest modified record is used in the current transaction without a warmboot.

  For more information on the POS-DH-PTD-STATIC-REPL parameter, refer to the ***BASE24 Logical Network Configuration File***.
- The POS Terminal Data Dynamic File—scratch pad (PTDD2) is an optional file used to store last message token data for reversal processing. If this file is not used, last message token data is retrieved from the PTLF for reversal processing.

The SPDH module uses the TERMINAL ID field contained in the standard message header of all request messages to access the appropriate records in the BASE24-pos Terminal Data files (PTD).

For more information on the BASE24-pos Terminal Data files (PTD), refer to the ***BASE24-pos Files Maintenance Manual***.

# POS Transaction Log File

The POS Transaction Log File (PTLF) contains information about the transactions processed by BASE24-pos. Approved or denied transactions are logged to the PTLF only if they are sent by a terminal. When a close batch request, a close shift request, or a close day request transaction is performed, totals are logged to the PTLF if logging PTLF totals is configured in the LOG TOTALS field on screen 3 of the PTD. If logging PTLF totals is not configured, no totals are logged to the PTLF.

Last message token data can be retrieved from the PTLF for reversal processing if a POS terminal is configured to do so.

For more information on the PTLF, refer to the ***BASE24-pos Files Maintenance Manual***.

## Formatting User Data in the PSTM

User data contained in the PSTM can be included in messages if it is formatted and sent by the terminal and if the PSTM USER FLD field on screen 2 of the ACNF is set to a value of Y (Yes). The PSTM USER FLD field indicates to the SPDH module whether to format user data in the PSTM for all financial transactions. The USER-DATA field from the PSTM in the BASE24 transaction log data can be extracted from the PTLF for host usage.

The USER-DATA field in the PSTM contains the information included in the table below. The information must be included in the transaction request in order to be placed in the USER-DATA field.

| FID | Field | PIC |
|---|---|---|
| K | Business Date | 9(06) |
| Q | Echo Data | X(16) |
| a | Optional Data | X(80) |
| d | Retailer ID | X(12) |

| FID | Field | PIC |
|---|---|---|
| h | Sequence Number | X(10) |
| i | Original Sequence Number | X(09) |

## Response Code Description File

The Response Code Description File (RCDF) contains information used to override BASE24 response code description contained in the ARSP. The RCDF also contains the ISO response codes and descriptions used in addition to BASE24 response codes.

For more information on the Response Code Description File (RCDF), refer to the ***BASE24-pos Files Maintenance Manual***.

## SPDH Names File (SPDHNAMS)

The SPDH Names File (SPDHNAMS) is used to configure the SPDH module if FID X is used in the POS standard message. Unlike the other files mentioned in this section, SPDHNAMS is a TGAL edit file that is required when the SPDH module is compiled.

SPDHNAMS contains the information needed to map a BASE24-pos response code to an ISO response code. SPDHNAMS can also be used to map a single BASE24-pos response code to multiple ISO response codes based upon account type. The default SPDHNAMS causes the SPDH module to place blanks in FID X.

The International Organization for Standardization (ISO) response code format used in FID X of the POS standard message can be a three-digit value based on the ISO 8583-1993 standard or a two-digit value based on the ISO 8583-1987 standard. The POS-DH-ISO-RESP-CDE-FRMT param in the LCONF controls which standard is used.

# TSPD Names Files (TSPDNAMS)

The TSPD Names File is an edit file that contains tables that are used to determine transaction routing. The TSPDNAMS file allows the SPDH Device Handler module to detect Issuer Identification Numbers (IIN) that signify a transaction subtype. The IIN is later used by the Transaction Context Manager (TCM) to route the transaction to the authorization destination(s) required.

*ACI Worldwide, Inc.*

# 3: BASE24 Standard POS Device Handler Module

In BASE24-pos, the Device Handler module is bound together with a Router/Authorization module to create the Device Handler/Router/Authorization process. The Device Handler/Router/Authorization process is composed of three modules—Device Handler, Router, and Authorization. Each module has distinct functions, but they operate together and are defined as one bound process. Because the three modules are bound, transactions passed between them do not travel through the XPNET process.

The Router and Authorization modules are the same for each Device Handler/Router/Authorization process. However, the Device Handler module differs between Device Handler/Router/Authorization processes depending on the type of POS device attached to the system. For example, a different Device Handler module is required for Hypercom devices than is required for BASE24-pos Standard POS devices.

This section is devoted to describing the processing associated with the Device Handler module required for BASE24-pos Standard POS devices—the SPDH module.

This section describes the processing performed by the SPDH module. It includes the following information about the SPDH module:

- An overview of the SPDH module
- A description of each file used by the SPDH module during processing
- A description of the steps the SPDH module performs during initialization and warmbooting
- Network text commands recognized by the SPDH module
- A description of event message generation and documentation

# SPDH Module Overview

The SPDH module is the primary interface between a POS device and BASE24-pos. Each type of POS device has its own message format and method of processing transactions. The SPDH module supports BASE24-pos Standard POS devices. The SPDH module reformats messages sent from BASE24-pos Standard POS devices into the POS Standard Internal Message (PSTM). The PSTM allows the other processes associated with BASE24-pos to process the transaction.

Likewise, all messages that BASE24-pos sends to BASE24-pos Standard POS devices must be reformatted into *native mode*, a message format the BASE24-pos Standard POS device can understand. The SPDH module performs this translation before the transaction message is forwarded to the BASE24-pos Standard POS device.

The SPDH module also controls several other functions in the BASE24-pos system. These functions are listed and briefly described in the following paragraphs.

## Enforcing Transaction Sequence

The SPDH module can verify that the sequence number in a message request matches the sequence number stored in the BASE24-pos Terminal Data files (PTD) for a specific transaction. Sequence number checking is intended to ensure that BASE24-pos has received and processed each transaction only once. When the terminal is unable to assign a transaction sequence number, the SPDH module assigns one when it receives the transaction.

## Allowed Transaction Checking

For each transaction acquired from a POS device, the SPDH module uses an acquirer transaction profile to determine whether the transaction is allowed from that terminal. Acquirer transaction profiles can be set at the terminal level in the BASE24-pos Terminal Data files (PTD), at the retailer level in the PRDF, or at the institution level in the IDF. The SPDH module always attempts to retrieve the acquirer transaction profile from the terminal's BASE24-pos Terminal Data files (PTD) record first. If the acquirer transaction profile is found in the BASE24-pos Terminal Data files (PTD), it is used to read the Acquirer Processing Code File (APCF) extended memory table as described below. If the acquirer transaction profile is blank in the terminal's BASE24-pos Terminal Data files (PTD) record, the module attempts to retrieve the acquirer transaction profile from the retailer's

PRDF record. If the acquirer transaction profile is found in the PRDF, it is used to read the APCF extended memory table as described below. If the acquirer transaction profile is blank in the retailer's PRDF record, the module attempts to retrieve the default acquirer transaction profile from the terminal owner's IDF record. If the acquirer transaction profile is found in the IDF, it is used to read the APCF extended memory table as described below. Finally, if the acquirer transaction profile is blank in the IDF, a value of all asterisks is used to read the APCF extended memory table.

Once the acquirer transaction profile value used to read the APCF extended memory table has been determined, the SPDH module attempts to read the APCF extended memory table using the acquirer transaction profile value, the message category code from the transaction, and the ISO processing code for the transaction. The SPDH module calls a utility to map the internal BASE24 transaction code for the transaction to the corresponding ISO processing code before reading the APCF extended memory table.

Using the acquirer transaction profile, message category, and ISO processing code, the SPDH module steps through the records in the APCF extended memory table looking for a match. If an exact match is not found, certain other alternatives are allowed. Below is the hierarchy used for selection.

| Preference | Acquirer Transaction Profile | Message Category | ISO Processing Code |
|---|---|---|---|
| 1 | Exact match | Exact match | Exact match |
| 2 | Exact match | * | Exact match |
| 3 | ********** | Exact match | Exact match |
| 4 | ********** | * | Exact match |

Once a match is found, the record is read to determine whether the transaction is allowed. If the transaction is not allowed, the transaction is declined with response code 055 (invalid transaction). If the transaction is allowed, the SPDH module performs the following:

- Adds the Transaction Profile token with the acquirer transaction profile used.
- Adds the Acquirer Routing token if the transaction is allowed and an authorization destination is specified in the APCF record.

**Note:** The APCF also allows you to specify an authorization destination for a transaction other than the Router/Authorization module bound in with the SPDH module. This mechanism allows you to route transactions received from a POS device to an application process in the XPNET system. When a transaction is routed in this way, you can also specify whether the transaction response returned from the authorization destination is logged to the POS Transaction Log File (PTLF).

- Adds the Transaction Description token with the transaction description from the APCF record. If adding this token causes the maximum message length to be exceeded, it is not added.

## Updating Accumulator Fields and Status Flags

The SPDH module updates the accumulator fields and status flags in the BASE24-pos Terminal Data files (PTD) record for each BASE24-pos Standard POS device. For example, when a transaction passes through the Device Handler/Router/Authorization process, the SPDH module updates the accumulation fields BATCH NUMBER, SHIFT NUMBER, NUMBER OF BATCHES, and NUMBER OF SHIFTS on screen 3 of the PTD by one and updates other fields and flags depending on the type of transaction and the processing required.

## Cutting Over Terminals

The SPDH module cuts over terminals for daily settlement using a cutover transaction. The institution designates a timeframe in which cutover must take place and designates a transaction to start cutover. When the Device Handler/Router/Authorization process receives that transaction during the designated timeframe, the terminal is cut over. If the SPDH module does not perform this cutover, the Settlement Initiator performs a force cutover.

## Decrypting PINs

The SPDH module can decrypt PINs using software for incoming transactions. However, if decryption is done using a hardware security module, the SPDH module passes the transaction to the Authorization module. The Authorization module then passes the transaction to the hardware security module. Refer to the ***BASE24 Transaction Security Manual*** for more information on PIN decryption.

## Authenticating Messages

The SPDH module can authenticate messages passed between BASE24-pos and a POS device by using message authentication codes (MACs). The options for using MACs with the SPDH module consist of software or hardware MAC generation and verification. Refer to section 6 of this manual for a detailed discussion of how MACs are used specifically with the SPDH module. For more information on using software and hardware MACs, refer to the ***BASE24 Transaction Security Manual***.

## Downloading

The SPDH module performs both full and partial downloads consisting of terminal configuration information. The information is downloaded from load files. The SPDH module downloads information from the ACNF to BASE24-pos Standard POS devices.

## Controlling Response Timers

The SPDH module sets and stops transaction response timers as transactions pass in and out of the SPDH module. When the SPDH module receives a transaction from a device, it sets a response timer and then passes the transaction on for processing. When processing completes, the SPDH module deletes the timer. If processing does not complete before the timer expires, the transaction is either declined or reversed, depending on what type of transaction timed out.

## Validating Return and Adjustment Limits

The SPDH module checks terminal return and adjustment limits in the BASE24-pos Terminal Data files (PTD) as part of its request validation function. If the limits have been exceeded, the SPDH module continues processing the transaction, denies the transaction, or refers the transaction based on the value of the LCONF parameter POS-DH-LMT-EXCEED-DISP.

Refer to the ***BASE24 Logical Network Configuration File Manual*** for more information about the POS-DH-LMT-EXCEED-DISP param. For more information about return and adjustment limit fields in the BASE24-pos Terminal Data files (PTD), refer to the ***BASE24-pos Files Maintenance Manual.***

## Checking Limits

The SPDH  module checks return and adjustment limits on the following transactions:

- Merchandise return
- Adjustment to merchandise return
- Adjustment to purchase
- Adjustment to cash advance

An adjustment to a merchandise return goes through two sets of checks.  After the SPDH module performs return limit checks, it then performs adjustment limit checks.

The first check is to see if the transaction is under the return or adjustment count limit.  The following table describes the return or adjustment count check:

| Transaction | Over Count Limit If... |
|---|---|
| Return | The value of the CREDIT BATCH COUNT field on PTD screen 4 plus 1 is greater than the value of the RETURN COUNT field on PTD screen 3. |
| Adjustment | The value of the ADJUST BATCH COUNT field on PTD screen 4 plus 1 is greater than the value of the ADJUSTMENT COUNT field on PTD screen 3. |

If the count limit has been exceeded, the SPDH module takes the action specified by the POS-DH-LMT-EXCEED-DISP param.  This is described in the topic "Processing Transactions Exceeding Limits."  If the count limit has not been exceeded, the SPDH module checks return or adjustment amount limits.

The following table describes the return or adjustment amount checks:

| Transaction | Over Amount Limit If... |
|---|---|
| Merchandise return | The value of the CREDIT BATCH AMOUNT field on PTD screen 4 plus the amount of the merchandise return is greater than the value of the RETURN LIMIT field on PTD screen 3. |

| Transaction | Over Amount Limit If... |
|---|---|
| Adjustment to merchandise return (return amount limit) | The value of the CREDIT BATCH AMOUNT field on PTD screen 4 plus the adjusted amount of the merchandise return is greater than the value of the RETURN LIMIT field on PTD screen 3. |
| Adjustment to purchase or adjustment to cash advance | The value of the ADJUST BATCH AMOUNT field on PTD screen 4 plus the adjusted amount of the purchase or cash advance is greater than the value of the TERM. ADJUSTMENT LIMIT field on PTD screen 3. |
| Adjustment to merchandise return (adjustments amount limit) | The value of the ADJUST BATCH AMOUNT field on PTD screen 4 minus the adjusted merchandise return amount is greater than the value of the TERM. ADJUSTMENT LIMIT field on PTD screen 3. |

If the transaction does not exceed amount limits, the SPDH module continues processing as usual. If the transaction exceeds the amount limit, the SPDH module takes the action specified by the POS-DH-LMT-EXCEED-DISP param, as discussed below.

## Processing Transactions Exceeding Limits

If a return or adjustment transaction has exceeded a count or amount limit, the SPDH module performs as follows:

1. Logs a message.
2. Sets the error flag in the BASE24-pos Release 5.0 token to E (return limit exceeded) or A (adjustment limit exceeded).
3. Takes action specified in the POS-DH-LMT-EXCEED-DISP param, as follows:
   - If the POS-DH-LMT-EXCEED-DISP param is set to 0 (continue processing) or if this is a force-post or store-and-forward transaction (message subtypes F and S, respectively), the SPDH module continues processing the transaction. Limits processing is now complete.

- If the POS-DH-LMT-EXCEED-DISP param is set to 1 (deny transaction), the SPDH module denies the transaction with a response code 094 and continues with step 4.
- If the POS-DH-LMT-EXCEED-DISP param is set to 2 (refer transaction), the SPDH module denies the transaction with a response code 102 and continues with step 4.

4. Sends the transaction to the Router/Authorization module to be logged to the PTLF. This is a log only transaction (message type 9900).
5. Responds to the terminal.

## Reversal Processing

The TOKEN RETRIEVAL OPTION field on Institution Definition File (IDF) screen 19 indicates whether BASE24-pos Device Handler modules include tokens in reversal messages, and if so, whether the token data is retrieved from the second BASE24-pos Terminal Data Dynamic File—scratch pad (PTDD2) or from the POS Transaction Log File (PTLF). This option can be overridden at the terminal level using the TOKEN RETRIEVAL OPTION field on POS Terminal Data files (PTD) screen 2.

When the SPDH module receives a request from a POS device, it creates the TLF token with a field indicating the token reversal processing option. If the option is set to a value of 2 in the TLF token, the Router/Authorization module updates the token with POS Transaction Log File (PTLF) key data when processing the transaction response. The SPDH module updates device-dependent PTD extended memory with the TLF token data and writes this data to the BASE24-pos Terminal Data Dynamic File—general data (PTDD1). It can then use this information to retrieve the record from the PTLF during reversal processing. If the token retrieval option is set to a value of 0 or 1, the Router/Authorization module does not update the TLF token.

When the SPDH module receives a response from the Router/Authorization module, it checks the token retrieval option in the TLF token. If the option is set to a value of 1, the SPDH module stores token data from the response message in the PTDD2 file. If no record exists in the PTDD2 file for the terminal from which the transaction originated, a new record is created using the terminal ID as the key. If a record already exists for the terminal, the record is updated with the token data from the current message. The SPDH module can then retrieve the record from the PTDD2 file during reversal processing.

## EMV Support

If you have installed the BASE24-pos Europay, MasterCard, and Visa (EMV) add-on product, the SPDH module performs the following functions:

- For requests, the module maps EMV data from the request message to tokens and appends these tokens to the PSTM. If EMV terminal capabilities are not provided in the request message, the module checks the EMV terminal capability settings for the terminal on PTD screen 17 to determine which EMV tokens to include. The module formats the EMV Request Data token, the EMV Status token, and the EMV Discretionary Data token as needed.
- For responses, the module maps EMV data from the EMV Response Data token and the EMV Script Data token to the response message sent back to the terminal.

For detailed information on PTD screen 17, refer to the ***BASE24-pos EMV Support Manual***.

## Multiple Currency Support

If you have installed the BASE24-pos Multiple Currency add-on product, the SPDH module performs the following multiple currency functions:

- Validates the transaction currency code passed in the request message against the primary terminal currency defined in the CURRENCY CODE field on PTD screen 2.
- Adds the Multiple Currency token with an entry for the primary terminal currency.
- If the MC TOTALLING TYPE field on PTD screen 2 is set to a value of 1, amounts in the PSTM are converted into the primary terminal currency before they are stored in the POS Terminal Data Dynamic File—general data (PTDD1). Otherwise, amounts in the PSTM are not converted into the primary terminal currency before they are stored in the PTDD1.

For detailed information on Multiple Currency support for the SPDH module, refer to the ***BASE24-pos Multiple Currency Support Manual***.

# File Usage

The SPDH module uses the following files during processing. Each file listed is followed by a brief description explaining how it is used by the SPDH module.

**ACI Standard Device Configuration File (ACNF) —** The ACNF is used exclusively by the SPDH module. This file is central to the configuration of the SPDH module. It specifies processing parameters, the fields to be included in requests and responses for each transaction type, and all the information downloaded to BASE24-pos Standard POS terminals.

**ACI Standard Device Response File (ARSP) —** The ARSP is used exclusively by the SPDH module. This file contains a table of terminal response displays in up to three different languages. The SPDH module uses the ARSP to develop a terminal response display that describes the results of transaction processing to the terminal. The SPDH module reads this file into memory at initialization and reinitialization.

**Acquirer Processing Code File (APCF) —** The SPDH module accesses the APCF extended memory table to determine whether a transaction is allowed from a POS device. The SPDH module accesses the APCF directly when downloading the transactions allowed table to a POS device.

**Institution Definition File (IDF) —** The SPDH module accesses the IDF extended memory table to obtain the acquirer transaction profile (if it is not defined in the POS Terminal Data files or PRDF) and token retrieval option (if it is not defined in the POS Terminal Data files).

**Logical Network Configuration File (LCONF) —** The assigns and params used during processing are read from the LCONF at initialization and reinitialization, and are maintained in memory.

**Log Message Configuration File (LMCF) —** The LMCF contains event message information such as text, routing code, and severity, that has been modified by the user. At initialization, the Device Handler/Router/Authorization process builds a table in memory from the LMCF records that contains modified information. When an event message is generated, the Device Handler/Router/Authorization process accesses the table to verify whether that event message has been modified. If the event message has been modified, the Device Handler/Router/Authorization process logs the modified information instead of the standard event message.

**POS Retailer Definition File (PRDF) —** The SPDH module accesses the PRDF to obtain the acquirer transaction profile if it is not defined in the BASE24-pos Terminal Data files (PTD).

**BASE24-pos Terminal Data files (PTD) —** The SPDH module uses records in the BASE24-pos Terminal Data files (PTD) to build a table of PTD key positions in extended memory. This table enables the SPDH module to access and update the BASE24-pos Terminal Data files (PTD) records as transactions are processed.

**POS Transaction Log File (PTLF) —** The PTLF contains information about the transactions processed by BASE24-pos. Approved or denied transactions are logged to the PTLF only if they are sent by a terminal. When a close batch request, a close shift request, or a close day request transaction is performed, totals are logged to the PTLF if logging PTLF totals is configured in the LOG TOTALS field on screen 3 of the PTD. If logging PTLF totals is not configured, no totals are logged to the PTLF.

**Response Code Description File (RCDF) —** The RCDF contains information used to override BASE24 response code description contained in the ARSP. The RCDF also contains the ISO response codes and descriptions used in addition to BASE24 response codes.

# SPDH Module Initialization and Warmboot

This section describes the steps taken by the SPDH module when it initializes or is warmbooted.

When the Device Handler/Router/Authorization process is initialized, the Router module begins initialization first. Part of the initialization process required by the Device Handler module is within the Router module. The steps described in this section assume that the Router module has already initialized.

The SPDH module performs the following steps to initialize:

1. Initializes the logging process with the name of the specific SPDH module being initialized and reads the Log Message Configuration File (LMCF).
2. Retrieves the following assigns and params from the Logical Network Configuration File (LCONF):

   **Assigns**

   APCF
   POS-SPDH-ACNF
   POS-SPDH-ARSP
   POS-PTD-DYN-GNRL
   POS-PTD-DYN-SCRATCH-PAD
   POS-PTD-STATIC-GNRL
   POS-DH-PTD-EXTMEM-SWAPVOL
   RCDFEMT

   **Params**

   POS-DH-CONS-MAC-ERR-LMT
   POS-DH-DUKPT-UPDATE-METHOD
   POS-DH-ISO-RESP-CDE-FRMT
   POS-DH-KEYD-FROM-DISK
   POS-DH-KSN-APPRV-OPT
   POS-DH-KSN-CNTR-APPRV-OPT
   POS-DH-KSN-DESCR
   POS-DH-KSN-MAX-DIFFERENCE
   POS-DH-LMT-EXCEED-DISP
   POS-DH-MAX-CNFGS
   POS-DH-MAX-TERMS
   POS-DH-MAX-TERMS-IN-DYN-TBL
   POS-DH-MAX-TERMS-IN-STATIC-TBL
   POS-DH-MAX-CNFGS

POS-DH-PTD-STATIC-REPL
POS-LOG-MAC-ERR
POS-PASSTHRU-ACQ-SEQ-NUM-CHK
POS-PASSTHRU-PROCESS-TYPE
POS-RETRY-TIMER
REVERSE-BAL-INQ

3. Opens the BASE24-pos Terminal Data files (PTD) and sets the file mode so any process attempting to lock or read a PTD record while the SPDH module has the record locked receives an error 73. This error indicates the file or record is locked. If the SPDH module is unable to open the BASE24-pos Terminal Data files (PTD), the SPDH module abends.
4. Allocates and initializes extended memory space for the number of POS Terminal Data file records it needs at one time. The number of records that can be in memory at one time is specified in the POS-DH-MAX-TERMS param.
5. Opens the ACI Standard Device Configuration File (ACNF). This file is used in the configuration of the SPDH module. If the SPDH module is unable to open the ACNF, the SPDH module abends.
6. Allocates extended memory for the ACNF.
7. Opens the ACI Standard Device Response File (ARSP). If the SPDH module is unable to open the ARSP, the SPDH module abends.
8. Allocates extended memory for the ARSP and loads the default records into extended memory.
9. Opens the Response Code Description File extended memory table (RCDFEMT) and places its contents into extended memory, overriding information from the ARSP where necessary.

   **Note:** The RCDFEMT must be built with the Base Extended Memory Table Build utility prior to initializing the SPDH module. Refer to the ***BASE24-pos Transaction Processing Manual*** for a complete discussion of the Base Extended Memory Table Build utility.

## Warmbooting the SPDH Module

When the SPDH module receives a WARMBOOT command, it performs the same steps described in the above initialization procedures with one exception. Step 1 in the initialization procedures described above indicates the SPDH module initializes the logging process with the name of the specific SPDH module being initialized. This step is not performed by the SPDH module when it receives a WARMBOOT command.

Instead, when a WARMBOOT command is received, the first step taken by the SPDH module is to read the Log Message Configuration File (LMCF). An internal table stored in the SPDH module containing LMCF information requires updating if event messages have been altered using the LMCF. The SPDH module requires warmbooting in order for any LMCF changes to take affect. The logging facility itself does not be reinitialized because it was already started during initialization.

The remaining steps (beginning with step 2) described in the initialization procedures above are identical when a WARMBOOT command is received by the SPDH module.

## Warmbooting Static POS Terminal Data

When the SPDH module receives a WARMBOOT PTD command, it warmboots static terminal data held in extended memory. The module re-reads its POS Terminal Data Static File—general data (PTDS1) into extended memory. Any time static data is changed for an ACI Standard POS Device on the POS Terminal Data files (PTD) screens, it is not used in processing until the associated SPDH module receives this command or is reinitialized.

## Warmbooting the RCDF Extended Memory Table

The SPDH module uses the ALTER ATTRIBUTE command to warmboot the RCDF extended memory table (RCDFEMT). When the SPDH receives an ALTER ATTRIBUTE RCDFEMT command, it places the new RCDF extended memory table in extended memory, starts using the new RCDF extended memory table, and removes the old RCDF extended memory table from extended memory. The SPDH module does not read the RCDF.

**Note:** The RCDFEMT must be built with the Base Extended Memory Table Build utility prior to placing it in extended memory. Refer to the ***BASE24-pos Transaction Processing Manual*** for a complete discussion of the Base Extended Memory Table Build utility.

# Text Commands

Network text commands are used to send information from a network control facility to a process. These commands can be entered by an operator using a network control facility and serve as a tool that operators can use to perform network management functions such as changing certain processing parameters, warmbooting a specific process, monitoring a specific process, and resolving problems encountered by a specific process. The following text commands can be used with the SPDH module. For detailed descriptions and the syntax of all text commands, refer to the ***BASE24 Text Command Reference Manual***.

**9501 (WARMBOOT) —** Directs the SPDH module to reinitialize.

For a list of the steps the SPDH module follows when a WARMBOOT command is issued, refer to the "Warmbooting the SPDH Module" topic elsewhere in this section.

**9525 (DEACTIVATE FLAG) —** Sets the status of the terminal to activated, deactivated, or pending deactivation. If the terminal is deactivated, the SPDH module declines transactions originated by that terminal. The default (initialized value) is deactivated. If the deactivate flag is in the command message set to 1 (deactivate) and all totals (batch, shift, day, and network) are zero, the terminal is deactivated. If any totals are not zero, the terminal is identified as pending deactivation and only balancing transactions are accepted from the terminal. If the deactivate flag in the command message is not 1, the terminal is activated.

**9529 (WARMBOOT PTD) —** Directs the SPDH module to warmboot static terminal data held in extended memory. Generally, this command should be issued any time an operator performs file maintenance on a BASE24-pos Terminal Data files (PTD) field that is stored in the BASE24-pos Terminal Data Static File—general data (PTDS1). Static data that is modified in the PTDS1 is not used in processing until the PTDS1 is warmbooted or the SPDH module is reinitialized.

This requirement can be overriden by using the POS-DH-PTD-STATIC-REPL parameter in the Logical Network Configuration File (LCONF). When the POS-DH-PTD-STATIC-REPL parameter is set to Y, and a change to static data in the PTDS1 is made, the latest modified record is used in the current transaction without a warmboot.

For more information on the POS-DH-PTD-STATIC-REPL parameter, refer to the ***BASE24 Logical Network Configuration File***.

**ALTERATTRIBUTE —** Directs the SPDH module to warmboot (reallocate) a specified shared extended memory table after it has been rebuilt. Whenever a shared extended memory table is modified, all processes that read the table must reallocate the table before the modified data is used in transaction processing. The Process Control Terminal (PCT) Server process automatically sends this command to each process specified in an EMT warmboot notify list param in the Logical Network Configuration File (LCONF) when a network operator enters an ALTERATTRIBUTE command from the EMT Control Command screen (screen 5) in the Device Control Terminal (DCT). You can also enter the ALTERATTRIBUTE command manually from a network control facility to warmboot an extended memory table for an individual process. The EMT warmboot notify list params are used only when issuing the command from the DCT. For more information on the EMT Control Command screen, refer to the ***BASE24 Device Control Manual***.

**LOADKEY —** Loads the PIN encryption key to the SPDH module (not supported).

**LOADFLAGON —** Indicates that download data is waiting for the terminal. Upon receipt of the LOADFLAGON command, the SPDH module sets the REQUEST-LOAD flag in the device dependent area of the POS Terminal Data Dynamic File—general data (PTDD1) to on. When the next response is being formatted to the terminal, the SPDH module sets the PROCESSING FLAG 2 in the standard header of the message to indicate that a download is waiting.

**LOADFLAGOFF —** Indicates that download data is not waiting for the terminal. The LOADFLAGON command sets the REQUEST-LOAD flag in the device dependent area of the POS Terminal Data Dynamic File—general data (PTDD1) to off. When the next response is being formatted to the terminal, the SPDH module sets the PROCESSING FLAG 2 in the standard header of the message to indicate that a download is not waiting.

**RESETSTATUS —** Resets the transaction status in the TRAN-STATUS field in the device dependent area of the POS Terminal Data Dynamic File—general data (PTDD1) to NO-TRAN-IN-PROGRESS. This command is intended for use if a transaction abends or some other uncontrollable event leaves the transaction status corrupted in the PTDD1.

**STATUSEXTMEM —** Directs the SPDH module to display statistics about its BASE24-pos Terminal Data files (PTD) extended memory segment on the network log. This command allows network operators to monitor extended memory usage and determine whether adjustments are needed.

**TRACEON —** Activates all trace functions within the SPDH module.

**TRACEOFF —** Deactivates the trace function within the SPDH module.

**TRACE1 —** Activates message trace only.

**TRACE2 —** Activates transaction information trace only.

**TRACE3 —** Activates key trace only.

**TRACE4 *number* —** Activates response time trace only. The variable *number* is the number of transactions that are accumulated before the average response time is displayed. This is entered by the network operator.

# Event Message Generation

The BASE24 Standard POS Device Handler (SPDH) module generates event messages to assist network operators in performing their daily tasks. Event messages are unsolicited messages generated by BASE24 processes that provide information regarding the internal condition of the system. They are a network operator's primary link to what is actually occurring in the system. Event messages provide operators with information ranging from system status, such as the completion of a procedure, to more serious circumstances requiring operator intervention.

ACI has developed a method that enables customers to access event message documentation electronically. Event messages are contained in files on a designated HP NonStop volume and subvolume. Each file contains event messages for a specific BASE24 process. Files on the designated HP NonStop volume and subvolume are organized according to product module ID (PMID). One PMID exists for each process in BASE24. PMIDs identify a specific process in the BASE24 system (e.g., PSSPDH identifies the SPDH module). Providing event message documentation in files on the HP NonStop gives customers greater flexibility by allowing them to print out event message documentation as needed at their location or to electronically manipulate the data to suit their needs.

Documentation is available in file form for the event messages generated by the SPDH module. It is located in the $*volume*.BA*xx*LOGM.PSSPDH file, where *volume* is the site-specific volume and *xx* is the number of the current release. For a complete explanation of event message documentation, refer to the ***BASE24 Event Message Reference Manual***.

# 4: The ACI Standard POS Message

This section contains the following information concerning the ACI standard POS message:

- Binary data conversion
- Control header prefixed to response messages for the X.21 protocol
- Standard message header
- Optional data fields
- Request and response message requirements
- External transaction code mapping to and from BASE24-pos internal transaction codes

The optional data fields to be included in requests from the terminal and responses from the SPDH module for each transaction type are specified in the ACI Standard Device Configuration File (ACNF). The ACNF is explained in more detail in sections 5 and 8.

# Binary Data Conversion

All data in an ACI standard POS message must be in ASCII format. Binary data used at the POS device must be converted into ASCII format before being included in a request message to the host. This conversion must be reversed to retrieve the binary data from an ASCII response received from the host. Hexadecimal characters are used to represent binary data in ASCII formatted messages as described below.

Several fields in EMV request and response data contain binary data. For conversion, the binary data is divided into groups of four bits. Each group of four bits is assigned a hexadecimal (hex) character. Thus, eight bits of binary data is represented by two hexadecimal characters. It is the hexadecimal characters that are carried in the ACI standard POS device message.

Each possible combination of the four bit values is assigned a hexadecimal character value according to the following table.

| Conversion Table | | | |
|---|---|---|---|
| **Bit Value** | **Hex Value** | **Bit Value** | **Hex Value** |
| 0000 | 0 | 1000 | 8 |
| 0001 | 1 | 1001 | 9 |
| 0010 | 2 | 1010 | A |
| 0011 | 3 | 1011 | B |
| 0100 | 4 | 1100 | C |
| 0101 | 5 | 1101 | D |
| 0110 | 6 | 1110 | E |
| 0111 | 7 | 1111 | F |

In all conversions, bit 0 of the binary data is always considered to be the most significant (i.e., “leftmost”) bit. Thus, for eight bits of binary data in a field, bits 0–3 are represented by the first hexadecimal character in the field, while bits 4–7 are represented by the second hexadecimal character in the field.

**Note:** In EMV specifications, bits are numbered from 8 to 1, with bit 8 considered to be the most significant bit. Thus, bit 8 in EMV specifications is represented by bit 0 in this manual, while bit 1 in EMV specifications is represent by bit 7 in this manual. The following table shows the EMV bit numbering scheme for the bits carried in the ACI standard POS message. The ACI bit numbering scheme is used for the applicable fields described in this manual.

| Bit Numbering | | | |
|---|---|---|---|
| **ACI** | **EMV** | **ACI** | **EMV** |
| 0 | 8 | 4 | 4 |
| 1 | 7 | 5 | 3 |
| 2 | 6 | 6 | 2 |
| 3 | 5 | 7 | 1 |

The following example illustrates how eight bits of binary data are converted by a POS device for placement in the Cryptographic Information Data field of an EMV request message sent to the BASE24 system.

| Bits | Binary Data | Hex Values |
|---|---|---|
| 0–3 | 1000 | 8 |
| 4–7 | 1011 | B |

Bits 0–3 of binary data are represented by the first hexadecimal character, while bits 4–7 are represented by the second hexadecimal character. In this example message, the cryptogram is an authorization request cryptogram (ARQC), issuer authentication failed, and an advice is required. The binary data for this information is 10001011, which is represented as “8B” in the ACI standard POS device message.

In EMV request messages, the EMV Request Data (FID 6, subFID O) field must contain the hexadecimal equivalents of binary data. In EMV response messages, the following fields must contain the hexadecimal equivalents of binary data.

- EMV Response Data (FID 6, subFID Q)
- EMV Additional Response Data (FID 6, subFID R)

# Control Header

For all response messages sent from the SPDH module to a device connected to the BASE24 system using the X.21 protocol, the module adds a control message header to the beginning of the message. The XPNET process uses this control header when responding to the POS device. The format is as follows:

| | |
|---|---|
| Originator | X |
| Control Code | X |
| Phone Number Length | X |
| Phone Number | X(21) |

Descriptions of the control message header fields are provided below. Possible values for each of these fields are listed throughout the descriptions. A value of a blank space is indicated with the symbol ␣. The data name associated with each field in the control message header is also provided.

**Originator —** A code identifying the originator of this message. For the SPDH module, this field is always set to 0 (application process).

Data Name: ORIGINATOR

**Control Code —** A code indicating the type of action for the XPNET process to take for this message. Valid values are as follows:.

1 = Disconnect
6 = Respond and disconnect
7 = Connect with data
8 = Connect with data and disconnect

Data Name: CONTROL-CODE

**Phone Number Length —** The length of the phone number used to connect with this POS device. The length is set to one greater than the length of the phone number in the TERMINAL PHONE field on PTD screen 1 for this terminal. This allows for a one-byte baud rate character to be appended to the end of the number.

Data Name: PHONE-NO-LEN

**Phone Number —** The phone number used to connect with this POS device. This is the value from the TERMINAL PHONE field on PTD screen 1 plus a binary baud rate character appended to the end. The binary baud rate character allows for multi-speed line support.

Data Name: PHONE-NO

# Standard Message Header

The standard header must be in every message sent from the terminal to the SPDH module and will be in every response sent from the SPDH module to the terminal. It consists of 13 fields, totaling 48 bytes, and is not configurable. The standard message header identifies information such as the type of message being sent, the employee entering the transaction, the ID of the terminal where the transaction was entered, and the transaction code associated with the transaction.

The customer and the terminal vendor decide the manner in which values in the standard header are determined. For example, when the values in the processing flag fields need to be changed, the customer and the vendor must agree on the manner in which to change the values.

Standard message header fields are summarized in the table below. Following the table are individual descriptions of the fields. Field descriptions include possible values for each of these fields along with the internal data name of the POS Standard Internal Message (PSTM) field, token, or file field to which the field is mapped. Fields not used in requests must be blank-filled or zero-filled. A value of a blank space is indicated with the symbol ␣.

## Standard Message Header Structure

The following table describes the structure of the standard message header—including for each of its fields, the positions occupied by the field, the length and format of each field, the name of the field, and the name of the internal field to which the field is mapped by the SPDH module.

| Position | Field Length | Field Name | Internal Field |
|---|---|---|---|
| 1–2 | 2 alphanumeric characters | Device Type | DEV-ADDR |
| 3–4 | 2 numeric characters | Transmission Number | XMIT-NUM |
| 5–20 | 16 alphanumeric characters | Terminal ID | TERM-ID |
| 21–26 | 6 alphanumeric characters | Employee ID | CLERK-ID |
| 27–32 | 6 numeric characters | Current Date | DAT |
| 33–38 | 6 numeric characters | Current Time | TIM |
| 39 | 1 alphanumeric character | Message Type | REQ-TYPE |

| Position | Field Length | Field Name | Internal Field |
|---|---|---|---|
| 40 | 1 alphanumeric character | Message Subtype | REQ-SUBTYPE |
| 41–42 | 2 numeric characters | Transaction Code | TRAN-CODE |
| 43 | 1 numeric character | Processing Flag 1 | PROC-FLAGS(1) |
| 44 | 1 numeric character | Processing Flag 2 | PROC-FLAGS(2) |
| 45 | 1 numeric character | Processing Flag 3 | PROC-FLAGS(3) |
| 46–48 | 3 numeric characters | Response Code | RESP-CODE |

## Positions 1–2 — Device Type

**Internal Field:** DEV-ADDR

A code that can be used by controllers on nonswitched lines to identify individual terminals. This code is also used by the XPNET process to route messages from dial-up terminals to the appropriate software process (SPDH module). Valid values are as follows:

9. = Dial or leased line terminal or network
Other = Not used by BASE24

## Positions 3–4 — Transmission Number

**Internal Field:** XMIT-NUM

A two-digit number used by the SPDH module to detect and drop duplicate requests. If this field is nonzero, and if it is equal to the transmission number of the last request and the previous response was an approval, the SPDH module drops the request and sends a response indicating that it received a duplicate request and that it was dropped. This field is right-justified and zero filled. Valid values are as follows:

00 = Transmission number not checked
Other = Transmission number checked for duplicates

## Positions 5–20 — Terminal ID

**Internal Field:** TERM-ID

A value that uniquely identifies the terminal. It is used by the SPDH module as the key to the appropriate record in the BASE24-pos Terminal Data files (PTD). This value must be manually entered at the terminal initially and whenever the value has been corrupted. This field is left-justified and blank filled.

## Positions 21–26 — Employee ID

**Internal Field:** CLERK-ID

One to six alphanumeric characters that uniquely identify the employee entering the transaction. It is used by the SPDH module to maintain employee totals. This field is left-justified and blank filled. For request headers, the valid values are as follows:

␣␣␣␣␣␣ = Employee ID is not validated (where ␣ is a space)
Other = Employee ID validated against BASE24-pos Terminal Data files (PTD)

For response headers, this information is echoed from the request.

In order for the employee totals to be accumulated, additional fields must be set in the BASE24-pos Terminal Data files (PTD). Values in the CLERK ID field on PTD screen 1, along with the CLERK TOTALS FLAG field on PTD screen 5 and the CLERK TOTALS field on PTD screen 6 determine whether clerk totals are supported. Clerk total counts and amounts can be kept for debit, credit, adjustment, cash back, and check transactions. For more information on the BASE24-pos Terminal Data files (PTD) fields used for clerk totals, refer to the ***BASE24-pos Files Maintenance Manual***.

## Positions 27–32 — Current Date

**Internal Field:** DAT

The date (YYMMDD) of the transaction. For requests, this field is optional and contains the local terminal date. If it is not provided in the request, or is invalid or incorrect, the SPDH defaults to the current system date. For responses, the SPDH returns the current system date taking into account differing time zones. The value in this field must be echoed in terminal-generated reversals.

If this field contains a value in the request, the Current Time field also must contain a valid value. The device must send both fields if the current system date and time of the host are not to be used.

In message authentication code (MAC) reversal messages from the terminal, the value in this field must match the date on the response being reversed.

## Positions 33–38 — Current Time

**Internal Field:** TIM

The time (hhmmss) of the transaction, where 000000 is midnight. For requests, this field is optional and contains the local terminal time. If it is not provided in the request, or is invalid or incorrect, the SPDH defaults to the current system time. For responses, the SPDH returns the current system time taking into account differing time zones. The value in this field must be echoed in terminal-generated reversals.

If this field contains a value in the request, the Current Date field also must contain a valid value. The device must send both fields if the current system date and time of the host are not to be used.

In message authentication code (MAC) reversal messages from the terminal, the value in this field must match the time on the response being reversed.

## Position 39 — Message Type

**Internal Field:** REQ-TYPE

A code that identifies whether the transaction is financial or administrative. Together with the value in the Transaction Code field, this field identifies the type of transaction to the SPDH module. A listing of the various combinations is included later in this section. Valid values are as follows:

A = Administrative transaction
F = Financial transaction
L = Pass-through administrative transaction (formerly called merchant link)
M = Pass-through financial transaction (formerly called merchant link)

## Position 40 — Message Subtype

**Internal Field:** REQ-SUBTYPE

A code that identifies the message as being either online, store-and-forward, force-post, or a reversal. Reversals are generated by the terminal or controller, or because of a message authentication code (MAC) problem. Valid values are as follows. All values are uppercase letters.

A = Timeout reversal—advice. Placed in this field by the terminal or controller when a store-and-forward transaction must be reversed because of a timeout.
C = Terminal or controller reversal. Placed in this field by the terminal or the controller when the transaction must be reversed. The original response message is then sent back to the SPDH module, with the exception of this field, thus reversing the transaction. A controller might place a C in this field if the controller is unable to send the response to the individual terminal that generated the transaction.
D = Terminal decryption reversal. Produced when a balance inquiry transaction must be reversed due to a data decryption error by the POS/PIN pad device. The response message is then sent back to the host as received with this field set to a value of D to reverse the transaction. This value is used only with balance inquiry transactions when the available balance cannot be decrypted.

E = Europay, MasterCard, and Visa (EMV) chip card log-only. Indicates that the transaction was approved by the terminal or EMV card offline. When an EMV transaction is authorized offline by the terminal/EMV card, the card generates a Transaction Certificate (TC). The TC is found in the Application Cryptogram (AC) field within FID 6 subFID O (EMV Request Data). The TC provides information about the transaction which can be used if the transaction is disputed. The terminal uploads EMV log-only transactions one at a time to the SPDH Device Handler module.

F = Force-post. Indicates the transaction is to be force-posted. Force posting is the manual posting of a transaction to an account. For example, if a POS device becomes inoperable but a merchant wants a transaction to be posted for settlement purposes, the transaction can be posted manually to the account.

O = Online. Indicates the terminal is online to the BASE24 system. Note: Type A messages must be subtype O.

R = MAC reversal. Placed in this field by the terminal or controller when it receives a response message containing a MAC that does not match the MAC computed by the terminal. The original response message is then sent back to SPDH module as received, with the exception of this field, thus reversing the transaction.

S = Store-and-forward. Indicates the terminal or controller was offline or otherwise not communicating with SPDH module when the transaction was initiated. The transaction was approved and held in the terminal memory until communications resumed, at which time the transaction was sent to SPDH module. BASE24-pos must accept and post the transaction.

T = Timeout reversal—online. Placed in this field by the terminal or controller when an online transaction must be reversed because of a timeout.

U = Customer-canceled reversal. Placed in this field by the terminal when a transaction is reversed by a clerk at the terminal.

V = EMV log-only cancellation. Placed in this field by the terminal or the controller when an EMV log-only transaction (subtype E) needs to be cancelled.

## Positions 41–42 — Transaction Code

**Internal Field:** TRAN-CODE

A code that identifies the transaction type associated with the message. Together with the value in the Message Type field, this code identifies the type of transaction sent to the SPDH module. A listing of the various combinations is included later in this section. Valid values for this field are as follows:

00 = Normal purchase
01 = Preauthorization purchase
02 = Preauthorization purchase completion
03 = Mail or telephone order
04 = Merchandise return
05 = Cash advance
06 = Card verification
07 = Balance inquiry
08 = Purchase with cash back
09 = Check verification
10 = Check guarantee
11 = Purchase adjustment
12 = Merchandise return adjustment
13 = Cash advance adjustment
14 = Cash back adjustment
15 = Card activation
16 = Additional card activation
17 = Replenishment
18 = Full redemption
50 = Logon request
51 = Logoff request
60 = Close batch request
61 = Close shift request
62 = Close day request
64 = Employee subtotals request
65 = Batch subtotals request
66 = Shift subtotals request
67 = Day subtotals request
70 = Read mail request
71 = Mail delivered request
75 = Send mail request
90 = Download request
95 = Handshake request
96 = Key change request

# Position 43 — Processing Flag 1

**Internal Field:** PROC-FLAGS(1)

A code that is used in requests to direct the SPDH module to disconnect after its response, or to remain connected for more transactions.

If this field is set to not disconnect the terminal, the SPDH module maintains the line indefinitely. In dial-up environments, it is the responsibility of the terminal to initiate a disconnect, either by sending a request (e.g., a Handshake request) with this field set to disconnect, or by dropping the line. If the terminal disconnects by dropping the line, this can be treated as an error condition by the protocol.

In general, when using Visa or POS-2 protocols, the terminal does not know if subsequent requests are pending and should set this field to disconnect. When the protocol receives a message from the terminal, the protocol determines whether more requests exist for the SPDH module in its queue before the SPDH module issues a disconnect message to the terminal.

Furthermore, the terminal cannot determine if subsequent requests are pending during a download if the terminal cannot determine whether the SPDH module response indicates the presence or absence of more download data. The SPDH module sends a disconnect message to the terminal if the SPDH module determines there is no more download data.

An example of this situation is a full download that requires multiple request and response messages. For downloads, BASE24 sends an 880 response code indicating that no more data exists or an 881 response code indicating that more data is forthcoming. Thus, this field should be set to disconnect for downloads. Then, when there is no more data to be sent, the SPDH module issues a disconnect message to the terminal. For requests, the valid values are as follows:

0 = Respond and disconnect
1 = Respond and do not disconnect

This field is used in responses to inform the terminal of waiting mail. For responses, the valid values are as follows:

0 = No mail waiting
1 = Mail waiting

## Position 44 — Processing Flag 2

**Internal Field:** PROC-FLAGS(2)

The use of this code depends on the value in the Message Type field.

For pass-through administrative transactions (message type L), this code is used in requests to indicate whether concurrent processing can be used. The valid values are as follows:

0 = No, transactions cannot be processed concurrently.
1 = Yes, transactions can be processed concurrently.

If concurrent processing is used, the PIN Pad ID will be capable of processing multiple credit card transactions at the same time.

For all other transactions, this code is used in requests to indicate whether the POS device is EMV capable. The valid values are as follows:

0 = No, the terminal is not EMV capable.
5 = Yes, the terminal is EMV capable.

For all transactions, this code is used in responses to inform the terminal that it should request a download. The valid values are as follows:

0 = No download waiting
1 = Download waiting

## Position 45 — Processing Flag 3

**Internal Field:** PROC-FLAGS(3)

The use of this code depends on the value in the Message Type field.

For failed pass-through administrative transactions (message type L), this code is used in requests to indicate whether concurrent processing can be used. If concurrent processing is used, the PIN Pad ID will be capable of processing multiple credit card transactions at the same time. The valid values are as follows:

0 = No, transactions cannot be processed concurrently.
1 = Yes, transactions can be processed concurrently.

For all other requests, this code is used to indicate the totals to return in response to a clerk totals request. This field is not used for responses. Valid values are as follows:

0 = All clerk totals for a specific terminal
1 = Totals for a clerk at a specific terminal
2 = Clerk totals for a specific batch for a terminal
3 = Totals for a specific clerk over all terminals

## Positions 46–48 — Response Code

**Internal Field:** RESP-CODE

This code is not used in requests.

This code is used in responses to inform the terminal of the transaction processing results. Valid values are as follows:

**Approved Codes**

000 = Approved balances available
001 = Approved no balances available
002 = Approved country club status
003 = Approved (maybe more identification is required)
004 = Approved pending identification (sign paper draft is required)
005 = Approved blind
006 = Approved VIP status
007 = Approved administrative transaction
008 = Approved national NEG hit OK
009 = Approved commercial status
010 = Approved for a lesser amount

**Declined Codes**

050 = General
051 = Expired card
052 = Number of PIN tries exceeded
053 = No sharing allowed
054 = No security module
055 = Invalid transaction
056 = Transaction not supported by institution
057 = Lost or stolen card
058 = Invalid card status
059 = Restricted status
060 = Account not found on CAF

061 = PBF record not found
062 = PBF update error
063 = Invalid authorization type in IDF
064 = Bad track information
065 = Adjustment not allowed in IDF
066 = Invalid credit card advance increment
067 = Invalid transaction date
068 = PTLF error
069 = Bad message edit
070 = No IDF
071 = Invalid routing to Authorization
072 = Card on National Negative File
073 = Invalid routing authorization service
074 = Unable to authorize
075 = Invalid PAN length
076 = Insufficient funds in PBF
077 = Preauthorization full
078 = Duplicate transaction received
079 = Maximum online refund reached
080 = Maximum offline refund reached
081 = Maximum credit per refund reached
082 = Maximum number of times used
083 = Maximum refund credit reached
084 = Customer selected NEG reason
085 = Inquiry not allowed—no balances
086 = Over floor limit
087 = Maximum number refund credits reached
088 = Place call
089 = CAF status equals 0 or 9
090 = Referral file full
091 = NEG file problem
092 = Advance less than minimum
093 = Delinquent
094 = Over limit table **Note:** Response code 094 has two meanings. For parametric authorization, the code indicates the transaction amount exceeded the amount available. For a return or adjustment transaction, the code indicates a BASE24-pos Terminal Data files (PTD) return or adjustment limit (count or amount) was exceeded.
095 = Amount over maximum
096 = PIN required
097 = Mod 10 check
098 = Force post
099 = Bad PBF

**Referral Codes**

100 = Unable to process transaction
101 = Unable to authorize—issue call
102 = Call
103 = NEG file problem
104 = CAF problem
105 = Card not supported
106 = Amount over maximum
107 = Over daily limit
108 = CAPF not found
109 = Advance less than minimum
110 = Number times used
111 = Delinquent
112 = Over limit table
113 = Timeout
115 = PTLF full
120 = Bad UAF
121 = ADMN file problem
122 = Unable to validate PIN; security module is down
130 = Authorization request cryptogram (ARQC) referral
131 = Card verification results (CVR) referral
132 = Terminal verification results (TVR) referral
133 = Reason online code referral
134 = Fallback referral

**Service Code**

150 = Merchant not on file

**Transaction Error Codes**

200 = Invalid account
201 = Incorrect PIN
202 = Cash advance is less than minimum
203 = Administrative card needed
204 = Enter lesser amount **Note:** Response code 204 has two meanings. This code can also be used when the transaction amount exceeds the POS Retailer Definition File (PRDF) ceiling limits.
205 = Invalid advance amount
206 = CAF not found
207 = Invalid transaction date
208 = Invalid expiration date
209 = Invalid transaction code
251 = Cash back exceeds daily limit
400 = Authorization request cryptogram (ARQC) failure
401 = Hardware security module parameter error

402 = Hardware security module failure
403 = ICC Key File (KEYI) not found
404 = Application transaction counter (ATC) check failure
405 = Card verification results (CVR) decline
406 = Terminal verification results (TVR) decline
407 = Reason online code decline
408 = Fallback decline
800 = Format error
801 = Invalid data
802 = Invalid employee number
809 = Invalid close transaction
810 = Transaction timeout
811 = System error
820 = Invalid terminal identifier
821 = Invalid response length

**Mail and Download Codes**

870 = Mail delivered
871 = Mail stored
880 = Mail message has been received in its entirety
881 = Mail message received successfully and there is more data for this mail message
880 = Download has been received in its entirety
881 = Download received successfully and there is more data for this download
882 = Download aborted (call for service)

**Decline Codes**

878 = Incorrect PIN length error
889 = MAC communications key (KMAC) synchronization error
898 = Invalid MAC
899 = Sequence error—resync

**POS Capture Codes**

900 = Number of PIN tries exceeded
901 = Expired card
902 = NEG capture code
903 = CAF status 3
904 = Advance less than minimum
905 = Number times used exceeded
906 = Delinquent
907 = Over limit table
908 = Amount over maximum
909 = Capture

910 = Authorization request cryptogram (ARQC) capture
911 = Card verification results (CVR) capture
912 = Terminal verification results (TVR) capture

**Decline Administrative Card—Call Required Codes**
950 = Administrative card not found
951 = Administrative card not allowed
959 = Administrative transactions not supported

**Approved Administrative Codes**
952 = Approved administrative request—in window
953 = Approved administrative request—out of window
954 = Approved administrative request—anytime

**Chargeback Codes**
955 = Chargeback—customer file updated
956 = Chargeback—customer file updated—acquirer not found
957 = Chargeback—incorrect prefix number
958 = Chargeback—incorrect response code or CPF configuration
960 = Chargeback—approved customer file not updated
961 = Chargeback—approved customer file not updated, acquirer not found
962 = Chargeback—accepted, incorrect destination

# Optional Data Fields

After the standard message header, the remaining portion of the message consists of a series of optional data fields. Optional data fields can be included in requests from the terminal and responses from the SPDH module for each transaction type and are identified in the system by Field Identifiers (FIDs). FIDs are specified by the customer in the ACI Standard Device Configuration File (ACNF).

Optional data fields are summarized in a table below, followed by individual descriptions of the fields.

**Note:** Optional data fields 6 through 9 contain subfields that supply additional optional information. These subfields are identified by Subfield Identifiers (SFIDs). For more information about these subfields, see the "Optional Data Subfield" topics later in this section.

## Summary Table

The optional data fields are summarized in the table below in order by FID, capital letters first, followed by lowercase letters, followed by numbers. The table lists the FID, the length of the field in the message, its associated field name, and the name of the internal field to which it is mapped by the SPDH module. In addition, a check mark (✓) appears in the RQST or RESP columns if the optional data field is available for requests or responses, respectively.

| FID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| A | 1 to 20 bytes | Customer Billing Address | PSTM.ADDR-FLDS.ADDR | ✓ | ✓ |
| B | 1 to 18 bytes | Amount 1 | PSTM.TRAN.AMT-1 | ✓ | ✓ |
| C | 1 to 18 bytes | Amount 2 | PSTM.TRAN.AMT-2 or AMT-OTHER field of EMV Request Data token | ✓ | ✓ |
| D | 1 byte | Application Account Type | PSTM.TRAN.TRAN-CDE.AA | ✓ | ✓ |
| E | 1 to 19 bytes | Application Account Number | PSTM.TRAN.ACCT | | ✓ |
| F | 8 bytes | Approval Code | PSTM.TRAN.APPRV-CDE | ✓ | ✓ |

| FID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| G | 8 bytes | Authentication Code | Not applicable | ✓ | ✓ |
| H | 16 to 48 bytes | Authentication Key | PTDD1.PTDD1-CORE.MAC-DATA.ENCR-KEYS.MAC-KEY | | ✓ |
| I | 16 to 48 bytes | Data Encryption Key | Not applicable | | ✓ |
| J | 18 bytes | Available Balance | PSTM.TRAN.AMT-1 or POS Balances token | | ✓ |
| K | 6 bytes | Business Date | PSTM.USER-DATA | ✓ | ✓ |
| L | 1 byte | Check Type | PSTM.TRAN.TRAN-CDE.C | ✓ | ✓ |
| M | 16 to 48 bytes | Communications Key | PTDD1.PTDD1-CORE. ENCR-PIN.ENCR-KEYS.P-KEY | | ✓ |
| N | 1 to 40 bytes | Customer ID | Check Guarantee/Verification token | ✓ | ✓ |
| O | 2 bytes | Customer ID Type | Check Guarantee/Verification token | ✓ | ✓ |
| P | 1 byte | Draft Capture Flag | PSTM.TRAN.DFT-CAPTURE-FLG | ✓ | ✓ |
| Q | 1 to 16 bytes | Echo Data | PSTM.USER-DATA | ✓ | ✓ |
| R | 1 byte | Card Type | PSTM.TRAN.TRAN-CDE.T | ✓ | ✓ |
| S | 1 to 10 bytes | Invoice Number | PSTM.INVOICE-NUM | ✓ | ✓ |
| T | 1 to 10 bytes | Invoice Number/ Original | PSTM.ORIG-INVOICE-NUM | ✓ | ✓ |
| U | 1 byte | Language Code | ARSP.PRIKEY.REC-NUM | ✓ | ✓ |
| V | 15 bytes | Mail/Download Key | PTDD1.PTDD1-CORE. MAIL-DATA.MAIL-MSG | ✓ | ✓ |

| FID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| W | 1 to 957 bytes | Mail Text/ Download Data | PTDD1.PTDD1-CORE. MAIL-DATA.MAIL-MSG | ✓ | ✓ |
| X | 3 bytes | ISO Response Code | PSTM.TRAN.ICHG-RESP | | ✓ |
| Y | 1 to 9 bytes | Customer ZIP Code | PSTM.ZIP-CDE | ✓ | ✓ |
| Z | 1 byte | Address Verification Status Code | PSTM.ADDR-FLDS.ADDR-VRFY-STAT | | ✓ |
| a | 1 to 250 bytes | Optional Data | PSTM.USER-DATA or Optional Data token | ✓ | ✓ |
| b | 16 bytes | PIN/Customer | PSTM.PIN | ✓ | |
| c | 16 bytes | PIN/Supervisor | PSTM.PIN | ✓ | |
| d | 1 to 12 bytes | Retailer ID | PSTM.USER-DATA.INFO | ✓ | ✓ |
| e | 2 bytes | POS Condition Code | PSTM.PT-SRV-COND-CDE | ✓ | ✓ |
| f | 1 to 200 bytes | PIN Length or Receipt Data | PSTM.PIN-SIZE | ✓ | ✓ |
| g | 1 to 48 bytes | Response Display | ARSP.DISPLAYS.TEXTS or RCDF.DESCR | | ✓ |
| h | 10 bytes | Sequence Number | PTDD1.PTDD1-CORE. SHIFT-NUM<br>PTDD1.PTDD1-CORE. BATCH-NUM<br>PTDD1.PTDD1-CORE. SEQ-NUM<br>PSTM.USER-DATA | ✓ | ✓ |
| i | 9 bytes | Sequence Number/ Original | PSTM.USER-DATA | ✓ | ✓ |

| FID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| j | 2 bytes | State Code | STATE-CDE or<br>Check Guarantee/Verification token | ✓ | |
| k | 0 to 25 bytes | Birth Date/Drivers License Expiration Date (for requests)/ Terminal Location (for responses) | Check Guarantee/Verification token | ✓ | ✓ |
| l | 75 bytes | Totals/Batch | PTDD1.PTDD1-CORE.<br>BATCH<br>PTDD1.PTDD1-CORE.<br>SHIFT<br>PTDD1.PTDD1-CORE.<br>BATCH-DC<br>PTDD1.PTDD1-CORE.<br>SHIFT-DC | ✓ | ✓ |
| m | 75 bytes | Totals/Day | PTDD1.PTDD1-CORE.<br>BATCH<br>PTDD1.PTDD1-CORE.<br>BATCH-DC<br>PTDD1.PTDD1-CORE.<br>SHIFT<br>PTDD1.PTDD1-CORE.<br>SHIFT-DC<br>PTDD1.PTDD1-CORE.<br>DAILY<br>PTDD1.PTDD1-CORE.<br>DAILY-DC | ✓ | ✓ |
| n | 75 bytes | Totals/Employee | PTDD1.PTDD1-CORE.<br>SHIFT-NUM<br>PTDD1.PTDD1-CORE.<br>BATCH-NUM<br>PTDD1.PTDD1-CORE.<br>CLERK-TOTS | ✓ | ✓ |

| FID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| o | 75 bytes | Totals/Shift | PTDD1.PTDD1-CORE.<br>BATCH<br>PTDD1.PTDD1-CORE.<br>BATCH-DC<br>PTDD1.PTDD1-CORE.<br>SHIFT<br>PTDD1.PTDD1-CORE.<br>SHIFT-DC | ✓ | ✓ |
| q | 1 to 40 bytes | Track 2/Customer | PSTM.TRACK2 | ✓ | ✓ |
| r | 1 to 40 bytes | Track 2/Supervisor | PSTM.TRACK2 | ✓ | ✓ |
| s | 1 to 24 bytes | Transaction Description | ARSP.DESCR.TBL | | ✓ |
| t | 16 bytes | PIN Pad Identifier | POS Merchant Token | ✓ | ✓ |
| u | 6 bytes | Acceptor Posting Date | POS Merchant Token | | ✓ |
| 0 | 46 to 118 bytes | American Express Data Collection | American Express token | ✓ | ✓ |
| 1 | 24 bytes | PS2000 Data | PS2000 token | ✓ | ✓ |
| 2 | 1 to 82 bytes | Track 1/Customer | PSTM.TRACK2 or<br>Track 1 token | ✓ | ✓ |
| 3 | 1 to 82 bytes | Track 1/Supervisor | PSTM.TRACK2 or<br>Track 1 token | ✓ | ✓ |
| 4 | 156 to 171 bytes | Industry Data | POS Industry Data token | ✓ | ✓ |
| 6 | variable | Product SubFIDs | Refer to subFIDs | ✓ | ✓ |
| 7 | variable | Product SubFIDs | Refer to subFIDs | ✓ | ✓ |

| FID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| 8 | variable | Product SubFIDs | Refer to subFIDs | ✓ | ✓ |
| 9 | variable | Customer SubFIDs | Not Applicable | ✓ | ✓ |

## FID A — Billing Address

**Request:** Optional. Variable length of 1 to 20 bytes.

**Response:** Optional. Fixed length of 20 bytes. If this field is included in the request message, then it can be echoed in the response.

**Internal Field:** PSTM.ADDR-FLDS.ADDR

The Billing Address is the cardholder's billing address. This address is used for address verification.

## FID B — Amount 1

**Request:** Required for the following transactions. Variable length of 1 to 18 bytes.

Normal purchase
Preauthorization purchase
Preauthorization purchase completion
Mail or telephone order
Merchandise return
Cash advance
Purchase with cash back
Check guarantee
Purchase adjustment
Merchandise return adjustment
Cash advance adjustment
Cash back adjustment
Mobile top-up cash
Mobile top-up with funds

**Response:** Optional. Fixed length of 18 bytes. If included in the request, the value can be echoed in the response. This field is right-justified and zero filled.

**Internal Field:** PSTM.TRAN.AMT-1

Amount 1 is the primary amount field. For transactions that involve one amount, this is the transaction amount. For transactions that involve more than one amount, this is the original or total amount.

For an EMV request, the data in this field is used for the AMT-AUTH field of the EMV Request Data token.

## FID C — Amount 2

**Request:** Required for the following transactions. Variable length of 1 to 18 bytes.

Purchase with cash back
Purchase adjustment
Merchandise return adjustment
Cash advance adjustment
Cash back adjustment

**Response:** Optional. Fixed length of 18 bytes. If included in the request, it can be echoed in the response. This field is right-justified and zero filled.

**Internal Field:** PSTM.TRAN.AMT-2
AMT-OTHER field of EMV Request Data token

Amount 2 is the secondary amount field.

For transactions with two amounts involved, this field contains the revised amount.

For transactions with cash back, this field contains the cash back amount.

For an EMV request, this field is used for the AMT-OTHER field of the EMV Request Data token.

For a passthrough financial transaction (message type M) that is a preauthorization with cash back or a preauthorization completion with cash back, this field contains the original transaction amount.

## FID D — Application Account Type

**Request:** Optional. Fixed length of 1 byte.

**Response:** Optional. Fixed length of 1 byte. If this field is included in the request message, it can be echoed in the response. If this field is not in the request message, it reflects the default account type in a host database.

**Internal Field:** PSTM.TRAN.TRAN-CDE.AA

The Application Account Type indicates the type of account for the transaction. If this field is not submitted by the terminal, BASE24-pos uses the DEFAULT ACCOUNT TYPE field on Card Prefix File (CPF) screen 7. Valid values are as follows:

0 = None (use default account type on host)
1 = Checking
2 = Savings
4 = Credit
5 = iDebit
8 = Food stamps
9 = Cash benefit

## FID E — Application Account Number

**Request:** Not available.

**Response:** Optional. Variable length of 1 to 19 bytes. Trailing blanks are removed. This field is left-justified and blank filled.

**Internal Field:** PSTM.TRAN.ACCT

The Application Account Number indicates the account against which the transaction was processed. This field can originate with BASE24-pos or with a host.

## FID F — Approval Code

**Request:** Optional. Fixed length of 8 bytes.

**Response:** Optional. Fixed length of 8 bytes. This field is left-justified and blank filled. If this field is included in the request message, then it can be echoed in the response. If this field is not included in the request message, it can be generated by BASE24-pos, a host, or an interchange.

**Internal Field:** PSTM.TRAN.APPRV-CDE

The Approval Code represents a unique number generated by the authorizer for the transaction.

## FID G — Authentication Code

**Request:** Optional. Fixed length of 8 bytes. If included, the MAC is verified.

**Response:** Optional. If responses are verified using MACs, this field is required. Fixed length of 8 bytes.

**Internal Field:** Not Applicable

The Authentication Code contains the eight-byte hexadecimal message authentication code (MAC) used to verify the message when MACs are being used.

## FID H — Authentication Key

**Request:** Not available.

**Response:** Optional. Variable length of 16, 32, or 48 bytes. Trailing blanks are removed. This field can be configured in any response and is included if more than the configured number of MAC errors occur. This field can also be included optionally in downloads.

**Internal Field:** PTDD1.PTDD1-CORE.MAC-DATA.ENCR-KEYS.MAC-KEY

The Authentication Key is a working key generated by the host and provided to the terminal. The authentication key is the MAC communications key (KMAC), encrypted under the MAC terminal master key. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

## FID I — Data Encryption Key

**Request:** Not available.

**Response:** Optional. Variable length of 16, 32, or 48 bytes. This field can be configured in any response and is included if more than the configured number of message encryption errors occur. This field can also be included optionally in downloads.

**Internal Field:** Not Applicable

The Data Encryption Key is a working key generated by the security module and provided to the terminal. The data encryption key is the data encryption communications key (KME) encrypted under the data encryption terminal master key. This key is used to encrypt the Available Balance field. The Data Encryption Key is also used for full message encryption and configurable message encryption. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

## FID J — Available Balance

**Request:** Not available.

**Response:** Optional. Fixed length of 18 bytes. The Available Balance can originate with the authorizer. This field is right-justified and zero filled.

**Internal Field:** PSTM.TRAN.AMT-1 or POS Balances token

The Available Balance is the amount available to the customer from the account against which a debit or credit card transaction was processed. The amount can be encrypted or in the clear. If the PTD is configured to return balances on all transactions, this FID contains balance information from the TXN-AMT-1 field in the POS Balances token.

The Available Balance can also be used for the available balance returned from the mobile operator in a mobile top-up transaction.

If the CPF is configured to return balances on purchase transactions, this FID contains balance information from the TXN-AMT-1 field in the POS Balances token for purchase, purchase with cash back, preauthorization purchase, and balance inquiry transactions.

If the PTD is not configured to return balances on all transactions and the CPF is not configured to return balances on purchase transactions, this FID is available in responses only on balance inquiry transactions. If configured in other responses, it is zero filled. With the Multiple Currency add-on product, the SPDH module can use the value in the POS Balances token for FID J. Otherwise, the SPDH module uses only the TRAN.AMT-1 field in the PSTM for FID J.

## FID K — Business Date

**Request:** Optional. Fixed length of 6 bytes. If this field is not included in the subtotals request, totals for the BASE24-pos Terminal Data files (PTD) posting date are returned.

**Response:** Optional. Fixed length of 6 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.USER-DATA

The Business Date (YYMMDD) allows terminals to specify an effective posting date for a transaction (other than the current BASE24-pos transaction log date). This date is recorded in the BASE24-pos log and it can vary from the date in the standard header.

## FID L — Check Type/Category

**Request:** Optional. Fixed length of 1 byte. Required for a check verification or check guarantee transaction.

**Response:** Optional. Fixed length of 1 byte. If included, the value is echoed from the request.

**Internal Field:** PSTM.TRAN.TRAN-CDE.C

For check verification or check guarantee transactions, the Check Type/Category field indicates the type of check involved. For other financial transactions, this field contains the transaction category.

**Financial Transaction Categories**

0 = Unspecified check/normal transaction
1 = Sales draft
2 = Representation
3 = Chargeback

**Check Types**

4 = Personal check for cash
5 = Personal check for purchase
6 = Personal check for purchase and cash
7 = Government
8 = Payroll
9 = Electronic

## FID M — PIN Communications Key

**Request:** Not available.

**Response:** Optional. Variable length of 16, 32, or 48 bytes. Trailing blanks are removed.

**Internal Field:** PTDD1.PTDD1-CORE.ENCR-PIN.ENCR-KEYS.P-KEY

The PIN Communications Key is a working key generated by the host or security module and provided to the terminal. This key is the PIN communications key (KPE), encrypted under the terminal master key. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

## FID N — Customer ID

**Request:** Optional. Variable length of 1 to 40 bytes. Trailing blanks are removed.

**Response:** Optional. Variable length of 1 to 40 bytes. If included, the value is echoed from the request. Trailing blanks are removed.

**Internal Field:** Check Guarantee/Verification token

The Customer ID identifies the customer in a check guarantee or check verification transaction. The value in this field could be a social security number, a drivers license number, or another type of identification. The value in this field is logged by BASE24-pos.

## FID O — Customer ID Type

**Request:** Optional. Fixed length of 2 bytes. If this FID is not present in a request, the value in the DFLT CHECK ID field on PTD screen 1 is used.

**Response:** Optional. Fixed length of 2 bytes. If included, the value is echoed from the request.

**Internal Field:** Check Guarantee/Verification token

The Customer ID Type specifies the type of identification used in the Customer ID field. The values in these fields are logged by BASE24-pos. Valid values are as follows:

00 = None
01 = Credit card
02 = Drivers license
03 = Checking account number
04 = Debit card
05 = Proprietary check cashing card
06 = State ID number
07 = Social security number
08 = Student ID number
09 = Employee ID
10 = Passport number
12–50 = Reserved for national use
51–75 = Reserved for ISO use
76–99 = Reserved for private use

## FID P —Draft Capture Flag

**Request:** Optional. Fixed length of 1 byte.

**Response:** Optional. Fixed length of 1 byte. This field in the response represents the draft capture flag used for transactions by BASE24-pos.

**Internal Field:** PSTM.TRAN.DFT-CAPTURE-FLG

The Draft Capture Flag can be specified by the terminal. However, the transaction profile kept in the BASE24-pos Terminal Data files (PTD) overrides the value in this field, unless the PTD transaction profile is 3 (terminal determines data capture mode for each transaction), in which case the SPDH module uses the value in this field. Values for this field would typically originate in data downloaded to the terminal by card prefix ranges. Valid values are as follows:

0 = Authorize only
1 = Authorize and capture

## FID Q — Echo Data

**Request:** Optional. Variable length of 1 to 16 bytes. The data is padded with trailing blanks.

**Response:** Optional. Variable length of 1 to 16 bytes. If included, the value is echoed from the request. Trailing blanks are removed.

**Internal Field:** PSTM.USER-DATA

The Echo Data represents data that the terminal requires to be echoed back to it in the response. BASE24-pos does not edit this field. It is recorded in the transaction log data and returned in the response to the terminal.

## FID R — Card Type

**Request:** Optional for credit or debit card. Mandatory for mobile top-up transactions using cash or funds. Fixed length of 1 byte. If this field is not submitted in a request, BASE24-pos uses a default value from the DEFAULT COMBO CARD TYPE field on Card Prefix File (CPF) screen 7 for combination cards. If this field is submitted for a non-combination type card, BASE24-pos ignores it and uses the card type from the CPF.

**Response:** Optional for credit or debit card. Mandatory for mobile top-up transactions using cash or funds. Fixed length of 1 byte. The value in this field is the card type as determined by BASE24-pos.

**Internal Field:** PSTM.TRAN.TRAN-CDE.T

The Card Type enables the terminal to specify the intended usage for a card used as a debit card and a credit card. Valid values are as follows:

C = Credit card
D = Debit card
N = No card type. Used with mobile top-up transactions using cash.

## FID S — Invoice Number

**Request:** Optional. Variable length of 1 to 10 bytes.

**Response:** Optional. Fixed length of 10 bytes. If included, the value is echoed from the request. This field is left-justified and blank filled.

**Internal Field:** PSTM.INVOICE-NUM

The Invoice Number enables a terminal to submit a unique stamp to further identify a transaction.

## FID T — Invoice Number/Original

**Request:** Optional. Variable length of 1 to 10 bytes. This field may be required by some interfaces or authorizing entities.

**Response:** Optional. Fixed length of 10 bytes. If included, the value is echoed from the request. This field is left-justified and blank filled.

**Internal Field:** PSTM.ORIG-INVOICE-NUM

The Invoice Number/Original field enables the terminal to uniquely identify a previous transaction that is now being adjusted.

## FID U — Language Code

**Request:** Optional. Fixed length of 1 byte.

**Response:** Optional. Fixed length of 1 byte. If included in both the request and response, this field echoes the request. If this field was not included in the request, then this field represents the default language code for the terminal from the BASE24-pos Terminal Data files (PTD).

**Internal Field:** ARSP.PRIKEY.REC-NUM

The Language Code enables the terminal to override the default language in the BASE24-pos Terminal Data files (PTD). It is used in formulating a terminal display response. The terminal can select one of three different language displays. Valid values are as follows:

0 = Table 1
1 = Table 2
2 = Table 3

## FID V — Mail/Download Key

**Request:** Optional. Fixed length of 15 bytes. Required for a read mail request transaction and a mail delivered transaction. Required for downloads except on initial requests.

**Response:** Optional. Fixed length of 15 bytes. Required for a read mail request transaction and a mail delivered transaction. Required for downloads except on initial requests.

**Internal Field:** PTDD1.PTDD1-CORE.MAIL-DATA.MAIL-MSG

The Mail/Download Key consists of a group of fields that identify mail to be read or marked as delivered, or the type of download to be performed.

Concerning mail, this group identifies a mail category (user-defined), the type of mail access desired (any mail, only undelivered mail, specific mail), a flag used for mail broadcast processing, and a mail ID. The mail ID serves to identify a specific piece of mail or the ID of the last mail read. The format is as follows:

| | |
|---|---|
| Category | X(2) |
| Access Code | 9(1) |
| Processing Flag | X(2) |
| Mail Date | 9(6) |
| Mail ID | 9(4) |

Concerning downloading, this group specifies either a full or partial download. If full, this group identifies whether this is the initial request or a continuation request (when the download spans several request/response pairs).

If partial, this group also identifies which download field is requested. The format is as follows:

| | |
|---|---|
| Category | X(2) |
| Access Code | X |
| Processing Flag | X(2) |
| Filler | X(10) |

**Note:** In continuation requests (i.e., read next mail or continue download), the terminal should echo the value contained in the previous response.

## FID W — Mail/Download Text

**Request:** Optional. Required for a send mail transaction request. Variable length of 1 to 449 bytes.

**Response:** Optional. Variable length of 1 to 449 bytes for mail transactions and 1 to 957 bytes for download text. With responses, only the text of the mail itself is used. If no mail was found, this field is not included in the response, even if the customer configuration supports read mail responses.

**Internal Field:** PTDD1.PTDD1-CORE.MAIL-DATA.MAIL-MSG

The Mail/Download Text field consists of a group of fields representing the text of a send mail request (from the terminal to the host) or the download fields from the host. Concerning mail, this group contains a maximum of 449 bytes and identifies the destination DPC number (when more than one exists) and the text itself.

The format is as follows:

| Position | Length | Description |
|---|---|---|
| **01** | **1** | **Destination DPC Number Code**<br>The values for the DPC Number Code are 0, 1, and 2. These values map to different four-character DPC numbers listed in the host application database. |
| **02–449** | **448** | **Mail Text (variable-length, 448 bytes maximum)**<br>Text of the mail message. |

Concerning downloading, this group contains the fields being downloaded from the ACNF. The group contains a maximum of 956 bytes of download data. It identifies the destination DPC number (when more than one exists) and the text itself. The format is as follows:

| Position | Length | Description |
|---|---|---|
| **01** | **1** | **Destination DPC Number Code**<br>This field is not used for downloads and contains a value of 0. |
| **02–957** | **956** | **Download Text (variable-length, 956 bytes maximum)**<br>Text being downloaded. |

## FID X — ISO Response Code

**Request:** Not available.

**Response:** Optional. Fixed length of 3 bytes.

**Internal Field:** PSTM.TRAN.ICHG-RESP

The ISO Response Code is the ISO equivalent of the BASE24 response code found in the SPDH message header. Used to inform the terminal of transaction processing results. The SPDH module uses the SPDH Names File (SPDHNAMS) to map BASE24 response codes to their ISO equivalents.

The ISO equivalent can be a three-digit value based on the International Organization for Standardization (ISO) 8583-1993 standard or a two-digit value based on the 8583-1987 standard. If used, the two-digit value must be left-justified with a trailing blank.

## FID Y — Postal (ZIP) Code

**Request:** Optional. Fixed length of 9 bytes.

**Response:** Optional. Fixed length of 9 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.ZIP-CDE

The Postal (ZIP) Code field contains the ZIP code of the cardholder's billing address. The ZIP code should be either five or nine digits in length. If it is five digits in length, the digits should be left-justified and the remaining positions are space filled.

## FID Z — Address Verification Status Code

**Request:** Not available.

**Response:** Optional. Fixed length of 1 byte.

**Internal Field:** PSTM.ADDR-FLDS.ADDR-VRFY-STAT

For responses, this FID contains the address verification status code. The address verification status code identifies the results of comparing address verification information received in the transaction and address verification information contained in the processor's database.

The following values can be set by BASE24-pos through its address verification processing. If a transaction is sent to an interchange for processing, the response message may contain other interchange-specific values (BASE24 does not change values received from the interchanges).

A = Address. Addresses matched, but ZIP codes did not match.
E = The transaction was either not eligible for address verification or an error occurred while attempting to process the message.
N = Error. Neither the addresses nor the ZIP codes matched.
R = Retry. The primary and secondary authorizers were either unavailable or they declined the transaction and address verification was not performed by BASE24-pos.
S = Service not supported. BASE24-pos authorized the transaction, but does not support the BASE24-pos Address Verification product.
U = Unavailable. Address information was not available to the processor performing address verification.
W = Whole ZIP. The nine-digit ZIP code matched, but the address did not match.
X = Exact. Both the addresses and the nine-digit ZIP codes matched.
Y = Yes. Both the addresses and the five-digit ZIP codes matched.

Z = ZIP. The five-digit ZIP codes matched, but the addresses did not match.

b = Address verification information was not included in the transaction.

0 = Address verification information was included in the transaction, but was not verified. This code is used by transactions to be verified by either a host or an interchange. In addition, transactions to be verified by BASE24-pos that are declined before address verification can be performed carry this code.

## FID a — Optional Data

**Request:** Optional. Variable length of 1 to 250 bytes.bytes. If the length of this field is 80 bytes or less, this field is padded with up to 80 trailing blanks and is placed in the USER-DATA field of the PSTM. If the length of the field is greater than 80 characters, no trailing blanks are added, and it is placed into the Optional Data token.

**Response:** Optional. Variable length of 1 to 250 bytes. If included, the value is echoed from the request. Trailing blanks are removed.

**Internal Field:** PSTM.USER-DATA or Optional Data token

The Optional Data field allows the terminal to exchange any type of optional data for BASE24-pos. Typically, this field is used to indicate product codes, quantities, and amounts within a total purchase amount.

When receiving a request from the POS device, the Device Handler module checks the length of the data in this field. If this field contains 80 bytes or fewer, the data is carried in the USER-DATA field of the PSTM. If the field contains more than 80 bytes, the data is carried in the Optional Data token.

When formatting a response to return to the device, the Device Handler module checks the contents of the optional data field in device-dependent data. If the device-dependent data field is not blank filled, the contents of this field are returned in the response. If the device-dependent data is blank filled or not present, the Device Handler module returns the data in the Optional Data token if present.

## FID b — PIN/Customer

**Request:** Optional. Variable length of 1 to 16 bytes. Trailing blanks are removed. This field is in PIN/PAD or PIN/PAN PIN block format. The PIN PAD character is 1 byte. Derived unique key per transaction (DUKPT) encrypted PIN blocks are also carried in this field.

**Response:** Not available.

**Internal Field:** PSTM.PIN

The PIN/Customer field contains the customer-entered PIN in a clear or encrypted form, depending on whether the terminal supports PIN encryption.

## FID c — PIN/Supervisor

**Request:** Optional. Variable length of 1 to 16 bytes. The PIN PAD character is 1 byte. Typically, this field is submitted only in certain transaction requests, like returns and adjustments. In these cases, BASE24-pos is also configured to indicate that supervisor security is to be applied to these transactions. This field is in PIN/PAD or PIN/PAN PIN block format. Derived unique key per transaction (DUKPT) encrypted PIN blocks are also carried in this field.

**Response:** Not available.

**Internal Field:** PSTM.PIN

The PIN/Supervisor field contains the supervisor-entered PIN in a clear or encrypted form, depending on whether the terminal supports PIN encryption.

## FID d — Retailer ID

**Request:** Optional. Variable length of 1 to 12 bytes. The ID value is padded with trailing blanks.

**Response:** Optional. Variable length of 1 to 12 bytes. If included, the value is echoed from the request. Trailing blanks are removed.

**Internal Field:** PSTM.USER-DATA.INFO

The Retailer ID field contains the ID assigned to a merchant group by organizations such as MasterCard, VISA, or American Express.

## FID e — POS Condition Code

**Request:** Optional. Fixed length of 2 bytes.

**Response:** Optional. Fixed length of 2 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.PT-SRV-COND-CDE

The POS Condition Code field further describes the transaction being submitted. Valid values are as follows:

00 = Normal presentment
01 = Customer not present
02 = Unattended terminal able to retain card
03 = Merchant suspicious
04 = Electronic cash register interface
05 = Customer present but card not present
06 = Preauthorization request
07 = Telephone device request
08 = Mail or telephone order
09 = Security alert
10 = Customer identity verified
11 = Suspected fraud
12 = Security reasons
13 = Representment of item
14 = Public utility terminal
15 = Customer terminal (Home terminal)
16 = Administration terminal
17 = Returned item (Chargeback)
18 = No check in envelope/all returned
19 = Deposit out-of-balance/all returned
20 = Payment out-of-balance/all returned
21 = Manual reversal
22 = Terminal error/counted
23 = Terminal error/not counted
24 = Deposit out-of-balance/applied contents
25 = Payment out-of-balance/applied contents

26 = Withdrawal had error/reversed
27 = Unattended terminal unable to retain card
28–40 = Reserved for ISO use
41–50 = Reserved for National use
51–99 = Reserved for Private use

## FID f — PIN Length or Receipt Data

**Request:** Optional. Variable length of 1 to 2 bytes representing 0–16 in binary.

**Response:** Optional. Variable length of 1 to 400 bytes. Trailing blanks are removed.

**Internal Field:** PSTM.PIN-SIZE

FID f identifies different data, depending on whether it is associated with a request or a response message. In a request message, FID f identifies the PIN length. In a response message, FID f identifies receipt data.

The PIN Length field contains the actual length of the PIN entered at the terminal.

This field can optionally contain 1 to 400 bytes of generic marketing message from the Mobile Operator File (MOF) or a marketing message from the mobile operator host.

The Receipt Data field applies only to situations where the customer has commissioned ACI to customize the SPDH module to format receipts. In this case, this field contains the receipt data. This FID is not supported for receipt data in standard BASE24-pos product.

## FID g — Response Display

**Request:** Not available.

**Response:** Optional. Variable length 1 to 48 bytes. Trailing blanks are removed.

**Internal Field:** ARSP.DISPLAYS.TEXTS or RCDF.DESCR

The Response Display field represents a 48-character display that explains the BASE24 response code contained in the standard header or the ISO response code contained in FID X. This field is extracted from one of the SPDH language tables in the ACI Standard Device Response File (ARSP) or from the Response Code Description File (RCDF).

ISO response codes must be defined in the RCDF. BASE24 response codes can be defined in the RCDF, the ARSP, or both. When a BASE response code is defined in both files, the display in the RCDF overrides the one in the ARSP.

The ARSP and the RCDF can support three languages. The language used depends on the default language code stored in the BASE24-pos Terminal Data files (PTD) or the language code included in the request.

## FID h — Sequence Number

**Request:** Optional, except for employee subtotals transaction requests for which the Sequence Number is required if totals for a specific batch are requested. Fixed length of 10 bytes.

**Response:** Optional. Fixed length of 10 bytes. When this field is included in the request message and the number is found to be valid, the value is echoed in the response if it is to be included. However, this field is returned in a response, even if it is not configured to be included, if the sequence number is included in the request, and the SPDH module finds the number to be invalid. If this field is included in the response but not in the request, BASE24 generates the sequence number internally.

**Internal Field:** PTDD1.PTDD1-CORE.SHIFT-NUM
PTDD1.PTDD1-CORE.BATCH-NUM
PTDD1.PTDD1-CORE.SEQ-NUM
PSTM.USER-DATA

The Sequence Number field consists of a group of fields. The purpose of this group of fields is to help ensure that the SPDH module receives and processes every transaction only once. The structure is shown below. Included for each field is its position in the element, its length, and a description of its contents. This group of fields consists of the following:

| Position | Length | Description |
|---|---|---|
| **01–03** | **3** | **Shift Number**<br><br>The Shift Number ranges from 001 to 999. The Shift Number increases by one digit whenever the shift is closed. It rolls to 001 after the 999th shift is closed. |
| **04–06** | **3** | **Batch Number**<br><br>The Batch Number ranges from 001 to 999. The Batch Number increases by one digit whenever the batch is closed. It rolls to 001 after the 999th batch is closed or when the shift is closed. Whenever the shift closes, the batch is also expected to close. |
| **07–09** | **3** | **Seq #**<br><br>The Seq # ranges from 001 to 999. It is a unique number for the current batch. When the Batch Number changes, the Seq # is set to 001. This portion of FID h is called Seq # so as not to confuse it with the name of FID h, which is Sequence Number. |
| **10** | **1** | **Reset Flag**<br><br>The Reset Flag indicates whether the terminal or the SPDH module is responsible for determining the correct Shift Number, Batch Number, and Seq #. Only the terminal can set the Reset Flag. Valid values are as follows:<br><br>0 = Do not reset the sequence number.<br>1 = Reset the sequence number.<br><br>When the terminal submits the Sequence Number field, the SPDH module checks the Shift Number, Batch Number, and Seq # against the expected values. For information on the actions of the SPDH module if it receives a Shift Number, Batch Number, or Seq # it does not expect, see section 4. |

## FID i — Sequence Number/Original

**Request:** Optional. Fixed length of 9 bytes. Required for preauthorization purchase transactions and preauthorization purchase completion transactions. Without this field, holds are not removed from customer accounts efficiently.

**Response:** Optional. Fixed length of 9 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.USER-DATA

The Sequence Number/Original field enables the terminal to optionally identify the sequence number of a previous transaction. For more information on this field, see FID h. FID i has the same format as positions 1 through 9 of FID h.

## FID j — State Code

**Request:** Optional. Fixed length of 2 alphanumeric bytes.

**Response:** Not available.

**Internal Field:** STATE-CDE
Check Guarantee/Verification token

The State Code is associated with the transaction request. Valid values are defined by the check verification provider.

**Note:** For electronic check authorization transactions sent to Visa, the state code entered must be a valid Visa state code.

## FID k — Birth Date/Drivers License/Terminal Location

**Request:** Optional. Fixed length of 6 bytes.

**Response:** Optional. Variable length of 1 to 25 bytes. Trailing blanks are removed.

**Internal Field:** Check Guarantee/Verification token

In requests, the Birth Date/Drivers License/Terminal Location field contains the birth date (MMDDYY) of the customer associated with the transaction. If no day was indicated on the customer ID containing the birth date, or if the date is an expiration date, the format is MMYY.

In responses, this FID contains the terminal location as indicated in the BASE24-pos Terminal Data files (PTD). This field is used primarily for Regulation E purposes.

## FID l — Totals/Batch

**Request:** Optional. Fixed length of 75 bytes.

**Response:** Optional. Fixed length of 75 bytes. Totals can be returned in any response, not just a balancing transaction.

**Internal Field:** PTDD1.PTDD1-CORE.BATCH
PTDD1.PTDD1-CORE.SHIFT
PTDD1.PTDD1-CORE.BATCH-DC
PTDD1.PTDD1-CORE.SHIFT-DC

The Totals/Batch field consists of a group of fields representing batch totals as accumulated by the terminal. This field includes a shift and batch ID, along with counts and amounts of debits, credits, and adjustments. BASE24-pos logs these totals and the totals accumulated by the SPDH module to the POS Transaction Log File (PTLF). These totals contain a sign character (+ or –) in the first byte of the amount fields. The format is as follows:

| | |
|---|---|
| Shift number | 9(3) |
| Batch number | 9(3) |
| Number of debits in batch | 9(4) |
| Amount of debits in batch | S9(16)v99 |
| Number of credits in batch | 9(4) |
| Amount of credits in batch | S9(16)v99 |
| Number of adjustments in batch | 9(4) |
| Amount of adjustments in batch | S9(16)v99 |

## FID m — Totals/Day

**Request:** Optional. Fixed length of 75 bytes.

**Response:** Optional. Fixed length of 75 bytes. Totals can be returned in any response, not just a balancing transaction.

**Internal Field:** PTDD1.PTDD1-CORE.BATCH
PTDD1.PTDD1-CORE.BATCH-DC
PTDD1.PTDD1-CORE.SHIFT
PTDD1.PTDD1-CORE.SHIFT-DC
PTDD1.PTDD1-CORE.DAILY
PTDD1.PTDD1-CORE.DAILY-DC

The Totals/Day field consists of a group of fields representing terminal day totals as accumulated by the terminal. This field includes a shift and batch count, along with counts and amounts of debits, credits, and adjustments. BASE24-pos logs these totals and the totals accumulated by the SPDH module to the POS Transaction Log File (PTLF). These totals contain a sign character (+ or –) in the first byte of the amount fields.

The format is as follows:

| | |
|---|---|
| Number of shifts in day | 9(3) |
| Number of batches in day | 9(3) |
| Number of debits in day | 9(4) |
| Amount of debits in day | S9(16)v99 |
| Number of credits in day | 9(4) |
| Amount of credits in day | S9(16)v99 |
| Number of adjustments in day | 9(4) |
| Amount of adjustments in day | S9(16)v99 |

## FID n — Totals/Employee

**Request:** Not available.

**Response:** Optional. Fixed-length of 121 bytes.

**Internal Field:** PTDD1.PTDD1-CORE.SHIFT-NUM
PTDD1.PTDD1-CORE.BATCH-NUM
PTDD1.PTDD1-CORE.CLERK-TOTS

The Totals/Employee field consists of a group of fields representing employee totals as accumulated by the terminal. This field includes a shift and batch ID, along with counts and amounts of debits, credits, adjustments, cash backs, and checks. These totals contain a sign character (+ or –) in the first byte of each amount field. The format is as follows:

| | |
|---|---|
| Current shift number | 9(3) |
| Current batch number | 9(3) |
| Number of debits for employee | 9(4) |
| Amount of debits for employee | S9(16)v99 |
| Number of credits for employee | 9(4) |
| Amount of credits for employee | S9(16)v99 |
| Number of adjustments for employee | 9(4) |
| Amount of adjustments for employee | S9(16)v99 |
| Number of cash outs for employee | 9(4) |
| Amount of cash outs for employee | S9(16)v99 |
| Number of checks for employee | 9(4) |
| Amount of checks for employee | S9(16)v99 |

## FID o — Totals/Shift

**Request:** Optional. Fixed length of 75 bytes.

**Response:** Optional. Fixed length of 75 bytes. Totals can be returned in any response, not just a balancing transaction.

**Internal Field:** PTDD1.PTDD1-CORE.BATCH
PTDD1.PTDD1-CORE.BATCH-DC
PTDD1.PTDD1-CORE.SHIFT
PTDD1.PTDD1-CORE.SHIFT-DC

The Totals/Shift field consists of a group of fields representing terminal shift totals as accumulated by the terminal. This field includes a shift ID and batch count, along with counts and amounts of debits, credits, and adjustments. BASE24-pos logs these totals and the BASE24-accumulated totals to the POS Transaction Log File. These totals contain a sign character (+ or –) in the first byte of the amount fields.

The format is as follows:

| | |
|---|---|
| Number of shifts in day | 9(3) |
| Number of batches in shift | 9(3) |
| Number of debits in shift | 9(4) |
| Amount of debits in shift | S9(16)v99 |
| Number of credits in shift | 9(4) |
| Amount of credits in shift | S9(16)v99 |
| Number of adjustments in shift | 9(4) |
| Amount of adjustments in shift | S9(16)v99 |

## FID q — Track 2/Customer

**Request:** Optional. Variable length of 1 to 40 bytes. Either this field or the Track 1/Customer field (FID 2) is required for the following transactions:

Normal purchase
Preauthorization purchase
Preauthorization purchase completion
Mail or telephone order
Merchandise return
Cash advance
Card verification
Balance inquiry
Purchase with cash back
Purchase adjustment
Merchandise return adjustment
Cash advance adjustment
Cash back adjustment
Mobile top-up with funds

The customer determines when or if Track 2 data is required on Card Prefix File (CPF) screen 1. The SPDH module checks that all financial transactions contain either FID q or FID 2 (Track 1). For more information about how the SPDH module formats track data, refer to appendix A.

**Response:** Optional. Variable length of 1 to 40 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.TRACK2

The Track 2/Customer field consists of a group of fields representing the customer's Track 2. This data can be entered manually, obtained from the terminal's card swipe reader, or obtained from a microprocessor chip on an integrated circuit card (ICC). The format indicates which of these methods was used. The format is as follows:

| | |
|---|---|
| **For manually entered data:** | X(29) |
| Entry ID (M) | X(1) |
| PAN (variable-length, 19 bytes maximum) | X(19) |
| Separator character (=) | X(1) |
| Expiration date (YYMM) | 9(4) |
| Member number | 9(3) |
| End sentinel (?) | X(1) |
| **For swiped data:** | X(40) |
| Start sentinel (;) | X(1) |
| PAN (variable-length, 19 bytes maximum) | X(19) |
| Separator character (=) | X(1) |
| Expiration date | 9(4) |
| Service Code | 9(3) |
| Discretionary data (14 bytes maximum) | X(14) |
| End sentinel (?) | X(1) |
| **For contactless magnetic stripe transaction data:** | X(40) |
| Start sentinel (;) | X(1) |
| PAN (variable-length, 19 bytes maximum) | X(19) |
| Separator character (=) | X(1) |
| Expiration date | 9(4) |
| Service Code | 9(3) |
| Discretionary data (14 bytes maximum) Note: This field can contain the information necessary for dynamic card verification (card verification value, application transaction counter, and unpredictable number). | X(14) |
| End sentinel (?) | X(1) |
| **For chip read data (if EMV Tag 57 is present):** | X(39) |
| Start sentinel (;) | X(1) |
| Track 2 equivalent data (**EMV Tag:** 57) | X(37) |
| End sentinel (?) | X(1) |
| **For chip read data (if EMV Tag 57 is not present):** | X(26) |
| Entry ID (M) | X(1) |
| PAN (variable-length, 19 bytes maximum) (**EMV Tag:** 5A) | X(19) |
| Separator character (=) | X(1) |

| | |
|---|---|
| Expiration date (YYMM) (**EMV Tag:** 5F24) | 9(4) |
| Service Code | 9(3) |
| End sentinel (?) | X(1) |

**Note:** For EMV transactions, the Application PAN Sequence Number (EMV Tag: 5F34) is carried in the EMV Additional Request Data field (FID 6, subFID P). EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this FID description for reference purposes only.

## FID r — Track 2/Supervisor

**Request:** Optional. Variable length of 1 to 40 bytes.

**Response:** Optional. Variable length of 1 to 40 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.TRACK2

The Track 2/Supervisor field consists of a group of fields representing the supervisor Track 2. Supervisor Track 2 data is typically submitted only in certain transaction requests such as returns or adjustments. In these cases, BASE24-pos is also configured to indicate that the supervisor security is to be applied to these transactions. The format of the Track 2/Supervisor data is the same as that of the Track 2/Customer data carried in FID q.

## FID s — Transaction Description

**Request:** Not available.

**Response:** Optional. Variable length of 1 to 24 bytes.

**Internal Field:** ARSP.DESCR.TBL

The Transaction Description field is a 24-character description of the transaction as defined in the ARSP Transaction Description Record. It is used for receipt purposes.

## FID t — PIN Pad Identifier

**Request:** Optional. Fixed length of 16 bytes. If not supplied, the acceptor uses the Terminal ID in the SPDH header.

**Response:** Optional. Fixed length of 16 bytes. The value is echoed from the PIN pad identifier received in the request.

**Internal Field:** POS Merchant token

The PIN pad identifier is the logical identifier of the PIN pad at the acquiring terminal and is unique in the accepting BASE24-pos environment.

## FID u — Acceptor Posting Date

**Request:** Not available.

**Response:** Optional. Fixed length of 6 bytes. If this date is not supplied, the acquirer Merchant Link process uses its own posting date, resulting in settlement totals that may not agree.

**Internal Field:** POS Merchant token

The acceptor posting date (YYMMDD) is the date of the POS Transaction Log File (PTLF) to which the acceptor logged the transaction. The acceptor SPDH module returns this field to the Merchant Link process so that transactions can be matched between the acquirer and acceptor, and reconciliation totals can be accumulated regardless of individual cutover configurations. This field can also be returned by the acceptor on transactions originated at POS devices.

## FID 0 — AMEX Data Collection

**Request:** Optional. Variable-length of 46 to 118 bytes, depending on the standard industry format being used. The data length for each specific standard industry format is fixed.

**Response:** Optional. Variable length of 46 to 118 bytes, depending on the standard industry format being used. The data length for each specific standard industry format is fixed. This field is not returned in a response if the transaction is declined.

**Internal Field:** American Express token

The AMEX Data Collection field is used to capture transactions originating from American Express cardholders. American Express has defined categories under which transactions are placed. These categories are as follows:

- Auto rental
- Lodging
- Restaurant
- General retail
- Oil

For each of these categories, American Express requires different data to be sent from the device. Therefore, American Express has set forth standard industry formats for each category to ensure the data they require is captured by the device and sent to the host. The SPDH module is able to recognize the standard industry formats carried in this field and, subsequently, capture and process transactions sent using them. In conjunction with this field, the customer must set up Standard Industrial Classification (SIC) Codes or Merchant Category Codes in the PTD. SIC and Merchant Category Codes identify the merchant's line of business. For more information on the PTD, refer to the ***BASE24-pos Files Maintenance Manual***. For the American Express standard industry formats, refer to the ***ACI Standard POS Device Message Specifications Manual***.

## FID 1 — PS2000 Data

**Request:** Optional. Fixed length of 24 bytes.

**Response:** Optional. Fixed length of 24 bytes.

**Internal Field:** PS2000 token

The PS2000 Data field consists of a group of fields used with the Visa PS2000 program. This field contains PS2000 data for both request and response messages. The terminal is responsible for correctly filling in any of the fields that are required in a given request, placing default values in the other fields, and parsing any of the fields out of the response message that need to be printed on a receipt or displayed.

The format for this field is as follows:

| | |
|---|---|
| Authorization characteristics indicator | PIC X(1) |
| Transaction identifier | PIC X(15) |
| Validation code | PIC X(4) |
| Market-specific data fields | |
| Market-specific data identifier | PIC X(1) |
| Duration | PIC 9(2) |
| Prestigious property indicator | PIC X(1) |

## FID 2 — Track 1/Customer

**Request:** Optional. Variable length of 1 to 82 bytes. Either this field or the Track 2/Customer field (FID q) is required for the following transactions:

Normal purchase
Preauthorization purchase
Preauthorization purchase completion
Mail or telephone order
Merchandise return
Cash advance
Card verification
Balance inquiry
Purchase with cash back
Purchase adjustment
Merchandise return adjustment
Cash advance adjustment
Cash back adjustment

The customer determines when or if Track 1 data is required on Card Prefix File (CPF) screen 1. The SPDH module checks that all financial transactions contain either FID q (Track 2) or FID 2. For more information about how the SPDH module formats track data, refer to appendix A.

**Response:** Optional. Variable length of 1 to 82 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.TRACK2 or
Track 1 token

The Track 1/Customer field consists of a group of fields representing the customer's Track 1. This data is obtained from the terminal's card swipe reader.

The format for customer Track 1 data, organized in ISO Standard Format, is as follows:

| Field | Description |
|---|---|
| Start sentinel | 1 character (%) |
| Format code | 1 character (B for credit cards) |
| Identification (PAN) | Up to 19 digits (variable length) |
| Field separator | 1 character (^) |
| Country code | 3 digits (only present when the PAN starts with 59) |
| Name | Up to 26 characters (variable length) |
| Field separator | 1 character (^) |
| Expiration date | 4 digits (YYMM) |
| Service code | 3 digits |
| Discretionary data | Up to 21 characters (variable length) |
| End sentinel | 1 character (?) |
| Longitudinal redundancy check character | 1 character |

**Note:** For contactless transactions, the Discretionary Data field can contain the information necessary for dynamic card verification (card verification value, application transaction counter, and unpredictable number).

## FID 3 — Track 1/Supervisor

**Request:** Optional. Variable length of 1 to 82 bytes. Supervisor Track 1 is typically submitted only in certain transaction requests, such as returns or adjustments. In these cases, BASE24-pos is also configured to indicate that the supervisor security is to be applied to these transactions.

**Response:** Optional. Variable length of 1 to 82 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.TRACK2 or
Track 1 token

The Track 1/Supervisor field consists of a group of fields representing the supervisor Track 1. This data is obtained from the terminal's card swipe reader. The supervisor Track 1 is typically submitted only in certain transaction requests, such as returns or adjustments. In these cases, the host is also configured to indicate that the supervisor security is to be applied to these transactions.

The format for supervisor Track 1 data, organized in ISO Standard Format, is as follows:

| Field | Description |
|---|---|
| Start sentinel | 1 character (%) |
| Format code | 1 character (B for credit cards) |
| Identification (PAN) | Up to 19 digits (variable length) |
| Field separator | 1 character (^) |
| Country code | 3 digits (only present when the PAN starts with 59) |
| Name | Up to 26 characters (variable length) |
| Field separator | 1 character (^) |
| Expiration date | 4 digits (YYMM) |
| Service code | 3 digits |
| Discretionary data | Up to 21 characters (variable length) |
| End sentinel | 1 character (?) |
| Longitudinal redundancy check character | 1 character |

## FID 4 — Industry Data

**Request:** Optional. Variable length of 156 to 171 bytes.

**Response:** Optional. Variable length of 156 to 171 bytes. If included, the value is echoed from the request.

**Internal Field:** POS Industry Data token

The Industry Data field contains information associated with lodging and vehicle rental. The format is as follows:

| | |
|---|---|
| Industry Type | PIC X(2) |
| Industry Data | PIC X(169), variable length |

These fields are further described below.

| Position | Length | Description |
|---|---|---|
| **01–02** | **2** | **Industry Type**<br>A code that identifies the type of industry data. Valid values are as follows:<br>LG = Lodging<br>VR = Vehicle rental |

| Position | Length | Description |
|---|---|---|
| **03–171** | **169** | **Industry Data (variable-length, 154–169 bytes)** |
| **03–156** | **154** | **Lodging**<br>The following fields are used for lodging data. |
| **03–08** | **6** | **Arrival Date (YYMMDD)**<br>The date the customer checked in. For a no-show or advanced lodging transaction, this is the scheduled arrival date. |
| **09–14** | **6** | **Departure Date (YYMMDD)**<br>The date the customer checked out. |
| **15–18** | **4** | **Total Room Nights**<br>The total number of room nights during the lodging stay. |
| **19–30** | **12** | **Room Rate**<br>The daily room charges exclusive of taxes and fees. Two decimal places are implied. |
| **31–42** | **12** | **Room Tax**<br>The daily room tax. Two decimal places are implied. |
| **43–54** | **12** | **Phone Charges**<br>The total amount of charges for all phone calls. Two decimal places are implied. |
| **55–66** | **12** | **Laundry Charges**<br>The total amount of laundry and dry cleaning charges. Two decimal places are implied. |
| **67–78** | **12** | **Gift Shop Charges**<br>The total amount of gift shop and specialty shop charges. Two decimal places are implied. |
| **79–90** | **12** | **Bar Charges**<br>The total amount of bar and in-room mini-bar charges. Two decimal places are implied. |

| Position | Length | Description |
|---|---|---|
| **91–102** | **12** | **Other Charges**<br>The total amount of other charges associated with the lodging stay. Two decimal places are implied. |
| **103–114** | **12** | **Total Tax Amount**<br>The total amount of sales tax or value-added tax on the total purchase. Two decimal places are implied. |
| **115–129** | **15** | **Property Phone Number**<br>Identifies the specific lodging property location by its local phone number. |
| **130–144** | **15** | **Customer Service Phone Number**<br>The phone number used to resolve cardholder questions and disputes. |
| **145–154** | **10** | **Folio Number**<br>The merchant's internal invoice or billing ID reference number. |
| **155** | **1** | **Fire Safety Act Indicator**<br>A code that identifies whether the facility is in compliance with the Hotel and Motel Fire Safety Act of 1990 (PL101-391), or similar legislation. Valid values are as follows:<br>Y = Yes, the facility is in compliance.<br>N = No, the facility is not in compliance. |
| **156** | **1** | **No Show Indicator**<br>A code indicating whether the individual showed up after making a reservation for lodging. Valid values are as follows:<br>0 = Not applicable.<br>1 = No show. Transaction amount is due. |
| **03–171** | **169** | **Vehicle Rental**<br>The following fields are used for vehicle rental data. |

| Position | Length | Description |
|---|---|---|
| **03–31** | **29** | **Renter Name**<br>The name of the individual making the vehicle rental agreement. |
| **32–35** | **4** | **Rental Class ID**<br>The classification of the vehicle rented, such as midsize or luxury. |
| **36–41** | **6** | **Rental Date (YYMMDD)**<br>The date the customer picked up the vehicle from the rental agency. |
| **42–59** | **18** | **Rental City**<br>The city where the customer picked up the vehicle. |
| **60–62** | **3** | **Rental State**<br>The state or province where the customer picked up the vehicle. This field must contain a valid U.S. state code if the rental country is USA. |
| **63–65** | **3** | **Rental Country**<br>The country where the customer picked up the vehicle. This field must contain a valid alphabetic ISO country code. |
| **66–71** | **6** | **Return Date (YYMMDD)**<br>The date the customer returned the vehicle. |
| **72–89** | **18** | **Return City**<br>The city where the customer returned the vehicle. |
| **90–92** | **3** | **Return State**<br>The state or province where the customer returned the vehicle. This field must contain a valid U.S. state code if the rental country is USA. |

| Position | Length | Description |
|---|---|---|
| **93–95** | **3** | **Return Country**<br>The country where the customer returned the vehicle. This field must contain a valid alphabetic ISO country code. |
| **96–105** | **10** | **Return Location ID**<br>The code, address, phone number, or other identifier used to identify the location where the customer returned the vehicle. |
| **106–109** | **4** | **Days Rented**<br>The number of days the vehicle was rented. |
| **110–121** | **12** | **Daily Rental Rate**<br>The daily rental rate, exclusive of taxes and fees. Two decimal places are implied. |
| **122–133** | **12** | **Extra Charges**<br>The total amount of extra charges associated with the vehicle rental. Two decimal places are implied. |
| **134–145** | **12** | **Total Tax Amount**<br>The total amount of sales tax or value-added tax on the total purchase. Two decimal places are implied. |
| **146–160** | **15** | **Customer Service Phone Number**<br>The phone number used to resolve cardholder questions and disputes. |
| **161–169** | **9** | **Agreement Number**<br>The invoice number of the original rental agreement. |
| **170** | **1** | **Tax Exempt Indicator**<br>A code indicating whether the goods or services were tax exempt. Valid values are as follows:<br>0 = Not applicable<br>1 = Tax exempt |

| Position | Length | Description |
|---|---|---|
| 171 | 1 | **No Show Indicator**<br>A code indicating whether the individual showed up after reserving a vehicle for rental. Valid values are as follows:<br>0 = Not applicable.<br>1 = No show. Transaction amount is due. |

## FID 6 — Product SubFIDs

**Request:** Optional. Variable length.

**Response:** Optional. Variable length.

This Product SubFIDs field consists of subordinate optional data subfields. For information about these subfields, see the topic "Optional Data Subfields — FID 6" later in this section.

When you include FID 6 in the request and response field maps of the ACI Standard Device Configuration File (ANCF), you are enabling all subfields to be added to the request or response.

## FID 7 — Product SubFIDs

**Request:** Optional. Variable length.

**Response:** Optional. Variable length.

This Product SubFIDs field consists of subordinate optional data subfields. For information about these subfields, see the topic "Optional Data Subfields — FID 7" later in this section.

When you include FID 7 in the request and response field maps of the ACI Standard Device Configuration File (ANCF), you are enabling all subfields to be added to the request or response.

## FID 8 — Product SubFIDs

**Request:** Optional. Variable length.

**Response:** Optional. Variable length.

This Product SubFIDs field consists of subordinate optional data subfields. For information about these subfields, see the topic "Optional Data Subfields — FID 8" later in this section.

When you include FID 8 in the request and response field maps of the ACI Standard Device Configuration File (ANCF), you are enabling all subfields to be added to the request or response.

## FID 9 — Customer SubFIDs

**Request:** Optional. Variable length.

**Response:** Optional. Variable length.

The Customer SubFIDs field is reserved for future use by customers.

# Optional Data Subfields — FID 6

The optional data subfields for FID 6 (Product SubFIDs) are summarized in a table below, followed by individual descriptions of the subfields.

**Note:** The information in FIDs 6 through 9 are contained in subfields. These subfields are included in the request or response message when FIDs 6 through 9 are specified in the host configuration. Specifying FIDs 6 through 9 in the host configuration specifies all subfields for FIDs 6 through 9, respectively. FIDs 6, 7, and 8 are reserved for product use. FID 9 is reserved for customer use.

## Summary Table

The subfields for FID 6 are described below according to their subfield identifiers (SFIDs). The table lists the SFID, the length of the subfield in the message, its associated field name, and the name of the internal field to which it is mapped by the SPDH module. In addition, a check mark (✓) appears in the RQST and RESP columns if a subfield is available for requests and responses, respectively.

| SFID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| A | 12 bytes | Host original data | PSTM.ORIG-DATA.<br>TRN-TIM<br>PSTM.ORIG-DATA.<br>TRN-DAT | ✓ | ✓ |
| B | 4 bytes | Manual card verification data (CVD)—customer | BASE24-pos Release 5.1 token | ✓ | ✓ |
| C | 4 bytes | Manual card verification data (CVD)—administrative | BASE24-pos Release 5.1 token | ✓ | ✓ |
| D | 30–876 bytes | Purchasing card/fleet card data | Purchasing Card token | ✓ | |
| E | 3 bytes | POS entry mode | PSTM.PT-SRV-ENTRY-MDE | ✓ | ✓ |
| F | 1–2 bytes | Electronic commerce data | BASE24-pos Release 5.1 token<br>BASE24-pos Data token | ✓ | |

| SFID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| G | 1 byte | Visa commercial card type indicator | BASE24-pos Release 5.1 token | | ✓ |
| H | 2 bytes | CVD presence indicator and CVD result | BASE24-pos Release 5.0 token | ✓ | ✓ |
| I | 3 bytes | Transaction currency code | Multiple Currency token<br>EMV Request Data token | ✓ | ✓ |
| J | 32 bytes | Cardholder certificate serial number | Cardholder Serial Number token | ✓ | |
| K | 32 bytes | Merchant certificate serial number | Merchant Serial Number token | ✓ | |
| L | 80 bytes | XID/trans stain | Trans Stain XID token | ✓ | |
| N | 4 bytes | Message reason code | RSN-ONL-CDE field of the EMV Status token | ✓ | |
| O | 136 bytes (variable length) | EMV request data | EMV Request Data token | ✓ | |
| P | 64 bytes (variable length) | EMV additional request data | EMV Discretionary Data token<br>EMV Status token | ✓ | |
| Q | 64 bytes (variable length) | EMV response data | EMV Response Data token | | ✓ |
| R | 258 bytes (variable length) | EMV additional response data | EMV Script Data token | | ✓ |
| S | 63 bytes | Stored value data | Stored Value token | ✓ | ✓ |
| T | 23 bytes | Key serial number and descriptor | DUKPT Data token | ✓ | |
| U | 16 bytes | Transaction subtype data | Transaction Subtype token | ✓ | |

| SFID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| V | 1 byte | Authentication collection indicator | BASE24-pos Release 5.1 token | ✓ | ✓ |
| W | 1 byte | CAVV/AAV result code | BASE24-pos Release 5.1 token | | ✓ |
| X | 6 bytes | Point of service data | Point of Service Data token | ✓ | |
| Y | 2–202 bytes | Authentication data | Authentication Data token | ✓ | ✓ |
| Z | 1 byte | Card verification flag 2 | POS Data token | ✓ | ✓ |
| b | 39 bytes | Electronic check conversion data | Check Authorization 2 token | ✓ | |
| c | 64 bytes | MICR data | MICR Data token | ✓ | |
| d | 115 bytes | Electronic check callback information | Check Authorization 2 token<br>Electronic Check Callback token | ✓ | ✓ |
| e | 21 bytes | Interchange compliance data | Interchange Compliance token | | ✓ |
| f | 1 byte | Response source or reason code | POS Data token | | ✓ |
| g | 4 bytes | POS merchant data | POS Merchant token | ✓ | |
| h | 6 bytes | System Trace Audit Number (STAN) | STAN | ✓ | |
| i | 12 bytes | Retrieval Reference Number | Switch Common Data token | ✓ | |
| j | 3–4 bytes | Debit Network/Sharing Group ID | Switch Common Data token | | ✓ |
| k | 2 bytes | Card Level Results | Switch Common Data token | | ✓ |

| SFID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| l | 20–120 bytes | Healthcare/Transit Data | Healthcare/Transit token | ✓ | ✓ |
| m | 19–95 bytes | Healthcare Service Data | Healthcare Service token | ✓ | ✓ |
| n | 1 byte | Error Flag | BASE24-pos Release 5.0 token | | ✓ |
| o | 3-300 bytes | American Express Additional Data | American Express Additional Data token. | ✓ | |

## SFID A — BASE24 Original Data

**Request:** Optional. Fixed length of 12 bytes.

**Response:** Optional. Fixed length of 12 bytes. If included, the value is echoed from the request.

**Internal Field:** PSTM.ORIG-DATA.TRN-TIM
PSTM.ORIG-DATA.TRN-DAT

The BASE24 Original Data field represents the original transaction time (hhmmsshh) and date (MMDD) information carried in the BASE24-pos Standard Internal Message (PSTM). The format of this data is identical to the subordinate data definitions TRN-TIM and TRN-DAT for the ORIG-DATA structure of the PSTM.

## SFID B — Manual CVD—Customer

**Request:** Optional. Fixed length of 4 bytes.

**Response:** Optional. Fixed length of 4 bytes. If included, the value is echoed from the request.

**Internal Field:** BASE24-pos Release 5.1 token

The Manual CVD—Customer field contains the manually entered card verification digits from the customer's card. The format is the same as the Manual CVD field of the BASE24-pos Release 5.1 token.

If this field contains a 3-digit American Express card security code (CSC), it must be left-justified and followed by a blank.

## SFID C — Manual CVD—Administrative

**Request:** Optional. Fixed length of 4 bytes.

**Response:** Optional. Fixed length of 4 bytes. If included, the value is echoed from the request.

**Internal Field:** BASE24-pos Release 5.1 token

The Manual CVD—Administrative field contains the manually entered card verification digits from the customer's card. The format is the same as the Manual CVD field of the BASE24-pos Release 5.1 token.

If this field contains a 3-digit American Express card security code (CSC), it must be left-justified and followed by a blank.

## SFID D — Purchasing Card or Fleet Card Data

**Request:** Optional. Variable length of 30–876 bytes, depending on the data type.

**Response:** Not applicable.

**Internal Field:** Purchasing Card token

The Purchasing Card/Fleet Card Data field contains purchasing or fleet card information used to create the Purchasing Card token. The format is the same as the Purchasing Card token.

## SFID E — POS Entry Mode

**Request:** Optional. Fixed length of 3 bytes.

**Response:** Optional. Fixed length of 3 bytes.

**Internal Field:** PSTM.PT-SRV-ENTRY-MDE

A code indicating the manner in which transaction data entered BASE24-pos. Valid values for the first and second digits are as follows:

00 = Unspecified
01 = Manually
02 = Magnetic stripe
03 = Bar code
04 = OCR
05 = Integrated circuit card
06 = Reserved for ISO use.
07 = Contactless chip card transaction
08–60 = Reserved for ISO use
61–80 = Reserved for national use
81–90 = Reserved for private use
91 = Contactless magnetic stripe transaction
92–99 = Reserved for private use

Valid values for the third digit are as follows:

0 = Unspecified
1 = PIN entry capability
2 = No PIN entry capability
3–5 = Reserved for ISO use
6–7 = Reserved for national use
8–9 = Reserved for private use

For EMV transactions, the first two digits of this field must be set to a value of 05 and the third digit must be set to a value of 0, 1, or 2.

## SFID F — Electronic Commerce Flag

**Request:** Optional. Variable length.

**Response:** Not applicable.

**Internal Field:** BASE24-pos Release 5.1 token
BASE24-pos Data token

The electronic commerce flag subfield contains codes to identify electronic commerce, mail or telephone order transactions, and recurring payments. The format is as follows:

| | |
|---|---|
| Electronic Commerce Flag | PIC X |
| Recurring Payment Indicator | PIC X |

Valid values for the Electronic Commerce Flag are as follows:

0 = Not an electronic commerce transaction
1 = Single mail or telephone order transaction
2 = Recurring mail or telephone order transaction
3 = Mail or telephone order installment payment
4 = Mail or telephone order unknown classification
5 = Secure electronic transaction with cardholder certificate
6 = Encrypted electronic commerce transaction where the merchant is capable of authenticating the cardholder, but was unable to complete the authentication, e.g., because the issuer or cardholder does not participate in the appropriate authentication program.
7 = Encrypted electronic commerce transaction
8 = Nonsecure electronic commerce transaction
9 = Non-authenticated security transaction; does not comply with secure electronic transaction and the merchant supports secure electronic transactions.
S = Internet electronic delivery
T = Internet physical delivery

Valid values for the Recurring Payment Indicator, if it is present, are as follows:

0 = Not a recurring payment
1 = Recurring payment

## SFID G — Commercial Card Type

**Request:** Not applicable.

**Response:** Optional. Fixed length of 1 byte.

**Internal Field:** BASE24-pos Release 5.1 token

The commercial card type. Valid values are as follows:

0 = Noncommercial card or unknown or unspecified card
B = Business card
R = Corporate card
S = Purchasing card

## SFID H — Card Verification Digits Presence Indicator and Result

**Request:** Optional. Fixed length of 2 bytes.

**Response:** Optional. Fixed length of 2 bytes. If included, the first byte is echoed from the request.

**Internal Field:** BASE24-pos Release 5.0 token

A field indicating whether the CVD/CVV2/CVD2/CSC is present, and if so, the result of the CVD check. The first byte is the CVD presence indicator. Valid values for this byte are as follows:

0 = CVV2 value is deliberately bypassed or is not provided by the merchant
1 = CVV2 value is present
2 = CVV2 value is on the card, but is illegible
9 = Cardholder states that the card has no CVV2 imprint

The second byte is the CVD result. Valid values for this byte are as follows:

0 = Card verification was not performed because the transaction was denied before card verification processing started.
C = Card verification was performed and the card verification digits (CVD) were invalid. The situation was noted, and transaction processing continued.
D = Card verification was performed and the CVD was invalid. The transaction was denied and the ERR-FLG field was set to C.
J = CVV checking was not performed. The track length was in error. The host database indicates that the transaction should be denied.
K = Card verification was not performed. The track length was in error. The situation was noted and the transaction was referred.
L = CVV checking was not performed. The track length was in error. The host database indicates that processing should continue.

N or ␣ = Authorizing entity has not attempted card verification or could not verify the CVD due to a security device error. (␣ indicates a blank character.)

O = Card verification was not performed. A CVD value was not on the card. Not all cards have a CVD value encoded. The card expiration date must be equal to or greater than an expiration date defined on the Card Prefix File (CPF) to ensure that the CVD field has been encoded. If the card expiration date is equal to or greater than the CPF date, the CVD checks are performed.

P = Card verification was not performed. Either the merchant ignored the CVD on purpose or the user falsely indicated no CVD was on the card.

R = Card verification was performed and the CVD was invalid. The situation was noted and the transaction should be referred.

U = Issuer has not certified or has not provided the encryption keys to the switch.

Y = Card verification was performed and the CVD was valid.

## SFID I — Transaction Currency Code

**Request:** Optional. Fixed length of 3 bytes.

**Response:** Optional. Fixed length of 3 bytes.

**Internal Field:** Multiple Currency token
EMV Request Data token

A numeric code indicating the currency of the transaction, as received from the POS device, according to the ISO 4217 standard, ***Codes for the Representation of Currencies and Funds***.

For an EMV request, this field is placed in the TRAN-CRNCY-CDE field of the EMV Request Data token.

## SFID J — Cardholder Certificate Serial Number

**Request:** Optional. Fixed length of 32 bytes.

**Response:** Not applicable.

**Internal Field:** Cardholder Serial Number token

The cardholder certificate serial number for secure electronic commerce.

## SFID K — Merchant Certificate Serial Number

**Request:** Optional. Fixed length of 32 bytes.

**Response:** Not applicable.

**Internal Field:** Merchant Serial Number token

The merchant certificate serial number for secure electronic commerce.

## SFID L — XID/TRANS STAIN

**Request:** Optional. Fixed length of 80 bytes. The first 40 bytes contain the transaction identifier. The second 40 bytes contain the transaction stain.

**Response:** Not applicable.

**Internal Field:** Trans Stain XID token

The transaction identifier and transactions stain. The transaction identifier (XID) is 40 bytes long. The transaction stain is a hash value calculated by applying a secure hash algorithm to the XID and CardSecret (a secret SET-defined value known only to the cardholder and the issuer of the cardholder certificate). It is 40 bytes long.

## SFID N — Message Reason Code

**Request:** Optional. Fixed length of 4 bytes.

**Response:** Not applicable.

**Internal Field:** RSN-ONL-CDE field of the EMV Status token

The message reason code specifies why a transaction is to be authorized online (rather than being completed locally) or why a transaction has been completed locally (rather than being authorized online). Values are defined in the ISO 8583 (1993) standard.

In an online request message (message subtype O), the message reason code contains one of the values in the table below. When more than one message reason code condition applies to a transaction, the applicable message reason code with the highest priority is used.

| Value | Priority | Description | Example Conditions |
|---|---|---|---|
| 1500 | 9 | ICC application, common data file (CDF) unable to process | Only used by integrated ICC/MSR terminals where the terminal has possession of the card and this condition can be accurately identified. |
| 1501 | 10 | ICC application, application data file (ADF) unable to process | Only used by integrated ICC/MSR terminals where the terminal has possession of the card and this condition can be accurately identified. |
| 1502 | 11 | ICC random selection | Not used. |
| 1503 | 6 | Terminal random selection | The "Transaction selected randomly for online processing" bit is set in byte 4, bit 5 of the Terminal Verification Results (TVR) in the EMV Request Data (SFID O). |
| 1504 | 1 | Terminal not able to process ICC | Fallback for ICC. |
| 1505 | 2 | Online forced by ICC | ICC forced online, no bits set in TVR. |
| 1506 | 3 | Online forced by card acceptor | Card used twice; used the maximum times per day; one in *n* authorization; PAN key entry authorization; exclusion band checks; prevalid card. |
| 1507 | 12 | Online forced by card accepting device (CAD) to be updated | Not used. |
| 1508 | 8 | Online forced by terminal | The TVR indicates that online processing is required. |
| 1509 | 4 | Online forced by card issuer | Expired card; new card; service code; hot card. |
| 1510 | 7 | Over floor limit | Transaction amount above floor limit. |
| 1511 | 5 | Merchant suspicious | Force authorization ("no" at signature check); card returns an inappropriate cryptogram. |

In a force-post message (message subtype F) or store-and-forward message (message subtype S) that the terminal has previously attempted to send to BASE24-pos as an online request message (message subtype O), the message reason code contains the same value as in the online request message.

In a force-post message (message subtype F) or store-and-forward message (message subtype S) that the terminal has not previously attempted to send to BASE24-pos as an online request message (message subtype O), the message reason code contains one of the values in the table below. When more than one message reason code condition applies to a transaction, the applicable message reason code with the highest priority is used.

| Value | Priority | Description | Example Conditions |
|---|---|---|---|
| 1004 | 1 | Terminal processed | Forced data capture (e.g., local transaction store is full). |
| 1005 | 2 | ICC processed | ICC authorized. |
| 1006 | 3 | Under floor limit | Amount is less than the terminal's floor limit in a non-ICC transaction. |
| 1007 | 4 | Stand-in processing at the acquirer's option | A preauth completion transaction accepted by the terminal. |

Acquirers can use the message reason code to determine the appropriate EMV Authorization Response Code (ARC) for transactions authorized at a terminal. If an approved advice message is received with a message reason code of 10xx, then the ARC is Y1 (offline approved). If an approved advice message is received with a message reason code of 15xx, then the ARC is Y3 (unable to go online, offline approved).

## SFID O — EMV Request Data

**Request:** Optional. Variable length of 121–136 bytes, depending on the Issuer Application Data field definition used.

**Response:** Not applicable.

**Internal Field:** EMV Request Data token

The EMV Request Data field contains the thirteen minimum request data elements, as defined by EMV. This subFID is required for all EMV transaction requests. The format of this subFID depends on the value in the Smart Card Scheme field.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

For terminals supporting the EMV version 1 scheme, the format is as follows:

| | |
|---|---|
| Smart Card Scheme | PIC 9(2) |
| Cryptographic Information Data | PIC 9(2) |
| Terminal Country Code | PIC 9(3) |
| EMV Date | PIC 9(6) |
| Application Cryptogram (AC) | PIC X(16) |
| Application Interchange Profile (AIP) | PIC X(4) |
| Application Transaction Counter (ATC) | PIC X(4) |
| Unpredictable Number | PIC X(8) |
| Terminal Verification Result (TVR) | PIC X(10) |
| Cryptogram Transaction Type | PIC 9(2) |
| Issuer Application Data | PIC X(64) variable data |

For terminals supporting the EMV version 2 scheme, the format is as follows:

| | |
|---|---|
| Smart Card Scheme | PIC 9(2) |
| Cryptographic Information Data | PIC 9(2) |
| Terminal Country Code | PIC 9(3) |
| EMV Date | PIC 9(6) |
| Application Cryptogram (AC) | PIC X(16) |
| Application Interchange Profile (AIP) | PIC X(4) |
| Application Transaction Counter (ATC) | PIC X(4) |
| Unpredictable Number | PIC X(8) |
| Terminal Verification Result (TVR) | PIC X(10) |
| Cryptogram Transaction Type | PIC 9(2) |
| Cryptogram Currency Code | PIC X(3) |
| Cryptogram Amount | PIC X(12) |
| Issuer Application Data | PIC X(64) variable data |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–02** | **2** | **Smart Card Scheme**<br><br>A code indicating the smart card scheme used for this transaction. Valid values are as follows:<br><br>00 = EMV version 1<br>01 = EMV version 2 |
| **03–04** | **2** | **Cryptographic Information Data**<br><br>Hexadecimal characters (0–F) representing eight bits of cryptographic information data. The first hexadecimal character represents bits 0–3, and the second hexadecimal character represents bits 4–7. The SPDH converts the hexadecimal characters received in the request message to binary data, as described earlier for the "Binary Data Conversion" topic, for storage in BASE24 tokens.<br><br>**EMV Tag:** 9F27. |
| **05–07** | **3** | **Terminal Country Code**<br><br>A three-digit code indicating the country where this terminal is located, according to the ISO 3166 standard, ***Codes for the Representation of Names of Countries***.<br><br>**EMV Tag:** 9F1A. |
| **08–13** | **6** | **EMV Date**<br><br>The local date (in YYMMDD format) that the transaction was authorized.<br><br>**EMV Tag:** 9A. |
| **14–29** | **16** | **Application Cryptogram (AC)**<br><br>The cryptogram returned by the ICC in response to the GENERATE AC command.<br><br>The field contains the offline EMV transaction's Transaction Certificate (TC) that was generated by the EMV card.<br><br>**EMV Tag:** 9F26. |

| Position | Length | Description |
|---|---|---|
| **30–33** | **4** | **Application Interchange Profile (AIP)**<br><br>Hexadecimal characters (0–F) representing 16 bits of Application Interchange Profile data. The first hexadecimal character represents bits 0–3 of byte 1, the second hexadecimal character represents bits 4–7 of byte 1, the third hexadecimal character represents bits 0–3 of byte 2, and so on. The SPDH converts the hexadecimal characters received in the request message to binary data, as described earlier for the "Binary Data Conversion" topic, for storage in BASE24 tokens.<br><br>**EMV Tag:** 82. |
| **34–37** | **4** | **Application Transaction Counter (ATC)**<br><br>A count of the number of transactions performed with this EMV card. The application on the chip maintains and increments the application transaction counter.<br><br>**EMV Tag:** 9F36. |
| **38–45** | **8** | **Unpredictable Number**<br><br>An unpredictable number, in hexadecimal format, used to provide variability and uniqueness to the generation of a cryptogram.<br><br>**EMV Tag:** 9F37. |
| **46–55** | **10** | **Terminal Verification Results (TVR)**<br><br>Multiple bit values stored as hexadecimal characters (0–F). The first hexadecimal character represents bits 0–3 of byte 1, the second hexadecimal character represents bits 4–7 of byte 1, the third hexadecimal character represents bits 0–3 of byte 2, and so on. The SPDH converts the hexadecimal characters received in the request message to binary data, as described earlier for the "Binary Data Conversion" topic, for storage in BASE24 tokens.<br><br>**EMV Tag:** 95. |

| Position | Length | Description |
|---|---|---|
| **56–57** | **2** | **Cryptogram Transaction Type**<br><br>A code indicating the type of financial transaction, represented by the first two digits of the processing code from the 1987 ISO 8583 standard, ***Bank Card Originated Messages—Interchange Message Specifications—Content for Financial Transactions***.<br><br>**EMV Tag:** 9C. |
| **58–60** | **3** | **Cryptogram Currency Code**<br><br>A code indicating the currency used for the transaction amount in the Cryptogram Amount field.<br><br>**Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme.<br><br>**EMV Tag:** 5F2A. |
| **61–72** | **12** | **Cryptogram Amount**<br><br>The transaction amount used to generate the value in the Application Cryptogram (AC) field.<br><br>**Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme.<br><br>**EMV Tag:** 9F02. |
| **58–121 or 73–136** | **64** | **Issuer Application Data**<br><br>Contains proprietary issuer application data for transmission to the issuer in an online transaction.<br><br>BASE24-pos currently supports multiple definitions for issuer application data. For more information on these fields, refer to DDL documentation or the individual card scheme documentation.<br><br>**Note:** The position of this field depends on which smart card scheme is used.<br><br>**EMV Tag:** 9F10. |

| Position | Length | Description |
|---|---|---|
| **58–121** | **64** | **Visa/UKIS Definition**<br>The following fields contain the Visa/UKIS definition of the Issuer Application Data field. |
| **58–59** | **2** | **Length** |
| **60–61** | **2** | **Derivation Key Index** |
| **62–63** | **2** | **Cryptogram Version Number** |
| **64–71** | **8** | **Card Verification Results** |
| **72–121** | **50** | **Issuer Discretionary Data** |
| **58–121** | **64** | **MasterCard/Europay M/CHIP 2.1 Definition**<br>The following fields contain MasterCard/Europay (MCPA) M/CHIP 2.1 definition of the Issuer Application Data field. |
| **58–59** | **2** | **Derivation Key Index** |
| **60–61** | **2** | **Cryptogram Version Number** |
| **62–69** | **8** | **Card Verification Results** |
| **70–73** | **4** | **Dynamic Authentication Code** |
| **74–121** | **48** | **Issuer Discretionary Data** |
| **73–136** | **64** | **MasterCard/Europay M/CHIP 4 Definition**<br>The following fields contain the MasterCard/Europay M/CHIP 4 definition of the Issuer Application Data field. |
| **73–74** | **2** | **Derivation Key Index** |
| **75–76** | **2** | **Cryptogram Version Number** |
| **77–84** | **12** | **Card Verification Results** |
| **85–88** | **4** | **Dynamic Authentication Code** |
| **89–104** | **16** | **Counters** |
| **105–136** | **28** | **Issuer Discretionary Data** |

| Position | Length | Description |
|---|---|---|
| **73–136** | **64** | **EMV CCD Definition**<br>The following fields contain the EMV CCD definition of the Issuer Application Data field. |
| **73–74** | **2** | **Length** |
| **75–76** | **2** | **Common Core ID** |
| **77–78** | **2** | **Derivation Key Index** |
| **79–88** | **10** | **Card Verification Results** |
| **89–104** | **16** | **Counters** |
| **105–106** | **2** | **Issuer Discretionary Data Length** |
| **107–136** | **30** | **Issuer Discretionary Data** |

**Note:** EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this SFID description for reference purposes only.

## SFID P — EMV Additional Request Data

**Request:** Optional. Variable length of 64 bytes. Currently only 48 bytes are used.

**Response:** Not applicable.

**Internal Field:** EMV Discretionary Data token and
EMV Status token

The EMV Additional Request Data field contains additional EMV transaction data. This subFID is optional for all EMV transaction requests. The format of this subFID depends on the value in the Smart Card Scheme field.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

For terminals supporting the EMV version 1 scheme, the format is as follows:

| | |
|---|---|
| Smart Card Scheme | PIC 9(2) |
| Application PAN Sequence Number | PIC 9(2) |
| EMV Terminal Type | PIC 9(2) |

For terminals supporting the EMV version 2 scheme, the format is as follows:

| | |
|---|---|
| Smart Card Scheme | PIC 9(2) |
| Application PAN Sequence Number | PIC 9(2) |
| EMV Terminal Type | PIC 9(2) |
| Cardholder Verification (CVM) Results | PIC X(6) |
| Application Version Number | PIC X(4) |
| Dedicated File Name | PIC X(32) variable data |

If an additional EMV transaction data element is not present, the corresponding field in this subFID is space-filled.

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–02** | **2** | **Smart Card Scheme**<br>A code indicating the smart card scheme used for this transaction. Valid values are as follows:<br>00 = EMV version 1<br>01 = EMV version 2 |
| **03–04** | **2** | **Application PAN Sequence Number**<br>The application PAN sequence number. This field identifies and differentiates cards with the same PAN.<br>**EMV Tag:** 5F34. |
| **05–06** | **2** | **EMV Terminal Type**<br>The EMV terminal type, indicating the environment of the terminal, its communications capability, and its operational control.<br>**EMV Tag:** 9F35. |

| Position | Length | Description |
|---|---|---|
| **07–12** | **6** | **Cardholder Verification (CVM) Results**<br><br>Hexadecimal characters (0–F) representing multiple values. The first hexadecimal character represents bits 0–3 of byte 1, the second hexadecimal character represents bits 4–7 of byte 1, the third hexadecimal character represents bits 0–3 of byte 2, and so on. The SPDH converts the hexadecimal characters received in the request message to binary data, as described earlier for the "Binary Data Conversion" topic, for storage in BASE24 tokens.<br><br>**Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme.<br><br>**EMV Tag:** 9F34. |
| **13–16** | **4** | **Application Version Number**<br><br>The application version number assigned by the payment system for the application.<br><br>**Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme.<br><br>**EMV Tag:** 9F09. |
| **17–48** | **32** | **Dedicated File Name**<br><br>The name of the dedicated file (as described in ISO/IEC 7816-4) or application identifier (as described in ISO/IEC 7816-5).<br><br>**Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme.<br><br>**EMV Tag:** 84. |

**Note:** EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this SFID description for reference purposes only.

# SFID Q — EMV Response Data

**Request:** Not applicable.

**Response:** Optional. Variable length of 64 bytes. Currently only 36 bytes are used.

**Internal Field:** EMV Response Data token

The EMV Response Data field contains the response cryptogram and the response code used to generate the response cryptogram. This subFID is required for EMV transaction response messages for which an ARPC has been generated. The format of this subFID depends on the value in the Smart Card Scheme field.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

For terminals supporting the EMV version 1 scheme, the format is as follows:

| | |
|---|---|
| Smart Card Scheme | PIC 9(2) |
| Issuer Authentication Data | PIC X(32) variable length |

For terminals supporting the EMV version 2 scheme, the format is as follows:

| | |
|---|---|
| EMV Card Scheme | PIC 9(2) |
| Authorization Response Code | PIC X(2) |
| Issuer Authentication Data | PIC X(32) variable length |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–02** | **2** | **Smart Card Scheme**<br>A code indicating the smart card scheme used for this transaction. Valid values are as follows:<br>00 = EMV version 1<br>01 = EMV version 2 |

| Position | Length | Description |
|---|---|---|
| **03–04** | **2** | **Authorization Response Code**<br><br>The EMV authorization response code used with the EMV version 2 card scheme. Valid values are based on BASE24-pos response codes, as follows:<br><br>00 = Approve (BASE24-pos response codes 000–049)<br>01 = Refer (BASE24-pos response codes 100–149)<br>04 = Capture (BASE24-pos response codes 900–949)<br>05 = Decline (all other BASE24-pos response codes)<br><br>**Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme.<br><br>**EMV Tag:** 8A. |
| **03–34 or 05–36** | **32** | **Issuer Authentication Data**<br><br>Contains proprietary issuer authentication data for transmission to the card in an online transaction.<br><br>For more information on the format of this field, refer to the individual card scheme documentation.<br><br>**EMV Tag:** 91. |

**Note:** EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this SFID description for reference purposes only.

## SFID R — EMV Additional Response Data

**Request:** Not applicable.

**Response:** Optional. Variable length of 258 bytes.

**Internal Field:** EMV Script Data token

The EMV Additional Response Data field contains EMV script data returned in an EMV transaction response from the card issuer. This subfield is required in all EMV transaction response messages for script processing.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

The format is as follows:

| | |
|---|---|
| Smart Card Scheme | PIC 9(2) |
| Issuer Script Data | PIC X(256) variable length |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–02** | **2** | **Smart Card Scheme**<br><br>A code indicating the smart card scheme used for this transaction. Valid values are as follows:<br><br>00 = EMV version 1<br>01 = EMV version 2 |
| **03–258** | **256** | **Issuer Script Data**<br><br>The issuer script template, in hexadecimal format, to be sent to the card for processing by the card application.<br><br>**EMV Tag:** 71 or 72. |

**Note:** EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this SFID description for reference purposes on**ly.**

## SFID S — Stored Value Data

**Request:** Optional. Fixed length of 63 bytes.

**Response:** Optional. Fixed length of 63 bytes.

**Internal Field:** Stored Value token

The Stored Value Data field contains stored value card information. This subFID is required for additional card activation transaction requests and all stored value transaction responses. The format is as follows:

| | |
|---|---|
| Balance as Cash Flag | PIC X |
| Card Balance | PIC 9(16)v99 |
| Expiration Date | PIC X(4) |
| Additional Track2 | PIC X(40) |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01** | **1** | **Balance as Cash Flag**<br>A flag indicating whether the remaining balance on the stored value account can be given as cash. This flag is provided in response messages only. Valid values are as follows:<br>0 = Do not give the remaining account balance as cash.<br>1 = Give the remaining account balance as cash. |
| **02–19** | **18** | **Card Balance**<br>The remaining balance on the stored value account. This amount is returned in stored value response messages only. |
| **20–23** | **4** | **Expiration Date**<br>The expiration date of this stored value account in YYMM format. This amount is returned in stored value response messages only. |

| Position | Length | Description |
|---|---|---|
| **24–63** | **40** | **Additional Card Track 2 Data**<br>The Track 2 data from the additional card. This data is present only in additional card activation transaction request messages from the terminal. The format is as follows:<br>Start sentinel (;) X(1)<br>PAN, left-justified X(19)<br>Separator character (=) X(1)<br>Member number (if one exists) 9(3)<br>Country code (if one exists) 9(3)<br>Expiration date YYMM (if one exists) X(4)<br>PIN offset (if one exists) X(4)<br>Algorithm offset (if one exists) X(4) (Blank fill to the right)<br>End sentinel (?) X(1) |

## SFID T — Key Serial Number and Descriptor

**Request:** Optional. Variable length of 23 bytes.

**Response:** Not applicable.

**Internal Field:** DUKPT Data token

The key serial number (KSN) and key descriptor are used with derived unique key per transaction (DUKPT) processing. The format is as follows:

| | |
|---|---|
| Key Serial Number | PIC X(20) |
| Key Serial Number Descriptor | PIC X(3) |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–20** | **20** | **Key Serial Number**<br>The key serial number (KSN) associated with the transaction. The layout of this field is as follows:<br>*FFFFkkkkkkkkkkssssss*<br>where,<br>F = "F" pad character (4 characters)<br>k = Static Key Serial Number data (11 characters) (static for all transactions)<br>s = Transaction counter (5 characters) (incremented for each transaction)<br>An example of Key Serial Numbers for two consecutive transactions might be as follows:<br>FFFF9876543210E00001<br>FFFF9876543210E00002<br>In this example, if the same PAN and PIN were used for both of transactions, the DUKPT-encrypted PIN block would look completely different due to the transaction counter change. |
| **21–23** | **3** | **Key Serial Number Descriptor**<br>The key serial number descriptor from the terminal. This field is required only if a Thales e-Security (Racal) security module is being used by the host. |

## SFID U — Transaction Subtype Data

**Request:** Optional. Fixed length of 16 bytes.

**Response:** Not applicable.

**Internal Field:** Transaction Subtype token

Transaction subtype data enables an institution to distinguish between types of transactions that possess identical transaction codes. For example, transaction subtypes enable an institution to configure multiple rate structures for a single transaction code. The format is as follows:

| | |
|---|---|
| Transaction Subtype | PIC X(4) |
| Acquirer Processing Code | PIC X(6) |
| Issuer Processing Code | PIC X(6) |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–04** | **4** | **Transaction Subtype**<br>A subtype identifier to further describe this transaction. All alphanumeric characters are valid in this field. Values P000 through RZZZ are provided for user-defined transaction subtypes. Subtypes in the ranges 0000 through OZZZ and S000 through WZZZ are reserved for use by BASE24 products. Subtypes in the ranges X000 through ZZZZ are reserved for use by distributors. |
| **05–10** | **6** | **Acquirer Processing Code**<br>The acquirer's external processing code. The first two characters of the processing code indicate the type of transaction, the next two characters specify the *from* account for the transaction, and the last two characters specify the *to* account for the transaction. |
| **11–16** | **6** | **Issuer Processing Code**<br>The issuer's external processing code. The first two characters of the processing code indicate the type of transaction, the next two characters specify the *from* account for the transaction, and the last two characters specify the *to* account for the transaction. |

## SFID V — Authentication Collection Indicator

**Request:** Optional. Fixed length of 1 byte.

**Response:** Optional. Fixed length of 1 byte.

**Internal Field:** BASE24-pos Release 5.1 token

The authentication collection indicator identifies whether customer authentication data is supported and whether the data is available. Valid values are as follows:

0 = Universal Cardholder Authentication Field (UCAF) data collection is not supported at the merchant's Web site.
1 = UCAF data collection is supported by the merchant, but UCAF data was not populated.
2 = UCAF data collection is supported by the merchant, and UCAF data was populated.

## SFID W — CAVV/AAV Result Code

**Request:** Not applicable.

**Response:** Optional. Fixed length of 1 byte.

**Internal Field:** BASE24-pos Release 5.1 token

The CAVV/AAV result code indicates the result of the CAVV (Visa method) or AAV (MasterCard method) validation. Valid values are as follows:

0 = Not validated due to erroneous data
1 = Failed validation
2 = Passed validation
3 = CAVV/AAV validation could not be performed (no e-commerce authentication file)
4 = CAVV/AAV validation could not be performed due to system error, or failure prevented authentication (error accessing the e-commerce authentication file)
5 = The acquirer is participating in authentication, but the issuer is not participating
6 = The issuer BIN is participating in authentication, but not verification
7 = Duplicate CAVV/AAV

# SFID X — Point of Service Data

**Request:** Optional. Fixed length of 6 bytes.

**Response:** Not applicable.

**Internal Field:** Point of Service Data token

Point of service data includes a group of flags that indicate card, cardholder, and transaction status associated with a point-of-sale transaction. The format is as follows:

| | |
|---|---|
| Cardholder Present Indicator | PIC X |
| Card Present Indicator | PIC X |
| Transaction Status Indicator | PIC X |
| Transaction Security Indicator | PIC X |
| Cardholder Activated Terminal Indicator | PIC X |
| Cardholder ID Method | PIC X |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01** | **1** | **Cardholder Present Indicator**<br><br>A code indicating whether the cardholder is present at the POS terminal. Valid values are as follows:<br><br>0 = The cardholder is present<br>1 = The cardholder is not present—unspecified reason<br>2 = The cardholder is not present—mail or fax order<br>3 = The cardholder is not present—telephone or ARU order<br>4 = The cardholder not present—standing order or recurring transaction<br>5 = The cardholder is not present—electronic order (home personal computer or Internet) |
| **02** | **1** | **Card Present Indicator**<br><br>A code indicating whether the card is present at the POS terminal. Valid values are as follows:<br><br>0 = The card is present<br>1 = The card is not present |

| Position | Length | Description |
|---|---|---|
| **03** | **1** | **Transaction Status Indicator**<br><br>A code indicating the purpose or status of the request. Valid values are as follows:<br><br>0 = Normal request<br>1 = Merchant authorization<br>4 = Preauthorized request<br>5 = Stand-in<br>6 = Address verification request<br>7 = Cash back<br>8 = Downtime submission request |
| **04** | **1** | **Transaction Security Indicator**<br><br>A code indicating the card acceptor's security level. Valid values are as follows:<br><br>0 = No security concern<br>1 = Suspected fraud (merchant suspicious—code 10)<br>2 = Identification verified |
| **05** | **1** | **Cardholder Activated Terminal Indicator**<br><br>A code indicating whether the cardholder activated the terminal with a card, and if so, the level of security. Valid values are as follows:<br><br>0 = The cardholder did not activate the terminal with a card<br>1 = Automated dispensing machine with PIN—level 1 security<br>2 = Self-service terminal—level 2 security<br>3 = Limited amount terminal—level 3 security<br>4 = In-flight commerce—level 4 security<br>5 = Script device<br>6 = Electronic commerce<br>7 = Radio frequency device |

| Position | Length | Description |
|---|---|---|
| 06 | 1 | **Cardholder ID Method**<br>A code indicating how the cardholder was verified at the point-of-service. Valid values are as follows:<br>0 = Unknown (default)<br>1 = Signature<br>2 = PIN<br>3 = None (Cardholder Present)<br>4 = None (Cardholder Not Present)<br>5 = Authentication Value<br>6 = Electronic Signature Analysis<br>7 = Biometrics<br>8 = Biographics<br>9 = Other |

## SFID Y — Authentication Data

**Request:** Optional. Variable length of 2–202 bytes.

**Response:** Optional. Variable length of 2–202 bytes.

**Internal Field:** Authentication Data token

Authentication Data standardizes the transport cardholder authentication data for e-commerce transactions. The format is as follows:

| | |
|---|---|
| Authentication Indicator Flag | PIC X(2) |
| Authentication Indicator Data | PIC X(200), variable length |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| 01–02 | 02 | **Authentication Indicator Flag**<br>The authentication indicator flag. The only valid value for this field is 01 (Universal Cardholder Authentication Field). |

| Position | Length | Description |
|---|---|---|
| **03–202** | **0–200** | **Authentication Indicator Data**<br>The generic data. For a MasterCard transaction using the Secure Payment Application Universal Cardholder Authentication field (SPA UCAF) method, this field contains the Accountholder Authentication Value (AAV). |

## SFID Z — Card Verification Flag 2

**Request:** Optional. Fixed length of 1 byte.

**Response:** Optional. Fixed length of 1 byte.

**Internal Field:** POS Data token

The card verification flag 2 indicates whether the card involved in a card-read transaction has already been verified using the CVV2/CVD2/CSC. Valid values are as follows:

0 = Card verification was not performed because the transaction was denied before card verification processing started.

C = Card verification was performed and the card verification digits (CVD) were invalid. The situation was noted and the transaction processing continued.

D = Card verification was performed and the CVD was invalid. The transaction was denied.

N or ␣ = Card verification was not attempted or a security device error occurred (where ␣ indicates a blank space).

O = Card verification was not performed. A CVD value was not on the card. Not all cards have a CVD value encoded. The card expiration date must be equal to or greater than an expiration date defined on the Card Prefix File (CPF) to ensure that the CVD field has been encoded. If the card expiration date is equal to or greater than the CPF date, the CVD checks are performed.

P = Card verification was not performed. Either the merchant ignored the CVD on purpose or the user falsely indicated no CVD was on the card.

R = Card verification was performed and the CVD was invalid. The situation was noted and the transaction should be referred.

S = CVV2 should be on the card but the merchant indicates that it is not.

U = Issuer has not certified or has not provided the encryption keys to the interchange.

Y = Card verification was performed and the CVD was valid.

## SFID b — Electronic Check Conversion Data

**Request:** Optional. Fixed length of 39 bytes.

**Response:** Not applicable

**Internal Field:** Check Authorization 2 token

Electronic check conversion information used with check guarantee or check verification transactions for electronic check authorization. The first 38 bytes of this field contains magnetic ink character recognition (MICR) data containing the institution routing number, account number, and check number of the check to be converted. The next byte contains a flag indicating the type of conversion to perform. The format is as follows:

| | |
|---|---|
| Institution Routing Number | PIC X(11) |
| Account Number | PIC X(19) |
| Check Number | PIC X(8) |
| Conversion Flag | PIC X |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–11** | **11** | **Institution Routing Number**<br>The institution routing number from the MICR data. |

| Position | Length | Description |
|---|---|---|
| **12–30** | **19** | **Account Number**<br>The account number from the MICR data. |
| **31–38** | **8** | **Check Number**<br>The check number from the MICR data. |
| **39** | **1** | **Conversion Flag**<br>A flag indicating the type of electronic conversion to perform. Valid values are as follows:<br>0 = No conversion.<br>1 = Perform a check verification or check guarantee transaction with conversion.<br>2 = Conversion only transaction. |

## SFID c — MICR Data

**Request:** Optional. Fixed length of 64 bytes.

**Response:** Not applicable

**Internal Field:** MICR Data token

The magnetic ink character recognition (MICR) data for the check in Raw Toad format (i.e., as read by the MICR reader). If this subFID is present in the message, the MICR data in the Electronic Check Conversion Data field (subFID b) is parsed from this data. If this subFID is not present in the message, the MICR data in the Electronic Check Conversion Data field (subFID b) is manually entered.

This field is not used by the host for processing. Only SFID b is used by the host and must be present. SFID c is present only if an interchange requires the full MICR in the Raw Toad format.

## SFID d — Electronic Check Callback Information

**Request:** Optional. Fixed length of 115 bytes.

**Response:** Optional. Fixed length of 115 bytes.

**Internal Field:** Check Authorization 2 token
Electronic Check Callback token

The callback information provided on a customer receipt. A request message contains this subFID only when the customer phone number or process control number are received from the device. A response message contains this subFID only when the external message received from the authorizer contains the non-bank authorizer information in the Check Callback Information token. The format is as follows:

| | |
|---|---|
| Customer Phone Number | PIC X(20) |
| Process Control Number | PIC X(6) |
| Non-bank Authorizer's Name | PIC X(25) |
| Non-bank Authorizer's Street | PIC X(20) |
| Non-bank Authorizer's City | PIC X(13) |
| Non-bank Authorizer's State | PIC X(2) |
| Non-bank Authorizer's Postal Code | PIC X(9) |
| Non-bank Authorizer's Phone Number | PIC X(20) |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–20** | **20** | **Customer Phone Number**<br>The customer's telephone number in the Check Authorization 2 token, if available. This field is filled in by the Interchange Interface process or the Device Handler module, and is blank-filled in a response when the information has not been provided in the request. |
| **21–26** | **6** | **Process Control Number**<br>The check process control number in the Check Authorization 2 token. This field is filled in by the Interchange Interface process or the Device Handler module, and is blank-filled in a response when the information has not been provided in the request. |
| **27–51** | **25** | **Non-bank Authorizer's Name**<br>The name of the authorizer, provided in the response by the authorizer. |

| Position | Length | Description |
|---|---|---|
| **52–71** | **20** | **Non-bank Authorizer's Street**<br>The street address of the authorizer, provided in the response by the authorizer. |
| **72–84** | **13** | **Non-bank Authorizer's City**<br>The city of the authorizer, provided in the response by the authorizer. |
| **85–86** | **2** | **Non-bank Authorizer's State**<br>The state of the authorizer, provided in the response by the authorizer. |
| **87–95** | **9** | **Non-bank Authorizer's Postal Code**<br>The postal code of the authorizer, provided in the response by the authorizer. |
| **96–115** | **20** | **Non-bank Authorizer's Phone Number**<br>The phone number of the authorizer, provided in the response by the authorizer. |

## SFID e — Interchange Compliance Data

**Request:** Not applicable.

**Response:** Optional. Fixed length of 21 bytes.

**Internal Field:** Interchange Compliance token

The interchange compliance data field contains interchange compliance data associated with a MasterCard transaction. The format is as follows:

| | |
|---|---|
| Trace ID | PIC X(15) |
| Valid Code | PIC X(4) |
| Monitoring Status | PIC X |
| Error Indicator | PIC X |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01–15** | **15** | **Trace ID**<br>The code assigned by the interchange to a transaction that has met the required compliance edits. A combination of the network ID, reference number, and date is filled into this field, depending on the interchange. |
| **16–19** | **4** | **Valid Code**<br>The code assigned by the interchange to a transaction that has met the required compliance edits and has been approved by the issuer. |
| **20** | **1** | **Monitoring Status**<br>A code returned from the interchange indicating whether MasterCard changed the point-of-service entry mode from 90 to 02. A value of Y indicates that the status is being monitored. |

| Position | Length | Description |
|---|---|---|
| **21** | **1** | **Error Indicator**<br>A code returned from the interchange indicating an error condition that may have occurred. Valid values are as follows:<br>␣ = No error occurred (where ␣ indicates a blank space)<br>A = Track 1 or Track 2 data not present in message<br>B = Track 1 and Track 2 data present in message<br>C = PAN not equal in PAN data<br>D = Expiration date not equal in PAN data<br>E = Card type invalid in track data<br>F = Field separator invalid in track data<br>G = A field within the track data exceeds the maximum length<br>H = Transaction category code is T<br>I = POS customer presence indicator is 1<br>J = POS card presence indicator is 1 |

## SFID f — Response Source or Reason Code

**Request:** Not applicable.

**Response:** Optional. Fixed length of 1 byte.

**Internal Field:** POS Data token

The response source or reason code subfield contains the response source or reason code set by an interchange to provide additional information regarding a response. Valid values are as follows:

1 = Request timed out at interchange
2 = Transaction amount below issuer limit
3 = Issuer is in suppress inquiries mode
4 = Issuer is not available for processing
5 = Response provided by issuer

7 = Reversal advice provided by interchange to identify a potential duplicate transaction
8 = Reversal advice provided by interchange to identify a probable duplicate authorization
A = Third party agent

The processing of EMV transactions assumes that all real-time authorization requests are sent to the card issuer for approval. In some instances, however, an intermediate system (for example, an acquirer or card acceptor) stands in for the issuer and authorizes the transaction. In this situation, the response message does not contain an Authorization Response Cryptogram (ARPC), that is, the Issuer Authentication Data (subFID 6Q) is not present, and the integrated circuit card may subsequently decline the transaction if the response purports to come from the issuer.

To avoid this situation, the intermediate system can use this subfield to indicate that a system other than the card issuer is the authorizing entity. If a value of 1 or 4 is returned in this subfield, then the transaction could not be sent to the issuer for authorization. If the Issuer Authentication Data (subFID 6Q) is not present in the response message, then the terminal should respond to the integrated circuit card with an EMV response code of Y3 (unable to go online, offline approved) or Z3 (unable to go online, offline declined). If a value of 2 or A is returned in this subfield, then the transaction was authorized on behalf of the issuer. If the Issuer Authentication Data (subFID 6Q) is not present in the response message, then the terminal should respond to the integrated circuit card with an EMV response code of Y1 (offline approved) or Z1 (offline declined).

## SFID g — POS Merchant Data

**Request:** Optional. Fixed length of 4 bytes.

**Response:** Not applicable

**Internal Field:** POS Merchant token

The POS merchant data subfield contains additional POS merchant data. The format is as follows:

| | |
|---|---|
| E-Commerce Goods Indicator | PIC X |
| Existing Debt Indicator | PIC X |
| Deferred Billing Indicator | PIC X |
| Relationship Participant Indicator | PIC X |

These fields are further described below:

| Position | Length | Description |
|---|---|---|
| **01** | **1** | **E-Commerce Goods Indicator**<br><br>A code indicating the type of merchandise being sold. This code is passed from the acquiring terminal and can be specific to a transaction or merchant. Valid values are as follows:<br><br>D = Digital<br>P = Physical goods<br>S = Services |
| **02** | **1** | **Existing Debt Indicator**<br><br>A code indicating whether a credit card is used to pay for an existing debt. If a credit card was not used to pay for an existing debt, this field is blank. This field is passed from the acquiring terminal. The only valid value is 9 (Payment on existing debt). |
| **03** | **1** | **Deferred Billing Indicator**<br><br>A code indicating whether a purchase is made with the payment deferred until a later date. This code is passed from the acquiring terminal. Valid values are as follows:<br><br>0 = Deferred billing is not provided.<br>1 = Deferred billing is used at the point of service. |
| **04** | **1** | **Relationship Participant Indicator**<br><br>A code indicating whether the merchant or acquirer has a special relationship with the cardholder. This code is passed from the acquiring terminal. Valid values are as follows:<br><br>0 = Not a relationship participant or relationship not provided.<br>1 = Relationship participant. |

## SFID h — System Trace Audit Number

**Request:** Not applicable

**Response:** Optional. Fixed length of 6 bytes.

**Internal Field:** STAN

This subfield contains a number assigned by a transaction originator to uniquely identify a transaction. The trace number remains unchanged for all messages within the transaction.

## SFID i — Retrieval Reference Number

**Request:** Not applicable

**Response:** Optional. Fixed length of 12 bytes.

**Internal Field:** Switch Common Data token

This subfield contains a reference number supplied by the system that retains the original source information.

## SFID j — Debit Network/Sharing Group ID

**Request:** Not applicable

**Response:** Optional. Fixed length of 4 bytes.

**Internal Field:** Switch Common Data token

This subfield is a network identifier for the message transmission and defines the program rules that apply to the transaction. It is usually set by the interchange network that acquires the transaction.

## SFID k — Card Level Results

**Request:** Not applicable.

**Response:** Optional. Fixed length of 2 bytes.

**Internal Field:** Switch Common Data token

This subfield contains a code indicating the participation program for the card involved in the transaction. The first byte has one of the following values:

| | | |
|---|---|---|
| A | = | Visa Traditional – Identifies any consumer credit card that does not offer rewards or meet the Visa Traditional Rewards product requirements and program standards. Default. |
| B | = | Visa Traditional Rewards – Identifies any consumer credit card offering rewards that meet or exceed the minimum Visa Traditional Rewards product requirements and rewards program standards |
| C | = | Visa Signature – Identifies any consumer credit card offering rewards that meet or exceed the minimum Visa Signature product requirements and rewards program standards. |
| D | = | Visa Infinite – Identifies any consumer credit card offering rewards that meet or exceed the minimum Visa Infinite product requirements and rewards program standards. |
| E-Z, 0-9 | = | Reserved for future use |

The second byte is a blank, A-Z or 0-9. All are reserved values.

## SFID I — Healthcare/Transit Data

**Request:** Optional. Variable length of 20-120 bytes.

**Response:** Optional. Variable length of 20-120 bytes.

**Internal Field:** Healthcare/Transit token

This subfield contains information associated with healthcare/transit auto-substantiation transactions and healthcare eligibility inquiry transactions. The format is as follows:

| | |
|---|---|
| Additional Amount | occurs 1 to 6 times |
| Account Type | PIC X(2) |
| Amount Type | PIC X(2) |
| Currency Code | PIC X(3) |
| Amount Sign | PIC X |

Amount PIC X(12)

For healthcare/transit auto-substantiation transactions, only entry 1 of the Additional Amount table is used.

For healthcare eligibility inquiry transactions, entries 1 through 6 of the Additional Amount table are used.

These fields are further described below.

| Position | Length | Description |
|---|---|---|
| 01-120 | 20-120 | **Additional Amount (occurs 1 to 6 times)**<br>Additional amount information. |
| 01-02 | 2 | **Account Type**<br>The type of account being used. A value of 00 indicates a non-specified type will be used for healthcare/transit auto-substantiation transactions and healthcare eligibility inquiry transactions. |
| 03-04 | 2 | **Amount Type**<br>The type of payment amount. Valid values are as follows:<br>3S = Amount co-payment<br>4S = Amount healthcare<br>4T = Amount transit<br>57 = Original amount |
| 05-07 | 3 | **Currency Code**<br>The standard ISO Currency Code of the amount. |
| 08 | 1 | **Amount Sign**<br>A code indicating whether the amount is positive or negative. Valid values are as follows:<br>C = Credit, positive balance<br>D = Debit, negative balance |
| 09-20 | 12 | **Amount**<br>The amount specified by the Amount Type field (right-justified and zero-filled on the left). |

## SFID m — Healthcare Service Data

**Request:** Optional. Variable length of 19 to 95 bytes.

**Response:** Optional. Variable length of 19 to 95 bytes.

**Internal Field:** Healthcare Service token

This subfield contains information associated with healthcare eligibility inquiry transactions. The format is as follows:

| | |
|---|---|
| Service | occurs 1 to 5 times |
| Provider ID | PIC X(9) |
| Type Code | PIC X(2) |
| Payer ID | PIC X(6) |
| Reason Code | PIC X(2) |

These fields are further described below.

| Position | Length | Description |
|---|---|---|
| 01-95 | 19-95 | **Service (occurs 1 to 5 times)** |
| 01-09 | 9 | **Provider ID**<br>The medical license number of the healthcare service provider. |
| 10-11 | 2 | **Type Code**<br>The healthcare service type code as defined by the Health Insurance Portability and Accountability Act (HIPAA). |
| 12-17 | 6 | **Payer ID**<br>The health insurance carrier/payer ID. |
| 18-19 | 2 | **Reason Code**<br>The eligibility approval or rejection reason code as defined by the HIPAA. |

## SFID n — Error Flag

**Request:** Not applicable.

**Response:** Optional. Fixed length of 1 byte

**Internal Field:** BASE24-pos Release 5.0 token

This subfield contains a code providing additional information about the disposition of the transaction. Valid values are as follows:

| | | |
|---|---|---|
| A | = | BASE24-pos Terminal Data files adjustment limit exceeded |
| C | = | Card verification failed |
| E | = | BASE24-pos Terminal Data files return limit exceeded |
| I | = | Invalid MAC |
| K | = | KMAC synchronization error |
| L | = | Invalid PIN length |
| M | = | MAC failure |
| P | = | Invalid PIN block |
| R | = | Sanity check error—previous zone |
| S | = | Sanity check error |
| T | = | BASE24 system error (token error) |
| U | = | Recurring payment cancellation service |
| V | = | Stop payment order |
| W | = | Revocation of authorization order |
| X | = | Revocation of all authorizations order |
| 1 | = | New account information available for recurring payments transaction |
| 2 | = | Try again later, recurring payments transaction |
| 3 | = | Do not try again for recurring payments transaction |
| blank | = | No information available |

## SFID o — American Express Additional Data

**Request:** Optional. Variable-length of up to 300 bytes.

**Response:** Not applicable.

**Internal Field:** American Express Additional Data token

This subfield contains additional data for American Express transactions. The subFID is variable-length, and the format is as follows:

| | |
|---|---|
| Additional Data ID | PIC X(3) |
| Additional Data | PIC X(297) |

These fields are further described below.

| Position | Length | Description |
|---|---|---|
| 01-03 | 3 | **Additional Data ID**<br>Indicates the type of additional data that follows. Valid values are as follows.<br>ITD = Card Not Present Data (mail, telephone and internet order)<br>APD = Airline Passenger Data |
| 04-300 | 0-297 | **Additional Data**<br>Contains either Card Not Present Data or Airline Passenger Data depending on the value of the Additional Data ID field. |
| 04-270 | 0-267 | **Card Not Present Data**<br>The following data fields may be used when the Additional Data ID field is set to a value of ITD. Each data field is preceded by a three-byte data ID and a two-byte data length indicator. Each data ID/data length/data field group is optional, and the groups may occur in any order. |
| | 3 | **Data ID**<br>CE~ = Customer Email (~ = blank character) |
| | 2 | **Data Length**<br>Length of the following field. |
| | 1-60 | **Customer Email**<br>Customer's email address. |
| | 3 | **Data ID**<br>CH~ = Customer Host Name (~ = blank character) |
| | 2 | **Data Length**<br>Length of the following field. |

| Position | Length | Description |
|---|---|---|
| | 1-60 | **Customer Host Name**<br>Name of the server to which the customer is connected. |
| | 3 | **Data ID**<br>HBT = HTTP Browser Type |
| | 2 | **Data Length**<br>Length of the following field. |
| | 1-60 | **HTTP Browser Type**<br>Customer's HTTP browser type. |
| | 3 | **Data ID**<br>STC = Ship To Country |
| | 2 | **Data Length**<br>Length of the following field. |
| | 3 | **Ship To Country**<br>Three-byte numeric ISO country code. |
| | 3 | **Data ID**<br>SM~ = Shipping Method (~ = blank character) |
| | 2 | **Data Length**<br>Length of the following field. |
| | 2 | **Shipping Method**<br>Shipping method code. Valid values are as follows:<br>01 = Same day<br>02 = Overnight/next day<br>03 = Priority, 2-3 days<br>04 = Ground, 4 or more days<br>05 = Electronic delivery<br>06–ZZ = Reserved for future use |
| | 3 | **Data ID**<br>MPS = Merchant Product SKU |

| Position | Length | Description |
|---|---|---|
| | 2 | **Data Length**<br>Length of the following field. |
| | 1-15 | **Merchant Product SKU**<br>Unique SKU (Stock Keeping Unit) inventory reference number of the product associated with this authorization request. For multiple items, enter the SKU of the most expensive item. |
| | 3 | **Data ID**<br>+IP = Customer IP (ACI-defined data ID) |
| | 2 | **Data Length**<br>Length of the following field. |
| | 15 | **Customer IP**<br>Customer's internet IP address in the following format: nnn.nnn.nnn.nnn |
| | 3 | **Data ID**<br>+AN = Customer ANI (ACI-defined data ID) |
| | 2 | **Data Length**<br>Length of the following two fields. |
| | 10 | **Customer ANI**<br>ANI (Automatic Number Identification). The specified phone number that the customer used to place the order with the merchant. |
| | 2 | **Customer II Digits**<br>Telephone company-provided ANI II (Information Identifier) code digits that are associated with the Customer ANI phone number and correspond to the call type (e.g., cellular, government institution). |

| Position | Length | Description |
|---|---|---|
| 04-300 | 0-297 | **Airline Passenger Data**<br><br>The following data fields may be used when the Additional Data ID field is set to a value of APD. Each data field is preceded by a three-byte data ID and a two-byte data length indicator. Each data ID/data length/data field group is optional and the groups may occur in any order. |
| | 3 | **Data ID**<br><br>+DD = Departure Date (ACI-defined data ID) |
| | 2 | **Data Length**<br><br>Length of the following field. |
| | 8 | **Departure Date**<br><br>Departure date in the format YYYYMMDD. |
| | 3 | **Data ID**<br><br>APN = Airline Passenger Name |
| | 2 | **Data Length**<br><br>Length of the following field. |
| | 23-40 | **Passenger Name**<br><br>Passenger name in the following format where ~ equals a blank character:<br><br>SURNAME~ FIRSTNAME~MIDDLEINITIAL~TITLE<br><br>Data must be 23 bytes minimum, left-justified, and blank-filled as needed. 40 bytes maximum, truncate if necessary. |
| | 3 | **Data ID**<br><br>CN~ = Cardmember Name (~ = blank character) |
| | 2 | **Data Length**<br><br>Length of the following field. |

| Position | Length | Description |
|---|---|---|
| | 23-40 | **Cardmember Name**<br>Cardmember name in the following format where ~ equals a blank character:<br>SURNAME~FIRSTNAME~MIDDLEINITIAL~TITLE<br>Data must be 23 bytes minimum, left-justified and blank-filled as needed. 40 bytes maximum, truncate if necessary. |
| | 3 | **Data ID**<br>+AP = Origination/Destination Airport (ACI-defined data ID) |
| | 2 | **Data Length**<br>Length of the following two fields. |
| | 5 | **Origination Airport**<br>Origination airport for the first segment of the trip. Five-character airport code allows for the anticipated expansion of the current three-character code. Left-justify and blank-fill as needed. |
| | 5 | **Destination Airport**<br>Destination airport for the first segment of the trip; not necessarily the final destination. Five-character airport code allows for the anticipated expansion of the current three-character code. Left-justify and blank-fill as needed. |
| | 3 | **Data ID**<br>RTG = Routing |
| | 2 | **Data Length**<br>Length of the following two fields. |
| | 2 | **Number of Cities**<br>Number of airports or cities on the ticket, maximum of 10. |

| Position | Length | Description |
|---|---|---|
| | 11-59 | **Routing Cities**<br>Routing airport or city code for each leg on the ticket (including Origination Airport and Destination Airport), in five-byte segments each separated by a "/". Example: "ABC~~/DEF~~/GHI~~", where ~ equals a blank character. |
| | 3 | **Data ID**<br>ALC = Airline Carriers |
| | 2 | **Data Length**<br>Length of the following two fields. |
| | 2 | **Number of Airline Carriers**<br>Number of airline carrier entries in the following field. Maximum of 9. |
| | 5-53 | **Airline Carriers**<br>Airline carrier code for each leg on the ticket (including Origination Airport and Destination Airport), in five-byte segments each separated by a "/". Each leg must have an airline carrier code entry, even if multiple (or all) legs are on the same airline. Example: "AB~~~/AB~~~", where ~ = blank character. |
| | 3 | **Data ID**<br>+FR = Fare Data (ACI-defined data ID) |
| | 2 | **Data Length**<br>Length of the following three fields. |
| | 24 | **Fare Basis**<br>Primary and secondary discount codes indicating the class of service and fare level associated with the ticket. Truncate to 24 bytes, if necessary. |
| | 3 | **Number of Passengers**<br>Number of passengers in the party. |

| Position | Length | Description |
|---|---|---|
| | 1 | **E-Ticket Indicator**<br>Indicates if the ticket is electronic. Valid values are as follows.<br>E = electronic ticket<br>blank = non-electronic ticket |
| | 3 | **Data ID**<br>RES = Reservation Code |
| | 2 | **Data Length**<br>Length of the following field. |
| | 6-15 | **Reservation Code**<br>A precursor to a ticket number. Corresponds to an airline ticket purchase made by an airline or a global distribution system (GDS). |

# Optional Data Subfields — FID 7

The optional data subfields for FID 7 (Product SubFIDs) are summarized in a table below, followed by individual descriptions of the subfields.

**Note:** The information in FIDs 6 through 9 are contained in subfields. These subfields are included in the request or response message when FIDs 6 through 9 are specified in the host configuration. Specifying FIDs 6 through 9 in the host configuration specifies all subfields for FIDs 6 through 9, respectively. FIDs 6, 7, and 8 are reserved for product use. FID 9 is reserved for customer use.

## Summary List

The subfields for FID 7 are described below according to their subfield identifiers (SFIDs). The table lists the SFID, the length of the subfield in the message, its associated field name, and the name of the internal field to which it is mapped by the SPDH module. In addition, a check mark (✓) appears in the RQST and RESP columns if a subfield is available for requests and responses, respectively.

| SFID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| a | 1–40 bytes | Mobile Top-up Track 2 | Pre-Pay Top-Up token | ✓ | ✓ |
| b | 15 bytes | Original Mobile Top-Up Reference Number for Refunds (For future use) | Pre-Pay Top-Up token | ✓ | |
| c | 65 bytes | Mobile Top-Up Response | Pre-Pay Top-Up token | | ✓ |

## SFID a — Mobile Top-Up Track 2

**Request:** Optional. Variable length of 1–40 bytes.

**Response:** Optional. Variable length of 1–40 bytes.

**Internal Field:** Pre-Pay Top-Up token

For requests, this subFID contains the mobile top-up Track 2. This subFID is required for mobile top-up transactions.

For responses, this subFID contains the mobile top-up Track 2.

## SFID b — Original Mobile Top-Up Reference Number (For future use)

**Request:** Optional. Fixed length of 15 bytes.

**Response:** Optional. Fixed length of 15 bytes.

**Internal Field:** Pre-Pay Top-Up token

For mobile top-up refund requests, this subFID contains the original mobile top-up reference number for refunds used to match the refund with the original request. This subFID is reserved for future use.

For responses, this subFID contains the original mobile top-up reference number for refunds used to match the refund with the original request. This subFID is reserved for future use.

## SFID c — Mobile Top-Up Response

**Request**: Not applicable.

**Response**: Optional. Variable length of 1–65 bytes

**Internal Field:** Pre-Pay Top-Up token

For responses, this subFID contains the pre-pay mobile top-up response. It includes the following fields:

| Position | Length | Description |
|---|---|---|
| 1–16 | 16 | **Top-UP** – Reference number returned for all top-up transactions returned from the mobile operator.<br>or<br>**Activation Code** – For all top-up transactions approved from the inventory stock manager. |
| 17–32 | 16 | **Operator Name**<br>The name of the mobile operator as defined in the Mobile Operator File (MOF). |

| Position | Length | Description |
|---|---|---|
| 33–47 | 15 | **Top-Up Approval Code**<br><br>Present on all approved top-up with purchase and refund top-up with purchase transactions as specified by the mobile operator. |
| 48–65 | 18 | **Other Balance**<br><br>The available time left on pre-pay phone as specified by the mobile operator or can also be an available balance or a tax amount. |

# Optional Data Subfields — FID 8

The optional data subfields for FID 8 (Product SubFIDs) are summarized in a table below, followed by individual descriptions of the subfields.

**Note:** The information in FIDs 6 through 9 are contained in subfields. These subfields are included in the request or response message when FIDs 6 through 9 are specified in the host configuration. Specifying FIDs 6 through 9 in the host configuration specifies all subfields for FIDs 6 through 9, respectively. FIDs 6, 7, and 8 are reserved for product use. FID 9 is reserved for customer use.

## Summary List

The subfields for FID 8 are described below according to their subfield identifiers (SFIDs). The table lists the SFID, the length of the subfield in the message, its associated field name, and the name of the internal field to which it is mapped by the SPDH module. In addition, a check mark (✓) appears in the RQST and RESP columns if a subfield is available for requests and responses, respectively.

| SFID | Length | Field Name | Internal Field | RQST | RESP |
|---|---|---|---|---|---|
| A | 24 bytes | EBT Voucher Number | EBT Voucher Number token | ✓ | |
| | 18 bytes | EBT Available Balance | EBT Available Balance token | | ✓ |
| B | 18 bytes | EBT Available Balance | EBT Available Balance token | | ✓ |

## SFID A — EBT Voucher Number or EBT Available Balance

**Request:** Optional. Fixed length of 24 bytes.

**Response:** Optional. Fixed length of 18 bytes.

**Internal Field:** EBT Voucher Number token or
EBT Available Balance token

For requests, this subFID can contain the electronic benefits transfer (EBT) voucher number. This subFID is required for manually entered EBT transaction requests that were voice authorized.

For responses, this subFID contains the available balance for a cash account.

**Note:** Most EBT authorization systems require balances to be printed on a receipt. The terminal may need to format the balance information returned in this subFID for printing on a customer receipt.

## SFID B — EBT Available Balance

**Request:** Not applicable.

**Response:** Optional. Fixed length of 18 bytes.

**Internal Field:** EBT Available Balance token

In EBT transaction responses, the EBT Available Balance field contains the available balance for a food stamp account.

**Note:** Most EBT authorization systems require balances to be printed on a receipt. The terminal may need to format the balance information returned in this subFID for printing on a customer receipt.

# Request Message Requirements

The customer and vendor have the option of including any number of optional data fields in requests, as long as the maximum size of the terminal read buffer and the host read buffer are not exceeded. The host read buffer is configurable to a maximum of 4,088 bytes, including all data communication control characters. When the customer determines that any of these data fields must be included in requests for certain transactions, those fields are specified in the ACI Standard Device Configuration File (ACNF). The SPDH module then enforces these fields as being required. However, the terminal cannot use the ACNF to determine the fields to include in its requests. The terminal itself must also be configured to determine the fields it requires for certain transactions.

For more information on required optional data fields for different transaction requests, refer to section 8 and the ***ACI Standard POS Device Message Specifications Manual***.

**Note:** Either FID q or FID 2 is required for all financial transaction requests. The customer selects the preferred track on Card Prefix File (CPF) screen 1. The SPDH module checks each financial transaction for the presence of either FID q or FID 2 and declines those transactions that have neither FID.

# Response Message Requirements

The SPDH module formats responses to BASE24-pos Standard POS terminals as indicated in the ACI Standard Device Configuration File (ACNF). The SPDH module formats responses using the fields identified in the ACNF as being required. Any optional data fields to be included in responses are listed in the ACNF.

Some fields may be included in responses regardless of the ACNF data, including: FID H, the Authentication Key; and FID V, the Mail/Download Key. However, these are returned only when found to be applicable. The Authentication Key is included when a new MAC communications key is generated due to the configured number of consecutive messages failing to be verified when using message authentication codes (MACs).

If the SPDH module is unable to format a response message due to errors in the ACNF, the response message consists of a header only, with the response code set to 821, indicating an invalid response length. A possible error in the ACNF could be that the format specified results in a response longer than the maximum allowable response.

**Note:** FIDs V and W are the only FIDs allowed in the response message for a download.

For more information on required optional data fields for different transaction responses, refer to section 8 and the ***ACI Standard POS Device Message Specifications Manual***.

# Determining Transaction Codes

The SPDH module determines the type of BASE24-pos transaction associated with an ACI standard POS message using two fields in the standard header. These two fields are the Message Type and the Transaction Code. The Message Type field specifies the transaction as being either a financial transaction (F) or an administrative transaction (A). The Transaction Code field identifies the type of transaction, as determined by the terminal.

The BASE24-pos transaction code, in addition to indicating the type of transaction, identifies whether a debit or a credit card initiated the transaction, and the type of application account used when the transaction was initiated by a debit card. In the case of debit cards, the type of application account is required in the transaction code because debit cards can be associated with checking, savings, or credit card accounts.

The list below identifies the message types and transaction codes sent in the standard message header and the BASE24 transaction codes subsequently computed by the SPDH module. The BASE24 transaction codes listed below include the letter *t* to indicate the position of the code identifying the type of card that generated the transaction. Also, the BASE24 transaction codes include the letters *aa* to indicate the position of the code that identifies the type of application account used in transactions initiated by debit cards. Refer to the ***BASE24-pos Transaction Processing Manual*** for a listing of valid BASE24-pos card types and account types.

| Message Type | Terminal Transaction Code | Description | BASE24-pos Transaction Code |
|---|---|---|---|
| F | 00 | Normal purchase | 10*taa* |
| F | 01 | Preauthorization purchase | 11*taa* |
| F | 02 | Preauthorization purchase completion | 12*taa* |
| F | 03 | Mail or telephone order | 13*taa* |
| F | 04 | Merchandise return | 14*taa* |
| F | 05 | Cash advance | 15*taa* |
| F | 06 | Card verification | 16*taa* |

| Message Type | Terminal Transaction Code | Description | BASE24-pos Transaction Code |
|---|---|---|---|
| F | 07 | Balance inquiry | 17*taa* |
| F | 08 | Purchase with cash back | 18*taa* |
| F | 09 | Check verification | 19*taa* |
| F | 10 | Check guarantee | 20*taa* |
| F | 11 | Purchase adjustment | 21*taa* |
| F | 12 | Merchandise return adjustment | 22*taa* |
| F | 13 | Cash advance adjustment | 23*taa* |
| F | 14 | Cash back adjustment | 24*taa* |
| F | 15 | Card activation | 25*taa* |
| F | 16 | Additional card activation | 26*taa* |
| F | 17 | Replenishment | 27*taa* |
| F | 18 | Full redemption | 28*taa* |
| F | 19 | Top-up cash | 29000 |
| F | 20 | Top-up with funds | 30*taa* |
| F | 21 | Refund top-up cash | 31000 |
| F | 22 | Refunds top-up with funds | 32*taa* |
| A | 50 | Logon request | Not applicable* |
| A | 51 | Logoff request | Not applicable* |
| A | 60 | Close batch request | 50*taa* |
| A | 61 | Close shift request | 51*taa* |
| A | 62 | Close day request | 52*taa* |
| A | 64 | Employee subtotals request | Not applicable* |

| Message Type | Terminal Transaction Code | Description | BASE24-pos Transaction Code |
|---|---|---|---|
| A | 65 | Batch subtotals request | Not applicable* |
| A | 66 | Shift subtotals request | Not applicable* |
| A | 67 | Day subtotals request | Not applicable* |
| A | 70 | Read mail request | 56*taa* |
| A | 71 | Mail delivered request | Not applicable* |
| A | 75 | Send mail request | 58*taa* |
| A | 90 | Download request | Not applicable* |
| A | 95 | Handshake request | Not applicable* |
| A | 96 | Key change request | Not applicable* |

* The transaction is submitted as a card verification (16*taa*) if administrative approval is required.

# 5: Download Data

The BASE24 ACI Standard POS Device Handler supports full and partial downloads to terminals. Downloads provide a way to transport and load the data necessary to configure a terminal. Configuration data downloaded to terminals is set up in the ACI Standard Device Configuration File (ACNF), the Acquirer Processing Code File (APCF), the BASE24-pos Terminal Data files (PTD), and the POS Retailer Definition File (PRDF). This section provides the following information on SPDH module download support:

- Downloading records
- Data elements and download field identifiers (DIDs)
- Data element records
- Full download requests and responses
- Partial download requests and responses

# Downloading Data to Terminals

BASE24-pos Standard POS terminals must request download information from the BASE24 Standard POS Device Handler (SPDH) module before they can receive any of the configuration data required for processing. When BASE24-pos Standard POS terminals request download information, the SPDH module accesses the ACI Standard Device Configuration File (ACNF). This file contains up to 12 records for each terminal group. Eleven of these records contain configuration data that can be downloaded to the terminal. One of these records contains BASE24-pos Terminal Data files (PTD) fields, Acquirer Processing Code File (APCF) fields, and one field from the POS Retailer Definition File (PRDF) that can be included in the download.

Configuration data downloaded to the terminal from the ACNF records depends on the request sent from the terminal. Terminals can request a full download, which results in all of the configuration data contained in the up to 12 ACNF records being sent to the terminal. Terminals can also request a partial download, asking for a single download field.

The data elements to be included within the ACNF records are determined by the customer. Information that can be included within these records includes telephone numbers, floor limits, and receipt data.

The information in these records is applicable for a group of terminals. Terminal groups are identified in the TERM GROUP field on PTD screen 1. When a download request is received from a terminal, the SPDH module checks the PTD to determine the group to which the terminal belongs. The SPDH module then accesses the appropriate records in the ACNF from which configuration data is to be taken.

# Download Records

Each terminal group has up to 12 download records in the ACNF. They are designated with record identifiers of 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 10, and 11. Records 00, 01, and 02 contain 26 40-byte fields. Records 03, 04, 05, 07, 08, 09, 10, and 11 contain processing parameters for 30 card prefix ranges. Record 06 contains 26 flags, indicating what fields from the PTD and PRDF are downloaded to the terminal. The download fields are set up in the ACNF using files maintenance functions. Refer to section 8 for more information on the download fields in the ACNF.

## Downloading ACNF Records

The information in the ACNF can be downloaded to the terminal. If the REL-NUM field in the BASE24-pos Terminal Data files (PTD) is set to 50 (release 5.0 and above), the SPDH module attempts to retrieve ACNF records in the following order: 00, 01, 02, 03, 04, 05, 07, 08, 09, 10, 11, and 06. Data is not required in each record in order for the SPDH module to perform downloads. The SPDH module skips over records that do not contain data and moves to the next record.

For example, the SPDH module attempts to retrieve record 00. If record 00 is found, the data from this record is downloaded and the SPDH module attempts to retrieve record 01. If record 00 is not found, the SPDH module does not download any records for record 00 and attempts to retrieve record 01.

In addition, the SPDH module continues to download card prefix range data (records 03, 04, 05, 07, 08, 09, 10, and 11) until the first blank card prefix range is found in the ACNF. The SPDH module then stops searching for card prefix range data and moves to record 06. For example, the SPDH module locates record 03 and all four card prefixes are defined in the record. The SPDH module formats this data in the download data message and attempts to locate record 04. If record 04 is not found, the SPDH module attempts to locate record 06 (i.e., the SPDH module does not attempt to locate records 05 and 07 through 11). As another example, assume the SPDH module locates record 03 and the third card prefix range in the record contains blanks. In this case, the SPDH module downloads only the first two card prefix ranges, then stops attempting to locate card prefix ranges and moves to record 06.

# Download Record Format

When a terminal requests a download, the response from the SPDH module must include FID V and FID W. FIDs V and W provide the processing mechanisms required in order for the SPDH module to perform downloads. The functions of these FIDs include initiating the download, notifying the terminal that download data exists, and transporting the download text to the terminal. FID V and FID W are described in detail in the ***ACI Standard POS Device Message Specifications Manual***.

# Defining Data Elements

Data elements are defined and set up by customers in the ACNF. Each data element that can be downloaded to the terminal is identified with a download field identifier (DID). The range of identified DIDs is A through Z, 0 through 29, and a through z.

Instructions for entering information in these download fields using files maintenance are included in section 8. The table shown below identifies the DIDs, specifies their location within the ACNF records, and provides a brief description of the type of information that can be entered for each DID.

| DIDs | Record | Description |
|---|---|---|
| A, B, C, D, E, F, G, H, I, J | 00 | The DIDs (A through J) associated with record 00 can consist of any information the terminal owner and operator want to include or the terminal vendor requires. This record is optional. Overall, DIDs A through Z can contain any type of data that the terminal requires. |
| K, L, M, N, O, P, Q, R, S, T | 01 | The DIDs (K through T) associated with record 01 can consist of any information the terminal owner and operator want to include or that the terminal vendor requires. This record is optional. Overall, DIDs A through Z can contain any type of data that the terminal requires. |
| U, V, W, X, Y, Z | 02 | The DIDs (U through Z) associated with record 02 can consist of any information the terminal owner and operator want to include or that the terminal vendor requires. This record is optional. Overall, DIDs A through Z can contain any type of data that the terminal requires. |
| 0, 1, 2, 3 | 03 | Record 03 contains card prefix ranges and associated card prefix range information. Up to 30 card prefix ranges can be defined in records 03, 04, 05, 07, 08, 09, 10, and 11. Record 03 contains data for up to four card prefixes. |

| DIDs | Record | Description |
|---|---|---|
| 4, 5, 6, 7 | 04 | Record 04 contains card prefix ranges and associated card prefix range information. Up to 30 card prefix ranges can be defined in records 03, 04, 05, 07, 08, 09, 10, and 11. Record 04 contains data for up to four card prefixes. |
| 8, 9 | 05 | Record 05 contains card prefix ranges and associated card prefix range information. Up to 30 card prefix ranges can be defined in records 03, 04, 05, 07, 08, 09, 10, and 11. Record 05 contains data for two card prefixes. |
| a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z | 06 | The DIDs (lower case a through z) associated with record 06 contain flags that indicate whether specific corresponding fields in the BASE24-pos Terminal Data files (PTD), Acquirer Processing Code File (APCF), and POS Retailer Definition File (PRDF) should be downloaded to the terminal. |
| 10, 11, 12, 13 | 07 | Record 07 contains card prefix ranges and associated card prefix range information. Up to 30 card prefix ranges can be defined in records 03, 04, 05, 07, 08, 09, 10, and 11. Record 07 contains data for up to four card prefixes. |
| 14, 15, 16, 17 | 08 | Record 08 contains card prefix ranges and associated card prefix range information. Up to 30 card prefix ranges can be defined in records 03, 04, 05, 07, 08, 09, 10, and 11. Record 08 contains data for up to four card prefixes. |
| 18, 19, 20, 21 | 09 | Record 09 contains card prefix ranges and associated card prefix range information. Up to 30 card prefix ranges can be defined in records 03, 04, 05, 07, 08, 09, 10, and 11. Record 09 contains data for up to four card prefixes. |
| 22, 23, 24, 25 | 10 | Record 10 contains card prefix ranges and associated card prefix range information. Up to 30 card prefix ranges can be defined in records 03, 04, 05, 07, 08, 09, 10, and 11. Record 10 contains data for up to four card prefixes. |

| DIDs | Record | Description |
|---|---|---|
| 26, 27, 28, 29 | 11 | Record 11 contains card prefix ranges and associated card prefix range information. Up to 30 card prefix ranges can be defined in records 03, 04, 05, 07, 08, 09, 10, and 11. Record 11 contains data for up to four card prefixes. |

# Records 00, 01, 02

The data elements contained in records 00, 01, and 02 can consist of any information the terminal owner and operator want to include or that the terminal vendor requires. All of these records are optional.

The data elements are identified with DIDs A through Z. DIDs A through J are stored in record 00. DIDs K through T are stored in record 01. DIDs U through Z are stored in record 02.

Each of the DIDs can be associated with up to 40 alphanumeric characters entered in the ACI Standard Device Configuration File (ACNF) through BASE24 files maintenance. An example of data that might be included in these data elements is receipt information. The SPDH module allows customers to configure receipts differently for various terminals or terminal groups.

The data elements contained in records 00, 01, and 02 are sent only if they are requested by the terminal and if they contain data. If the terminal requests these data elements, but the data elements contain no data, they are not sent to the terminal.

# Records 03, 04, 05, 07, 08, 09, 10, 11

The data elements contained in records 03, 04, 05, 07, 08, 09, 10, and 11 consist of card prefix range information defined by the terminal owner and operator. Up to 30 card prefix ranges can be defined in these records. Each card prefix range is identified with a DID of 0 through 29. The DIDs are stored in the following manner:

- DIDs 0 through 3 are stored in record 03
- DIDs 4 through 7 are stored in record 04
- DIDs 8 through 9 are stored in record 05
- DIDs 10 through 13 are stored in record 07
- DIDs 14 through 17 are stored in record 08
- DIDs 18 through 21 are stored in record 09
- DIDs 22 through 25 are stored in record 10
- DIDs 26 through 29 are stored in record 11

Data concerning the card prefix ranges is entered in the ACI Standard Device Configuration File (ACNF) through BASE24 files maintenance. The instructions for setting up card prefix ranges in the ACNF are described in section 8. The data elements in records 03, 04, 05, 07, 08, 09, 10, and 11 are sent to the terminal only if they are requested by the terminal and if they contain data.

**Note:** For releases 5.0 and above, the SPDH module can support up to 30 card prefix ranges. In prior releases, only 10 card prefix ranges were supported. In order for the SPDH module to support up to 30 card prefixes, the REL-NUM field in the BASE24-pos Terminal Data files (PTD) must be set to 50 (release 5.0 or above). The REL-NUM field identifies the BASE24-pos message format used by the terminal. This field is used by the SPDH module to determine the BASE24-pos release number of the message format the terminal understands. If the REL-NUM field is set to 50, the SPDH module can download up to 30 card prefix ranges. If the REL-NUM field is not set to 50, only 10 card prefix ranges can be downloaded. In this situation, if an operator attempts to download more than 10 card prefix ranges (records 07 through 11), only the first 10 are successful. In addition, a message is logged indicating that 30 card prefix ranges cannot be downloaded for the release number specified in the REL-NUM field. An 880 response code is also sent indicating no more data exists.

# Card Prefix Data Element Structures

The following table describes the data element structure of the card prefix information included in DIDs 0 through 29. The table includes the positions in the DID occupied by each card prefix data element, a picture clause containing the field length and format of the card prefix data element, and the name of the card prefix data element. The card prefix information associated with records 03, 04, 05, 07, 08, 09, 10, and 11 consist of 108 alphanumeric characters per prefix, organized in the following format.

**Card Prefix Data Element Structures for Records 03, 04, 05, 07, 08, 09, 10, and 11**

| Position | Picture | Data Element Name |
|---|---|---|
| 1–11 | PIC X(11) | Low Prefix |
| 12–22 | PIC X(11) | High Prefix |
| 23–42 | PIC X(20) | Main Network Telephone Number |
| 43–62 | PIC X(20) | Backup Network Telephone Number |
| 63–82 | PIC X(20) | Referrals Telephone Number |
| 83–94 | PIC X(12) | Retailer ID |
| 95 | PIC X(01) | Draft Capture |
| 96 | PIC X(01) | Totals |
| 97 | PIC X(01) | PIN |
| 98 | PIC X(01) | Receipt |
| 99 | PIC X(01) | Mod-Check |
| 100 | PIC X(01) | PAN Fraud Check |
| 101–108 | PIC X(08) | User Defined Area |

The card prefix data elements listed above are taken from records 03, 04, 05, 07, 08, 09, 10, and 11 in the ACNF. For descriptions of the card prefix data elements, refer to the “Download Records 03, 04, 05, 07, 08, 09, 10, and 11 Screens” topic documented in section 8.

# Record 06

The data elements in record 06 consist of flags corresponding to fields in the BASE24-pos Terminal Data files (PTD), Acquirer Processing Code File (APCF), and the POS Retailer Definition File (PRDF). These flags are used to determine the PTD and PRDF fields to be downloaded. Record 06 can store 26 download flags. DIDs a through z identify the 26 data elements that can be included in record 06, although only 13 of the data elements are currently supported.

Flags concerning the data elements are set in the ACI Standard Device Configuration File (ACNF) through BASE24 files maintenance. The instructions for setting up data elements in the ACNF are described in section 8. The data elements in record 06 are sent to the terminal only if they are requested by the terminal and if they contain data.

## Data Element Structures

The following table describes the structure of the data elements associated with DIDs a through z. The table includes the DID associated with each data element, a picture clause containing the field length and format of the data element, and the name of the data element. The data elements contained in record 06 are organized in the following format.

| Data Element Structures for Record 06 | | |
|---|---|---|
| **DID** | **Picture** | **Data Element Name** |
| a | PIC X(25) | TERM-NAME-LOC |
| b | PIC X(16) | TERM-CITY-ST |
| c | PIC X(22) | TERM-OWNER-LOC |
| d | N/A | RESERVED |
| e | N/A | RESERVED |
| f | PIC X(88) | SERVICE-REP |
| g | PIC X(48) | P-KEY |
| h | PIC X(48) | A-KEY |
| i | PIC X(01) | PIN PAD CHARACTER |

| Data Element Structures for Record 06 | | |
|---|---|---|
| **DID** | **Picture** | **Data Element Name** |
| j | PIC X(48) | DATA ENCRYPTION KEY |
| k | PIC X(1170) | SERVICE (OCCURS 30 TIMES) |
| l | PIC X(44) | LIMITS |
| m | N/A | RESERVED |
| n | N/A | RESERVED |
| o | N/A | RESERVED |
| p | PIC X(30) | ALLOWED-TRANS |
| q | PIC X(19) | RETAILER-ID |
| r | PIC X(40) | MERCHANT NAME |
| s | PIC X(20) | REFERRAL PHONE # |
| t | N/A | RESERVED |
| u | N/A | RESERVED |
| v | N/A | RESERVED |
| w | N/A | RESERVED |
| x | N/A | RESERVED |
| y | N/A | RESERVED |
| z | N/A | RESERVED |

## Data Element Descriptions

The following pages provide descriptions of the data elements that can be taken from record 06 in the ACNF. The descriptions include any values associated with the data element. For elements that consist of multiple fields, the structure of the elements are shown below. Included for each field is the field position in the element, the field length, and a description of its contents if necessary.

**TERM-NAM-LOC —** DID a contains the terminal name/description for printing on receipts and to comply with Regulation E. This element is taken from the POS Terminal Data Static File—general data (PTDS1).

**TERM-CITY-ST —** DID b contains the city and state in which this terminal is located for printing receipts and to comply with Regulation E. The following fields are taken from the PTDS1:

| Position | Length | Description |
|---|---|---|
| 1–13 | 13 | CITY |
| 14–16 | 3 | STATE |

**TERM-OWNER-NAME —** DID c contains the name of the retailer who owns the terminal. This element is taken from the PTDS1.

**RESERVED —** DID d is reserved for future use.

**RESERVED —** DID e is reserved for future use.

**SERVICE-REP —** DID f contains the name, address, and telephone number of the service representative to contact when problems occur with this terminal. The following fields are taken from the PTDS1:

| Position | Length | Description |
|---|---|---|
| 01–25 | 25 | NAME |
| 26–50 | 25 | ADDRESS |
| 51–63 | 13 | CITY |
| 64–66 | 3 | STATE |
| 67–68 | 2 | COUNTRY |
| 69–88 | 20 | PHONE |

**P-KEY —** DID g contains the PIN encryption key used by this terminal. This element is taken from the POS Terminal Data Dynamic File—general data (PTDD1). The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

**A-KEY —** DID h contains the message authentication code (MAC) generation key used by this terminal. This element is taken from the PTDD1. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

**PIN PAD CHARACTER —** DID i contains the character used to pad the PIN block. This element is taken from the PTDD1.

**DATA ENCRYPTION KEY —** DID j contains the data encryption key used by this terminal. This element is taken from the PTDD1. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

### SERVICE (OCCURS 30 TIMES)

DID k contains the card processing parameters for this terminal. These parameters are taken from the PTDD1.

**Note:** Release 3.4 and newer releases of the SPDH module support functionality that allows downloading of fields whose lengths typically exceed the maximum response length of the SPDH module. The only field to which this enhancement currently applies is DID k. The following paragraph describes the way in which the SPDH module processes fields whose lengths exceed the maximum response length allowed by the SPDH module.

When DID k is configured to be included in a full download to a release 3.4 or newer terminal, the SPDH module adds as many occurrences as will fit to the current response. The SPDH module then sends the response to the terminal with the first byte of FID V (Main/Download Key) set as an ASCII 1 to indicate to the terminal that more data remains in this field. Next, the terminal can request the remaining portion of this field by changing the second byte of FID V to an ASCII 1 and using the resulting FID V in its next request, or the terminal can proceed to the next DID by echoing FID V unaltered. The following structure represents one occurrence of this field. This field can occur up to 30 times.

| Position | Length | Description |
|---|---|---|
| **01–02** | **2** | **TYP**<br>Indicates the type of card accepted at this terminal. |
| **03–11** | **9** | **NP-FLOOR-LIM**<br>The floor limit as used by BASE24, in whole currency units, for normal purchase transactions performed at this terminal for this card type. |
| **12–20** | **9** | **CA-FLOOR-LIM**<br>The floor limit as used by BASE24, in whole currency units, for cash advance transactions performed at this terminal for this card type. |
| **21–29** | **9** | **MO-FLOOR-LIM**<br>The floor limit as used by BASE24, in whole currency units, for mail or telephone order transactions performed at this terminal for this card type. |
| **30–38** | **9** | **TRAN-LIMIT**<br>The transaction amount limit for this card type as used by BASE24, in whole currency units, for this terminal. Transactions for amounts exceeding this limit are denied. This limit does not apply to cards with VIP status. |

| Position | Length | Description |
|---|---|---|
| 39 | 1 | **TRAN-PROFILE**<br>A code indicating the transaction profile. Valid values are as follows:<br>0 = Authorize only<br>1 = Authorize and capture<br>2 = Authorize and expect electronic follow up<br>3 = Terminal determines data capture mode for each transaction |

**LIMITS —** DID l contains the limit parameters for the terminal. The following parameters are taken from the PTDS1:

| Position | Length | Description |
|---|---|---|
| 01–04 | 4 | **ADJ-LIMIT-CNT**<br>The maximum number of adjustment transactions allowed to be performed on this terminal for each batch. When this limit is exceeded, an event message is generated. |
| 05–22 | 18 | **ADJ-LIMIT-AMT**<br>The maximum amount, in whole and fractional currency units, that can be accepted at this terminal using adjustment transactions. This limit is invoked for each batch. When this limit is exceeded, an event message is generated. |
| 23–26 | 4 | **RETURN-LIMIT-CNT**<br>The maximum number of merchandise return transactions allowed to be performed on this terminal for each batch. When this limit is exceeded, an event message is generated. |

| Position | Length | Description |
|---|---|---|
| **27–44** | **18** | **RETURN-LIMIT-AMT**<br>The maximum amount, in whole and fractional currency units, that can be accepted at this terminal using merchandise return transactions. This limit is invoked for each batch. When this limit is exceeded, an event message is generated. |

**RESERVED —** DID m is reserved for future use.

**RESERVED —** DID n is reserved for future use.

**RESERVED —** DID o is reserved for future use.

**ALLOWED-TRANS —** DID p contains the following fields used to determine if a transaction is allowed at this terminal. Whenever a transaction is allowed, a 1 is placed in the appropriate field shown below. A 0 indicates the transaction is not allowed. These elements are taken from the APCF and PTDS1. The following list contains the name of the field controlling the transaction. The length of each field shown below is one byte.

**Normal Purchase**
**Preauthorization Purchase**
**Preauthorization Purchase Completion**
**Mail or Telephone Order**
**Merchandise Return**
**Cash Advance**
**Card Verification**
**Balance Inquiry**
**Purchase With Cash Back**
**Check Verification**
**Check Guarantee**
**Purchase Adjustment**
**Merchandise Return Adjustment**
**Cash Advance Adjustment**
**Close Batch**
**Close Shift**
**Close Day**
**Reserved**
**Read Mail**

**Send Mail**
**Mail Delivered**
**Sales Drafts**
**Clerk Totals**
**Cash Back Adjustment**
**Adjustments when Amt2 > Amt1**
**Preauthorizations for a lesser amount**
**Card Activation**
**Additional Card Activation**
**Replenishment**
**Full Redemption**

**RETAILER-ID —** DID q contains the retailer identifier of the retailer who owns this terminal, as defined in the RETAILER ID field on PTD screen 1.

**MERCHANT NAME —** DID r contains the merchant name, as defined in the RETAILER NAME field on POS Retailer Definition File (PRDF) screen 1.

**REFERRAL PHONE # —** DID s contains the retailer referral telephone number, as defined in the REFERRRAL PHONE field on PRDF screen 1.

**RESERVED —** DID t is reserved for future use.

**RESERVED —** DID u is reserved for future use.

**RESERVED —** DID v is reserved for future use.

**RESERVED —** DID w is reserved for future use.

**RESERVED —** DID x is reserved for future use.

**RESERVED —** DID y is reserved for future use.

**RESERVED —** DID z is reserved for future use.

# Requesting a Download

When download data is required, the SPDH module sets the PROCESSING FLAG 2 in the standard message header. This indicates to the terminal that a download should be requested. The terminal should then programmatically request a full download.

In addition, an operator can force the SPDH module to set the PROCESSING FLAG 2 in the standard header through the network control facility LOADFLAGON command. The format of this text command is contained in section 3. The LOADFLAGON command instructs the SPDH module to set the PROCESSING FLAG 2 in the standard message header in responses to the terminal to indicate a download is required. Upon receipt of the command with the LOADFLAGON parameter, the SPDH module sets the appropriate flag within the PTD device dependent data area. When the next response is being formatted to the device, the SPDH module sets the PROCESSING FLAG 2 in the standard header to indicate that a download is waiting. For more information on text commands sent using the network control facility, refer to the ***BASE24 Text Command Reference Manual***.

For a full download, the configuration data specified in the 12 ACNF records is sent to the terminal. Responses to a full download request have as much of the 12 records included as fits in a response. The maximum response length is configured in the ACI Standard Configuration File (ACNF).

The terminal can also make a request to the SPDH module for a partial download (i.e., a specific DID). A partial download consists of sending one item to the terminal.

Detailed request and response message requirements for full and partial downloads, including message flow examples, are provided in the ***ACI Standard POS Device Message Specifications Manual***.

*ACI Worldwide, Inc.*

# 6: SPDH Module Configuration Considerations

Besides setting up files, customers are responsible for making decisions concerning several additional configuration issues. These issues determine the way the SPDH module processes transactions. This section discusses the following configuration decisions the customer must make when using the SPDH module:

- Configurable receipts
- Returning account balances
- Chargebacks for preauthorization completions
- Draft capture
- Message sequencing
- Transaction accumulation totals
- Europay, MasterCard, and Visa (EMV) Transaction Certificates
- PIN encryption
- Data encryption
- Message authentication codes (MACs)
- American Express card security codes (CSCs)
- Dynamic key management
- Handshaking
- BASE24-mail support
- Interac Online Payment transaction identification requirements
- Suppression of consumer transaction data

# Configurable Receipts

The SPDH module does not format receipts. Instead, it is the customer's responsibility to configure each response in the terminal configuration data to include sufficient data for the terminal to format and print a receipt.

However, the SPDH module has the capability of returning a response determined by the customer in up to three different languages as well as returning a 48-character response. This is set up in the ACI Standard Device Response File (ARSP) or the Response Code Description File (RCDF).

This subsection provides information on the following topics concerning receipts:

- Standard receipt information
- SPDH module response considerations
- Optional data field information

## Receipt Information

Although it is the customer's responsibility to determine the information to be included on receipts, several suggestions follow.

### Standard Message Header Fields

The standard header fields typically printed on a receipt include the following:

- Terminal ID
- Employee ID
- Date
- Time
- Transaction code
- Response code

## Optional Data Fields

The optional data fields typically printed on a receipt include the following:

| FID | FID Name |
|---|---|
| B | Amount 1 |
| D | Application Account Type |
| E | Application Account Number |
| F | Approval Code |
| J | Available Balance |
| K | Business Date |
| h | Sequence Number |
| k | Terminal Location |
| q* | Track 2/Customer |
| r* | Track 2/Supervisor |
| s | Transaction Description |
| 2* | Track 1/Customer |
| 3* | Track 1/Supervisor |

* Either Track 2 or Track 1 data must be specified in the ACI Standard Device Configuration File (ACNF) to print the primary account number (PAN).

### Mobile Top-Up Receipts

In addition to the above optional data fields, the following optional data fields are typically printed on a mobile top-up transaction receipt.

| FID | FID Name |
|---|---|
| f | Marketing message |
| 7 | SubFID a– Mobile Top-Up Track 2<br>SubFID c–Mobile Top-Up Response (includes reference number or activation code, mobile operator name, top-up approval code, and other balance.) |

## Response Considerations

Although the customer and the terminal vendor are responsible for determining what the receipt looks like, several aspects of the receipts must be set up through BASE24 software. These include setting parameters so the SPDH module knows which response to return to the terminal and developing the exact text of the responses to be sent.

### Language Code

The SPDH module uses the request first to determine the language in which to return responses. Specifically, the SPDH module checks for the presence of the language code field identified with FID U. If present, the SPDH module checks the value in this field. If this field is not included in the request, the SPDH module uses the default language code set in the LANGUAGE ID field on POS Terminal Data files (PTD) screen 1.

Therefore, the customer is required to set a default language code in the PTD. If a specific language code is required for certain transactions or for any other reason, the customer needs to include FID U in the request message. For consistency, the ACI Standard Device Configuration File (ACNF) should also be set appropriately, indicating when FID U is going to be included in requests.

### Language Index and Responses

The SPDH module uses the language code, along with the BASE24-pos response code returned from the Authorization module, to determine the response to send to the terminal.

When the SPDH module receives the response code from the Authorization module, the SPDH module accesses the ACI Standard Device Response File (ARSP) using the terminal group identified in the PTD or the default terminal group (****) and record 00. An index in record 00 of the ARSP links all BASE24-pos response codes to a number. The SPDH module then uses this number as an offset into the appropriate language table to determine the correct response. The language tables are located in records 01, 02, and 03 of the ARSP.

Customers must enter the appropriate responses in the language tables and the correct numbers in the index in the ARSP for the response to be accurate. Instructions on entering this information are included in section 8.

## Optional Responses

If the ACNF is set to include FID g in responses to the terminal, the SPDH module returns a customer-configured response to the terminal. Although these responses are typically used to inform the terminal operator of processing results, the response text is optional and up to the discretion of the customer. The only restriction on these responses is that they not exceed 48 characters. If the response is less than 48 characters, trailing spaces are removed by the SPDH module.

These customer-configured responses are based on transaction type. For example, different responses can be made for each transaction type. These responses can contain variable data fields using the ARSP or RCDF. Up to the maximum field length, one or more of the following optional or download data fields can be included in any display.

| ARSP or RCDF Entry | FID | FID Name |
|---|---|---|
| \B | B | Amount 1 |
| \C | C | Amount 2 |
| \E | E | Application Account Number |

| | | |
|---|---|---|
| \F | F | Approval Code |
| \J | J | Available Balance |
| \K | K | Business Date |
| \N | N | Customer ID |
| \Q | Q | Echo Data |
| \S | S | Invoice Number |
| \T | T | Invoice Number/Original |
| \Z | Z | Address verification status code |
| \d | d | Retailer ID |
| \h | h | Sequence Number |
| \k | k | Birth Date/Drivers License/Terminal Location |

Responses are configured in the response code description fields on the ARSP Language Display screens and in the DESCRIPTION field on RCDF screen 2. The ARSP screens and field descriptions are in section 8. The RCDF screens and field descriptions are in the ***BASE24-pos Files Maintenance Manual.***

The ARSP contains BASE24 response codes, while the RCDF can contain BASE24 and ISO response codes. Descriptions for ISO response codes must be defined in the RCDF when responses include FID X (ISO Response Code). Descriptions for BASE24 response codes must be defined in the ARSP. However, if a BASE24 response code is defined in the ARSP and RCDF, the RCDF description overrides the ARSP description.

# Returning Account Balances

The SPDH module returns account balances on balance inquiry transactions. In addition, the SPDH module can be configured to return balances on purchase transactions and other transactions. The value in the BALANCE RETURNED field on PTD screen 2 controls whether account balances are returned on transactions other than balance inquiries. The value in the RETURN BALANCES field on CPF screen 7 controls whether account balances are returned on purchase transactions other than balance inquiries. Purchase transactions include purchase, purchase with cash back, and preauthorization purchase. Valid values for the RETURN BALANCES field on CPF screen 7 are as follows:

0 = Balances are not returned
1 = Balances are returned, regardless of the PTD setting
2 = Balances are returned only if the PTD setting also specifies that balances are returned

The value in the RETURN BALANCES field on CPF screen 7 cannot override the value in the BALANCE RETURNED field on PTD screen 2, as shown in the following table.

| PTD Value | CPF Value | Processing Performed for Purchase Transactions |
|---|---|---|
| N | 0 | Balances are not returned |
| N | 1 | Balances are returned in the POS Balances token |
| N | 2 | Balances are not returned because the CPF value cannot override the PTD value |
| Y | 0 | Balances are returned in the POS Balances token |
| Y | 1 | Balances are returned in the POS Balances token |
| Y | 2 | Balances are returned in the POS Balances token |

The SPDH module returns the available balance in FID J in response messages. When configured to return balances only for account balance inquiries, the Multiple Currency add-on determines where the SPDH module obtains the available balance it returns for balance inquiry transactions.

- With the Multiple Currency add-on, the SPDH can obtain the available balance from the POS Balances token if the transaction involves multiple currencies or the TRAN.AMT-1 field in the PSTM if the transaction does not involve multiple currencies.
- Without the Multiple Currency add-on, the SPDH obtains the available balance only from the TRAN.AMT-1 field in the PSTM.

When configured to return account balances for purchase transactions or all transactions, the SPDH module obtains the available balance only from the POS Balances token.

Refer to the ***BASE24 Tokens Manual*** for information about the POS Balances token.

The SPDH module can encrypt the account balance. Refer to the Data Encryption discussion in this section for more information.

The SPDH module uses the REVERSE-BAL-INQ param in the Logical Network Configuration File (LCONF) to determine whether balance inquiry transactions should be reversed if they do not complete as intended. While balance inquiry transactions typically are not reversed, it may be appropriate to do so when a service charge is applied for an inquiry. The REVERSE-BAL-INQ param applies only to the balance inquiry transaction, not the return of balances on other transactions.

# Chargebacks for Preauthorized Hold Completions

A preauthorized hold transaction requests that funds be placed on hold against a future purchase or payment. These funds are not available to the cardholder until the preauthorized hold request expires or the BASE24 system receives a preauthorized completion transaction indicating that the purchase or payment for which funds were being held was completed as authorized. The BASE24-pos Standard POS Device Handler module supports standard and enhanced preauthorized holds processing.

The BASE24-pos product cannot reject a preauthorized completion transaction. However, the BASE24-pos product can be configured to generate a chargeback (0402 message) following a preauthorization completion transaction that does not meet the approval requirements.

- In standard preauthorized holds processing, a chargeback can be generated if the hold has expired.
- In enhanced preauthorized holds processing, a chargeback can be generated if any of the following processing requirements are not met.
  - The preauthorization completion amount must not exceed the preauthorization hold amount.
  - The preauthorization hold must not have expired.
  - The preauthorization hold completion timer must not have expired.

See the ***BASE24-pos Transaction Processing Manual*** for specific configuration information.

# Draft Capture

BASE24-pos allows for processing retailer transactions as draft capture or non-draft capture. Non-draft capture transactions are transactions for which the paper draft represents the actual settlement instrument. Paper drafts must be physically presented for deposit in order for the transaction to be settled. While the transaction may have been authorized by BASE24-pos, the BASE24-pos record cannot be used to settle the item.

Draft capture transactions are electronically captured at the time of the transaction and are sufficient in themselves to enact a funds transfer for the transaction amount. BASE24-pos identifies draft capture transactions in its files and includes draft capture totals on its retailer reports so that the retailer sponsor can settle with the retailer.

Draft capture is a settlement tool only. It has no impact on the authorization of transactions. Cardholders are still checked against the same authorization criteria without regard for whether a transaction is draft capture or non-draft capture. Thus, the retailer is not incurring additional risk of cardholder fraud by using or not using draft capture.

## Draft Capture Options

Customers using the BASE24 Standard POS Device Handler (SPDH) module have three draft capture options available to them. These are as follows:

- Authorization only with paper draft follow-up
- Authorize and draft capture
- Terminal-defined draft capture

Different draft capture options may be selected for each terminal and card type. Customers set the TP field on PTD screens 8, 9, and 10 to indicate the draft capture option they have selected for each terminal and card type. The values that can be placed in the PTD TP (transaction profile) field for BASE24-pos Standard POS terminals are as follows:

0 = Authorize only with paper draft follow-up (first-generation terminal)
1 = Authorize and draft capture (third-generation terminal)
3 = Terminal determines draft capture profile

For request or force-post transactions (PSTM message types 0200 and 0220, respectively) from an SPDH terminal, the transaction profile from the terminal can be 0 or 1. For a reversal transaction (PSTM message type 0420) from an SPDH terminal, the transaction profile must be the same as that of the original transaction.

Although the value of 2 (authorize and expect electronic follow-up) is generally acceptable in this field, the SPDH module does not support the value. The value in the PTD is used by the SPDH module, unless the value in the PTD is a 3, indicating the terminal is to determine the draft capture value.

If the terminal is to determine the draft capture value, the SPDH module uses the value sent in FID P (Draft Capture Flag) of the request. FID P contains the following values for the SPDH module:

0 = Authorize only with paper follow-up
1 = Authorize and draft capture

## Authorization Only With Paper Follow-up

The authorization only with paper follow-up method of draft capture obtains online authorization for the amount of the customer purchase or other type of data, depending on the transaction. In an authorization only environment, the card issuer waits for paper follow-up before posting the transaction and settling with the retailer. If a card type is set up to use this option, none of the transactions for the card type from the specified terminal are draft capture.

With this method, BASE24 updates the cardholder usage and activity totals, logs transactions to the PTLF, and updates the service totals and authorization totals in the PTD.

The host waits for paper to post and settle with retailer and on-us cardholder accounts.

The value in the PTD, or the value in FID P, must be set to 0 or 3 with the terminal sending 0, indicating to authorize with paper follow-up.

## Authorization and Draft Capture

The authorization and draft capture option is a one-step process that allows settlement to occur without paper follow-up. This option performs an authorization on the transaction and at the same time automatically captures the draft electronically. Using electronic draft capture, no paper follow-up is required by the card issuer to post the transaction and settle with the retailer. The procedure is completed in one transaction.

With this method, BASE24-pos updates the cardholder usage and activity totals, logs transaction to the PTLF, and updates service totals, authorization totals, and draft capture totals on the PTD.

The host posts and settles with the retailer and on-us cardholder accounts without paper follow-up.

The value in the PTD, or the FID P (Draft Capture Flag), must be set to 1 or 3 with the terminal sending 1, indicating to use the authorization and draft capture method.

## Terminal-Defined Draft Capture

The terminal-defined draft capture option allows the terminal to determine the draft capture mode for each transaction based on card type. If a card type is set up with this option, the terminal selects one of the draft capture options described previously (i.e., authorization only with paper follow-up or authorization and draft capture) for each transaction involving the card type. The SPDH module uses the value sent in FID P (Draft Capture Flag) of the request, as described previously.

## Setting the Draft Capture Flag in the PSTM

The draft capture flag indicates whether BASE24-pos balances should immediately reflect transaction activity. The SPDH module sets this draft capture flag using the SRV.TRAN-PROFILE field in the PSTM or the transaction profile value set by the terminal (using FID P). The Router/Authorization module then sets the DFT-CAPTURE-FLG field in the PSTM. The valid values for the DFT-CAPTURE-FLG field in the PSTM for the SPDH module are as follows:

0 = Authorize only with paper follow-up
1 = Authorize and draft capture

The SPDH module uses the TP field on PTD screens 8, 9, and 10 or the transaction profile sent by the terminal to set the TRAN-PROFILE field in the PSTM. The Router module then moves the value in the TRAN-PROFILE field of the PSTM to the DFT-CAPTURE-FLG field in the PSTM. The transaction profile values and the corresponding draft capture flag settings for the SPDH module are shown in the following table.

| TRAN-PROFILE | DFT-CAPTURE-FLG | TRANSACTION TYPE |
|---|---|---|
| 0 | 0 | All transactions |
| 1 | 0 | Balance inquiry, check guarantee, check verification, card verification, and preauthorization purchase transactions |
| | 1 | Adjustment, cash advance, mail/ phone order, normal purchase, purchase with cash back, preauthorization purchase completion, and refund transactions |
| 2 | 1 | 2 is an invalid TRAN-PROFILE value for the SPDH module. The DFT-CAPTURE-FLG defaults to 1. |
| 3 | Value sent from the terminal | All transactions |

# Message Sequencing

A terminal that conforms to the SPDH module can request the SPDH module to validate transmission numbers and sequence numbers by including these in the requests. Transmission number checking is only intended to avoid duplicates. Sequence number checking is intended to ensure that BASE24-pos has received and processed every transaction only once. Any combination of these checking techniques can be implemented.

## Transmission Number Checking

Transmission numbers optionally occur in the Transmission Number field in the standard message header. If no transmission number checking is required by the customer, the Transmission Number field in the standard message header must be set to zeros. If the customer wants to check transmission numbers, the Transmission Number field in the standard message header must be set to a nonzero value. Any nonzero values in the Transmission Number field are checked by the SPDH module. Although the use of the transmission number field is optional, ACI recommends that most POS terminals use it, especially when terminals have offline authorization capabilities and send transactions in a store-and-forward mode.

For online transactions (the Message Subtype field in the standard message header set to O), it is theoretically sufficient for the terminal to supply only the sequence number on requests as a means of controlling duplicate transmissions. When the SPDH module receives an online request with a sequence number that it is not expecting, it can execute the resynchronization procedure described later in this section, thereby avoiding duplicate processing.

If the transmission number is not included as recommended by ACI, ACI cannot guarantee adequate protection against posting transactions twice in situations where the terminal times out or when processing force-post transactions.

ACI requires transmission numbers to be used in the manner specified below.

- The terminal must assign a new transmission number to each new transaction. The SPDH module expects the terminal to increment the number from 1 to 99 and then roll back to 1 again.
- When a new PTD record is initialized, the transmission number is set to 00, so the terminal can be configured to send 01 as an initial value.

- When transactions are retransmitted as force-post transactions due to a terminal timing out, the transmission number that accompanied the original transaction is used on the force post transaction.
- When the SPDH module receives a message having the same transmission number as the last message received by the SPDH module, and the last message received by the SPDH module was approved, the message is declined with a response code of 078, indicating the transaction is a duplicate. When the SPDH module receives a message having the same transmission number as the last message received by the SPDH module, and the last message received by the SPDH module was declined, the message is processed.

## Store-and-Forward Considerations

For store-and-forward transactions (the Message Subtype field in the standard message header set to S) no message resynchronization processing is done. The transaction is processed without checking its sequence number.

With store-and-forward transactions, the possibility exists for the terminal to transmit the transaction, but not receive a response from the SPDH module. The terminal has no way of determining whether the SPDH module actually received the transaction and has no alternative but to retransmit until a valid response is received.

Because the SPDH module does not support sequence number resynchronization for store-and-forward transactions, it processes any retransmission as a new transaction unless there is some way to determine that the transaction has already been processed.

The only way to do this check on store-and-forward transactions is through the use of transmission numbers. The terminal should assign the transmission number to a transaction when it is sent for the first time. Only after a valid response for the transaction is received from the SPDH module should the terminal increment the transmission number for use on the next transaction. This way a terminal can retransmit repeatedly knowing that the SPDH module only processes the transaction once, regardless of communication line failures.

### Force-post Considerations

The SPDH module offers an alternate solution to the duplicate detection problem. Transactions that are offline-authorized by the terminal can be sent as force-post transactions (the Message subtype field in the standard message header set to F). In this case, sequence number checking logic is applied, exactly as with online transactions, before force posting the transaction.

This alternate solution is less efficient than the transmission number solution because it compromises the distinction that the SPDH module draws between force-post and store-and-forward transactions.

A force-post transaction is technically a transaction for which prior authorization was received from the host. For example, a preauthorization purchase completion transaction that received prior authorization with a preauthorization purchase transaction. Another example of a force-post transaction is a transaction that received authorization from a CRT authorization center.

A store-and-forward transaction is one for which the terminal is standing-in without communicating with an outside authorizer. In some cases, if offline-authorized transactions are sent to the host as force-posts, BASE24-pos reporting is not able to identify them as authorized by a stand-in POS terminal.

## Sequence Number Checking

Sequence number checking is intended to ensure that BASE24-pos has received and processed each transaction only once. When the SPDH module encounters FID h (Sequence Number) in a request, the SPDH module verifies that the sequence number in this field is the one the SPDH module is expecting. The SPDH module checks the sequence number sent in the request with the sequence number stored in the PTD.

Sequence number checking is done by the SPDH module only when FID h is included in the request from the terminal. This field consists of 10 numeric characters. The first three characters indicate the shift number and range from 001 to 999, rolling back to 001. The next three characters identify the current batch and range from 001 to 999, rolling back to 001. The next three characters consist of a unique sequence number within the shift and batch which ranges from 001 to 999, rolling back to 001. The last character is a reset flag, indicating whether the terminal or the SPDH module is responsible for determining the correct sequence number.

The SPDH module is initialized to expect sequence number 001001001, with the shift number set to 001, the batch number set to 001, and the sequence number set to 001. From this point on, the SPDH module expects the sequence number to increment by one. When 999 is reached, the SPDH module expects 001 as the next sequence number.

## Batch Close Transactions

When the first batch close transaction is performed, the SPDH module next expects a sequence number of 001002001. Since the batch close transaction is optional, the SPDH module is conditioned to always allow the terminal to send a sequence number with the sequence number set to 001 without incrementing the batch number to 002. When this occurs the SPDH module can perform an implicit batch close.

## Shift Close Transactions

When the first shift close transaction is performed, the SPDH module next expects a sequence number of 002001001. Since the shift close transaction is optional, the SPDH module is conditioned to always allow the terminal to send a sequence number with the batch number set to 001 without incrementing the shift number to 002. When this occurs the SPDH module can perform an implicit shift close transaction. An implicit batch close transaction may also be in order at this time.

## Close Day Transactions

When the first day close transaction is performed, the SPDH module next expects a sequence number of 001001001. Since the day close transaction is optional, the SPDH module is conditioned to always allow the terminal to send a sequence number with the shift number set to 001 without resetting the shift, batch, and sequence number to 001. When this occurs the SPDH module can perform an implicit day close transaction. An implicit shift close transaction and an implicit batch close transaction may also be in order at this time.

## Implied Closes from the Terminal

Implied closes are processed only if the terminal group is configured to support implied closes in the ACNF. If the terminal group is not configured to support implied closes, the SPDH module does not perform closes and the BASE24 Report

programs do not execute correctly. In addition, implied closes are completed only if the reset flag in the sequence number is set to 1, indicating that the terminal is to determine the sequence number.

## Unexpected Sequence Number, Batch Number, or Shift Number

When the SPDH module receives an unexpected sequence number, batch number, or shift number and the reset flag in the sequence number is set to 0, indicating the SPDH module is to determine the sequence number, the SPDH module responds to the terminal with a denial response code of 899 and FID h. Response code 899 indicates a sequence error has occurred and resynchronization is to be performed. FID h contains the expected sequence number.

The terminal has two options when it receives a response code of 899 with FID h. It can adjust its sequence number, batch number, or shift number to correspond with the sequence number, batch number, or shift number received from the SPDH module. This implies the terminal is able to adjust its sequence numbers, batch numbers, and shift numbers backward or forward in its internal queue and send subsequent transactions that correspond to the adjusted sequence number, batch number, or shift number.

The terminal can also direct the SPDH module to adjust its sequence number, batch number, or shift number to match the sequence number, batch number, or shift number maintained by the terminal. To do this, the terminal needs to put the sequence number, batch number, and shift number that the terminal determines to be correct in FID h, set the reset flag to 1, and send the transaction that originally received a response code of 899 back to the SPDH module. Once the transaction is resubmitted to the SPDH module, the SPDH module accepts the transaction.

In addition, if the batch or shift number has been changed from the batch or shift number in the original transaction, the SPDH module determines that the transaction being resubmitted contains a new batch or shift number and the SPDH module performs an implied batch close or an implied shift close. The SPDH module then processes the transaction and responds to the terminal.

## Resetting the Sequence Number

An LCONF param, POS-DH-SEQNUM-RESET-OPT, determines how the SPDH module resets the sequence number when a Day Close transaction is performed. Valid values for this param are as follows:

0 = Use standard reset procedures (reset, shift, batch, and seq # values)
1 = Reset shift and seq # values, but increment batch value
2 = Reset batch and seq # values, but increment shift value
3 = Reset seq # value, but increment shift and batch values
9 = Increment shift, batch, and seq # values

If this param is missing or invalid, the SPDH module uses standard sequence number reset procedures.

The standard sequence number reset procedure is resetting shift, batch, and sequence number (seq #) values to 001 when a Day Close transaction is performed. Under most operating conditions, this procedure is sufficient to provide unique sequence numbers for all transactions. However, in certain environments where a cardholder does not use ATMs and uses the same POS terminal on a regular schedule, the possibility exists for transactions performed on different days to have the same sequence number. In this situation, the SPDH module could decline a transaction because the transaction sequence numbers are the same, even though the transactions are unique because they are performed on different days.

The SPDH module can be configured to increment various shift, batch, and sequence number (seq #) combinations as a way to eliminate such duplicate sequence number situations from occurring, as shown in the following table. Values that differ from the standard reset are ***highlighted in red italics.***

| Transaction | Standard Reset | Reset Shift and Seq #, Increment Batch | Reset Batch and Seq #, Increment Shift | Reset Seq #, Increment Shift and Batch | Increment Shift, Batch, and Seq # |
|---|---|---|---|---|---|
| Tran #1 of day | 001001001 | 001001001 | 001001001 | 001001001 | 001001001 |
| Tran #2 of day | 001001002 | 001001002 | 001001002 | 001001002 | 001001002 |
| Tran #3 of day | 001001003 | 001001003 | 001001003 | 001001003 | 001001003 |
| Batch close | 001002000 | 001002000 | 001002000 | 001002000 | 00100200***3*** |

| Transaction | Standard Reset | Reset Shift and Seq #, Increment Batch | Reset Batch and Seq #, Increment Shift | Reset Seq #, Increment Shift and Batch | Increment Shift, Batch, and Seq # |
|---|---|---|---|---|---|
| Tran #4 of day | 001002001 | 001002001 | 001002001 | 001002001 | 00100200***4*** |
| Tran #5 of day | 001002002 | 001002002 | 001002002 | 001002002 | 00100200***5*** |
| Tran #6 of day | 001002003 | 001002003 | 001002003 | 001002003 | 00100200***6*** |
| Batch close | 001003000 | 001003000 | 001003000 | 001003000 | 00100300***6*** |
| Tran #7 of day | 001003001 | 001003001 | 001003001 | 001003001 | 00100300***7*** |
| Batch close | 001004000 | 001004000 | 001004000 | 001004000 | 00100400***7*** |
| Shift close | 002001000 | 00200***4***000 | 002001000 | 00200***4***000 | 00200***4***00***7*** |
| Tran #8 of day | 002001001 | 00200***4***001 | 002001001 | 00200***4***001 | 00200***4***00***8*** |
| Tran #9 of day | 002001002 | 00200***4***002 | 002001002 | 00200***4***002 | 00200***4***00***9*** |
| Batch close | 002002000 | 00200***5***000 | 002002000 | 00200***5***000 | 00200***5***00***9*** |
| Shift close | 003001000 | 00300***5***000 | 003001000 | 00300***5***000 | 00300***5***00***9*** |
| Day close | 001001000 | 00100***5***000 | 00***3***001000 | 00***3***00***5***000 | 00***3***00***5***00***9*** |
| Tran #1 of next day | 001001001 | 00100***5***001 | 00***3***001001 | 00***3***00***5***001 | 00***3***00***5***0***10*** |

## Sequence Number Checking Examples

Refer to the ***ACI Standard POS Device Message Specifications Manual*** for detailed sequence number checking message flow examples.

# Transaction Accumulation Totals

The customer can choose to have the terminal accumulate any set of batch, shift, and day totals and forward them to the SPDH module with the corresponding close request. If the totals differ from SPDH module totals, they are recorded in the POS Transaction Log File (PTLF). No attempt is made to indicate or reconcile any differences.

The SPDH module always accumulates batch, shift, and day totals. When an explicit or implicit close for the corresponding period is received, the SPDH module records these totals in the PTLF and resets the totals as appropriate. The SPDH module also supports reconciliation with a series of subtotals request transactions prior to an explicit or implicit close.

The BALANCE SUPPORT field on PTD screen 3 indicates to the SPDH module which totals are supported by the terminal by means of close transactions. Totals specified in the PTD are allowed to accumulate until reset by means of a close transaction.

The SPDH module records clerk totals as required by the terminal. It logs these totals to the PTLF. When a clerk totals request is received from the terminal, the SPDH module accumulates the clerk totals by employee ID or terminal ID and returns them to the terminal.

## Grouping Terminals at a Site for Configuration and Balancing

The SPDH module maintains totals for each terminal at a merchant site. Additionally, the SPDH module can maintain a set of totals that combines all of the terminals at a merchant site. Terminals in a department store or a pay-at-the-pump service station are examples of site configurations.

A site configuration uses one PTD record for each terminal plus one PTD record for the site itself. The value in the SITE ID field on PTD screen 1 identifies the site. The site-level PTD records are called primary records, and the terminal-level PTD records are called secondary records. Each site can have only one primary PTD record.

Once all primary and secondary PTD records have been defined, modifications made to certain fields of the primary record can be applied to all secondary records with the same value in the SITE ID field on PTD screen 1 by pressing a single function key. Secondary PTD fields that can be updated from the primary PTD

screen include all user-modifiable PTDS1 fields other than TERMINAL ID and TERMINAL TYPE. The data name associated with each PTD field in the ***BASE24-pos Files Maintenance Manual*** identifies whether it is a PTDS1 field.

**Note:** User-modifiable PTDS1 fields of all secondary PTD records associated with the primary PTD record are overwritten by this update function. Any changes made to user-modifiable PTDS1 fields of individual secondary PTD records will have to be reapplied whenever this update function is used.

A site configuration accumulates the following transaction totals for the individual terminals at the site and displays the aggregate totals in the primary PTD record.

- Service totals (also called totals by card type) displayed on PTD screens 11 through 16
- Batch totals displayed on PTD screen 4
- Shift totals displayed on PTD screen 4
- Daily totals displayed on PTD screen 4
- Current network totals displayed on PTD screen 4

The following settings are required for the totals on the primary PTD record to be displayed accurately.

- Every card type that is defined on screens 8 through 10 of any secondary PTD record at a site must also be configured on screens 8 through 10 of the primary PTD record for the site. Service totals are displayed on PTD screens 11 through 16 of the primary record only if the card type is defined on screens 8 through 10 of the primary PTD record.
- The TERMINAL CUTOVER field on screen 3 of the primary PTD record must be set to a value of 3 (network forced-cutover by the Settlement Initiator process). The Settlement Initiator process clears the site aggregate transaction totals at the end of the retailer cutover window.
- The TERMINAL CUTOVER field on screen 3 of each of the secondary PTD records must be set to a value of 3 (network forced-cutover by the Settlement Initiator process). This ensures that terminal totals are cleared at the same time as the site totals.

# PIN Encryption

Customers are not required to support PIN encryption, but if they do decide to support it, the SPDH module offers several methods of PIN encryption. It is up to the customer to determine which method best suits their particular requirements.

Depending on how the terminal is configured in the BASE24-pos Terminal Data files (PTD), FID b and FID c can be encrypted using the following methods:

- Single-length key (16 bytes), single DES performed in software
- Single-length key (16 bytes), single DES performed in hardware
- Double-length key (32 bytes), triple DES performed in hardware
- Triple-length key (48 bytes), triple DES performed in hardware

The SPDH module can perform PIN verification with single-length keys. The Transaction Security Services process is optional for the single DES hardware option and is required for the triple DES options. The terminal and the appropriate BASE24 process (SPDH module or Transaction Security Services process) must be configured for the same key length. Refer to the ***BASE24 Transaction Security Services Processing Guide*** for information about the Transaction Security Services process.

If PIN encryption is selected, the terminal must support the manual entry or injection of a master key. Master/session keys are used with master/session key management. Base derivation keys are used when the POS devices derive a unique key per transaction (DUKPT). For more information on the PIN fields contained for terminals in the PTD, refer to the ***BASE24-pos Files Maintenance Manual***.

## Master/Session Key Management

When using master/session key management, the SPDH module uses data in the ACNF to determine when to provide the terminal with a new PIN communications (session) key. The SPDH module generates a new PIN communications key, a working key that is also known as the KPE, under the following conditions:

- When a response message is configured to contain FID M (PIN Communications Key). This is set in the Field Map Data record in the ACI Standard Device Configuration File (ACNF).
- When a dynamic key management threshold is reached. Refer to the Dynamic Key Management discussion in this section for more information.

- During a download if download field identifier (DID) g (PIN Communications Key) is to be sent. In order to include the communications key in a download, record 06 in the ACNF must have DID g set.

Whenever a new PIN communications key is sent to the terminal in a response, it is encrypted under the terminal master key. Once a new key is generated, the SPDH module updates the PTD or the Transaction Security Services process updates the Transaction Security Services database. For more information about the Transaction Security Services process and database, refer to the ***BASE24 Transaction Security Services Processing Guide.***

If the SPDH module is required to perform PIN decryption and the communications key is all spaces, the SPDH module assumes the terminal has never received a communications key from the SPDH module and rejects the PIN as invalid.

# PIN Encryption

Customers are not required to support PIN encryption, but if they do decide to support it, the host offers several methods of PIN encryption. It is up to the customer to determine which method best suits their particular requirements.

Depending on how the terminal is configured in the host database, the PIN/Customer field (FID b) and the PIN/Supervisor field (FID c) can be encrypted using the following methods:

- Single-length key (16 bytes), single DES performed in software
- Single-length key (16 bytes), single DES performed in hardware
- Double-length key (32 bytes), triple DES performed in hardware
- Triple-length key (48 bytes), triple DES performed in hardware

If PIN encryption is selected, the terminal must support the manual entry or injection of a master key or base derivation key. Master/session keys are used with master/session key management. Base derivation keys are used when the POS devices derive a unique key per transaction (DUKPT).

## Master/Session Key Management

When using master/session key management, the host uses data in the host database to determine when to provide the terminal with a new communications (session) key. The host generates a new communications key, which is also known as the KPE, under the following conditions:

- When a response message is configured to contain FID M (Communications Key).
- When a dynamic key management threshold is reached. Refer to the Dynamic Key Management discussion in this section for more information.
- During a download if download field identifier (DID) g (Communications Key) is to be sent.

If the host is required to perform PIN decryption and the communications key is all spaces, the host assumes the terminal has never received a communications key from the host and rejects the PIN as invalid.

# Europay, MasterCard and Visa (EMV) Transaction Certificates

When an EMV transaction is authorized offline by the terminal or EMV card, the card generates a Transaction Certificate (TC), which is a card-generated cryptogram containing information about the transaction that can be used if the transaction is disputed.

The SPDH Device Handler module accepts TCs uploaded, one at a time, from the terminal. Each TC is sent to the device handler module as an EMV log-only transaction identified by the message subtype E in the message header. The TC is carried in the Application Cryptogram (AC) field in FID 6 subFID O (EMV Request Data).

If the terminal does not receive a response to an EMV log-only transaction request prior to timing out, the terminal should immediately send an EMV log-only cancellation request (message subtype V). From the perspective of the terminal, an EMV log-only cancellation should be handled similarly to a timeout reversal (message subtype T).

- The EMV log-only cancellation must immediately follow the original EMV log-only transaction.
- The EMV log-only cancellation must take precedence in the queue over online requests and SAF requests.
- The transmission number of the EMV log-only cancellation request must be set to match that of the original EMV log-only transaction. The SPDH uses the transmission number to match the EMV log-only cancellation with the original EMV log-only transaction.

## EMV Log-Only Transaction Processing

Upon receipt of an EMV log-only transaction, the SPDH performs as follows:

- Creates an internal message (message type 9920) and passes it to the BASE24-pos Router/Authorization module for logging.
- Updates the terminal and clerk totals in the BASE24-pos Terminal Data files (PTD) to enable the EMV terminal to balance during settlement.
- Formats and sends an appropriate response to the terminal.

Within BASE24-pos, cardholder balances are not impacted by EMV log-only transactions; however, the transactions are logged to the PTLF with a message type of 9920. Note that the PTLF Extract module can extract PTLF records with the 9920 message type and that EMV log-only transactions with message type 9920 are not used by the Refresh module when it performs transaction impacting.

The SPDH creates the internal message for EMV log-only transactions as follows:

| Message Field | Description |
|---|---|
| Message Type | Set to 9920 (EMV log-only transaction). |
| Response Code | Set to 000. |
| Responder | Set to 1 (device controlled by the BASE24-pos system). |
| Draft Capture Flag | Set to 3 (electronic follow-up to settle a previously authorized transaction).<br>This causes the Router/Authorization module to set the impact indicator in the PTLF to 0, which in turn prevents the BASE24-pos Refresh module from using the EMV log-only record for impacting purposes. |

## EMV Log-Only Cancellation Transaction Processing

Upon receipt of an EMV log-only cancellation, the SPDH performs as follows:

- Verifies that it is not a duplicate request. If a terminal, for example, does not receive a response to an initial EMV log-only cancellation request and sends a second, the SPDH will verify that the previous request was processed and, if so, return a decline response to the terminal for the duplicate.
- Verifies that it is not out-of-sequence. If the transmission number does not match that of the original EMV log-only transaction, the request is declined.
- Decreases the POS Terminal Data files (PTD) terminal and clerk totals by the transaction amount, effectively cancelling the impacts of the original EMV log-only transaction.

- Creates an internal message (message type 9921) and passes it to the BASE24-pos Router/Authorization module for logging. The message type 9921 allows the Extract module to recognize the transaction as an EMV log-only cancellation.
- Formats and sends an appropriate response to the terminal.

If an EMV log-only cancellation request is declined, the SPDH sets the Response Code in the message header to 050 (general decline).

The SPDH creates the internal message for EMV log-only cancellation transactions as follows:

| Message Field | Description |
|---|---|
| Message Type | Set to 9921 (EMV log-only cancellation transaction). |
| Response Code | Set to 000. |
| Responder | Set to 1 (device controlled by the BASE24-pos system). |
| Draft Capture Flag | Set to 3 (electronic follow-up to settle a previously authorized transaction).<br>This causes the Router/Authorization module to set the impact indicator in the PTLF to 0, which in turn prevents the BASE24-pos Refresh module from using the EMV log-only record for impacting purposes. |

# Data Encryption

The SPDH data encryption module is an additional licensable module that, when bound with the standard SPDH module, supports data encryption for FID J (Available Balance). The SPDH also supports full message encryption and configurable message encryption.

Before encrypted data other than PINs and MACs can be returned to a terminal that conforms to the SPDH module, the data encryption terminal master key must be manually entered or injected into the terminal. The same key also must be entered in the Transaction Security Services database. The other key used with data encryption is the data encryption communications key, a working key that is also known as the message encryption key (KME). The data encryption communications key is encrypted under the data encryption terminal master key before it is downloaded to the terminal.

The SPDH module generates a new data encryption communications key under the following conditions:

- When a response message is configured to contain FID I (Data Encryption Key). This is set in the Field Map Data record in the ACNF.
- When a dynamic key management threshold is reached. Refer to the Dynamic Key Management discussion in this section for more information.
- During a download if download field identifier (DID) j (Data Encryption Key) is to be sent. In order to include the data encryption communications key in a download, record 06 in the ACNF must have DID j set.

The ACI standard POS message supports full message encryption and configurable message encryption. Full message encryption allows the institution to encrypt all optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. Configurable message encryption allows the institution to encrypt specific optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. The optional data fields are configured to be encrypted in the ACI Standard Device Configuration File (ACNF) request and response field maps. Full message encryption and configurable message encryption are enabled using the DATA ENCRYPTION TYPE field in the POS Terminal Data File.

A code for the type of data encryption must be entered in the DATA ENCRYPTION TYPE field on PTD screen 7. Valid values are as follows:

00 = No encryption
01 = Available balance encryption–ASCII

02 = Available balance encryption–Binary
03 = Full message encryption
04 = Configurable message encryption

Data encryption support requires the Transaction Security Services process. Refer to the ***BASE24 Transaction Security Services Processing Guide*** for information about the Transaction Security Services process and database.

# Message Authentication Codes

The SPDH module support of message authentication codes (MACs) uses the standards documented in ***Financial Institution Retail Message Authentication Standard,*** ANSI X9.19 (1986). The ACI MAC, along with the ANSI standard, is designed to protect financial transaction messages against accidental or deliberate alteration. In addition, it protects against the fraudulent insertion of messages.

The customer determines whether to use MACs and, if so, how they are implemented. MAC generation and verification options include the following:

- Single-length key (16 bytes), single DES performed in software
- Single-length key (16 bytes), single DES performed in hardware
- Double-length key (32 bytes), triple DES performed in hardware
- Triple-length key (48 bytes), triple DES performed in hardware

The SPDH module can perform MAC generation and verification with single-length keys. The Transaction Security Services process is optional for the single DES hardware option and is required for the triple DES options. The terminal and the appropriate BASE24 process (SPDH module or Transaction Security Services process) must be configured for the same key length. Refer to the ***BASE24 Transaction Security Services Processing Guide*** for information about this process.

**Note:** The remainder of this subsection is discusses how MACs are used specifically with the SPDH module. For more information on using software and hardware MACs plus specific instructions for entering required information in the BASE24 database, refer to the ***BASE24 Transaction Security Manual*** and the ***BASE24 Transaction Security Services Processing Guide.***

If MACs are used in any form, the following three fields in the standard message header are authenticated in every request:

- Transmission Number
- Terminal ID
- Transaction Code

The same three fields are authenticated in every response using MACs in any form. In addition, the Response Code field in the standard message header is always authenticated in responses using MACs.

Any number of optional fields totaling up to 1000 bytes can be verified using MACs, but BASE24-pos does not support authenticating the entire message. While the data within the fields are verified using MACs, the FIDs identifying the optional data fields are not included in the MAC.

In addition, ACNF Processing Record screen 2 contains the following flags that identify whether one or more of the communications keys are included when MACs are computed.

INCLUDE KMAC IN MAC
INCLUDE KME IN MAC
INCLUDE DPE IN MAC

## Setting Up MACs

The fields to be verified using MACs are set in the ACNF for requests and responses for every transaction type. However, the customer and the vendor must agree on the fields using MACs in order for the messages to be verified.

ACI recommends that several optional data fields be included when MACs are used. These fields, including their field identifier (FID), follow.

B (Amount 1)
C (Amount 2)
F (Approval Code)
b (PIN/Customer)
q (Track 2/Customer)
2 (Track 1/Customer)

Whenever MACs are used at a terminal that conforms to the SPDH module, FID G (Authentication Code) must be included in both the request and the response. The MAC terminal master key must be manually entered or injected into the terminal. The same key also must be entered in the BASE24 database. The other key used with MACs, the MAC communications key (KMAC), is randomly generated by the SPDH module or the Transaction Security Services process.

## Generating a New MAC Communications Key

The SPDH module generates a new MAC communications key under the following conditions:

- When a response message is configured to contain FID H (Authentication Key). This is set in the Field Map Data record in the ACNF.
- When a dynamic key management threshold is reached. Refer to the Dynamic Key Management discussion in this section for more information.
- During a download if download field identifier (DID) h (Authentication Key), is to be sent. In order to include the authentication key in a download, record 06 in the ACNF must have DID h set.

Whenever a new MAC communications key is sent to the terminal in a response, it is encrypted under the MAC terminal master key. Once a new key is generated, the SPDH module updates the PTD.

## Failed MAC Procedure

When the SPDH module receives a request that cannot be verified using MACs, it formats a response message with a response code of 898, indicating an invalid MAC request. The SPDH module also increments the invalid MAC counter in the PTD.

When an invalid MAC request occurs, the POS-LOG-MAC-ERR param in the Logical Network Configuration File (LCONF) controls whether the SPDH module logs a response to the PTLF.

When a terminal is unable to verify a MAC received from the SPDH module in a transaction response with monetary impact, the terminal must generate a reversal. The reversal is identical to the response message, with two exceptions. The response code is set to 989, indicating an invalid MAC response. Also the Message Subtype field in the standard message header contains an R, indicating a reversal. The reversal is not verified using a MAC.

When the SPDH module receives a MAC reversal, the SPDH module sends a warning message to the logging facility. The SPDH module also sends a 0420 reversal message to the transaction authorizer. The MAC reversal must be the next request from the terminal. If any other requests from the terminal are processed before the reversal, the reversal is dropped and a message is logged.

# Derived Unique Key Per Transaction

POS devices attached to the SPDH module can use derived unique key per transaction (DUKPT) key management for PIN communications (session) keys and MAC communications keys. The SPDH module supports DUKPT management of single-length PIN communications keys. The Transaction Security Services process is optional for single-length PIN communications keys, and is required for longer PIN communications keys and all MAC communications keys.

When using DUKPT key management, a unique base derivation key must be manually entered or injected into each POS device. The same base derivation key must be loaded into a database maintained by the host. When a POS device derives a unique key for a transaction, it must include the Key Serial Number (KSN) and Descriptor field (FID 6, subFID T) in the request message sent to the host.

The Derivation Key File (KEYD) is used for ACI Standard POS device terminals that use DUKPT key management. The KEYD can contain derivation keys or key locators. Derivation keys are used by the SPDH module to translate incoming DUKPT-encrypted PIN blocks received from the terminal into a single-length Master/Session key PIN block. Key locators are nonencrypted values used by the Transaction Security Services process to locate hardware-encrypted key information maintained in the Transaction Security Services database.

Values in the PIN ENCRYPTION TYPE and MAC TYPE fields on PTD screen 7 control the use of DUKPT for PIN and MAC communications keys. Refer to the ***BASE24-pos Files Maintenance Manual*** for more information about the PTD.

For more information on DUKPT key management, refer to the ***BASE24 Transaction Security Services Processing Guide.***

# American Express Card Security Codes (CSCs)

The ACI standard POS message contains data enabling the SPDH module to verify American Express card security codes. The three types of card security codes (CSCs) are as follows:

- Three-digit CSC located on the signature panel
- Four-digit CSC located on the front of the card
- Five-digit CSC located on the magnetic stripe

Card-swipe transactions use the CSCs as follows:

- The five-digit CSC is mandatory and is submitted as part of the Track 2 or Track 1 data. If Track 2 or Track 1 is missing, the SPDH module rejects the transaction. The SPDH module cannot check for the CSC value itself because the SPDH module cannot determine the location of the CSC within the track data.
- The four- and three-digit CSCs are optional.
- The four- and three-digit CSCs are mutually exclusive.
- If a transaction contains a five-digit CSC and one of the shorter CSCs, both sets of data are passed to the Router/Authorization module for verification.

Manually entered transactions use the CSCs as follows:

- The five-digit CSC cannot be submitted.
- The four- and three-digit CSCs are optional, meaning some transactions will not have either of these CSCs.
- The four- and three-digit CSCs are mutually exclusive.

The value in the BAD CV ACTION—TRACK DATA COMPLETE field on Card Prefix File (CPF) screen 2 determines how the SPDH module processes a card-swipe transaction when full track data is received and the five-digit CSC is missing or incorrect.

The value in the BAD CV ACTION—TRACK DATA UNCERTAIN field on CPF screen 2 determines how the SPDH module processes a card-swipe transaction when full track data is not received and the five-digit CSC is missing or incorrect.

The value in the BAD CV ACTION—MANUAL ENTRY field on CPF screen 2 determines how the SPDH module processes a manual entry transaction with an incorrect four- or three-digit CSC.

Refer to the ***BASE24 Base Files Maintenance Manual*** for more information about the CPF.

Refer to the ***BASE24 Transaction Security Manual*** for more information about the location of card verification information on Track 2 and Track 1.

# Dynamic Key Management

The SPDH Dynamic Key Management module is an additional licensable module that, when bound with the standard BASE24-pos SPDH Device Handler module, supports the automatic replacement of working keys for a terminal based on one or more thresholds, for the following working keys:

- The PIN communications key (KPE), which is returned in FID M.
- The MAC communications key (KMAC), which is returned in FID H.
- The data encryption communications key (KME), which is returned in FID I.

The SPDH module uses thresholds based on the following values set on ACNF screen 3. If a nonzero value is placed in the ACNF field, the SPDH module generates a new key and sends it to the terminal with the next response whenever a threshold is reached. The new key is included in the response regardless of whether its FID is set in the Field Map Data record in the ACNF.

- The value in the MAC KEY THRESHOLD field identifies the total number of messages that can be authenticated with a MAC communications key.
- The value in the MAC KEY ERROR THRESHOLD field identifies the number of failed message authentication attempts with a MAC communications key.
- The value in the CONSECUTIVE MAC KEY ERROR THRESHOLD field identifies the maximum number of consecutive message authentication errors allowed before the SPDH module generates a new MAC communications key and instructs the terminal to request a download. The SPDH module resets the counter associated with this threshold when a message is authenticated successfully.
- The value in the PIN KEY THRESHOLD field identifies the total number of PINs that can be verified with a PIN communications key.
- The value in the PIN KEY ERROR THRESHOLD field identifies number of failed PIN verification attempts with a PIN communications key.
- The value in the DATA KEY THRESHOLD field identifies the total number of times a data encryption communications key can be used.
- The value in the DATA KEY ERROR THRESHOLD field identifies the number of times a data encryption communications key can be used unsuccessfully.

# Handshaking

The customer must decide whether to support text-level handshake requests. The purpose of this request is to allow the terminal to verify the status of the communications link and, optionally, validate the communication key and authentication key for the terminal.

If the handshake request is made with only the standard message header, the SPDH module ignores every field in the header except for the Transaction Code, Processing Flag 1, and Processing Flag 2 fields. The SPDH module responds with the standard message header, including a response code of 007, and the valid local terminal date and time. In addition, appropriate values in the Processing Flag 1 and Processing Flag 2 fields can be sent in the response, if a mail message or a download is waiting. The SPDH module also returns any optional data fields as configured by the customer in the ACNF.

If the handshake request includes FID b (PIN/Customer), the SPDH module decrypts the PIN and compares the decrypted PIN to 16 zeros. If not equal, the response code is set to 201, indicating an invalid PIN.

If the handshake request includes FID G (Authentication Code), the SPDH module validates the MAC, using the specified MAC FIDs. If not equal, the response code is set to 898, indicating an invalid MAC.

# BASE24-mail Support

The SPDH module is fully compatible with release 6.0 of the BASE24-mail product. Although BASE24-mail must be purchased separately from BASE24-pos, it provides customers with the added advantage of a fully functional electronic mail system. However, it is the customers' decision whether to support BASE24-mail, and to determine the degree to which the product is supported. For example, the customer can choose to allow the terminal only to receive mail, or allow the terminal only to send mail. As another option, the customer can choose to support mail message transmission in both directions. The terminal can also be configured to accept unsolicited mail, or it can be configured to not allow mail to be sent to the terminal without the terminal first initiating a read mail request transaction. In short, customers can choose from several options of sending and receiving mail electronically using BASE24-mail.

## SPDH Module Integration with BASE24-mail

BASE24-mail allows host processors to send informational messages to POS and CRT terminals throughout a network. Mail messages can also be transmitted from these terminals to the host processor.

The host processor has the ability to attach an expiration date and time to every mail message, as well as specify that it is to be notified when a particular message is delivered. The message can be displayed on the terminal immediately, or stored by BASE24-mail on the HP NonStop computer until the terminal requests receipt of any outstanding mail.

Broadcast lists can be created that allow a series of terminals to receive the same message at the same time.

An extracted tape image of all mail messages that were stored and have expired is available for the host. The image shows the message that was transmitted, and indicates if the message was successfully delivered to the terminal. All mail messages that were unable to be transmitted to the host are also included on an extract tape.

A mail message received by a user terminal can be read again by the user until the message expires. The user can specify which message is to be received, or can ask for the next message within the system. However, messages destined for a host are transmitted successfully to the host once and are then purged.

## Role of the SPDH Module During BASE24-mail Processing

With the SPDH module in place along with BASE24-mail, the terminal owner and operator can choose to support mail in any form offered by the product. This includes allowing terminals to send messages or only receive messages, depending on the customer. In addition, customers must determine whether they want to support unsolicited mail messages. Unsolicited mail messages are messages sent to the terminal without a request to read mail messages being sent to the SPDH module.

Any BASE24-mail messages coming from or going to a BASE24-pos Standard POS terminal must pass through the SPDH module. Messages from the terminal are formatted into a message format understood by the BASE24-mail process. The SPDH module formats messages coming from the BASE24-mail process into the terminal native format.

## Unsolicited Mail

When the SPDH module receives an unsolicited mail message from the BASE24-mail process intended for a terminal, the SPDH module checks the ACI Standard Configuration File (ACNF) to determine if the terminal can accept unsolicited mail. The SPDH module also checks if the terminal is a dial-up terminal.

The SPDH module sends unsolicited mail messages to the terminal if the terminal is not a dial-up terminal and the ACNF indicates the terminal can accept unsolicited mail. If both of these conditions are met, the SPDH module sends unsolicited mail messages to the terminal even if the terminal has not sent a read mail request to the SPDH module. The only time the SPDH module does not send unsolicited mail messages to the terminal when these two conditions are met is if the terminal has outstanding requests being processed. Unsolicited mail is sent to the terminal containing a standard message header that appears as though the SPDH module actually received a read mail request. FID V is also included in the message and is configured so the terminal can return it in a subsequent read mail request to retrieve the next piece of undelivered mail.

If mail exists for a terminal and the terminal is unable to accept the mail (i.e., the terminal is a dial-up terminal) or is not configured to support unsolicited mail, the SPDH module sets Processing Flag 1 in the standard message header in each successive response to the terminal. This indicates to the terminal that mail is waiting. The SPDH module continues to notify the terminal that mail exists in this manner until the terminal makes any read mail request (typically, a read first undelivered mail request).

In situations where the terminal is configured to support unsolicited mail, only one unsolicited delivery attempt per piece of mail is made by the SPDH module. When the terminal receives a message with Processing Flag 1 set, the terminal should send a read mail request to retrieve the mail messages that are waiting.

## Send Mail Request

The terminal can send mail to a DPC (data processing center) using the send mail request. In order to support send mail request transactions, the terminal must be able to compose a piece of mail.

The only optional request field required by the SPDH module for this transaction is FID W. FID W is the Mail Text field, which is composed of a destination DPC and a maximum of 449 characters of mail text.

## Read Mail Request

The terminal can request any outstanding mail messages using the read mail request. Typically, the FID V, Mail Key, is required in read mail requests other than initial requests. If the terminal does not include FID V in an initial read mail request, the SPDH module assumes the terminal is required to read the first of any mail of any category. Thus, the initial read mail request that the terminal makes for any mail can be made without FID V. However, all subsequent read mail requests should contain FID V. FID V contains the following four types of information.

**Category Code —** The category code is included whenever the terminal is making a generalized read request. The category code is a 2-position alphanumeric field. Category 99 is reserved for download alerts. The category code is a user-defined value.

**Access Code —** The type of access desired. The access code is a 1-position numeric field. The valid values for this field are as follows:

1 = Read first message. This value directs the SPDH module to read the first piece of any mail existing for the terminal.
2 = Read any next message. This value directs the SPDH module to retrieve the next piece of any mail existing for the terminal.

3 = Read first undelivered message. This value directs the SPDH module to retrieve the first piece of mail existing for the terminal that has not already been delivered.
4 = Read next undelivered message. This value directs the SPDH module to retrieve the next piece of mail existing for the terminal that has not already been delivered.
5 = Read specific message. This value directs the SPDH module to retrieve a specific piece of mail existing for the terminal. A valid Mail ID field must be present to use this value.

**Processing Flag —** Reserved for future use.

**Mail ID —** The date and identification for the message. The mail ID is a 10-position alphanumeric field. The first six bytes contain the date (YYMMDD) of the message. The last four bytes contain the mail message number. This field specifies the starting point for a read next request or the specific piece for a read specific request.

If the SPDH module receives a request from the terminal to read mail, it formats the request and sends it to the BASE24-mail process. The BASE24-mail process then responds and sends the appropriate message back to the SPDH module, which sends it back to the terminal. When the SPDH module receives an acknowledgment from the terminal, the SPDH module sends a completion message to the BASE24-mail process.

## Read Mail Response

The SPDH module responds to a read mail request after requesting the appropriate mail message from the BASE24-mail process. The mail message received from BASE24-mail is placed in FID W, if it is equal to or less than 440 bytes and fits in the field. If the mail fits in FID W, the response code is set to 880, indicating that no more data exists.

If the response does not fit, the response code is set to 881, indicating that more data exists. The SPDH module saves the message context for the anticipated read next request. The terminal echoes the FID V from the previous response in read next requests.

When returning a read mail response, the SPDH module also translates the value in the Access Code field to the value it anticipates will follow.

An 880 response, indicating that no more data exists for a particular piece of mail, does not mean that no more mail exists for the terminal. In order to determine if no more mail exist for the terminal, the terminal must make another read next request. No more mail is implied when the response code is 880 and no mail text is included in the response. In this case, FID W is not included in the response.

## Mail Delivered Request

Terminals are allowed to mark mail as delivered so that subsequent read undelivered mail requests ignore the mail and move on to the next undelivered piece. In order to perform this function, the terminal must submit a mail delivered request containing the FID V (Mail Key) of the read mail request (the previous response). In this case, the Mail ID field in the FID V identifies the mail to be marked as delivered and the Processing Flag field in the FID V is used to mark the mail as delivered.

Terminals must also respond to unsolicited mail they receive with a mail delivered request. If a terminal receives unsolicited mail and does not respond with a mail delivered request, the BASE24-mail process marks the terminal down. No more unsolicited mail is sent to the terminal until a subsequent mail request from the terminal is processed by the BASE24-mail process. If a terminal is capable of supporting unsolicited mail, an unsolicited mail message should be acknowledged by a mail delivered request.

# Interac Online Payment Transaction Identification Requirements

The following are transaction identification requirements for Interac Online Payment (IOP) transactin support. These transactions are used by institutions which support this type of Internet Payment transaction through the Interac network. These transactions were formerly known as iDebit transactions.

Interac Online Payment transactions are supported through the SPDH Device Handler process with the following fields in the inbound device messages:

- FID D (Application Account Type) must be set to a value of 5.
- FID e (POS Condition Code) must be set to a value of 01.
- FID q (Track 2) must begin with an M to indicate that Track 2 was manually entered.
- The Normal Purchase and Merchandise Return options in the ACI Standard Device Configuration File (ACNF) must be set to Y to enable IOP transactions.

# Suppression of Consumer Transaction Data

For security purposes, the SPDH module can be configured to ensure that Track 1 and Track 2 discretionary data does not reside in the temporary context area of the POS Terminal Data File. This applies to transactions that are authorized by BASE24, a host system, or an interchange. Consumer data is suppressed as it is carried internally in the BASE24-pos Standard Internal Message (PSTM) in the 0210 response message.

This data protection is controlled through the POS-DISCR-DATA-PSTM-SUPPRS parameter in the Logical Network Configuration File (LCONF).

See the ***BASE24 Logical Network Configuration File (LCONF) Manual*** for specific configuration information.

*ACI Worldwide, Inc.*

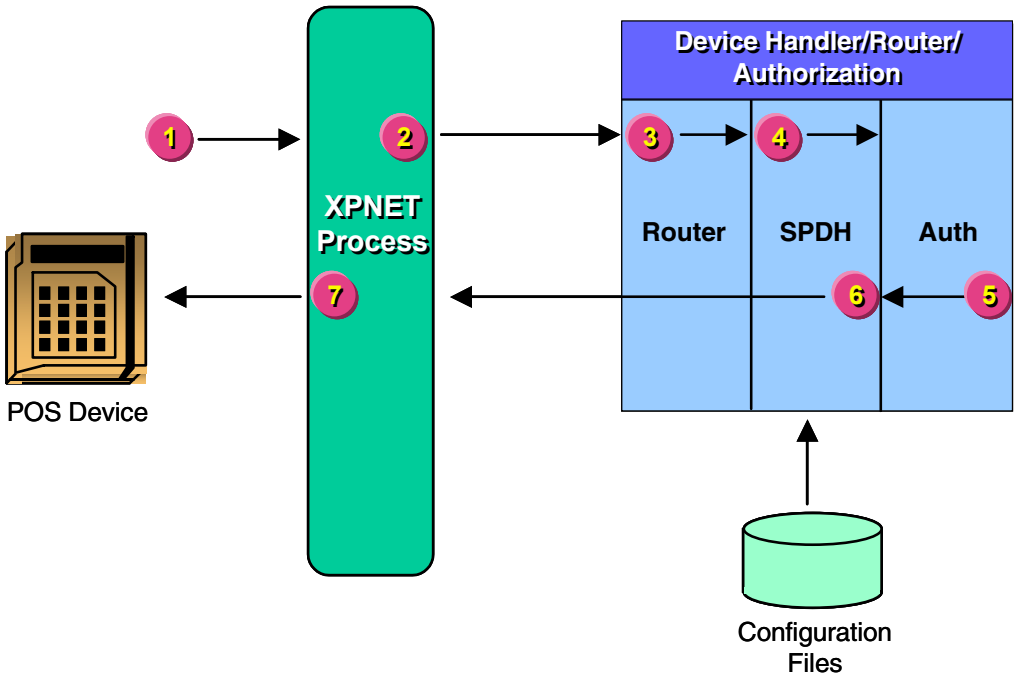
# 7: Transaction Message Flows

This section contains examples of the message flows between a POS device and the SPDH module. Each example includes a diagram illustrating the message flow, followed by a description of each step. The message flows listed below are documented in this section.

- Approved normal purchase received by the device
- Approved normal purchase; communications between device and BASE24 down
- Declined normal purchase received by the device
- Declined normal purchase; communications between the device and BASE24 down
- Reversal generated by a POS device or controller
- Approved transaction reversal
- MAC reversal
- Transaction reversed by a clerk at the POS device
- Unsolicited mail
- Terminal send mail request
- Mail pick up request—single response
- Mail pick up request—multiple response
- Mail pick up request—no mail stored

# Approved Online Purchase Received By Device

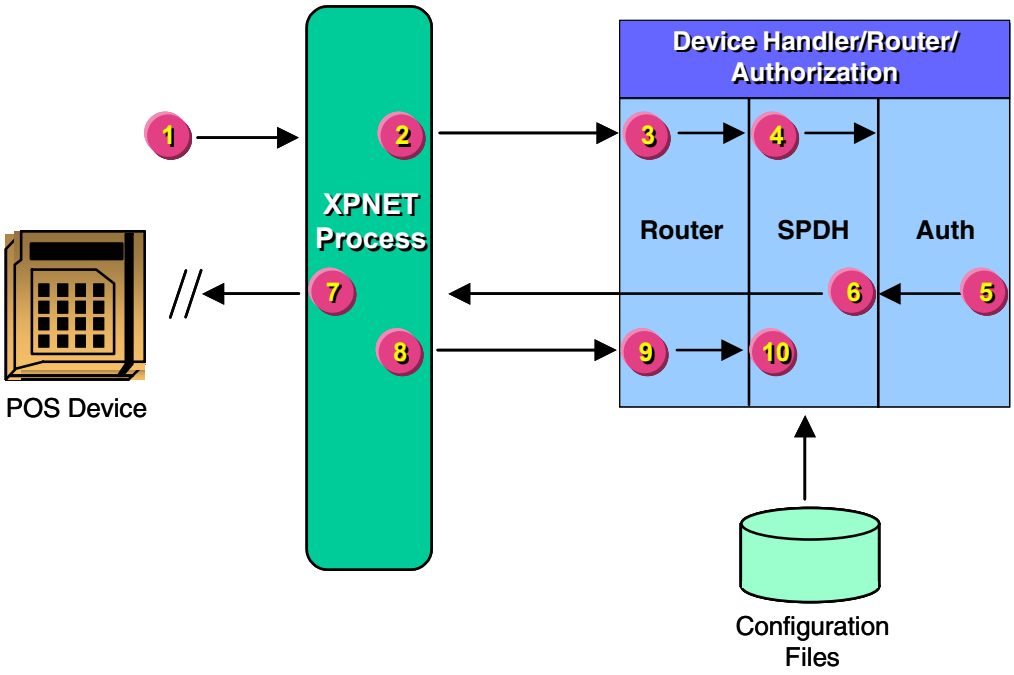
The diagram below illustrates the transaction message flow in a scenario containing an online purchase transaction with a response. In this example, the transaction is approved.




| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device sends its native message for a normal purchase to the SPDH module. This message passes through the XPNET process and the Router module. |
| 4 | The SPDH module validates the request and formats the message in POS Standard Internal Message (PSTM) for a normal purchase. The SPDH module then sends the message to the Authorization module. |
| 5 | The Authorization module returns a response to the SPDH module indicating the transaction was approved. |
| 6, 7 | The SPDH module updates the PTD totals, translates the response from PSTM format into the POS device's native language, and sends the response to the POS device. The native mode response travels though the XPNET process. |

# Approved Normal Purchase; Communications between Device and BASE24 Down

The diagram below illustrates the transaction message flow in a scenario containing a normal purchase transaction that was not received by the device because communications between the device and BASE24 were down. In this example, the transaction was reversed.

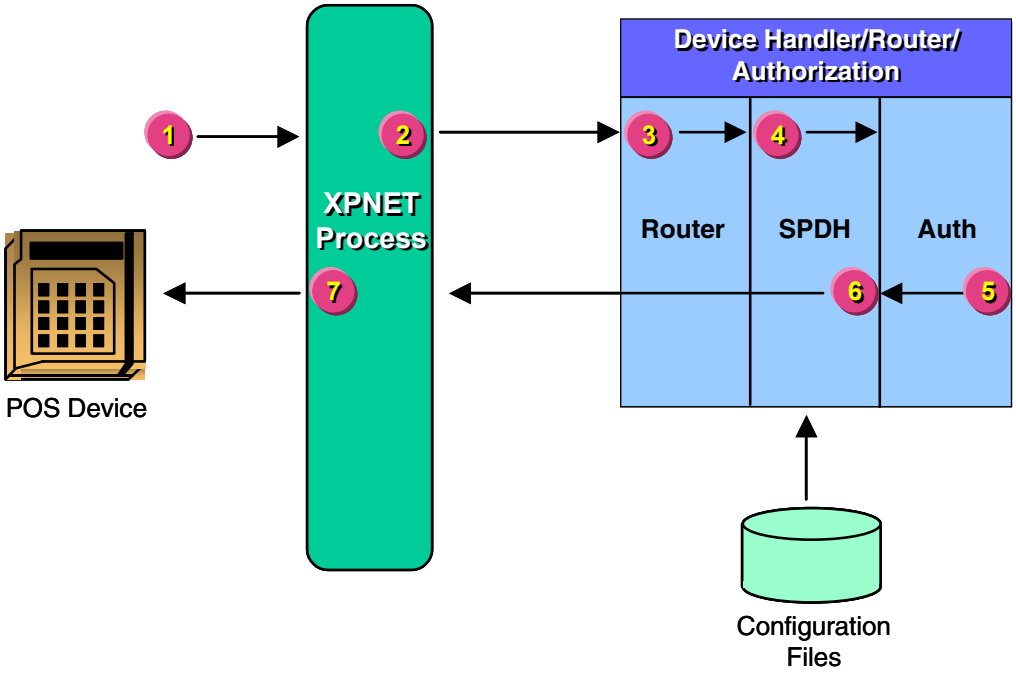



| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device sends its native message for a normal purchase to the SPDH module. This message travels through the XPNET process and the Router module. |
| 4 | The SPDH module validates the request and formats the message in POS Standard Internal Message (PSTM) for a normal purchase. The SPDH module then sends the message to the Authorization module. |
| 5 | The Authorization module returns a response to the SPDH module indicating the transaction was approved. |

| Steps | Processing |
|---|---|
| 6, 7 | The SPDH module updates the PTD totals, translates the response from PSTM format into the POS device's native language, and sends the response to the POS device by way of the XPNET process. The XPNET process attempts to send the response to the POS device, but is unsuccessful. |
| 8, 9 | The XPNET process returns the failed message to the SPDH module. This message travels through the Router module. |
| 10 | The SPDH module generates a reversal transaction and sends it to the Authorization module.<br>When formatting the reversal after validating, the SPDH module formats the message in PSTM format again using the context for a reversal. The SPDH module sends the reversal to the Authorization module and updates PTD totals. |

# Declined Normal Purchase Received by Device

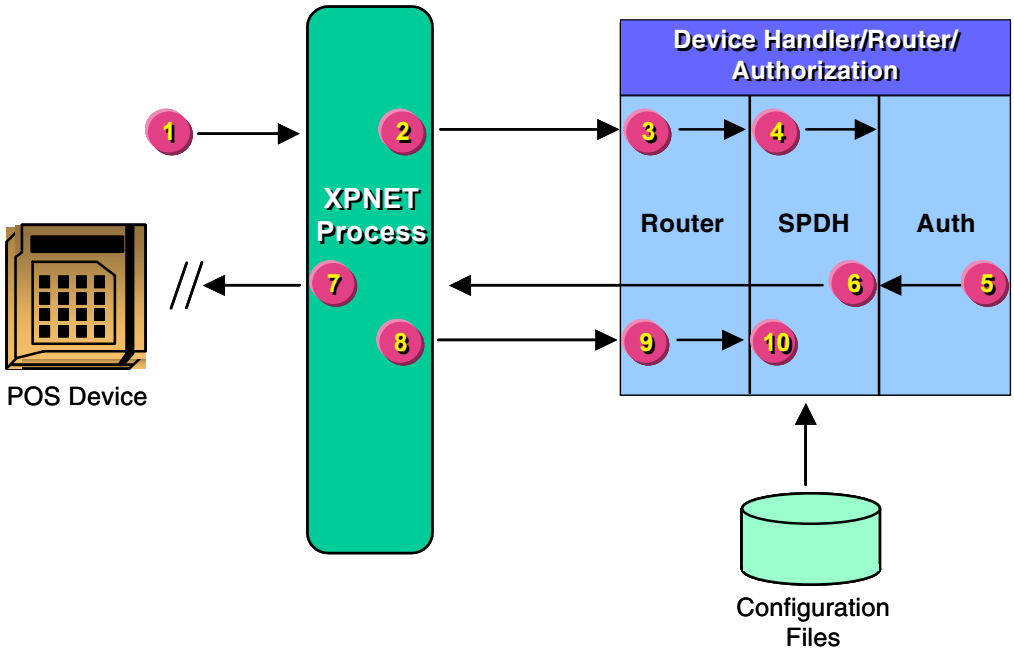
The diagram below illustrates the transaction message flow in a scenario containing a normal purchase transaction with a response. In this example the transaction was declined.




| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device sends its native message for a normal purchase to the SPDH module. This message travels through the XPNET process and the Router module. |
| 4 | The SPDH module validates the request and formats the message in POS Standard Internal Message (PSTM) for a normal purchase. The SPDH module then sends the message to the Authorization module. |
| 5 | The Authorization module returns a response to the SPDH module indicating the transaction was declined. |
| 6, 7 | The SPDH module translates the response from PSTM format into the POS device's native language and sends the response to the POS device. This message travels through the XPNET process. |

# Declined Normal Purchase; Communications between Device and BASE24 Down

The diagram below illustrates the transaction message flow in a scenario containing a normal purchase transaction that was not received by the device because communications between the device and BASE24 were down. In this example, the transaction was declined.



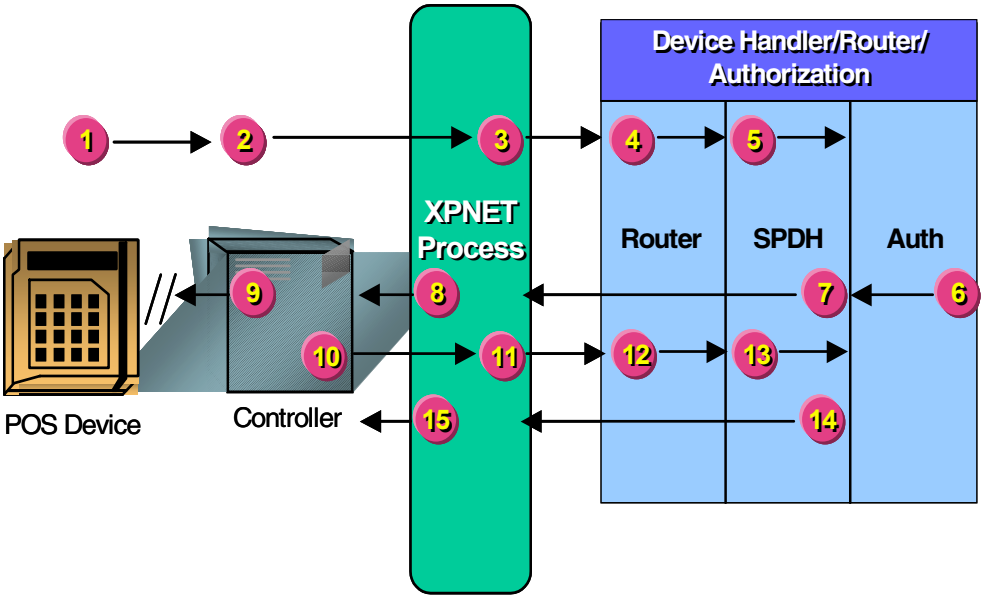

| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device sends its native message for a normal purchase to the SPDH module. This message travels through the XPNET process and the Router module. |
| 4 | The SPDH module validates the request and formats the message in POS Standard Internal Message (PSTM) for a normal purchase. The SPDH module then sends the message to the Authorization module. |
| 5 | The Authorization module returns a response to the SPDH module indicating the transaction was declined. |

| Steps | Processing |
|---|---|
| 6 | The SPDH module translates the response from PSTM format into the POS device's native language and sends the response to the POS device by way of the XPNET process. |
| 7 | The XPNET process attempts to send the response to the POS device, but is unsuccessful. The device times out accordingly. |
| 8, 9 | The XPNET process sends the failed message to the SPDH module. This message travels through the Router module. |
| 10 | The SPDH module drops the message. |

# Controller Reversal

The diagram below illustrates the transaction message flow in a scenario containing a transaction reversed from a controller. In this transaction, the lines of communication between the POS device and the controller go down, and the controller detects this before it receives a response from the SPDH module. The message does not time out. See appendix B for transaction flows illustrating how reversals are processed when the transaction times out.

Refer to controller documentation for more information about the communication between the controller and the POS device.

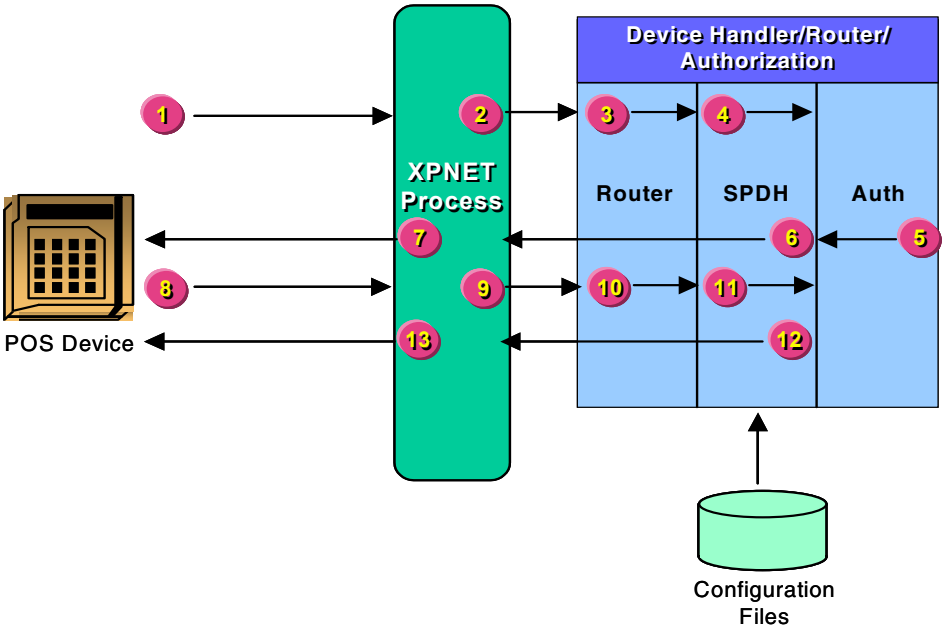



| Steps | Processing |
|---|---|
| 1, 2, 3, 4 | The POS device sends its native message for a normal purchase to the SPDH module. The message passes through the controller, the XPNET process, and the Router module. |
| 5 | The SPDH module validates the request and formats the message in POS Standard Internal Message (PSTM) for a normal purchase. The SPDH module then sends the message to the Authorization module. |
| 6 | The Authorization module returns a response to the SPDH module indicating the transaction was approved. |

| Steps | Processing |
|---|---|
| 7, 8 | The SPDH module updates the PTD totals, translates the response from the PSTM format into the POS device's native language, and sends it to the controller. This message passes through the XPNET process. |
| 9 | The controller tries to send the response to the POS device, but the line is down. |
| 10, 11, 12 | The controller generates a reversal with a message subtype of C and sends it to the SPDH module. This message travels through the XPNET process and the Router module. |
| 13 | The SPDH module updates totals in the PTD, sets the reversal code to 03 (destination not available) and generates a 0420 reversal message and sends it to the Authorization module. |
| 14, 15 | The SPDH module checks the reversal response flag in the ACNF. The flag is set to yes, so the SPDH module echoes the reversal request back to the controller. (If the flag had been set to no, the SPDH module would not echo the reversal request back to the controller.) This message passes through the XPNET process. |

# Approved Transaction Reversal

The diagram below illustrates the transaction message flow in the scenario of a reversal of an approved purchase. In this case, the device received a transaction approval correctly, but for one reason or another the merchant—or the card (in the case of an EMV transaction)—reversed the transaction.

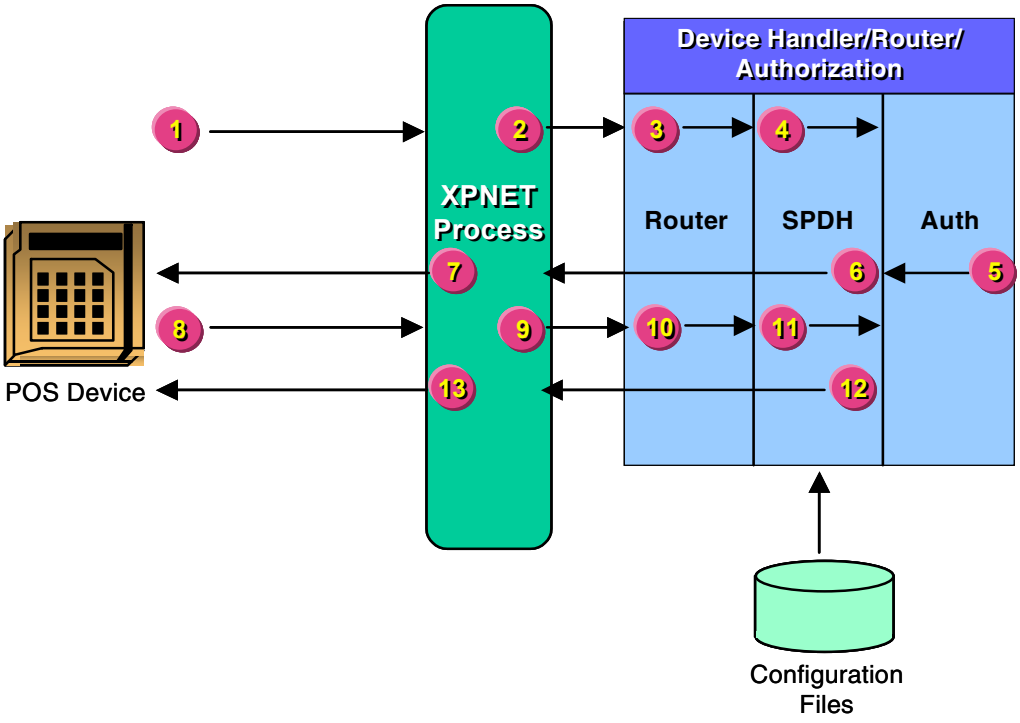



| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device sends its native message for a normal purchase to the SPDH module. The message passes through the XPNET process and the Router module. |
| 4 | The SPDH module validates the request and formats the message in POS Standard Internal Message (PSTM) for a normal purchase. The SPDH module then sends the message to the Authorization module. |
| 5 | The Authorization module returns a response to the SPDH module indicating the transaction was approved. |
| 6, 7 | The SPDH module updates the PTD totals, translates the response from the PSTM format into the POS device's native language, and sends it to the POS device. This message passes through the XPNET process. |

| Steps | Processing |
|---|---|
| 8, 9, 10 | The POS device receives the approval; however, the merchant—or the card itself (if this is an EMV transaction)—does not complete the transaction as planned and initiates a reversal. The POS device generates a reversal with a message subtype of C, and sends it to the SPDH module. This message travels through the XPNET process and the Router module. |
| 11 | The SPDH module updates totals in the PTD, generates a 0420 reversal message and sends it to the Authorization module. |
| 12, 13 | The SPDH module checks the reversal response flag in the ACNF. The flag is set to yes, so the SPDH module echoes the request back to the POS device. (If the flag had been set to no, the SPDH module would not echo the request back to the POS device.) This message passes through the XPNET process. |

# MAC Reversal

The diagram below illustrates the transaction message flow in a scenario containing a reversal of a message with an invalid message authentication code (MAC).

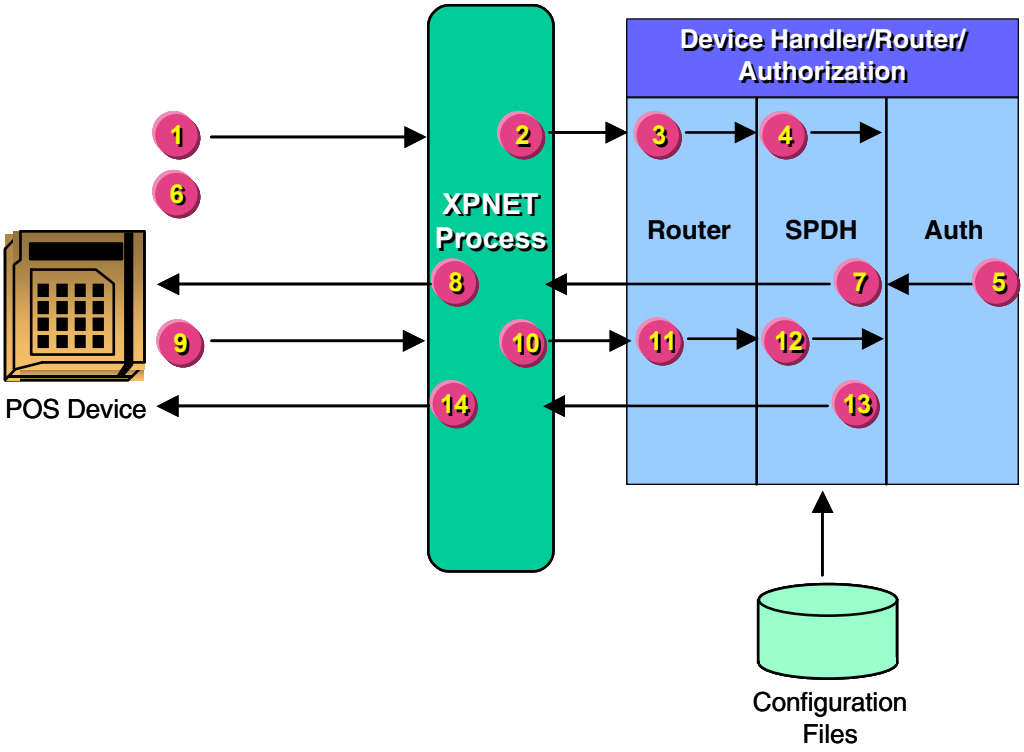



| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device sends its native message for a normal purchase to the SPDH module. The message passes through the XPNET process and the Router module. |
| 4 | The SPDH module validates the request and formats the message in POS Standard Internal Message (PSTM) for a normal purchase. The SPDH module then sends the message to the Authorization module. |
| 5 | The Authorization module returns a response to the SPDH module indicating the transaction was approved. |

| Steps | Processing |
|---|---|
| 6, 7 | The SPDH module updates the PTD totals, translates the response from the PSTM format into the POS device's native language, and sends it to the POS device. This message passes through the XPNET process. |
| 8, 9, 10 | The POS device suspects a corrupted message, generates a reversal with a message subtype of R, and sends it to the SPDH module. This message travels through the XPNET process and the Router module. |
| 11 | The SPDH module updates totals in the PTD, sets the reversal code to 21 (MAC failure) and generates a 0420 reversal message and sends it to the Authorization module.<br>The SPDH module checks the POS-DH-CONS-MAC-ERR-LMT parameter in the LCONF. If the limit has been reached, the SPDH module sets a flag that indicates the SPDH module needs to send a new MAC key on its next response to the POS device. If the limit has not been reached, the SPDH module increases the consecutive MAC error counter by one. |
| 12, 13 | The SPDH module checks the reversal response flag in the ACNF. The flag is set to yes, so the SPDH module echoes the request back to the POS device. (If the flag had been set to no, the SPDH module would not echo the request back to the POS device.) This message passes through the XPNET process. |

# Customer-Cancellation Reversal

The diagram below illustrates the transaction message flow in a scenario containing a normal purchase transaction that is cancelled by the clerk at the POS device before the SPDH module sends a response.




| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device sends its native message for a normal purchase to the SPDH module. The message passes through the XPNET process and the Router module. |
| 4 | The SPDH module validates the request and formats the message in POS Standard Internal Message (PSTM) for a normal purchase. The SPDH module then sends the message to the Authorization module. |
| 5 | The Authorization module returns a response to the SPDH module indicating the transaction was approved. |

| Steps | Processing |
|---|---|
| 6 | The clerk cancels the transaction at the POS device. The POS device waits to send the reversal to the SPDH module until after it receives the response to the original transaction. |
| 7, 8 | The SPDH module updates the PTD totals, translates the response from the PSTM format into the POS device's native language, and sends it to the POS device. This message passes through the XPNET process. |
| 9, 10, 11 | The POS device generates a reversal with a message subtype of U and sends it to the SPDH module. This message travels through the XPNET process and the Router module. |
| 12 | The SPDH module updates totals in the PTD, sets the reversal code to 08 (customer canceled), generates a 0420 reversal message, and sends it to the Authorization module. |
| 13, 14 | The SPDH module checks the reversal response flag in the ACNF. The flag is set to yes, so the SPDH module echoes the request back to the POS device. (If the flag had been set to no, the SPDH module would not echo the request back to the POS device.) This message passes through the XPNET process. |

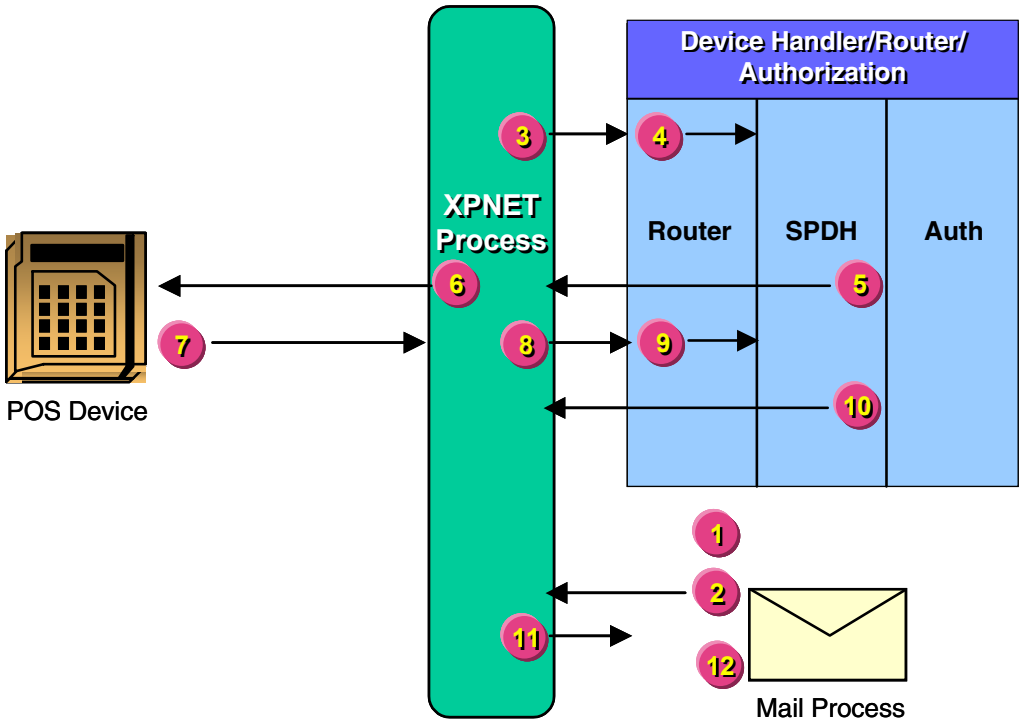
# BASE24-mail Transaction Flows

The SPDH module is fully compatible with BASE24-mail. The following diagrams illustrate the exchange of messages between the SPDH module and the POS device during transaction processing using BASE24-mail. The data shown is assumed to follow the standard message header and to be terminated with an ETX in each situation. The quotes merely delimit the text and are not part of the request.

## Unsolicited Mail

The diagram below illustrates the transaction message flow in a scenario where the SPDH module receives unsolicited mail for a POS device.

**Note:** Dial-up POS devices cannot receive unsolicited mail.

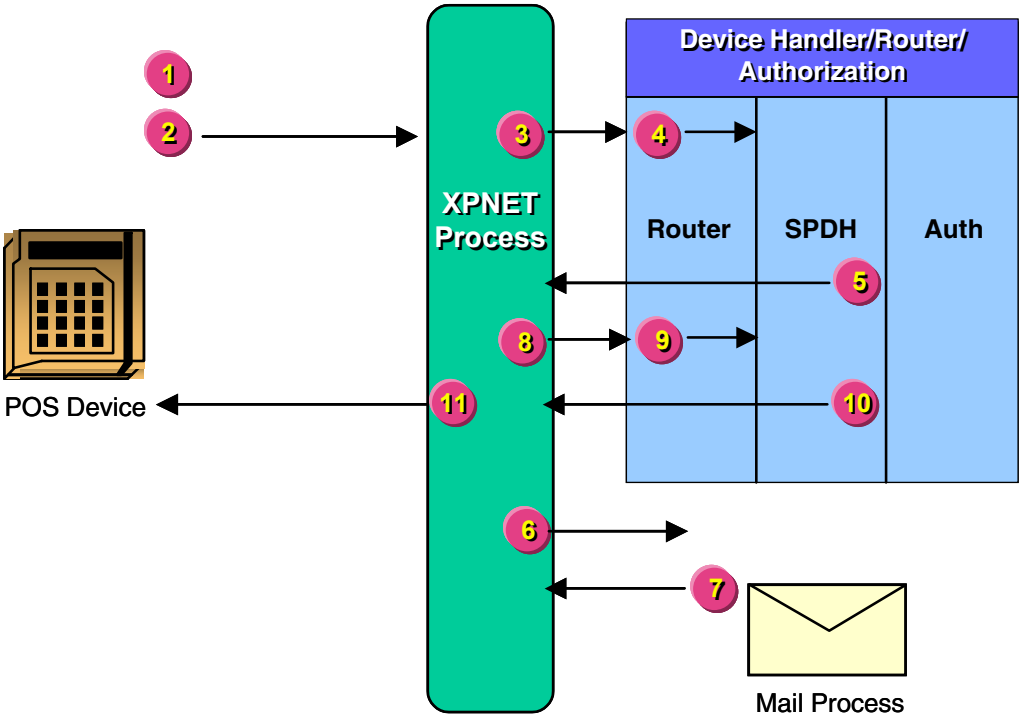



| Steps | Processing |
|---|---|
| 1 | The Mail process receives a message indicating that it must send a message to the POS device for immediate delivery. This message can come from the host, a Pathway screen, or another POS device. |
| 2, 3, 4 | The Mail process sends an MB641 message to the SPDH module. This message travels through the XPNET process and Router module. |

| Steps | Processing |
|---|---|
| 5, 6 | If the POS device can receive unsolicited mail and there is no transaction in progress, the SPDH module formats a native mode message containing the mail message and sends it to the POS device. This message travels through the XPNET process.<br>If the POS device cannot receive unsolicited mail or there is a transaction in progress, the SPDH module sets a flag indicating that there is mail waiting. The next time the POS device receives a response, the response code indicates that there is mail waiting. The POS device can then request the mail. Refer to "Mail Pick-up Request—Single Response" later in this section for more information about how a POS device can receive this mail. |
| 7, 8, 9 | The POS device responds with a "mark mail as delivered" message to the SPDH module. This message travels through the XPNET process and Router module. |
| 10, 11 | The SPDH module sends an MB643 message to the Mail process. This message travels through the XPNET process. |
| 12 | The Mail process receives the MB643 message and marks the piece of mail as delivered. |

# Terminal Send Mail Request

The diagram below illustrates the transaction message flow in a scenario where the POS device allows the operator to compose a piece of mail and route it to one of the three DPCs set up in the ACI Standard Device Configuration File (ACNF).

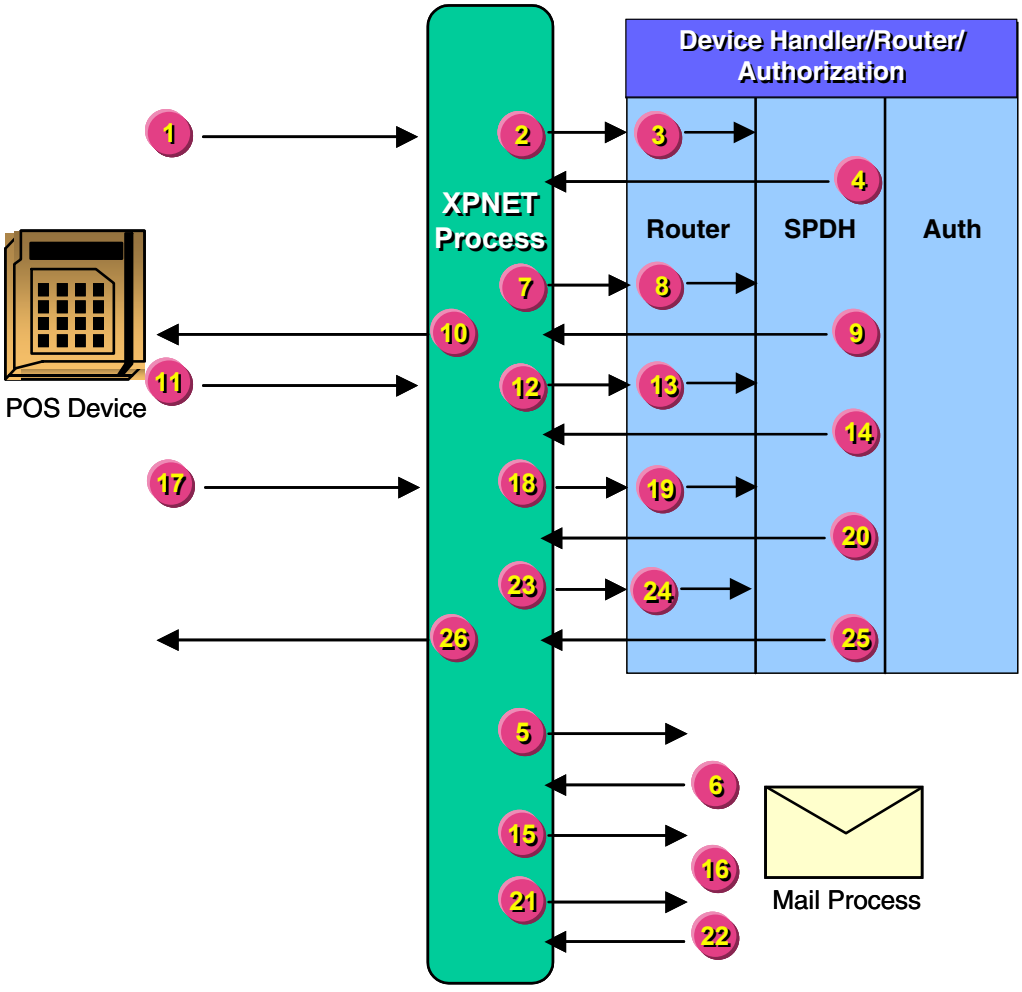



| Steps | Processing |
|---|---|
| 1 | The POS device allows the operator to compose a piece of mail and then formats a Send Mail Request containing the mail text. This is included in FID W (Mail/Download Text) along with the number of the data processing center (DPC) to which the mail is to be delivered. A POS device can route mail to one of three possible DPCs. DPCs are defined in the ACI Standard Device Configuration File (ACNF). The request is formatted as ".W1*text*". |
| 2, 3, 4 | The POS device sends the Send Mail Request to the SPDH module. The message travels through the XPNET process and Router module. |

| Steps | Processing |
|---|---|
| 5, 6 | The SPDH module translates the Send Mail Request from native mode to an MB640 message and sends the message to the Mail process. This message travels through the XPNET process. |
| 7, 8, 9 | The Mail process attempts to send the mail to the indicated DPC. If it cannot, it stores the mail in the Mail Box File (MBF) for later delivery. It then responds to the SPDH module indicating whether the mail was delivered or stored. This message travels through the XPNET process and Router module. |
| 10, 11 | If the attempt is successful, the SPDH module responds to the POS device with a response code of 870 (OK, delivered). If the attempt is unsuccessful, the SPDH module responds with a response code of 871 (OK, mail stored). This message travels through the XPNET process. |

# Mail Pick Up Request—Single Response

The following diagram illustrates a scenario where there is mail waiting to be delivered to the POS device and the POS device initiates a request to receive the mail. In this example, the text of the mail message is small enough for one response.



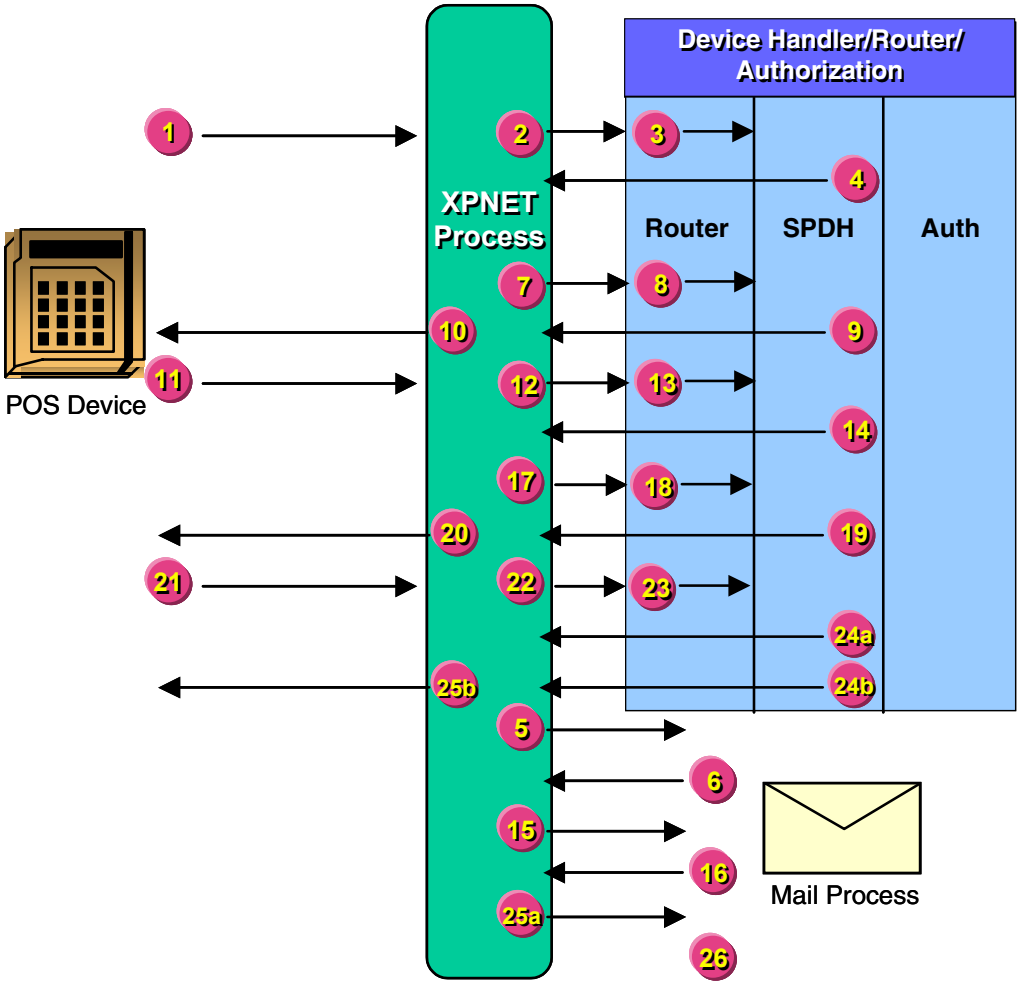
| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device formats a Read Mail Request and sends it to the SPDH module. This message travels through the XPNET process and Router module.<br>The POS device is asking for the first of any undelivered mail using FID V (Mail/Download Key). The POS device could have instead requested the first of any mail available. In this case, the presence of FID V is not required. The message is formatted as ".V00300000000000". |
| 4, 5 | The SPDH module translates the request from the POS device's native mode and sends an MB640 request to the Mail process. This message travels through the XPNET process. |
| 6, 7, 8 | The Mail process determines from the MB640 request that the POS device is requesting mail. The Mail process reads the Mail Box File (MBF), formats an MB641 mail response message, and sends the message to the SPDH module. This message travels through the XPNET process and Router module. |
| 9, 10 | The SPDH module translates the response into the POS device's native mode and sends it to the POS device. This message travels through the XPNET process.<br>The SPDH module responds to the POS device with the first piece of undelivered mail. The mail includes the mail key and mail text. If the mail fits into a single response, as it does in this example, the response code is 880 (mail message has been received in its entirety). The response is formatted as ".V00419202170068".<br>**Note:** In this example, the Processing Flag in FID V is set to 1. The POS device must ensure this value is echoed as received. |
| 11, 12, 13 | The POS device sends a message to the SPDH module to mark the mail as delivered. This message travels through the XPNET process and Router module. |

| Steps | Processing |
|---|---|
| 14, 15 | The SPDH module formats an MB643 completion and sends it to the Mail process. This message travels through the XPNET process.<br>The SPDH module then formats and sends a native mode response to the POS device that includes response code 880 (mail message has been received in its entirety) and the mail key for the delivered message. The request is formatted as "V00419202170068". |
| 16 | The Mail process updates the mail record in the MBF and marks the mail as being read. Information in the MBF concerning the POS device is also updated at this time by the Mail process. |
| 17, 18, 19 | Since FID W (Mail/Download Text) was included in the previous mail response, the POS device operator sends a Read Next Request to the SPDH module to read the next piece of mail. The request is formatted as ".V00419202170068". This message travels through the XPNET process and Router module. |
| 20, 21 | The SPDH module sends an MB640 request to the Mail process. This message travels through the XPNET process.<br>The SPDH module always sends the mail key from the previous piece of mail to the Mail process when it requests the next piece of mail. When the last packet of a mail message is delivered, the Mail process uses the mail key to update the MBF. Any subsequent Read Next Requests would then cause the Mail process to retrieve the next piece of mail from the MBF starting after the key of the last piece of mail read. |
| 22, 23, 24 | The Mail process reads the MBF, finds no mail, and responds to the SPDH module with an MB641 no mail response. This response travels through the XPNET process and Router module. |

| Steps | Processing |
|---|---|
| 25, 26 | The SPDH module translates the response from native mode and responds to the POS device with a response code of 880 (mail message has been received in its entirety) and the absence of FID W (Mail/Download Text). This indicates that all mail has been read. The response is formatted as ".V00419202170068".<br><br>**Note:** The POS device operator can continue to send Read Next Requests to the SPDH module as long as the SPDH module responses contain a response code of 880 and information in FID W (Mail/Download Text). The absence of FID W indicates that all mail has been read. |

# Mail Pick Up Request—Multiple Response

The following diagram illustrates a scenario where there is mail waiting to be delivered to the POS device and the POS device initiates a request to receive the mail. In this example, the text of the mail message is too large for one response.

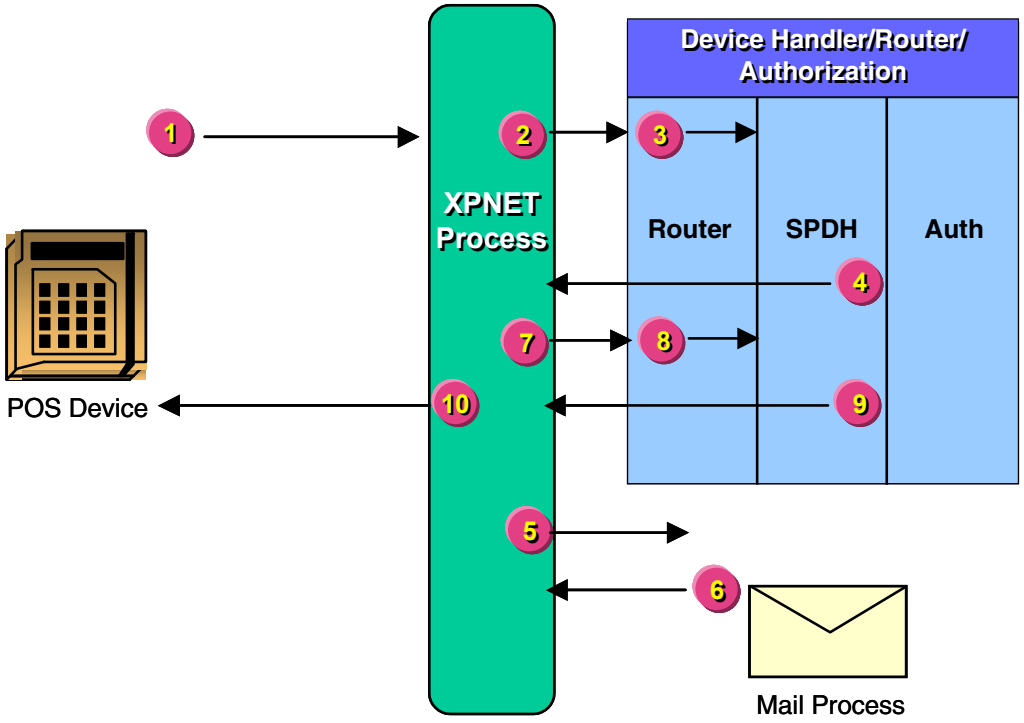


| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device formats a Read Mail Request and sends it to the SPDH module. This message travels through the XPNET process and Router module.<br>The POS device is asking for the first of any undelivered mail using FID V (Mail/Download Key). The POS device could have instead requested the first of any mail available. In this case, the presence of FID V is not required. The message is formatted as ".V00300000000000". |
| 4, 5 | The SPDH module translates the request from the POS device's native mode and sends an MB640 request to the Mail process. This message travels through the XPNET process. |
| 6, 7, 8 | The Mail process determines from the MB640 request that the POS device is requesting mail. The Mail process reads the Mail Box File (MBF), formats an MB641 mail response message, and sends the message to the SPDH module. This message travels through the XPNET process and Router module. |
| 9, 10 | The SPDH module translates the response into the POS device's native mode and sends it to the POS device. The response code is set to 881 (mail message received successfully and there is more data for this mail message). This message travels through the XPNET process. |
| 11, 12, 13 | The POS device sends a request to the SPDH module for the next packet of information. This message travels through the XPNET process and Router module. |
| 14, 15 | The SPDH module sends an MB641 request to the Mail process for more data. This message travels through the XPNET process. |
| 16, 17, 18 | The Mail process responds to the SPDH module with the mail text. This message travels through the XPNET process and the Router module. |

| Steps | Processing |
|---|---|
| 19, 20 | The SPDH module translates the response into the POS device's native mode and sends it to the POS device. The response contains the next packet of information. This message travels through the XPNET process.<br>Steps 11–20 repeat until no more packets are needed. Along with the last packet the SPDH module responds with a response code of 880 (mail message has been received in its entirety) to the POS device. |
| 21, 22, 23 | The POS device sends a message to the SPDH module to mark the mail as delivered. This message travels through the XPNET process and Router module. |
| 24, 25 | The SPDH module performs as follows:<br>a. Formats an MB643 completion and sends it to the Mail process. This message travels through the XPNET process. Processing continues with step 26.<br>b. Formats and sends a native mode response to the POS device that includes response code 880 (mail message has been received in its entirety) and the mail key for the delivered message. The request is formatted as ".V00419202170068". This leg of processing is now complete. |
| 26 | Steps 1–25 repeat until in step 6 the Mail process reads the MBF, finds no mail, and responds with MB641 no mail response message. |

# Mail Pick Up Request—No Mail Stored

The following diagram illustrates a scenario where the POS device initiates a request to receive mail. In this example, no mail is stored for the specified POS device.




| Steps | Processing |
|---|---|
| 1, 2, 3 | The POS device sends a mail pick up request to the SPDH module. This message travels through the XPNET process and Router module. |
| 4, 5 | The SPDH module translates the request from the POS device's native mode and sends an MB640 request to the Mail process. This message travels through the XPNET process. |
| 6, 7, 8 | The Mail process determines from the MB640 request that the POS device is requesting mail. The Mail process reads the Mail Box File (MBF), determines there is no mail for the specified POS device, formats an MB641 no mail response message, and sends the message to the SPDH. This message travels through the XPNET process and Router module. |

| Steps | Processing |
|---|---|
| 9, 10 | The SPDH translates the response into the POS device's native mode and sends it to the POS device. The response code 880 (Mail message has been received in its entirety) is included. This message travels through the XPNET process. |

*ACI Worldwide, Inc.*

# 8: Configuration Files Maintenance and Set Up

The BASE24 Standard POS Device Handler (SPDH) module is the exclusive reader of the following files:

- ACI Standard Device Configuration File (ACNF)
- ACI Standard Device Response File (ARSP)
- Response Code Description File (RCDF)

The ACNF and ARSP must contain a record or set of records for each terminal group in the network. The RCDF contains records to override response codes contained in the ARSP. All three files are accessed through BASE24-pos files maintenance.

BASE24-pos files maintenance manipulates files through the use of formatted screens. This section describes the formatted screens used to access the ACNF and ARSP for files maintenance and set up. The screens used to access the RCDF are described in the ***BASE24-pos Files Maintenance Manual.***

The SPDH configuration files are set up in the same manner as most other files in the BASE24-pos system. Additional information on BASE24-pos files maintenance can be found in the ***BASE24 Base Files Maintenance Manual***, the ***BASE24-pos Files Maintenance Manual***, and the ***BASE24 CRT Access Manual***. The information provided in this section, however, should be sufficient to set up the necessary records in the ACNF, ARSP, and RCDF for persons familiar with BASE24 CRT access and files maintenance.

# Prerequisites for Configuration

The SPDH module requires Device Handler-specific files, as well as modifications to standard BASE24-pos and Base files. The files required to use the SPDH module are outlined in section 2. Before a customer can be configured to use the SPDH module, the SPDH module must be properly installed, the module needs to be added to the system, and the BASE24-pos network must be enabled and available for files maintenance.

# ACI Standard Device Configuration File

The ACI Standard Device Configuration File (ACNF) contains configuration data used for processing by the SPDH module and downloading to the terminal.

The data in this file is divided into three categories defined as processing data, format data, and download data. The processing data contains timeout values and other processing parameters. The format data consists of up to 30 card prefix ranges used to define processing parameters by card prefix range. The download data contains all the information sent to the terminal in a download.

ACNF records are added and maintained through BASE24-pos files maintenance. To access the ACNF, select POS from the Primary Menu and ACNF from the POS Product Menu. There are four types, or sets, of screens associated with the ACNF. These are used to access the seven records that can be stored in the ACNF. These screen types are as follows:

- Selection Screen
- Processing Record Screen
- Field Map Screens
- Download Record Screens

# Selecting ACNF Records

The first screen associated with the ACNF allows you to choose the record type. This screen is your main menu to ACNF records. The selection screen is used to determine whether SPDH module processing parameters, formatting parameters, or download parameters are being configured. The terminal group is used to identify the terminals to which the configuration information applies once it is set up. The terminal group corresponds to all terminals that have the same value in the TERMINAL GROUP field on POS Terminal Data files (PTD) screen 1.

## Accessing the Selection Screen

The ACNF Selection screen is the customer's primary link to accessing ACNF records. To access the ACNF Selection Screen, perform the following steps. For more information on CRT access, refer to the ***BASE24 CRT Access Manual***.

1. Access the Virtual Menu by entering your assigned logon, password, and logical network on the BASE24 Logon screen. Press the **F1** key to move to the Virtual Menu.
2. Enter ACNF in the FILE DESTINATION field at the bottom of the Virtual Menu and press the **F1** key to display ACNF screen 1. From a file screen, press the **F16** key to display ACNF screen 1.
3. Enter the appropriate values in the following fields on ACNF screen 1, based on the parameters you want to configure (i.e., processing, formatting, or download), and the terminal group to which you want the parameters to apply:

   TERMINAL GROUP
   RECORD TYPE
   RECORD ID

   The screen you access is determined by the values placed in the RECORD TYPE and RECORD ID fields.

   Field descriptions for these fields, together with ACNF screen 1, are provided later in this section.
4. Move to the appropriate ACNF screen by pressing the **F9** key.

## Selection Screen

ACI Standard Device Configuration File (ACNF) screen 1 is shown below, followed by its field descriptions.

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  01 OF 01
         TERMINAL GROUP:  0000      RECORD TYPE:      RECORD ID:

     RECORD      RECORD                    RECORD    RECORD  DOWNLOAD
      TYPE         ID    DESCRIPTION        TYPE       ID    PACKETS
  -------------  ------  -----------    ----------  ------  --------------
  Processing  P    00    Requests &     Download D    00    A  - J
                         Key                          01    K  - T
                         Thresholds                   02    U  - Z  plus 2
                                                      03    0  - 3
  Field map   F    00    Requests                     04    4  - 7
                   01    Responses                    05    8  - 9  plus 2
                                                      06    a  - z
                                                      07    10 - 13
  PLEASE FILL IN KEYS AND PRESS KEY DESIRED           08    14 - 17
                                                      09    18 - 21
                                                      10    22 - 25
                                                      11    26 - 29


 ********************************** BASE24 ***********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

**TERMINAL GROUP —** The terminal group combined with the RECORD TYPE and RECORD ID is the primary key into the ACNF. It identifies the terminal or group of terminals to which the configuration information set up in the ACNF applies. The configuration information is used by all terminals that have a value corresponding to this field in the TERMINAL GROUP field on POS Terminal Data files (PTD) screen 1.

Field Length: 4 alphanumeric characters
Required Field: Yes
Default Value: 0000
Data Name: ACNF.KEY-S.GRP

**RECORD TYPE —** A value identifying whether the record is a processing record, a formatting record, or a download record. This field is used in conjunction with the RECORD ID field. Valid values include the following:

D = Download record
F = Formatting record
P = Processing record

| | |
|---|---|
| Field Length: | 1 alphanumeric character |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ACNF.KEY-S.REC-TYPE |

**RECORD ID —** A value identifying the type of record being accessed. Only certain values are allowed for the different record types. For example, 00 must be entered in this field when P is entered in the RECORD TYPE field. The other valid combinations are listed below.

D allows 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11
F allows 00 or 01
P allows 00

| | |
|---|---|
| Field Length: | 2 numeric characters |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ACNF.KEY-S.REC-ID |

# Processing Records

The Processing Record screen allows customers to set up general processing parameters used by the SPDH module when handling messages. From this screen, you can configure several timeout options, including the amount of time the SPDH module waits for responses from the Authorization and Mail processes. You can also determine the maximum length of responses from the SPDH module to the terminal and whether the SPDH module is to format user data from the USER-DATA field in the POS Standard Internal Message (PSTM) with every financial transaction. In addition, you can determine if totals returned to the terminal in responses are draft capture totals only or if the totals include all transactions.

Several BASE24-mail options can also be set from this screen. You can determine whether unsolicited mail messages are sent to the terminal, whether a terminal is allowed to perform implicit closes, and the data processing centers to which mail should be sent.

## Setting Up Processing Records

The ACNF Processing Record screen is accessed through BASE24 files maintenance much in the same manner as the ACNF Selection screen. Once accessed, you can set up the processing parameters to meet your specific requirements. The following steps describe the procedures to set up the Processing Record screen. These procedures assume you have already accessed the ACNF Selection screen. To set up the ACNF Processing Record screen, perform the following steps:

1. Enter the appropriate terminal group in the TERMINAL GROUP field on screen 1 of the ACNF (Selection screen).
2. Enter P in the RECORD TYPE field on screen 1 of the ACNF. The P in this field identifies the record type as processing records.
3. Enter 00 in the RECORD ID field on screen 1 of the ACNF. The 00 in this field identifies the record as information used in requests from the terminal. 00 is the only value allowed in the RECORD ID field when the RECORD TYPE is P.
4. Press the **F9** key from the ACNF Selection screen to display ACNF Processing Record screen 2.

5. Enter the appropriate values in the following fields on ACNF Processing Record screen 2, based on the processing parameters you want to configure:

   UNSOLICITED MAIL
   DC/ALL TOTALS
   IMPLICIT CLOSES
   OLD MSGS
   PSTM USER FLD
   REVERSAL RESPONSES
   INCLUDE KMAC IN MAC
   INCLUDE KME IN MAC
   INCLUDE DPE IN MAC
   MAX RESP LEN (BYTES)
   AUTH TIMEOUT (SECS)
   LOAD TIMEOUT (SECS)
   MAIL TIMEOUT (SECS)
   OLD MSG TIMEOUT (SECS)
   DPC #0
   DPC #1
   DPC #2

   Field descriptions for these fields, together with ACNF Processing Record screen 2, are provided later in this section.

6. Press the **F9** key from the ACNF Selection screen to display ACNF Processing Record screen 3.

7. Enter the appropriate values in the following fields on ACNF Processing Record screen 3, based on the dynamic key processing parameters you want to configure:

   MAC KEY THRESHOLD
   MAC KEY ERROR THRESHOLD
   CONSECUTIVE MAC KEY ERROR THRESHOLD
   PIN KEY THRESHOLD
   PIN KEY ERROR THRESHOLD
   DATA KEY THRESHOLD
   DATA KEY ERROR THRESHOLD

8. Press the **F3** key to add the record.

9. Read the data associated with these records by pressing the **F9** key to access the screen, then press the **F2** key to read the record.

# Processing Record Screen 2

ACI Standard Device Configuration File (ACNF) screen 2 is shown below, followed by its field descriptions.

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 03
        TERMINAL GROUP:  0000      RECORD TYPE:  P   RECORD ID:  00
                       PROCESSING RECORD SCREEN

                          ( N = NO, Y = YES )
   UNSOLICITED MAIL:      N  DC/ALL TOTALS:        N  IMPLICIT CLOSES:      N
   OLD MSGS:              N  PSTM USER FLD:        N  REVERSAL RESPONSES:   N
   INCLUDE KMAC IN MAC:   N  INCLUDE KME IN MAC:   N  INCLUDE DPE IN MAC:   N
   RESERVED:              N  RESERVED:             N  RESERVED:             N
   RESERVED:              N  RESERVED:             N  RESERVED:             N
   RESERVED:              N  RESERVED:             N  RESERVED:             N
   RESERVED:              N  RESERVED:             N  RESERVED:             N

 MAX RESP LEN( BYTES ):          0  AUTH TIMEOUT( SECS ):           0
 LOAD TIMEOUT( SECS ):           0  MAIL TIMEOUT( SECS ):           0
 OLD MSG TIMEOUT( SECS ):        0

 DPC #0:    0               DPC #1:    0          DPC #2:    0


 ********************************** BASE24 **********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

**UNSOLICITED MAIL —** Indicates whether unsolicited mail messages are sent to the terminal. Valid values are as follows:

Y = Yes, unsolicited mail messages are sent to the terminal.
N = No, unsolicited mail messages are not sent to the terminal.

| | |
|---|---|
| Field Length: | 1 alphabetic character |
| Required Field: | Yes |
| Default Value: | N |
| Data Name: | ACNF.PRO-DATA.FLAGS |

**DC/ALL TOTALS —** Indicates if totals returned to the terminal in responses includes all transactions (both draft capture and non–draft capture) or just draft capture transactions. Valid values are as follows:

Y = Yes, totals for all transactions are returned to the terminals.
N = No, totals for only draft capture transactions are returned to the terminals.

Field Length: 1 alphabetic character
Required Field: Yes
Default Value: N
Data Name: ACNF.PRO-DATA.FLAGS

**IMPLICIT CLOSES —** Indicates whether the terminal is allowed to perform implicit closes. Valid values are as follows:

Y = Yes, implicit closes are allowed.
N = No, implicit closes are not allowed.

Field Length: 1 alphabetic character
Required Field: Yes
Default Value: N
Data Name: ACNF.PRO-DATA.FLAGS

**OLD MSGS —** Indicates the procedure to be taken with messages that have expired. Valid values are as follows:

Y = Yes, log an error message before dropping the expired message.
N = No, do not log an error message before dropping the expired message.

Field Length: 1 alphabetic character
Required Field: Yes
Default Value: N
Data Name: ACNF.PRO-DATA.FLAGS

**PSTM USER FLD —** Indicates whether the SPDH module is to format user data in the POS Standard Internal Message (PSTM) with every financial transaction. Valid values are as follows:

Y = Yes, format user data in the PSTM with every financial transaction.
N = No, do not format user data in the PSTM with every financial transaction.

Field Length: 1 alphabetic character
Required Field: Yes
Default Value: N
Data Name: ACNF.PRO-DATA.FLAGS

**REVERSAL RESPONSES —** Indicates whether the SPDH module is to send a reversal response to the terminal. Valid values are as follows:

Y = Yes, echo responses to the device for reversal requests.
N = No, do not echo responses to the device for reversal requests.

If your terminal software or controller software supports timeout reversals, this flag should be set to Y.

Field Length: 1 alphabetic character
Required Field: Yes
Default Value: N
Data Name: ACNF.PRO-DATA.FLAGS

**INCLUDE KMAC IN MAC —** Indicates whether the MAC in messages exchanged between the SPDH module and the terminal include the MAC communications key (KMAC). Valid values are as follows:

Y = Yes, include the KMAC when generating the MAC.
N = No, do not include the KMAC when generating the MAC.

Field Length: 1 alphabetic character
Required Field: Yes
Default Value: N
Data Name: ACNF.PRO-DATA.FLAGS

**INCLUDE KME IN MAC —** Indicates whether the MAC in messages exchanged between the SPDH module and the terminal include the data encryption communications key (KME). Valid values are as follows:

Y = Yes, include the KME when generating the MAC.
N = No, do not include the KME when generating the MAC.

Field Length: 1 alphabetic character
Required Field: Yes
Default Value: N
Data Name: ACNF.PRO-DATA.FLAGS

**INCLUDE KPE IN MAC —** Indicates whether the MAC in messages exchanged between the SPDH module and the terminal include the PIN communications key (KPE). Valid values are as follows:

Y = Yes, include the KPE when generating the MAC.
N = No, do not include the KPE when generating the MAC.

Field Length: 1 alphabetic character
Required Field: Yes
Default Value: N
Data Name: ACNF.PRO-DATA.FLAGS

**RESERVED —** The fields marked RESERVED are included to allow for future expansion. All of these fields default to N.

**MAX RESP LEN( BYTES ) —** Defines the maximum length, in bytes, for a transaction response message that can be sent to a terminal. This value varies depending on the protocol being used. The length set in this field must be 3 to 5 bytes less than the value specified in the XMITLEN attribute for the device in the XPNET system. Valid values are 96 to 1024 bytes.

Field Length: 1 to 4 numeric characters
Required Field: Yes
Default Value: 0
Data Name: ACNF.PRO-DATA.MAX-RESP-LEN

**AUTH TIMEOUT( SECS ) —** Defines the length of time, in seconds, the SPDH module waits for a response from the Authorization module before responding to the terminal with a timeout response code. This value should be less than the value specified in the TIMEOUT attribute for the line in the XPNET system. The SPDH module checks all timeout fields for a valid value between 5 and 600 inclusive.

Field Length: 1 to 4 numeric characters
Required Field: Yes
Default Value: 0
Data Name: ACNF.PRO-DATA.AUTH-TIMEOUT

**LOAD TIMEOUT( SECS ) —** Defines the length of time, in seconds, the SPDH module waits for a download response from the terminal before aborting the download. The SPDH module checks all timeout fields for a valid value between 5 and 600 inclusive.

Field Length: 1 to 4 numeric characters
Required Field: Yes
Default Value: 0
Data Name: ACNF.PRO-DATA.LOAD-TIMEOUT

**MAIL TIMEOUT( SECS ) —** Defines the length of time, in seconds, the SPDH module waits for a response from the Mail process before responding to the terminal with a timeout response code. The SPDH module checks all timeout fields for a valid value between 5 and 600 inclusive.

Field Length: 1 to 4 numeric characters
Required Field: Yes
Default Value: 0
Data Name: ACNF.PRO-DATA.MAIL-TIMEOUT

**OLD MSG TIMEOUT( SECS ) —** Defines how old, in seconds, a message can be before it is considered to be an expired message. If this field is set to 0, no expired message processing is performed. Instead, expired messages go through the system like any other message. If this field is set greater than 0, expired message processing is performed based on the value in the OLD MSGS field on this screen.

Expired messages are messages received by the Device Handler module after the terminal has already timed out. This occurs when an SPDH module is stopped and terminals are still sending transaction messages. These messages are placed in the queue and are processed when the SPDH module is restarted.

The SPDH module checks all timeout fields for a valid value between 5 and 600 inclusive.

Field Length: 1 to 4 numeric characters
Required Field: Yes
Default Value: 0
Data Name: ACNF.PRO-DATA.OLDMSG-TIMEOUT

**DPC #0 —** Defines the DPC to which mail sent to DPC #0 is delivered. This field is used by FID W when sending BASE24-mail messages.

Field Length: 1 to 4 numeric characters
Required Field: Yes
Default Value: 0
Data Name: ACNF.PRO-DATA.DPC-0

**DPC #1 —** Defines the DPC to which mail sent to DPC #1 is delivered. This field is used by FID W when sending BASE24-mail messages.

Field Length: 1 to 4 numeric characters
Required Field: Yes
Default Value: 0
Data Name: ACNF.PRO-DATA.DPC-1

**DPC #2 —** Defines the DPC to which mail sent to DPC #2 is delivered. This field is used by FID W when sending BASE24-mail messages.

Field Length: 1 to 4 numeric characters
Required Field: Yes
Default Value: 0
Data Name: ACNF.PRO-DATA.DPC-2

# Processing Record Screen 3

ACI Standard Device Configuration File (ACNF) screen 3 is shown below, followed by its field descriptions.

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  03 OF 03
        TERMINAL GROUP:  0000      RECORD TYPE:  P   RECORD ID:  00
                         KEY THRESHOLDS SCREEN



                           MAC KEY THRESHOLD:    0
                     MAC KEY ERROR THRESHOLD:    0
          CONSECUTIVE MAC KEY ERROR THRESHOLD:    0

                           PIN KEY THRESHOLD:    0
                     PIN KEY ERROR THRESHOLD:    0

                          DATA KEY THRESHOLD:    0
                    DATA KEY ERROR THRESHOLD:    0




 ********************************** BASE24 **********************************
  NEW PAGE:         FILE DESTINATION:         NEW LOGICAL NETWORK ID:
                           F12-HELP
```

**MAC KEY THRESHOLD —** Defines the maximum number of transactions involving a MAC that can be performed before a MAC key exchange is required.

Default Value: 0
Field Length: 1 to 4 numeric characters
Required Field: Yes
Data Name: ANCF.PRO-DATA.MAC-TXN-LMT

**MAC KEY ERROR THRESHOLD —** Defines the maximum number of transactions resulting in MAC errors that can be performed before a MAC key exchange is required.

Default Value: 0
Field Length: 1 to 4 numeric characters
Required Field: Yes
Data Name: ANCF.PRO-DATA.MAC-ERR-LMT

**CONSECUTIVE MAC KEY ERROR THRESHOLD —** Defines the maximum number of consecutive transactions resulting in MAC errors that can be performed before a MAC key exchange is required.

Default Value: 0
Field Length: 1 to 4 numeric characters
Required Field: Yes
Data Name: ANCF.PRO-DATA.CNSC-MAC-ERR-LMT

**PIN KEY THRESHOLD —** Defines the maximum number of transactions involving a PIN that can be performed before a PIN key exchange is required.

Default Value: 0
Field Length: 1 to 4 numeric characters
Required Field: Yes
Data Name: ANCF.PRO-DATA.PIN-TXN-LMT

**PIN KEY ERROR THRESHOLD —** Defines the maximum number of transactions resulting in PIN errors that can be performed before a PIN key exchange is required.

Default Value: 0
Field Length: 1 to 4 numeric characters
Required Field: Yes
Data Name: ANCF.PRO-DATA.PIN-ERR-LMT

**DATA KEY THRESHOLD —** Defines the maximum number of transactions involving data encryption that can be performed before a data key exchange is required.

Default Value: 0
Field Length: 1 to 4 numeric characters
Required Field: Yes
Data Name: ANCF.PRO-DATA.DATA-TXN-LMT

**DATA KEY ERROR THRESHOLD —** Defines the maximum number of transactions resulting in data encryption errors that can be performed before a data key exchange is required.

| | |
|---|---|
| Default Value: | 0 |
| Field Length: | 1 to 4 numeric characters |
| Required Field: | Yes |
| Data Name: | ANCF.PRO-DATA.DATA-ERR-LMT |

# Field Map Records

Field map records allow customers to determine the fields sent in requests and responses. Separate maps are used for request and response messages. Besides allowing different fields in requests and responses, the field maps allow customers to include different fields based on transaction type. For example, different fields can be included in a normal transaction request message than in a preauthorization request message. Field maps also allow customers to specify which fields are required to be encrypted in request messages and which fields are required to be encrypted in response messages.

In addition, the field maps enable you to determine which fields in any request or response message should be verified using message authentication codes (MACs). The SPDH module allows individual message fields to be verified using MACs. You are responsible for identifying these fields on this screen.

Changes to this data can be made online through BASE24 files maintenance. However, the customer and vendor must coordinate on this data because the field maps do not determine the data sent in requests. The purpose of the field map data in the ACNF is to allow the SPDH module to know what to expect from the terminal and how to format the responses to the terminal. Terminals do not receive the field map data. It is the responsibility of the terminal software to determine the data in requests.

## Setting Up Field Map Records

There are six Field Map screens, listing all the possible transactions supported by the SPDH module. Each screen has two pages, one for requests and one for responses.

For each transaction type, the FIDs associated with the optional data fields are listed across the top of the screen. The fields listed on the Field Map screens allow optional data fields to be included in requests and responses for each type of transaction. A detailed explanation of each FID is included in section 4.

The transaction types are listed down the left side of the screen, along with two rows. The first row indicates which optional data fields you want to include in the message and the second row indicates which optional data fields in the message you want to be verified using MACs.

If an FID is to be included in a message, a Y is placed in the one-byte MSG field corresponding to the appropriate FID for the specific transaction type. If an FID is to be included and encrypted in a message, an E (encrypted) is used instead of a Y. If a field is to be verified using MACs, a Y is placed in the one-byte MAC field corresponding to the appropriate FID for the specific transaction type.

No FIDs can be set to a value of E for the download transaction type. The ACNF Field Map screens do not allow the following FIDs to be set to a value of E.

| FID | Description |
|---|---|
| G | Authentication code |
| H | Authentication key |
| I | Data encryption key |
| M | PIN Communications key |
| b | PIN/Customer |
| c | PIN/Supervisor |

Although specific FIDs can be configured to be encrypted, specific SFIDs (subfields) cannot. All SFIDs under FIDs 6, 7, 8, or 9 must be encrypted, or none.

The ACNF Field Map screens are accessed through BASE24 files maintenance much in the same manner as the ACNF Processing Records screen. Once accessed, you can set up the field maps to meet your specific requirements. The following steps describe the procedures to set up the Field Map screens. These procedures assume you have already accessed the ACNF Selection screen. To set up the ACNF Field Map screen for requests, perform the following steps:

1. Enter the appropriate terminal group in the TERMINAL GROUP field on screen 1 of the ACNF (Selection screen).
2. Enter F in the RECORD TYPE field on screen 1 of the ACNF. The F in this field identifies the record type as Field Map.
3. Enter 00 or 01 in the RECORD ID field on screen 1 of the ACNF. 00 in this field identifies the record as the field map used to determine the fields that are required in request messages and those that are required to be encrypted. 01 in this field allows you to configure the fields to include in response messages and those that are required to be encrypted.

4. Press the **F9** key from the ACNF Selection screen to display the first ACNF Field Map screen for requests or responses. The field maps for requests and responses each consist of seven screens. Each screen contains five transaction types. Then press the **F2** key to read the record.

   The first Field Map screen for requests (00 in the RECORD ID field) is shown below.

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 09
         TERMINAL GROUP:  0000      RECORD TYPE:  F   RECORD ID:  00

    TRANSACTION                 FIELD MAP ( REQUEST  )

                ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
 NORM PURCH MSG NYNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNENNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
 PRE AUTH   MSG NYNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNENNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
 PRE AUTH C MSG NYNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNENNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
 PH/MAIL    MSG NYNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNENNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
 MERCH RET  MSG NYNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNENNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN




 ********************************** BASE24 ***********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

The first Field Map screen for responses (01 in the RECORD ID field) is shown below.

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 09
         TERMINAL GROUP:  0000      RECORD TYPE:  F   RECORD ID:  01

    TRANSACTION                FIELD MAP ( RESPONSE )

                ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
 NORM PURCH MSG NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
 PRE AUTH   MSG NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
 PRE AUTH C MSG NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
 PH/MAIL    MSG NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
 MERCH RET  MSG NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
            MAC NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN



 ********************************** BASE24 **********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

The remaining five Field Map request and response screens are formatted in the same manner as screen 02. Each screen displays five transaction types that must be configured separately. Press the **F9** key to access subsequent Field Map screens. The following table identifies the transaction types that correspond to each Field Map screen. Generally, screens 02 through 04 are used to configure financial transactions, screens 05 through 07 are used to configure administrative transactions, screen 8 is used to configure stored value transactions, and screen 9 is used for mobile top-up transactions.

| Field Map Request and Response Screens | |
|---|---|
| **Field Map Screen Number** | **Transaction Type** |
| 02 of 09 | Normal purchase<br>Preauthorization<br>Preauthorization completion<br>Mail or telephone order<br>Merchandise return |

| Field Map Request and Response Screens | |
|---|---|
| **Field Map Screen Number** | **Transaction Type** |
| 03 of 09 | Cash advance<br>Card verification<br>Balance inquiry<br>Purchase with cash back<br>Check verification |
| 04 of 09 | Check guarantee<br>Purchase adjustment<br>Merchandise return adjustment<br>Cash advance adjustment<br>Cash back adjustment |
| 05 of 09 | Logon<br>Logoff<br>Close batch<br>Close shift<br>Close day |
| 06 of 09 | Clerk totals<br>Batch totals<br>Shift totals<br>Day totals<br>Read mail |
| 07 of 09 | Mail delivery<br>Send mail request<br>Download<br>Handshake<br>New key |
| 08 of 09 | Card activation<br>Additional card activation<br>Replenishment<br>Full redemption<br>Reserved—Reserved for future use |
| 09 of 09 | Mobile Top-Up Cash<br>Mobile Top-Up Fund<br>Mobile Top-Up Ref C<br>Mobile Top-Up Ref F<br>Reserved—Reserved for future use |

5. Determine which optional data fields (FIDs) to include for each transaction type. For a list of FIDs and their corresponding field names, press the **F7** key to display the following Field Map Help screen. The Field Map Help screen displays the optional data fields allowed for requests or responses.

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 09
         TERMINAL GROUP:  0000      RECORD TYPE:  F   RECORD ID:  01
       FIELD MAP INFORMATION FOR SCREENS 2 THRU 9 OF FIELD TYPE F
 A- Address          Q- Echo Data         g- Response Dsply    w- Reserved
 B- Amount 1         R- Card Type         h- Sequence Num      x- Reserved
 C- Amount 2         S- Invoice Num       i- Orig. Seq. Num    y- Reserved
 D- Appl Acct Type   T- Orig Invoice Num  j- Drvrs Lic ST      z- Reserved
 E- Appl Acct Num    U- Language Code     k- Term Loc/D.L. DOB 0- AMEX Data
 F- Approval Code    V- Mail/DLL Key      l- Totals/Batch      1- PS2000 Data
 G- Authen. Code     W- Mail/DLL Text     m- Totals/Day        2- Trk1/Customer
 H- Authen. Key      X- ISO Response Code n- Totals/Clerk      3- Trk1/Super
 I- Data Encry. Key  Y- Zip Code          o- Totals/Shift      4- Reserved
 J- Avail Bal        Z- Addr Vrfy Stat    p- Reserved          5- Reserved
 K- Business Date    a- Optional Data     q- Trk2/Customer     6- Product SFIDs
 L- Check Typ/Ctgry  b- PIN/ Customer     r- Trk2/Supervisor   7- Product SFIDs
 M- Comm key         c- PIN/ Supervisor   s- Tran Dscrption    8- Chnl/Dst SFIDs
 N- Customer ID      d- Retailer ID       t- PIN Pad ID        9- Customer SFIDs
 O- Customer ID Typ  e- POS Condition Cde u- Posting Date
 P- Draft Captr Flg  f- Receipt Data      v- Reserved

 ********************************** BASE24 ***********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
 ANY FUNCTION KEY EXCEPT SF9-SF16, F10, OR F16 RETURNS. F10 PRINTS AND RETURNS
```

It may be helpful to print this screen for use as a reference tool when setting up the Field Map screens. To print this screen, perform the following steps:

a. Specify a printer location in the SCREEN PRINTER field located on the BASE24 Logon Menu.

b. Access the ACNF screens.

c. Press the **F7** key to move to the Field Map Help screen.

d. Press the **F10** key. The **F10** key is used to print the screen.

For more information on printing screens, refer to the ***BASE24 CRT Access Manual***.

6. Enter Y, N, or E in the MSG row after each transaction type for each optional data field listed across the top of the screen. Optional data fields are identified across the top of the screen by their corresponding FIDs. A Y indicates the FID is included in requests, an N indicates the FID is not included in requests, an E indicates the FID is included in requests, and it is encrypted under the data encryption key.

7. Determine which optional data fields should be verified using MACs and enter Y or N in the MAC row after each transaction type for each optional data field (FID) listed across the top of the screen. A Y indicates the optional data field should be verified using MACs and an N indicates the optional data field should not be verified using MACs.
8. Verify that the optional data fields set up for each transaction type are correct, and that the appropriate optional data fields are being verified using MACs.
9. Repeat steps 6 through 8 for each of the remaining request screens.
10. Change the value in the RECORD ID field to 01 and repeat steps 6 through 8 for each of the response screens.
11. Press the **F3** key to add the record.
12. Read the data associated with these records by pressing the **F9** key to access the screen. Then press the **F2** key to read the record.

# FID Requirements by Transaction Type

Although much of the information contained in message requests and responses is configurable using the ACNF Field Map screens, some of the optional data fields (FIDs) are required for certain transaction types. These FIDs must be used (i.e., have a setting of Yor E on the Field Map screens) in order for processing to occur for a specific transaction type. The following tables outline the FIDs required in requests and responses for each transaction type. Other optional data fields can also be included, but the FIDs shown in the tables must be present for the specified transaction type. Only the transaction types that have required FIDs are listed.

## Purchases, Preauthorization Purchases, Preauthorization Completions, Merchandise Returns, and Cash Advances

The following table contains the ACNF settings for FIDs and MAC values for purchases, preauthorization purchases, preauthorization completions, merchandise returns, and cash advances.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| B | Amount 1 (PSTM.TRAN.AMT-1) | Y or E | N | N | N |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |

* Either FID q or 2 is required for request and response messages.

## Mail or Telephone Orders

The following table contains the ACNF settings for FIDs and MAC values for mail or telephone order transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| B | Amount 1 (PSTM.TRAN.AMT-1) | Y or E | N | N | N |
| q | Track2/Customer (PSTM.TRACK2) | Y or E | N | N | N |

## Card Verification and Balance Inquiries

The following table contains the ACNF settings for FIDs and MAC values for card verification and balance inquiry transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |

* Either FID q or 2 is required for request and response messages.

## Purchases with Cash Back and Adjustments

The following table contains the ACNF settings for FIDs and MAC values for purchase with cash back, purchase adjustment, merchandise return adjustment, cash advance adjustment, and cash back adjustment transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| B | Amount 1 (PSTM.TRAN.AMT-1) | Y or E | N | N | N |
| C | Amount 2 (PSTM.TRAN.AMT-2) | Y or E | N | N | N |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |

* Either FID q or 2 is required for request and response messages.

## Check Verification

The following table contains the ACNF settings for FIDs and MAC values for check verification transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| L | Check Type | Y or E | N | N | N |
| N | Customer ID | N* | N | N | N |
| O | Customer ID Type | N | N | N | N |

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| j | State Code | N | N | N | N |
| k | Birth Date or Terminal Location | N | N | N | N |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (PSTM.TRACK2) | N* | N | N | N |

* Either FID q or 2 is required if the customer ID type (FID O) is defined as a debit or credit card. Otherwise, FID N contains the customer ID.

## Check Guarantee

The following table contains the ACNF settings for FIDs and MAC values for check guarantee transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| B | Amount 1 (PSTM.TRAN.AMT-1) | Y or E | N | N | N |
| L | Check Type | Y or E | N | N | N |
| N | Customer Id | N* | N | N | N |
| O | Customer Id Type | N | N | N | N |
| j | State Code | N | N | N | N |
| k | Birth Date or Terminal Location | N | N | N | N |

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |

* FID q is required if the customer ID type (FID O) is defined as a debit or credit card. Otherwise, FID N contains the customer ID.

## Clerk Totals

The following table contains the ACNF settings for FIDs and MAC values for clerk totals.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| n | Totals/Clerk | N | N | Y | N |

## Batch Totals

The following table contains the ACNF settings for FIDs and MAC values for batch totals.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| l | Totals/Batch | N | N | Y | N |

## Shift Totals

The following table contains the ACNF settings for FIDs and MAC values for shift totals.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| o | Totals/Shift | N | N | Y | N |

## Day Totals

The following table contains the ACNF settings for FIDs and MAC values for day totals.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| m | Totals/Day | N | N | Y | N |

## Read Mail and Mail Delivered Requests

The following table contains the ACNF settings for FIDs and MAC values for read mail and mail delivered transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| V | Mail/Download Key | Y or E | N | Y | N |

## Send Mail Requests

The following table contains the ACNF settings for FIDs and MAC values for send mail request transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| W | Mail/Download Text | Y or E | N | Y | N |

## Download Request

The following table contains the ACNF settings for FIDs and MAC values for download request transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| V | Mail/Download Key | Y or E* | N | Y* | N |

* FID V is required for downloads except on initial requests.

## Mobile Top-Up Transactions

The following tables contains the ACNF settings for FIDs and MAC values for mobile top-up with cash and mobile top-up with funds transactions.

## Mobile Top-Up with Cash Request

The following table contains the ACNF settings for FIDs and MAC values for mobile top-up with cash transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| B | Amount 1 (PSTM.TRAN.AMT-1) | Y or E | N | N | N |
| R | Card Type | Y or E | N | N | N |
| 7 | Mobile Top-Up Track 2 | Y or E | N | N | N |

## Mobile Top-Up with Funds Request

The following table contains the ACNF settings for FIDs and MAC values for mobile top-up with funds transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| B | Amount 1 (PSTM.TRAN.AMT-1) | Y or E | N | N | N |
| R | Card Type | Y or E | N | N | N |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |
| 7 | Mobile Top-Up Track 2 | Y or E | N | N | N |

* Either FID q or 2 is required for request and response messages.

# Stored Value Transactions

The following tables contains the ACNF settings for FIDs and MAC values for card activation, additional card activation, replenishment, card activation, and full redemption transactions.

## Card Activation

The following table contains the ACNF settings for FIDs and MAC values for stored value card activation transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |

* Either FID q or 2 is required for request and response messages.

## Additional Card Activation

The following table contains the ACNF settings for FIDs and MAC values for stored value additional card activation transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |
| 6 | Stored Value Data | Y or E | N | N | N |

* Either FID q or 2 is required for request and response messages.

## Replenishment

The following table contains the ACNF settings for FIDs and MAC values for stored value replenishment transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| B | Amount 1 (PSTM.TRAN.AMT-1) | Y or E | N | N | N |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |

* Either FID q or 2 is required for request and response messages.

## Full Redemption

The following table contains the ACNF settings for FIDs and MAC values for stored value full redemption transactions.

| Required FIDs | | | | | |
|---|---|---|---|---|---|
| **FID** | **Element Name** | **Request** | | **Response** | |
| | | **MSG** | **MAC** | **MSG** | **MAC** |
| q | Track2/Customer (PSTM.TRACK2) | N* | N | N | N |
| 2 | Track1/Customer (Track 1 token) | N* | N | N | N |

* Either FID q or 2 is required for request and response messages.

# Download Records

The SPDH module supports full and partial downloads to terminals. The information to be sent to terminals during a download is configured using the ACNF. Download records 00, 01, and 02 allow customers to download up to 26 fields of 40 characters each. The data to be included is left to the discretion of the customer. Download records 03, 04, 05, 07, 08, 09, 10, and 11 contain card prefix range processing parameters for up to 30 card prefix ranges. Download record 06 is used to identify which fields from the BASE24-pos Terminal Data files (PTD) and POS Retailer Definition File (PRDF) are downloaded to the terminal. Additional information on download records is included in section 5.

## Setting Up Download Records 00, 01, and 02

Download records 00, 01, and 02 allow customers to configure download data to be sent to the terminal. This information can consist of any information the terminal owner and operator find useful in performing day-to-day functions or that the terminal vendor requires. All of these records are optional.

The data elements are identified with DIDs A through Z. DIDs A through J are stored in record 00. DIDs K through T are stored in record 01. DIDs U through Z are stored in record 02.

The data elements in records 00, 01, and 02 are sent to the terminal only if they are requested by the terminal and if they contain data.

To access ACNF download records 00, 01, 02, perform the following steps:

1. Enter the appropriate terminal group in the TERMINAL GROUP field on screen 1 of the ACNF (Selection screen).
2. Enter D in the RECORD TYPE field on screen 1 of the ACNF.
3. Enter 00, 01, or 02 in the RECORD ID field on screen 1 of the ACNF.

   Record 00 is accessed by placing D in the RECORD TYPE field and 00 in the RECORD ID field on the ACNF Selection screen.

   Record 01 is accessed by placing D in the RECORD TYPE field and 01 in the RECORD ID field on the ACNF Selection screen.

   Record 02 is accessed by placing D in the RECORD TYPE field and 02 in the RECORD ID field on the ACNF Selection screen.

4. From screen 1 of the ACNF, press the **F9** key to display the first ACNF Download Data screen. The three possible screens that can be displayed are shown below and on the following page. The DIDs are listed in a column on the left side of the screens, followed by fields of 40 characters each. Data can be entered in any of the 40-character fields.

   If the RECORD ID field is set to 00, the following screen is displayed:

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 02
         TERMINAL GROUP:  0000      RECORD TYPE:  D   RECORD ID:  00

                     DOWNLOAD DATA RECORD

             A
             B
             C
             D
             E
             F
             G
             H
             I
             J




 ********************************** BASE24 **********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                         F12-HELP
```

If the RECORD ID field is set to 01, the following screen is displayed:

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 02
         TERMINAL GROUP:  0000      RECORD TYPE:  D   RECORD ID:  01

                     DOWNLOAD DATA RECORD

             K
             L
             M
             N
             O
             P
             Q
             R
             S
             T




 ********************************** BASE24 ***********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                         F12-HELP
```

If the RECORD ID field is set to 02, the following screen is displayed:

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 02
         TERMINAL GROUP:  0000      RECORD TYPE:  D   RECORD ID:  02

                     DOWNLOAD DATA RECORD

             U
             V
             W
             X
             Y
             Z









 ********************************** BASE24 ***********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                         F12-HELP
```

5. Determine the download data to be included in DIDs A through Z.
6. Enter the download data in the appropriate DIDs in fields A through J on the first download data record screen.
7. Verify the data you entered is correct.
8. Press the **F3** key to add the record.
9. Press the **F9** key to move to the next download record screen, which contains download data for DIDs K through T.
10. Repeat steps 6 through 8 to add download data for DIDs K through T.
11. Press the **F9** key to move to the next download record screen, which contains download data for DIDs U through Z.
12. Repeat steps 6 through 8 to add download data for DIDs U through Z.

## Setting up Download Records 03, 04, 05, 07, 08, 09, 10, and 11

Download records 03, 04, 05, 07, 08, 09, 10, and 11 contain card prefix range processing parameters. Up to 30 ranges can be defined within this set of records. Card prefix information is defined by the terminal owner and operator.

**Note:** For customers who do not want to upgrade their firmware to support 30 card prefixes (Release 5.0 and later releases), only records 03, 04, and 05 are used. These records support up to 10 card prefixes.

Records 03, 04, 07, 08, 09, 10, and 11 can contain parameters for four separate card prefix ranges. Record 05 can contain parameters for two separate card prefix ranges.

To access ACNF download records 03, 04, 05, 07, 08, 09, 10, and 11, perform the following steps:

1. Enter the appropriate terminal group in the TERMINAL GROUP field on screen 1 of the ACNF (Selection screen).
2. Enter D in the RECORD TYPE field on screen 1 of the ACNF.

3. Enter one of the following in the RECORD ID field on screen 1 of the ACNF:

   Record 03 is associated with two screens accessed by placing 03 in the RECORD ID field on the ACNF Selection screen.

   Record 04 is associated with two screens accessed by placing 04 in the RECORD ID field on the ACNF Selection screen.

   Record 05 is associated with one screen accessed by placing 05 in the RECORD ID field on the ACNF Selection screen.

   Record 07 is associated with two screens accessed by placing 07 in the RECORD ID field on the ACNF Selection screen.

   Record 08 is associated with two screens accessed by placing 08 in the RECORD ID field on the ACNF Selection screen.

   Record 09 is associated with two screens accessed by placing 09 in the RECORD ID field on the ACNF Selection screen.

   Record 10 is associated with two screens accessed by placing 10 in the RECORD ID field on the ACNF Selection screen.

   Record 11 is associated with two screens accessed by placing 11 in the RECORD ID field on the ACNF Selection screen.

4. Enter the following fields for each card prefix range to identify the parameters:

   LOW PREFIX
   HIGH PREFIX
   NETWORK PH
   NETWORK BKUP PH
   REFERRAL PH
   RETAILER ID
   DRAFT CAP FLG
   TOTAL FLG
   PIN VALIDATION FLAG
   RECEIPT FLG
   MOD-CK FLG
   PAN FRAUD CK FLG
   USER DEF

   For a description of the PTD fields, refer to the topic “Download Records 03, 04, 05, 07, 08, 09, 10, and 11 Screens” where the fields displayed on the screens are described.

5. Press the **F3** key to add the record.

# Download Records 03, 04, 05, 07, 08, 09, 10, and 11 Screens

ACI Standard Device Configuration File (ACNF) screens for download record 03 are shown below, followed by their field descriptions. The screens for records 04, 05, 07, 08, 09, 10, and 11 are formatted in the exact same manner as record 03 and, therefore, additional screen examples are unnecessary to describe these records. Download record 03 defines information for up to four card prefix ranges. Download records 04, 05, 07, 08, 09, 10, and 11 also have two screens. Download record 05 does not have the second screen because it only defines information for two card prefix ranges. The first screen associated with download record 03 is formatted as follows:

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 03
         TERMINAL GROUP:  0000      RECORD TYPE:  D   RECORD ID:  03
                       DOWNLOAD DATA RECORD

 LOW PREFIX(1)                          HIGH PREFIX(1)
 NETWORK PH(1)                        NETWORK BKUP PH(1)
 REFERRAL PH(1)                         RETAILER ID(1)
 DRAFT CAP FLG(1)       TOTAL FLG(1)        PIN VALIDATION FLAG(1)
 RECEIPT FLG(1)         MOD-CK FLG(1)          PAN FRAUD CK FLG(1)
 USER DEF(1)

 LOW PREFIX(2)                          HIGH PREFIX(2)
 NETWORK PH(2)                        NETWORK BKUP PH(2)
 REFERRAL PH(2)                         RETAILER ID(2)
 DRAFT CAP FLG(2)       TOTAL FLG(2)          PIN VALIDATION FLG(2)
 RECEIPT FLG(2)         MOD-CK FLG(2)          PAN FRAUD CK FLG(2)
 USER DEF(2)



 ********************************** BASE24 **********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                         F12-HELP
```

The second screen associated with download record 03 is formatted as follows:

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  03 OF 03
         TERMINAL GROUP:  0000      RECORD TYPE:  D   RECORD ID:  03
                       DOWNLOAD DATA RECORD

 LOW PREFIX(3)                          HIGH PREFIX(3)
 NETWORK PH(3)                        NETWORK BKUP PH(3)
 REFERRAL PH(3)                         RETAILER ID(3)
 DRAFT CAP FLG(3)       TOTAL FLG(3)           PIN VALIDATION FLG(3)
 RECEIPT FLG(3)         MOD-CK FLG(3)          PAN FRAUD CK FLG(3)
 USER DEF(3)

 LOW PREFIX(4)                          HIGH PREFIX(4)
 NETWORK PH(4)                        NETWORK BKUP PH(4)
 REFERRAL PH(4)                         RETAILER ID(4)
 DRAFT CAP FLG(4)       TOTAL FLG(4)           PIN VALIDATION FLG(4)
 RECEIPT FLG(4)         MOD-CK FLG(4)          PAN FRAUD CK FLG(4)
 USER DEF(4)



 ********************************** BASE24 **********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                         F12-HELP
```

**LOW PREFIX(1) —** The lowest card prefix value defined by this range. Any card prefix values greater than or equal to the value specified in this field, and less than or equal to the value specified in the HIGH PREFIX(1) field are included within the card prefix range defined. The (1) indicates the information set up defines the first card prefix range for download record 03. A (2) indicates the information set up defines the second card prefix range for download record 03, and so on. Download record 03 can define information for up to four card prefix ranges.

| | |
|---|---|
| Field Length: | 1 to 11 alphanumeric characters |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ACNF.FORMAT2.FLD.INFO |

**HIGH PREFIX(1) —** The highest card prefix range defined by this range. Any card prefix range values equal to or less than the value specified in this field, and greater than or equal to the value specified in the LOW PREFIX(1) field are included within the card prefix range defined.

Field Length: 1 to 11 alphanumeric characters
Required Field: Yes
Default Value: No default value
Data Name: ACNF.FORMAT2.FLD.INFO

**NETWORK PH(1) —** The telephone number of the primary network associated with transactions initiated with cards in the defined card prefix range.

Field Length: 1 to 20 alphanumeric characters
Required Field: Yes
Default Value: No default value
Data Name: ACNF.FORMAT2.FLD.INFO

**NETWORK BKUP PH(1) —** The telephone number of the backup network associated with transactions initiated with cards in the defined card prefix range.

Field Length: 1 to 20 alphanumeric characters
Required Field: Yes
Default Value: No default value
Data Name: ACNF.FORMAT2.FLD.INFO

**REFERRAL PH(1) —** The telephone number of the network designated to receive referrals concerning transactions initiated with cards in the defined card prefix range.

Field Length: 1 to 20 alphanumeric characters
Required Field: Yes
Default Value: No default value
Data Name: ACNF.FORMAT2.FLD.INFO

**RETAILER ID(1) —** An identifier for the retailer associated with this card prefix range. The retailer ID is assigned by MasterCard, Visa, American Express, or another card issuer.

Field Length: 1 to 12 alphanumeric characters
Required Field: Yes
Default Value: No default value
Data Name: ACNF.FORMAT2.FLD.INFO

**DRAFT CAP FLG(1) —** A flag indicating whether cards within the defined range use draft capture processing. This field is taken from the ACNF and overrides the value in the TP field on PTD screens 8, 9, or 10 for the card prefix, if the PTD field is set so the terminal determines the draft capture mode for each transaction. The following valid values can be entered in this field:

0 = Authorize only
1 = Authorize and capture

This field remains blank if the corresponding prefix range is not defined.

Field Length: 1 alphanumeric character
Required Field: Yes
Default Value: No default value
Data Name: ACNF.FORMAT2.FLD.INFO

**TOTAL FLG(1) —** A flag indicating whether the terminal maintains totals for the card prefix range. The terminal must have the capability of maintaining totals if the value entered in this field is 1. The following valid values can be entered in this field:

0 = No totals for card prefix range
1 = Totals for card prefix range

This field remains blank if the corresponding prefix range is not defined.

Field Length: 1 alphanumeric character
Required Field: Yes
Default Value: No default value
Data Name: ACNF.FORMAT2.FLD.INFO

**PIN VALIDATION FLAG(1) —** A flag indicating whether cards within the card prefix range require PINs. The following valid values can be entered in this field:

0 = PINs not required for card prefix range
1 = PINs required for card prefix range

This field remains blank if the corresponding prefix range is not defined.

| | |
|---|---|
| Field Length: | 1 alphanumeric character |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ACNF.FORMAT2.FLD.INFO |

**RECEIPT FLG(1) —** A flag indicating whether transactions initiated with cards within the card prefix range require a receipt. The following valid values can be entered in this field:

0 = Receipts not required for card prefix range
1 = Receipts required for card prefix range

This field remains blank if the corresponding prefix range is not defined.

| | |
|---|---|
| Field Length: | 1 alphanumeric character |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ACNF.FORMAT2.FLD.INFO |

**MOD-CK FLG(1) —** A flag indicating whether MOD-10 checks are used for cards within the card prefix range. The following valid values can be entered in this field:

0 = No terminal MOD-10 checks for card prefix range
1 = Terminal MOD-10 checks for card prefix range

This field remains blank if the corresponding prefix range is not defined.

| | |
|---|---|
| Field Length: | 1 alphanumeric character |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ACNF.FORMAT2.FLD.INFO |

**PAN FRAUD CK FLG(1) —** A flag indicating whether the terminal performs PAN fraud checks on cards within the card prefix range. If PAN fraud checks are performed, this field also indicates the type. The following valid values can be entered in this field:

0 = No fraud check for card prefix range
1 = Visually check last four digits of the PAN
2 = Visually check Track 2

This field remains blank if the corresponding prefix range is not defined.

| | |
|---|---|
| Field Length: | 1 alphanumeric character |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ACNF.FORMAT2.FLD.INFO |

**USER DEF(1) —** Up to eight characters of user-defined data can be entered in this field. This data is used by the terminal for cards within the card prefix range.

| | |
|---|---|
| Field Length: | 1 to 8 alphanumeric characters |
| Required Field: | No |
| Default Value: | No default value |
| Data Name: | ACNF.FORMAT2.FLD.FILLER |

## Setting Up Download Record 06

The data elements in download record 06 consist of flags corresponding to fields in the BASE24-pos Terminal Data files (PTD) and POS Retailer Definition File (PRDF). These flags are used to determine the PTD fields to be downloaded to the terminal. Record 06 can store up to 26 download flags. DIDs a through z identify the 26 data elements that can be included in record 06, although only 14 of the data elements are currently supported. The data elements in download record 06 are sent to the terminal only if they are requested by the terminal and if the element is to be taken from the PTD.

To access ACNF download record 06, perform the following steps:

1. Enter the appropriate terminal group in the TERMINAL GROUP field on screen 1 of the ACNF (Selection screen).
2. Enter D in the RECORD TYPE field on screen 1 of the ACNF.

3. Enter 06 in the RECORD ID field on screen 1 of the ACNF.
4. Enter Y for each PTD field to be included in download record 06. Enter N for each PTD field to be excluded from download record 06.
5. Press the **F3** key to add the record.

## Download Record 06 Screen

ACI Standard Device Configuration File (ACNF) screens for download record 06 is shown below. Refer to section 5 for descriptions of the PTD fields shown on the screen.

```
BASE24-POS   SPDH ACNF MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 02
         TERMINAL GROUP:  0000      RECORD TYPE:  D   RECORD ID:  06

                     DOWNLOAD DATA RECORD

  a TERM-NAM-LOC         N  b TERM-CITY-ST        N  c TERM-OWNER-NAM      N
  d RESERVED             N  e RESERVED            N  f SRV-REP             N
  g P-KEY                N  h A-KEY(DEV-DEP-DATA) N  i PIN-PAD-CHAR        N
  j DATA-ENCRYPTION-KEY  N  k SRV                 N  l FLOOR LIMITS        N
  m RESERVED             N  n RESERVED            N  o RESERVED            N
  p ALLOWED-TRANS        N  q RETAILER-ID         N  r MERCHANT NAME       N
  s REFERRAL PHONE #     N  t RESERVED            N  u RESERVED            N
  v RESERVED             N  w RESERVED            N  x RESERVED            N
  y RESERVED             N  z RESERVED            N

                     ( VALID ENTRIES  ARE Y OR N )



********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                        F12-HELP
```

## DOWNLOAD DATA RECORD

The DOWNLOAD DATA RECORD fields correspond to DIDs a through z. Each occurrence of the field indicates whether the corresponding DID is to be included in record 06.

Field Length: 1 alphabetic character
Occurs: 26 times

Required Field: No
Default Value: N
Data Name: ACNF.FORMAT3.FLD.PRESENT

# ACI Standard Device Response File

The ACI Standard Device Response File (ARSP) allows customers to alter the responses sent to the terminal, depending on the transaction being sent and the language being used.

The ARSP contains a map linking BASE24 response codes to response code descriptions. This map is one record within the ARSP. Up to three more records can be added to the file. These additional records can contain the wording to send a response explanation in different languages. The wording can be sent to the terminal in the response and used for printing receipts and displaying messages to the terminal.

**Note:** The descriptions in the ARSP can be overridden by a corresponding record in the Response Code Description File (RCDF). Refer to the ***BASE24-pos Files Maintenance Manual*** for information about the RCDF.

# Selecting ARSP Records

The first screen associated with the ARSP allows you to choose the record type. This screen is your main menu for selecting the appropriate ARSP record to configure. The ARSP contains various types of information required to configure responses sent from the SPDH module to the terminal.

There are four types, or sets, of screens associated with the ARSP. These are used to access five records that can be stored in the ARSP. These screen types are as follows:

- Selection Screen
- Response Display Map Screens
- Language Display Screens
- Transaction Description Screens

Each of the above screen groups are described on the following pages.

## Accessing the Selection Screen

The ARSP Selection screen is your primary link to accessing ARSP records. To access the ARSP Selection screen, perform the following steps:

1. Enter ARSP in the FILE DESTINATION field at the bottom of the CRT access screen. From a menu, press the **F1** key to display ARSP screen 1. From a file screen, press the **F16** key to display ARSP screen 1.
2. Enter the appropriate values in the TERMINAL GROUP and RECORD NUMBER fields on ARSP screen 1, based on the parameters you want to configure (i.e., responses, language display, or transaction descriptions) and the terminal group to which you want the parameters to apply.

   The Response Display Map record, accessed by placing 00 in this field, links each response code to a response display.

   The three language display records, accessed by placing 01, 02, or 03 in this field, can contain responses in different languages.

   The Transaction Description record, accessed by placing 04 in this field, enables customers to determine the name of transactions.
3. Once values are entered in these fields, press the **F9** key to access the record.
4. Configure the selected record and press the **F3** key to add the record.

## Selection Screen

ACI Standard Device Response File (ARSP) screen 1 is shown below, followed by descriptions of its fields.

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  01 OF 06
    TERMINAL GROUP:                             RECORD NUMBER:  00

                           DIRECTIONS:


     PLEASE ENTER THE DESIRED TERMINAL GROUP AND RECORD NUM.


                PLEASE PRESS DESIRED FUNCTION KEY TO CONTINUE


     REC NUM 00 =  RESPONSES
     REC NUM 01 =  DISPLAY LANGUAGE 1
     REC NUM 02 =  DISPLAY LANGUAGE 2
     REC NUM 03 =  DISPLAY LANGUAGE 3
     REC NUM 04 =  DESCRIPTIONS OF TRANSACTIONS


 ********************************** BASE24 **********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                           F12-HELP
```

**TERMINAL GROUP —** The terminal group identifies the terminal or group of terminals to which the configuration information in the ARSP applies. The terminal group combined with the RECORD NUMBER field provides the primary key into the ARSP. The configuration information is used by all terminals that have a value corresponding to this field in the TERMINAL GROUP field on POS Terminal Data files (PTD) screen 1.

| | |
|---|---|
| Field Length: | 4 alphanumeric characters |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ARSP.PRI-KEY.TERM-GRP |

**RECORD NUMBER —** The number associated with the record to be accessed. The value entered in this field determines the type of screens displayed. Valid values are as follows:

00 = Response Display Map
01 = Display Table 1
02 = Display Table 2
03 = Display Table 3
04 = Transaction Description

| | |
|---|---|
| Field Length: | 2 numeric characters |
| Required Field: | Yes |
| Default Value: | 00 |
| Data Name: | ARSP.PRI-KEY.REC-NUM |

# Response Display Map Records

The ARSP Response Display Map record allows you to provide a text description for the response code contained in the standard message header. The Response Display Map screens are used in conjunction with the Language Display records (described later) to provide customized response code descriptions in up to three languages. The Response Display Map screens link response codes to response code descriptions contained in the Language Display records. For example, you may prefer certain wording for a response code or require response code descriptions in different languages. Records can be entered for all three languages.

The main purpose of the Response Display Map screens is to provide an index into the Language Display screens. The Response Display Map record consists of four screens that list the BASE24-pos response codes and their corresponding descriptions as defined in the POS Standard Internal Message (PSTM). From these screens, you can enter an index number for each response code. The index number corresponds to text contained in the Language Display records.

**Note:** The descriptions in the ARSP can be overridden by a corresponding record in the Response Code Description File (RCDF). Refer to the ***BASE24-pos Files Maintenance Manual*** for information about the RCDF.

## SPDH Module Response Code Processing

Two FIDs control the ARSP Response Display Map records; FID g (Response Display) and FID U (Language Code). FID g must be configured in the ACI Standard Configuration File (ACNF) for inclusion in messages if ARSP Response Display Map records are used. FID g contains up to 48 characters of alphanumeric text that you define in this ARSP record to describe the response code contained in the standard message header. If FID g is configured in the ACNF for a particular transaction type, a description corresponding to a specific response code contained in the standard message header is included in the message.

If FID g is not configured in the ACNF for a particular transaction type, no response code description is included in the message. You can determine whether to include FID g in messages for each transaction type when configuring the Field Map screens in the ACNF. For more information on setting up the ACNF Field Map screens, refer to the ACNF documentation presented earlier in this section.

FID U is used to determine which of the three language displays to use for responses. If FID U is included in the request, this language code overrides the default language code defined in the BASE24-pos Terminal Data files (PTD) and

is echoed in responses. If FID U is not included in the request, the value from the LANGUAGE ID field on PTD screen 1 identifies the language display used for the response.

## Setting Up Response Display Map Record 00

Response Display Map record 00 consists of five screens that list all of the BASE24-pos response codes and their corresponding descriptions as defined in the PSTM. From these screens, you can enter a number for each response code. This number corresponds to an index entry in the Language Display record. The index entry in the Language Display record contains text describing the response code. The SPDH module determines which Language Display record to use based on the language code it receives from the terminal in the request. If no language code is contained in the request, the SPDH module uses the default language code defined in the LANGUAGE ID field on PTD screen 1. To access and set up the ARSP Response Display Map record 00, perform the following steps:

1. Check the ACNF Field Map screens to ensure that FID g (Response Display) has been configured to be included in messages for the transaction types you want. If FID g is not configured for the transaction types you want, add it to the configuration using the steps described in the "Setting Up Field Map Records" topic documented earlier in this section.
2. Enter ARSP in the FILE DESTINATION field at the bottom of the ACNF screen and press the **F16** key to access the ARSP Selection screen.
3. Enter the appropriate value in the TERMINAL GROUP field, based on the terminal group to which you want the ARSP parameters to apply.
4. Enter 00 in the RECORD NUMBER field on the ARSP Selection screen and press the **F9** key to access the first screen associated the ARSP Response Display Map record.

   Record 00 consists of five screens of response codes. Continue to press the **F9** key to access subsequent screens.

   Each screen associated with record 00 is divided into two columns, each containing three fields. The first field in each column contains the BASE24-pos standard response code number. Response codes are listed on the screens in numeric order from left to right across the screen.

   The second field in each column contains the standard BASE24-pos response code description that corresponds to the response code number identified in the first field. The response code number and description are taken from the COBNAMES table.

The third field in each column is the only field on the screens that you can alter. This field contains the number used to index the response code description entered in the Language Display Record and placed in FID g of the response by the SPDH module.

5. Select the response code you want to index by pressing the **Tab** key until you are positioned directly to the right of the response code on the screen.
6. Enter a two digit number to index the Language Display record.
7. Repeat the previous two steps until you have completed indexing the response codes on one screen.
8. Verify the response codes are indexed as you want them on the screen, make any corrections, and move to the next screen by pressing the **F9** key. The **F9** key is used to access subsequent pages associated with the record.

   **Note:** The ARSP Response Display record consists of five screens, each containing response codes. You must enter information on each screen for each response code you want to index.
9. Repeat the procedures described above on each ARSP Response Display screen for each response code you want to index.
10. When you have completed indexing the response codes on all of the ARSP Response Display screens, press the **F3** key to add the record.
11. Press the **F2** key to read the record to verify it has been added correctly.

## Response Display Map Screen

ACI Standard Device Response File (ARSP) Response Display Map screens are shown below. Since the fields on the screens associated with the Response Display Map do not have field names, the data on the screens is described according to the column in which it is located. The screens consist of two sets of three columns each. The first column in each set contains a three-digit number. The second column in each set contains a description of the value in the first column. The third column in each set contains a two-digit number. The screens associated with the Response Display Map are shown on the following pages in the sequence they are displayed.

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 06
   TERMINAL GROUP:                              RECORD NUMBER:  00

  000 APPROVED(BALANCES)             00    001 APPROVED(NO BALANCES)       00
  002 APPROVED(COUNTRY CLUB)         00    003 APPROVED(MAYBE MORE ID)     00
  004 APPROVED(PEND. ID. SIGN)       00    005 APPROVED(BLIND)             00
  006 APPROVED(VIP)                  00    007 APPROVED(ADMIN TRAN)        00
  008 APPROVED(NATL NEG HIT OK)      00    009 APPROVED(COMMERCIAL CRD)    00
  010 APPROVED(LESSER AMOUNT)        00    050 DECLINED                    00
  051 EXPIRED CRD                    00    052 PIN TRIES EXCEEDED          00
  053 NO SHARING W/USER              00    054 NO ATALLA BOX               00
  055 INVALID TRANSACTION            00    056 TRAN NOT SUPPORTED BY FI    00
  057 LOST OR STOLEN CARD            00    058 INVALID CARD STATUS         00
  059 INQUIRY ONLY                   00    060 ACCT NOT ON CAF             00
  061 PBF REC NOT FOUND              00    062 PBF UPDATE ERROR            00
  063 INVALID AUTH TYPE IN IDF       00    064 BAD TRACK INFO              00
  065 ADJ NOT ALLOWED IN IDF         00    066 INVALID CCD ADV INCREMENT   00
  067 INVALID TRAN DATE              00    068 TLF ERROR                   00


********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  03 OF 06
   TERMINAL GROUP:                              RECORD NUMBER:  00

  069 BAD MSG IN PSTM                00    O70 NO IDF                      00
  071 INVALID ROUT TO AUTH           00    072 CARD ON NATL NEG            00
  073 INVLD ROUT TO AUTH-SER FLD     00    074 UNABLE TO AUTHORIZE         00
  075 INVALID PAN LENGTH             00    076 INSUF FNDS IN PBF           00
  077 PRE-AUTH FULL                  00    078 DUP TRAN RECVD AND DROPPED  00
  079 MAX ONLIN REFND REACHED        00    080 MAX OFFLIN REFND REACHED    00
  081 MAX CRED % REFND REACHED       00    082 MAX NUM TIMES USED          00
  083 MAX REFND CRED REACHED         00    084 CUST SELECT NEG REASN       00
  085 INQ NOT ALLOWED-NO BAL         00    086 OVER FLOOR LIMIT            00
  087 MAX NUM REFND REACHED          00    088 PLACE CALL                  00
  089 CAF STAT 0-9                   00    090 REFERRAL FILE FULL          00
  091 NEG FILE PROBLEM               00    092 ADV LESS THAN MINIMUM       00
  093 DELINQUENT                     00    094 OVER TBL LIMIT              00
  095 AMT OVER MAX                   00    096 PIN REQUIRED                00
  097 MOD 10 CHECK                   00    098 DECLINE FORCE POST          00


********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  04 OF 06
   TERMINAL GROUP:                              RECORD NUMBER:  00

  099 BAD PBF                        00    100 UNABLE TO PROCESS TRAN      00
  101 UNABLE TO AUTHORIZE-CALL       00    102 CALL                        00
  103 NEG FILE ERROR                 00    104 CAF FILE PROBLEM            00
  105 CARD NOT SUPPORTED             00    106 AMT OVER MAX                00
  107 OVER DAILY LIMIT               00    108 CAPF NOT FOUND              00
  109 ADV LESS THAN MIN              00    110 NUM TIMES USED              00
  111 DELINQUENT                     00    112 OVER LIMIT TBL              00
  113 TIMEOUT                        00    115 PTLF/PRF FILE PROBLEM       00
  120 UAF FILE PROBLEM               00    121 ADMIN FILE PROBLEM          00
  122 UNABLE TO VERIFY PIN           00    130 ARQC REFERRAL               00
  131 CVR REFERRAL                   00    132 TVR REFERRAL                00
  133 RSN ONLINE REFERRAL            00    134 FALLBACK REFERRAL           00
  150 MERCHANT NOT ON FILE           00        RESERVED FOR FUTURE USE     00
      RESERVED FOR FUTURE USE        00        RESERVED FOR FUTURE USE     00
      RESERVED FOR FUTURE USE        00        RESERVED FOR FUTURE USE     00


********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  05 OF 06
   TERMINAL GROUP:                              RECORD NUMBER:  00

  200 INVALID ACCT                   00    201 INCORRECT PIN               00
  202 CASH ADV LESS THAN MIN         00    203 ADMIN CARD NEEDED           00
  204 ENTER LESSER AMT               00    205 INVALID CREDIT CARD ADV AMT 00
  206 CAF NOT FOUND                  00    207 INVALID TRAN DATE           00
  208 INVALID EXPIRATION DATE        00    209 INVALID TRAN CODE           00
      RESERVED FOR FUTURE USE        00    251 ENTER LESSER CASH BACK AMT  00
  400 ARQC FAILURE                   00    401 HSM PARAMETER ERROR         00
  402 HSM FAILURE                    00    403 KEYI NOT FOUND              00
  404 ATC CHECK FAILURE              00    405 CVR DECLINE                 00
  406 TVR DECLINE                    00    407 RSN ONLINE DECLINE          00
  408 FALLBACK DECLINE               00        RESERVED FOR FUTURE USE     00
  800 FORMAT ERROR                   00    801 INVALID DATA                00
  802 INVALID CLERK NUM              00    809 INVALID CLOSE TRANSACTION   00
  810 TRANSACTION TIMEOUT            00    811 SYSTEM ERROR                00
  820 INVALID TERM ID                00    821 INVALID RESP LENGTH         00


********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  06 OF 06
   TERMINAL GROUP:                                RECORD NUMBER:  00

  870 OK - DELIVERED                00    871 OK - STORED                 00
  877 INVALID PIN BLOCK             00    878 INCORRECT PIN ERROR         00
  880 OK - NO MORE DATA             00    881 OK - MORE DATA EXISTS       00
  898 AUTHENTICATE ERR- INVALD MAC  00    899 SEQ ERR / BEGIN RESYNC      00
  900 PIN TRIES EXCEEDED            00    901 EXPIRED CRD                 00
  902 NEG CAPTURE CODE              00    903 CAF STATUS 3                00
  904 ADV LESS THAN MIN             00    905 NUM TIMES USED              00
  906 DELINQUENT                    00    907 OVER LIMIT TBL              00
  908 AMT OVER MAX                  00    909 CAPTURE                     00
  910 ARQC FAILURE                  00    911 CVR CAPTURE                 00
  912 TVR CAPTURE                   00    950 ADMIN CRD NOT ON FILE       00
  951 ADMIN CRD NOT ALLOWED         00    952 ADMIN REQ / IN WINDOW       00
  953 ADMIN REQ / OUT OF WINDOW     00    954 ADMIN REQ / ANYTIME         00
      RESERVED FOR FUTURE USE       00        RESERVED FOR FUTURE USE     00
      RESERVED FOR FUTURE USE       00        RESERVED FOR FUTURE USE     00


********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

**<rc> —** The values in this column are BASE24-pos response codes and cannot be altered by the user. All BASE24-pos response codes are listed in the record. The screens on which particular responses codes are listed are noted below. The screen numbers are in the upper-right corner of the screen.

Several of the spaces in the response code column are blank, indicating that numbers have been reserved for future use.

The BASE24-pos response codes listed here correspond to the RESP-CDE field in the POS Standard Internal Message (PSTM). Additional information about the response codes can be found in the ***BASE24-pos Transaction Processing Manual***.

| Response Codes | Screen Number |
|---|---|
| 000 through 069 | 2 |
| 070 through 099 | 3 |
| 100 through 150 | 4 |
| 200 through 821 | 5 |
| 870 through 954 | 6 |

Field Length: System-protected
Occurs: Up to 30 times per screen

**<description> —** The values in this column are descriptions of the BASE24-pos response codes listed in the first column. These descriptions cannot be modified by the user.

Field Length: System-protected
Occurs: Up to 30 times per screen

**<ld> —** The values in this column relate to the various language displays. The numbers in this column are used when determining the language to be included in the response.

Field Length: 2 numeric characters
Required Field: Yes
Default Value: 00
Data Name: ARSP.RESP.TBL

# Language Display Records

The ARSP Language Display records contain response code descriptions that correspond to index entries contained in the Response Display Map records. There are three separate Language Display records, each consisting of four screens. Each Language Display record is organized by index entry numbers of 0 through 39. You can enter response code descriptions following each one of these index entry numbers and you can enter response code descriptions in up to three different languages in these records. If you enter a response code description following one of the index entry numbers, you must link the description to the corresponding response code in the Response Display Map record.

**Note:** The descriptions in the ARSP can be overridden by a corresponding record in the Response Code Description File (RCDF). Refer to the ***BASE24-pos Files Maintenance Manual*** for information about the RCDF.

## SPDH Module Language Code Processing

FID U controls the language code used to access the appropriate Language Display record. FID U must be configured in the ACNF if it is to be included in messages. FID U is used to determine which of the three language displays to use for responses. If FID U is included in the request from the terminal, its value overrides the default language code defined in the BASE24-pos Terminal Data files (PTD) and is echoed in responses. If FID U is not included in the request, the SPDH module uses the default language code set in the LANGUAGE ID field on PTD screen 1 to determine which record to access.

The SPDH module reads the language code it receives in the request from the terminal. The SPDH module uses FID U in the request to determine which Language Display record to access. If FID U is absent, the SPDH module uses the default language code set in the LANGUAGE ID field on PTD screen 1 to determine which record to access. The language codes that can be included in the request from the terminal are as follows:

0 = Language Display record 1
1 = Language Display record 2
2 = Language Display record 3

The SPDH module reads the Response Display Map records and determines which response codes have corresponding response code descriptions indexed in the Language Display records. The SPDH module accesses the appropriate Language

Display record, matches the correct response code description to the response code number it is indexed to in the Response Display record, and sends the information to the terminal in the response.

## Setting Up Language Display Records

The ARSP Language Display records consists of three separate records, each containing four screens. Record 01 retrieves the record for Language Display Table 1 responses; record 02 retrieves the record for Language Display Table 2 responses; and record 03 retrieves the record for Language Display Table 3 responses.

To access the ARSP Language Display screen, perform the following steps:

1. Check the ACNF Field Map screens to ensure that FID U (Language Code) has been configured to be included in messages for the transaction types you want. If FID U is not configured for the transaction types you want, add it to the configuration using the steps described in the "Setting Up Field Map Records" topic documented earlier in this section. An alternative to this step is to use the default language code set in the LANGUAGE ID field on PTD screen 1. If the default language code is being used, skip this step and move to step 2.
2. Enter ARSP in the FILE DESTINATION field at the bottom of the ACNF screen and press the **F16** key to access the ARSP Selection screen.
3. Enter the appropriate value in the TERMINAL GROUP field, based on the terminal group to which you want the ARSP parameters to apply.
4. Enter 01, 02, or 03 in the RECORD NUMBER field on the ARSP Selection screen and press the **F9** key to access the first screen associated with the ARSP Language Display record.
5. Select the index number to which you want to add a response code by pressing the **Tab** key until you are positioned directly to the right of the index number on the screen.

   The **Tab** key automatically positions you in the second column on the screens where you can enter your modified response code descriptions.
6. Enter a response code description of up to 48 characters in length.
7. Repeat the previous two steps until you have completed indexing the response code descriptions on one screen.

8. Verify the response code descriptions are entered as you want them on the screen, make any corrections, and move to the next screen by pressing the **F9** key. The **F9** key is used to access subsequent pages associated with the record.

   **Note:** The response code descriptions indexed in the ARSP Language Display record correspond to response code numbers in the ARSP Response Display records. You should verify that response code descriptions match their corresponding response codes. In addition, you must enter response code descriptions on each screen for each response code number indexed in the ARSP Response Display records.

9. Press the **F9** key to move to the next screen and repeat the procedures described above on each ARSP Language Display screen for each response code description required.

10. When you have completed indexing the response code descriptions on all of the ARSP screens, press the **F3** key to add the record.

11. Press the **F2** key to read the record to verify it has been added correctly.

# Language Display Screens

Since all of the Language Display records are formatted in exactly the same manner, only the screens for record 01 are included in this section. Record 01 is described on the following pages, followed by descriptions of its fields. In addition, since the fields on the language response screens do not have names, the data on the screens is described according to the column in which it is located. The first column on the screen is headed with INDEX. The second column consists of a digit followed directly by a response.

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 05
    TERMINAL GROUP:                              RECORD NUMBER:  01

                              LANGUAGE 1

          INDEX   <response code description>
            0
            1
            2
            3
            4
            5
            6
            7
            8
            9



 ********************************** BASE24 **********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                           F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  03 OF 05
    TERMINAL GROUP:                              RECORD NUMBER:  01

                              LANGUAGE 1

           INDEX   <response code description>
            10
            11
            12
            13
            14
            15
            16
            17
            18
            19




 ********************************** BASE24 ***********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  04 OF 05
    TERMINAL GROUP:                              RECORD NUMBER:  01

                              LANGUAGE 1

           INDEX   <response code description>
            20
            21
            22
            23
            24
            25
            26
            27
            28
            29




 ********************************** BASE24 ***********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  05 OF 05
    TERMINAL GROUP:                              RECORD NUMBER:  01


                              LANGUAGE 1

           INDEX   <response code description>
            30
            31
            32
            33
            34
            35
            36
            37
            38
            39



 ********************************** BASE24 **********************************
  NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                          F12-HELP
```

**INDEX —** The number corresponding to the Response Display Map. When a number is listed in the third column of the Response Display Map, this column is searched for the same number. The response on the same line is used when sending a response to the terminal.

| | |
|---|---|
| Field Length: | System-protected |
| Occurs: | 10 times per screen |

**<response code description> —** The description to be sent to the terminal when the number in the first column is referenced on the Response Display Map. The first character in the column is a digit that reflects the codes sent from the terminal to the SPDH module to indicate the language.

The terminal sends 0 to indicate the responses in Table 1; 1 to indicate Table 2; and 2 to indicate Table 3.

Optional data may be included in the responses listed on this screen, if certain codes are included in the response. Refer to the Configurable Receipts discussion in section 6 for a complete list of available optional data fields. The codes allow variable data to be included in the response. The codes must be uppercase or lowercase as shown.

| | |
|---|---|
| Example: | 0\D\K\s |
| Field Length: | 1 to 48 alphanumeric characters |
| Occurs: | 10 times per screen |
| Required Field: | Yes |
| Default Value: | No default value |
| Data Name: | ARSP.DISPLAYS.TEXTS |

# Transaction Description Records

The ARSP Transaction Description Record screens allow you to provide transaction descriptions for each transaction type known to the SPDH module.

The ARSP Transaction Description Record consists of four screens, each containing two columns. The first column provides descriptions of all transaction types known to the SPDH module. The second column contains corresponding blank fields into which you can enter up to 24 characters of text to create your own transaction descriptions for the different transaction types.

**Note:** The descriptions in the ARSP can be overridden by a corresponding record in the Response Code Description File (RCDF). Refer to the ***BASE24-pos Files Maintenance Manual*** for information about the RCDF.

## SPDH Module Transaction Description Processing

The inclusion of transaction descriptions in transaction responses is controlled by the presence of FID s (Transaction Description).

If FID s is configured for inclusion in the transaction response, the response will contain a transaction description. If FID s is not configured for inclusion in the transaction response, the response will not contain a transaction description.

You can specify whether to include FID s in responses for each transaction type when configuring the Field Map screens in the ACNF. For more information on setting up the ACNF Field Map screens, refer to the ACNF documentation earlier in this section.

When the SPDH module is configured to include FID s in its response, the SPDH module places information in the FID in the following order of priority:

1. If a description exists for the transaction in the Response Code Description File (RCDF), the SPDH module uses that description.
2. If a description exists for the transaction in the ACI Standard Device Response File (ARSP), the SPDH module uses that description.
3. If no description exists for the transaction in the RCDF or ARSP, the SPDH module sets FID s to blanks.

**Note:** The transaction code in the standard message header of the request is used by the SPDH module to identify the appropriate transaction description.

# Setting Up Transaction Description Records

The following instructions explain how you can access and set up the ARSP Transaction Description Record screens. Setting up this record may require you to use information contained in the ACI Standard Configuration File (ACNF), which is explained earlier in this section. To access the ARSP Transaction Description Record screens, perform the following steps:

1. Check the ACNF Field Map screens to ensure that FID s (Transaction Description) has been configured to be included in messages for the transaction types you want. If FID s is not configured for the transaction types you want, add it to the configuration using the steps described in the "Setting Up Field Map Records" topic documented earlier in this section.
2. Enter ARSP in the FILE DESTINATION field at the bottom of the ACNF screen and press the **F16** key to access the ARSP Selection screen.
3. Enter the appropriate value in the TERMINAL GROUP, based on the terminal group to which you want the ARSP parameters to apply.
4. Enter 04 in the RECORD NUMBER field on the ARSP and press the **F9** key to access the first screen associated with the ARSP Transaction Description record.
5. Select the transaction description you want to enter by pressing the **Tab** key until you are positioned directly to the right of the applicable transaction listed in column one.

   You cannot alter the transaction descriptions displayed in the first column on the screens. The **Tab** key automatically positions you in the second column on the screens where you can enter or modify your transaction description text.
6. Enter up to 24 characters of text to create your own transaction description.
7. Repeat the previous two steps until you have completed altering transaction descriptions on one screen.
8. Verify the transaction descriptions are as you want them on the screen, make any corrections, and move to the next screen by pressing the **F9** key. The **F9** key is used to access subsequent pages associated with the record.

   **Note:** The ARSP Transaction Description record consists of four screens, each containing transaction descriptions.
9. Repeat the procedures described above on each ARSP Transaction Description Record screen for each transaction description you want to alter.

10. When you have completed altering the transaction descriptions on all of the ARSP Transaction Description Record screens, press the **F3** key to add the record.
11. Press the **F2** key to read the record to verify it has been added correctly.

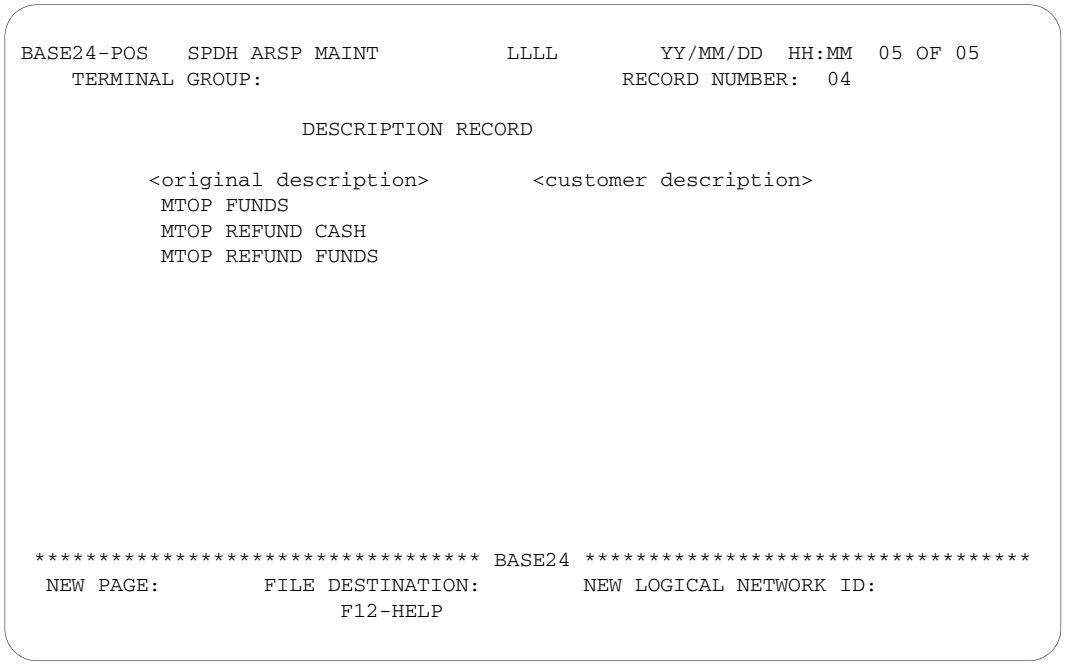
## Transaction Description Record Screens

ACI Standard Device Response File (ARSP) Transaction Description Record screens are shown on the following pages, followed by descriptions of their fields. Since the Transaction Description Record screens do not have field names, the data on the screens are described according to the column in which the data is displayed.

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  02 OF 05
   TERMINAL GROUP:                              RECORD NUMBER:  04

                    DESCRIPTION RECORD

         <original description>        <customer description>
          NORMAL PURCH
          PRE-AUTH PURCH
          PRE-AUTH PURCH/COMP
          PHONE/MAIL ORDER
          MERCH RETURN
          CASH ADVANCE
          CARD VERIFICATION
          BALANCE INQ
          PURCH/CASH BACK
          CHK VERIFICATION
          CHK GUARANTEE
          PURCH ADJUSTMENT


********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                        F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  03 OF 05
   TERMINAL GROUP:                              RECORD NUMBER:  04

                     DESCRIPTION RECORD

          <original description>        <customer description>
           MERCH RETURN ADJUSTMENT
           CASH ADVANCE ADJUSTMENT
           CASH BACK ADJUSTMEMT
           LOGON REQUEST
           LOGOFF REQUEST
           CLOSE BATCH REQUEST
           CLOSE SHIFT REQUEST
           CLOSE DAY REQUEST
           EMPLOYEE SUBTOT REQUEST
           BATCH SUBTOT REQUEST
           SHIFT SUBTOT REQUEST
           DAY SUBTOT REQUEST


********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                        F12-HELP
```

```
BASE24-POS   SPDH ARSP MAINT          LLLL        YY/MM/DD  HH:MM  04 OF 05
   TERMINAL GROUP:                              RECORD NUMBER:  04

                     DESCRIPTION RECORD

          <original description>        <customer description>
           READ MAIL REQUEST
           MAIL DELIVERED REQUEST
           SEND MAIL REQUEST
           DOWNLOAD REQUEST
           HANDSHAKE REQUEST
           RESERVED
           CARD ACTIVATION
           ADNL CARD ACTIVATE
           REPLENISHMENT
           FULL REDEMPTION
           RESERVED
           MTOP CASH


********************************** BASE24 **********************************
 NEW PAGE:        FILE DESTINATION:        NEW LOGICAL NETWORK ID:
                        F12-HELP
```




# DESCRIPTION RECORD

The following fields describe supported transactions with original descriptions and customer descriptions.

**<original description> —** The transaction description as it is known by the SPDH module.

| | |
|---|---|
| Field Length: | System-protected |
| Occurs: | Up to 12 times per screen |

**<customer description> —** The transaction description as defined by the customer for the purpose of responses.

| | |
|---|---|
| Field Length: | 1 to 24 alphanumeric characters |
| Occurs: | Up to 12 times per screen |
| Required Field: | No |
| Default Value: | No default value |
| Data Name: | ARSP.DESCR.TBL |

*ACI Worldwide, Inc.*

# A: Track Data Processing

One of the functions of the SPDH module is to translate the native message into the POS Standard Internal Message (PSTM) before sending the message to the Authorization module.

Native message track data is found in optional data fields. The FID for Track 2 customer or supervisor data is q or r, respectively. The FID for Track 1 customer or supervisor data is 2 or 3, respectively. These optional data fields are defined in section 4.

This appendix describes the actions the SPDH module takes when translating track data to the PSTM under various scenarios.

For more information about the PSTM, refer to the ***BASE24-pos Transaction Processing Manual.*** For more information about the Track 1 token, refer to the ***BASE24 Tokens Manual.***

# PSTM Track Data

This appendix describes how the SPDH module formats the PSTM based on the track data in the native message. The subsections that follow discuss these scenarios:

- The native message contains Track 2 data only
- The native message contains Track 1 data only
- The native message contains both Track 1 and electronically entered Track 2 data
- The native message contains both Track 1 and Track 2 data, but the Track 2 data was manually entered
- The native message contains no track data

Note that the SPDH module does not support manually entered Track 1 data.

## Track 2 Data Only

When the native message contains only Track 2 data (the Track 2/Customer or Track 2/Supervisor data FID contains a value; the Track 1/Customer and Track 1/ Supervisor FID are not present), the SPDH module fills the TRAN.TRACK2 field in the BASE24-pos Standard Internal Message (PSTM) with the information from the appropriate Track 2 data field.

If the data is entered manually, the SPDH module sets the first byte of the TRAN. TRACK2 field in the PSTM to M and the PT-SRV-ENTRY-MDE field in the PSTM to 01 (manual). If an expiration date is not specified, the SPDH module uses the appropriate default value. The SPDH module also sets the COMPLETE-TRACK2-DATA field in the BASE24-pos Release 5.0 token to the corresponding value from the COMPLETE TRACK DATA field on POS Terminal Data files (PTD) screen 2. The value in this field should not be used for further processing.

If the data is entered electronically, the SPDH module sets the PT-SRV-ENTRY-MDE field in the PSTM is set to 02 (magnetic stripe). The SPDH module also sets the COMPLETE-TRACK2-DATA field in the BASE24-pos Release 5.0 token to the corresponding value from the COMPLETE TRACK DATA field on PTD screen 2.

If FID 6, subFID E (POS Entry Mode) is present in the message and contains a value of 05 (integrated circuit card), 07 (contactless chip card), or 91 (contactless magnetic stripe card), the SPDH module sets the PT-SRV-ENTRY-MDE field in the PSTM to that value.

## Track 1 Data Only

When the native message contains only Track 1 data (the Track 2/Customer and Track 2/Supervisor FIDs are not present; the Track 1/Customer or Track1/ Supervisor FID contains a value), the SPDH module extracts the PAN and expiration date from the Track 1 FID and places them in the TRAN.TRACK2 field of the PSTM. The SPDH module sets the first position of the field to M. However, the SPDH module sets the PT-SRV-ENTRY-MDE field in the PSTM to 02 to indicate the data was read electronically.

The SPDH module then moves the information from the Track 1 data field to the Track 1 token and adds the token to the PSTM. If an error occurs when adding the Track 1 token, the SPDH module continues processing the transaction if possible. If the Track 1 token cannot be added, the SPDH module sets the PT-SRV-ENTRY-MDE field to 01 to indicate that the track data is manually entered.

If FID 6, subFID E (POS Entry Mode) is present in the message and contains a value of 05 (integrated circuit card), 07 (contactless chip card), or 91 (contactless magnetic stripe card), the SPDH module sets the PT-SRV-ENTRY-MDE field in the PSTM to that value.

In addition, the SPDH module sets the COMPLETE-TRACK2-DATA field in the BASE24-pos Release 5.0 token to the corresponding value from the COMPLETE TRACK DATA field on PTD screen 2.

## Both Track 1 and Track 2 Data

If the native message contains both Track 1 and electronically entered Track 2 data (both the Track 2 and Track 1 fields contain values; the first byte of the Track 2 field is not M), the SPDH module places the information from the Track 2 data field in the TRAN.TRACK2 field in the PSTM. It sets the PT-SRV-ENTRY-MDE field in the PSTM to 02 to indicate the cardholder information was read electronically.

The SPDH module then moves the Track 1 information from the Track 1 data field to the Track 1 token and adds the token to the PSTM. If an error occurs when adding the Track 1 token, the SPDH module continues processing the transaction if possible.

If FID 6, subFID E (POS Entry Mode) is present in the message and contains a value of 05 (integrated circuit card), 07 (contactless chip card), or 91 (contactless magnetic stripe card), the SPDH module sets the PT-SRV-ENTRY-MDE field in the PSTM to that value.

In addition, the SPDH module sets the COMPLETE-TRACK2-DATA field in the BASE24-pos Release 5.0 token to the corresponding value from the COMPLETE TRACK DATA field on PTD screen 2.

## Track 1 Data and Manually Entered Track 2 Data

If the SPDH module receives Track 1 data and manually entered Track 2 data (the first character of the Track 2/Customer or Track 2/Supervisor field is M; the Track 1/Customer or Track 1/Supervisor field contains a value), the SPDH module ignores the manually entered data and processes the track data as described in the topic, “Track 1 Data Only.”

## No Track Data

If the SPDH module receives no track data (not all transaction types require track data), the SPDH module determines whether an error condition exists and processes accordingly.

# B: Timeout Reversals

This appendix illustrates the transaction flows in scenarios that contain a reversal at the POS device, controller, or SPDH module.

This appendix includes diagrams of the following scenarios in environments where transactions are sent to a host processor or interchange for authorization and where transactions are authorized by the BASE24-pos Authorization module.

- Timeout of an online transaction at the controller
- Timeout of a store-and-forward transaction at the controller
- Timeout of an online transaction at the SPDH module
- Communication failure during a request to the SPDH module
- Communication failure during a response to the controller (online transaction); XPNET process is aware of the failure
- Communication failure during a response to the controller (store-and-forward transaction); XPNET process is aware of the failure
- Communication failure during a response to the controller (online transaction); XPNET process not aware of the failure
- Communication failure during a response to the controller (store-and-forward transaction); XPNET process not aware of the failure
- Timeout of a timeout reversal at the controller
- Store-and-forward transaction arrives at the SPDH module before a late response from the authorizer

The diagram for a communication failure during a request to the SPDH module is the same for both online and offline authorization. The scenario is documented only in the subsection “Transaction Flows for Online Authorization.”

Two scenarios, timeout of an online transaction at the SPDH module and store-and-forward transaction arrives at the SPDH module before a late response from the authorizer, cannot occur in offline authorization.

# Timeout Reversal Message

When a timeout causes a transaction to be reversed, the controller or POS device (if the POS device is capable of generating a reversal) sends a timeout reversal message to the SPDH module. This appendix describes how the SPDH module responds to this message.

The Message Subtype field in the standard message header of a timeout reversal is set to a value of T (timeout reversal—online) or A (timeout reversal—advice). The different values are for the benefit of the controller. For each physical POS device attached to the controller, some controller software uses one logical terminal to process online transactions and one logical terminal to process store-and forward transactions. The different message subtypes enable the controller to distinguish between online transactions and store-and-forward transactions, so it can route reversal responses to the correct logical terminal. The SPDH module processes reversals with message subtypes T and A identically.

The message header of the timeout reversal message also must contain a transmission number. The SPDH module uses the transmission number to match the timeout reversal to the transaction it is intended to reverse.

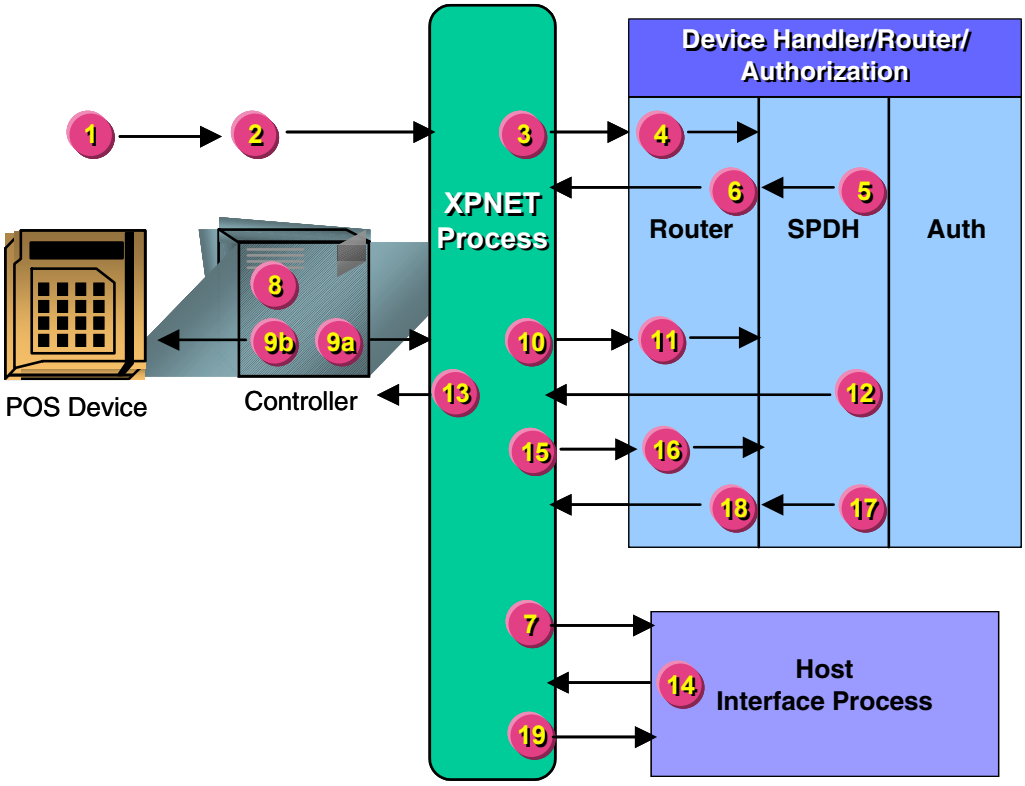
The SPDH responds to a timeout reversal message with the optional timeout reversal response message. The reversal response message is an echo of the reversal request. If your POS device software or controller software supports timeout reversals, the REVERSAL RESPONSES flag on ACI Standard Configuration File (ACNF) screen 2 must be set to a value of Y.

# Transaction Flows For Online Authorization

The following message flows use online authorization (authorization level 1). The authorizer is a host, so the SPDH module sends the messages to the Router module, which sends the message to the Host Interface process by way of the XPNET process. The message flows are the same for transactions going through an Interchange Interface process for authorization by an interchange. For more information about routing and authorization, refer to the ***BASE24-pos Transaction Processing Manual.***

## Timeout of an Online Transaction at the Controller

The diagram below illustrates the transaction message flow in a scenario that contains an online transaction that times out at the controller. This transaction begins at the POS device.

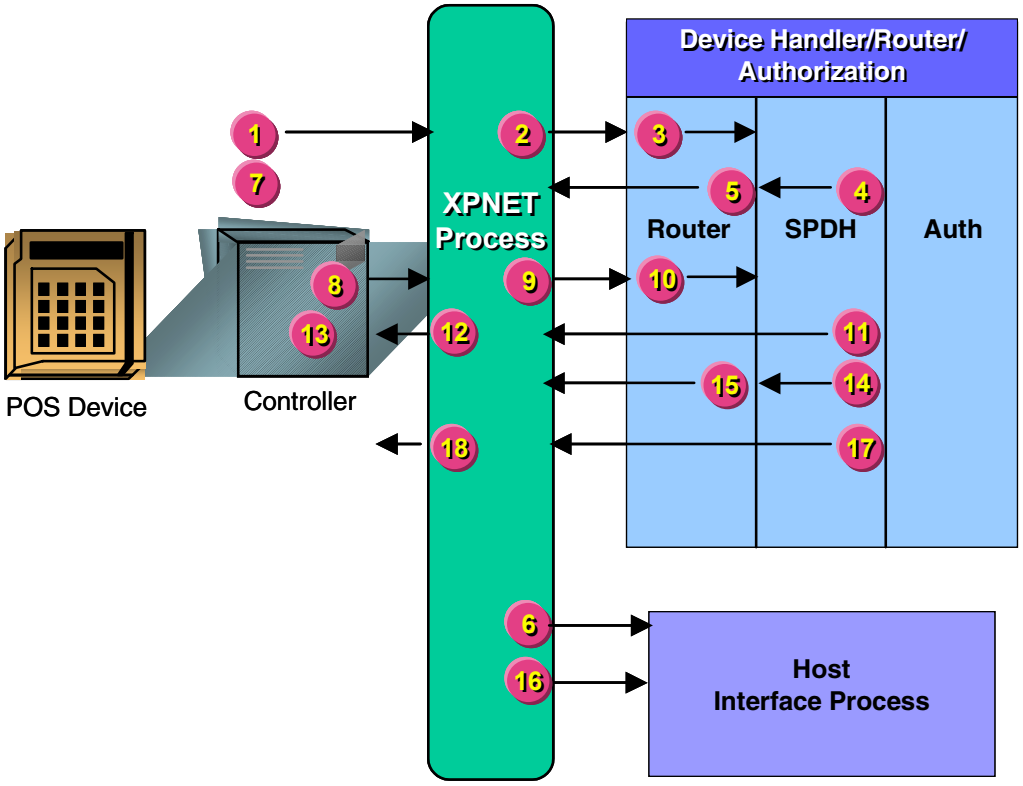



| Steps | Processing |
|---|---|
| 1, 2, 3, 4 | The POS device sends an online request to the SPDH module. This message travels through the controller, the XPNET process, and the Router module. |
| 5, 6, 7 | The SPDH module formats a 0200 request and sends it to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 8 | The transaction times out at the controller. |

| Steps | Processing |
|---|---|
| 9 | The controller performs as follows:<br>a. Sends a timeout reversal message with a message subtype of T to the XPNET process. The transmission number of the timeout reversal message must match that of the online request in step 2. Processing continues with step 10.<br>b. Responds to the POS device. This leg of processing is now complete. |
| 10, 11 | The XPNET process forwards the message to the SPDH module. This message travels through the Router module. |
| 12, 13 | The SPDH module changes the transaction status in device dependent data to a value of 3 (the response from the Authorization module timed out) and echoes the timeout reversal to the controller. This message travels through the XPNET process. |
| 14, 15, 16 | The Host Interface process sends a late 0210 response to the SPDH module. This message travels through the XPNET process and the Router module. |
| 17, 18, 19 | The SPDH module checks the device-dependent data for the transaction status. The status indicates that the transaction has timed out, so the SPDH module returns a 0420 reversal to the Host Interface process. This message travels through the Router module and the XPNET process. |

## Timeout of a Store-and-Forward Transaction at the Controller

The diagram below illustrates the transaction message flow in a scenario containing a store-and-forward transaction that times out at the controller. This transaction begins at the controller.

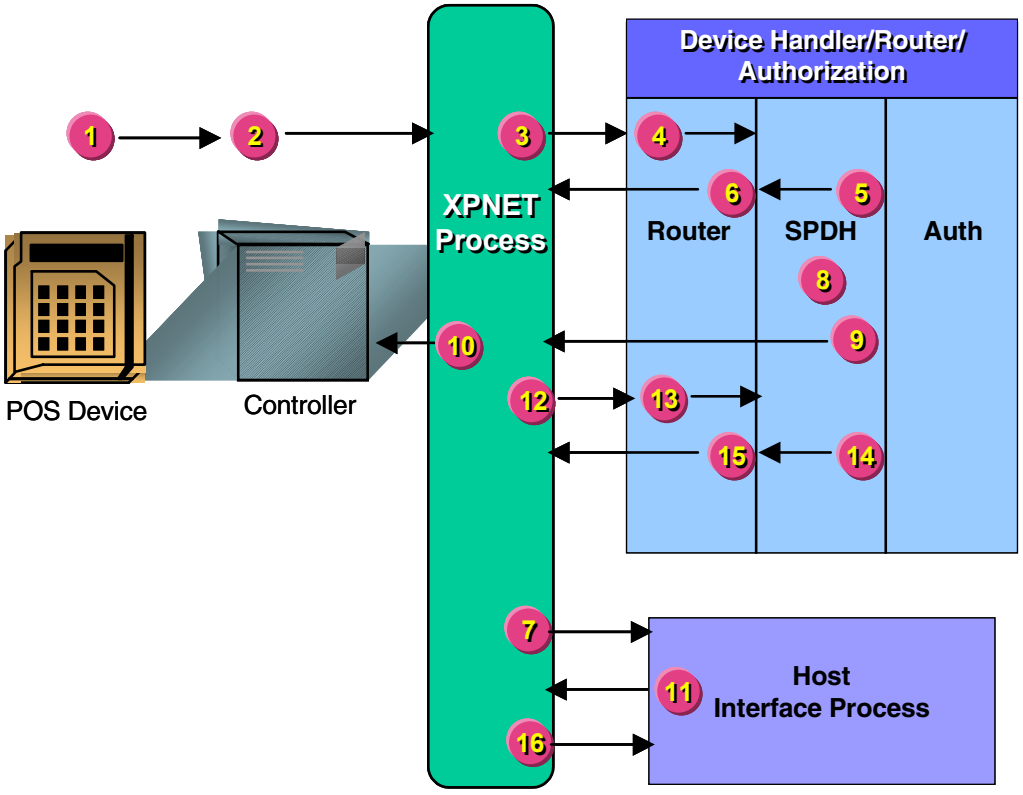



| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends a store-and-forward transaction request to the SPDH module. This message travels through the XPNET process and the Router module. |
| 4, 5, 6 | The SPDH module sends a 0220 advice to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 7 | The transaction times out at the controller. |

| Steps | Processing |
|---|---|
| 8, 9, 10 | The controller sends a timeout reversal message to the SPDH module with a message subtype of A. The transmission number of the timeout reversal message must match that of the store-and-forward transaction request in step 1. This message travels through the XPNET process and the Router module. |
| 11, 12 | The SPDH module sends a late store-and-forward response to the controller. This message travels through the XPNET process. |
| 13 | The controller drops the message. |
| 14, 15, 16 | The SPDH module sends a 0420 reversal message to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 17, 18 | The SPDH module echoes the timeout reversal response to the controller. This message travels through the XPNET process. |

## Timeout of an Online Transaction at the SPDH Module

The diagram below illustrates the transaction message flow in a scenario that contains an online transaction that times out at the SPDH module. This transaction begins at the POS device.

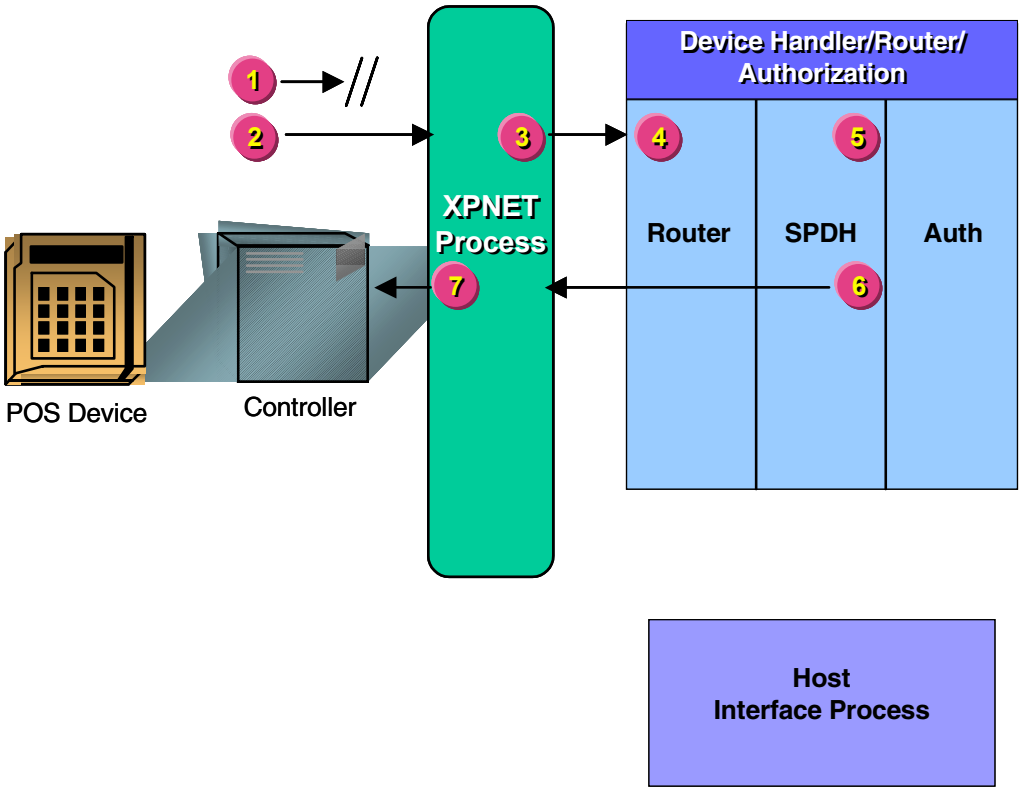



| Steps | Processing |
|---|---|
| 1, 2, 3, 4 | The POS device sends an online request to the SPDH module. This messages travels through the controller, the XPNET process, and the Router module. |
| 5, 6, 7 | The SPDH module formats and sends a 0200 request to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 8 | The transaction times out at the SPDH module. The SPDH module changes the transaction status in device dependent data to a value of 3 (the response from the Authorization module timed out). |

| Steps | Processing |
|---|---|
| 9, 10 | The SPDH module sends a response with a response code of 810 (timeout) to the controller. This message travels through the XPNET process. |
| 11, 12, 13 | The Host Interface process sends a late 0210 response to the SPDH module. This message travels through the XPNET process. |
| 14, 15, 16 | The SPDH module reads the transaction status in the device-dependent data. The status indicates that the transaction has timed out, so the SPDH module returns a 0420 reversal to the Host Interface process. This message travels through the Router module and the XPNET process. |

## Communication Failure During a Request to the SPDH Module

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a request to the SPDH module. The transaction message flow is the same for online and store-and-forward transactions and online and offline authorization. This transaction begins at the controller.

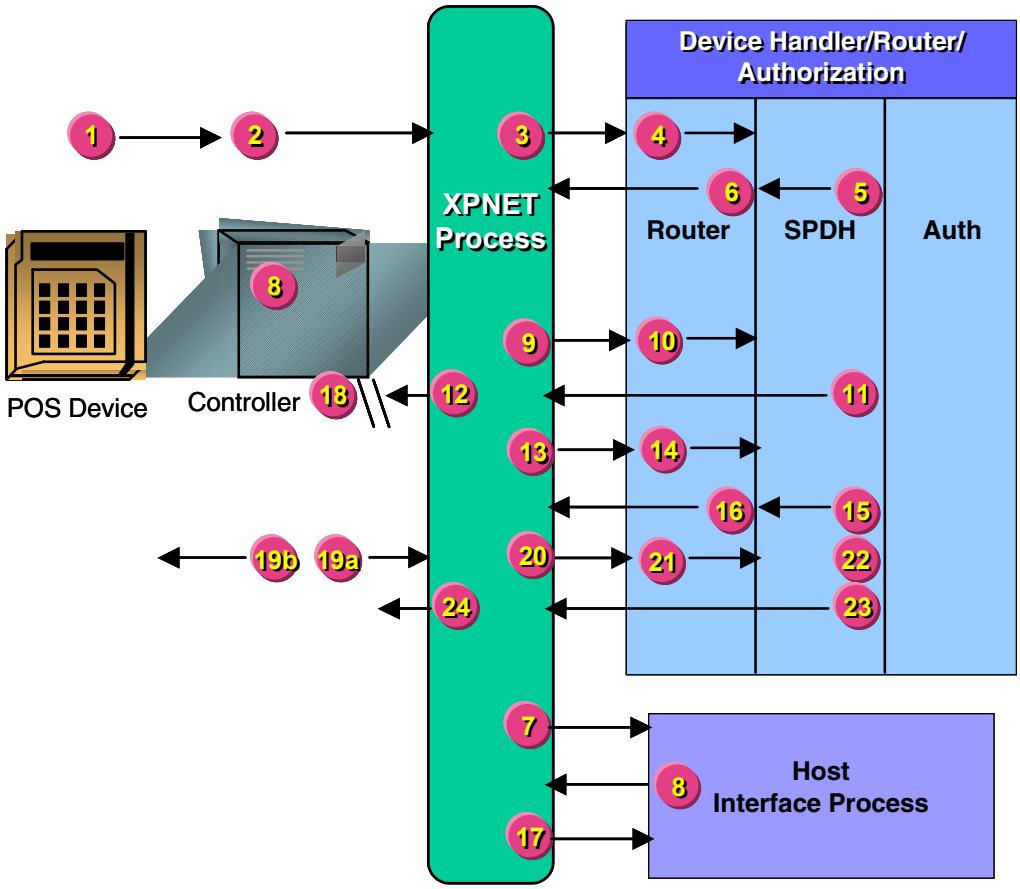



| Steps | Processing |
|---|---|
| 1 | The controller sends a request to the SPDH module, but the request fails to reach its destination. |
| 2, 3, 4 | The controller sends a timeout reversal with a message subtype of T or A to the SPDH module. The transmission number of the timeout reversal matches that of the request sent in step 1. This message travels through the XPNET process and the Router module. |

| Steps | Processing |
|---|---|
| 5 | The SPDH module determines that the number of the timeout reversal does not match the transmission number of the last transaction processed and drops the timeout reversal. |
| 6, 7 | The SPDH module echoes a timeout reversal response to the controller. This message travels through the XPNET process. |

# Communication Failure During a Response to the Controller (Online); XPNET Process Aware of Failure

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for an online transaction. The XPNET process is aware of the failure. This transaction begins at the POS device.

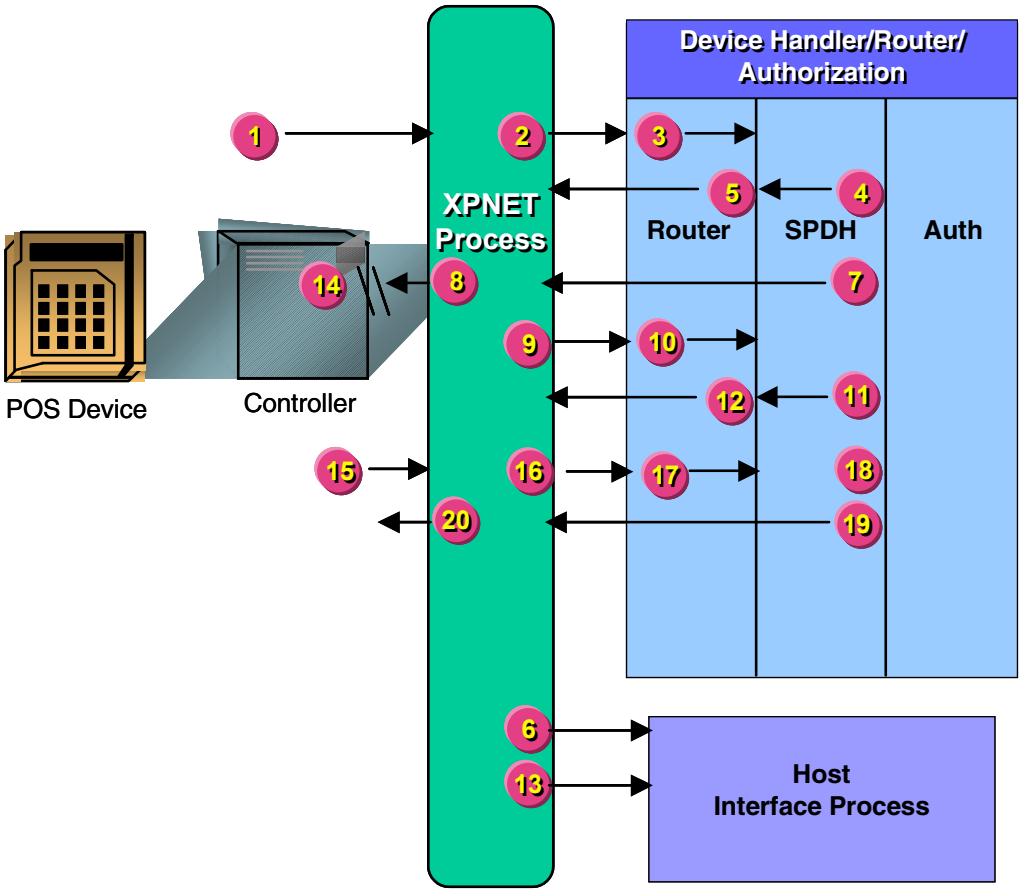



| Steps | Processing |
|---|---|
| 1, 2, 3, 4 | The POS device sends an online request to the SPDH module. This message travels through the controller, the XPNET process, and the Router module. |
| 5, 6, 7 | The SPDH module sends a 0200 request to the Host Interface process. This message travels through the Router module and the XPNET process. |

| Steps | Processing |
|---|---|
| 8, 9, 10 | The Host Interface process sends a 0210 response to the SPDH module. This message travels through the XPNET process and the Router module. |
| 11, 12 | The SPDH module sends a response to the controller. The message travels through the XPNET process but fails to reach its destination. |
| 13, 14 | The XPNET process sends a message failure notification to the SPDH module. This message travels through the Router module. |
| 15, 16, 17 | The SPDH module generates and routes a 0420 reversal to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 18 | The transaction times out at the controller. |
| 19 | The controller performs as follows:<br>a. Sends a timeout reversal message with a message subtype of T to the XPNET process. The transmission number of the timeout reversal message must match that of the online request in step 2. Processing continues with step 20.<br>b. Responds to the POS device. This leg of processing is now complete. |
| 20, 21 | The XPNET process forwards the message to the SPDH module. This message travels through the Router module. |
| 22 | The SPDH module reads the transaction status in the device-dependent data, determines that the transaction has been reversed, and drops the timeout reversal. |
| 23, 24 | The SPDH module echoes the timeout reversal response to the controller. This message travels through the XPNET process. |

# Communication Failure During a Response to the Controller (Store-and-Forward Transaction); XPNET Process Aware of Failure

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for a store-and-forward transaction. The XPNET process is aware of the failure.

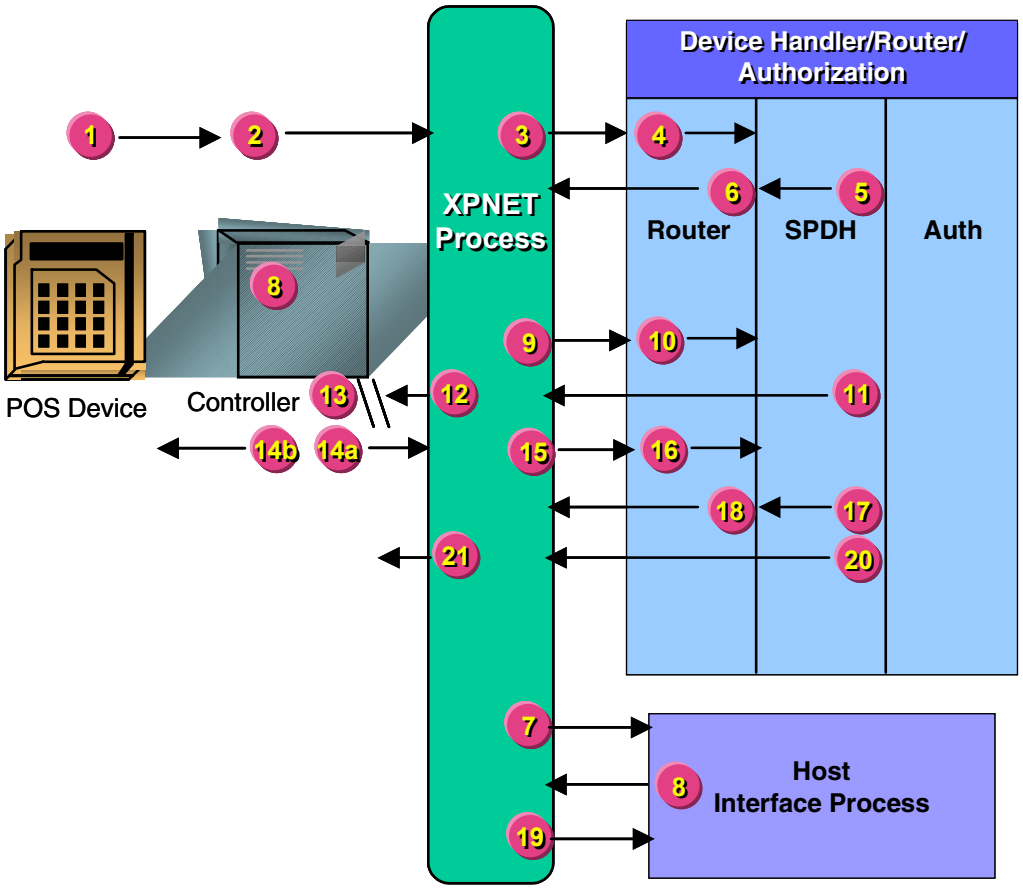



| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends a store-and-forward transaction request to the SPDH module. This message travels through the XPNET process and Router module. |
| 4, 5, 6 | The SPDH module formats and sends a 0220 advice to the Host Interface process. This message travels through the Router module and the XPNET process. |

| Steps | Processing |
|---|---|
| 7, 8 | The SPDH module sends a store-and-forward transaction response to the controller. This message travels through the XPNET process, but fails to reach its destination. |
| 9, 10 | The XPNET process sends a message failure notification to the SPDH module. This message travels through the Router module. |
| 11, 12, 13 | The SPDH module formats and sends a 0420 reversal to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 14 | The transaction times out at the controller. |
| 15, 16, 17 | The controller sends a timeout reversal with a message subtype of A to the SPDH module. The transmission number of the timeout reversal must match that of the store-and-forward transaction request sent in step 1. This message travels through the XPNET process and the Router module. |
| 18 | The SPDH module reads the transaction status in the device dependent data, determines that the transaction has already been reversed, and drops the timeout reversal. |
| 19, 20 | The SPDH module echoes a timeout reversal response to the controller. This message travels through the XPNET process. |

## Communication Failure During a Response to the Controller (Online); XPNET Process Not Aware of Failure

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for an online transaction. The XPNET process is not aware of the failure. This transaction begins at the POS device.

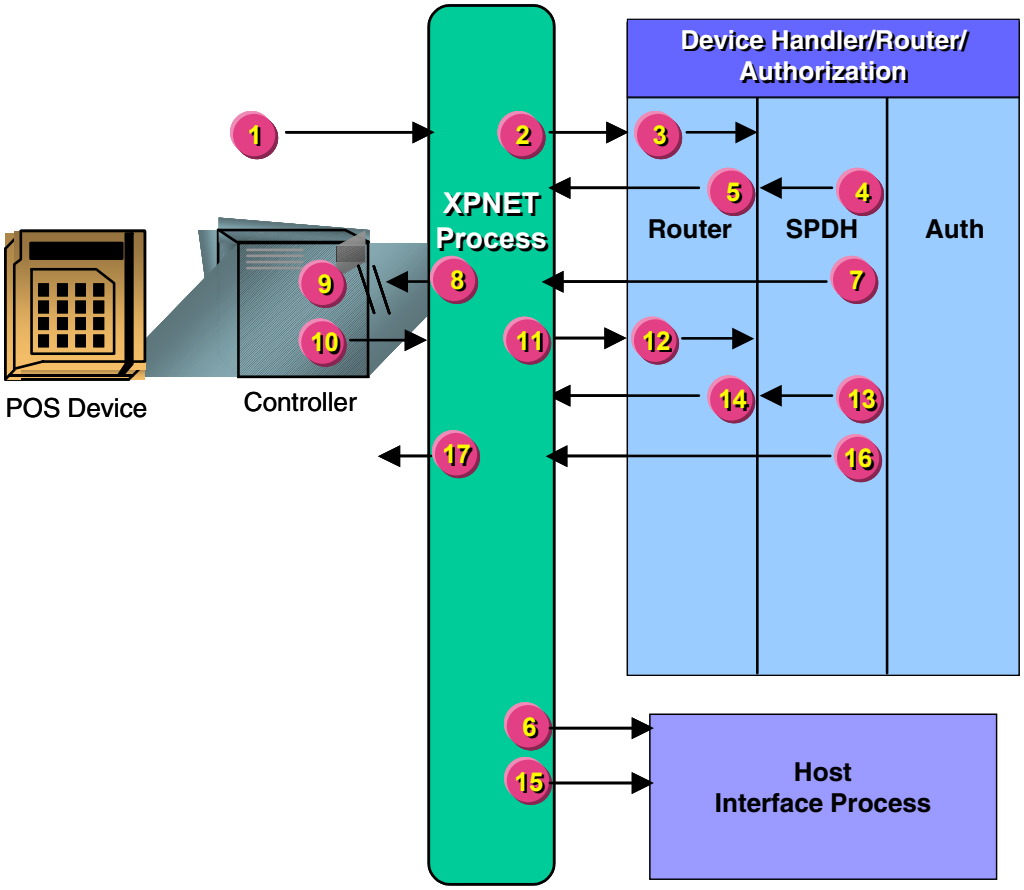



| Steps | Processing |
|---|---|
| 1, 2, 3, 4 | The POS device sends an online request to the SPDH module. This message travels through the controller, XPNET process, and the Router module. |
| 5, 6, 7 | The SPDH module routes a 0200 request to the Host Interface process. This message travels through the Router module and the XPNET process. |

| Steps | Processing |
|---|---|
| 8, 9, 10 | The Host Interface process sends a 0210 response to the SPDH module. This message travels through the XPNET process and Router module. |
| 11, 12 | The SPDH module sends a response to the controller. This message travels through the XPNET process. The message fails to reach its destination, but the XPNET process does not detect it. |
| 13 | The transaction times out at the controller. |
| 14 | The controller performs as follows:<br>a. Sends a timeout reversal message with a message subtype of T to the XPNET process. The transmission number of the timeout reversal message must match that of the online request in step 2. Processing continues with step 15.<br>b. Responds to the POS device. This leg of processing is now complete. |
| 15, 16 | The XPNET process forwards the message to the SPDH module. This message travels through the Router module. |
| 17, 18, 19 | The SPDH module sends a 0420 reversal message to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 20, 21 | The SPDH module echoes a timeout reversal response to the controller. This message travels through the XPNET process. |

# Communication Failure During a Response to the Controller (Store-and-Forward Transaction); XPNET Process Not Aware of Failure

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for a store-and-forward transaction. The XPNET process is not aware of the failure. This transaction begins at the controller.

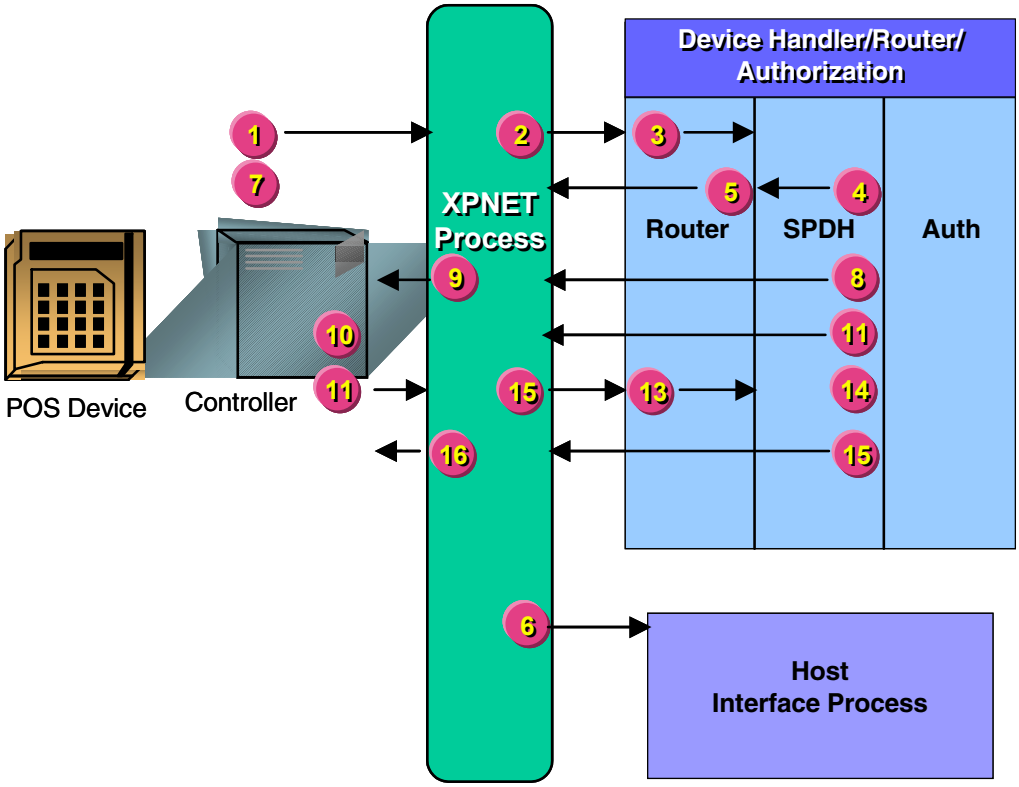



| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends a store-and-forward transaction request to the SPDH module. This message travels through the XPNET process and the Router module. |

| Steps | Processing |
|---|---|
| 4, 5, 6 | The SPDH module sends a 0220 advice to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 7, 8 | The SPDH module sends a store-and-forward transaction response to the controller by way of the XPNET process. The message fails to reach its destination, but the XPNET process does not detect it. |
| 9 | The transaction times out at the controller. |
| 10, 11, 12 | The controller sends a timeout reversal message with a message subtype of A to the SPDH module. The transmission number of the timeout reversal must match that of the message sent in step 1. This message travels through the XPNET process and the Router module. |
| 13, 14, 15 | The SPDH module routes a 0420 reversal to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 16, 17 | The SPDH module echoes a timeout reversal response to the controller. This message travels through the XPNET process. |

# Timeout of a Timeout Reversal Message at the Controller

The diagram below illustrates the transaction message flow in a scenario that contains a timeout of the timeout reversal message at the controller. This transaction begins at the controller.

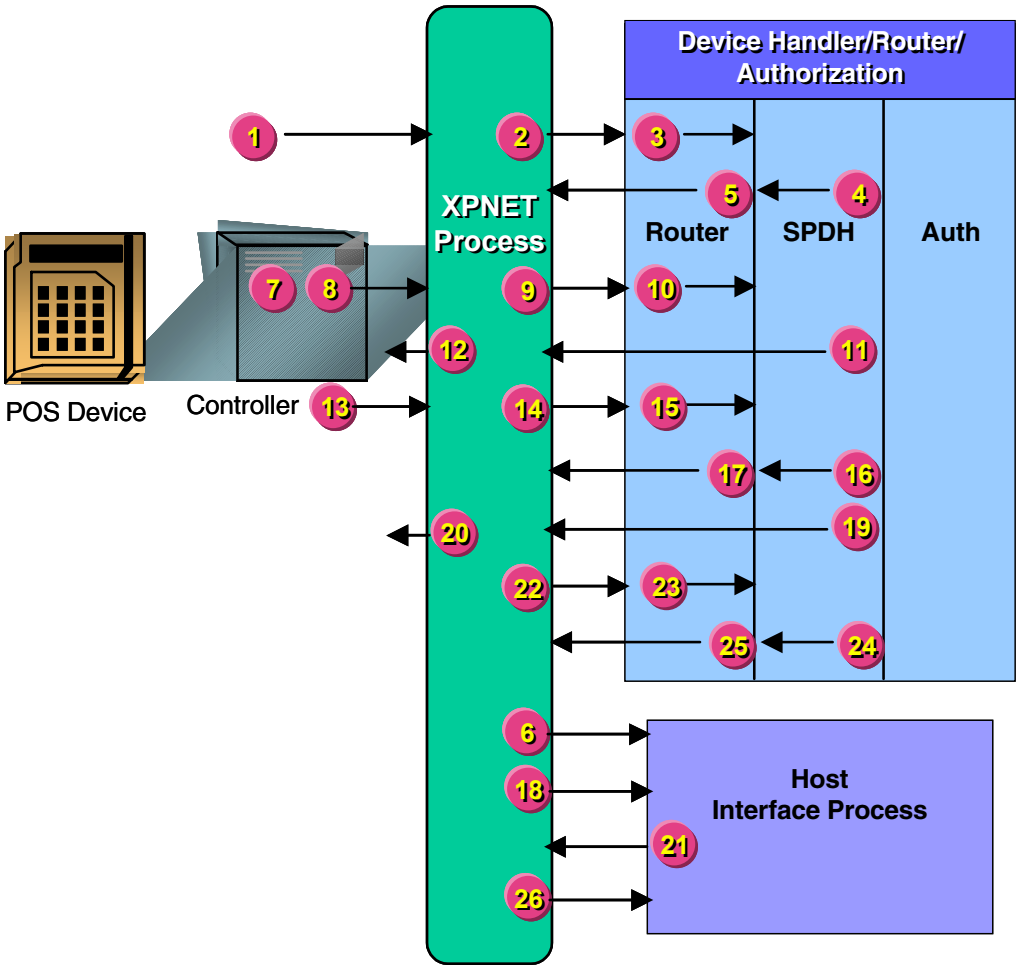



| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends a timeout reversal with a message subtype of T or A to the SPDH module. This message travels through the XPNET process and the Router module. |
| 4, 5, 6 | The SPDH module sends a 0420 reversal message to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 7 | The timeout reversal times out at the controller. The controller sets a delay timer. |

| Steps | Processing |
|---|---|
| 8, 9 | The SPDH module echoes a late timeout reversal response to the controller. This message travels through the XPNET process. |
| 10 | The controller drops the late timeout reversal response. The controller delay timer expires. |
| 11, 12, 13 | The controller resends the timeout reversal to the SPDH module. This message travels through the XPNET process and the Router module. Steps 7 through 13 repeat until the SPDH module echoes a timeout reversal response before the timeout reversal times out at the controller. |
| 14 | The SPDH module reads the transaction status in device-dependent data, determines that the transaction has already been reversed and drops the timeout reversal. |
| 15, 16 | The SPDH module echoes a timeout reversal response to the controller. |

# Store-and-Forward Transaction Arrives at the SPDH Module Before a Late Response from the Host Interface Process

The diagram below illustrates the transaction message flow in a scenario that contains a store-and-forward transaction that arrives at the SPDH module before the late response from the Host Interface process. This transaction begins at the controller.




| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends an online request to the SPDH module. This message travels through the XPNET process and the Router module. |

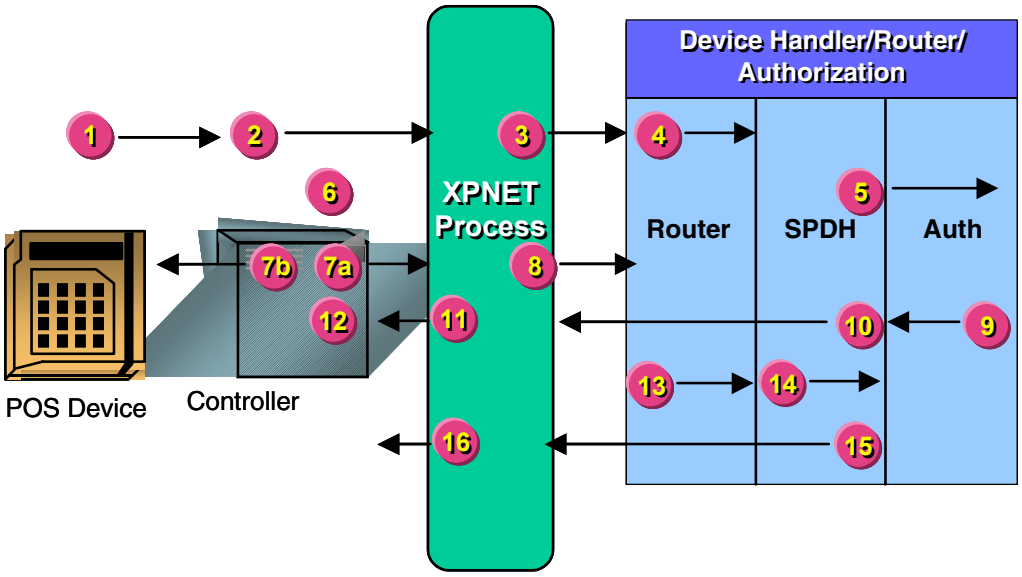
| Steps | Processing |
|---|---|
| 4, 5, 6 | The SPDH module sends a 0200 request to the Host Interface process. This message travels through the Router module and the XPNET process. |
| 7 | The transaction times out at the controller. |
| 8, 9, 10 | The controller sends a timeout reversal with a message subtype of T to the SPDH module. The transmission number of the message must match that of the message in step 1. This message travels through the XPNET process and the Router module. |
| 11, 12 | The SPDH module changes the transaction status to a value of 3 (the response from the Authorization module timed out) and echoes a timeout reversal response to the controller. This message travels through the XPNET process. |
| 13, 14, 15 | The controller sends a store-and-forward transaction request to the SPDH module. This message travels through the XPNET process and the Router module. |
| 16, 17, 18 | The SPDH module sends a 0220 advice to the Host Interface process. The SPDH module sets the transaction status to a value of B (the advice was sent to the Authorization module). This message travels through the Router module and the XPNET process. |
| 19, 20 | The SPDH module sends a store-and-forward transaction response to the controller. This message travels through the XPNET process. |
| 21, 22, 23 | The Host Interface process sends a late 0210 response to the SPDH module. This message travels through the XPNET process and the Router module. |
| 24, 25, 26 | The SPDH module checks the transaction status in the device-dependent data. The status has changed (that is, it is not set to indicate that the request was sent to the Authorization module). The SPDH module returns a 0420 reversal of the request sent in step 4 to the Host Interface process. This message travels through the Router module and the XPNET process. |

# Transaction Flows for Offline Authorization

The following message flows use offline authorization (authorization level 2). The authorizer is the BASE24-pos Authorization module. For more information about routing and authorization, refer to the ***BASE24-pos Transaction Processing Manual.***

## Timeout of an Online Transaction at the Controller

The diagram below illustrates the transaction message flow in a scenario that contains an online transaction that times out at the controller. This transaction begins at the POS device.

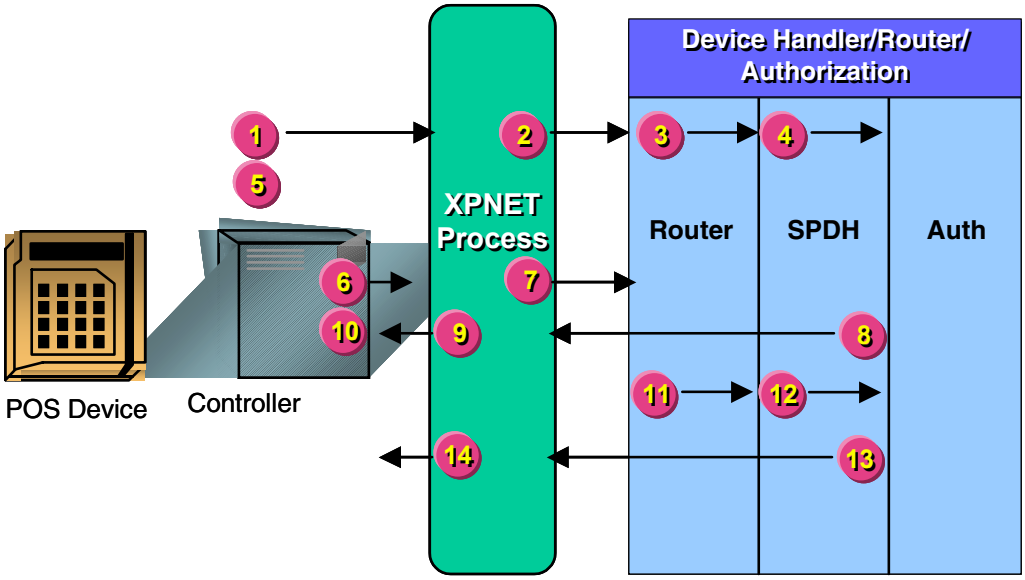



| Steps | Processing |
|---|---|
| 1, 2, 3, 4 | The POS device sends an online request to the SPDH module. This message travels through the controller, the XPNET process, and the Router module. |
| 5 | The SPDH module sends a 0200 request message to the Authorization module. |
| 6 | The transaction times out at the controller. |

| Steps | Processing |
|---|---|
| 7 | The controller performs as follows:<br>a. Sends a timeout reversal message with a message subtype of T to the XPNET process. The transmission number of the timeout reversal message must match that of the online request in step 2. Processing continues with step 8.<br>b. Responds to the POS device. This leg of processing is now complete. |
| 8 | The XPNET process forwards the message to the Router module, where it stays in the $receive queue until the SPDH module sends a transaction response to the controller. |
| 9 | The Authorization module sends a 0210 response message to the SPDH module. |
| 10, 11 | The SPDH sends a response to the controller. This message travels through the XPNET process. |
| 12 | The controller receives the late 0210 response, recognizes that it has already sent a timeout reversal message for that transaction, and drops the message. |
| 13 | The Router module retrieves the timeout reversal from the $receive queue and sends it to the SPDH module. |
| 14 | The SPDH module formats and sends a 0420 reversal message to the Authorization module. |
| 15, 16 | The SPDH module echoes the reversal response to the controller. This message passes through the XPNET process. |

## Timeout of a Store-and-Forward Transaction at the Controller

The diagram below illustrates the transaction message flow in a scenario containing a store-and-forward transaction that times out at the controller. This transaction begins at the controller.

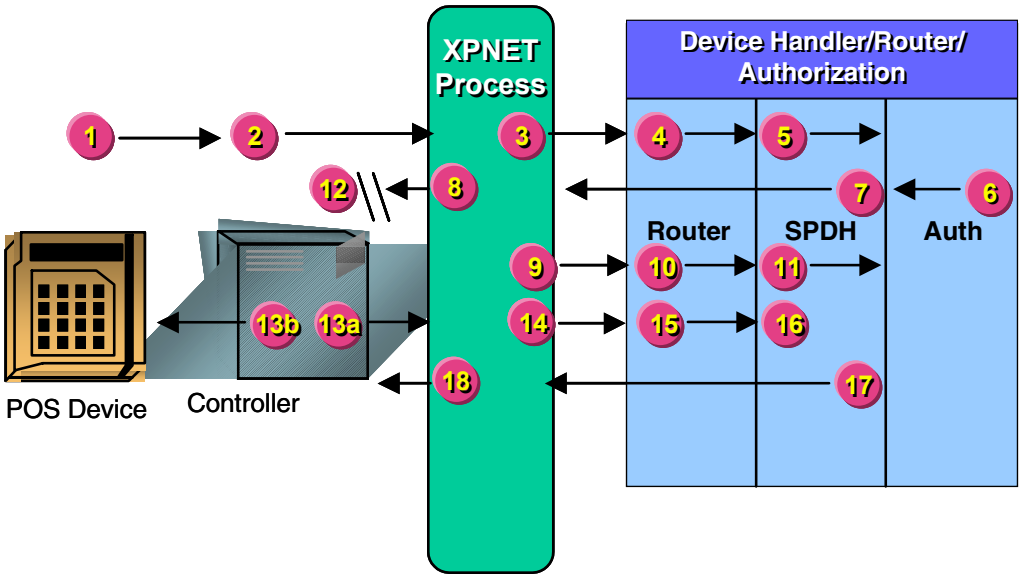



| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends a store-and-forward transaction request to the SPDH module. This message travels through the XPNET process and the Router module. |
| 4 | The SPDH module formats and sends a 0220 advice to the Authorization module. |
| 5 | The transaction times out at the controller. |
| 6, 7 | The controller sends a timeout reversal message to the SPDH module with a message subtype of A. The transmission number of the timeout reversal message must match that of the online request in step 1. This message travels through the XPNET process to the Router module, where it stays in the $receive queue until the SPDH module sends a transaction response to the controller. |

| Steps | Processing |
|---|---|
| 8, 9 | The SPDH module sends a store-and-forward transaction response to the controller. This message travels through the XPNET process. |
| 10 | The controller receives the late store-and-forward transaction response, recognizes that it has already sent a timeout reversal message for that transaction, and drops the message. |
| 11 | The Router module retrieves the timeout reversal from the $receive queue and sends it to the SPDH module. |
| 12 | The SPDH module formats and sends a 0420 reversal message to the Authorization module. |
| 13, 14 | The SPDH module echoes the reversal response to the controller. This message passes through the XPNET process. |

# Communication Failure During a Response to the Controller (Online); XPNET Process Aware of Failure

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for an online transaction. The XPNET process is aware of the failure. This transaction begins at the POS device.

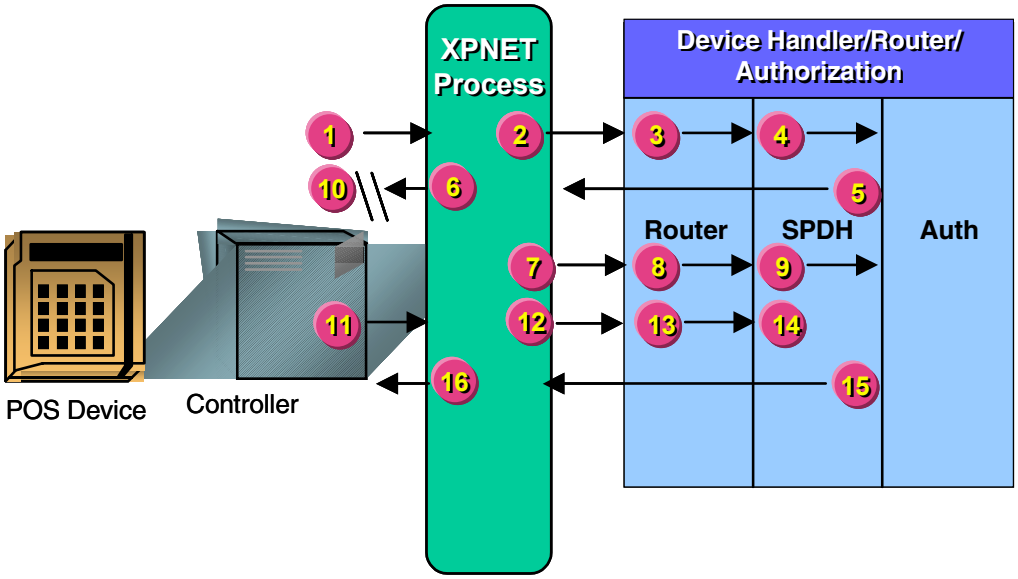



| Steps | Processing |
|---|---|
| 1, 2, 3, 4 | The POS device sends an online request to the SPDH module. This message travels through the controller, the XPNET process, and the Router module. |
| 5 | The SPDH module sends a 0200 request to the Authorization module. |
| 6 | The Authorization module sends a 0210 response to the SPDH module. |
| 7, 8 | The SPDH module sends a response to the controller. This message travels through the XPNET process but fails to reach its destination. |
| 9, 10 | The XPNET process sends a message failure notification to the SPDH module. This message travels through the Router module. |

| Steps | Processing |
|---|---|
| 11 | The SPDH module generates and routes a 0420 reversal to the Authorization module. |
| 12 | The transaction times out at the controller. |
| 13 | The controller performs as follows:<br>a. Sends a timeout reversal message with a message subtype of T to the XPNET process. The transmission number of the timeout reversal message must match that of the online request in step 2. Processing continues with step 14.<br>b. Responds to the POS device. This leg of processing is now complete. |
| 14, 15 | The XPNET process forwards the message to the SPDH module. This message travels through the Router module. |
| 16 | The SPDH module reads the transaction status in the device-dependent data, determines that the transaction has been reversed and drops the timeout reversal. |
| 17, 18 | The SPDH module echoes the timeout reversal to the controller. This message travels through the XPNET process. |

# Communication Failure During a Response to the Controller (Store-and-Forward Transaction); XPNET Process Aware of Failure

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for a store-and-forward transaction. The XPNET process is aware of the failure. This transaction begins at the controller.

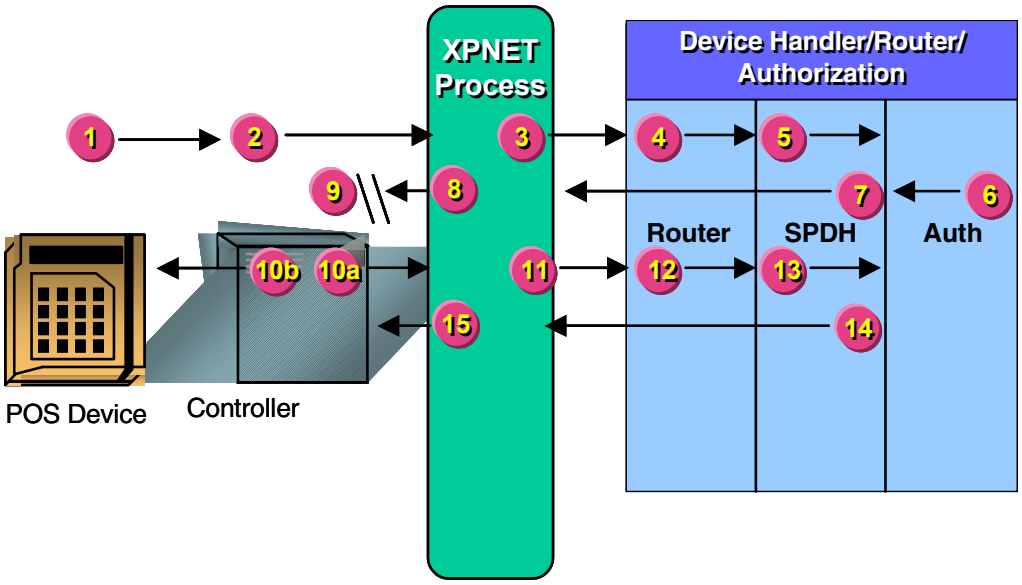



| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends a store-and-forward transaction request to the SPDH module. This message travels through the XPNET process and Router module. |
| 4 | The SPDH module formats and sends a 0220 advice to the Authorization module. |
| 5, 6 | The SPDH module sends a store-and-forward transaction response to the controller. This message travels through the XPNET process, but fails to reach its destination. |
| 7, 8 | The XPNET process sends a message failure notification to the SPDH module. This message travels through the Router module. |

| Steps | Processing |
|---|---|
| 9 | The SPDH module formats and sends a 0420 reversal to the Authorization module. |
| 10 | The transaction times out at the controller. |
| 11, 12, 13 | The controller sends a timeout reversal with a message subtype of A to the SPDH module. The transmission number of the timeout reversal must match that of the store-and-forward transaction request sent in step 1. This message travels through the XPNET process and the Router module. |
| 14 | The SPDH module reads the transaction status in the device dependent data, determines that the transaction has already been reversed, and drops the timeout reversal. |
| 15, 16 | The SPDH module echoes a timeout reversal response to the controller. This message travels through the XPNET process. |

# Communication Failure During a Response to the Controller (Online); XPNET Process Not Aware of Failure

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for an online transaction. The XPNET process is not aware of the failure. This transaction begins at the POS device.

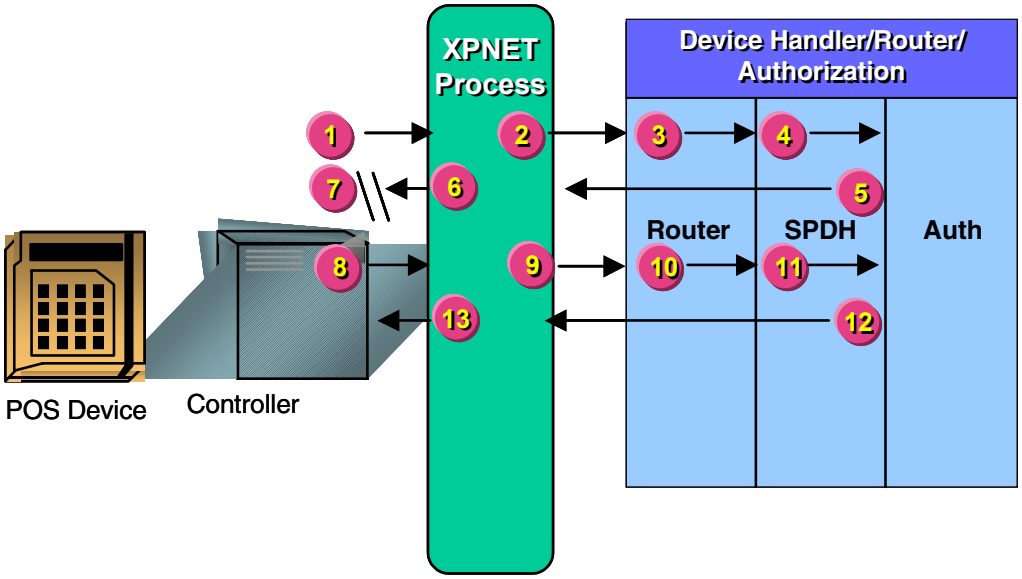



| Steps | Processing |
|---|---|
| 1, 2, 3, 4 | The POS device sends an online request to the SPDH module. This message travels through the controller, XPNET process, and the Router module. |
| 5 | The SPDH module routes a 0200 request to the Authorization module. |
| 6 | The Authorization module sends a 0210 response to the SPDH module. |
| 7, 8 | The SPDH module sends a response to the controller. This message travels through the XPNET process. The message fails to reach its destination, but the XPNET process does not detect it. |
| 9 | The transaction times out at the controller. |

| Steps | Processing |
|---|---|
| 10 | The controller performs as follows:<br>a. Sends a timeout reversal message with a message subtype of T to the XPNET process. The transmission number of the timeout reversal message must match that of the online request in step 2. Processing continues with step 11.<br>b. Responds to the POS device. This leg of processing is now complete. |
| 11, 12 | The XPNET process forwards the timeout reversal message to the SPDH module. This message travels through the Router module. |
| 13 | The SPDH module sends a 0420 reversal message to the Authorization module. |
| 14, 15 | The SPDH module echoes a timeout reversal response to the controller. This message travels through the XPNET process. |

# Communication Failure During a Response to the Controller (Store-and-Forward Transaction); XPNET Process Not Aware of Failure

The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for a store-and-forward transaction. The XPNET process is not aware of the failure. This transaction begins at the controller.

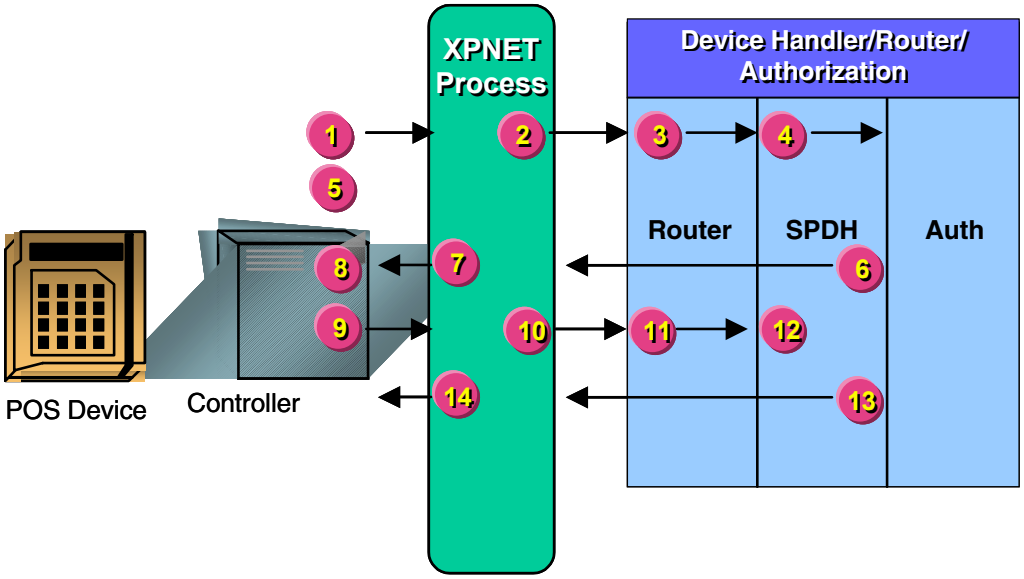



| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends a store-and-forward transaction request to the SPDH module. This message travels through the XPNET process and the Router module. |
| 4 | The SPDH module sends a 0220 advice to the Authorization module. |
| 5, 6 | The SPDH module sends a store-and-forward transaction response to the controller by way of the XPNET process. The message fails to reach its destination, but the XPNET process does not detect it. |
| 7 | The transaction times out at the controller. |

| Steps | Processing |
|---|---|
| 8, 9, 10 | The controller sends a timeout reversal message with a message subtype of A to the SPDH module. The transmission number of the timeout reversal must match that of the message sent in step 1. This message travels through the XPNET process and the Router module. |
| 11 | The SPDH module routes a 0420 reversal to the Authorization module. |
| 12, 13 | The SPDH module echoes a timeout reversal response to the controller. This message travels through the XPNET process. |

## Timeout of a Timeout Reversal at the Controller

The diagram below illustrates the transaction message flow in a scenario that contains a timeout of the timeout reversal message at the controller. This transaction begins at the controller.




| Steps | Processing |
|---|---|
| 1, 2, 3 | The controller sends a timeout reversal with a message subtype of T or A to the SPDH module. This message travels through the XPNET process and the Router module. |
| 4 | The SPDH module sends a 0420 reversal message to the Authorization module. |
| 5 | The timeout reversal times out at the controller. The controller sets a delay timer. |
| 6, 7 | The SPDH module echoes a late timeout reversal response to the controller. This message travels through the XPNET process. |
| 8 | The controller drops the late timeout reversal response. The controller delay timer expires. |

| Steps | Processing |
|---|---|
| 9, 10, 11 | The controller resends the timeout reversal to the SPDH module. This message travels through the XPNET process and the Router module. Steps 5 through 11 repeat until the SPDH module echoes a timeout reversal response before the timeout reversal times out at the controller. |
| 12 | The SPDH module reads the transaction status in the device-dependent data, determines that the transaction has already been reversed, and drops the timeout reversal. |
| 13, 14 | The SPDH module echoes a timeout reversal response to the controller. |

*ACI Worldwide, Inc.*

# C: BASE24-pos Mobile Top-Up Extension

The BASE24-pos Mobile Top-Up Device Handler extension enables cardholders to replenish, or *top up*, their mobile telephone accounts at the point-of-sale (POS) device.

BASE24-pos is enhanced to use an institution's POS network to load airtime onto a consumer's prepaid mobile phone account. The Standard POS Device Handler process (SPDH) supports acquiring mobile top-up purchases through cash or card payments.

A telecommunications services provider or an individual mobile operator, rather than the merchant, terminal owner, or institution that issues cards, is responsible for authorizing the minutes purchase, maintaining the consumer mobile telephone account, and calculating any taxes applied to the transaction. The BASE24-pos product authorizes the funds portion of the transaction.

Cardholders who perform mobile top-up transactions receive real-time replenishment if the institution supports a telecommunications service provider or an individual mobile operator. Otherwise, the BASE24-inventory product may also be used to purchase airtime.

Refer to the ***BASE24-inventory Reference Manual*** for more information on the BASE24-inventory product.

# Terminology

The following terms appear in this documentation to describe the BASE24-pos Mobile Top-Up Device Handler extension.

**Funds authorizer —** The entity responsible for authorizing the withdrawal, or *funds*, portion of a mobile top-up transaction (e.g., the BASE24-pos Router/ Authorization process).

**Issuer Identification Number (IIN) —** Phone network operators within some implementations issue ISO 7813 standard cards with 19-digit PAN numbers. The first portion of the PAN is comprised of the IIN for the phone network

**Mobile operator —** A company that provides application services for the transaction (e.g., an individual mobile telephone company).

**Telecommunication service provider —** An entity that provides telecommunication services for individual mobile operators.

**Top-up authorizer —** The entity responsible for authorizing the top-up portion of a mobile top-up transaction. This entity can verify that the mobile telephone number exists, that the account can be replenished, and that the amount or number of minutes requested is valid.

**Transaction Context Manager (TCM) —** A BASE24 module that enables the BASE24 system to receive a single request from an endpoint and split the routing between two or more destinations.

# Processing Mobile Top-Up Transactions

A mobile top-up transaction is initiated by the consumer at the POS device. The consumer selects the top-up transaction and amount and then swipes their phone card which identifies their Issuer Identification Number (IIN). The transaction is processed first by the SPDH Device Handler process and then the BASE24-pos Transaction Context Manager (TCM) process. The TCM routes the transaction, based on settings in the Split Transaction Routing File (STRF), to the BASE24-pos Router/Authorization process and the mobile operator or telecommunication service provider or to the BASE24-inventory Stock Manager process.

## Standard POS Device Handler Process

The SPDH Device Handler process performs the following functions for a mobile top-up transaction:

- Receives the transaction request from the POS device and appends the necessary tokens to identify the transaction as a mobile top-up transaction
- Forwards customer actions to the TCM process
- Receives information from the TCM process
- Formats and sends native mode messages to the POS device

Replenish, or top-up, transactions are initiated by the consumer at the point-of-sale terminal. The consumer can pay with cash or use a debit or credit card. The consumer inserts a top-up card, issued by the mobile operator, containing information identifying the consumer to the BASE24 system and the mobile interface host. After the consumer enters all necessary data, the transaction is sent from the POS device to the SPDH Device Handler process.

The SPDH Device Handler process uses the mtu_iin_subtyp_tbl in the TSPDNAMS file to fill in the TXN-SUBTYP field of the Transaction Subtype token.

A mobile top-up transaction message has two processing requirements: the funds portion and the top-up portion (includes the mobile telephone number and all processing information). If the POS-SPLIT-TXN-RTG-DEST param is present in the Logical Network Configuration File (LCONF), the Device Handler process forwards the transaction to the TCM process specified in the param, which can route the transaction to multiple authorizing entities. If the param is not present, the transaction is declined.

After the SPDH Device Handler process retrieves the destination name from the LCONF, it adds the Split Transaction Routing token with the name of the Authorization process in the FUNDS-AUTH-DEST field.

When the BASE24-pos system uses the Transaction Context Manager process, all messages between the Device Handler process and Authorization process are routed through the Transaction Context Manager process.

The Device Handler process formats the transaction as a mobile top-up transaction and builds the pre-pay top-up token which contains the phone PAN and the amount to be replenished. Upon completion of building the transaction and all its associated data, the Device Handler process sends the transaction to the TCM.

If the transaction involves a credit or a debit card, the TCM formats a purchase transaction and routes the funds portion of the transaction to the FUNDS-AUTH-DEST defined in the Split Transaction Routing token. Depending on the funds issuer, the Authorization process may or may not route the funds portion of the transaction to an interchange.

If the funds portion of the transaction is approved or if this is a cash transaction, the TCM forwards the transaction to the secondary service, a mobile operator interface, or the BASE24-inventory Stock Manager process. The mobile operator interface sends a request to the mobile operator host. Upon receiving a response from the mobile operator host, the mobile operator interface logs a record to the Interchange Log File (ILF) and sends the response to the TCM.

The TCM completes the transaction by logging the secondary service transaction to the POS Transaction Log File (PTLF) and forwards the response to the SPDH Device Handler process.

## Transaction Context Manager Process

The Transaction Context Manager (TCM) process performs routing to the telecommunication service provider or the BASE24-inventory Stock Manager process and the issuer financial institution. When the TCM process receives a transaction, it searches for the Transaction Subtype token. If the token is not present, the transaction is declined. If the token is present, the TCM retrieves the transaction subtype. Using the transaction subtype as the key, the TCM process reads the Split Transaction Routing File (STRF) extended memory table to find the secondary source and the authorizer (primary or secondary) to which it sends the transaction first. The ROUTING HIERARCHY field on the STRF screen specifies the order in which the TCM process sends the transaction message to the destinations.

Refer to the ***BASE24-inventory Reference Manual*** for more information on the Transaction Context Manager.

## Splitting the Transaction Routing

The Transaction Context Manager process splits the transaction routing to different destinations. In the case of mobile top-up processing, mobile top-up with funds transaction routing is split between the top-up authorizer and the funds authorizer. The TCM process uses the routing hierarchy specified in the Split Transaction Routing File (STRF) to determine the order to send the transaction message to the authorizing entities. The TCM process uses the Split Transaction Routing token to keep track of the transaction routing. The TCM process updates the TXN-RESP-IND, the FUNDS-AUTH-STAT, and the SCND-SVC-STAT fields in the Split Transaction Routing token each time it passes the transaction to either the funds authorizer or the top-up authorizer.

# Configuration Considerations

The following files contain fields needed for the mobile top-up product.

- TSPDNAMS
- Logical Network Configuration File (LCONF)
- Transaction Code/Subtype Relationship File (TSRF)
- Split Transaction Routing File (STRF)
- Mobile Operator File (MOF)

## Modifying the TSPDNAMS

The TSPDNAMS is an edit file that contains a table used to by the SPDH extension for mobile top-up transactions. This file allows the Device Handler process to detect the Issuer Identification Number (IIN) value and subtype for each mobile top-up authorizer destination. All IINs supported at the SPDH terminal for mobile top-up transactions must be entered in this table. The subtype is used by the Transaction Context Manager to route the transaction to the appropriate authorization destination or destinations. The IIN values in the TSPDNAMS file must match the Operator IIN values in the corresponding Mobile Operator File (MOF) records.

## Modifying the Logical Network Configuration File

The Logical Network Configuration File (LCONF) contains one param specific to the Mobile Top-Up Device Handler extension.

The POS-SPLIT-TXN-RTG-DEST param specifies the destination of the Transaction Context Manager process. The Transaction Context Manager process is required for split transaction routing. In a mobile top-up transaction, part of the transaction is routed to the third-party telecommunications process or BASE24-inventory Stock Manager process and part to the BASE24-pos Router/ Authorization process.

## Modifying the Transaction Code/Subtype Relationship File

The Transaction Code Subtype Relationship File (TSRF) contains one record for each transaction subtype supported in the network. The TSRF is used to set up a transaction subtype and related transaction codes for mobile top-up transactions. The transaction subtype distinguishes the mobile top-up transaction from other withdrawal transactions. The subtype is used when configuring the Split Transaction Routing File (STRF).

## Modifying the Split Transaction Routing File

The Split Transaction Routing File (STRF) contains one record for each transaction subtype in the BASE24 network that requires split routing. Once you have set up transaction subtypes for mobile top-up transactions in the Transaction Code Subtype Relationship File (TSRF), you must create STRF records for each subtype so that the transactions can be routed properly. This file contains the process name of the top-up authorizer for the subtype.

The ROUTING HIERARCHY field specifies the order to which the Transaction Context Manager process routes the transaction for authorization of the funds and top-up portions of the transaction. For mobile top-up with funds processing, you can configure sequential routing, with the transaction going to the funds authorizer first by setting this field to the value B0. For mobile top-up with cash processing, you can configure routing so that the transaction goes to the secondary service only by setting this field to the value BS.

Refer to the ***BASE24 Base Files Maintenance Manual*** for more information about this file.

## Modifying the Mobile Operator File

The Mobile Operator File (MOF) contains one record for each mobile operator defined to the BASE24 system.

Refer to the ***BASE24 Base Files Maintenance Manual*** for more information about the MOF.

## Tokens

In addition to tokens already used in BASE24-pos processing, the BASE24-pos Mobile Top-Up solution uses the following tokens:

**Inventory Voucher Token —** This token contains information associated with the purchase of a top-up voucher.

**POS Split Transaction Routing Token —** This token (token ID CR) carries POS-specific data that enables BASE24 to route multiple transaction requests related to a single cardholder request. It also allows BASE24 to identify and merge the multiple responses received into a single response destined for the cardholder. The token is added by the BASE24-pos Transaction Context Manager process.

**Pre-Pay Generic Receipt Token —** This token is used by the Device Handler process and the Transaction Context Manager process. It contains the timestamp when the generic receipt message was last changed.

**Pre-Pay Merchant Token —** This token is used by the Device Handler process and the Transaction Context Manager process. It contains information from the telecommunications service provider Mobile Operator File (MOF).

**Pre-Pay Original Data Token —** This token contains data from an original transaction that can be modified during pre-pay processing. This data is restored to the appropriate fields in the internal message before returning a response to the transaction originator.

**Pre-Pay Receipt Token —** This token is used by the Device Handler process and the Transaction Context Manager process. It contains information receipt information.

**Pre-Pay Top-Up Token —** This token is used by the Device Handler process and Transaction Context Manager process. This token contains the information captured by the acquiring interface, for example, the phone primary account number and the amount to be replenished.

**Split Transaction Routing Token —** This token is created by the Device Handler process when it receives a mobile top-up transaction from the device. The token contains data that allows the Transaction Context Manager process to route

multiple transaction requests related to a single cardholder. It also allows the Transaction Context Manager process to identify and merge multiple transaction responses into a single response for the cardholder.

**Transaction Subtype Token —** This token is created by the Device Handler process upon receiving a mobile top-up transaction from the device. This token contains processing codes and account types.

Refer to the ***BASE24 Tokens Manual*** for more information on these tokens.

# Field Identifiers

**Field Identifier (FID) R (Card Type) —** Supports a value of N (No card type) for mobile top-up transactions using cash. This FID is mandatory for top-up transactions.

**FID J (Available Balance) —** Contains the available balance returned from the mobile operator in a mobile top-up transaction.

**FID f (Receipt Data) —** Optionally contains 1–400 bytes of generic marketing message from the Mobile Operator File (MOF) and/or a marketing message from the mobile operator host or customer message from the BASE24-inventory Inventory Stock Configuration File (NSCF).

**FID 7 (Mobile Top- Data) —** Contains data for mobile top-up transactions.

## Subfield Identifiers

The following are the descriptions for the subfields (SFIDs) for FID 7.

**SFID a (Mobile Top-Up Track 2) —** Used to identify a pre-pay top-up transaction.

**SFID b (Original Mobile Top-Up Reference Number) —** Contains the original top-up reference number for refunds. Designated for future use.

**SFID c (Mobile Top-Up Response) —** Contains the pre-pay top-up response.

Refer to section 4 for complete information on the mobile top-up FIDs and SubFIDs.

## Logging to the POS Transaction Log File (PTLF)

The Transaction Context Manager process logs the secondary service (i.e., top-up) transaction information to the POS Transaction Log File (PTLF). This is done prior to sending the final response for the mobile top-up transaction to the Device Handler process. All approved mobile top-up transactions have two transactions logged to the PTLF: one for the funds authorization and one for the top-up authorization.

The TCM process logs the top-up transactions with a transaction code of 29 (cash) or 30 (funds). The TCM process logs the value S in the responder field, indicating a secondary service transaction and enables Transaction-based Pricing and POS settlement reports to process the secondary service transactions appropriately.

# Index

## A



## B

## C



## D



## E

## F



## H



## I



## K



## L

## M



## O

## P



## R

## S



## T

## V



## W



## X

*ACI Worldwide, Inc.*