IntelliPIN
PROGRAMMING REFERENCE MANUAL

Manual Part Number: 99875047 Rev 17

APRIL 2008

MAGTEK®
REGISTERED TO ISO 9001:2000
1710 Apollo Court
Seal Beach, CA 90740
Phone: (562) 546-6400
FAX: (562) 546-6301
Technical Support: (651) 415-6800
www.magtek.com

REVISIONS

Rev Number	Date	Notes
1	8/8/96	Initial Release.
2	9/23/96	Name Change SmartPad to IntelliPad
3	12/12/96	Name Change to IntelliPIN; Changed figures and regrouped commands in Sec 1; Added and deleted commands; Changed Setup section and Appendices.
4	12/18/96	Clarified Cmd 83.
5	6/18/97	Added Cmds 20 and 21; Removed Operator material and placed in Install and Op Manuals; Complete Revision; Title change.
6	11/6/97	Added E and F Values to Cmd 50; Modified Sec 2 – 10 to include new examples; Added page numbers references in Appendix C; Editorial Changes.
7	6/8/98	Added Commands 23, 24, 34, 35, and 36. Cmd 50, changed switches. Added Appendix E, Flow Diagrams. Examples added and editorial changes throughout. New Fig. 1-1.
8	8/17/99	Added Commands 02, 04, 08, 70, 71, Q1, Q2, Q4, Z42, Z43. Reformatted entire manual. Editorial changes throughout.
9	4/10/00	Added Commands 07, 37, 38, Z3, Z66, Z67.
10	5/11/00	Limited Warranty statement removed. Editorial changes throughout. Section 1, added Multiuse PIN. Section 2, Cmd 37, added PVNTYP statement, VALSP statement, VALLEN statement.. Confirmation Values added. Cmd 50, added MultiUse PIN (Bit 6) and IC Verify Format (Bit 7).Cmd 81, Returned Data Settings completely revised. Cmd 95 KCV added. Appendix C moved to Appendix F, all other appendices designation changed and moved forward.
11	8/04/00	Editorial changes throughout. Section 2, added Request Type 2 to Cmd 51. Added Cmd 99. Appendix A, added Extended Messages.
12	2/05/02	Editorial Throughout. Added entries to Index. Sec 2: Cmd 20 added command notes; Cmd 50 added note to switch B; Added 51C command; Cmd 60 added command note; Cmd 90 added note to Response; Cmd 99 added note to Request; Appendix A Revised second table completely.
13	4/11/03	Front Matter: Editorial throughout, added several terms to index, added ISO statement to Logo. Sec 1: Added note regarding DES and DEA. Sec 2: Cmd 31added Offset Length to request, Cmd 37 corrected Request Table, Cmd 51 Corrected Request Type 1 Table, Cmd 58 corrected Request Example, Cmd 80 added Response Table, Cmd 81 removed error status table, Cmd 95 added Session Key statement for DES and DEA, Cmd 96 added Session Key statement for DES and DEA, Cmd Z66 removed Possible Request Error table.
14	7/01/04	Added Cmd Z62, Accept And Encrypt Pin (With Custom Prompts).
15	3/23/07	Corrected formatting and standardized all examples. Changed [VD] to [VALDAT] to make document's mnemonic for Validation Data consistent. Changed the example format from using tabs to using tables for better maintainability
16	10/12/07	Fixed Command 70
17	4/3/08	Changed '20' command to indicate that correct key parity is NOT required

TABLE OF CONTENTS

FIGURES and TABLES

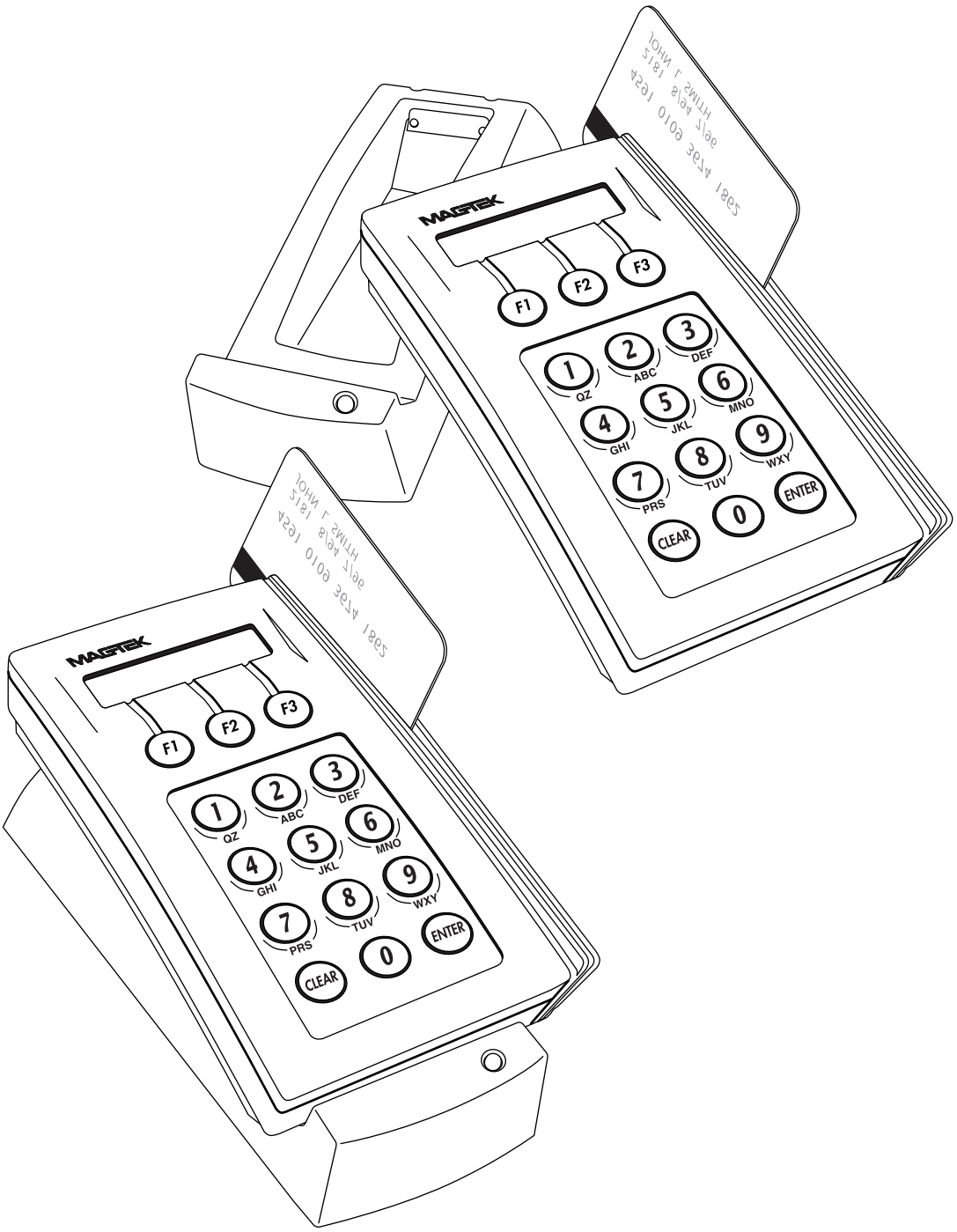


Figure 1-1. IntelliPIN, Portable and Nonportable

SECTION 1. CHARACTERISTICS AND MESSAGE COMMANDS

This document applies to the IntelliPIN®, shown in Figure 1-1, and three command sets. Two are for methods of key management: the Derived Unique Key Per Transaction, or DUKPT, and the Master/Session Key, or MSK. The third command set is for adding and removing Financial Institution Tables (FIT) when the unit is operating in the standalone mode.

RELATED DOCUMENT

MagTek Part Number 99875125	*MagTek Device Drivers For Windows Programming Reference Manual*. The MagTek Device Drivers for Windows program, Part Number 30037385, may be used with this manual.

DERIVED UNIQUE KEY PER TRANSACTION (DUKPT)

The Derived Unique Key Per Transaction, or DUKPT, is a method, which uses a derivation, or base, key to encrypt an initial Key Serial Number, which produces an initial PIN encryption key. There is a unique PIN encryption key for each transaction. Each time a pin is encrypted, the IntelliPIN outputs the PIN block along with the clear text Key Serial Number to the PC. Then a new PIN encryption key is generated to be used for the next transaction. The Derivation Key is always stored in the security module. The initial PIN encryption Key is derived from the serial number of the IntelliPIN and is therefore unique. Figure 1-2 is a simplified diagram of key loading showing the Key Injector and the relationship to the IntelliPIN and the Host Security Module.

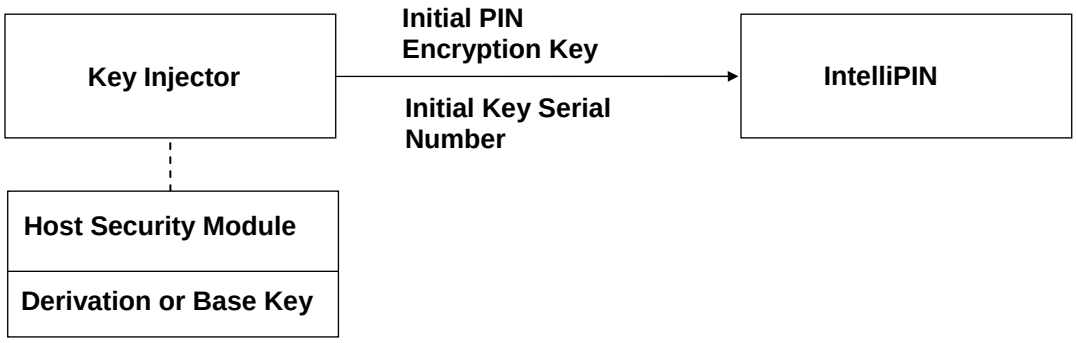


Figure 1-2. DUKPT Key Loading

Figure 1-3 indicates the DUKPT operation among the IntelliPIN, Local PC, and the Host Security Module.

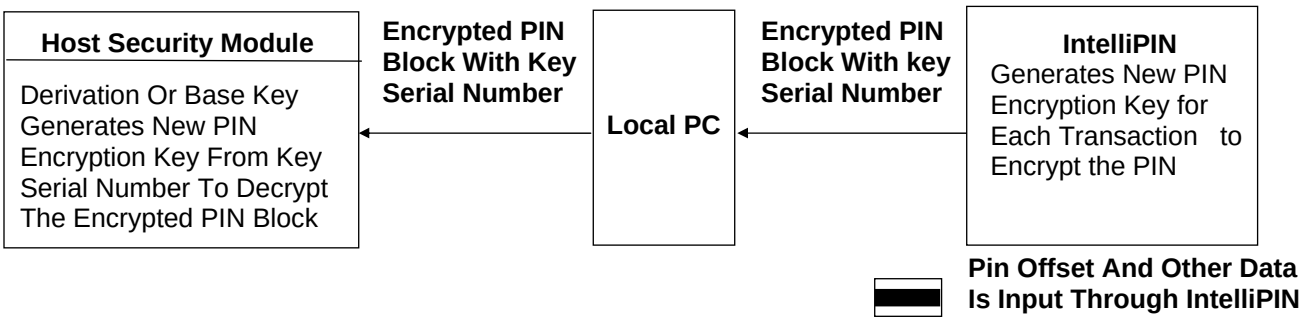


Figure 1-3. DUKPT Operation

MASTER/SESSION KEY

The Master/Session Key is used in encryption and decryption between the IntelliPIN and the host. Since there is no common key in either the IntelliPIN or the host at startup, the Master Key will be loaded in clear text; this should be done in a secure environment. The Master Key is used to encrypt the Session Key before loading to the IntelliPIN.

A simplified diagram of Key Loading for the Master/Session Key (MSK) is shown in Figure 1-4, and the operation is indicated in Figure 1-5. The Session Keys or the Working Keys can be the PIN Encryption Key.

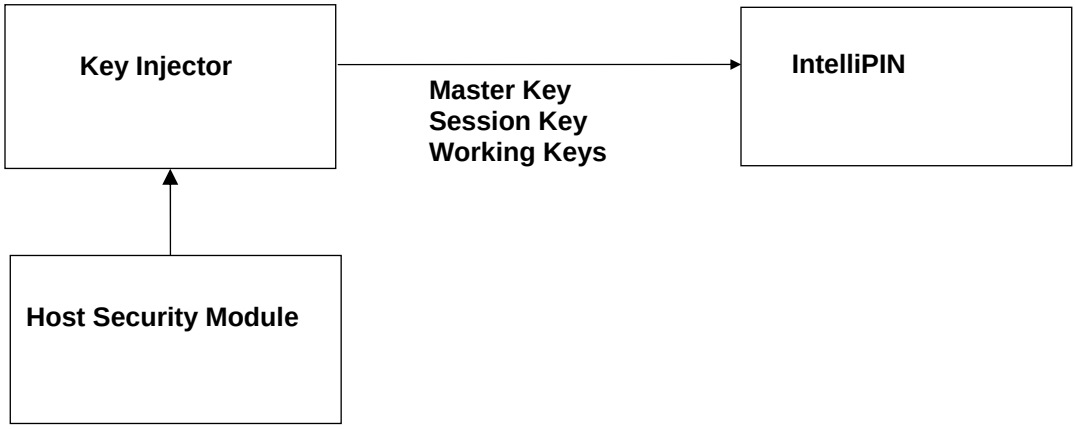


Figure 1-4. Master/Session Key Loading

The Session Key can be used as a PIN encryption key to encrypt customer PINs for transmission to the PC. For additional security, the Session Key may be used to encrypt the Working Keys which are used to encrypt customer PINs.

Note

When a key is entered as a single length (16 characters) key, it will be used in a single length DES operation. When a key is entered as a double-length (32 characters) key, all operations involving that key will follow the triple DEA method of encryption or decryption.

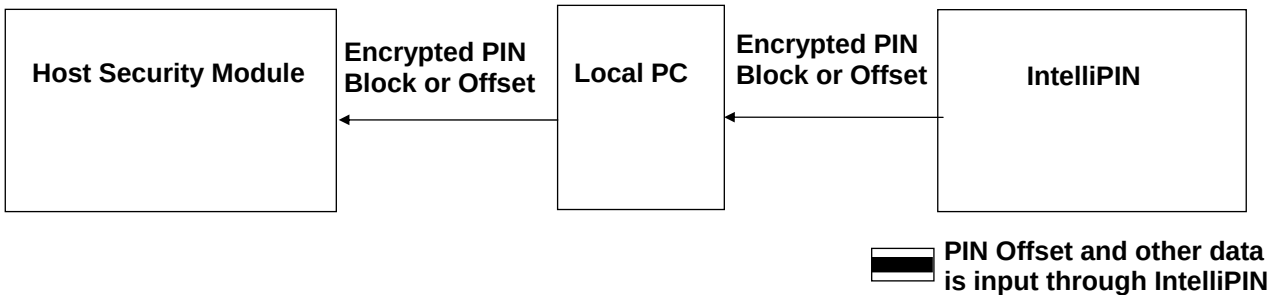


Figure 1-5. Master/Session Key Operation

MULTI-MASTER KEY (MMK)

The Multi-Master Key functions provide storage for up to ten Master Keys. These keys are used in conjunction with the PIN Entry Request (MMK) commonly used in point of sale applications. The Multi-Master Keys are independent of any other keys stored in the IntelliPIN. One of the Master Keys in this group can be used either to encrypt a PIN block to be sent to the PC or to decrypt a key received from the PC.

INTELLIPIN COMMAND MESSAGES

The messages sent to and from the IntelliPIN to manipulate operations or control specific IntelliPIN functions are divided into eight functional groups: Standalone Commands (Table 1-1), PIN Entry Commands (Table 1-2), Customer I/O Commands (Table 1-3), Configuration Commands (Table 1-4), Transaction Counter Commands (Table 1-5), Pre-authorization Commands (Table 1-6), Card Entry Commands (Table 1-7), Multi-Master Key Commands (Table 1-8), and Key Loading Commands (Table 1-9).

The tables below list the command type, a brief description of the commands, whether the command is for DUKPT, Master/Session Key, or Multi-Master Key and the page number where the command is described. The commands are linked; click on the page number.

Note

Throughout this document, all values are indicated as ASCII characters unless otherwise noted. Hexadecimal values are shown in the form 0x00 (e.g., 0x02 is the STX character). See Appendix F, ASCII Chart.

Standalone Commands

Table 1-1. Standalone Commands

Command Type	Description	Command Set	Page Number
20	Load A FIT Table	Standalone	19
21	Delete All FIT Tables	Standalone	23
23	Set/Retrieve Date	Standalone	25
24	Remote Password Entry	Standalone	27

PIN Entry Commands

Table 1-2. PIN Entry Commands

Command Type	Description	DUKPT	MSK	MMK	Page Number
30	PIN Entry Request		X		29
31	PIN Offset Request		X		33
32	PIN Verification Request		X		37
33	Encryption Test Request		X		40
34	CVV Request		X		42
35	PVV Request		X		45
36	PVV Verification Request		X		48
37	Identikey PIN Offset Request	-	-	-	51
38	Verify Identikey Offset	-	-	-	54
58	Activate or Deactivate Offset/Verify		X		93
70	PIN Entry Request (DUKPT)	X			103
71	PIN Entry Response (DUKPT)	X			107
72	Cancel Session Request	X	X	X	110
74	Pin Entry/ Data Authentication Request	X			111
75	PIN Entry and MAC Response	X			113
76	Pin Entry Test Request	X			115
78	Pin Entry Test/Data Authentication Request	X			116
Z1	Cancel Session Request	X	X	X	151

Customer I/O Commands

Table 1-3. Customer I/O Commands

Command Type	Description	DUKPT	MSK	MMK	Page Number
40	KeyPad Input Request	X	X	X	58
41	String Input Request	X	X	X	61
42	Display Single String Message	X	X	X	64
43	Display Alternating Messages	X	X	X	65
44	Firmware Part Number and Version Request	X	X	X	67
Q1	Display Swipe Card		X	X	148
Q2	Indicate Host Done		X	X	149
Q4	Turn Card Reader On/Off		X	X	150
Z2	Display a String	X	X	X	152
Z3	Display Rotating Messages	X	X	X	153
Z42	Request Noncoded Key	X	X	X	155
Z43	Return Noncoded Key	X	X	X	156

Configuration Commands

Table 1-4. Configuration Commands

Command Type	Description	DUKPT	MSK	MMK	Page Number
50	Set or Request Soft Switches	X	X	X	69
51	Replace Default Display	X	X	X	79
52	Enable Default Display	X	X	X	83
Z8	Reset/Set Idle Prompt	X	X	X	154

Transaction Counter Commands

Table 1-5. Transaction Counter Commands

Command Type	Description	DUKPT	MSK	MMK	Page Number
53	Transaction Counter Request		X	X	84
54	Transaction Counter Reset		X	X	86

Pre-Authorization Commands

Table 1-6. Pre-authorization Commands

Command Type	Description	DUKPT	MSK	MMK	Page Number
60	Pre-Authorization: PIN Entry Request	X			95
62	Pre-Authorization: Transaction Amount Authorization Request	X			96
63	Authorization Response	X			97
64	Pre-Authorization: Transaction Amount Authorization/Data Authentication Request	X			98
65	Authorization and MAC Response	X			100
66	Pre-Authorization: PIN Entry Test Request	X			102
Z60	Pre-Authorization: PIN Entry Request	X			157

Card Entry Commands

Table 1-7. Card Entry Commands

Command Type	Description	DUKPT	MSK	MMK	Page Number
80	Card Data Entry Request	X	X	X	117
81	Card Data Response	X	X		119
82	Cancel and Display	X	X	X	126
83	Card Holder Data And PIN Entry Request	X	X	X	127

Multi-Master Key Commands

Table 1-8. Multi-Master Key Commands

Command Type	Description	DUKPT	MSK	MMK	Page Number
02	Load Multi-Master Key			X	13
04	Check Multi-Master Key			X	14
07	DES Algorithm Reliability Test			X	16
08	Select Multi-Master Key			X	18
70	PIN Entry Request (MMK)			X	105
71	PIN Entry Response (MMK)			X	109
Z62	Accept And Encrypt Pin (With Custom Prompts)			X	158
Z66	Request MAC			X	160
Z67	Return MAC			X	164

Key Loading Commands

Table 1-9. Key Loading Commands

Command Type	Description	DUKPT	MSK	MMK	Page Number
55	Key Serial Number Request		X		87
56	Key Check Value Request		X		89
57	Load Substitution Table		X		91
90	Load Initial Key Request	X			131
91	Load Initial Key Response	X			132
92	Reinitialization Request	X			133
93	Reinitialization Request	X			135
94	Load Master Key		X		136
95	Load Session Key		X		138
96	Load Working Key		X		140
97	Load Key Serial Number		X		142
98	Delete Keys		X		144
99	Set/Retrieve DSN	X	X	X	146

CONTROL CHARACTERS, STRUCTURE, AND TIME-OUT

Control Character Definitions

In addition to accepting specific messages to manipulate operations, the IntelliPIN message commands include the abbreviations and special characters listed in Table 1-10.

Table 1-10. Control Character Definitions

Abbreviation	Hex Value	Description
<STX>	0x02	Start of Text
<ETX>	0x03	End of Text
<EOT>	0x04	End of Transmission
ACK	0x06	Acknowledge
<SI>	0x0F	Start of Message
<SO>	0x0E	End of Message
NAK	0x15	Negative Acknowledge
<FS>	0x1C	Field Separator
{LRC}	-	Longitudinal Redundancy Check
.	0x2E	Command Delimiter (Period)

Command Structures

The IntelliPIN accepts two request formats:

<STX> command <ETX> {LRC}

or

<SI> command <SO> {LRC}

Any other type of command will be ignored by the IntelliPIN.

Calculating Longitudinal Redundancy Check (LRC)

The LRC character provides a means of checking whether a message has been received correctly.

To calculate the LRC for a given message, exclusive-OR the binary values for all characters in the message (excluding the STX or SI), starting from an initial value of 0.

The receiver performs its own LRC calculation on the message, and compares the result with the received LRC character. If the result is not identical, the receiver sends a NAK response, indicating a communications failure. Another method is to include the received LRC in the calculations. If the final result is 0 (zero), then the message was good.

For the keyboard wedge interface, the PC receiver needs to calculate the LRC twice because of the caps lock. The first LRC is the normal LRC when the caps lock is off. The second LRC is the LRC when the caps lock is on, all the upper case characters A to Z (0x41 to 0x5a) are changed to lower case characters a to z (0x61 to 0x7a) and vice versa.

Calculating LRC Example

The <STX> is not included in the calculation.

The step-by-step calculation of the LRC for the command <STX>42MagTek<ETX> is:

Character Value From Message	Hex Message Value	Current LRC
Starting LRC Value	N/A	0x00
“4”	0x34	0x34
“2”	0x32	0x06
“M”	0x4D	0x4B
“a”	0x61	0x2A
“g”	0x67	0x4D
“T”	0x54	0x19
“e”	0x65	0x7C
“k”	0x6B	0x17
<ETX>	0x03	0x14
Final LRC	N/A	0x14

This final command as sent to the IntelliPIN would be:

<STX>	4	2	M	a	g	T	e	k	<ETX>	{LRC}
02	34	32	4D	61	67	54	65	6B	03	14

Receiving A NAK

If during a communication session either the IntelliPIN or the PC receives a NAK (Negative Acknowledge), the transmitting unit retransmits its last message and increments its NAK counter. If more than three NAKs occur while attempting to transmit the same message, the transmitting unit sends an EOT (End of Transmission), terminating communication.

Note

If the message was not correct (e.g., missing ETX, SO or bad LRC), then a NAK (0x15) will be returned.

Receiving An ACK

When the IntelliPIN receives an ACK (Acknowledge), it means the message transmitted by the IntelliPIN was received without error. When the IntelliPIN receives a command from the PC without error, the IntelliPIN will transmit an ACK.

Receiving An EOT

If during a communication session, the IntelliPIN receives an EOT (End of Transmission), it means termination of the communication session and return to the idle state.

Communications Time-Out

During a communication session, the IntelliPIN will time out if it does not receive the expected response (for example, ACK) within about 3 seconds. The unit sends an EOT to terminate the communication session. The PC program should also time-out if the IntelliPIN does not respond within one second.

IntelliPIN Header

In some applications, such as when connected through a MICR Plus unit, it may be desirable to know which unit is sending responses to the PC. The IntelliPIN can be set to included a header character, I (0x49), following the STX character and before the rest of the message. If the header option was enabled, a response would appear as:

<STX> I <response> <ETX>

Key Management

All of the Master/Session keys in the IntelliPIN can be either single- or double-length. When a key is specified as double-length (32 hex digits), the triple-DEA process will be used for encryption. If only a single-length key (16 hex digits) is used, a single DES process will be used for encryption.

When injecting a key, whether it is clear-text or encrypted, the parity will be checked. If the *Enable Key Parity* option is enabled and the parity of each byte of the key is NOT odd, an error will be generated and the key will be ignored. This option can be used to ensure that an encrypted key has been decrypted under the proper key encrypting key. If no effort has been made to ensure that the key contains odd parity, the *Enable Key Parity* option should be set to *ignore*.

Programming Hints

Keyboard Wedge Interface Timing

When implementing the keyboard wedge version of the IntelliPIN, it will be necessary to insert delays at certain points in the communication process. There are two specific points that can cause problems:

1) By sending the acknowledge (ACK) too soon after a response from the IntelliPIN has been received.
2) By sending the next command too soon after receiving an ACK (or NAK) from the IntelliPIN.

These conditions are unique to keyboard interfaces. The problem results from the fact that most characters require six scan codes, yet the PC recognizes the character after only 2 of the scan codes. (You can test this yourself by pressing the 'Enter' key and not releasing it.) Since the clock and data lines are shared by the PC and the keyboard, the data coming from the PC must wait until the data from the keyboard (IntelliPIN in this case) has been completed.

This can be illustrated with the ACK character, which is a Control-F. Here are the six scan codes: control key (going down), 'F' (going down), release code (0xF0), 'F' (going up), release code (0xF0), and control key (going up). The PC recognizes the ctrl-F character before the 4 'release' scan codes have been transmitted. When the IntelliPIN is operating at its fastest rate (80 cps), it takes about 12 ms per scan code. For this reason, we recommend at least a 100 ms delay at two places in the program:

1) After detecting an ACK from a message before sending the next command; and
2) After detecting a response before sending an ACK.

The 100 ms delay will allow the remaining 4 scan codes to be completed.

Operator Prompt Delay

In cases where operator entry on the PC keyboard is used to initiate communication with the IntelliPIN, there should be at least a 500 ms delay after any physical key is pressed to allow the IntelliPIN to reset its receive routine and prepare for the next message.

Capitalization Issues with Keyboard Wedge versions

If the Caps Lock key is active, the normal case of a response may be inverted. In particular, any response errors, which are represented by lowercase letters, will be presented as uppercase. Similarly, the card data may be presented as lowercase letters. The program should be able to accommodate the responses regardless of case.

MultiUse PIN

In some cases, it may be necessary to create two or more PIN validation codes when a customer selects a PIN. For example, it may be necessary to create both a PVV and an offset for a particular account number. In cases where the IntelliPIN will be used to compute these values (specifically with the 31, 35, and 37 commands), the customer would have to enter the PIN once for each algorithm or set of keys. With the MultiUse feature (see command 50, switch A), the PIN will be securely stored within the IntelliPIN for multiple calculations until a cancel command (72) has been received. This will permit the application program to obtain two or more offsets without burdening the customer with multiple PIN entries.

SECTION 2. COMMANDS

02 LOAD MULTI-MASTER KEY

Command Set: Multi-Master Key

Purpose: To load one of the 10 Multi-Master Keys

Command Notes: Because the IntelliPIN can store up to 10 Multi-Master Keys, the Master Key address to be loaded must be specified when using this message.

The current Multi-Master Key at the given address, if any, will be overwritten without warning. To ensure against this possibility, use the 04 Command to verify the selected address is empty.

If the *Enable Key Parity* option is enabled and the parity of each byte of the key is NOT odd, an error will be generated and the key will be ignored.

Request: <SI>02[ADDRESS][MK]<SO>{LRC}

Type	Field	Length	Description
<SI>	Start of Message	1	Start of Message (0x0F)
02	Request Type	2	Load Multi-Master Key
[ADDRESS]	Parameter	1	Multi-Master Key Number ‘0’ (0x30) to ‘9’ (0x39)
[MK]	Parameter	16 or 32	Master Key (hexadecimal)
<SO>	End of Message	1	End of Message (0x0E)
{LRC}		1	Error Check Character

Request Example: Load Special Master Key #3 with 23AB 4589 EF67 01CD.

ASCII: **<SI>**02323AB4589EF6701CD**<SO>**9

Hex: 0F 30 32 33 32 33 41 42 34 35 38 39 45 46 36 37 30 31 43 44 0E 39

```
<SI>        0F
Request     30 32                                           (02)
[ADDRESS]   33                                              (3)
[MK]        32 33 41 42 34 35 38 39 45 46 36 37 30 31 43 44 (23AB4589EF6701CD)
<SO>        0E
{LRC}       39                                              (9)
```

Response: The received Master Key Command is literally echoed back to the PC as verification that the key was accepted. If there is anything wrong in the command format, the command will be ACKed but no additional information will be returned. The Multi-Master Key is stored when the response has been ACKed by the PC. After the ACK (or if no ACK is received within 15 seconds), the IntelliPIN sends an EOT.

Response Example: It will be exactly the same as the Request Example, followed by an EOT.

04 CHECK MULTI-MASTER KEY

Command Set: Multi-Master Key

Purpose: To determine if a key is stored in one of the 10 Multi-Master Key locations.

Command Notes: The controller sends the request packet to verify if a Master Key is in a specific location. The IntelliPIN checks the key address and sends a response packet back to the controller indicating whether there is or is not a resident Master Key in that memory location. The IntelliPIN has 10 Master Key memory locations. Use 04 before sending new keys to prevent the accidental over-writing of a Master Key.

Request: <SI>04[ADDRESS]<SO>{LRC}

Type	Field	Length	Description
<SI>	Start of Message	1	Start of Message (0x0F)
04	Request Type	2	Check Multi-Master Key
[ADDRESS]	Parameter	1	Multi-Master Key Number '0' (0x30) to '9' (0x39)
<SO>	End of Message	1	End of Message (0x0E)
{LRC}		1	Error Check Character

Request Examples: These examples assume the following:

Key 1 is NOT loaded
Key 3 is loaded.

Request Example #1: Check to ensure Multi-Master Key #3 has been loaded.

ASCII: **<SI>**043**<SO>**9

Hex: 0F 30 34 33 0E 39

```
<SI>        0F
Request     30 34  (04)
[ADDRESS]   33     (3)
<SO>        0E
{LRC}       39     (9)
```

Response: <SI>04[READY]<SO>{LRC}

Type	Field	Length	Description
<SI>	Start of Message	1	Start of Message 0x0F
04	Response Type	2	Check Multi-Master Key Response
[READY]	Parameter	1	Response Code: '0' (0x30)=No Master Key at Address 'F' (0x46)=Master Key at Address
<SO>	End of Message	1	End of Message (0x0E)
{LRC}		1	Error Check Character

After the ACK, the IntelliPIN sends <EOT>.

Response Example #1: Showing that Multi-Master Key #3 has been loaded.

```
ASCII:  <SI>04F<SO>L

Hex:  0F 30 34 46 0E 4C

<SI>      0F
Response  30 34  (04)
[READY]   46     (F)
<SO>      0E
{LRC}     4C     (L)
```

Request Example #2: Check to ensure Multi-Master Key #1 loaded.

```
ASCII:  <SI>041<SO>;

Hex:  0F 30 34 31 0E 3B

<SI>       0F
Request    30 34  (04)
[ADDRESS]  31     (1)
<SO>       0E
{LRC}      3B     (;)
```

Response Example #2: Showing that Multi-Master Key #1 was not loaded.

```
ASCII:  <SI>040<SO>:

Hex:  0F 30 34 30 0E 3A

<SI>     0F
Request  30 34  (04)
[READY]  30     (0)
<SO>     0E
{LRC}    3A     (:)
```

07 DES ALGORITHM RELIABILITY TEST

Command Set: Multi-Master Key

Purpose: To determine if the DES algorithm is operating correctly.

Command Notes: The controller sends a known DES key and test data along with the known encryption value. The IntelliPIN will encrypt the test data using the key and compare this result with the encryption value sent. If the encrypted test data matches the known encryption, the IntelliPIN will display "**DES PASSED**". If the encrypted test data does not match the known encryption, the IntelliPIN will display "**DES FAILED**".

[TSTKEY] is not stored and does not affect normal IntelliPIN operation.

The IntelliPIN will check the parity of [TSTKEY]. If the parity is incorrect (i.e., not odd parity) and the "Check Key Parity Mode" is enabled, the IntelliPIN will display an error and will not complete the test.

The display will show "DES PASSED" or "DES FAILED" until the CLEAR key is pressed or until another command is received that changes the display.

The IntelliPIN will always return an EOT character (0x04).

Request: <SI>07[TSTKEY][TSTDAT][CMPDATA]<SO>{LRC}

Type	Field	Length	Description
<SI>	Start of Message	1	Start of Message (0x0F)
07	Request Type	2	DES Algorithm Reliability Test
[TSTKEY]	Parameter	16	Test Master Key (Hexadecimal)
[TSTDAT]	Parameter	16	Clear text test data (Hexadecimal)
[CMPDAT]	Parameter	16	Encrypted value of [TSTDAT] as encrypted under [TSTKEY] (Hexadecimal)
<SO>	End of Message	1	End of Message (0x0E)
{LRC}		1	Error Check Character

Request Example: Make sure the IntelliPIN's DES algorithm calculates **0611 F087 A5B6 601D** when encrypting the test data **0123 4567 89AB CDEF** using the Master Key of **23AB 4589 EF67 01CD**.

Test Key: **23AB 4589 EF67 01CD**
Test Data: **0123 4567 89AB CDEF**
Encrypted data: **0611 F087 A5B6 601D**

ASCII: **<SI>**0723AB4589EF6701CD0123456789ABCDEF0611F087A5B6601D**<SO>**{0x05}

Hex: 0F 30 37 32 33 41 42 34 35 38 39 45 46 36 37 30 31 43 44 30 31 32 33 34
35 36 37 38 39 41 42 43 44 45 46 30 36 31 31 46 30 38 37 41 35 42 36 36
30 31 44 0E 05

<SI>	0F	
Request	30 37	(07)
[TSTKEY]	32 33 41 42 34 35 38 39 45 46 36 37 30 31 43 44	(23AB4589EF6701CD)
[TSTDAT]	30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46	(0123456789ABCDEF)
[CMPDAT]	30 36 31 31 46 30 38 37 41 35 42 36 36 30 31 44	(0611F087A5B6601D)
<SO>	0E	
{LRC}	05	

The IntelliPIN should display "**DES PASSED**".

Response: This function has no response except the EOT character, which is always sent.
ASCII: <EOT>
Hex: 04

08 SELECT MULTI-MASTER KEY

Command Set: Multi-Master Key

Purpose: To select one of the 10 Master Keys (0-9) stored in the IntelliPIN as the key to use on subsequent Multi-Master transactions.

Command Notes: After the specific Multi-Master Key has been selected, it remains in effect until it is changed by another 08 Command.

Request: <SI>08[ADDRESS]<SO>{LRC}

Type	Field	Length	Description
<SI>	Start of Message	1	Start of Message (0x0F)
08	Request Type	2	Select Multi-Master Key
[ADDRESS]	Parameter	1	Multi-Master Key Number '0' (0x30) to '9' (0x39)
<SO>	End of Message	1	End of Message (0x0E)
{LRC}		1	Error Check Character

Request Example: Choose Special Master Key #3 to be active.

ASCII: **<SI>**083**<SO>**5

Hex: 0F 30 38 33 0E 35

```
<SI>       0F
Request    30 38  (08)
[ADDRESS]  33     (3)
<SO>       0E
{LRC}      35     (5)
```

Response: This function has no response except the EOT character, which is always sent.

```
ASCII:  <EOT>
Hex:    04
```

20 LOAD A FIT TABLE

Command Set: Standalone

Purpose: To load one of twelve FIT tables

Command Notes: The Keys used in this command are encrypted under the Master Key and *must* contain correct parity unless the Key Parity setting in command 50B is set to ignore ("0"). (This feature was added in revision V of 30037367 &30037368 firmware—February 2006.)

Request Format: (for VISA BIN)

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
20	Command Type	2	Load FIT Table
[BIN]	Parameter	6	BIN, left justified with F's (e.g., 1234FF)
[FUNCT]	Parameter	4	Function word (0001 + other parameters) (Hex)
[VALPAD]	Parameter	1	Always '0' (0x30) (Hex)
[VALLEN]	Parameter	2	Always '16' (0x31 0x36) (Dec)
[VALDSP]	Parameter	2	Always '00' (0x30 0x30) (Dec)
[OFFLEN]	Parameter	2	PVV length '04' (0x30 0x34) to '06' (0x30 0x36) (Dec)
[OFFDSP]	Parameter	2	Always '00' (0x30 0x30) (Dec)
[PVKI]	Parameter	1	PVKI '0' (0x30) to '9' (0x39) (Dec)
[KEY1]	Parameter	16	First key* (encrypted under Master Key) (Hex)
[KEY2]	Parameter	16	Second key* (encrypted under Master Key) (Hex)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Format: (for DES BIN)

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
20	Command Type	2	Load FIT Table
[BIN]	Parameter	6	BIN, left justified with 'F's (e.g., '1234FF')
[FUNCT]	Parameter	4	Function word (0004 + other parameters) (Hex)
[VALPAD]	Parameter	1	Validation Pad character ('0'-'F') (Hex)
[VALLEN]	Parameter	2	Validation Length '01' to '16' (Dec)
[VALDSP]	Parameter	2	Validation Displacement '00' to '15' (Dec)
[OFFLEN]	Parameter	2	Offset length '00' to '15' (Dec)
[OFFDSP]	Parameter	2	Offset Displacement '00' to '15' (Dec)
[PVKI]	Parameter	1	Always '0' (0x30) (Dec)
[KEY1]	Parameter	16	First key* (encrypted under Master Key) (Hex)
[DECTAB]	Parameter	16	Decimalization Table (encrypted under Master Key) (Hex)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

* Each key must contain proper odd parity unless the Key Parity Switch in 50B is set to '0'.

Request Format: (for Diebold BIN)*

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
20	Command Type	2	Load FIT Table
[BIN]	Parameter	6	BIN, left justified with F's (e.g., 1234FF)
[FUNCT]	Parameter	4	Function word (0040 + other parameters) (Hex)
[VALPAD]	Parameter	1	Always '0' (Dec)
[TYPE]	Parameter	2	Algorithm type ('00' to '99') (Dec)
[VALDSP]	Parameter	2	Always '00' (0x30 0x30)(Dec)
[OFFLEN]	Parameter	2	Offset length: always '04' (Dec)
[OFFDSP]	Parameter	2	Offset Displacement '00' to '15' (Dec)
[PVKI]	Parameter	1	Always '0' (Dec)
[KEY1]	Parameter	16	Test key: 0101010101010101 (encrypted under Master Key) (Hex)
[KEY2]	Parameter	16	Sixteen 0's (not used) (Hex)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

* In order to use a Diebold BIN, the Diebold Table must be loaded at the MagTek factory.

Function Word Parameter List

The following table describes the bit configuration of the FUNCT Parameters in the preceding tables:

Value	Description
0001	Visa BIN Definition
0002	Count from Field Separator, otherwise use End Sentinel
0004	IBM DES BIN definition
0008	PIN length is 4, otherwise length can be from 4 to 12
0010	Enable key entry of PAN
0020	Perform a Mod-10 check on key-entered PAN
0040	Diebold BIN definition
1000	Add Mod-10 digit to key-entered PAN

Example:

This example is for a DES BIN (0004) of 1234 that allows key-entered PANs (0010) and appends the MOD-10 check digit (1000). The Offset Reference is from Field Separator (0002) and the PIN length is fixed at 4 digits (0008). This configuration gives a function word of 0x101E (0004 + 0010 + 1000 + 0002 + 0008).

The Pad character is D. The validation length is 10 displaced 01 characters to the left of Field Separator. The offset length is 04 and is displaced 11 characters to the right of Field Separator. The PVKI field is not used and is set to 0. The PIN key is 0101010101010101 and the decimalization table is 0123456789012345; both are encrypted under the key of 0101010101010101.

Request Example:

```
ASCII:  <STX>201234FF101ED100104110994D4DC157B96C52EEF27B163AC24D6E<ETX>{0x7F}

Hex:  02 32 30 31 32 33 34 46 46 31 30 31 45 44 31 30 30 31 30 34 31 31 30 39
      39 34 44 34 44 43 31 35 37 42 39 36 43 35 32 45 45 46 32 37 42 31 36 33
      41 43 32 34 44 36 45 03 7F
```

<STX>	02	
Request	32 30	(20)
[BIN]	31 32 33 34 46 46	(1234FF)
[FUNCT]	31 30 31 45	(101E)
[VALPAD]	44	(D)
[VALLEN]	31 30	(10)
[VALDSP]	30 31	(01)
[OFFLEN]	30 34	(04)
[OFFDSP]	31 31	(11)
[PVKI]	30	(0)
[KEY1]	39 39 34 44 34 44 43 31 35 37 42 39 36 43 35 32	(994D4DC157B96C52)
[KEY2]	45 45 46 32 37 42 31 36 33 41 43 32 34 44 36 45	(EEF27B163AC24D6E)
<ETX>	03	
{LRC}	7F	

Response Format: <STX>20[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
20	Command Type	2	Load FIT Table Response
[CS]	Parameter	1 or 2	Confirmation Status (as shown in the table below)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Responses:

Condition	[CS]	Display
No Errors	0 (0x30)	No Change
FIT tables full	x1 (0x78 0x31)	“No More FIT Tables Available”
Transfer card not read	x2 (0x78 0x32)	“Unit has not been initialized”
BIN is invalid	x3 (0x78 0x33)	“Invalid BIN”
Function word invalid	x4 (0x78 0x34)	“Function Word Invalid”
Val data invalid	x5 (0x78 0x35)	“Validation Data Field is Invalid”
Offset or PVV is bad	x6 (0x78 0x36)	“Offset/PVV Field Invalid”
PVKI invalid	x7 (0x78 0x37)	“PVKI not 0-9”
Key 1 has bad parity, is too short, or has invalid characters	x8 (0x78 0x38)	“Bad Key Data”
Key 2 has bad parity, is too short, or has invalid characters	x9 (0x78 0x39)	“Bad Key Data”

Response Example (Everything OK, FIT Loaded Correctly):

ASCII: **<STX>**200**<ETX>**1

Hex: 02 32 30 30 03 31

```
<STX>     02
Response  32 30  (20)
[CS]      30     (0)
<ETX>     03
{LRC}     31     (1)
```

Response Example (Error, Master Key Not Loaded):

ASCII: **<STX>**20x2**<ETX>**K

Hex: 02 32 30 78 32 03 4B

```
<STX>     02
Response  32 30  (20)
[CS]      78 32  (x2)
<ETX>     03
{LRC}     4B     (K)
```

21 DELETE ALL FIT TABLES

Command Set: Standalone

Purpose: To clear all FIT tables

Request: <STX>2100=00<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
21	Command Type	2	Destroy All FIT Information
00=00	Constant	5	Must be present to activate this command
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example:

ASCII: **<STX>**2100=00**<ETX>**=

Hex: 02 32 31 30 30 3D 30 30 03 3D

```
<STX>    02
Request  32 31           (21)
Const    30 30 3D 30 30  (00=00)
<ETX>    03
{LRC}    3D              (=)
```

Response: <STX>21[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
21	Command Type	2	Destroy All FIT Information
[CS]	Parameter	1	Confirmation Status (as shown in the table below)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Responses:

Condition	[CS]	Display
No errors	0 (0x30)	No Change
“00=00” not sent	q (0x71)	“User Data field is Invalid”
FIT data not cleared (e.g., if memory is corrupted)	1 (0x31)	No Change

Response Example (Everything OK, FIT Tables Cleared):

ASCII: **<STX>**210**<ETX>**0

Hex: 02 32 31 30 03 30

<STX>	02	
Response	32 31	(21)
[CS]	30	(0)
<ETX>	03	
{LRC}	30	(0)

23 SET/RETRIEVE DATE

Command Set: Standalone

Purpose: To set the date for verify modes and to retrieve the date.

Command Notes: The date value will not be checked for being a valid date. If the date has not been set yet, either manually or via this command, then the retrieved date will be eight "X"s (0x58), i.e., "XXXXXXXX"

Request: <STX>23[MM][DD][YYYY]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
23	Command Type	2	Date Set/Retrieve Request
The following values are needed only to set the date. To retrieve the date, omit them.			
[MM]	Parameter	2	Month (two decimal digits, '01' – '12')
[DD]	Parameter	2	Day (two decimal digits, '01' – '31')
[YYYY]	Parameter	4	Year (four decimal digits, e.g. '1998')
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Set the date to January 23, 1998 (01231998)

```
ASCII:  <STX>2301231998<ETX>{0x0B}

Hex:  02 32 33 30 31 32 33 31 39 39 38 03 0B

<STX>    02
Request  32 33        (23)
[MM]     30 31        (01)
[DD]     32 33        (23)
[YYYY]   31 39 39 38  (1998)
<ETX>    03
{LRC}    0B
```

To retrieve the date, command 23 is sent without any parameters:

```
ASCII:  <STX>23<ETX>{0x02}

Hex:  02 32 33 03 02

<STX>     02
Request:  32 33  (23)
<ETX>     03
{LRC}     02
```

Response: <STX>23[CS][MM][DD][YYYY]<ETX>{LRC}

Field	Type	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
23	Command Type	2	Date Response
[CS]	Parameter	1	Confirmation value as shown below
The following values are returned only if the date is retrieved			
[MM]	Parameter	2	Month (two decimal digits)
[DD]	Parameter	2	Day (two decimal digits)
[YYYY]	Parameter	4	Year (four decimal digits)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No errors	0 (0x30)	No change
Non-decimal values in date or is not eight digits	q (0x71)	"User Data field is invalid"

Response Example (After Set Request):

ASCII: **<STX>**230**<ETX>**2

Hex: 02 32 33 30 03 32

```
<STX>     02
Response  32 33  (23)
[CS]      30     (0)
<ETX>     03
{LRC}     32     (2)
```

Response Example (After Retrieve Request):

ASCII: **<STX>**23001231998**<ETX>**;

Hex: 02 32 33 30 30 31 32 33 31 39 39 38 03 3B

```
<STX>     02
Response  32 33        (23)
[CS]      30           (0)
[MM]      30 31        (01)
[DD]      32 33        (23)
[YYYY]    31 39 39 38  (1998)
<ETX>     03
{LRC}     3B           (;)
```

If the date has not been set, the response will be:

<STX>230XXXXXXXX<ETX>2

24 REMOTE PASSWORD ENTRY

Command Set: Standalone

Purpose: To activate the IntelliPIN from the host computer

Command Notes: This command is valid only when:

a) the IntelliPIN is in the "Activate with password only" mode

AND

b) the IntelliPIN is shut down and display "Unit is Shut Down Pswd"

If the above conditions are not met, the command is ignored.

If five bad passwords are sent (consecutively), the IntelliPIN displays "Too many retries on password", beeps twice and reverts to the "card and password" mode forcing the user to also use an activate card.

Request: <STX>24[PSWD]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
24	Command Type	2	Password Request
[PSWD]	Parameter	4	Four digit password (dec)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Send password of "7638"

ASCII: **<STX>**247638**<ETX>**{0x0F}

Hex: 02 32 34 37 36 33 38 03 0F

```
<STX>    02
Request  32 34        (24)
[PSWD]   37 36 33 38  (7638)
<ETX>    03
{LRC}    0F
```

Response: <STX>24[CS]<ETX>{LRC}

Field	Type	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
24	Command Type	2	Remote Password Response
[CS]	Parameter	1	Confirmation value as shown below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	"New Password accepted"
Invalid password or incorrect (but valid) password sent	q (0x71)	"Passwords do not match"
Not in mode where this command is valid	1 (0x31)	No change

Response Example: If everything is ready to accept this password, then the response is:

ASCII: **<STX>**240**<ETX>**{35}

Hex: 02 32 34 30 03 35

```
<STX>     02
Response  32 34  (24)
[CS]      30     (0)
<ETX>     03
{LRC}     35     (5)
```

30 PIN ENTRY REQUEST

Command Set: Master/Session Key

Purpose: To obtain a PIN from the customer.

Command Notes: The IntelliPIN will display the messages below until the customer enters the PIN.

Message 1 **`Total`**
`$xxxxxxxx.xx` (Amount of Sales [AMT])

Message 2 **`Please enter PIN`**
`Then press ENTER`

If the amount field is not present, the display will show message 2 only.

The 16-digit PIN block, encrypted under the selected key [KN], will be returned to the host. The host can then decrypt the PIN block to recover the PIN.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned in place of the response. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message and restarts IntelliPIN entry.

Request 72 from the PC can cancel the operation and return to the idle state.

After a PIN has been entered, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

The format of the PIN block is set by Soft Switch B (See Command 50).

Request: <STX>30[ACCT]<FS>[KN1][AMT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
30	Request Type	2	PIN Entry Request
[ACCT]	Parameter	0 - 19	Account Number (decimal) (Optional)
<FS>	Separator	1	Field Separator (0x1c)
[KN]	Parameter	1	Key: • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key
[AMT]	Parameter	0 or 3 - 12	Transaction Amount in cents (decimal) (omit the decimal point and commas) (Optional)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Collect the PIN with an Account Number of 4761234567812348 using the Master Key (key number 4) and displaying the amount of $1.23. The check digit of the account number (final digit) is removed by the IntelliPIN before generating the PIN Block.

```
ASCII:  <STX>304761234567812348<FS>4123<ETX>{0x19}

Hex:  02 33 30 34 37 36 31 32 33 34 35 36 37 38 31 32 33 34 38 1C 34 31 32 33
      03 19

<STX>    02
Request  33 30                                                  (30)
[ACCT]   34 37 36 31 32 33 34 35 36 37 38 31 32 33 34 38        (4761234567812348)
<FS>     1C
[KN]     34                                                     (4)
[AMT]    31 32 33                                               (123)
<ETX>    03
{LRC}    19
```

Response: <STX>30[CS][EPIN]<ETX>LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
30	Response Type	2	PIN Entry Request
[CS]	Parameter	1	Confirmation Value as shown below
[EPIN]	Parameter	16	Encrypted PIN block (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = '0' when the selected key is valid and request parameters are correct.
- [EPIN] (the Encrypted PIN block) will only be returned if the [CS] value is '0'.
- If the CLEAR key is pressed as the first key, an EOT (0x04) will be returned (not the whole response).

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	"PINPad is Processing"
Account Number contains non-decimal digits or is more than 19 digits	A (0x61)	"Account Number field is invalid"
No Field Separator found	G (0x67)	"Missing Field Separator"
Key number is not present or not correct	S (0x73)	"Value should be 0-5, A-Z or a-z"
Amount Field contains non-decimal digits or Is less than 3 digits or is more than 12 digits	B (0x62)	"Amount Field is invalid"
Requested Key has not been loaded	M (0x6d)	"Selected Key not Ready"

With the PIN block format set to ANSI 9.8, and using the example Master Key of `23AB 4589 EF67 01CD`, and a PIN entry of `6565`, the result will be as follows:

Response Example: Received encrypted PIN block of `D5D6 DF8D 0DB8 97AB`

ASCII: **<STX>**300D5D6DF8D0DB897AB**<ETX>**N

Hex: 02 33 30 30 44 35 44 36 44 46 38 44 30 44 42 38 39 37 41 42 03 4E

```
<STX>     02
Response  33 30                                          (30)
[CS]      30                                             (0)
[EPIN]    44 35 44 36 44 46 38 44 30 44 42 38 39 37 41 42  (D5D6DF8D0DB897AB)
<ETX>     03
{LRC}     4E                                             (N)
```

Example: The following example is for recovering the PIN from an encrypted PIN block.

Encrypted PIN Block from the IntelliPIN response	D5D6	DF8D	0DB8	97AB
Master Key used by the IntelliPIN for this example	23AB	4589	EF67	01CD
Decrypting the PIN block using the Master Key yields	0465	77CB	A987	EDCB
The account number is processed as follows before using:				
1. Remove the check-digit (the final digit of this account number) (4761234567812348)	----	----	----	----
2. Insert up to the 12 rightmost digits of the remaining values into the field, right justified (476**123456781234**)	----	1234	5678	1234
3. Pad any empty digits to the left with zeros until the field is 16 digits long. This is the "Padded Account Number"	**0000**	1234	5678	1234
Exclusive Oring (XORing) the Decrypted PIN Block with the Padded Account Number yields:	**0465**	**65**FF	FFFF	FFFF
Remove all the F's. The recovered PIN is 4 digits (4) long with a value of "6565"	**04 6565**			

31 PIN OFFSET REQUEST

Command Set: Master/Session Key

Purpose: To obtain a PIN offset from the customer.

Command Notes: The display will show **`Please enter PIN then press ENTER`** until the customer enters the PIN.

The Validation Data [VALDAT] is encrypted under Key Number [KN] and is then replaced using the substitution table (loaded by request 57). Then the result is subtracted from the PIN entered (zero filled to the right) using mod 10 or 16 subtraction (mod 10 for decimal table, mod 16 for hexadecimal table). The resulting encrypted 16-digit offset is returned to the PC.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned instead of the response message. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message and restarts IntelliPIN entry.

Request 72 from the PC can cancel the operation and return to the idle state.

After a PIN has been entered (once or twice depending on the setup), the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

Activation: (See Appendix E for flow diagrams)

- DEACTIVATED at power up.
- Activation can be toggled by request 58.

NOTE

If the Offset/Verify commands have not been activated, the following actions need to be taken:
1. Load the Master Key (see the command 94 example).
2. Load the Key Serial Number (see the command 97 example).
3. Activate the Offset/Verify commands (see the command 58 example).

Request: <STX>31[KN][VALDAT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
31	Request Type	2	PIN Offset Request
[KN]	Parameter	1	Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key
[VALDAT]	Parameter	16	Validation Data (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example (Single-length Key): Collect the PIN offset with a Validation Data of 0122 1960 FFFF FFFF using the Master Key (key number 4).

ASCII: **<STX>**31401221960FFFFFFFF**<ETX>**:

Hex: 02 33 31 34 30 31 32 32 31 39 36 30 46 46 46 46 46 46 46 46 03 3A

```
<STX>     02
Request   33 31                                          (31)
[KN]      34                                             (4)
[VALDAT]  30 31 32 32 31 39 36 30 46 46 46 46 46 46 46 46  (01221960FFFFFFFF)
<ETX>     03
{LRC}     3A                                             (:)
```

Response: <STX>31[CS][OFFSET]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
31	Response Type	2	PIN Offset Response
[CS]	Parameter	1	Confirmation Value as shown below
[OFFSET]	Parameter	16	Offset (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = '0' when the selected key is valid and request parameters are correct.
- [OFFSET] will only be returned if the [CS] value is '0'.
- If the CLEAR key is pressed as the first key, an EOT (0x04) will be returned (not the whole response).

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	"PINPad is processing"
Key number is not present or not correct	s (0x73)	"Value should be 0-5, A-Z or a-z"
Validation Data [VALDAT] is non-hex or is not 16 digits	r (0x72)	"Validation Data field is invalid"
Command is inactive (needs request 58 first)	d (0x64)	"Command not Activated"
Requested Key [KN] has not been loaded	m (0x6D)	"Selected Key not Ready"
Substitution Table not loaded yet	o (0x6F)	"Substitution Table not Ready"

Response Example (Single-length Key): With a substitution table of 0123 4567 8901 2345, (see command 57 for decimal example) and with the sample Master Key of 23AB 4589 EF67 01CD, and a PIN of 1234, the response will be as follows:

Received offset data of `0165 0500 9769 4927`

ASCII: **<STX>** 3100165050097694927**<ETX>**?

Hex: 02 33 31 30 30 31 36 35 30 35 30 30 39 37 36 39 34 39 32 37 03 3F

```
<STX>     02
Response  33 31                                          (31)
[CS]      30                                             (0)
[OFFSET]  30 31 36 35 30 35 30 30 39 37 36 39 34 39 32 37  (0165050097694927)
<ETX>     03
{LRC}     3F                                             (?)
```

With this response, the offset, as placed on Track 2 of the card, would be 0165 (the first four digits).

Request Example (Double-Length Key): Collect PIN offset with a Validation Data of **0122 1960 FFFF FFFF** using the Master Key (see the command 94 double-length key example) and a substitution table of **0F1E 2D3C 4B5A 6978** (see the command 57 hex example). Enter a PIN of **1234**.

ASCII: **<STX>**31401221960FFFFFFFF**<ETX>**:

Hex: 02 33 31 34 30 31 32 32 31 39 36 30 46 46 46 46 46 46 46 46 03 3A

```
<STX>     02
Request   33 31                                          (31)
[KN]      34                                             (4)
[VALDAT]  30 31 32 32 31 39 36 30 46 46 46 46 46 46 46 46  (01221960FFFFFFFF)
<ETX>     03
{LRC}     3A                                             (:)
```

Response Example (Double-Length Key):

ASCII: **<STX>**31031B91CCB9D629F69**<ETX>**2

Hex: 02 33 31 30 33 31 42 39 31 43 43 42 39 44 36 32 39 46 36 39 03 32

<STX>	02	
Response:	33 31	(31)
[CS]	30	(0)
[OFFSET]	33 31 42 39 31 43 43 42 39 44 36 32 39 46 36 39	(31B91CCB9D629F69)
<ETX>	03	
{LRC}	32	(2)

32 PIN VERIFICATION REQUEST

Command Set: Master/Session Key

Purpose: To verify the customer's PIN.

Command Notes: This command compares a PIN offset generated in the IntelliPIN with the PIN offset in the request message.

The display will show **`Please enter PIN then press ENTER`** until the customer enters the PIN.

After the Enter key is pressed, the IntelliPIN will calculate the offset and compare it to the offset in the request. The IntelliPIN will return Y if they match or N if they do not match.

The Validation Data [VALDAT] is encrypted under key [KN] and this encrypted data will be replaced using the substitution table (loaded by command 57). Then the result is subtracted from the PIN (0 (zero) filled to the right) using mod 10 or 16 subtraction (mod 10 for decimal table, mod 16 for hexadecimal table). The first "n" digits of the results are compared to the "n" digits of the offset from the PC [OFFSET]. If the first "n" digits are the same, the IntelliPIN will return a **`Y`** for yes. If they are not the same, the IntelliPIN will return an **`N`** for no.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned instead. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message and restarts IntelliPIN entry.

Request 72 from the PC can cancel the operation and return to the idle state.

After a PIN has been entered, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

Activation: (See Appendix E for flow diagrams)

- DEACTIVATED at power up.
- Activation can be toggled by request 58.

NOTE

If the Offset/Verify commands have not been activated, the following actions need to be taken:
1. Load the Master Key (see the command 94 example).
2. Load the Key Serial Number (see the command 97 example).
3. Activate the Offset/Verify commands (see the command 58 example).

Request: <STX>32[KN][VALDAT][OFFSET]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
32	Request Type	2	PIN Verification Request
[KN]	Parameter	1	Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key
[VALDAT]	Parameter	16	Validation Data (hexadecimal)
[OFFSET]	Parameter	1-16	Offset (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example (Single-length Key): Compare the PIN offset with a Validation Data of 0122 1960 FFFF FFFF and an Offset of 0165 using the Master Key (key number 4).

ASCII: **<STX>**32401221960FFFFFFFF0165**<ETX>**;

Hex: 02 33 32 34 30 31 32 32 31 39 36 30 46 46 46 46 46 46 46 46 30 31 36 35 03 3B

<STX>	02	
Request	33 32	(32)
[KN]	34	(4)
[VALDAT]	30 31 32 32 31 39 36 30 46 46 46 46 46 46 46 46	(01221960FFFFFFFF)
[OFFSET]	30 31 36 35	(0165)
<ETX>	03	
{LRC}	3B	(;)

Response: <STX>32[CS][Y/N]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
32	Response Type	2	PIN Verification Response
[CS]	Parameter	1	Confirmation Value as shown below
[Y/N]	Parameter	1	Offsets match? • 'Y' (0x59) = Yes, the offsets match • 'N' (0x4E) = No, the offsets are different
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = '0' when the key is valid and request parameters are correct.
- [Y/N] will only be returned if the [CS] value is '0'.
- If the CLEAR key is pressed as the first key, an EOT (0x04) will be returned (not the whole response).

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	“PINPad is processing”
Key number is not present or not correct	s (0x73)	“Value should be 0-5, A-Z or a-z”
Offset data is non-hex or is not 1-16 digits	q (0x71)	“User Data field is invalid”
Validation Data is non-hex or is not 16 digits	r (0x72))	“Validation Data field is invalid”
Command is inactive (needs request 58 first)	d (0x64)	“Command not Activated”
Requested key has not been loaded	m (0x6D)	“Selected Key not Ready”
Substitution Table not loaded yet	o (0x6F)	“Substitution Table not Ready”

Response Example: With a substitution table of 0123 4567 8901 2345, (see command 57 for decimal example) and with the sample Master Key of 23AB 4589 EF67 01CD, and a PIN of 1234, the response will be as follows:

Enter a PIN of 1234 and the following response should be received:

ASCII: **<STX>**320Y**<ETX>**k

Hex: 02 33 32 30 59 03 6B

```
<STX>     02
Response  33 32  (32)
[CS]      30     (0)
[Y/N]     59     (Y)
<ETX>     03
{LRC}     6B     (k)
```

33 ENCRYPTION TEST REQUEST

Command Set: Master/Session Key

Purpose: To encrypt and return the supplied data under a given key.

Command Notes: The data [DATA] is encrypted under the key [N] and the result is returned to the PC.

Activation: (See Appendix E for flow diagrams)

- DEACTIVATED at power up.
- Activation can be toggled by request 58.

NOTE

If the Offset/Verify commands have not been activated, the following actions need to be taken:
1. Load the Master Key (see the command 94 example).
2. Load the Key Serial Number (see the command 97 example).
3. Activate the Offset/Verify commands (see the command 58 example).

Request: `<STX>33[KN][DATA]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
33	Request Type	2	Encryption Test Request
[KN]	Parameter	1	• Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key
[DATA]	Parameter	16	Data (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Encrypt `0465 51CB A987 EDCB` under Working Key 1 (Key Number 1) of 4FF4 4FF4 4FF4 4FF4.

```
ASCII:  <STX>331046551CBA987EDCB<ETX>G

Hex:  02 33 33 31 30 34 36 35 35 31 43 42 41 39 38 37 45 44 43 42 03 47
```

<STX>	02	
Request	33 33	(33)
[KN]	31	(1)
[DATA]	30 34 36 35 35 31 43 42 41 39 38 37 45 44 43 42	(046551CBA987EDCB)
<ETX>	03	
{LRC}	47	(G)

Response: <STX>33[CS][DATA]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
33	Response Type	2	Encryption Test Request
[CS]	Parameter	1	Confirmation Value as shown below
[DATA]	Parameter	16	Encrypted Data (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = '0' only if the key is valid, the request parameters are correct and this request has been activated.
- [DATA] will be present only if [CS] = 0.

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	No change
Key number is not present or not correct	s (0x73)	"Value should be 0-5, A-Z or a-z"
Requested key has not been loaded	m (0x6D)	"Selected Key not Ready"
User data is non-hex or is not 16 digits	q (0x71)	"User Data field is invalid"
Request inactive (needs request 58 first)	d (0x64)	"Command not Activated"

Response Example: Received encrypted data of `1FD4 13E4 CBA6 48F1`

```
ASCII:  <STX>3301FD413E4CBA648F1<ETX>J

Hex:  02 33 33 30 31 46 44 34 31 33 45 34 43 42 41 36 34 38 46 31 03 4A
```

<STX>	02	
Response	33 33	(33)
[CS]	30	(0)
[DATA]	31 46 44 34 31 33 45 34 43 42 41 36 34 38 46 31	(1FD413E4CBA648F1)
<ETX>	03	
{LRC}	4A	(J)

34 CVV REQUEST

Command Set: Master/Session Key

Purpose: To obtain the CVV from the card.

Command Notes: The display will show the idle message since there is no input required from the user.

Activation: (See Appendix E for flow diagrams)

- DEACTIVATED at power up.
- Activation can be toggled by request 58.

NOTE

If the Offset/Verify commands have not been activated, the following actions need to be taken:
1. Load the Master Key (see the command 94 example).
2. Load the Key Serial Number (see the command 97 example).
3. Activate the Offset/Verify commands (see the command 58 example).

Request: `<STX>34[KN1][KN2][LEN][DATA1][DATA2]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
34	Command Type	2	CVV Request
[KN1]	Parameter	1	Left Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key
[KN2]	Parameter	1	Right Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key • **'*'** (0x2A) Use double-length KN1
[LEN]	Parameter	1	Length of CVV, '1' to '9' (0x31 to 0x39)
[DATA1]	Parameter	16	Data block 1 (decimal)
[DATA2]	Parameter	16	Data block 2 (decimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Assuming the following:

Master Key: 23AB 4589 EF67 01CD
Session Key: F48A 40B3 1004 9D75
Serial Number: 0123 4567 89AB CDEF
PAN: 5499 7500 0000 0007
Expiration Date: 9912
Service Code: 101

And CVV is generated using the Master Key and the Session Key then the command is:

ASCII: **<STX>**34456549975000000000799121010000000**<ETX>**4

Hex: 02 33 34 34 35 36 35 34 39 39 37 35 30 30 30 30 30 30 30 30 30 37 39 39
31 32 31 30 31 30 30 30 30 30 30 30 30 30 03 34

<STX>	02	
Request	33 34	(34)
[KN1]	34	(4)
[KN2]	35	(5)
[LEN]	36	(6)
[DATA1]	35 34 39 39 37 35 30 30 30 30 30 30 30 30 30 37	(5499750000000007)
[DATA2]	39 39 31 32 31 30 31 30 30 30 30 30 30 30 30 30	(9912101000000000)
<ETX>	03	
{LRC}	34	(4)

Response: <STX>34[CS][CVV]<ETX>{LRC}

Field	Type	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
34	Command Type	2	CVV Response
[CS]	Parameter	1	Confirmation value as shown below
[CVV]	Parameter	0 to 9	CVV (decimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = '0' only if the key is valid, the command parameters are correct and this command has been activated.
- [CVV] is returned only if the [CS] value is '0'
- If the CLEAR is pressed as the first key, only an EOT (0x04) will be returned.

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	“PINPad is processing”
Command is inactive (needs request 58 first)	d (0x64)	“Command not Activated”
Requested key has not been loaded	m (0x6D)	“Selected Key not Ready”
Verification data is invalid (non-decimal or not 32 digits)	q (0x71)	“Account Number field is invalid”
Key number is not present or not correct	s (0x73)	“ Value should be 0-5, A-Z or a-z”
CVV Length [LEN] not 1 through 9	s (0x73)	“Length value is incorrect”
Both Keys are “*”	s (0x73)	“Bad Key Data”

Response Example:

ASCII: **<STX>**340156707**<ETX>**6

Hex: 02 33 34 30 31 35 36 37 30 37 03 36

```
<STX>     02
Response  33 34              (34)
[CS]      30                 (0)
[CVV]     31 35 36 37 30 37  (156707)
<ETX>     03
{LRC}     36                 (6)
```

35 PVV REQUEST

Command Set: Master/Session Key

Purpose: To obtain the PVV from a PIN entered by the user.

Command Notes: The display will show **`Please enter PIN then press Enter`** until the customer enters a PIN.

The user-entered PIN along with the data block are encrypted using a double-length key as defined in the PVV algorithm. This value is then returned to the PC.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned instead of the response message. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message and restarts IntelliPIN entry.

Activation: (See Appendix E for flow diagrams)

- DEACTIVATED at power up.
- Activation can be toggled by request 58.

NOTE

If the Offset/Verify commands have not been activated, the following actions need to be taken:
1. Load the Master Key (see the command 94 example).
2. Load the Key Serial Number (see the command 97 example).
3. Activate the Offset/Verify commands (see the command 58 example).

Request: <STX>35[KN1][KN2][LEN][DATA]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
35	Command Type	2	PVV Request
[KN1]	Parameter	1	Left Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key
[KN2]	Parameter	1	Right Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key • **'*'** (0x2A) Use double-length KN1
[LEN]	Parameter	1	Length of PVV, '1' to '9' (0x31 to 0x39)
[DATA]	Parameter	12	[11 digits of PAN][PVKI (one digit)] both decimal
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Assuming the following:

Master Key: 23AB 4589 EF67 01CD
Session Key: F48A 40B3 1004 9D75
Serial Number: 0123 4567 89AB CDEF
PAN: 4267 **3896 5700 264**5 (only the highlighted characters are used)
PVKI: 2
PIN: 73 29 83

And PVV is generated using the Master Key and the Session Key then the command is:

ASCII: **<STX>**35456389657002642**<ETX>**6

Hex: 02 33 35 34 35 36 33 38 39 36 35 37 30 30 32 36 34 32 03 36

<STX>	02	
Request	33 35	(35)
[KN1]	34	(4)
[KN2]	35	(5)
[LEN]	36	(6)
[DATA]	33 38 39 36 35 37 30 30 32 36 34 32	(389657002642)
<ETX>	03	
{LRC}	36	(6)

Response: <STX>35[CS][PVV]<ETX>{LRC}

Field	Type	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
35	Command Type	2	PVV Response
[CS]	Parameter	1	Confirmation value as shown below
[PVV]	Parameter	0 to 9	PVV (decimal) '0' (0x30) to '9' (0x39)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = '0' only if the key is valid, the command parameters are correct and this command has been activated.
- [PVV] is returned only if the [CS] value is '0'
- If the CLEAR is pressed as the first key, only an EOT (0x04) will be returned

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	"PINPad is processing"
Command is inactive (needs request 58 first)	d (0x64)	"Command not Activated"
Selected key not ready	m (0x6D)	"Selected Key not Ready"
Data invalid (non-decimal or not 12 digits)	q (0x71)	"Account Number field is invalid"
Key number is not present or not correct	s (0x73)	"Value should be 0-5, A-Z or a-z"
PVV Length [LEN] not 1 through 9	s (0x73)	"Length value is incorrect"

Response Example:

Enter "732983" <ENTER> on the IntelliPIN's keypad twice.

The response should be:

ASCII: **<STX>**350757063**<ETX>**5

Hex: 02 33 35 30 37 35 37 30 36 33 03 35

```
<STX>     02
Response  33 35              (35)
[CS]      30                 (0)
[PVV]     37 35 37 30 36 33  (757063)
<ETX>     03
{LRC}     35                 (5)
```

36 PVV VERIFICATION REQUEST

Command Set: Master/Session Key

Purpose: To verify a supplied PVV.

Command Notes: The PVV is calculated using the supplied data and then is verified against the supplied PVV. A 'Y' (for Yes it is valid) or a 'N' (for No, it is not valid) is then returned to the PC.

Activation: (See Appendix E for flow diagrams)

- DEACTIVATED at power up.
- Activation can be toggled by request 58.

NOTE

If the Offset/Verify commands have not been activated, the following actions need to be taken:
1. Load the Master Key (see the command 94 example).
2. Load the Key Serial Number (see the command 97 example).
3. Activate the Offset/Verify commands (see the command 58 example).

Request: <STX>36[KN1][KN2][DATA][PVV]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
36	Command Type	2	PVV Verification Request
[KN1]	Parameter	1	Left Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key
[KN2]	Parameter	1	Right Key • '0' to '3' (0x30 to 0x33) = lower Working Key • '4' (0x34) = Master Key • '5' (0x35) = Session Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key • **'*'** (0x2A) Use double-length KN1
[DATA]	Parameter	12	[PAN (11 digits)][PVKI (one digit)]
[PVV]	Parameter	1 to 9	PVV to be verified (decimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Assuming the following:

Master Key: 23AB 4589 EF67 01CD
Session Key: F48A 40B3 1004 9D75
Serial Number: 0123 4567 89AB CDEF
PAN: 4267 **3896 5700 264**5 (only the highlighted characters are used)
PVKI: 2
PIN: 73 29 83
PVV 757063

And PVV is generated using the Master Key and the Session Key then the command is:

ASCII: **<STX>**3645389657002642757063**<ETX>**{0x03}

Hex: 02 33 36 34 35 33 38 39 36 35 37 30 30 32 36 34 32 37 35 37 30 36 33 03 03

<STX>	02	
Request	33 36	(36)
[KN1]	34	(4)
[KN2]	35	(5)
[DATA]	33 38 39 36 35 37 30 30 32 36 34 32	(389657002642)
[PVV}	37 35 37 30 36 33	(757063)
<ETX>	03	
{LRC}	03	

Response: <STX>36[CS][Y/N]<ETX>{LRC}

Field	Type	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
36	Command Type	2	PVV Verification Response
[CS]	Parameter	1	Confirmation value as shown below
[Y/N]	Parameter	1	PVV is valid? 'Y' (0x59) = Yes, PVV is valid 'N' (0x4E) = No, the PVV does not match the data
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = '0' only if the key is valid, the command parameters are correct and this command has been activated.
- [PVV] is returned only if the [CS] value is '0'
- If the CLEAR is pressed as the first key, only an EOT (0x04) will be returned

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	"PINPad is processing"
Command is inactive (needs request 58 first)	d (0x64)	"Command not activated"
The key [KN1] or [KN2] not loaded	m (0x6D)	"Selected Key not read"
Verification data [DATA] not invalid	q (0x71)	"Account Number field is invalid"
PVV value not valid	r (0x72)	"Validation Data field is invalid"
A key number [KN1] or [KN2] is invalid or Both keys are set to "use the other" (*)	s (0x73)	"Value should be 0-5, A-Z or a-z"

Response Example:

Enter "732983" <ENTER> on the IntelliPIN's keypad once.

The response should be:

ASCII: **<STX>**360Y**<ETX>**O

Hex: 02 33 36 30 59 03 6F

```
<STX>     02
Response  33 36  (36)
[CS]      30     (0)
[PVV]     59     (Y)
<ETX>     03
{LRC}     6F     (o)
```

37 IDENTIKEY PIN OFFSET REQUEST

Command Set: n/a

Purpose: To obtain the IdentiKey PIN offset from the customer.

Command Notes: The display will show **`Please enter PIN then press ENTER`** until the customer enters the PIN.

After the ENTER key is pressed, the IntelliPIN will calculate and return the IdentiKey offset.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned instead of the response message. The IntelliPIN will display **`Cancel Requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message allowing the user to re-enter the PIN.

Request 72 from the PC will cancel the operation and return to the idle state.

After a PIN has been entered, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is pressed or another request is received.

Activation: not needed

Request format: `<STX>37[BINLEN][BIN][PVNSIZ][PVNTYP][VALDSP][VALLEN][PAN]<ETX>{LRC}`

Type	Field	Length	Description
<STX>		1	Start of Text (0x02)
37	Command Type	2	IdentiKey PIN Offset Request
[BINLEN]	Parameter	1	[BIN] length (numeric) must be ‘2’ (0x32), ‘6’ (0x36) or ‘8’ (0x38)
[BIN]	Parameter	2, 6 or 8	Bank ID (numeric) ‘2’, ‘6’, or ‘8’ Length must match [BINLEN]
[PVNSIZ]	Parameter	1	PVN Size must be ‘4’ (0x34), ‘6’ (0x36) or ‘8’ (0x38)
[PVNTYP]	Parameter	1	PVN Type (Numeric) must be ‘1’, ‘2’ or ‘3’ ‘1’ (0x31) = Left ‘2’ (0x32) = Middle ‘3’ (0x33) = Right
[VALDSP]	Parameter	1	Validation displacement (Hexadecimal)
[VALLEN]	Parameter	1	Validation length (Hexadecimal)
[ACCT]	Parameter	1-19	Account Number in decimal
<ETX>		1	End of Text (0x03)
{LRC}		1	Error Check Character

Details on the [PVNSIZ], [PVNTYP], [VALDSP] and [VALLEN] fields:

The [PVNSIZ] and [PVNTYP] determine how much of the eight-digit IdentiKey offset is returned to the host.

- If [PVNSIZ] is '8' then all eight digits are returned.
- If [PVNSIZ] is '6' then the middle six digits are returned.
- If [PVNSIZ] is '4' then the [PVNTYP] field is used to extract a subset of the middle six digits of the offset.
- PVNTYP is only used if [PVNSIZ] is '4' but must be present and valid.
- VALDSP is the displacement starting from the end of [ACCT]. A value of 0 starts at the character immediately before the field separator. A value of A (10) skips the last 10 digits of the account number.
- VALLEN indicates how many digits are included starting from the end of [ACCT] - [VALDSP]. A value of 0 will be interpreted at 16.

The table below summarizes the possible values with an offset in the sequence of "ABCDEFGH". (Of course, the real offset will be all digits.)

[PVNSIZ]	[PVNTYP]	Offset	Comments
8	1	ABCDEFGH	[PVNSIZ] is 8 so [PVNTYP] is ignored and all 8 digits returned
8	2	ABCDEFGH	[PVNSIZ] is 8 so [PVNTYP] is ignored and all 8 digits returned
8	3	ABCDEFGH	[PVNSIZ] is 8 so [PVNTYP] is ignored and all 8 digits returned
6	1	BCDEFG	[PVNSIZ] is 6 so [PVNTYP] is ignored and the middle 6 digits are returned
6	2	BCDEFG	[PVNSIZ] is 6 so [PVNTYP] is ignored and the middle 6 digits are returned
6	3	BCDEFG	[PVNSIZ] is 6 so [PVNTYP] is ignored and the middle 6 digits are returned
4	1	BCDE	[PVNSIZ] is 4 and [PVNTYP] = 1 (LEFT) so the left-most 4 digits of the middle 6 digits are returned
4	2	CDEF	[PVNSIZ] is 4 and [PVNTYP] = 2 (MIDDLE) so the middle 4 digits of the middle 6 digits are returned
4	3	DEFG	[PVNSIZ] is 4 and [PVNTYP] = 3 (RIGHT) so the right-most 4 digits of the middle 6 digits are returned

Response format: <STX>37[CS][OFFSET]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
37	Response Type	2	PIN Offset Verification Response
[CS]	Parameter	1-2	Confirmation value: 0 (0x30) = no error in request xn (0x78 0x3?) = error 'n' (see Confirmation Values table below)
[OFFSET]	Parameter	1	Identikey Offset (only returned if [CS] = 0
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	“PINPad is processing”
[ACCT] bad	x1 (0x78 0x31)	“Account Number field is invalid”
[PVNSIZ] or [PVNTYP] bad	x2 (0x78 0x32)	“PVN size or type bad”
[BINLEN] or [BIN] bad	x3 (0x78 0x33)	“Invalid BIN”
[VALDSP] or [VALLEN] bad	x5 (0x78 0x35)	“Validation Data field is invalid”

Request example: Assumes the following:

Field	Parameter	Comments
[BINLEN]	6	The [BIN] field is six digits long
[BIN]	267121	
[PVNSIZ]	8	All eights digits of the offset will be returned
[PVNTYP]	1	This value will not be used but it is valid
[VALDSP]	0	The validation data’s last digit will be the last digit of [ACCT]
[VALLEN]	4	The validation data is four digits long
[ACCT]	123456789	
PIN	1234	Enter this PIN on the IntelliPIN

ASCII: **<STX>**37626712181041234567 89**<ETX>**{0x0C}

Hex: 02 33 37 36 32 36 37 31 32 31 38 31 30 34 31 32 33 34 35 36 37 38 39 03 0C

```
<STX>     02
Request   33 37                        (37)
[BINLEN]  36                           (6)
[BIN]     32 36 37 31 32 31            (267121)
[PVNSIZ]  38                           (8)
[PVNTYP]  31                           (1)
[VALDSP]  30                           (0)
[VALLEN]  34                           (4)
[ACCT]    31 32 33 34 35 36 37 38 39   (123456789)
<ETX>     03
{LRC}     0C
```

Response Example:

ASCII: **<STX>**37024774576**<ETX>**{LRC}

Hex: 02 33 37 30 32 34 37 37 34 35 37 36 03 31

```
<STX>     02
Response  33 37                    (37)
[CS]      30                       (0)
[OFFSET]  32 34 37 37 34 35 37 36  (24774576)
<ETX>     03
{LRC}     31                       (1)
```

38 VERIFY IDENTIKEY OFFSET

Command Set: n/a

Purpose: This command compares an IdentiKey PIN offset generated in the IntelliPIN with the IdentiKey PIN offset [REFOFST] in the request message.

Command Notes: The display will show **Please enter PIN then press ENTER** until the customer enters the PIN.

After the ENTER key is pressed, the IntelliPIN will calculate the IdentiKey offset and compare it to the offset in the request [REFOFST]. The IntelliPIN will return a **Y** if they match or an **N** if they do not match.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned instead of the response message. The IntelliPIN will display **Cancel Requested** for two seconds then revert to the **Welcome** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message allowing the user to re-enter the PIN.

Request 72 from the PC will cancel the operation and return to the idle state.

After a PIN has been entered, the IntelliPIN displays **PINPad is processing** until the CLEAR key is pressed or another request is received.

Activation: not needed

Request format:

<STX>38[BINLEN][BIN][PVNSIZ][PVNTYP][VALDSP][VALLEN][PAN]<FS>[REFOFST]<ETX>{LRC}
(See request 37 for explanation of the fields [PVNSIZ], [PVNTYP], [VALDSP] and [VALLEN].)

Type	Field	Length	Description
<STX>		1	Start of Text (0x02)
38	Request Type	2	PIN Offset Verification Request
[BINLEN]	Parameter	1	[BIN] length (numeric) must be '2' (0x32), '6' (0x36) or '8' (0x38)
[BIN]	Parameter	2, 6 or 8	Bank ID Number (numeric) the length of [BIN] must match [BINLEN]
[PVNSIZ]	Parameter	1	PVN Size must be '4' (0x34), '6' (0x36) or '8' (0x38)
[PVNTYP]	Parameter	1	PVN Type (Numeric) must be '1' (0x31), 2 (0x32) or 3 (0x33) '1' = Left '2' = Middle '3' = Right (only used if [PVNSIZ] is '4' but must be present and valid)
[VALDSP]	Parameter	1	Validation displacement (hexadecimal)
[VALLEN]	Parameter	1	Validation length (hexadecimal)
[ACCT]	Parameter	1-19	Account Number (decimal)
<FS>	Field Separator	1	Field Separator (0x1C)
[REFOFST]	Parameter	4, 6 or 8	Reference offset (decimal) the length of [REFOFST] must match [PVNSIZ]
<ETX>		1	End of Text (0x03)
{LRC}		1	Error Check Character

See Command 37 for an explanation of the fields.

Response format: <STX>38[CS][Y/N]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
38	Response Type	2	PIN Offset Verification Response
[CS]	Parameter	1-2	Confirmation value: 0 (0x30) = no error in request Xn (0x78 0x3?) = error 'n' (see Confirmation Values table below)
[Y/N]	Parameter	1	Offsets match? 'Y' (0x59) = Yes, the offsets match 'N' (0x4E) = No, the offsets do not match
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	"PINPad is processing"
[ACCT] bad	x1 (0x78 0x31)	"Account Number field is invalid"
[PVNSIZ] or [PVNTYP] bad	x2 (0x78 0x32)	"PVN size or type bad"
[BINLEN] or [BIN] bad	x3 (0x78 0x33)	"Invalid BIN"
[VALDSP] or [VALLEN] bad	x5 (0x78 0x35)	"Validation Data field is invalid"

Request example: Assuming the following:

[BINLEN]	6
[BIN]	267121
[PVNSIZ]	8
[PVNTYP]	1
[VALDSP]	0
[VALLEN]	4
[ACCT]	123456789
[REFOFST]	24774576
Good PIN	1234

ASCII: **<STX>**38626712181041234567 89**<FS>**24774576**<ETX>**{0x19}

Hex: 02 33 38 36 32 36 37 31 32 31 38 31 30 34 31 32 33 34 35 36 37 38 39 1C
32 34 37 37 34 35 37 36 03 19

```
<STX>      02
Request    33 38                        (38)
[BINLEN]   36                           (6)
[BIN]      32 36 37 31 32 31            (267121)
[PVNSIZ]   38                           (8)
[PVNTYP]   31                           (1)
[VALDSP]   30                           (0)
[VALLEN]   34                           (4)
[ACCT]     31 32 33 34 35 36 37 38 39   (123456789)
<FS>       1C
[REFOFST]  32 34 37 37 34 35 37 36      (24774576)
<ETX>      03
{LRC}      19
```

Response Example: If "1234" was entered as the PIN in the above example, then the following response will be returned:

ASCII: **<STX>**380Y**<ETX>**{LRC}

Hex: 02 33 38 30 59 03 61

```
<STX>     02
Response  33 38  (38)
[CS]      30     (0)
[Y/N]     59     (Y)
<ETX>     03
{LRC}     61     (a)
```

If some other PIN was entered, then the following will be returned:

ASCII: **<STX>**380N**<ETX>**{LRC}

Hex: 02 33 38 30 4E 03 76

<STX>	02	
Response	33 38	(38)
[CS]	30	(0)
[Y/N]	4E	(N)
<ETX>	03	
{LRC}	76	(v)

40 KEYPAD INPUT REQUEST

Command Set: Master/Session Key and DUKPT

Purpose: To get a single key press from the IntelliPIN.

Command Notes:

- This request may be preceded by one of the display requests 42, 43 or Z2. (This request uses the last display loaded with one of these commands.)
- This request can specify how long the IntelliPIN should wait for the key press before timing out.
- If the time-out expires, a question mark (?) will be returned to the PC.
- If [TIME] is omitted, it will be set to 000 (infinite).
- The IntelliPIN will send the ASCII key code to the PC (see table below for the returned values).
- The IntelliPIN does not echo the input on its display.
- Request 72 from the PC can cancel the command.

Request: <STX>40[TIME]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
40	Request Type	2	KeyPad Input Request
[TIME]	Parameter	3 Optional	Time-out in seconds: • 001 - 255 = wait nnn seconds • 000 = wait forever
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Example: Wait 25 seconds for a key press on the IntelliPIN.

ASCII: **<STX>** 40025**<ETX>**0

Hex: 02 34 30 30 32 35 03 30

```
<STX>    02
Request  34 30     (40)
[TIME]   30 32 35  (025)
<ETX>    03
{LRC}    30        (0)
```

Request Errors:

- If the command is bad, no response will be returned, but an error message will be shown (see Table below).
- If a time-out occurs without a key being pressed, then a question mark (?) will be returned in place of the [KEY] parameter.
- If the response message is not ACKed after three retries, then a single EOT (0x04) will be returned.

Response: <STX>40[KEY]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
40	Response Type	2	Key Input Response
[KEY]	Parameter	1	Key Value (see below) or '?' (0x3F) if time-out occurred
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes: Shown below is the table of the values returned for each key pressed.

Key	1	2	3	4	5	6	7	8	9	0	ENTER	CLEAR	F1	F2	F3
ASCII	1	2	3	4	5	6	7	8	9	0	A	B	C	D	E
Hex	31	32	33	34	35	36	37	38	39	30	41	42	43	44	45

Error Displays

Error	Display
Incorrect Timeout Value	"Timeout not 000-255"

Response Example:

If the '5' is pressed, the following will be returned:

ASCII: **<STX>**405**<ETX>**2

Hex: 02 34 30 35 03 32

```
<STX>     02
Response  34 30  (40)
[Key]     35     (5)
<ETX>     03
{LRC}     32     (2)
```

Response Example (Timeout):

This response will be sent if the key is not pressed within the timeout period.

ASCII: **<STX>**40?**<ETX>**8

Hex: 02 34 30 3F 03 38

<STX>	02	
Response	34 30	(40)
[TIMEOUT]	3F	(?)
<ETX>	03	
{LRC}	38	(8)

41 STRING INPUT REQUEST

Command Set: Master/Session Key and DUKPT

Purpose: To obtain one or more digits from the IntelliPIN.

Command Notes:

- This request may be preceded by one of the display requests (42, 43 or Z2). (This request uses the last display loaded with one of these commands.)
- The string of digits will be returned when the ENTER key is pressed. The ENTER key and the three functions keys will not be included in the returned string. This request specifies if the key will be echoed on the IntelliPIN display and how they will be echoed.
- This request specifies the maximum number of digits.
- The IntelliPIN will return to the PC the ASCII equivalent of the keys pressed (0 through 9).
- The IntelliPIN will accept a null entry (i.e., just the ENTER pressed).
- Request 72 from the PC can cancel the operation and return to idle.

Request: `<STX>41[EFLG][TIME][MAXL]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
41	Request Type	2	String Input Request
[EFLG]	Parameter	1	Echo flag, should be one of the following: • 0 = Echo input as asterisks • 1 = Echo input as numbers • 2 = Do not echo inputs • 3 = Echo as US dollar amount ($0.00 format). See MAXL below for limitation of this format.
[TIME]	Parameter	3	(optional) Time-out Value in Seconds: • if omitted, default is 000 • 001-255 = wait nnn seconds • 000 = wait forever.
[MAXL]	Parameter	2	(optional) Maximum acceptable length for input: • 00 - 99 legal range for this value • if omitted, default is 32 • if exceeds 32, will be set to 32 • if 00, will be set to 01 • If EFLG is set to '3', then MAXL will be limited to 12 digits, i.e., MAXL can be less than 12 but never greater.
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Example: Wait up to 25 seconds for up to 6 digits from the IntelliPIN. The digits will be displayed as asterisks (echo flag 0).

```
ASCII:  <STX>41002506<ETX>{0x07}

Hex:  02 34 31 30 30 32 35 30 36 03 07

<STX>    02
Request  34 31     (41)
[EFLG]   30        (0)
[TIME]   30 32 35  (025)
[MAXL]   30 36     (06)
<ETX>    03
{LRC}    07
```

Request Errors:

- If the command is bad, no response will be returned, but an error message will be shown (see Table below).
- If a time-out occurs without a key being pressed, then a question mark **?** (0x3F) will be returned in place of the [LIST] parameter.
- If the response message is not ACKed after three retries, then a single EOT (0x04) will be returned.

Error Displays

Error	Display
Incorrect Timeout Values	"Timeout not 000-255"
Incorrect Echo Flag	"Bad Echo Flag"
Incorrect Maximum Length Values	"Bad Maximum Length"

Response: <STX>41[ENTR]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
41	Response Type	2	String Input Request
[LIST]	Parameter	0 - 32	Character Input as an ASCII String – not present if only ENTER was pressed. ? (0x3F) if time-out.
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Example: (12345<ENTER> pressed)

ASCII: **<STX>**4112345**<ETX>**7

Hex: 02 34 31 31 32 33 34 35 03 37

```
<STX>     02
Response  34 31           (41)
[LIST]    31 32 33 34 35  (12345)
<ETX>     03
{LRC}     37              (7)
```

42 DISPLAY SINGLE STRING MESSAGE

Command Set: Master/Session Key and DUKPT

Purpose: To display a single message.

Command Notes: The IntelliPIN displays the message until the CLEAR key is pressed or it receives another request from the PC that displays a new message.

Request: <STX>42[LINE1]<FS>[LINE2]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
42	Request Type	2	Display a String
[LINE1]	Parameter	0 – 16	Message for line 1
<FS>	Separator	1	Field Separator (0x1C) Needed only if [LINE1] is less than 16 characters long
[LINE2]	Parameter	0 – 16	Message for line 2
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
A value in [LINE1] or [LINE2] was not in the range of 0x20 - 0x7E	Display Data is invalid

Request Example: Display **`Line 1`** on the top line and **`And two`** on line 2 of the IntelliPIN.

ASCII: **<STX>**42Line 1**<FS>**And two**<ETX>**!

Hex: 02 34 32 4C 69 6E 65 20 31 1C 41 6E 64 20 74 77 6F 03 21

```
<STX>    02
Request  34 32                (42)
[LINE1]  4C 69 6E 65 20 31    (Line 1)
<FS>     1C
[LINE2]  41 6E 64 20 74 77 6F  (And two)
<ETX>    03
{LRC}    21                   (!)
```

Response: This request has no response.

43 DISPLAY ALTERNATING MESSAGES

Command Set: Master/Session Key and DUKPT

Purpose: To display two alternating messages on the IntelliPIN.

Command Notes: Two alternate messages can be displayed. The IntelliPIN displays the message strings at two-second intervals until the CLEAR key is pressed or it receives another request that displays a new message.

If only the data for the first message ([M1L1] and/or [M1L2]) is present, then only the first message will be displayed. This is the equivalent of the 42 request.

Request: <STX>43[M1L1]<FS>[M1L2]<FS>[M2L1]<FS>[M2L2]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
43	Request Type	2	Display Alternate Messages
[M1L1]	Parameter	0 - 16	Message 1 line 1
<FS>	Separator	1	Field Separator (0x1C) needed only if [M1L1] is less than 16 characters
[M1L2]	Parameter	0 - 16	Message 1 line 2
<FS>	Separator	1	Field Separator (0x1C) needed only if [M1L2] is less than 16 characters
[M2L1]	Parameter	0 - 16	Message 2 line 1
<FS>	Separator	1	Field Separator (0x1C) needed only if [M2L1] is less than 16 characters
[M2L2]	Parameter	0 - 16	Message 2 line 2
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Notes:

- If any Field Separator is missing, the IntelliPIN will attempt to fill in as much data as will fit on a line and put the rest on subsequent lines.

Error Displays:

Error	Display
A value in [M1L1], [M1L2], [M2L1] or [M2L2] was not in the range of 0x20 - 0x7E	Display Data is invalid

Request Example: Display **One** and **Two** on the top and bottom lines respectively of the first message. Display **Three** and **Four** on the top and bottom lines respectively of the second message.

ASCII: **<STX>**43One**<FS>**Two**<FS>**Three**<FS>**Four**<ETX>**(p)

Hex: 02 34 33 4F 6E 65 1C 54 77 6F 1C 54 68 72 65 65 1C 46 6F 75 72 03 70

<STX>	02	
Request	34 33	(43)
[M1L1]	4F 6E 65	(One)
<FS>	1C	
[M1L2]	54 77 6F	(Two)
<FS>	1C	
[M2L1]	54 68 72 65 65	(Three)
<FS>	1C	
[M2L2]	46 6F 75 72	(Four)
<ETX>	03	
{LRC}	70	(p)

Response: This request has no response. However, if any invalid data is supplied in the message, the display will show “Display Data is invalid” and the command will be ignored.

44 FIRMWARE PART NUMBER AND VERSION REQUEST

Command Set: Master/Session Key and DUKPT

Purpose: To retrieve the part number and version of the IntelliPIN's firmware.

Command Notes: This is used at the factory for automatic configuration and should not be needed in the field, unless some question arises as to the operation of the unit.

Request: <STX>44<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
44	Request Type	2	Firmware Part Number and Version Request
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request the part number from the IntelliPIN.

ASCII: **<STX>**44**<ETX>**{0x03}

Hex: 02 34 34 03 03

<STX> 02
Request 34 34 (44)
<ETX> 03
{LRC} 03

Response: <STX>44[PN][REV][SUB]<ETX>{LRC}

Field	Length	Contents
<STX>	1	Start of Text (0x02)
44	2	Response type
[PN]	8	MAGTEK part number
[REV]	1	Revision Level (A-Z) (0x41 - 0x5A)
[SUB]	2	Two Digits of sub Revision Level
<ETX>	1	End Of Text (0x03)
{LRC}	1	Error Check Character

Response Example:

Example of the returned response:

```
ASCII:  <STX>4430037367J01<ETX>M

Hex:  U02U U34 34U U33 30 30 33 37 33 36 37U U4AU U30 31U U03U U4DU

<STX>     02
Response  34 34           (44)
[PN]      33 30 30 33 37  (30037367)
[Rev]     4A              (J)
[Sub]     30 31           (01)
<ETX>     03
{LRC}     4D              (M)
```

50 SET OR REQUEST SOFT SWITCHES

Command Set: Any

Purpose: To configure the IntelliPIN.

Command Notes:

- If soft switch A (SWA) is changed, the communication parameters will be changed after the IntelliPIN has received an ACK from the PC. This allows the PC to receive the response before changing its communication parameters to match the new IntelliPIN values.
- The MSK Parity Check (SWB bit 2), affects only commands 94, 95 and 96. These commands will check and reject the keys from the request if this bit is a '1' (one) and the key has bad parity. In this case, the key will not be loaded.
- SWD (Power Time Out Value) has a minimum value of 5 (five) minutes. If the command tries to set it to a lower value, it will automatically be set to 5.
- If the switch data is present, then the switch will be set.
- If the switch data is omitted, then the switch will be read and returned.

Two of the bits in switch A affect how the IntelliPIN responds to commands.

MultiUse PIN (Bit 6)

If bit 6 (SWA6) is set to a '1' (one), a PIN will be collected and stored until a cancel command (72) is received or until the unit is shut down. This feature facilitates the computation of multiple results (PVV or offsets) without having to request the customer to enter a PIN more than once. This function is useful for the following commands: 31, 32, 35, 36, 37, and 38. After all offsets and/or PVV have been collected, a cancel command (72) should be sent so that the PIN will be cleared. If SWA6 is '0' (zero), the PIN will be cleared as soon as the offset or PVV has been transmitted.

IC Verify Format (Bit 7)

This bit affects how the 81 response is formatted after a Q40 command has been issued. If bit 7 (SWA7) is '0' (zero), the track format contained in the 81 response is similar to the response to the 80 command. However, if SWA7 is '1' (one), an <STX> is inserted before the start sentinel of each track and a <DLE> separates track 1 and 2 if both are present. See the tables in the 81 Card Data Response section for complete details.

Switch A Values For The RS-232 IntelliPIN

7	6	5	4	3	2	1	0	Bit Position
1	2	3	4	5	6	7	8	Character Position
-	-	-	-	-	0	0	0	Baud Rate : undefined
-	-	-	-	-	0	0	1	:300
-	-	-	-	-	0	1	0	:600
-	-	-	-	-	0	1	1	:1200
-	-	-	-	-	1	0	0	:2400
-	-	-	-	-	1	0	1	:4800
-	-	-	-	-	1	1	0	:9600
-	-	-	-	-	1	1	1	:undefined
-	-	-	0	0	-	-	-	Parity: Space (0), 7 data bits
-	-	-	0	1	-	-	-	: Mark (1), 7 data bits
-	-	-	1	0	-	-	-	: Even, 7 data bits
-	-	-	1	1	-	-	-	: Odd, 7 data bits
-	-	0	-	-	-	-	-	CTS/DSR : ignore
-	-	1	-	-	-	-	-	: use
-	0	-	-	-	-	-	-	MultiUse PIN: Disabled
-	1	-	-	-	-	-	-	: Enabled
0	-	-	-	-	-	-	-	IC Verify Format : Disabled
1	-	-	-	-	-	-	-	: Enabled
0	0	0	1	0	1	1	0	Defaults: • Baud Rate: 9600 bps • Parity: Even • CTS/DSR: ignore • MultiUse PIN: Disabled • IC Verify Format: Disabled

Assembling the Switch A Request RS-232 Example

In this example the following values will be set:

- Ignore CTS/DSR
- Mark Parity/7 bits
- 1200 Baud

50A	The basic Switch A request command
50A**00**	Positions 1 and 2 are undefined currently so they are added in as zeros
50A00**0**	**Ignore CTS/DSR** is defined as a zero at position 3, so that is added in
50A000**01**	**Mark Parity** is defined as a "zero one" at positions 4 and 5, so that is added in
50A00001**011**	**1200 Baud** is defined as a "zero one one" at positions 6, 7 and 8, so that is added in

The Switch A request is now complete and ready to send to the IntelliPIN.

Switch A For The Keyboard Wedge IntelliPIN

7	6	5	4	3	2	1	0	**Bit Position**
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Character Position**
-	-	-	-	-	-	0	0	Speed : 80 characters per second
-	-	-	-	-	-	0	1	: 40 characters per second
-	-	-	-	-	-	1	0	: 30 characters per second
	-	-	-	-	-	1	1	: 20 characters per second
-	-	-	-	0	x	-	-	Scan Code : automatic
-	-	-	-	1	0	-	-	: manual, set 1 (PS2/25, PS2/30)
-	-	-	-	1	1	-	-	: manual, set 2 (AT, PS2/50)
			0					Ack Commands : Enabled
			1					: Disabled
		0						Send EOT on Clear KeyPress: Enabled
		1						Disabled
-	0	-	-	-	-	-	-	MultiUse PIN: Disabled
-	1	-	-	-	-	-	-	: Enabled
0	-	-	-	-	-	-	-	IC Verify Format : Disabled
1	-	-	-	-	-	-	-	: Enabled
0	0	0	0	0	0	0	0	Defaults: • Speed: 80 characters per second • Scan Code: Automatic • Ack Commands: Enabled • Send EOT: Enabled • Multiuse PIN: Disabled • IC Verify Format: Disabled

x = Don't Care

Assembling the Switch A Request Keyboard Wedge Example

In this example, the following values will be set:

- Enable EOT transmission
- Disabled command Acking
- Set Scan Code to Set 1
- Set Speed to 40 CPS

50A	The basic Switch A request command
50A**00**	Positions 1 and 2 are undefined currently so they are added in as zeros
50A00**0**	**Enable EOT** is defined as a zero at position 3, so that is added in
50A000**1**	**Disable Command Acking** is defined as a “one” at position 4, so that is added in
50A0001**10**	**Scan Code Set 1** is defined as a “one zero” at positions 5 and 6, so that is added in
50A00110**01**	**40 CPS** is defined as a “zero one” at positions 7 and 8, so that is added in

The Switch A request is now complete and ready to send to the IntelliPIN.

Switch B Values For RS-232 And Wedge IntelliPIN

7	6	5	4	3	2	1	0	Bit Position
1	2	3	4	5	6	7	8	Character Position
-	-	-	-	-	-	-	0	Magnetic Track 1: Disabled
-	-	-	-	-	-	-	1	: Enabled
-	-	-	-	-	-	0	-	Magnetic Track 2: Disabled
-	-	-	-	-	-	1	-	: Enabled
-	-	-	-	-	0	-	-	Magnetic Track 3: Disabled
-	-	-	-	-	1	-	-	: Enabled
-	-	-	-	0	-	-	-	Double PIN Entry: Disabled
-	-	-	-	1	-	-	-	: Enabled
-	-	-	0	-	-	-	-	Trivial PIN Check : Disabled
-	-	-	1	-	-	-	-	: Enabled
-	-	0	-	-	-	-	-	PIN Block: ANSI 9.8
-	-	1	-	-	-	-	-	: IBM 3624
-	0	-	-	-	-	-	-	Key Parity: Ignore
-	1	-	-	-	-	-	-	: Check
0	-	-	-	-	-	-	-	Reserved for future use
0	1	0	0	0	1	1	1	Defaults: • Track 1: Enabled • Track 2: Enabled • Track 3: Enabled • Double PIN Check: Disabled • Trivial PIN Check: Disabled • PIN Block: ANSI 9.8 • Key Parity: Check

Switch C Values For RS-232 And Wedge IntelliPIN

1	2	3	4	5	6	7	8	Character Position	
				8	4	2	1	Bit Value for the PIN Length below	
	.	.	.	0	0	0	0	16	Notes: • Legal ANSI 9.8 values are 4 to 12 • Legal IBM 3624 values are 1 to 16 (16 is entered as four zeros)
.	.	.	.	0	0	0	1	01	
.	.	.	.	0	0	1	0	02	
.	.	.	.	0	0	1	1	03	
.	.	.	.	0	1	0	0	04	
.	.	.	.	0	1	0	1	05	
.	.	.	.	0	1	1	0	06	
.	.	.	.	0	1	1	1	07	
.	.	.	.	1	0	0	0	08	
.	.	.	.	1	0	0	1	09	
.	.	.	.	1	0	1	0	10	
.	.	.	.	1	0	1	1	11	
.	.	.	.	1	1	0	0	12	
.	.	.	.	1	1	0	1	13	
.	.	.	.	1	1	1	0	14	
.	.	.	.	1	1	1	1	15	
8	4	2	1					Bit Value for the Fill Digit below	
0	0	0	0	.	.	.	.	0	Fill Digit when IBM 3624 PIN block is selected, otherwise not applicable
0	0	0	1	.	.	.	.	1	
0	0	1	0	.	.	.	.	2	
0	0	1	1	.	.	.	.	3	
0	1	0	0	.	.	.	.	4	
0	1	0	1	.	.	.	.	5	
0	1	1	0	.	.	.	.	6	
0	1	1	1	.	.	.	.	7	
1	0	0	0	.	.	.	.	8	
1	0	0	1	.	.	.	.	9	
1	0	1	0	.	.	.	.	A	
1	0	1	1	.	.	.	.	B	
1	1	0	0	.	.	.	.	C	
1	1	0	1	.	.	.	.	D	
1	1	1	0	.	.	.	.	E	
1	1	1	1	.	.	.	.	F	
1	1	1	1	1	1	0	0	Default Values • PIN Length: 12 digits • Fill Digit: ‘F’	

Switch D Values For RS-232 And Wedge IntelliPIN

7	6	5	4	3	2	1	0	Bit Position
1	2	3	4	5	6	7	8	Character Position
(128)	(64)	(32)	(16)	(8)	(4)	(2)	(1)	Power Off Time Delay in Minutes (enter as a binary value)
0	0	1	1	1	1	0	0	Default values: 60 minutes (32 + 16 + 8 + 4 = 60)

Switch E Values For RS-232 And Wedge IntelliPIN

7	6	5	4	3	2	1	0	Data Position
1	2	3	4	5	6	7	8	Character Position
-	-	-	-	-	0	0	0	Mode: Interactive ("Interactive")
-	-	-	-	-	0	0	1	PIN with Card ("PIN w/Card") (Generate Offset)
-	-	-	-	-	0	1	0	PIN with or without Card ("PIN w/oCard")
-	-	-	-	-	0	1	1	Verification of Offset on Card ("Verify Cust")
-	-	-	-	-	1	0	0	Verify Offset on Card or Generate Offset ("PIN&Verify")
-	-	-	-	-	1	0	1	Verification of PIN ("Verify Ofst") (No Offset on Card)
-	-	-	-	0	-	-	-	Return PAN, Offset/Auth Count
-	-	-	-	1	-	-	-	Return ALL Track data, Offset/Auth, Count
-	-	0	0	-	-	-	-	Activated Mode: Card Only
-	-	0	1	-	-	-	-	Card and Password
-	-	1	0	-	-	-	-	Password Only
-	-	1	1	-	-	-	-	Not Defined
-	0	-	-	-	-	-	-	Include Header: No
-	1	-	-	-	-	-	-	Yes
0	-	-	-	-	-	-	-	Allow Viewing of Offset: No
1	-	-	-	-	-	-	-	Yes
1	0	0	0	0	1	0	0	Default Values: • Mode: Verify & Offset • Return: PAN • Activate: Card Only • Header: Don't Include • Offset: Allow Viewing

Switch F Values For RS-232 And Wedge IntelliPIN

1	2	3	4	5	6	7	8	Character Position
(128)	(64)	(32)	(16)	(8)	(4)	(2)	(1)	**Bit value for the "seconds" below**
X	X	X	X	X	X	X	X	Operational Time-out in seconds - (legal values 15 to 255. If a value less than 15 is sent, it will be translated to 15 [00001111])
0	0	0	1	1	1	1	0	Default values: 30 seconds (16 + 8 + 4 + 2) = 30)

Switch G Values For RS-232 And Wedge IntelliPIN

1	2	3	4	5	6	7	8	Character Position
			16	**8**	**4**	**2**	**1**	**Bit value for the "hours" below**
.	.	.	X	X	X	X	X	Time to shut down in hours: 00 to 31 (will be translated to 01 to 32)
.	.	0	.	.	.	.	.	Shut Down Time Out?: No
.	.	1	.	.	.	.	.	Yes
.	0	.	.	.	.	.	.	Allow viewing of authorization code: No
.	1	.	.	.	.	.	.	Yes
0								Allow reswipe on bad card read: Yes
1	.	.	.	.	.	.	.	No (kiosk mode)
0	1	0	0	0	1	1	1	Default values: • 8 hour time out • Don't shut down automatically • Allow viewing of authorization code • Allow reswipe on bad reads

Assembling the Switch G Request, Example

In this example, the following values will be set:

- Authorization codes will not be shown
- Enable Shut Down Time Out
- Shut Down in ten hours
- Don't allow reswipe on bad card read

50G	The basic Switch G request command
50G**1**	Don't Allow Reswipe on Bad Card Read is defined as a "one" at position 1, so that is added in
50G11	**Allow Viewing of Authorization Code** is defined as a "one" at position 2, so that is added in
50G111	**Enable Shut Down Time Out** is defined as a "one" at position 3, so that is added in
50G111**01001**	The **Shut Down Time Out Value** is always stored as the actual number of hours minus one. Therefore, to set the value to ten hours, a nine is sent. This can be thought of how many hour the IntelliPIN will go before timing out. Nine hours has a 5-bit binary value of "zero one zero zero one" (01001) at positions 4 through 8, so that is added in

The Switch G request is now complete and ready to send to the IntelliPIN.

Request: <STX>50[SW][DATA]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
50	Request Type	2	Set or Request Soft Switch
[SW]	Parameter	1	Switch Number: • A (0x41) • B (0x42) • C (0x43) • D (0x44) • E (0x45) • F (0x46) • G (0x47)
[DATA]	Parameter	8	(optional) Switch data, Bit 7...Bit 0. Each is a zero (0x30) or a one (0x31).
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Set switch D to 30 minutes (00011110).

ASCII: **<STX>**50D00011110**<ETX>**B

Hex: 02 35 30 44 30 30 30 31 31 31 31 30 03 42

```
<STX>    02
Request  35 30                    (50)
[SW]     44                       (D)
[DATA]   30 30 30 31 31 31 31 30  (00011110)
<ETX>    03
{LRC}    42                       (B)
```

Response: <STX>50[CS][DATA]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
50	Response Type	2	Set or Request Soft Switch
[CS]	Parameter	1	Confirmation Values: • '0' (0x30) = request accepted and data follows. • '1' (0x31) = Invalid Command (no data follows)
[DATA]	Parameter	8	Switch data, contains zeros (0x30) and ones (0x31). This field is present only if a read was requested by omitting the [DATA] field in the request.
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	(No changes)
The Switch ID is not A, B, C, D, E, F or G, or Switch ID is missing	1 (0x31)	“Switch ID is Incorrect”
One or more of the switch data is not a 0 or a 1 or the switch data is not eight characters long	1 (0x31)	“Bad Switch Data”

Response Example:

ASCII: **<STX>**500**<ETX>**6

Hex: 02 35 30 30 03 36

```
<STX>    02
Request  35 30  (50)
[SW]     30     (0)
<ETX>    03
{LRC}    36     (6)
```

51 REPLACE DEFAULT DISPLAY

Command Set: Any

Purpose: To replace the default displays with new text. See Appendix A, Default Display Messages, for a list of the modifiable messages. The first form of the command can be used to change the common messages numbered 00 through 21. The second form of the command can be used to modify any of the extended messages used in the IntelliPIN. The third form is used to modify the Currency Character to a user-defined shape (e.g., $1.23 to €1.23). See Appendix C for a worksheet on modifying the Currency Character.

Request Type 1: Modify standard messages
Request Type 2: Modify extended messages
Request Type 3: Modify the Currency Character

For Request Type 1 and Type 2, the present setting of the display message can be retrieved. This may be useful for diagnostic purposes or to change specific characters.

Request Type 1: `<STX>51[NUM][LINE1]<FS>[LINE2]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
51	Request Type	2	Replace Default Display With New Display
[NUM]	Parameter	2	Display Number 00 – 21
[LINE1]	Parameter	0 - 16	Message for line 1
<FS>	Separator	1	Field Separator (0x1C) needed only if [LINE1] is less than 16 characters
[LINE2]	Parameter	0 - 16	Message for line 2
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Type 1 (Retrieve Message): `<STX>51[NUM]<ETX>{LRC}`

Request Type 2: <STX>51X[NUM][LINE1]<FS>[LINE2]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
51	Request Type	2	Replace Default Display With New Display
X	Constant	1	'X' (0x58)
[NUM]	Parameter	3	Display Number '000' – '199'*
[LINE1]	Parameter	0 – 20	Message for line 1 (includes formatting characters)
<FS>	Separator	1	Field Separator (0x1C) needed only if [LINE1] is less than 16 characters
[LINE2]	Parameter	0 – 20	Message for line 2 (includes formatting characters)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

*Not all message numbers are supported. See list in Appendix A.

Request Type 2 (Retrieve Message): <STX>51X[NUM]<ETX>{LRC}

Request Type 3: <STX>51C[H0][H1][H2][H3][H4][H5][H6][H7]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
51	Request Type	2	Replace Default Currency Character With New Character
C	Constant	1	'C' (0x43)
[H0] to [H7]	Parameter	16	Character bit data, (Hex, two characters each).
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
The value [NUM] was not in the range of 00 to 21	"Display Number is Incorrect"
A value in [LINE1] or [LINE2] was not in the range of 0x20 - 0x7E	"Display Data is invalid"

Request Example 1: (Change a message) Set the default idle display (display number 00) to **Hello**. In this example, only the top line of the display (line 1) is used, so no Field Separator is needed.

```
ASCII:  <STX>5100Hello<ETX>E

Hex:  02 35 31 30 30 48 65 6c 6c 6f 03 45

<STX>    02
Request  35 31           (51)
[NUM]    30 30           (00)
[LINE1]  48 65 6c 6c 6f  (Hello)
<ETX>    03
{LRC}    45              (E)
```

Response Example 1: This request has no response.

Request Example 2: (Retrieve a message) Retrieve the contents of message number 02.

```
ASCII:  <STX>5102<ETX>{0x05}

Hex:  02 35 31 30 32 03 05

<STX>    02
Request  35 31  (51)
[NUM]    30 32  (02)
<ETX>    03
{LRC}    05
```

Response Example 2:

```
ASCII:  <STX>510PINPad is       processing      <ETX>$

Hex:  02 35 30 30 03 36  69 73 20 20 69 6E 67 20

<STX>    02
Request  35 31                                          (51)
[OK]     30                                             (0)
[LINE1]  50 49 4E 50 61 64 20 69 73 20 20 20 20 20 20 20  (PINPad is       )
[LINE2]  70 72 6F 63 65 73 73 69 6E 67 20 20 20 20 20 20  (processing      )
<ETX>    03
{LRC}    24                                             ($)
```

Request Example 3: (Change the currency character to the Euro. The "1" bits with black dots on the IntelliPIN's display. They are shown highlighted to emphasis that. The hex table is for reference.

Hex Field	Hex Value	1	8	4	2	1
[H0]	06	0	0	1	1	0
[H1]	09	0	1	0	0	1
[H2]	1C	1	1	1	0	0
[H3]	08	0	1	0	0	0
[H4]	1C	1	1	1	0	0
[H5]	09	0	1	0	0	1
[H6]	06	0	0	1	1	0
[H7]	00	0	0	0	0	0

```
ASCII:  <STX>51C06091C081C090600<ETX>L

Hex:  02 35 31 43 30 36 30 39 31 43 30 38 31 43 30 39 30 36 30 30 03 4C

<STX>    02
Request  35 31  (51)
C        43     (C)
[H0]     30 36  (06)
[H1]     30 39  (09)
[H2]     31 43  (1C)
[H3]     30 38  (08)
[H4]     31 43  (1C)
[H5]     30 39  (09)
[H6]     30 36  (06)
[H7]     30 30  (00)
<ETX>    03
{LRC}    4C     (L)
```

Response Example 3: This request has no response.

52 ENABLE DEFAULT DISPLAY

Command Set: Any

Purpose: To restore all the default displays. This request removes any Command 51 requests.

Command Notes: This command takes about one second to complete (after the IntelliPIN returns the ACK response). Delay sending any further command for at least 1500 msec after receiving the ACK response.

Request: <STX>52<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
52	Request Type	2	Enable Default Display
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Re-enable the default display values.

ASCII: **<STX>**52**<ETX>**{0x04}

Hex: 02 35 32 03 04

```
<STX>    02
Request  35 32  (52)
<ETX>    03
{LRC}    04
```

Response: This request has no response.

53 TRANSACTION COUNTER REQUEST

Command Set: Master/Session Key

Purpose: To retrieve the current MSK Transaction Counter from the IntelliPIN.

Command Notes: The transaction counter is returned as four hex digits. The transaction counter will be incremented only after a PIN is successfully collected using one of the Master/ Session Key requests. The counter counts up to FFFF and then rolls over to 0000.

Request: <STX>53<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
53	Request Type	2	Transaction Counter Request
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request the current value of the Master/Session Key Transaction Counter.

```
ASCII:  <STX>53<ETX>{0x05}

Hex:  02 35 33 03 05

<STX>    02
Request  35 33  (53)
<ETX>    03
{LRC}    05
```

Response: <STX>53[TCN]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
53	Response Type	2	Transaction Counter Response
[TCN]	Parameter	4	Transaction Counter (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Displays: None

Response Example:

If the counter has advanced to 3:

ASCII: **<STX>**530003**<ETX>**{0x06}

Hex: 02 35 33 30 30 30 33 03 06

```
<STX>     02
Response  35 33        (53)
[TCN]     30 30 30 33  (0003)
<ETX>     03
{LRC}     06
```

54 TRANSACTION COUNTER RESET

Command Set: Master/Session Key

Purpose: To reset the MSK Transaction Counter.

Command Notes: The Master/Session Key transaction counter will be reset to 0000 (hexadecimal).

Request: <STX>54<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
54	Request Type	2	Transaction Counter Reset
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Reset the Master/Session Key Transaction Counter.

```
ASCII:  <STX>54<ETX>{0x02}

Hex:  02 35 34 03 02

<STX>    02
Request  35 34  (54)
<ETX>    03
{LRC}    02
```

Response: <STX>54[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
54	Response Type	2	Transaction Counter Response
[CS]	Parameter	1	Confirmation Value: • 0 (0x30) = MSK Transaction counter was reset • 1 (0x31) = MSK Transaction counter was not reset
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Example: Transaction Counter was reset to 0000.

```
ASCII:  <STX>540<ETX>2

Hex:  02 35 34 30 03 32

<STX>    02
Request  35 34  (54)
[CS]     30     (0)
<ETX>    03
{LRC}    32     (2)
```

55 KEY SERIAL NUMBER REQUEST

Command Set: Master/Session Key

Purpose: To request the Key Serial Number from the IntelliPIN in clear text.

Command Notes: This request will not be valid if the Key Serial Number has not been loaded.

Request: <STX>55<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
55	Request Type	2	Key Serial Number Request
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request the current Master/Session Key Serial Number.

ASCII: **<STX>**55**<ETX>**{0x03}

Hex: 02 35 35 03 03

<STX> 02
Request 35 35 (55)
<ETX> 03
{LRC} 03

Response: <STX>55[CS][KSN]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
55	Response Type	2	Key Serial Number Response
[CS]	Parameter	1	Confirmation Value as shown below
[KSN]	Parameter	0 or 16	Key Serial Number (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = 0 only if the Key Serial Number is valid.
- [KSN] will only be present if [CS] = 0.

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	(No change)
Serial Number has not been loaded	v (0x76)	“Serial Number not loaded yet”

Response Example:

The following is the response if the serial number 0123456789ABCDEF was loaded (using the 97 command.)

ASCII: **<STX>**5500123456789ABCDEF**<ETX>**5

Hex: 02 35 35 30 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 03 35

<STX>	02	
Response	35 35	(55)
[CS]	30	(0)
[KSN]	30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46	(0123456789ABCDEF)
<ETX>	03	
{LRC}	35	(5)

56 KEY CHECK VALUE REQUEST

Command Set: Master/Session Key

Purpose: To request a Key Check Value from the IntelliPIN.

Command Notes: The Key Check Value (KCV) is calculated by encrypting a 16-digit field which is filled with zeros under the key (specified by [KN]). The Key Check Value (the leftmost six digits of the result) is returned to the PC. This request will not be valid if the selected key has not been loaded.

Request: <STX>56[KN1]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
56	Request Type	2	Key Check Value Request
[KN]	Parameter	1	Key • '0' to '3' (0x30 to 0x33) = lower Working key • '4' (0x34) = Master key • '5' (0x35) = Session key • 'A' to 'Z' (0x41 to 0x5A) = upper Working key • 'a' to 'z' (0x61 to 0x7A) = upper Working key
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Retrieve the Check Value for the Master Key (key number 4).

```
ASCII:  <STX>564<ETX>4

Hex:  02 35 36 34 03 34

<STX>    02
Request  35 36  (56)
[KN]     34     (4)
<ETX>    03
{LRC}    34     (4)
```

Response: <STX>56[CS][KCV]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x20)
56	Response Type	2	Key Check Value Request Response
[CS]	Parameter	1	Confirmation Value as shown below
[KCV]	Parameter	6	Key Check Value (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response Notes:

- [CS] = 0 only if the key is valid.
- [KCV] will only be present if [CS] = 0.

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	No change
Key number is not present or not correct	s (0x73)	"Value should be 0-5, A-Z or a-z"
Requested key has not been loaded	m (0x6D)	"Selected Key not Ready"

Response Example:

With a Maser Key of `23AB 4589 EF67 01CD`, the received KCV is `588161`:

```
ASCII:  <STX>560588161<ETX>3

Hex:  02 35 36 30 35 38 38 31 36 31 03 33

<STX>     02
Response  35 36              (56)
[CS]      30                 (0)
[KCV]     35 38 38 31 36 31  (588161)
<ETX>     03
{LRC}     33                 (3)
```

57 LOAD SUBSTITUTION TABLE

Command Set: Master/Session Key

Purpose: To load the Substitution table to the IntelliPIN.

Command Notes: The Substitution table is used during offset computation to change the encrypted validation data prior to further processing (for requests 31 and 32 only).

Activation: (See Appendix E for flow diagrams)

- DEACTIVATED at power up.
- Activation can be toggled by request 58.

NOTE

If the Offset/Verify commands have not been activated, the following actions need to be taken:
1. Load the Master Key (see the command 94 example).
2. Load the Key Serial Number (see the command 97 example).
3. Activate the Offset/Verify commands (see the command 58 example).

Request: <STX>57[TYPE][DATA]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
57	Request Type	2	Load Substitution Table
[TYPE]	Parameter	1	Table type: • H (0x48) = Hexadecimal table • D (0x44) = Decimal table
[DATA]	Parameter	16	Substitution Table in Decimal or Hexadecimal depending on [Type]
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example (Hexadecimal): Set the substitution table to hexadecimal with a content of 0F1E2D3C4B5A6978.

ASCII: **<STX>**57H0F1E2D3C4B5A6978**<ETX>**O

Hex: 02 35 37 48 30 46 31 45 32 44 33 43 34 42 35 41 36 39 37 38 03 4F

```
<STX>    02
Request  35 37                                              (57)
[TYPE]   48                                                 (H)
[DATA]   30 46 31 45 32 44 33 43 34 42 35 41 36 39 37 38    (0F1E2D3C4B5A6978)
<ETX>    03
{LRC}    4F                                                 (O)
```

Request Example (Decimal): Set the substitution table to decimal with a content of 0123456789012345.

```
ASCII:  <STX>57D0123456789012345<ETX>E

Hex:  02 35 37 44 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 03 45

<STX>    02
Request  35 37                                          (57)
[TYPE]   44                                             (D)
[DATA]   30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35  (0123456789012345)
<ETX>    03
{LRC}    45                                             (E)
```

Response: <STX>57[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
57	Response Type	2	Load Substitution Table Response
[CS]	Parameter	1	Confirmation Value as shown below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	(No change)
Bad table type, not D or H	j (0x6A)	Missing ‘D’ or ‘H’
Request inactive	d (0x64)	Command not Activated
Table data missing or is not 16 digits or incorrect character in data field	p (0x70)	Table data field invalid

Response Example:

```
ASCII:  <STX>570<ETX>1

Hex:  02 35 37 30 03 31

<STX>     03
Response  35 37  (57)
[CS]      30     (0)
<ETX>     03
{LRC}     31     (1)
```

58 ACTIVATE OR DEACTIVATE OFFSET/VERIFY

Command Set: Master/Session Key

Purpose: To activate or deactivate the PIN offset or PVV request, PIN verification request, encryption test request, and load substitution table requests (requests 31-36 and 57).

Command Notes: The IntelliPIN will accept the request only if the Key Serial Number encrypted under the Master Key is correct. For security reasons, requests 31-36 and 57 will be deactivated on power up.

Activation: (See Appendix E for flow diagrams)

Request: <STX>58[KSNE][A/D]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
58	Request Type	2	Activate or Deactivate Offset/Verify
[KSNE]	Parameter	16	Key Serial Num. encrypted under Master Key
[A/D]	Parameter	1	Request Option: • A (0x41) = Activate • D (0x44) = Deactivate
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example (Single-length Key): Deactivate the Offset/Verify commands. The encrypted Key Serial Number is 0611 F087 A5B6 601D (0123 4567 89AB CDEF encrypted under the key of 23AB 4589 EF67 01CD)

ASCII: **<STX>**580611F087A5B6601DA**<ETX>**C

Hex: 02 35 38 30 36 31 31 46 30 38 37 41 35 42 36 36 30 31 44 44 03 46

<STX>	02	
Request	35 38	(58)
[KSNE]	30 36 31 31 46 30 38 37 41 35 42 36 36 30 31 44	(0611F087A5B6601D)
[A/D]	41	(A)
<ETX>	03	
{LRC}	43	(C)

Request Example (Double-Length Key): Activate the Offset/Verify commands using a double-length Master Key. The serial number 0123 4567 89AB CDEF encrypted under the Master Key 23AB 4589 EF67 01CD F48A 40B3 1004 9D75 is 49A6 4338 3FF4 1EB7.

```
ASCII:  <STX>5849A643383FF41EB7A<ETX>{LRC}

Hex:  02 35 38 34 39 41 36 34 33 33 38 33 46 46 34 31 45 42 37 41 02 3F

<STX>    02
Request  35 38                                              (58)
[KSNE]   34 39 41 36 34 33 33 38 33 46 46 34 31 45 42       (49A6 4338 3FF4
         37                                                 1EB7)
[A/D]    41                                                 (A)
<ETX>    03
{LRC}    3F                                                 (?)
```

Response: <STX>58[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
58	Response Type	2	Activate or Deactivate Offset/Verify Response
[CS]	Parameter	1	Confirmation Value as shown below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	(No change)
Master Key not ready	k (0x6b)	Selected Key not ready
Bad encrypted Serial Number	f (0x66)	Key Serial no. field is invalid
Decrypted Serial Number fails comparison	n (0x6E)	Serial Numbers do not match
Activate/Deactivate request not A or D	h (0x68)	Missing ‘A’ or ‘D’
Serial Number has not been loaded	v (0x76)	Serial Number not loaded yet

Response Example:

```
ASCII:  <STX>580<ETX>>

Hex:  02 35 38 30 03 3E

<STX>     02
Response  35 38  (58)
[CS]      30     (0)
<ETX>     03
{LRC}     3E     (>)
```

60 PRE-AUTHORIZATION: PIN ENTRY REQUEST

Command Set: DUKPT

Purpose: To get a PIN from the customer in the form of an encrypted PIN block.

Command Notes:

When the command is received, the IntelliPIN will show the last message received from the 42, 43, or Z2 Commands. It is suggested that a '42' Command be used to load the desired message prior to sending the 60 Command.

The 16-digit encrypted PIN block, along with the Key Serial Number, will be returned to the PC. The PC can then decrypt the PIN block to recover the PIN.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned in place of the response. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message and restarts IntelliPIN entry.

Request 72 from the PC can cancel the operation and return to the idle state.
After a PIN has been entered, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

The format of the PIN block is set by Soft Switch B (see Command 50).

Request: `<STX>60[ACCT]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
60	Request Type	2	Pre-Authorization: PIN Entry Request
[ACCT]	Parameter	0 – 19	Account Number including check digit (decimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request PIN block using account number `4012345678909`.

ASCII: **<STX>**604012345678909**<ETX>**9

Hex: 02 36 30 34 30 31 32 33 34 35 36 37 38 39 30 39 03 39

```
<STX>    02
Request  36 30                                    (60)
[ACCT]   34 30 31 32 33 34 35 36 37 38 39 30 39   (4012345678909)
<ETX>    03
{LRC}    39                                       (9)
```

Response: See Command 71 PIN ENTRY RESPONSE (DUKPT).

62 PRE-AUTHORIZATION: TRANSACTION AMOUNT AUTHORIZATION REQUEST

Command Set: DUKPT

Purpose: To get a yes or no response from the customer in response to a displayed dollar amount on the IntelliPIN.

Command Notes: The IntelliPIN will display the following messages, alternatively, until the customer enters the selection (yes or no).

Message 1 **`Total`**
`$xxxxxxxx.xx` **(Amount of Sale from Host)**

Message 2 **`Please select`**
`Yes  No`

After a function key that corresponds to the 'yes' or 'no' has been pressed, the key will be converted to either '0' for approval or '1' for declination.

Request: <STX>62[C/D][AMT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
62	Request Type	2	Pre-Authorization: Transaction Amount Authorization Request
[C/D]	Parameter	1	Authorization Request Credit / Debit Indicator: • C (0x43) = Credit • D (0x44) = Debit
[AMT]	Parameter	3 – 12	Transaction Amount in cents (decimal) (omit the decimal point and commas)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request a yes/no response from the IntelliPIN for a credit of $123.45.

```
ASCII  <STX>62C12345<ETX>u

Hex:  02 36 32 43 31 32 33 34 35 03 75

<STX>    02
Request  36 32           (62)
[C/D]    43              (C)
[AMT]    31 32 33 34 35  (12345)
<ETX>    03
{LRC}    75              (u)
```

Response: See Command 63 AUTHORIZATION RESPONSE.

63 AUTHORIZATION RESPONSE

Command Set: DUKPT

Purpose: To return a yes (approved) or no (declined) response.

Response Notes: The response is only returned after a valid command as been received and the proper keypad selection has been pushed.

Response: <STX>63[CODE]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
63	Response Type	2	Pre-Authorization: Transaction Amount Authorization Response
[CODE]	Parameter	1	Response Code: 0 (0x30) = Approved 1 (0x31) = Declined
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
Credit / Debit indicator not found	Missing ‘C’ or ‘D’
Amount field has non-decimal characters in it or is less than 3 digits or is more than 12 digits	Amount field is invalid

Response Example: (declined, user pressed “No”)

```
ASCII:  <STX>631<ETX>7

Hex:  02 36 33 31 03 37

<STX>     02
Response  36 33  (63)
[CODE]    31     (1)
<ETX>     03
{LRC}     37     (7)
```

64 PRE-AUTHORIZATION: TRANSACTION AMOUNT AUTHORIZATION/DATA AUTHENTICATION REQUEST

Command Set: DUKPT

Purpose: To obtain a Yes or No response from the customer to a displayed amount and obtain the Message Authentication Codes (MAC).

Command Notes: The IntelliPIN will display the following messages, alternatively, until the customer enters the selection (yes or no).

Message 1 **Total**
$xxxxxxxx.xx (Amount of Sale from Host)

Message 2 **Please select**
yes no

After a function key that corresponds to the "yes" or "no" has been pressed, the key will be converted to either a "0" for approval or "1" for declination.

Request 64 works like request 62 (see request 62 for more information). The difference is the addition of the MAC calculation (it requires the [ACCT], [C/D], and [AMT] fields).

Request: <STX>64[ACCT]<FS>[C/D][AMT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
64	Request Type	2	Pre-Authorization Transaction Amount Authorization / Data Authentication Request
[ACCT]	Parameter	0 – 19	Account Number (decimal)
<FS>	Separator	1	Field Separator (0x1C)
[C/D]	Parameter	1	Credit / Debit Indicator: • C (0x43) = Credit • D (0x44) = Debit
[AMT]	Parameter	3 – 12	Transaction Amount in cents (decimal) (omit the decimal point and commas)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request an accept/decline from the IntelliPIN with an account number of 123456781234 and a credit of $123.45.

ASCII: **<STX>**64123456781234**<FS>**C12345**<ETX>**C

Hex: 02 36 34 31 32 33 34 35 36 37 38 31 32 33 34 1C 43 31 32 33 34 35 03 63

<STX>	02	
Request	36 34	(64)
[ACCT]	31 32 33 34 35 36 37 38 31 32 33 34	(123456781234)
<FS>	1C	
[C/D]	43	(C)
[AMT]	31 32 33 34 35	(12345)
<ETX>	03	
{LRC}	63	(c)

Response: See Command 65 AUTHORIZATION AND MAC RESPONSE

65 AUTHORIZATION AND MAC RESPONSE

Command Set: DUKPT

Purpose: To return a yes (approved) or no (declined) response and to validate the PAN and amount with a set of MAC codes.

Response Notes: The response is only returned after a valid command has been received and the proper keypad selection has been pushed. It also returns the message authentication code values that are used to ensure that the account number and amount values have not been altered.

Response: <STX>65[CODE][MAC1][MAC2][MAC3]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
65	Response Type	2	Pre-Authorization Transaction Amount / Authentication Response
[CODE]	Parameter	1	Response Code: • 0 (0x30) = Approved • 1 (0x31) = Declined
[MAC1]	Parameter	8	*Message Authentication Code #1 (hexadecimal)
[MAC2]	Parameter	8	*Message Authentication Code #2 (hexadecimal)
[MAC3]	Parameter	8	*Message Authentication Code #3 (hexadecimal)
[KSN]	Parameter	10 – 20	Key Serial Number: hexadecimal (leading F's suppressed)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

*See Visa Point-of-Sale Equipment Requirements: PIN Processing and Data Authentication, International, Version 1.0, August 2004.

Error Displays:

Error	Display
No Field Separator found	"Missing Field Separator"
Account number field longer than 19 digits or contains non-decimal digits	"Account Number field is invalid"
Credit / Debit Flag not found	"Missing 'C' or 'D'"
Amount field contains non-decimal digits or is less than 3 digits or is longer than 12 digits	"Amount field is invalid"

Response Example: If this command immediately follows the Pre-authorization: PIN Entry Request as shown in the example for the 60 command and a "yes" is entered, the following will be received: `65 0 0161EBEE 6CDDBF15 3E0BC618 9876543210E00002`:

ASCII: **<STX>**6500161EBEE6CDDBF153E0BC6189876543210E00001**<ETX>**J

Hex: 02 36 35 30 30 31 36 31 45 42 45 45 36 43 44 44 42 46 31 35 33 45 30 42
43 36 31 38 39 38 37 36 35 34 33 32 31 30 45 30 30 30 30 32 03 4A

<STX>	02	
Response	36 35	(65)
[CS]	30	(0)
[MAC1]	30 31 36 31 45 42 45 45	(0161EBEE)
[MAC2]	36 43 44 44 42 46 31 38	(6CDDBF15)
[MAC3]	33 45 30 42 43 36 31 38	(3E0BC618)
[KSN]	39 38 37 36 35 34 33 32 31 30 45 30 30 30 30 31	(9876543210E00001)
<ETX>	03	
{LRC}	4A	(J)

66 PRE-AUTHORIZATION: PIN ENTRY TEST REQUEST

Command Set: DUKPT

Purpose: To obtain a test PIN (always 1234) from the IntelliPIN in the form of an encrypted PIN block.

Comments: The response to this request is response 71. The PIN will be set to **1234**.

Request: <STX>66[ACCT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
66	Request Type	2	PIN Entry Test Request
[ACCT]	Parameter	0 – 19	Account Number (decimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request a test PIN using an account number of 4012345678909.

ASCII: **<STX>**604012345678909**<ETX>**?

Hex: 02 36 36 34 30 31 32 33 34 35 36 37 38 39 30 39 03 3F

```
<STX>    02
Request  36 36                                    (66)
[ACCT]   34 30 31 32 33 34 35 36 37 38 39 30 39   (4012345678909)
<ETX>    03
{LRC}    3F                                       (?)
```

Response: See Command 71 PIN ENTRY RESPONSE (DUKPT).

70 PIN ENTRY REQUEST (DUKPT)

Command Set: DUKPT

Purpose: To get a PIN from the customer after displaying the following messages:

Message 1 **`Total`**
`$xxxxxxxxxx.xx` (Amount of sale from the host)

Message 2 **`Please enter PIN`**
`then press Enter`

Command Notes: The IntelliPIN will display the messages, alternatively, until the customer starts entering the PIN.

The 16-digit encrypted PIN block, along with the Key Serial Number, will be returned to the PC. The PC can then decrypt the PIN block to recover the PIN.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned in place of the response. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message and restarts IntelliPIN entry.

Request 72 from the PC can cancel the operation and return to the idle state.

After a PIN has been entered, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

The format of the PIN block is set by Soft Switch B. See Command 50.

Request: `<STX>70[ACCT]<FS>[C/D][AMT]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
70	Request Type	2	PIN Entry Request
[ACCT]	Parameter	0 – 19	Account Number including check digit (decimal)
<FS>	Separator	1	Field Separator (0x1C)
[C/D]	Parameter	1	Credit / Debit Indicator: • C (0x43) = Credit • D (0x44) = Debit
[AMT]	Parameter	3 – 12	Transaction Amount in cents (decimal) (omit the decimal point and commas)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Get the Key Serial Number and the encrypted PIN block from the IntelliPIN using the account number `4012345678909` and a credit display of `$19.95`.

```
ASCII:  <STX>704012345678909<FS>C1995<ETX>C

Hex:  02 37 30 34 30 31 32 33 34 35 36 37 38 39 30 39 1C 43 31 39 39 35 03 63
```

<STX>	02	
Request	37 30	(70)
[ACCT]	34 30 31 32 33 34 35 36 37 38 39 30 39	(4012345678909)
<FS>	1C	
[C/D]	43	(C)
[AMT]	31 39 39 35	(1995)
<ETX>	03	
{LRC}	63	(c)

Response: See Command 71 PIN ENTRY RESPONSE (DUKPT).

70 PIN ENTRY REQUEST (MMK)

Command Set: Multi-Master Key

Purpose: To get a PIN from the customer after displaying the following messages:

Message 1 **`Total`**
`$xxxxxxxxxx.xx` (Amount of sale from the host)

Message 2 **`Please enter PIN`**
`then press Enter`

Command Notes: After a PIN has been entered, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

The IntelliPIN will only execute Command 72, Cancel Session Request, while it waits for the customer's PIN entry.

Use Command 60 if amount is not needed

Request: `<STX>70.[ACCT]<FS>[KEY][AMT]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
70	Request Type	2	PIN Entry Request
.	Separator	1	Period (0x2E)
[ACCT]	Parameter	8-19	Account Number including check digit (decimal)
<FS>	Separator	1	Field Separator (0x1C)
[KEY]	Parameter	16	Working Key encrypted using current Master Key; if zero filled, current Master Key used as Working Key
[AMT]	Parameter	3 – 8	Amount for display on the PINPad; must not include the decimal point (optional field)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example:

ASCII: **<STX>**70.00001234567812349**<FS>**CCC707472B6AF6CF12345**<ETX>{0x1E}**

Hex: 02 37 30 2E 30 30 30 30 31 32 33 34 35 36 37 38 31 32 33 34 39 1C 43 43
43 37 30 37 34 37 32 42 36 41 46 36 43 46 31 32 33 2E 34 35 03 1E

<STX>	02	
Request	37 30	(70)
Cmd	2E	(.)
Delim		
[ACCT]	30 30 30 30 31 32 33 34 35 36 37 38 31 32 33 34 39	(00001234567812349)
<FS>	1C	
[KEY]	43 43 43 37 30 37 34 37 32 42 36 41 46 36 43 45	(CCC707472B6AF6CF)
[AMT]	31 32 33 2E 34 35	(12345)
<ETX>	03	
{LRC}	1E	

Response: See Command 71 PIN ENTRY RESPONSE (MMK).

71 PIN ENTRY RESPONSE (DUKPT)

Command Set: DUKPT

Purpose: To return a PIN from the customer in the form of an encrypted PIN block.

Response Notes: The format of this response is common to all DUKPT PIN request commands. If an error is detected, no response is returned, but an error message will be shown. The error displays shown below include all possible errors. They may not apply to every command.

Response: <STX>71[CS][KSN][EPIN]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
71	Response Type	2	PIN Entry Response
[CS]	Parameter	1	Always 0
[KSN]	Parameter	10 – 20	Key Serial Number (hexadecimal) (leading F's suppressed)
[EPIN]	Parameter	16	Encrypted PIN Block
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
All future keys have been used	"No More Key"
Initial key not loaded yet	"Initial Key has not been loaded"
Account Number contains non-decimal digits or is more than 19 digits	"Account Number field is invalid"
No Field Separator found	"Missing Field Separator"
Credit / Debit Flag not found	"Missing 'C' or 'D'"
Amount field contains non-decimal digits or is less than 3 digits or is more than 12 digits	"Amount field is invalid"
No period found	"Missing '.'"

Response Example: If this command immediately follows the Load Initial Key Request as shown in the example for the 90 command and a PIN of 1234 is entered, the following will be received: 71 0 9876543210E00001 A0BF F43E 87FA 1B4B:

ASCII: **<STX>**7109876543210E00001A0BFF43E87FA1B4B**<ETX>**<

Hex: 02 37 31 30 39 38 37 36 35 34 33 32 31 30 45 30 30 30 30 31 41 30 42 46
46 34 33 45 38 37 46 41 31 42 34 42 03 3C

<STX>	02	
Response	37 31	(71)
[CS]	30	(0)
[KSN]	39 38 37 36 35 34 33 32 31 30 45 30 30 30 30 31	(9876543210E00001)
[EPIN]	41 30 42 46 46 34 33 45 38 37 46 41 31 42 34 42	(A0BFF43E87FA1B4B)
<ETX>	03	
{LRC}	3C	(<)

71 PIN ENTRY RESPONSE (MMK)

Command Set: Multi-Master Key

Purpose: To return encrypted PIN block to MMK Commands.

Response Notes: This format of the response is common to all Multi-Master Key Commands. It returns the encrypted PIN block. No Response will be returned if the command is improperly formatted, but the display will show an error message.

Response: <STX>71.0[LEN][01][EPIN]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
71	Response Type	2	PIN Entry Response
.	Separator	1	Period
0	Constant	1	Always 0
[LEN]	Parameter	2	PIN Length (04-12)
[01]	Constant	2	Always 01
[EPIN]	Parameter	16	Encrypted PIN Block
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
Account number contains non-decimal digits or is invalid length	"Account Number field is invalid"
No Field Separator found	"Missing Field Separator"
Amount field contains non-decimal digits, or no decimal point, or is less than 3 or more than 8 digits.	"Amount field is invalid"
Invalid Format of Key Field	"Bad Key Data"
MMK not loaded	"Selected Key not Ready"

Response Example: Assumes MMK #3 loaded and selected as shown in the commands 02 and 08 examples. PIN entered was "1234".

```
ASCII:  <STX>71.004018EA4ECE85B6EB8E6<ETX>'

Hex:  02 37 31 2E 30 30 34 30 31 44 35 44 36 44 46 38 44 30 44 42 38 39 37 41
42 03 60

<STX>      02
Response   37 31                                              (71)
Delimiter  2E                                                 (.)
Constant   30                                                 (0)
[LEN]      30 34                                              (04)
Constant   30 31                                              (01)
[EPIN]     38 45 41 34 45 43 45 38 35 42 36 45 42 38 46 36    (8EA4ECE85B6EB8E6)
<ETX>      03
{LRC}      60                                                 (')
```

72 CANCEL SESSION REQUEST

Command Set: Master/Session Key and DUKPT

Purpose: To return the PINPad to its idle state.

Command Notes: This request is used to cancel/abort the following commands:
PIN Entry: 30, 31, 32, 60, 70, 74, Z60
Data Entry: 40, 41, 62, 64, 80

This request does not display **`Cancel requested`**.

Request: <STX>72<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
72	Request Type	2	Cancel Session Request
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Cancel the current session.

ASCII: **<STX>**72**<ETX>{0x06}**

Hex: 02 37 32 03 06

<STX> 02
Request 37 32 (72)
<ETX> 03
{LRC} 06

Response: This request has no response, but the display will revert to the idle message (typically “Welcome” in Interactive Mode).

74 PIN ENTRY/DATA AUTHENTICATION REQUEST

Command Set: DUKPT

Purpose: To obtain a PIN from the customer in the form of an encrypted PIN block and obtain the Message Authentication Codes (MAC).

Command Notes: The IntelliPIN will display the following messages, alternatively, until the customer starts entering the PIN.

Message 1 **`Total`**
`$xxxxxxxxxx.xx` (Amount of sale from host)

Message 2 **`Please enter PIN`**
`then press Enter`

The 16-digit encrypted PIN block, along with the Key Serial Number and the message authentication codes, will be returned to the PC. The PC can then decrypt the PIN block to recover the PIN.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned in place of the response. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message and restarts IntelliPIN entry.

Request 72 from the PC can cancel the operation and return to the idle state.

After a PIN has been entered, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

The format of the PIN block is set by Soft Switch B, see Command 50.

Request: <STX>74[ACCT]<FS>[C/D][AMT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
74	Request Type	2	PIN Entry/Data Authentication Request
[ACCT]	Parameter	0 - 19	Account Number including check digit (decimal)
<FS>	Separator	1	Field Separator (0x1C)
[C/D]	Parameter	1	Credit / Debit Indicator: • C (0x43) = Credit • D (0x44) = Debit
[AMT]	Parameter	3 - 12	Transaction Amount in cents (decimal) (omit the decimal point and commas)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Get an encrypted PIN block from the IntelliPIN with the account number if 4012345678909 and a credit of $19.95. PIN = 1234.

```
ASCII:  <STX>744012345678909<FS>C1995<ETX>g

Hex:  02 37 34 34 30 31 32 33 34 35 36 37 38 39 30 39 1C 43 31 39 39 35 03 67

<STX>    02
Request  37 34                                     (74)
[ACCT]   34 30 31 32 33 34 35 36 37 38 39 30 39    (4012345678909)
<FS>     1C
[C/D]    43                                        (C)
[AMT]    31 39 39 35                               (1995)
<ETX>    03
{LRC}    67                                        (g)
```

Response: See Command 75 PIN ENTRY AND MAC RESPONSE.

75 PIN ENTRY AND MAC RESPONSE

Command Set: DUKPT

Purpose: To return a PIN from the customer in the form of an encrypted PIN block and to validate the PAN and amount with a set of MAC codes.

Response Notes: The response is only returned after a valid command has been received and a PIN has been entered.

<STX>75[MAC1][MAC2][MAC3]0[KSN][EPIN]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
75	Response Type	2	PIN Entry/Data Authentication Response
[MAC1]	Parameter	8	*Message Authentication Code #1 (hexadecimal)
[MAC2]	Parameter	8	*Message Authentication Code #2 (hexadecimal)
[MAC3]	Parameter	8	*Message Authentication Code #3 (hexadecimal)
0	Filler	1	Value must be zero (0x30)
[KSN]	Parameter	10 - 20	Key Serial Number (hexadecimal) (leading F's suppressed)
[EPIN]	Parameter	16	Encrypted PIN Block (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

* See Visa Point-of-Sale Equipment Requirements: PIN Processing and Data Authentication, International, Version 1.0, August 2004.

Error Displays:

Error	Display
All future keys have been used	No More Key
Initial Key has not been loaded	Initial Key has not been loaded
No Field Separator found	Missing Field Separator
Account Number contains non-decimal digits or is more than 19 digits	Account Number field is invalid
Credit / Debit Flag not found	Missing 'C' or 'D'
Amount field contains non-decimal digits or is less than 3 digits or is more than 12 digits	Amount field is invalid

Response Example: If this command immediately follows the PIN Entry Request as shown in the example for the 70 (DUKPT) command and a PIN of 1234 is entered, the following will be received: 75 6DF3F51C C82D75FA 57F8CF03 0 9876543210E00002 E442 CECE 9E46 D31F:

ASCII: **<STX>**756DF3F51CC82D75FA57F8CF0309876543210E00001E442CECE9E46D31F**<ETX>**:

Hex: 02 37 35 36 44 46 33 46 35 31 43 43 38 32 44 37 35 46 41 35 37 46 38 43
46 30 33 30 39 38 37 36 35 34 33 32 31 30 45 30 30 30 30 31 45 34 34 32
43 45 43 45 39 45 34 36 44 33 31 46 03 39

<STX>	02	
Response	37 35	(75)
[MAC1]	36 44 46 33 46 35 31 43	(6DF3F51C)
[MAC2]	43 38 32 44 37 35 46 41	(C82D75FA)
[MAC3]	35 37 46 38 43 46 30 33	(57F8CF03)
Fill	30	(0)
[KSN]	39 38 37 36 35 34 33 32 31 30 45 30 30 30 30 31	(9876543210E00001)
[EPIN]	45 34 34 32 43 45 43 45 39 45 34 36 44 33 31 46	(E442CECE9E46D31F)
<ETX>	03	
{LRC}	39	(:)

76 PIN ENTRY TEST REQUEST

Command Set: DUKPT

Purpose: To simulate a customer's PIN entry with a fixed PIN of 1234.

Command Notes: This request acts the same as request 70 (PIN Entry Request) except the PIN entry is made automatically and is set to the Value of **1234**. This is useful in writing test programs for the IntelliPIN.

Request: <STX>76[ACCT]<FS>[C/D][AMT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
76	Request Type	2	PIN Entry Test Request
[ACCT]	Parameter	0 - 19	Account Number (decimal)
<FS>	Separator	1	Field Separator (0x1C)
[C/D]	Parameter	1	Credit / Debit Indicator: • C (0x43) = Credit • D (0x44) = Debit
[AMT]	Parameter	3 – 12	Transaction Amount in cents (decimal) (omit the decimal point and commas)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Get the test PIN block with an account number 1234567812345 and a debit of $19.95.

ASCII: **<STX>**761234567812345**<FS>**D1995**<ETX>**g

Hex: 02 37 36 31 32 33 34 35 36 37 38 31 32 33 34 35 1c 44 31 39 39 35 03 67

```
<STX>    02
Request  37 36                                    (76)
[ACCT]   31 32 33 34 35 36 37 38 31 32 33 34 35   (1234567812345)
<FS>     1C
[C/D]    44                                       (D)
[AMT]    31 39 39 35                              (1995)
<ETX>    03
{LRC}    67                                       (g)
```

Response: See Command 71 PIN ENTRY RESPONSE (DUKPT).

78 PIN ENTRY TEST/DATA AUTHENTICATION REQUEST

Command Set: DUKPT

Purpose: To request a PIN Entry test with authentication.

Command Notes: This request functions the same as request 74 except the display does not change and the PIN is automatically entered as **1234**.

Request: <STX>78[ACCT]<FS>[C/D][AMT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
78	Request Type	2	PIN Entry Test / Data Authentication Request
[ACCT]	Parameter	0 - 19	Account Number (decimal)
<FS>	Separator	1	Field Separator (0x1C)
[C/D]	Parameter	1	Credit / Debit Indicator: • C (0x43) = Credit • D (0x44) = Debit
[AMT]	Parameter	3 - 12	Transaction Amount (decimal) (omit decimal point and commas)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Get test PIN block from the IntelliPIN.

```
ASCII:  <STX>781234567812345<FS>D1995<ETX>i

Hex:  02 37 38 31 32 33 34 35 36 37 38 31 32 33 34 35 1c 44 31 39 39 35 03 69

<STX>    02
Request  37 38                                     (78)
[ACCT]   31 32 33 34 35 36 37 38 31 32 33 34 35    (1234567812345)
<FS>     1C
[C/D]    44                                        (D)
[AMT]    31 39 39 35                               (1995)
<ETX>    03
{LRC}    69                                        (i)
```

Response: See Command 75 PIN ENTRY AND MAC RESPONSE.

80 CARD DATA ENTRY REQUEST

Command Set: Master /Session Key and DUKPT

Purpose: To read a magnetic stripe card.

Command Notes: This request optionally displays a message and enables the card reader on the IntelliPIN. After reading the card, the data will be formatted and returned to the PC.

Request 72 from the PC will cancel the operation and return to the idle state. Request 82 from the PC will cancel the operation and display a message. If the customer presses the CLEAR key, an EOT (0x04) character will be returned in place of the response. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message.

After a card is read, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

If the Field Separator (<FS>) is not present, the IntelliPIN will place the first 16 characters it finds on line one and any remaining characters on line two.

The IntelliPIN will put blanks in any unfilled display areas. This means, for instance, if you have just line 1 data, then line 2 will be blank. If you wish line 1 to be blank, place the Field Separator immediately after the "80" in the request.

It is okay to omit [LINE1] and/or [LINE2], but the display will be blank.

If there is too much data for a line, it will be truncated at 16 characters and the rest will be discarded.

Request: `<STX>80[LINE1]<FS>[LINE2]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
80	Request Type	2	Card Holder Data Entry Request
[LINE1]	Parameter	0 - 16	(optional) Message for Line 1
<FS>	Separator	1	Field Separator (0x1C) needed only if [LINE1] is less than 16 characters
[LINE2]	Parameter	0 - 16	(optional) Message for Line 2
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request a card read with a display of **`Please swipe the card...`**.

```
ASCII:  <STX>80Please swipe<FS>the card...<ETX>{0x02}

Hex: 02 38 30 50 6C 65 61 73 65 20 73 77 69 70 65 1C 74 68 65 20 63 61 72 64 2E
     2E 2E 03 02
```

<STX>	02	
Request	38 30	(80)
[LINE1]	50 6C 65 61 73 65 20 73 77 69 70 65	(Please swipe)
<FS>	1C	
[LINE2]	74 68 65 20 63 61 72 64 2E 2E 2E	(the card...)
<ETX>	03	
{LRC}	02	

Response: (After card swiped) See Command 81 CARD DATA RESPONSE.

Possible Request Errors	[STS]	Display
[LINE1] or [LINE2] not 0x20 - 0x7E	w (0x77)	“Display Data is invalid”
[MnLn] not 0x20 - 0x7E (83 command only)	w (0x77)	“Display Data is invalid”
[REQ] not 0 or 1	q (0x71)	“User Data field is invalid”
[AMT] not decimal or not 3-12 digits	b (0x62)	“Amount field is invalid”

81 CARD DATA RESPONSE

Command Set: Master/Session Key and DUKPT

Purpose: To return data from a magnetic stripe card.

Response Notes: This response can come directly from an 80 or 83 command or at any time in the Interactive mode if the card reader is set to "On" (see the Q4 Request).

This is a three track card reader. Each track can be enabled or disabled (see the 50 command, Switch B). The reader can read ISO standard card data format, California Driver Licenses (CDL) and AAMVA driver licenses. The returned card data is in the following format:

ISO	%	TK1 data	?	;	TK2 data	?	+	TK3 data	?
AAMVA	%	TK1 data	?	;	TK2 data	?	#	TK3 data	?
CDL	%	TK1 data	?	;	TK2 data	?	!	TK3 data	?

where:

Track 1 Start Sentinel	%	0x25
Track 1 data, 0 to 76 alphanumeric chars and field separator	A-Z, 0-9, ^	0x41-5A, 0x30-39, and 0x5E
Track 1 End Sentinel	?	0x3F

Track 2 Start Sentinel	;	0x3B
Track 2 data, 0 to 37 numeric digits and field separator	0-9, =	0x30-39 and 0x3D
Track 2 End Sentinel	?	0x3F

Track 3 Start Sentinel	+, #, !	0x2B, 0x23, 0x21
Track 3 data, 0 to 104 numeric digits and field separator	0-9, =	0x30-39 and 0x3D
Track 3 End Sentinel	?	0x3F

If a track is disabled, no information will be returned for that track (no start sentinel, data or end sentinel).

If there is a read error on an enabled track, a capital 'E' (0x45) will be placed in the data field.

The format of the returned data depends on the current settings of the IntelliPIN.

For the following descriptions, assume the card data to be:
Track 1: `%ABCD?`
Track 2: `;1234?`
Track 3: `+5678?`

Error and Status Display

Error	Status	Display
Error on at least one track	None	"Bad reading Swipe again?" "Please Select Yes No"

Commands 80/83. Tracks 1, 2 and 3 enabled.

Trks read	Trks returned	Example
1	1	<STX>81<space>%ABCD?<ETX>{34}
2	2	<STX>81<space>;1234?<ETX>{2a}
3	3	<STX>81<space>+5678?<ETX>{32}
1,2	1,2	<STX>81<space>%ABCD?;1234?<ETX>{34}
1,3	1,3	<STX>81<space>%ABCD?+5678?<ETX>{2c}
2,3	2,3	<STX>81<space>;1234?+5678?<ETX>{32}
1,2,3	1,2,3	<STX>81<space>%ABCD?;1234?+5678?<ETX>{2c}

Commands 80/83. Track 1 enabled.

Trks read	Trks returned	Example
1	1	<STX>81<space>%ABCD?<ETX>{34}
2	(no response)	
3	(no response)	
1,2	1	<STX>81<space>%ABCD?<ETX>{34}
1,3	1	<STX>81<space>%ABCD?<ETX>{34}
2,3	(no response)	
1,2,3	1	<STX>81<space>%ABCD?<ETX>{34}

Commands 80/83. Track 2 enabled.

Trks read	Trks returned	Example
1	(no response)	
2	2	<STX>81<space>;1234?<ETX>{2a}
3	(no response)	
1,2	2	<STX>81<space>;1234?<ETX>{2a}
1,3	(no response)	
2,3	2	<STX>81<space>;1234?<ETX>{2a}
1,2,3	2	<STX>81<space>;1234?<ETX>{2a}

Commands 80/83. Track 3 enabled.

Trks read	Trks returned	Example
1	(no response)	
2	(no response)	
3	3	<STX>81<space>+5678?<ETX>{32}
1,2	(no response)	
1,3	3	<STX>81<space>+5678?<ETX>{32}
2,3	3	<STX>81<space>+5678?<ETX>{32}
1,2,3	3	<STX>81<space>+5678?<ETX>{32}

Commands 80/83. Tracks 1 and 2 enabled.

Trks read	Trks returned	Example
1	1	<STX>81<space>%ABCD?<ETX>{34}
2	2	<STX>81<space>;1234?<ETX>{2a}
3	(no response)	
1,2	1,2	<STX>81<space>%ABCD?;1234?<ETX>{34}
1,3	1	<STX>81<space>%ABCD?<ETX>{34}
2,3	2	<STX>81<space>;1234?<ETX>{2a}
1,2,3	1,2	<STX>81<space>%ABCD?;1234?<ETX>{34}

Commands 80/83. Tracks 1 and 3 enabled.

Trks read	Trks returned	Example
1	1	<STX>81<space>%ABCD?<ETX>{34}
2	(no response)	
3	3	<STX>81<space>+5678?<ETX>{32}
1,2	1	<STX>81<space>%ABCD?<ETX>{34}
1,3	1,3	<STX>81<space>%ABCD?+5678?<ETX>{2c}
2,3	3	<STX>81<space>+5678?<ETX>{32}
1,2,3	1,3	<STX>81<space>%ABCD?+5678?<ETX>{2c}

Commands 80/83. Tracks 2 and 3 enabled.

Trks read	Trks returned	Example
1	(no response)	
2	2	<STX>81<space>;1234?<ETX>{2a}
3	3	<STX>81<space>+5678?<ETX>{32}
1,2	2	<STX>81<space>;1234?<ETX>{2a}
1,3	3	<STX>81<space>+5678?<ETX>{32}
2,3	2,3	<STX>81<space>;1234?+5678?<ETX>{32}
1,2,3	2,3	<STX>81<space>;1234?+5678?<ETX>{32}

Interactive Mode with Q4 set to On. Tracks 1, 2 and 3 enabled.

Trks read	Trks returned	SWA7	Example
1 or 1 & 3	1	1	<STX>81.<STX>%ABCD?<ETX>{38}
2 or 2 & 3	2	1	<STX>81.<STX>;1234?<ETX>{26}
3	none	1	<STX>81.<ETX>{24}
1,2 or 1, 2, 3	1,2	1	<STX>81.<STX>%ABCD?<DLE><STX>;1234?<ETX>{2a}
1	1	0	<STX>81.%ABCD?<ETX>{3a}
2	2	0	<STX>81.;1234?<ETX>{24}
1,2	1,2	0	<STX>81.%ABCD?;1234?<ETX>{3a}
3	3	0	<STX>81.+5678?<ETX>{3c}

Interactive Mode with Q4 set to On. Track 1 enabled.

Trks read	Trks returned	SWA7	Example
1	1	1	<STX>81.ABCD<ETX>{20}
2 or 3	(no response)	X	
1,2	1	1	<STX>81.ABCD<ETX>{20}
1,3	1	1	<STX>81.ABCD<ETX>{20}
1,2,3	1	1	<STX>81.ABCD<ETX>{20}
1	1	0	<STX>81.%ABCD?<ETX>{3a}

Interactive Mode with Q4 set to On. Track 2 enabled.

Trks read	Trks returned	SWA7	Example
1 or 3	(no response)	1	
2	2	1	<STX>81.1234<ETX>{20}
1,2	2	1	<STX>81.1234<ETX>{20}
2,3	2	1	<STX>81.1234<ETX>{20}
1,2,3	2	1	<STX>81.1234<ETX>{20}
2	2	0	<STX>81.;1234<ETX>{1b}

Interactive Mode with Q4 set to On. Track 3 enabled.

Trks read	Trks returned	SWA7	Example
1 or 2	(no response)	X	
3	none	1	<STX>81.<ETX>{24}
1,3	none	1	<STX>81.<ETX>{24}
2,3	none	1	<STX>81.<ETX>{24}
1,2,3	none	1	<STX>81.<ETX>{24}
3	3	0	<STX>81. +5678?<ETX>{3c}

Interactive Mode with Q4 set to On. Tracks 1 and 2 enabled.

Trks read	Trks returned	SWA7	Example
1 or 1 & 3	1	1	<STX>81.<STX>%ABCD?<ETX>{38}
2 or 2 & 3	2	1	<STX>81.<STX>;1234?<ETX>{26}
3	(no response)	X	
1,2 or 1,2,3	1,2	1	<STX>81.<STX>%ABCD?<DLE><STX>;1234?<ETX>{2a}
1	1	0	<STX>81.%ABCD?<ETX>{3a}
2	2	0	<STX>81.;1234?<ETX>{24}
1,2	1,2	0	<STX>81.%ABCD?;1234?<ETX>{3a}

Interactive Mode with Q4 set to On. Tracks 1 and 3 enabled.

Trks read	Trks returned	SWA7	Example
1	1	1	<STX>81.<STX>%ABCD?<ETX>{38}
2	(no response)	X	
3 or 2 &3	none	1	<STX>81.<ETX>{24}
1,2	1	1	<STX>81.<STX>%ABCD?<ETX>{38}
1,3	1	1	<STX>81.<STX>%ABCD?<ETX>{38}
1,2,3	1	1	<STX>81.<STX>%ABCD?<ETX>{38}
1	1	0	<STX>81.%ABCD?<ETX>{3a}
3	3	0	<STX>81.+5678?<ETX>{3c}

Interactive Mode with Q4 set to On. Tracks 2 and 3 enabled.

Trks read	Trks returned	SWA7	Example
1	(no response)	X	
2	2	1	<STX>81.<STX>;1234?<ETX>{26}
3	none	1	<STX>81.<ETX>{24}
1,2	2	1	<STX>81.<STX>;1234?<ETX>{26}
1,3	none	1	<STX>81.<ETX>{24}
2,3	2	1	<STX>81.<STX>;1234?<ETX>{26}
1,2,3	2	1	<STX>81.<STX>;1234?<ETX>{26}
2	2	0	<STX>81.;1234?<ETX>{24}
3	3	0	<STX>81.+5678?<ETX>{3c}

Response: <STX>81[STS][DAT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
81	Response Type	2	Card Data Entry Response
[STS]	Parameter	1	Read Status Indicator: • B (0x42) = All 3 tracks disabled by software • D (0x44) = Bad parity or LRC • [space] (0x20) = successful read • . [period] (0x2E) = successful read after Q40 Command*
[DAT]	Parameter	--	see Track Data above
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

* When the reader is enabled with the Q40 Command, the response will not include the start sentinel and the end sentinel if only a single track is enabled. (See Example #2 below.)

Response Notes: This response can come directly from an 80 or 83 command or at any time in the Interactive mode (see Command 50, switch E) if the card reader is set to "On" (see the Q4 Request).

The format of the data varies depending on the current mode of the reader and the enabled tracks as follows. To avoid permutations, the actual track numbers have been replaced with the letters a, b and c.

IntelliPIN Operational Mode	Q4 mode	Enabled Tracks	Tracks read	Data Format
all modes 80/83	Off	all	a	<STX> 81 [STS] [TKaSS] [TKaDAT] [ES] <ETX> {LRC}
all modes 80/83	Off	all	a,b	<STX> 81 [STS] [TKaSS] [TKaDAT] [ES] [TKbSS] [TKbDAT] [ES] <ETX> {LRC}
all modes 80/83	Off	all	a,b,c	<STX> 81 [STS] [TKaSS] [TKaDAT] [ES] [TKbSS] [TKbDAT] [ES] [TKcSS] [TKcDAT] [ES] <ETX> {LRC}
Interactive no command sent	Off	doesn't matter	doesn't matter	nothing returned
Interactive no command sent	On	all	a	<STX> 81 [STS] <STX> [TKaSS] [TKaDAT] [ES] <ETX> {LRC}
Interactive no command sent	On	all	a,b	<STX>81[STS] <STX> [TKaSS] [TKaDAT] [ES] <DLE><STX> [TKbSS] [TKbDAT] [ES] <ETX>{LRC}

Response Example # 1: The example shows the results from a typical card read operation as requested by an 81 or 83 Command:

```
ASCII:  <STX>81{0x20}%B1234567890123456^TEST
        CARD^991210100000?;1234567890123456=991210100000? <ETX>i

Hex:  02 38 31 25 42 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 5E 54 45
      53 54 20 43 41 31 52 44 5E 39 39 31 32 31 30 31 30 30 30 30 30 3F 3B 31
      32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 3D 39 39 31 32 31 30 31 30
      30 30 30 30 3F 03 69

<STX>     02
Response  38 31                                 (81)
[STS]     20                                    (Space character)
[TK1]     25 42 31 32 33 34 35 36 37 38         (%B1234567890123456^TEST
          39 30 31 32 33 34 35 36 5E 54         CARD^991210100000?)
          45 53 54 20 43 41 52 44 5E 39
          39 31 32 31 30 31 30 30 30 30
          30 3F
[TK2]     3B 31 32 33 34 35 36 37 38 39         (;1234567890123456=991210100000?)
          30 31 32 33 34 35 36 3D 39 39
          31 32 31 30 31 30 30 30 30 30
          3F
<ETX>     03
{LRC}     69                                    (i)
```

Response Example #2: The example shows the results from a typical card read operation after a Q40 command has enabled a read operation:

```
81 . 1234567890123456=991210100000

ASCII:  <STX>81.1234567890123456=991210100000<ETX>{0x1C}

Hex:  02 38 31 2E 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 3D 39 39 31
      32 31 30 31 30 30 30 30 30 03 1C

<STX>     02
Response  38 31                                 (81)
[STS]     2E                                    (.)
[TK2]     31 32 33 34 35 36 37 38 39 30 31 32   (1234567890123456=991210100000)
          33 34 35 36 3D 39 39 31 32 31 30 31
          30 30 30 30 30
<ETX>     03
{LRC}     1C
```

82 CANCEL AND DISPLAY

Command Set: Master /Session Key and DUKPT

Purpose: To display a message and cancel the current request.

Request: <STX>82[LINE1]<FS>[LINE2]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
82	Request Type	2	Cancel and Display
[LINE1]	Parameter	0 - 16	(optional) Message for Line 1
<FS>	Separator	1	Field Separator (0x1C) needed only if [LINE1] is less than 16 characters long
[LINE2]	Parameter	0 - 16	(Optional) Message for Line 2
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
A Value in [LINE1] or [LINE2] was not in the range of 0x20 to 0x7E	Display Data is invalid

Request Example: Display **Request Cancelled** and cancel the current request.

```
ASCII:  <STX>82Request<FS>Cancelled<ETX>{0x0F}

Hex: 02 38 32 52 65 71 75 65 73 74 1C 43 61 6E 63 65 6C 6C 65 64 03 0F

<STX>    02
Request  38 32                     (82)
[LINE1]  52 65 71 75 65 73 74      (Request)
<FS>     1C
[LINE2]  43 61 6E 63 65 6C 6C 65 64  (Cancelled)
<ETX>    03
{LRC}    0F
```

Response: This request has no response.

83 CARD HOLDER DATA AND PIN ENTRY REQUEST

Command Set: Master/Session Key and DUKPT

Purpose: To read a magnetic stripe card and collect a PIN. The PIN is stored until a PIN request is issued by the host.

Command Notes: This request is used with the portable IntelliPIN to allow the host to collect magnetic stripe data and a PIN while the IntelliPIN is removed from the base. Without this command, the following host sequence would have to be performed:

1) The host sends a collect magnetic stripe data command (request 80) to the IntelliPIN while the IntelliPIN is in the dock.
2) The customer lifts the IntelliPIN from its dock, swipes the card and returns the IntelliPIN to the dock.
3) The host sends a PIN entry command (request 32 for example).
4) The customer lifts the IntelliPIN from its dock, enters the PIN and returns the IntelliPIN to the dock.
5) The host collects the PIN data.

Using the 83 command, the following events happen:

1) The host sends a "collect magnetic stripe data and store PIN command" (request 83) to the IntelliPIN while the IntelliPIN is in the dock.
2) The customer lifts the IntelliPIN from its dock, swipes the card, enters the PIN, and returns the IntelliPIN to the dock.
3) The host collects and processes the card data.
4) The host sends a PIN entry command (request 30 for example).
5) The host collects the PIN data request.
6) The host sends a cancel (72) to restore display to "Welcome".

The main difference is that the IntelliPIN needs to be lifted only once for both the card swipe and PIN entry requests.

The IntelliPIN needs to emulate the display (and error codes) of the PIN entry command that will be used. The request has three different formats depending on the PIN request it will imitate:

Type 1a) Display an amount, alternating with a "Please enter PIN" message.
Type 1b) Display a "Please enter PIN" message.
Type 2) Display a user defined message without any amount.

Request:

Type 1 <STX>83[LINE1]<FS>[LINE2]<FS>[REQ][AMT]<ETX>{LRC}

Type 2 <STX>83[LINE1]<FS>[LINE2]<FS>[REQ][M1L1]<FS>[M1L2]<FS>
[M2L1]<FS>[M2L2]<ETX>{LRC}

[REQ] = 0 (type 1)

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
83	Request Type	2	Card Holder Data And PIN Entry Request
[LINE1]	Parameter	0-16	(optional) Message for Line 1
<FS>	Separator	1	(optional) Field Separator (0x1C) if [LINE1] < 16 chars
[LINE2]	Parameter	0-16	(optional) Message for Line 2
<FS>	Separator	1	(optional) Field Separator (0x1C) if [LINE2] < 16 chars
[REQ]	Parameter	1	0 (0x30) PIN entry display format
[AMT]	Amount	3-12	(Optional) If [AMT] is absent, only “Enter PIN then press Enter” will be displayed
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

[REQ] = 1 (type 2)

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
83	Request Type	2	Card Holder Data And PIN Entry Request
[LINE1]	Parameter	0-16	(optional) Message for Line 1
<FS>	Separator	1	(optional) Field Separator (0x1C) if [LINE1] < 16 chars
[LINE2]	Parameter	0-16	(optional) Message for Line 2
<FS>	Separator	1	(optional) Field Separator (0x1C) if [LINE2] < 16 chars
[REQ]	Parameter	1	1 (0x31) PIN entry display format
[M1L1]	Parameter	0-16	(Optional) Message 1, Line 1
<FS>	Separator	1	(Optional) Field Separator (0x1C) if [M1L1] < 16 chars
[M1L2]	Parameter	0-16	(Optional) Message 1, Line 2
<FS>	Separator	1	(Optional) Field Separator (0x1C) if [M1L2] < 16 chars
[M2L1]	Parameter	0-16	(Optional) Message 2, Line 1
<FS>	Separator	1	(optional) Field Separator (0x1C) if [M2L1] < 16 chars
[M2L2]	Parameter	0-16	(Optional) Message 2, Line 2
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Command Example:

In this example, it is assumed that before the 83 and 30 requests are sent, the IntelliPIN's Offset/Verify has been activated using the example commands in this document in the following sequence:

94 – Load Master Key
97 – Load Serial Number
58 – Activate the Offset/verify commands (See Appendix E for flow diagrams)

When this 83 request is sent, the unit will display **`Please swipe your card`** until the card is read. At that time, it will display **`Total $1.23`** and **`Please enter PIN then press ENTER`**. Enter "6565" as a PIN at this time and the IntelliPIN will return the card data and store the PIN (in an encrypted PIN block) for later transmission.

Request Example: <STX>83Please swipe<FS>your card<FS>0123<ETX>{LRC}

```
ASCII:  <STX>83Please swipe<FS>your card<FS>0123<ETX>[

Hex:  02 38 33 50 6C 65 61 73 65 20 73 77 69 70 65 1C 79 6F 75 72 20 63 61 72 6
      1C 30 31 32 33 03 5B

<STX>    02
Request  38 33                                    (83)
[LINE1]  50 6C 65 61 73 65 20 73 77 69 70 65      (Please swipe)
<FS>     1C
[LINE2]  79 6F 75 72 20 63 61 72 64               (your card)
<FS>     1C
[REQ]    30                                       (0)
[AMT]    31 32 33                                 (123)
<ETX>    03
{LRC}    5B                                       ([)
```

The card is swiped and “6565” is entered on IntelliPIN’s keypad. At this time, the IntelliPIN is returned to the dock (if not already there) and the card data is sent to the host.

Response: See Command 81 CARD DATA RESPONSE.

90 LOAD INITIAL KEY REQUEST

Command Set: DUKPT

Purpose: To load the initial PIN encryption key and the corresponding Key Serial Number.

This should be done in a secure environment.

Command Notes: After the initialization of 21 future keys, the IntelliPIN will respond with the response 91 with Confirmation Value. If the IntelliPIN receives a PIN entry request before request 90, the error message **`Initial Key has not been loaded`** is displayed.

Request: <STX>90[IPEK][KSN]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
90	Request Type	2	Load Initial Key Request
[IPEK]	Parameter	16	Initial PIN Encryption Key (hexadecimal)
[KSN]	Parameter	20	Key Serial Number (hexadecimal) (leading F's included)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Load the initial key of `123456789ABCDEF0` with a Key Serial Number of `FFFF9876543210E00000`.

```
ASCII:  <STX>90123456789ABCDEF0FFFF9876543210E00000<ETX>X

Hex:  02 39 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 30 46 46 46 46 39
      38 37 36 35 34 33 32 31 30 45 30 30 30 30 30 03 78
```

<STX>	02	
Request	39 30	(90)
[IPEK]	31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 30	(123456789ABCDEF0)
[KSN]	46 46 46 46 39 38 37 36 35 34 33 32 31 30 45 30 30 30 30 30	(FFFF9876543210E00000)
<ETX>	03	
{LRC}	78	(x)

Response: See Command 91 LOAD INITIAL KEY RESPONSE.
Since it takes a while to compute the future keys, the response will be returned about 1 second after the command has been received and acknowledged.

91 LOAD INITIAL KEY RESPONSE

Command Set: DUKPT

Purpose: To indicate whether the initial PIN encryption key and corresponding Key Serial Number have been loaded.

Response Notes: The response contains a confirmation value to indicate if the Key and Serial Number parameters have been loaded.

Response: <STX>91[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
91	Response Type	2	Load Initial Key Response
[CS]	Parameter	1	Confirmation Value: • 0 (0x30) = Confirmed, initial key loaded • 1 (0x31) = Not loaded (request error or could not create keys, see below).
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
The IntelliPIN's case was opened	"Case switch is open"
The initial key was not loaded	"Initial Key has not been loaded"
Characters in the request's IPEK are not valid	"New Initial Key field is invalid"
Characters in the request's serial number are not valid	"Key Serial no. field is invalid"

Response Example:

```
ASCII:  <STX>910<ETX>;

Hex  02 39 31 30 03 3B

<STX>     02
Response  39 31  (91)
[CS]      30     (0)
<ETX>     03
{LRC}     3B     (;)
```

92 REINITIALIZATION REQUEST

Command Set: DUKPT

Purpose: To load a new initial PIN encryption key and/or a new Key Serial Number while the IntelliPIN is in service. This feature allows: 1) Extension of the IntelliPIN service life beyond the one million transaction limit, 2) Changing the IntelliPIN from use of one acquirer's derivation key to another's, 3) Recovery from possible compromise of a derivation key.

Command Notes: The IntelliPIN uses the current PIN encryption key to perform the inverse of the "special encrypt" function on the encrypted new initial PIN encryption key. This provides the Clear Text new initial PIN encryption key. This key is then used to encrypt, via the "special encrypt" function, the complement of the new Key Serial Number (excluding the 4 rightmost digits). If the leftmost 32 bits of this result (grouped as 8 hexadecimal digits) match the Check Value, the IntelliPIN performs the initialization and uses a new initial PIN encryption key as the "initial PIN encryption key" and the new Key Serial Number with 4 zeros concatenated to the right as the Key Serial Number.

After the initialization of 21 future keys, the IntelliPIN will respond with the response 93 with current Key Serial Number and Confirmation Value. If the load is successful, the current Key Serial Number will be based on the new Key Serial Number from request 92. If the load is not successful, the current Key Serial Number will be based on that which existed prior to receipt of request 92.

Request: `<STX>92[KSN][IPEK][CHECK]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
92	Request Type	2	Load Initial Key Request
[KSN]	Parameter	16	New Key Serial Number (hexadecimal) (*excluding* the four rightmost digits but *including* the leading F's)
[IPEK]	Parameter	16	New Initial PIN Encryption Key (hexadecimal) (encrypted under current key)
[CHECK]	Parameter	8	Check Value (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Reinitialize the IntelliPIN with new Key Serial Number of FFFF1234567890120000 which gets sent as FFFF123456789012 and the new Initial Pin Encryption Key encrypted as F9716B22F3E068E3 and a Check Value of E70496AD.

ASCII: **<STX>**92FFFF123456789012F9716B22F3E068E3E70496AD**<ETX>{0x03}**

```
Hex: 02 39 32 46 46 46 46 31 32 33 34 35 36 37 38 39 30 31 32 46 39 37 31 36 42
     32 32 46 33 45 30 36 38 45 33 45 37 30 34 39 36 41 44 03 03
```

<STX>	02	
Request	39 32	(92)
[KSN]	46 46 46 46 31 32 33 34 35 36 37 38 39 30 31 32	(FFFF123456789012)
[IPEK]	46 39 37 31 36 42 32 32 46 33 45 30 36 38 45 33	(F9716B22F3E068E3)
[CHECK]	45 37 30 34 39 36 41 44	(E70496AD)
<ETX>	03	
{LRC}	03	

Response: See Command 93 REINITIALIZATION RESPONSE.

93 REINITIALIZATION RESPONSE

Command Set: DUKPT

Purpose: To indicate whether the new Initial PIN Encryption Key and new Key Serial Number have been loaded.

Response Notes: The response contains the current Key Serial Number along with a Confirmation Value to indicate if the key and serial number parameters have been loaded.

Response: <STX>93[CKSN][CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
93	Response Type	2	Load Initial Key Response
[CKSN]	Parameter	20	Current Key Serial Number (hexadecimal) (leading F's included)
[CS]	Parameter	1	Confirmation Value: • 0 (0x30) = Confirmed, reinitialization complete • 1 (0x31) = Not Confirmed, error, see below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
The initial key was not loaded	"Initial Key has not been loaded"
All future keys have been used	"No More Key"
Characters in the request's IPEK are not valid	"New Initial Key field is invalid"
Characters in the request's serial number are not valid	"Key Serial No field is invalid"
The Characters in the check Value field are invalid	"Check Value field is invalid"
Check Value does not match	"Check Value does not match"

Response Example:

ASCII: **<STX>**93FFFF9876543210E000011**<ETX>**M

```
Hex:  02 39 33 46 46 46 46 39 38 37 36 35 34 33 32 31 30 45 30 30 30 30 31 31
      03 4D

<STX>     02
Response  39 33                                          (93)
[CKSN]    46 46 46 46 39 38 37 36 35 34 33 32 31 30 45   (FFFF9876543210E00001)
          30 30 30 30 31
[CS]      31                                             (1)
<ETX>     03
{LRC}     4D                                             (M)
```

94 LOAD MASTER KEY

Command Set: Master/Session Key

Purpose: To load the Master Key to the IntelliPIN in clear text.

This should be done in a secure environment.

Command Notes: The clear text Master Key should be loaded to the IntelliPIN in the secure environment. Loading Master Key will clear the Session Key and all Working Keys but will not affect the Multi-Master Keys

If the *Enable Key Parity* option is enabled and the parity of each byte of the key is NOT odd, an error will be generated and the key will be ignored.

Request: <STX>94[MK]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
94	Request Type	2	Load Master Key
[MK]	Parameter	16 or 32	Master Key (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example (Single-length Key): Load a Master Key of 23AB 4589 EF67 01CD. The KCV is 588161.

```
ASCII:  <STX>9423AB4589EF6701CD<ETX>{0x08}

Hex: 02 39 34 32 33 41 42 34 35 38 39 45 46 36 37 30 31 43 44 03 08

<STX>    02
Request  39 34                                          (94)
[MK]     32 33 41 42 34 35 38 39 45 46 36 37 30 31 43 44  (23AB4589EF6701CD)
<ETX>    03
{LRC}    08
```

Request Example (Double-length key): Load double-length Master key of 23AB 4589 EF67 01CD F48A 40B3 1004 9D75. The KCV is D19834.

```
ASCII:  <STX>9423AB4589EF6701CDF48A40B310049D75<ETX>{0x0C}

Hex:  02 39 34 32 33 41 42 34 35 38 39 45 46 36 37 30 31 43 44 46 34 38 41 34
      30 42 33 31 30 30 34 39 44 37 35 03 0C

<STX>    02
Request  39 34                                  (94)
[MK]     32 33 41 42 34 35 38 39 45 46 36       (23AB4589EF6701CDF48A40B310049D75)
         37 30 31 43 44 46 34 38 41 34 30
         42 33 31 30 30 34 39 44 37 35
<ETX>    03
{LRC}    0C
```

Response: <STX>94[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
94	Response Type	2	Load Master Key Response
[CS]	Parameter	1	Confirmation Value as shown below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	No change
Key is not 16 or 32 characters	C (0x63)	“Bad Key Data”
Key parity is incorrect	E (0x65)	“Key Parity is Bad”

Response Example:

ASCII: **<STX>**940**<ETX>**>

Hex: 02 39 34 30 03 3E

```
<STX>     02
Response  39 34  (94)
[CS]      30     (0)
<ETX>     03
{LRC}     3E     (>)
```

95 LOAD SESSION KEY

Command Set: Master/Session Key

Purpose: To load the Session Key to the IntelliPIN in clear text or encrypted under the Master Key.

Command Notes: If the *Enable Key Parity* option is enabled and the parity of each byte of the key is NOT odd, an error will be generated and the key will be ignored.

If the Session Key [SSK] is defined as a single-length key (16), Working Keys will be decrypted with a single DES operation. If the SSK is defined as a double-length key (32), all Working Keys will be decrypted with a triple-DEA operation.

Request: <STX>95[ENC][SSK]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
95	Request Type	2	Load Session Key
[ENC]	Parameter	1	Key type flag: • C (0x43) = Clear Text • E (0x45) = Encrypted
[SSK]	Parameter	16 or 32	Session Key (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example (Single-length Key) : Load an encrypted Session Key of 99E1 E835 662D EA94. This is a Session Key of F48A 40B3 1004 9D75 encrypted under a Master Key of 23AB 4589 EF67 01CD. The KCV is B6B812.

ASCII: **<STX>**95E99E1E835662DEA94**<ETX>**:

Hex: 02 39 35 45 39 39 45 31 45 38 33 35 36 36 32 44 45 41 39 34 03 3A

<STX>	02	
Request	39 35	(95)
[ENC]	45	(E)
[SSK]	39 39 45 31 45 38 33 35 36 36 32 44 45 41 39 34	(99E1E835662DEA94)
<ETX>	03	
{LRC}	3A	(:)

Response: <STX>95[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
95	Response Type	2	Load Session Key Response
[CS]	Parameter	1	Confirmation Value as shown below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	No change
Bad Encryption type flag (not E or C)	i (0x69)	"Missing 'E' or 'C'"
Master Key not ready	k (0x6B)	"Selected key not ready"
Bad Session Key data	c (0x63)	"Bad Key Data"
Key Parity is Incorrect	e (0x65)	"Key Parity is Bad"

Response Example:

ASCII: **<STX>**950**<ETX>**?

Hex: 02 39 35 30 03 3F

```
<STX>     02
Response  39 35  (95)
[CS]      30     (0)
<ETX>     03
{LRC}     3F     (?)
```

96 LOAD WORKING KEY

Command Set: Master/Session Key

Purpose: To load a Working Key to the IntelliPIN encrypted under the Session Key. (See Appendix E for more information about Working Keys.)

Command Notes: This request cannot be used until the Session Key has been loaded.

If the *Enable Key Parity* option is enabled and the parity of each byte of the key is NOT odd, an error will be generated and the key will be ignored.

If Working Key [WK] is a single-length key (16), operations using this key will be single DES encrypted. If the WK is a double-length key (32), all encryptions will use the triple-DEA method.

Request: <STX>96[KN][WK]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
96	Request Type	2	Load Working Key
[KN]	Parameter	1	Working Key Number (total of 56 Keys) • '0' to '3' (0x30 to 0x33) = lower Working Key • 'A' to 'Z' (0x41 to 0x5A) = upper Working Key • 'a' to 'z' (0x61 to 0x7A) = upper Working Key
[WK]	Parameter	16 or 32	Encrypted Working Key (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example (Single-length Key): Load Working Key number 1 of 4FF4 4FF4 4FF4 4FF4 into the IntelliPIN. When encrypted under the Session Key of F48A 40B3 1004 9D75, the result is D2CE 4DF9 FAF7 E562. The KCV is C806B6.

```
ASCII:  <STX>961D2CE4DF9FAF7E562<ETX>@

Hex: 02 39 36 31 44 32 43 45 34 44 46 39 46 41 46 37 45 35 36 32 03 40

<STX>    02
Request  39 36                                              (96)
[KN]     31                                                 (1)
[WK]     44 32 43 45 34 44 46 39 46 41 46 37 45 35 36 32    (D2CE4DF9FAF7E562)
<ETX>    03
{LRC}    40                                                 (@)
```

Response: <STX>96[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
96	Response Type	2	Load Working Key Response
[CS]	Parameter	1	Confirmation Value as shown below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	No change
Session Key not loaded	k (0x6B)	“Selected Key not Ready”
Bad Working Key data	c (0x63)	“Bad Key Data”
Decrypted Working Key Parity is Bad	e (0x65)	“Key Parity is Bad”
Working Key number bad	u (0x75)	“Working Key not 0-3, A-Z, or a-z”

Response Example:

ASCII: **<STX>**960**<ETX>**<

Hex: 02 39 36 30 03 3C

<STX> 02
Response 39 36 (96)
[CS] 30 (0)
<ETX> 03
{LRC} 3C (<)

97 LOAD KEY SERIAL NUMBER

Command Set: Master/Session Key

Purpose: To load the Key Serial Number to the IntelliPIN in clear text. The KSN is used with the Activate Command 58. The Key Serial Number can be retrieved with the 55 Command.

Request: <STX>97[KSN] <ETX> {LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
97	Request Type	2	Load Key Serial Number
[KSN]	Parameter	16	Key Serial Number (hexadecimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Load a Serial Number of 0123 4567 89AB CDEF.

ASCII: **<STX>**970123456789ABCDEF**<ETX>{0x0B}**

Hex: 02 39 37 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 03 0B

```
<STX>    02
Request  39 37                                          (97)
[KSN]    30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46  (0123456789ABCDEF)
<ETX>    03
{LRC}    0B
```

Response: <STX>97[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
97	Response Type	2	Load Key Serial Number Response
[CS]	Parameter	1	Confirmation Value as shown below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	No change
Key Serial Number is not 16 digits or has non-hexadecimal digits	f (0x66)	"Key Serial no. field is invalid"

Response Example:

ASCII: **<STX>**970**<ETX>**=

Hex: 02 39 37 30 03 3D

<STX>	02	
Response	39 37	(97)
[CS]	30	(0)
<ETX>	03	
{LRC}	3D	(=)

98 DELETE KEYS

Command Set: Master/Session Key, Multi-Master Key, and DUKPT

Purpose: To delete the Encryption Keys.

Command Notes: The substitution table and Key Serial Number will also be deleted if the "All keys" option (A) is selected. If Working Keys option is selected, only the Working Keys will be deleted.

Request: <STX>9800=00[KEYS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
98	Request Type	2	Delete Keys
00=00	Constant	5	Verification flag (0x30 0x30 0x3D 0x30 0x30)
[KEYS]	Parameter	0 or 1	(optional) Keys to delete: • A (0x41)= All Master/Session Keys (default) • M (0x4D)= Multi-Master only • W (0x57)= Working Keys only
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Notes:

If the [KEYS] parameter is omitted, all keys except the Multi-Master Keys will be deleted.

Request Example: Clear all the Master/Session Keys.

```
ASCII:  <STX>9800=00<ETX>?

Hex: 02 39 38 30 30 3D 30 30 03 3F

<STX>     02
Request   39 38           (98)
Constant  30 30 3D 30 30  (00=00)
<ETX>     03
{LRC}     3F              (?)
```

Response: <STX>98[CS]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
98	Response Type	2	Delete Keys Response
[CS]	Parameter	1	Confirmation Value as shown below
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Values:

Condition	[CS]	Display
No error	0 (0x30)	No change
The verification flag, was missing or not exactly 00=00 or [KEYS] not A, M, or W	q (0x71)	“User Data field is invalid”
One or more of the keys were not zeroed out (possible hardware failure)	1 (0x31)	No change

Response Example:

ASCII: **<STX>**980**<ETX>**>

Hex: 02 39 38 30 03 3D

<STX> 02
Request 39 38 (98)
[CS] 30 (0)
<ETX> 03
{LRC} 3e (>)

99 SET/RETRIEVE DSN

Command Set: Any

Purpose: To set or retrieve the Device Serial Number

Request: <STX>99[DSN]<ETX>{LRC}

Type	Field	Length	Description
<STX>		1	Start of Text (0x02)
99	Command Type	2	Set/Retrieve DSN Request
[DSN]	Parameter	0 or 16	Device Serial Number (AlphaNumeric)* if not present, return DSN if 16 characters, set as DSN else return an error (see Confirmation Value below)
<ETX>		1	End of Text (0x03)
{LRC}		1	Error Check Character

*Only supports ASCII Characters 0x20 through 0x7E

Response: <STX>99[CS][DSN]<ETX>{LRC}

Type	Field	Length	Description
<STX>		1	Start of Text (0x02)
99	Command Type	2	Set/Retrieve DSN Response
[CS]	Parameter	1	Confirmation Value as shown below
[DSN]	Parameter	16	Device Serial Number (Alphanumeric) (present only if DSN is retrieved)
<ETX>		1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Value:

Condition	[CS]	Display
No error	'0' (0x30)	(no change)
DSN data was invalid (not 16 characters or not in range of 0x20 to 0x7E)	'q' (0x71)	"User Data field is invalid"

Request Example #1 (Set the DSN):

Set the DSN to "0123456789ABCDEF".

```
ASCII:  <STX>990123456789ABCDEF<ETX>{0x05}

Hex:  02 39 39 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 03 05

<STX>    02
Request  39 39                                          (99)
[DSN]    30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46  (0123456789ABCDEF)
<ETX>    03
{LRC}    05
```

Response Example #1 (Set the DSN):

```
ASCII:  <STX>990<ETX>3

Hex:  02 39 39 30 03 33

<STX>      02
Response:  39 39  (99)
[CS]       30     (0)
<ETX>      03
{LRC}      33     (3)
```

Request Example #2 (Retrieve the DSN):

```
ASCII:  <STX>99<ETX>{0x03}

Hex:  02 39 39 03 03

<STX>    02
Request  39 39  (99)
<ETX>    03
{LRC}    03
```

Response Example #2 (Retrieve the DSN):

```
ASCII:  <STX>9900123456789ABCDEF<ETX>5

Hex:  02 39 39 30 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 03 35

<STX>     02
Response  39 39                                          (99)
[CS]      30                                             (0)
[DSN]     30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46  (0123456789ABCDEF)
<ETX>     03
{LRC}     35                                             (5)
```

Q1 DISPLAY SWIPE CARD

Command Set: Any

Command Notes: The IntelliPIN displays the message `Swipe Card` until either the customer slides the card through the card reader slot or the IntelliPIN receives another command. The IntelliPIN automatically reads the card data, formats and returns it to the PC while displaying the `PINPad is processing` message. The Q4 Command must have been sent to enable card reading.

Request: `<STX>Q1<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Q1	Request Type	2	Slide Card Display
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example:

```
ASCII:  <STX>Q1<ETX>C

Hex:  02 51 31 03 63

<STX>    02
Request  51 31  (Q1)
<ETX>    03
{LRC}    63     (c)
```

This message tells the IntelliPIN to display the message `Swipe Card`.

Response:

There is no response to this command.

Q2 INDICATE HOST DONE

Command Set: Any

Purpose: To let the customer know that the transaction is complete.

Command Notes: When the IntelliPIN receives the Q2 Command, it displays the Thank you message (message 21) for three seconds, followed by the idle prompt.

Note

The Card Reader will not accept card swipes while displaying the "Thank you" message.

Request: <STX>Q2<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Q1	Request Type	2	Indicate Host Done
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example:

ASCII: **<STX>**Q2**<ETX>**`

Hex: 02 51 32 03 60

```
<STX>    02
Request  51 32  (Q2)
<ETX>    03
{LRC}    60     (‘)
```

Response: There is no Response to this command.

Q4 TURN CARD READER ON/OFF

Command Set: Any

Purpose: To turn the Card Reader on or off.

Command Notes: The PC may send this command to toggle the card reader. Q4 tells the IntelliPIN to allow data entry from the card reader (Interactive mode only) and does not affect the IntelliPIN display. If the optional flag character is omitted, "0" is assumed. The Q1 Command should be used to cause the unit to show **`Swipe Card`**.

Request: <STX>Q4[FLG]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Q4	Request Type	2	Turn Card Reader On
[FLG]	Parameter	1 Optional	Card Reader Status: '0' (0x30) = On '1' (0x31) = Off
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example:

```
ASCII:  <STX>Q40<ETX>V

Hex:  02 51 34 30 03 56

<STX>    02
Request  51 34  (Q4)
[FLG]    30     (0)
<ETX>    03
{LRC}    56     (V)
```

Response: There is no direct response to this command. However, when a card is swiped, the card data will be transmitted. See Command 81 CARD DATA RESPONSE.

Z1 CANCEL SESSION REQUEST

Command Set: Master/Session Key and DUKPT

Purpose: To return the PINPad to its idle state.

Command Notes: This request is used to cancel/abort the following:

PIN Entry: 30, 31, 32, 60, 70, 74, Z60
Data Entry: 40, 41, 62, 64, 80

This request does not display **`Cancel requested`** but causes the unit to return to its idle state (**`Welcome`**).

Request: <STX>Z1<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z1	Request Type	2	Cancel Session Request
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Cancel the current session.

ASCII: **<STX>**Z1**<ETX>**h

Hex: 02 5A 31 03 68

```
<STX>    02
Request  5A 31  (Z1)
<ETX>    03
{LRC}    68     (h)
```

Response: This request has no response.

Z2 DISPLAY A STRING

Command Set: Master/Session Key and DUKPT

Purpose: To display a single message on the IntelliPIN.

Command Notes: The IntelliPIN optionally clears the display then displays a message until the CLEAR key is pressed or it receives another display request from the PC. The <SUB> parameter is used to clear the display prior to showing the message.

Request: <STX>Z2<SUB>[MESSAGE]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z2	Request Type	2	Display a String Request
<SUB>	Parameter	1	(Optional), (0x1A) if present, clears display first
[MESSAGE]	Parameter	0 - 32	Display message
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Errors Displays:

Errors	Display
Message has illegal values	"Display Data is invalid"

Request Example: Display **`Test of Z2 Command`** on the IntelliPIN (appending to display if anything present.)

```
ASCII:  <STX>Z2Test of Z2      Command<ETX>{

Hex:  02 5A 32 54 65 73 74 20 6F 66 20 5A 32 20 20 20 20 20 20 43 6F 6D 6D 61 6E
      64 03 7B
```

<STX>	02	
Request	5A 32	(Z2)
[MESSAGE]	54 65 73 74 20 6F 66 20 5A 32 20 20 20 20 20	(Test of Z2
	20 43 6F 6D 6D 61 6E 64	Command)
<ETX>	03	
{LRC}	7B	({)

Response: This request has no response.

Z3 DISPLAY ROTATING MESSAGES

Command Set: Master/Session Key and MMK

Purpose: Display up to 2 messages in rotation with optional clear screen.

Request: <STX>Z3[CNT]<SUB>[MSG1]<FS>[MSG2]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z3	Request Type	2	Display a String Request
[CNT]	Parameter	1	not used
<SUB>	Parameter	1	(Optional) (0x1a) has no effect
[MSG1]	Parameter	0-32	Message 1
<FS>	Field Separator	1	Present only if two display lines or [MSG1] less than 32 characters
[MSG2]	Parameter	0-32	Message 2
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

The IntelliPIN displays the message until the **CLEAR** key is pressed or it receives another command from the host that displays a new message.

Error Display:

Error	Display
[MSG] not ASCII 32 - 126	“Display Data is invalid”

Response: This request has no response.

Z8 RESET/SET IDLE PROMPT

Command Set: Master/Session Key and DUKPT

Purpose: To set or reset the IntelliPIN idle prompt display.

Command Notes: To reset the PINPad idle prompt to **`Welcome`**, omit the [PROMPT] parameter (i.e., just send <STX>Z8<ETX>{LRC}).

Request: <STX>Z8[PROMPT]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z8	Request Type	2	Reset / Set Idle Prompt Request
[PROMPT]	Parameter	0 - 32	(optional) Line 1 and 2 of the idle prompt
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Errors	Display
Message has illegal values	"Display Data is invalid"

Request Example: Set the Idle Prompt to **`New Idle Prompt`**.

```
ASCII:  <STX>Z8New Idle Prompt<ETX>=

Hex: 02 5A 38 4E 65 77 20 49 64 6C 65 20 50 72 6F 6d 70 74 03 3(

<STX>    02
Request  5A 38                                          (Z8)
Prompt   4E 65 77 20 49 64 6C 65 20 50 72 6F 6D 70 74   (New Idle Prompt)
<ETX>    03
{LRC}    3D                                             (=)
```

Response: This request has no response.

Z42 REQUEST NONCODED KEYSTROKE

Command Set: Any

Purpose: To request a noncoded customer key entry from the IntelliPIN.

Command Notes: This message does not change the IntelliPIN display. Message Z2, Display a String, prompts the customer to press a key. Use message Z42 to request the single key entry from the IntelliPIN and Message 43, Return Noncoded Key, to return the message to the controller. This message tells the IntelliPIN not to code the result for transmission.

This command also specifies how long the IntelliPIN should wait for the single key entry before timing out.

Request: <STX>Z42[TIME]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z42	Request Type	3	Request Noncoded Key
[TIME]	Parameter	3	Time in Seconds (000 - 255) to wait for key to be pressed
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example:

```
ASCII:  <STX>Z42045<ETX>n

Hex:  02 5A 34 32 30 34 35 03 4E

<STX>    02
Request  5A 34 32  (Z42)
[TIME]   30 34 35  (045)
<ETX>    03
{LRC}    4E        (n)
```

Request a single key entry from the IntelliPIN. Wait up to 45 seconds for the entry.

Response: See Z43 RETURN NONCODED KEY.

Z43 RETURN NONCODED KEYSTROKE

Command Set: Any

Purpose: To return a response to Message Z42 Request Noncoded Keystroke.

Command Notes: Message Z43 returns the Single Key Value in the same format it was entered at the keypad. If an error has been detected in the Z42 command, no response is returned. The Function Keys are ignored.

Error Displays:

Errors	Display
Invalid [TIME] value	"Timeout not 000 - 255"

Response: <STX>Z43[KEY]<ETX>{LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z43	Request Type	3	Return Noncoded Key
[KEY]	Parameter	1	Key Values (See below). '?' (0x35) if timed out.
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Example:

ASCII: **<STX>**Z43*<**ETX**>t

Hex: 02 5A 34 33 2A 03 74

```
<STX>    02
Request  5A 34 33  Z43
[KEY]    2A        (*)
<ETX>    03
{LRC}    74        (t)
```

This message indicates the customer pressed **CLEAR.**

Key Values:

Key Pressed		1	2	3	4	5	6	7	8	9	0	ENTER	CLEAR	F1	F2	F3
[KEY]	ASCII	1	2	3	4	5	6	7	8	9	0	#	*	n/a	n/a	n/a
	Hex	31	32	33	34	35	36	37	38	39	30	23	2A	-	-	-

Z60 PRE-AUTHORIZATION: PIN ENTRY REQUEST

Command Set: DUKPT

Purpose: To obtain a PIN from the customer in the form of an encrypted PIN block.

Command Notes: This request may be preceded by a display request 42, 43 or Z2.

The 16-digit encrypted PIN block, along with the Key Serial Number, will be returned to the PC. The PC can then decrypt the PIN block to recover the PIN.

If the customer presses the CLEAR key without entering a PIN, an EOT (0x04) character will be returned in place of the response. The IntelliPIN will display **`Cancel requested`** for two seconds then revert to the **`Welcome`** message. If at least one digit has been typed, the IntelliPIN clears the entry and redisplays the previous message and restarts IntelliPIN entry.

Request Z1 or 72 from the PC will cancel the operation and return to the idle state.

After a PIN has been entered, the IntelliPIN displays **`PINPad is processing`** until the CLEAR key is entered or another request is sent to the IntelliPIN.

The format of the PIN block is set by Soft Switch B, see Command 50.

Request: `<STX>Z60.[ACCT]<ETX>{LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z60	Request Type	3	Pre-Authorization: PIN Entry Request
.	Separator	1	(REQUIRED) Period (0x2E)
[ACCT]	Parameter	0 - 19	Account Number including check digit (decimal)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Request Example: Request a PIN block from the IntelliPIN with an account number of `4012345678909`. This assumes command 42, 43, or Z2 was just sent to the IntelliPIN.

```
ASCII:  <STX>Z60.4012345678909<ETX>M

Hex:  02 5A 36 30 2E 34 30 31 32 33 34 35 36 37 38 39 30 39 03 4D

<STX>    02
Request  5A 36 30                                    (Z60)
period   2E                                          (.)
[ACCT]   34 30 31 32 33 34 35 36 37 38 39 30 39      (4012345678909)
<ETX>    03
{LRC}    4D                                          (M)
```

Response: See Command 71 PIN ENTRY RESPONSE (DUKPT).

Z62 ACCEPT AND ENCRYPT PIN (WITH CUSTOM PROMPTS)

Command Set: Multi-Master Key

Purpose: To get a PIN from the customer with custom prompts and configuration.

Command Notes: This command is only available on IntelliPIN that use the MagTek firmware 30037367, 30037397, 30037447.

Because the display is not cleared before displaying [PROCMSG], its length must be equal to or greater than [MSG1]

Request: `<STX> Z62. [ACCT] <FS> [WRKKEY1] [MINPIN] [MAXPIN] [NULLKEY] [MSG1] <FS> [MSG2] <FS> [PROCMSG] <ETX> {LRC}`

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z62	Request Type	2	PIN Entry Request
.	Separator	1	Period/Dot (0x2E)
[ACCT]	Parameter	8-19	Account Number including check digit (decimal, 0x30 to 0x39)
<FS>	Separator	1	Field Separator (0x1C)
[WRKKEY1]	Parameter	16	Working Key encrypted using current Master Key; if zero filled, current Master Key used as Working Key
[MINPIN]	Parameter	2	Minimum acceptable PIN length, decimal ‘01’ to ‘16’ (0x30 0x31 to 0x31 0x36)
[MAXPIN]	Parameter	2	Maximum acceptable PIN length, (decimal ‘01’ to ‘16’ (0x30 0x31 to 0x31 0x36)
[NULLKEY]	Parameter	1	Not used but must be correct and present ‘Y’ (0x59) or ‘N’ (0x4E)
[MSG1]	Parameter	0-16	[MSG1] Message to alternate with [MSG2] until customer presses the first key of their PIN
<FS>	Separator	1	Field Separator (0x1C)
[MSG2]	Parameter	0-16	[MSG2] Message to alternate with [MSG1] until customer presses the first key of their PIN
<FS>	Separator	1	Field Separator (0x1C)
[PROCMSG]	Parameter	0-16	[PROCMSG] Process message to display after PIN entry is complete
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Error Displays:

Error	Display
[ACCT] is wrong length or not hex	"Account number is invalid"
[WRKKEY1] not valid	"Bad key data"
[MSG1], [MSG2] or [PROCMSG] not valid	"Display data is invalid"
[WRKKEY1] did not decrypt correctly	"Key Parity is bad"
'.' (dot) missing after Z62	"Missing '.'"
One or more of the Field Separators missing	"Missing field separator"
[WRKKEY1] not loaded	"Selected Key not ready"
[MINPIN], [MAXPIN] or [NULLKEY] not valid	"User data field is invalid"

Request Example: Obtain an encrypted PIN block from the customer assuming the following:

- MMK #3 loaded and selected as shown in the request examples for commands 02 and 08 in this manual
- Account Number: 00001234567812349
- PIN Entered: 1234

```
ASCII:  <STX>Z62.00001234567812349<FS>CCC707472B6AF6CE0406NHAVE YOUR
        CARD?<FS>PLEASE ENTER PIN<FS>Thank You       Have a nice day<ETX>;

Hex:  02 5A 36 32 2E 30 30 30 30 31 32 33 34 35 36 37 38 31 32 33 34 39 1C 43
      43 43 37 30 37 34 37 32 42 36 41 46 36 43 45 30 34 30 36 4E 48 41 56 45
      20 59 4F 55 52 20 43 41 52 44 3F 1C 50 4C 45 41 53 45 20 45 4E 54 45 52
      20 50 49 4E 1C 54 68 61 6E 6B 20 59 6F 75 20 20 20 20 20 20 20 48 61 76
      65 20 61 20 6E 69 63 65 20 64 61 79 03 3B
```

<STX>	02	
Request	5A 36 32	(Z62)
Separator	2E	(.)
[ACCT]	30 30 30 30 31 32 33 34 35 36 37 38 31 32 33 34 39	(00001234567812349)
<FS>	1C	
[WRKKEY1]	43 43 43 37 30 37 34 37 32 42 36 41 46 36 43 45	(CCC707472B6AF6CE)
[MINPIN]	30 34	(04)
[MAXPIN]	30 36	(06)
[NULLKEY]	4E	(N)
[MSG1]	48 41 56 45 20 59 4F 55 52 20 43 41 52 44 3F	(HAVE YOUR CARD?)
<FS>	1C	
[MSG2]	50 4C 45 41 53 45 20 45 4E 54 45 52 20 50 49 4E	(PLEASE ENTER PIN)
<FS>	1C	
[PROCMSG]	54 68 61 6E 6B 20 59 6F 75 20 20 20 20 20 20 20 48 61 76 65 20 61 20 6E 69 63 65 20 64 61 79	(Thank You Have a nice day)
<ETX>	03	
{LRC}	3B	(;)

Response Example: See 71 PIN ENTRY RESPONSE (MMK)

Z66 REQUEST MAC

Command Set: MMK

Purpose: To create a Message Authentication Code (MAC) for a given message.

Command Notes: This request is used by the host to generate a MAC for a given set of values. These values may be supplied in ASCII or binary format. In the ASCII mode, up to 224 characters can be processed at once. In binary mode, the characters are sent as two hexadecimal values per byte so the total number of characters that can be processed at one time is halved (112 characters). If the data to be MACed is longer than these values, then it must be parsed into multiple messages. The MAC function may be called up to 100 times to create the final MAC. This limits the maximum message size to 22,400 characters for ASCII and 11,200 characters for binary.

The first and non-last MAC requests must have a [PKT] value of 1, 3, 5 or 7. These values will reset the MAC generation for the particular mode.

The final request is sent with a [PKT] value of 0, 2, 4 or 6.

If the message will fit into a single request, then the MAC will be returned when the IntelliPIN sees a final packet flag [PKT] with a sequence number [SEQNO] of “00”. (The IntelliPIN will return a Confirmation status of 0 (0x30) and the MAC value.)

If the message will not fit into a single request, each subsequent request must contain an incrementing sequence number [SEQNO]. (The IntelliPIN will return a Confirmation status of 1 (0x31) and no MAC value until the last request at which time it will return a Confirmation status of 0 (0x30) and the MAC value.)

Request Format:

<STX>Z66[PKT][SEQNO][MMK]<FS>[WRKKEY]<FS>[SECKEY]<FS>[MSGDAT]<ETX>{LRC}

Type	Field	Length	Description
<STX>		1	Start of Text (0x02)
Z66	Request Type	3	Request MAC
[PKT]	Parameter	1	Packet Type (See “Packet Type Values” below)
[SEQNO]	Parameter	2	Sequence Number (decimal) 00-99
[MMK]	Parameter	1	Optional: Master Key to use (decimal)
<FS>		1	Field Separator (0x1C)
[WRKKEY]	Parameter	0 or 32	Working Key as encrypted under the selected Multi Master Key
<FS>		1	Field Separator (0x1C)
[SECKEY]	Parameter	1	Optional: Secondary Key to use (decimal)
<FS>		1	Field Separator (0x1C)
[MSGDAT]	Parameter	0-224	Message to be MACed.
<ETX>		1	End of Text (0x03)
{LRC}		1	Error Check Character

Packet Type Values:

	BPI[1] specific MAC		ANSI Standard MAC	
	ASCII Data	Binary[2] Data	ASCII Data	Binary[2] Data
For first or middle of multiple packets, set to:	1	3	5	7
For last or only packet, set to:	0	2	4	6

1-BPI = Baseline Privacy Interface
2-Binary Data is passed as hex values

Request example #1 (Single part message):

Assumes:

MMK #7	0123 4567 89AB CDEF
Active MMK Master Key:	#7
Passed Master Key [MMK]:	none
Passed Working Key [WRKKEY]:	none
Passed Secondary Key [SECKEY]:	none
Data to be MACed [MSGDAT]:	11<FS>91827
MAC type	ANSI Standard MAC, ASCII data

Request for example #1:

The packet flag will be '4' because the data is ASCII and this is the last (and only) request message. The sequence number [SEQNO] "00" along with the packet type "4" (last) indicate that this is the only request and to return the MAC if the request is valid.

ASCII: **<STX>**Z66400**<FS><FS><FS>**11**<FS>**91827**<ETX>**X

Hex: 02 5A 36 36 34 30 30 1C 1C 1C 31 31 1C 39 31 38 32 37 03 58

```
<STX>     02
Request   5A 36 36                 (Z66)
[PKT]     34                       (4)
[SEQNO]   30 30                    (00)
[MMK]     (nothing passed)
<FS>      1C
[WRKKEY]  (nothing passed)
<FS>      1C
[SECKEY]  (nothing passed)
<FS>      1C
[MSGDAT]  31 31 1C 39 31 38 32 37  (11<FS>91827)
<ETX>     03
{LRC}     58                       (X)
```

Request example #2 (Multiple part message):

Assumes:

MMK #7	0123 4567 89AB CDEF
Active MMK Master Key:	#7
Passed Master Key [MMK]:	none
Passed Working Key [WRKKEY]:	none
Passed Secondary Key [SECKEY]:	none
Data to be MACed [MSGDAT]:	11 <FS> 918273645 <FS> <FS> 58143276 <FS> <FS> ;1234567890123456=991210000? <FS> 00012500 <FS> 9786534124876923 <FS>
MAC type	ANSI Standard MAC, ASCII data

Request for example #2:

The packet flag will be '5' for the intermediate messages and '4' for the final message. The sequence number starts at 00 and ends at 09. The data below is shown in ASCII with unprintable characters shown in angular brackets (e.g. <STX> for 0x02). The final MAC is `C156 F1B8 CDBF B451`

Step	Requests	Responses
1	<STX>Z665007<FS><FS><FS>11<FS>91827<ETX>n	<STX>Z671<ETX>I (send next)
2	<STX>Z665017<FS><FS><FS>3645<FS><FS>58<ETX>O	<STX>Z671<ETX>i (send next)
3	<STX>Z665027<FS><FS><FS>143276<FS><FS><ETX>@	<STX>Z671<ETX>i (send next)
4	<STX>Z665037<FS><FS><FS>;1234567<ETX>O	<STX>Z671<ETX>i (send next)
5	<STX>Z665047<FS><FS><FS>89012345<ETX>C	<STX>Z671<ETX>i (send next)
6	<STX>Z665057<FS><FS><FS>6=991210<ETX>K	<STX>Z671<ETX>i (send next)
7	<STX>Z665067<FS><FS><FS>000?<FS>000<ETX>b	<STX>Z671<ETX>i (send next)
8	<STX>Z665077<FS><FS><FS>12500<FS>97<ETX>d	<STX>Z671<ETX>i (send next)
9	<STX>Z665087<FS><FS><FS>86534124<ETX>D	<STX>Z671<ETX>i (send next)
10	<STX>Z664097<FS><FS><FS>876923<FS><ETX>R	<STX>Z670C156F1B8CDBFB451<ETX>e (final MAC)

RESPONSE: See Z67, RETURN MAC

Z67 RETURN MAC

Command Set: MMK

Purpose: To return the Message Authorization Code (MAC) for a given message.

Response Format: <STX>Z67[CS][MAC]<ETX>{LRC}

Type	Field	Length	Description
<STX>		1	Start of Text (0x02)
Z67	Response Type	3	Return MAC
[CS]	Parameter	1	See Confirmation Status below
[MAC]	Parameter	16	Returned MAC
<ETX>		1	End of Text (0x03)
{LRC}		1	Error Check Character

Confirmation Status:

[CS]	Description	[MAC] returned?
0	No error, MAC follows	Yes
1	Ready for next Z66 request	no
2	Sequence number [SEQNO] out-of-order	no
3	Master Key [MMK] not ready	no
4	Secondary Key [SECKEY] not ready	no
5	Message [MSGDAT] length invalid (too long or not even number in binary mode)	no
6	Packet flag [PKT] invalid	no
7	Message [MSGDAT] contents error	no
8	Working Key [WRKKEY] invalid	no

Error Response: <STX>Z67[CS]<ETX> {LRC}

Type	Field	Length	Description
<STX>	Start of Text	1	Start of Text (0x02)
Z67	Request Type	3	PIN Entry Response
[CS]	Parameter	1	Error code (see above)
<ETX>	End of Text	1	End of Text (0x03)
{LRC}		1	Error Check Character

Response from example #1:

Returned MAC is 356C 20A9 E603 04D9.

ASCII: **<STX>**Z670356C20A9E60304D9**<ETX>**h

Hex: 02 5A 36 37 30 33 35 36 43 32 30 41 39 45 36 30 33 30 34 44 39 03 68

<STX>	02	
Response	5A 36 37	(Z67)
[CS]	30	(0)
[MAC]	33 35 36 43 32 30 41 39 45 36 30 33 30 34 44 39	(356C20A9E60304D9)
<ETX>	03	
{LRC}	68	(h)

APPENDIX A. DEFAULT DISPLAY MESSAGES

The following display messages can be customized (messages must be 32 characters or less, 2 lines by 16 characters) by using command 51. Changes will remain in place until the default display messages are reloaded by using command 52.

Number 00 – Welcome

```
Welcome
```

This message is displayed after powering up and at the idle state in the Interactive Mode.

Number 01 – Enter PIN

```
Please enter PIN
then press ENTER
```

This message is displayed prior to PIN entry.

Number 02 – Processing

```
PINPad is
processing
```

This message is displayed after the IntelliPIN has received an ACK from the PC responding to the transmission of the encrypted PIN block.

Number 03 – Total

```
Total
```

This message is displayed to show the total of the amount field from the PC. If changed, it must be exactly 16 characters long. This can be done by padding with spaces.

Number 04 – Reenter PIN

```
Please re-enter
the PIN
```

This message is displayed after the first PIN entry when double PIN entry for verification is enabled.

Number 05 – Illegal PIN

```
Illegal PIN
entered ...
```

This message is displayed for 2 seconds when one or more of the following conditions exist:
- Length of the PIN entered is fewer than minimum number of digits.
- Length of the PIN entered exceeds the maximum set in SWC.
- Trivial PIN pattern is entered while the trivial PIN check is enabled.

Number 06 – PINs Do Not Match

```
PINs don’t match
Please re-enter
```

This message is displayed for 2 seconds when the first and second PIN entries do not match.

Number 07 – Cancel Requested

```
Cancel requested
```

This message is displayed when the PIN entry is aborted by pressing the CLEAR key at the beginning of PIN entry.

Number 08 – Connect PINPad to Dock

```
Place PINPad in
dock to continue
```

This message is displayed when the IntelliPIN needs to transmit the data to the PC and the IntelliPIN is not in the dock.

Number 09 – Select Yes or No

```
Please select
 Yes     No
```

This message is displayed for the transaction amount authorization request.

Number 10 – Bad Read

```
Bad reading
Swipe again?
```

This message is displayed after an unsuccessful card reading.

Number 11 – Please Swipe Your Card

```
Please swipe
your card
```

This message is shown when ready for a customer card in the PIN with card mode.

Number 12 – When No FITs Loaded

```
Ready for
program data
```

This message is shown in the standalone mode when no FIT table information has been loaded.

Number 13 – Unit Is Shut Down And Activate Card Is Required

```
Unit is
Shut Down
```

This message is shown when the unit has been shut down and an Activate Card or Program Card is required to place the unit in customer mode.

Number 14 (No longer used. Display 17 is used instead.)

Number 15 – When A PAN Can Be Key-Entered

```
Key enter PAN
or swipe a card
```

This message is shown in the standalone mode when the unit is set to accept key-entered PANs. When this message is shown, a customer card or Template card can be used.

Number 16 – When Switching Between Offset And Verify Modes

```
Select Mode:
Offset    Verify
```

This message is shown in the PIN and Verify mode after pressing the F2 function key to switch between offset generation and PIN verification modes.

Number 17 – After Downloading New Firmware

```
Transfer card
needed
```

This message is shown in the PIN and Verify mode after downloading new firmware and prior to loading a Master Key.

Number 18 – Unit Is Shut Down And Password Is Required

```
Unit is
Shut Down   Pswd
```

This message is shown when the unit has been shut down and a password is required to place the unit in customer mode.

Number 19 – When Reading Program Cards

```
Read next card
or press  *Done*
```

This message is shown after reading a Program Card in the standalone mode if more FIT entries are available.

Number 20 – Swipe Card

```
Swipe Card
```

This message is shown when a command 80 or 83 is executed.

Number 21 – Thank You

```
Thank You
```

This message is shown in place of an Offset, PVV, or authorization if the “Don’t Show” option is selected.

This Table shows all of the messages in the IntelliPIN. The first column shows the default message. The "Num" column shows the NUM value to be included in the 51 (Type 2) Command.

Existing Text	Num	Usage and notes
"**** WARNING ***Init flag failed"	184	
"**** WARNING ***Switches failed"	185	
"2nd key appended"	193	Second Master Key appended from Program Card
"2nd Master Key: Append Replace"	192	
"ABORT"	067	Sent when user abort in stand-alone mode
"Account Number field is invalid"	009	30, 34, 35, 36, 38, 65, 71(MMK and (DUKPT), Z66
"Ack/Nak:Disabled"	156	
"Ack/Nak:Enabled "	155	
"Activate card was read"	073	
"Actvat:Card Only"	124	
"Actvat:Card+Pswd"	125	
"Actvat:Pswd Only"	126	
"Amount field is invalid"	012	30, 63, 65, 71 (MMK and DUKPT), 81
"Auth # = %4.4s Count = %06.6ld"	064	%4.4s = 4 chars, %06.6ld = 6 chars
"AUTH=%4.4s"	063	%4.4s = 4 chars
"Authrz:Dont Show"	177	Config menu
"Authrz:Show "	178	Config menu
"Auto Shut Off?:N"	173	Config menu
"Auto Shut Off?:Y"	174	Config menu
"Bad card number or user aborted"	045	
"Bad Echo Flag"	032	41
"Bad Key Data"	019	20, 71 (MMK)
"Bad Maximum Length"	033	41
"Bad reading Swipe again? "	10	81
"Bad Switch Data"	031	50
"Battery is low, please recharge"	198	
"Baud: 300 "	131	Config menu
"Baud: 600 "	132	Config menu
"Baud: 1200 "	133	Config menu
"Baud: 2400 "	134	Config menu
"Baud: 4800 "	135	Config menu
"Baud: 9600 "	136	Config menu
"Baud: undefined "	137	Config menu
"Bksp"	175	
"Baud: undefined "	137	Config menu
"Cancel requested"	07	30, 32, 60, 74, 80, Z60
"Cannot use a Driver License"	052	
"Cannot use a system card"	048	
"Card Number: ____"	088	
"Card Reader Trks"	111	
"Case Switch is Open"	017	

Existing Text	Num	Usage and notes
“Change Password “	115	
“Char Rate: 20cps”	130	
“Char Rate: 30cps”	129	
“Char Rate: 40cps”	128	
“Char Rate: 80cps”	127	
“Check Digit = x (Press any key)”	062	
“Check Value field is invalid”	003	
“Check Value doesnot match”	004	
“Check-Digit error”	094	
“Clear error flagYes No”	182	
“Command not Activated”	025	31, 32, 33, 34, 35, 36, 57
“Communications “	110	
“Could not read the card, retry”	044	
“CTS/DSR: Ignore “	159	
“CTS/DSR: Use “	160	
“Current Count: %06.6ld”	093	
“Dbl PIN: Disable”	161	
“Dbl PIN: Enable “	162	
“DES FAILED”	187	07
“DES PASSED”	188	07
“Diebold table fully loaded”	060	
“Diebold Table went bad”	050	
“Display Data is invalid”	024	42, 43, 51, 81, 82, Z2, Z3, Z8
“Display Number is Incorrect”	016	51
“Enab Tracks: --- Acpt Skip”	120	
“Enter to (Press any key) “	176	
“Enter _____ Time: HHMM”	081	
“Enter _____ Date: MMDDYYYY”	087	
“Enter PAN: “	089	
“Enter Password ____”	097	
“Error flags cleared”	183	
“Error flags: “	186	
“FAILED”,	196	
“FIT Table BAD Clearing to zero”	043	
“FIT# Field Exit”	107	
“Function Word invalid “	039	20
“Generate Offset”	068	Mode x
“Illegal Diebold table index “	061	
“Illegal PIN entered “	05	Customize 51; Default 52
“Initial Key has not been loaded “	002	71 (DUKPT)
“Insert Hdr:No “	169	Config menu
“Insert Hdr:Yes “	170	Config menu
“Invalid Activate card “	096	
“Invalid BIN”	038	20, 37, 38, Part of LCD message
“INVALID BIN”	181	Output to PC
“Invalid Program card “	036	
“Key address is not 0-9”	180	
“Key Entry not Allowed “	095	
“Key enter PAN or swipe a card “	15	Standalone Mode. Cust or Temp crd

Existing Text	Num	Usage and notes
“Key Parity is Bad”	010	
“Key Parity Check”	114	
“Key Serial no. field is invalid”	001	58
“KeyParity:Check “	151	Config menu
“KeyParity:Ignore”	150	Config menu
“Length value is incorrect”	014	34, 35
“Master Key replaced”,	194	
“Missing ‘A’ or ‘D’“	029	58
“Missing ‘C’ or ‘D’“	007	63, 65, 71 (DUKPT)
“Missing ‘D’ or ‘H’“	027	57
“Missing ‘E’ or ‘C’“	020	
“Missing ‘.’“	013	71 (DUKPT), Z66
“Missing field separator”	006	30, 65, 71 (MMK and DuKPT)
“Mode not changed”	070	
“Mode:Interactive”	138	
“Mode:PIN w/Card “	139	
“Mode:PIN w/oCard”	140	
“Mode:PIN&Verify “	142	
“Mode:Verify Cust”	141	
“Mode:Verify Ofst”	143	
“New Initial Key field is invalid”	000	
“New Password ____”	084	
“New Password accepted”	092	
“Next Edit Exit”	108	Config menu control
“No card Tracks Active”	008	
“No Diebold FIT Table Loaded”	053	
“No FIT Tables Loaded”	054	20
“No More FIT Tables Available”	037	
“No More Key”	005	71 (DUKPT)
“No room for remaining FITs”	051	
“Not a system card”	049	
“Not recognized Invalid BIN”	047	
“Offset = %4.4s Count = %06.6ld”	065	
“Offset/PVV fieldinvalid”	040	20
“Offset:Dont Show”	171	Config menu
“Offset:Show “	172	Config menu
“Op.Time: ___ sec Acpt Skip”	121	Config menu
“Operat. Timeout “	116	
“Pad Char: __ (?) Acpt Skip”	119	
“PAN too short Press any key”	046	
“Parity: EVEN “	148	Config menu
“Parity: MARK/1 “	147	Config menu
“Parity: ODD “	149	Config menu
“Parity: SPACE/0 “	146	Config menu
“Passwords do not match”	091	
“PIN Blk:ANSI 9.8”	165	Config menu
“PIN Blk:IBM 3624”	166	Config menu
“PIN incorrect...Failed to verify”	056	
“PIN incorrect...Press CLEAR”	055	

Existing Text	Num	Usage and notes
“PIN Length: __ Acpt Skip”	118	
“PIN Options”	112	Config menu
“PINPad is Processing”	02	30, 31, 32, 34, 35, 36, 37, 38, 60, 74, 80, Z60
“PINs don’t matchPlease re-enter”	06	51, 52
“Place PINPad in doc to continue	08	
“Please enter PINthen press ENTER”	01	32, 51, 52, 30, 50, 70, 74
“Please re-enter the PIN”	04	Customize 51; Default 52
“Please select Yes No “	09	62, 64, 81
“Please swipe your card “	11	
“Please wait...”	074	
“Power Timeout “	113	
“PVKI not 0-9”	041	
“PVN size or typebad”	191	37, 38
“PVV = %4.4s Count = %06.6ld”	066	
“PwrTime: ___ min Acpt Skip”	122	
“Read next card or press *Done*”	086	
“Read next card or press *Done*”	19	Standalone Mde for FIT entry reads
“Ready for program data “	12	Used when no FIT info loaded
“ReEnter Password____”	085	
“REJECT”,	195	
“Scan Code: Auto “	152	Config menu
“Scan Code: Set 1”	153	Config menu
“Scan Code: Set 2”	154	Config menu
“Sel Acpt Skip”	083	
“Select Mode: Offset Verify”	16	
“Selected Key not Ready”	028	30, 31, 32, 33, 34, 35, 36, 56, 58
“Serial Number not loaded yet”	035	55, 58
“Serial Numbers do not match”	026	58
“Set Operate Mode”	109	
“Shut down? Yes No”	090	
“ShutDown Timeout”	117	
“ShutOff in:__hrs Acpt Skip”	123	
“Snd EOT on CLR:N”	158	Config menu
“Snd EOT on CLR:Y”	157	Config menu
“Substitution Table not Ready”	011	31, 32
“Swipe Card “	20	Shows when 80 or 83 executed
“Switch ID is Incorrect “	015	50
“Table Data field invalid”	034	57
“Table not loaded”	042	
“Thank You “	21	Used when PVV, Offset not shown
“Timeout not 000 - 255”	023	40, 41, Z43
“Too many retrieson password”	179	
“Total”	03	30, 62, 64, 70, 74
“Transfer card needed “	17	PIN and Verify before Master Ky Ld
“Triv PIN:Disable”	163	Config menu – Trivial PIN Check
“Triv PIN:Enable “	164	Config menu – Trivial PIN Check
“Unit initialized”	080	
“Unit is Shut Down “	13	Activate or Program Card required

Existing Text	Num	Usage and notes
“Unit is Shut Down Pswd”	18	Passwd needed for Customer Mode
“User Data field is invalid”	022	21, 23, 32, 33, 81, 99
“Validation Data field is invalid”	021	20, 23, 31, 32, 36, 37, 38
“Value should be 0 thru 3 “	082	20
“Value should be 0-5, A-Z or a-z “	018	30, 31, 33, 34, 35, 36, 56
“VeriFoneSsEs:Dis”	189	Config menu -
“VeriFoneSsEs:Enb”	190	Config menu -
“Verify Customer”	069	Mode 3 display
“VeriFoneSsEs:Dis”	189	Config menu -
“Welcome”	00	30, 32, 51, 52, 60, 74, 80, Z60
“Working Key NOT 0-3, A-Z or a-z “	030	
“XmitData:w/PAN “	167	Config menu
“XmitData:w/Trks “	168	Config menu
“Yes No “	072	Menu choice

APPENDIX B. PIN BLOCK FORMATS

The IntelliPIN supports ANSI 9.8 PIN and ISO 9564 block format and IBM 3624 PIN block format. This section also lists references to PIN Encryption/Decryption/Message Authentication Code.

ANSI 9.8 / ISO 9564 PIN Block Format

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
C	N	P	P	P	P	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	F	F

Note

The values 0000 to 1111 below are binary values.

C - Control field (Format number) = 0000 (Does not support Format 1 or Format 3)
N - PIN length entered field = 0100 to 1100 (4-12) (0x4 – 0xC)
P - PIN digit = 0000 to 1001 (0-9)
F - Fill digit = 1111 (F)
P/F - Pin digit or fill digit, as determined by PIN Length N.
PIN Length is 4 to 12.

IBM 3624 PIN Block Format

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
P	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F

P - PIN digit = 0000 to 1001 (0-9)
F - Fill digit = 0000 to 1111 (0-F)
P/F - PIN digit or fill digit, as determined by PIN length entered.
PIN Length is 1 to 16.

Primary Account Number Block Format

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	0	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12

A- The twelve rightmost digits of the primary account number (PAN), excluding the check digit. A1 is the most significant digit and A12 is the digit immediately preceding the PAN's check digit. If the primary account number excluding the check digit is less than twelve digits, the digits are right justified and padded on the left with zeroes. Permissible values are 0000 to 1001.

0 - Pad digit = 0000. The first four digits of the account number block are always padded with this value.

Formatted Clear-Text PIN Block

The PIN and account number blocks are Exclusive ORed before being assembled in the DES (Data Encryption Standard) input register. When the account number is not available, only the PIN block is assembled in the DES input register. An example of creating a PIN block is as follows:

Values used in this example:

PIN:	6565
Account Number:	4761234567812348
Master Key	23AB4589EF6701CD

For ANSI 9.8

First create the PIN Block as follows:				
1. The length of the PIN is inserted into the PIN Block field as one digit (the first digit is format code = 0)	**0 4**--	----	----	----
2. The PIN is entered immediately after the length	04**65**	**65**--	----	----
3. The remainder of the field is padded with F's until the length is 16 digits	0465	65**FF**	**FFFF**	**FFFF**
Next, the Account Number is processed as follows:				
1. Remove the check-digit (this is the final digit of this account number) (476123456781234**8**)	-476	1234	5678	1234
2. Insert up to the 12 rightmost digits of the remaining value into the field, right justified (476**123456781234**)	----	1234	5678	1234
3. Pad any empty digits to the left with zeros until the field it is 16 digits long. This is the "Padded Account Number"	0000	1234	5678	1234
Exclusive OR (XOR) the PIN Block with the Padded Account Number to make the Formatted Clear-Text PIN Block which is fed into the DES routine	0465	77CB	A987	EDCB
Do the DES function with the Master Key (as shown on the right)	23AB	4589	EF67	01CD
The results of the DES is the Encrypted PIN Block (this is the Formatted Clear-Text PIN Block encrypted under the Master Key)	**D5D6**	**DF8D**	**0DB8**	**97AB**

The encrypted PIN Block is **D5D6DF8D0DB897AB**

Key Representation

DES keys are used in a 64-bit form that includes a parity bit on each byte. The bits are numbered left to right.

When keys are written, they are represented in 64-bit form as sixteen hexadecimal characters (0-9, A-F) with a space between every pair of characters. Thus a valid DES key is 01 23 45 67 89 AB CD EF. There is an odd number of one-bits in every pair (i.e., the eight-bit byte contains an odd number of ones).

PIN ENCRYPTION / DECRYPTION / MESSAGE AUTHENTICATION CODE

The IntelliPIN meets the following standards when encrypting a PIN:

- DES Encryption: ANSI X3.92, American National Standard for Data Encryption Algorithm

- PIN Encryption: ANSI X9.8, American National Standard for Personal Identification Number (PIN) Management and Security

- Key Management: ANSI 9.24, American National Standard for Financial Services Retail Key Management

- VISA Point-of Sale Equipment Requirements: PIN Processing and Data Authentication, International, Version 1.0, August 2004

APPENDIX C. DEFINING THE CURRENCY CHARACTER

This Appendix describes how to define a new currency character as used in command 41 (String Input Request)

The Currency Character is defined using the grid below. Please note that the [H7] value is the descender line. Most Currency Characters will use the first seven values and have [H7] set to 00.

Notes:

- You can use this page for a permanent reference to the changed Currency Character or photocopy this page and make changes to the copy.
- Fill in each square for the part of the character you want to be black. These squares will be counted as '1's. Each blank square will be counted as a '0'.
- The left-most hex value will be either '0' (zero) or '1' (one).
- The right-most hex value is the sum of the '1' bits in the right four fields. To simplify this conversion, use the hex table below.
- To hide the Currency Character, set all the lines to zeros (**`51C0000000000000000`**).
- For examples, see below and Command 51 on page 79.

Hex Field	Write in Hex Value	1	8	4	2	1
[H0]						
[H1]						
[H2]						
[H3]						
[H4]						
[H5]						
[H6]						
[H7]						

(descender line)

Hex Table (these are the ASCII values to insert into the 51 command)

Hex	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1

Hex	8	4	2	1
8	1	0	0	0
9	1	0	0	1
A	1	0	1	0
B	1	0	1	1
C	1	1	0	0
D	1	1	0	1
E	1	1	1	0
F	1	1	1	1

Example: Change the Currency Character to a Yen symbol (¥)

Hex Field	Write in Hex Value	1	8	4	2	1	
[H0]	11	1	0	0	0	1	
[H1]	0A	0	1	0	1	0	
[H2]	1F	1	1	1	1	1	
[H3]	04	0	0	1	0	0	
[H4]	1F	1	1	1	1	1	
[H5]	04	0	0	1	0	0	
[H6]	04	0	0	1	0	0	
[H7]	00	0	0	0	0	0	(descender line)

Thus the command would be:
`51C110A1F041F040400`

The eighth line ([H7]) can be used if the character is too complex to be defined in the top seven lines or it requires a descender.

APPENDIX D. GLOSSARY

Address	One of the ten MMK numbers. The value range is 0 to 9.
Algorithm	A process used for security reasons to change plain, readable text into unreadable encrypted text. The encryption algorithm is used to calculate the PIN offset or PVV.
Baud Rate	Communication speed between two devices.
BIN	Bank Identification Number; 1 to 6 digits; first digits of PAN.
Customer Card	Usually a bank or ATM card that a financial institution issues to customer.
CPU	Central Processing Unit; refers to the IntelliPIN microprocessor.
CVV	Card Verification Value
DES	Data Encryption Standard. An algorithm developed in the1970s by the IBM Corporation, since adopted by the US government and ANSI (the American National Standards Institute) as the encryption standard for financial institutions.
DUKPT	The Derived Unique Key Per Transaction, or DUKPT, is a method which uses a derivation, or base, key to encrypt an initial key serial number which produces an initial PIN encryption key. There is a unique PIN encryption key for each transaction.
End Sentinel	A character used in laying out track data. It indicates the end of the data.
Field Separator	A character used in laying out the track data. It indicates a separation between fields of data.
FIT	Financial Institution Table. The FIT contains the BIN and other details regarding the generation of offset.
ILSK	Institution Loaded Security Key. This is a Security Key that the financial institutions loads into its IntelliPIN, rather than the Security Key installed by MagTek (MLSK).

Key Parity	Key Parity is used to ensure that a key used for encryption is valid. This is especially useful when decrypting a key to validate the key if each byte has odd parity. The Key Parity can be used to ensure that a Key has been correctly entered and, in the case of an encrypted key, that it is being decrypted under the proper key.
LCD	The Liquid Crystal Display is a 2-line by 16-character display that shows status, messages, and information on the magnetic stripe.
LED	The Light Emitting Diode is used for the power indicator on the dock.
MAC	Message Authentication Code
MCAT	MagTek Card Activating Terminal. Creates Program Cards used with the IntelliPIN.
MLSK	MagTek Loaded Security Key.
MMK	The Multi-Master Key functions provide storage for up to ten Master Keys. These keys are used in conjunction with the PIN Entry Request (MMK) commonly used in point of sale applications. The Multi-Master Keys are independent of any other keys stored in the IntelliPIN.
MSK	The Master/Session Key is used in encryption and decryption between the IntelliPIN and the host. Since there is no common key in either the IntelliPIN or the host at startup, the Master Key will be loaded in clear text; this should be done in a secure environment. The Master Key is used to encrypt the Session key before loading to the IntelliPIN. The Session Key can be used as a PIN encryption key to encrypt customer PINs for transmission to the PC. For additional security, the Session Key may be used to encrypt working keys which are used to encrypt customer PINs.
MSR	Magnetic Stripe Reader.
Offset	The PIN is encrypted through an algorithm into an offset, which cannot be deciphered to reveal the actual PIN. The offset is used to authenticate the user of the card.

PAN	Primary Account Number. It is the account or card number, which also includes the BIN. The PAN includes all the data between the Start Sentinel and the first Field Separator. The PAN length is usually from 13 to 19 digits.
PIN	Personal Identification Number. Customer's number used with a card.
Program Card	A card used to load data into the IntelliPIN. The card contains BINs, Offset information, and other information relevant to the institution. This card can be generated on an MCAT.
PVV	PIN Verification Values are encoded on the customer cards so that PINs can be verified at the ATM network switch or transmitted to the host for verification. The PVV is used by VISA to authenticate the user of a card.
RAM	Random Access Memory
Start Sentinel	A character used in laying out track data. It indicates the start of the data.
Transfer Card	A card generated on the MCAT used to transfer the Master key into the IntelliPIN. The Master key of both the MCAT and the IntelliPIN must be the same in order for the Program Cards to operate. This card can be generated on an MCAT.
Trivial PIN	If the Trivial PIN check is enabled, simple PIN's (e.g., 1234, 2222, etc.) are not allowed.

APPENDIX E. FLOW DIAGRAMS

The flow diagrams in this appendix include the Activation Sequence and Uploading Working Keys.

Activation Sequence for MSK:

To issue any of the requests that require activation before use (see below), the IntelliPIN must first be set to the *Activated* mode. To prevent these requests from being executed, the IntelliPIN must be set to the *Deactivated* mode. Any request that is designated as "Activation: N/A" will execute whether the IntelliPIN is Activated or Deactivated.

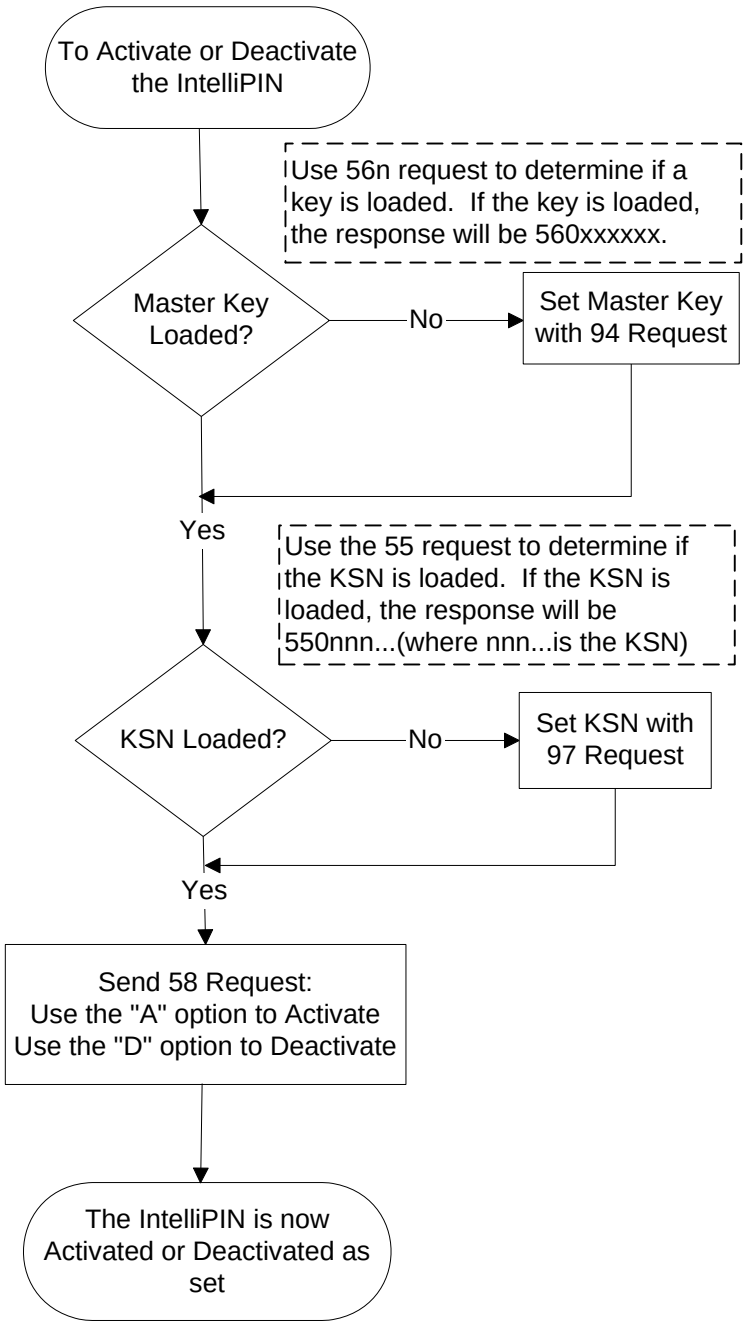
Note: The IntelliPIN will stay in the current Activated/Deactivated mode until it is either Shut Down or it powers itself off (i.e., a Power Time Out occurs.) When the IntelliPIN is started, it will be in the Deactivated mode.

To Activate or Deactivate the IntelliPIN, the Master Key (which is normally loaded at the factory or head office) and the Key Serial Number (KSN) must already be loaded. The Activate/Deactivate request (Request #58) requires the KSN in encrypted format (called the eKSN here). This is done using the Master Key as the encryption key and the KSN as the data. The reason for this is that, as a security measure, it verifies that whoever generated the eKSN knows both the Master Key and the plain text KSN.

Upon receipt, the 58 request decrypts the eKSN under its Master Key and compares this value with the stored KSN. If these two values match, the IntelliPIN will then set either Activated or Deactivated as given in the request.

The commands that must be Activated for use are:

31 – PIN Offset Request
32 – PIN Verification Request
33 – Encryption Test Request
34 – CVV Request
35 – PVV Request
36 – PVV Verification Request
57 – Load Substitution Table Request



Uploading Working Keys (MSK Only)

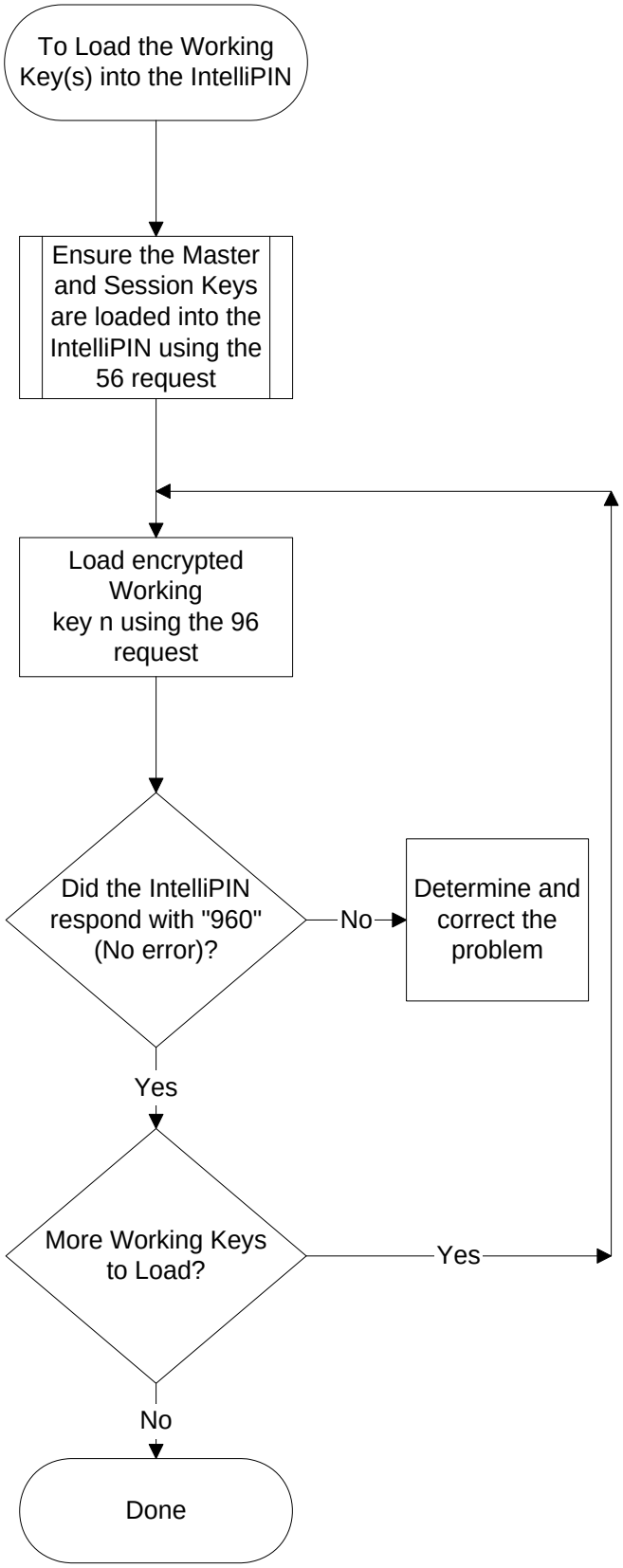
The Master Key and the Session Key must be present in the IntelliPIN before the Working Keys can be loaded.

There are 56 Working Keys numbered zero (0) to three (3), A - Z and a - z. The first four Working Keys (0 through 3) are known as the Lower Working Keys. The Alpha Keys are known as the Upper Working Keys.

The Working Keys are always sent in encrypted form to the IntelliPIN. They are encrypted under the Session Key. This prevents anyone from intercepting the Working Keys.

Use the 96 request to download the key. The response should always be "960" otherwise check the following:

Resp	Comment
96l	The Session Key is not ready. It must be loaded before loading Working Keys
96c	The encrypted Working Key data contains non-hex digits (0 to 9 and A to F) OR is not 16 digits
96e	The decrypted Working Key has invalid parity indicating that either the wrong Session Key is loaded or the Working Key data is incorrect
96u	The Working Key number is not in the range of 0 - 3, A - Z, or a - z.



APPENDIX F. ASCII CHART

This Appendix describes conversions among hex, ASCII, and decimal systems.

The chart below lists the following:

- The top row and left column are in hex.
- This chart shows only the 7-bit "normal" ASCII values. The 8-bit "extended" ASCII chart is non-standard.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

To find the hex value of a character:

1. Locate the character in the chart.
2. Note the value at the beginning of the row the character is in. This is the first digit of the hex value.
3. Note the value at the top of the column the character is in. This is the second digit of the hex value.

e.g.: What is the hex value of capital G?
Find 'G' in the chart. The row is numbered '4' and the column is numbered '7'. So the hex value is 0x47.

To find what character is represented by a give hex value, do the following:

1. Find the row with the first digit of the hex value. If it is a single character, the use row '0'.
2. Find the column with the second digit of the hex value.
3. The intersection of the row and column gives the character.

e.g.: What character is represented by 0x6E?
Locate row '6' on the left of the chart. Follow this row to column 'E'. The intersection is lowercase 'n'.

Converting Hex to Decimal

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F.
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
2	32	033	34	35	36	37	38	39	40	41	42	43	44	45	46	47
3	48	049	50	51	52	53	54	55	56	57	58	59	60	61	62	63
4	64	065	66	67	68	69	70	71	72	73	74	75	76	77	78	79
5	80	081	82	83	84	85	86	87	88	89	90	91	92	93	94	95
6	96	097	98	99	100	101	102	103	104	105	106	107	108	109	110	111
7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
9	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
A	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
B	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
C	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
D	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
E	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
F	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

To convert a value from hex to decimal:

1. Find the row that starts with the first digit of the hex value. If it's a single digit hex value, use row '0'.
2. Find the column with the second hex digit.
3. The intersection of the row and column gives the decimal value.

e.g. What is the decimal value for 0x3B?
Going across row '3' to column 'B', the decimal value is 59.

To convert a decimal value to hex:

1. Find the decimal value in the chart (not including the top row or left column).
2. The row heading to the left is the first digit of the hex value.
3. The column heading above is the second digit of the hex value.

e.g. What is the hex value for decimal 76?
Decimal value '76' is locate on row '4' and column 'C' so the hex value is 0x4C.

APPENDIX G. COMMAND AND RESPONSE SUMMARY

Message Number	Description	Stand-alone	DUKPT	MSK	MMK	Page
02	Load Multi-Master Key				X	13
04	Check Multi-Master Key				X	14
07	DES Algorithm Reliability Test				X	16
08	Select Multi-Master Key				X	18
20	Load A FIT Table	X				19
21	Delete All FIT Tables	X				23
23	Set/Retrieve Date	X				25
24	Remote Password Entry	X				27
30	PIN Entry Request			X		29
31	PIN Offset Request			X		33
32	PIN Verification Request			X		37
33	Encryption Test Request			X		40
34	CVV Request			X		42
35	PVV Request			X		45
36	PVV Verification Request			X		48
37	Identikey PIN Offset Request	-	-	-		51
38	Verify Identikey Offset	-	-	-		54
40	KeyPad Input Request		X	X		58
41	String Input Request		X	X		61
42	Display Single String Message		X	X		64
43	Display Alternating Messages		X	X		65
44	Firmware Part Number and Version Request		X	X		67
50	Set or Request Soft Switches		X	X		69
51	Replace Default Display		X	X		79
52	Enable Default Display		X	X		83
53	Transaction Counter Request			X		84
54	Transaction Counter Reset			X		86
55	Key Serial Number Request			X		87
56	Key Check Value Request			X		89
57	Load Substitution Table			X		91
58	Activate or Deactivate Offset/Verify			X		93
60	Pre-Authorization: PIN Entry Request		X			95
62	Pre-Authorization: Transaction Amount Authorization Request		X			96
63	Authorization Response		X			97
64	Pre-Authorization: Transaction Amount Authorization/Data Authentication Request		X			98
65	Authorization and MAC Response		X			100
66	Pre-Authorization: PIN Entry Test Request		X			102
70	PIN Entry Request (DUKPT)		X			103
70	PIN Entry Request (MMK)				X	105

Continued next page

Message Number	Description	Stand-alone	DUKPT	MSK	MMK	Page
71	PIN Entry Response (DUKPT)		X	X		107
71	PIN Entry Response (MMK)				X	109
72	Cancel Session Request		X	X		110
74	Pin Entry/ Data Authentication Request		X			111
75	PIN Entry and MAC Response		X			113
76	Pin Entry Test Request		X			115
78	Pin Entry Test/Data Authentication Request		X			116
80	Card Data Entry Request		X	X		117
81	Card Data Response		X	X		119
82	Cancel and Display		X	X		126
83	Card Holder Data And PIN Entry Request		X	X		127
90	Load Initial Key Request		X			131
91	Load Initial Key Response		X			132
92	Reinitialization Request		X			133
93	Reinitialization Response		X			135
94	Load Master Key			X		136
95	Load Session Key			X		138
96	Load Working Key			X		140
97	Load Key Serial Number			X		142
98	Delete Keys			X		144
99	Set/Retrieve DSN	X	X	X	X	146
Q1	Display Swipe Card	X	X	X	X	146
Q2	Indicate Host Done	X	X	X	X	149
Q4	Turn Card Reader On/Off	X	X	X	X	150
Z1	Cancel Session Request		X	X		151
Z2	Display a String		X	X		152
Z3	Display Rotating Messages		X	X	X	153
Z8	Reset/Set Idle Prompt		X	X		154
Z42	Request Noncoded Key	X	X	X	X	155
Z43	Return Noncoded Key	X	X	X	X	156
Z60	Pre-Authorization: PIN Entry Request		X			157
Z62	Accept And Encrypt Pin (With Custom Prompts)				X	158
Z66	Request MAC				X	160
Z67	Return MAC				X	164

INDEX

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

V

W

Z