TELOCATOR ALPHANUMERIC

PROTOCOL

(TAP)

Version 1.8
February 4, 1997

TABLE OF CONTENTS

UPDATE SUMMARY

Questions concerning the TAP protocol, as well as submissions for protocol updates, may be directed to:

> Jay Moskowitz, Chairman - TAP Committee
> Real Time Strategies, Inc.
> 960 S. Broadway, Suite 118c
> Hicksville, NY 11801
> Phone: (516) 939-6655; Fax: (516) 939-6189

Revision History:

Revision 1.8 - February 04, 1997
Updates:
1) Updated section 3.0 step 6 to indicate that the text following a response code is not normally examined by the remote entry device software, with certain exceptions.
2) Appendix A - inserted an explanation prior to the response code table, indicating that certain responses require further examination of the message text.
3) Added error response 214 to accept a message which has been truncated to a specific maximum size which is returned by the paging terminal.
4) Added error response 517 to reject a message which has exceeded a specific maximum size which is returned by the paging terminal.

Revision 1.7 - August 14, 1996
Updates:
1) Section numbers added
2) Note updated at the start of section 3.0
3) Comment added to step 5A in section 3.0 to explicitly denote the log on sequence <ESC>PG1<CR>
4) Section 5.0 - Checksum calculation described in greater detail.
5) Section 6.0 - Checksum program expanded to include the calculation of the arithmetic sum of 7 bit values.
6) Appendix A - Added Response Codes 514 through 516
7) Appendix A - Update the telephone number of the PCIA
8) ASCII table was added as Appendix B
9) Sample session was added as Appendix C

Revision 1.6 - July 27, 1995
Edited By: J. Stephen Holyer
　　　　Paging Network, Inc.

Updates:
1) Message sequences are no longer optional
2) Message Sequence Response Codes defined in protocol
3) Response Codes listed in Appendix A
4) References to TDP added to Introduction
5) Pager ID clarified with explanation of function digit
6) Transparency mechanism for including non-printable characters in a message
7) Additional Implementation Notes:
　　　　a)　Clarified end-of-line marker
　　　　b)　Clarified Check Digit and Function Digit in some implementations
　　　　c)　Recommended that trailing spaces not be sent in a message
　　　　d)　Recommended that extra spaces and other characters not be used for display formatting
　　　　e)　Noted that message sequences were optional prior to this version
　　　　f)　Noted that older implementations do no support "transparency" for non printable characters

g) Noted that the pager interpretation of non-printable characters is pager dependent

Revision 1.5 - July 21,1994
Updates:
 1) Additional Implementation Notes.

Revision 1.4 - May 2,1994
Updates:
 1) Addition of Implementation Notes.

Revision 1.3 - September 24, 1993
Updates:
 1)   Addition of a sample checksum calculation program in BASIC.

Revision 1.2 - August 20, 1992
Updates:
 1)   The section headed Character Sets was removed. This information is now contained in the specification of the format conversion process (TFC) of the TDP suite of protocols.

Revision 1.1 - July 30, 1992
Updates:
 1)   Retype document from September, 1988 copy.
 2)   Define timing and retry values in parametric form.
 3)   Define ASCII to BCD translation for all sixteen possible POCSAG numeric values.

Revision 1.0 - September 1, 1988
Initial release by Telocator (PCIA)

The Telocator Alphanumeric Input Protocol (TAP)

1.0   Introduction

In order to decrease holding times on input lines to alphanumeric systems, it is desirable to promote input devices which will allow off-line entry of paging information and dump this data quickly after connection to the central paging terminal. A recommended protocol is contained in this specification. This protocol was known as the Motorola Page Entry (PET) as well as the IXO alphanumeric entry protocol until it was adopted by Telocator (now known as the Personal Communications Industry Association - PCIA) in September 1988, as an industry standard protocol for the input of paging requests. It is now referred to as the Telocator Alphanumeric Protocol (TAP).

This protocol is compatible with special versions of small input devices available from numerous sources. It is also suitable for automatic input from a remote computer and has been distributed to numerous manufacturers of paperless TAS systems. Note that there are several options within the protocol:

1.   It may be used for paging with 2 fields per transaction or other services with a different number of fields per transaction.

2.   The use of manual input devices is provided in the log on procedure. Such provision is optional with the system operator.

3.   Optional messages to the remote entry device may be added to control responses from the central terminal. For paging, these will probably be used for a message acceptance or rejection summary at the end of a message.

Since this protocol's inception, the capabilities and uses of paging receivers have been dramatically expanded. Today many applications and paging receivers exist that support the transmission of 8 bit data. Data applications are beyond the scope of the TAP protocol. PCIA (formerly Telocator) has adopted a suite of protocols that are specifically designed for accommodating 8 bit transfer through existing paging systems; this protocol suite is called the Telocator Data Protocols (TDP). Two of the five protocols in the TDP suite facilitate sending 8 bit data through paging companies using the 7 bit TAP specification.

Telocator Format Conversion (TFC). Defines a method for converting 8 bit information for transmission through an existing TAP input port.

Telocator Radio Transport (TRT). This defines the method for packetizing the information into logical blocks for transmission and provides for reconstruction at the receiving devices.

The use of these methods for sending 8 bit data is strongly encouraged because they were designed to make very efficient use of the radio transmission protocols.

2.0   TAP Operating Environment

The standard protocol will be ASCII, with X ON, X OFF either direction, using a 10 bit code (1 start, 7 data, even parity, 1 stop).

It is recommended that Paging Terminals be equipped to receive 300 baud full duplex data using a Bell 103 compatible modem. Optionally, certain inputs may be capable of receiving 110 baud Bell 103 full duplex, or 300/1200 baud Bell 212 full duplex, utilize CCITT compatible modems

and/or operate at higher speeds. No echo shall be employed in full duplex mode. Any attempts at automatic baud rate determination shall be within the constraints of the specified protocol.

3.0   Recommended Sequence Of Call Delivery From An Entry Device

The following pages describe the steps to follow in sending a message via the TAP protocol. The first column shows the steps followed by the sending device, the second column describes the steps followed by the paging network control system. The paging control equipment is referred to as a "paging terminal".

These steps are the TAP standard. All paging terminals and all sending software must implement the TAP protocol in the manner described.

Note: All quotation marks and the symbols < > shown in this document are used for notation only and are not transmitted. The characters between the symbols < > are ASCII control codes as shown in Appendix B.

Remote Entry Device Does	Paging Terminal Does	COMMENTS

1)   Off hook-Access DDD Line.
      Await dial tone.
      Dial Stored Access number		

	Ring Answer	
2) Carrier up		
	Carrier up	
3) "<CR>"		<CR> is repeated at intervals of t1 seconds until paging terminal responds with "ID=" at the correct baud rate or until n1 transmissions have been completed. (This step exists to allow for possible baud rate recognition).
		Some systems have chosen to send "ID=" from the paging terminal if they do not receive <CR> in about t1 seconds.
4)	"ID ="	Request for ID returned within t2 seconds of receipt of <CR>.
		The paging terminal shall wait up to t5 seconds for a response to "ID=". Paging terminal may resend "ID=" up to n3 times, if a proper response is not received.

Remote Entry Device Does	Paging Terminal Does	COMMENTS

5A) (For automatic remote entry devices only)
 	
   "<ESC>SST
		"<ESC>" signifies entry device intends to use automatic mode. "SS" is a set of two alphanumeric characters signifying a type of service to be accessed.
		
		For a paging service where:
		     Field 1 = "Pager ID", and
		     Field 2 = "Message" (if any)
		(see step 8) SS will be sent as "PG".
		
		"T" is a single alphanumeric character relating to the type of terminal or device attempting to send the message.
		
		T = 1 is a category of entry devices using the same protocol. At the present time, all entry devices and computer programs utilize T=1. The values T = 7, 8, 9 are reserved for devices which may relate to a specific user's system.

PPPPPP<CR>"		6 alphanumeric character password (PPPPPP). Password is optional and is, in general, reserved for future services. Password may be interpreted as either a caller ID or a system entry key. Length of password, when used, may be different in some systems. When an incorrect logon sequence beginning with <ESC> is received, the paging terminal may respond with an "ID=" if it requires a retransmission. To send a message to a paging service the remote entry device would normally send the sequence <ESC>PG1<CR> as step 5A.

Remote Entry Device Does	Paging Terminal Does	COMMENTS
5M) (For manual remote entry only)		Lack of <ESC> at beginning of response to "ID=" signifies manual operation, when supported.
"M<CR>"		Any manual operation after logon is user defined.
		Echo transmission is allowed after manual conversation is established. "M<CR>" can be replaced by any non-null sequence ending in <CR> and not beginning with <ESC>
6)	"<Message sequence> <CR><ACK><CR>"	Logon accepted
	or	
	"<Message sequence> <CR><NAK><CR>"	Requested again
	or	
	"<Message sequence> <CR><ESC><EOT><CR>"	Forced disconnect
		This response shall arrive within t3 seconds of step 5.
		A message sequence is defined as a series of short messages separated by <CR>'s. A <CR> always follows a message sequence.
		Each text message in a message sequence will have a response code prepended to the message text with a space separating the response code from the message text. The response code is a three digit number which will indicate the general meaning of the response message and is intended for use by the message entry software. The three digits will contain enough encoded information for the Message Entry Device software to determine the meaning of the error message

Remote Entry Device Does	Paging Terminal Does	COMMENTS
		and take appropriate action. Any text following the response code will not normally need to be examined by the software. The text is intended to provide additional information to a user and may be displayed or logged to an information file. While the text of the message sequence varies from implementation to implementation, it must always be consistent with the given response code. It is highly recommended that the message text be made available to the user in all implementations to aid in troubleshooting a failing session.
		The Response code's are further defined in Appendix A, which also notes when the remote entry device should process data returned in the text.
		Paging terminals will send a message as part of the first message sequence formatted to indicate the protocol revision number (response code 110). This message indicates to the remote entry device the level of features supported by the paging terminal.
6a)	Optional Message sequence"<CR>	The paging terminal may insert a message sequence between steps 6 & 7.
7)	"<ESC>[p<CR>"	Message go ahead is sent when the paging terminal is prepared to receive the first transaction. NOTE: p is lower case. This response shall be returned within t3 seconds after step 6.

Remote Entry Device Does	Paging Terminal Does	COMMENTS
8) Transaction #1 Block #1 __ __ |"\<STX> |Field #1\<CR> |Field #2\<CR> | |Field #N\<CR> |\<ETX>\<CHKSUM>\<CR>"		A transaction should be sent by the entry device within t4 seconds of a response from the paging terminal.

A "block" is up to 256 characters in length, with up to 250 characters of info, plus 3 control characters and a 3 character checksum. The block carries one transaction (one set of all fields 1 through N) or a portion of one transaction. A block may be less than 256 characters to accommodate short transactions. Trailing spaces should be eliminated from messages in order to conserve "over-the-air" transmission time when the page request is transmitted. Any character with a value less than or equal to DEL (Hex 7F) may be included in the transaction, unless the character would cause a protocol conflict.

Earlier version of TAP did not allow ASCII Control-Code characters (characters whose value is less then Hex 20) to be sent to a Pager. This is now supported by a control transparency mechanism.

If a control character which would cause a protocol conflict must be transmitted within the packet the originator must perform "transparency insertion" to make the control character transparent to the protocol. This is done by converting the control byte to two bytes consisting of a SUB (Hex 1A) character followed by the printable ASCII character formed by adding Hex 40 to the ASCII value of the control character to be sent.

Any control character may be made transparent at the implementor's discretion; the following control characters must be made transparent if they are to be transmitted in a packet in order to prevent conflicts with the TAP protocol:

Remote Entry Device Does	Paging Terminal Does	COMMENTS

8) continued ...

Character	Hex
CR	0D
LF	0A
ESC	1B
STX	02
ETX	03
US	1F
ETB	17
EOT	04
SUB	1A

Transaction #2
  Block #2

```
__
|"<STX>
|Field #1<CR>
|
|
|
|Field #J<CR>
|<ETB><CHKSUM><CR>"
|__
```

  Block #3

```
__
|"<STX>
|Field #J + 1<CR>
|
|
|
|Field #L
|<US><CHKSUM><CR>"
|__
```

  Block #4

```
__
|"<STX>
|Field #L (CONT.) <CR>
|
|
|<CR>
|<ETX><CHKSUM><CR>"
|__
```

A field may be any length and where necessary may be continued in succeeding blocks. A field always ends with a <CR>. A block always begins with an <STX> and ends with a checksum followed by a <CR>. The characters preceding the checksum depend on what, if anything, is continued beyond the block boundary.

The <ETX> is used as a block termination indicator if a given transaction (Fields 1 through N) ends within the block currently being transmitted

The <ETB> is used as a block terminator if the transaction is continued into the next block, but the last field in the current block is complete.

If the last field within the current block is to be continued in the next block, no <CR> is inserted at the end of the first portion of the field and the <US> character is used as a block termination character. The <CR> terminating the broken field is sent at the end of the field in whatever block the field actually terminates.

No limit is established within the protocol itself regarding the number of transactions, the number of fields or the number of blocks per field; however, a particular user system may have limits on any of these items.

Remote Entry Device Does	Paging Terminal Does	COMMENTS

8) continued ...
   Last Transaction
   Last Block
 	
 ‾‾
 |"<STX>
 |Field   <CR>
 |
 |
 |Field #N <CR>
 |<ETX><CHKSUM><CR>"
 |__
		Some systems may be limited to one block per transaction and one transaction per phone connection.
		
		 Each checksum is computed by performing the simple arithmetic sum of the 7-bit values of all characters preceding it in that block. (This means that STX and ETB/ETX are included in the sum). The checksum is then the least significant 12 bits of this resulting sum.
		
		 The checksum is transmitted as 3 printable ASCII characters having values from Hex 30 to Hex 3F (the characters 0123456789:;<=>?). The most significant 4 bits of the sum are encoded in the 4 LSB of the first character and the least significant 4 bits of the sum are encoded as the 4 LSB of the third character. (See example attached).
		
		 Typically, a paging system transaction will have 2 fields only:
		       Field 1 = Pager ID
		       (May optionally include a trailing function code).
		       Field 2 = Message.
		
		 Field 1 or Field 2 may be empty. For example, when a page is Tone Only, Field 2 will be empty. Even when empty, a field is followed by a <CR>. Note that some systems will reject transactions which have an empty Field 2 for a display page or transactions which have an empty Field 1. Other systems are less restrictive.

Remote Entry Device Does	Paging Terminal Does	COMMENTS
8) continued ...		Valid Pager ID's are determined by the paging service. While the Pager ID has traditionally been a 7 numeric digit PIN, many systems use 4 numeric digits and some systems use 10 or more numeric digit ID's. Some systems allow alphanumeric "handles" to be entered for the Pager ID. There is no restriction placed on the Pager ID by the protocol; it may be of any length and contain any character that is valid in the protocol character set.
		On systems that allow a one digit function identifier to be appended to the Pager ID the function digit specified determines the pager's feature set that will control the presentation of the page. For example, the function digit may control the pager's beep pattern. For systems that allow the use of a function digit, the sender and the receiver must implicitly agree on the meaning of the function digit.
		The support of the function code is an optional capability of the paging terminal. If this capability is not being supported, this additional digit should not appear in field 1.
		The response to each block, is sent within t3 seconds. The possible responses are:
	"<Message sequence> <CR><ACK><CR>" or	OK, send next block.
	"<Message sequence> <CR><NAK><CR>" or	Checksum or transmission error, send last block again.
	"<Message sequence> <CR><RS><CR>" or	Abandon current transaction and go to next. RS may occur when the checksum is OK, but the current transaction violates a system rule. At the option of the system, it may occur in other cases.

Remote Entry Device Does	Paging Terminal Does	COMMENTS
8) continued ...		The <RS> response typically follows an attempt to page an invalid pager ID.
	"<Message sequence> <CR><ESC><EOT><CR>"	Begin disconnect. The paging terminal is abandoning the current call. The message sequence will indicate the reason.
		Prior to version 1.6 a response could be preceded by an optional message sequence. As of revision 1.6 a message sequence containing a standard Message Sequence Response Code (Appendix A), is required.
		The next transaction must be initiated by the remote entry device within t4 seconds of the paging terminal's last response. If no response is received from the paging terminal within t3 seconds, the transaction may be resent. The remote entry device may resend up to n2 times before considering the connection as failed. The disconnect sequence may then be executed.
9) "<EOT><CR>" .		After reception of an <ACK> or <RS> for the last transaction, the entry device sends the protocol disconnect sequence, <EOT><CR>, meaning there are no more transactions remaining.

Remote Entry Device Does	Paging Terminal Does	COMMENTS
10a)	"<Message sequence><CR>"	An optional message sequence may be sent at this point to indicate degree of acceptability of information in all transactions received during the current interchange. Although optional, this message is highly desirable.
10b)	"<RS><CR>"	An <RS><CR> may be sent at this point if the paging terminal finds any data <ACK>'d in step 8 by the system to be unacceptable because of content (e.g. an invalid pager number or a message field inappropriate for the type of page, etc.). NOTE: It is most desirable to catch all types of errors in step 8, but, some paging control equipment may not be able to catch content errors as they happen.
10c)	"<ESC><EOT><CR>" followed by dropping of carrier and hanging up.	Paging terminal disconnect sequence.

11) Drops carrier and hangs up.

4.0   Implementation Notes

There are thousands of systems worldwide which are capable of accepting alphanumeric messages in the TAP format. Unfortunately, many of these systems are not strictly adhering to all aspects of the protocol. This section contains information to implementors of the protocol at the remote entry device, regarding some of the anomalies which may be found in sending TAP formatted messages into a system. Any new paging terminal implementations should strictly adhere to the protocol and not the anomalies described here.

- The <CR> character (Hex 0D) is used as an end-of-line marker in TAP. Some paging terminals have used other "standard" end-of-line markers such as an <LF> (Hex 0A) or a combination of <LF> and <CR>. If the paging terminal sends these characters (<CR>, <LF>) or character combinations (<CR><LF>, <LF><CR>) when the Remote Entry Device is expecting to receive a <CR>, then the Remote Entry Device should interpret these characters as the end-of-line. NOTE: This means that <LF> characters immediately preceding a <CR> or immediately following a <CR> may be ignored.

- Some systems send "ID=" followed by an end-of-line marker while others send "ID=" alone.

- The "ID=" logon string is preceded by other text or end-of-line markers on some systems.

- The "message continued in the next packet" end of packet indicator ETB (Hex 17) has been implemented as a US (Hex 1F) character in some systems.

- Many paging terminals allow a <CR> character to be imbedded in the message text portion of a paging system transaction. In this case the message itself is spread across multiple fields (fields #2, fields #3, etc.) of the transaction block.

- Although the TAP transaction block may contain 250 characters of information, the entire block need not be filled if there are fewer characters to be sent to the pager. Trailing spaces should be eliminated from messages in order to conserve "over-the-air" transmission time when the page request is transmitted.

- Most pagers allow display formatting characters to be sent to the pager for properly formatting the display. It is recommended that these formatting characters be used (as specified by the pager manufacturer) if display formatting is desired. Extra spaces, dashes(-), underlines(_), and periods(.) should not be used in a message for the purposes of "formatting" the display of a message in order to conserve "over-the-air" transmission time when the page request is transmitted.

- There is a message size limitation placed on input in many systems. Some systems may only accept 80 characters, 200 characters, 1000 characters, or another message size limit specified by the particular system operator.

- Some systems do not support multi-block messages. In some cases the entire message must fit in one block.

- Some systems place a limit on the number of messages which may be sent on a single connection.

- Support for non-printable ASCII control characters was added in Version 1.6 of the TAP specification. Many older implementations do not support the non-printable ASCII characters nor do they support control transparency insertion. Some older systems will allow the entry of control-characters if they do not cause a protocol conflict.

- The interpretation of, and reaction to, non-printable ASCII control-characters sent to a paging receiver is specific to the model of the paging receiver in use.

- The response code numbers prepended to the Optional Message sequences were added in Version 1.6 of the TAP specification. Older implementations do not prepend response codes to the message sequence.

- Prior to Version 1.6 of the TAP specification message sequences were optional. Older implementations may not send message sequences. Many implementations send <CR><Control-Code><CR> with no message text if a message sequence is not included in the response while some systems send the sequence "<Control-Code><CR>" without the preceding <CR>.

- Some systems are not following the even parity specification and are using no parity [sending 1 start bit, 8 data bits (normally with the 8th bit always 0) and 1 stop bit]. A sending device should have the option of operating in a no parity environment.

- The Pager ID which is entered through normal (non-computer) telephone dial access to a paging terminal (often through touch tone input), is normally the same number as that specified in the Pager ID field of TAP input. In some systems, the Pager ID entered from the telephone includes a trailing digit which is known as the Check Digit. It is utilized to detect common touch-tone keypad keystroke errors during entry of a subscriber's Pager ID. This Check Digit can be thought of as part of the Pager ID, and is usually included as part of the Pager ID field for TAP input. There are systems which require that the trailing Check Digit be removed when specifying the pager ID for TAP input. Please note, that if an optional function code is also used in Field 1 the Check Digit precedes the Function Code since the Check Digit is considered to be a part of the Pager ID.

- An optional function code value is described in step 8. This function code may be appended to the Pager ID in Field 1 of the TAP transaction. As noted above, a Check Digit can also be considered a part of the Pager ID. Public access paging systems in North America do not typically support these optional modifiers. Many private paging systems do support these additional modifiers as well as many public access systems outside of North America.

Since these anomalies can not be determined without sending messages into a system, they should be considered when implementing the protocol. It may be necessary to develop a "profile" of configuration parameters to set when calling into particular systems.

5.0   Checksum Calculation

Each checksum is computed by performing the simple arithmetic sum of the 7-bit values of all characters preceding it in that block. (This means that STX and ETB/ETX are included in the sum). The checksum is then derived from the least significant 12 bits of this resulting sum.

The checksum is transmitted as 3 printable ASCII characters having values from Hex 30 to Hex 3F (the characters 0123456789:;<=>?). The most significant 4 bits of the 12 bit sum are encoded into the 4 LSBs of the first character (HEX 30 [decimal 48] plus the 4 bit value becomes the first ASCII character). The middle 4 bits of the 12 bit checksum are encoded into the 4 LSBs of the second ASCII character and the least significant 4 bits of the 12 bit sum are encoded as the 4 LSBs of the third character.

The following is an example of how the checksum is derived for the message <STX>123<CR>ABC<CR><ETX>. The following section contains a sample "BASIC" program for calculating the three character checksum value.

<div align="center">CHECKSUM EXAMPLE</div>

Character	7 bit ASCII representation		Decimal
STX	000	0010	2
1	011	0001	49
2	011	0010	50
3	011	0011	51
CR	000	1101	13
A	100	0001	65
B	100	0010	66
C	100	0011	67
CR	000	1101	13
ETX	000	0011	3
12 bit Sum =	0001	0111	1011	379
	(Hex 1)	(Hex 7)	(Hex B)	
	[decimal 1]	[decimal 7]	[decimal 11]	

Encoded bits=	+Hex30	+Hex30	+Hex30
	(decimal 48)	(decimal 48)	(decimal 48)
	Hex 31	Hex 37	Hex3B
	(decimal 49)	(decimal 55)	decimal 59)
	|	|	|
	|	|	|
ASCII	1	7	;

<div align="center">THREE CHARACTER CHECKSUM = 17;</div>

Therefore, an example of a complete block containing a correct checksum is:

"<STX>123<CR>ABC<CR><ETX>17;<CR>"

6.0   Checksum Calculation Program

6.1   Step 1 - Calculation of arithmetic sum of 7 bit values

```
REM -      This sample BASIC program processes the ASCII
REM -      characters of the checksum example of
REM -      the prior section (defined as decimal values
REM -      in the DATA statement), and derives the
REM -      arithmetic sum of 7 bit values.  The INT
REM -      function returns the integer portion of a number.
REM -      As shown in the example of the prior section, this
REM -      example should result in a value of 379.
REM -      <STX> 1 2 3 <CR> A B C <CR> <ETX>
DATA       2, 49, 50, 51, 13, 65, 66, 67, 13, 3, 0
sum = 0
10 READ i
IF i = 0 THEN 20
i = i - (INT(i/128) * 128)
SUM = SUM +i
GOTO  10
20 PRINT "The arithmetic sum of 7 bit values is "; sum
```

6.2   Step 2 - Arithmetic sum to 3 printable ASCII characters

```
REM -      This sample BASIC program converts the checksum value "sum" into the
REM -      three characters which are sent as part of the TAP protocol.  The variables
REM -      d1, d2 and d3 contain the three digits which are to be added to the
REM -      transmitted data block.  "INT" is the integer function which returns the
REM -      integer portion of a number.  This function is required if the variables
REM -      are floating point numbers.  If they are declared as integers then the INT
REM -      function is not required.  This BASIC program may easily be converted to
REM -      other programming languages.
REM -
sum = 379
REM -
REM -      Following the checksum example in the TAP Specification Document:
REM -      <STX> 1 2 3 <CR> A B C <CR> <ETX> the checksum value is 379.
REM -      The following code will create the three characters to be transmitted
REM -      in order to represent this checksum.
REM -
d3 = 48 + sum - INT(sum / 16) * 16
sum = INT(sum / 16)
d2 = 48 + sum - INT(sum / 16) * 16
sum = INT(sum / 16)
d1 = 48 + sum - INT(sum / 16) * 16
REM -
REM -      Print the three character checksum in decimal and ASCII
REM -
PRINT     "d1="; d1, "d2="; d2, "d3="; d3
PRINT     "d1$="; CHR$(d1), "d2$="; CHR$(d2), "d3$="; CHR$(d3)
```

7.0   Timing and Retry Parameters

The initial release of the TAP specification defined fixed values for various time-outs and retry parameters. These values have been specified as parameters as of revision 1.1 of the specification. The default values of these parameters are those specified in revision 1.0 of the specification. It is recommended that implementations of TAP allow for the on-line modification of the various parameters to adjust the operation of the protocol for systems which have not strictly adhered to the specification.

Timing parameters
    t1 - 2 secs.
    t2 - 1 sec.
    t3 - 10 secs.
    t4 - 4 secs.
    t5 - 8 secs.

Retry Parameters
    n1 - 3
    n2 - 3 (undefined in rev. 1.0)
    n3 - 3 (undefined in rev. 1.0)

Appendix A
Message Sequence Response Codes

The first digit of a paging terminal response code number can be interpreted as follows:

    1yz   Informational Text
            Messages sent as part of the logon process or the disconnect process
    2yz   Positive Completion
            Messages regarding an operation that was performed successfully. Examples are a logon was successful or a page was accepted for delivery.
    3yz   Unused
    4yz   Unused
    5yz   Negative Completion
            Messages regarding an operation that was not performed successfully.
    6yz-
    9yz   Unused

The last two digits will further identify the response code.

The definitions in the following table, describe the meaning of the numeric response code. The response codes may optionally be followed by a text string or strings which may vary between paging terminals. The general format of the text is described in the Note which follows the table. Certain text strings are required to follow a specific format as described in this section.

Defined response codes are:

Response Code	Definition
110	Paging Terminal TAP Specification Supported * (See Note)
111	Paging terminal is processing the previous input -- please wait
112	Maximum pages entered for session
113	Maximum time reached for session
114	Welcome banners (These messages are sent only at the beginning of a session to present "service" information)
115	Exit Messages (Service related messages sent before the Paging Terminal terminates)
211	Page(s) Sent Successfully
212	Long message truncated and sent
213	Message accepted - held for deferred delivery
214	### character maximum, message has been truncated and sent. (If this response is returned in lieu of response 212, it must start with an ASCII digit string as described in the Note which follows).
501	A "time-out" occurred waiting for user input
502	Unexpected characters received before the start of a transaction. A character sequence other then <STX> or <EOT><CR> was received while the paging terminal was waiting to receive a Transaction Block from the Remote Entry Device.
503	Excessive attempts to send/re-send a transaction with checksum errors.

504	The message field of the TAP transaction contained characters, but message characters are not allowed for the Pager format. Perhaps the paging receiver for the given PIN is a 'Tone Only' pager.
505	Message portion of the TAP transaction contained alphabetic characters, but alphabetic characters are not allowed for the Pager format. Perhaps the paging receiver for the given PIN is a 'numeric' pager.
506	Excessive invalid pages received
507	Invalid Logon attempt: Incorrectly formed logon sequence
508	Invalid Logon attempt: Service type and category given is not supported
509	Invalid Logon attempt: Invalid password supplied
510	Illegal Pager ID - The pager ID contains illegal characters or is too long or short
511	Invalid Pager ID - There is no subscriber to match this ID
512	Temporarily cannot deliver to Pager ID - Try Later
513	Long message rejected for exceeding maximum character length
514	Checksum error
515	Message format error
516	Message quota temporarily exceeded
517	### character maximum, message rejected. (If this response is returned in lieu of response 513, it must start with an ASCII digit string as described in the Note which follows).

NOTE: The "special" response code 110 will always contain the text indicating the version of the TAP specification supported with no other message text.   The format of this message will be:

"110 M.m<CR>"

where M.m indicates the current major and minor version number.   For Version 1.6 of the TAP specification, the optional message sequence "110 1.6<CR>" would represent the version.   It is recommended that the paging terminal software always send this optional message to the Remote Entry Device as part of the messages sent at logon. This will allow the Remote Entry software to determine what version of the TAP protocol is implemented by the paging terminal.

Response codes 110, 214 and 517 contain text that is intended for interpretation by the remote entry device software. No other message text should be sent with message number 110 other than version information described.   Messages 214 and 517 return an ASCII string containing a decimal number. This string of digits is terminated by any non ASCII digit such as space, carriage return <CR>, an alphabetic character or punctuation character such as comma.

Message sequences can be made up of several short lines of text separated by carriage returns, and multiple message sequences can be sent to the message entry device *provided* a <CR> character separates two messages. Only the first line of each message given in a sequence of optional messages will contain the response code. If more than one message is contained in one sequence of messages, then each message will contain a response code. Any line that begins with a number that is not a response code must be indented one space.

Example:

118 This is the first line of message 118 <cr>
This is the second line of message 118 <cr>
  980 this is the third line of message 118 and it begins with a number 980<cr>
119 This is the first line of message 119 <cr>
This is the second line of message 119<cr>
<ack><cr>

The Personal Communications Industry Association (PCIA) maintains the list of response code numbers. The list of response codes is intended to be comprehensive for all messages that paging terminals will send. Implementors of paging terminal software should contact PCIA at (703) 739-0300 or the TAP Committee Chairman, to request that additional response codes be assigned, if their implementation contains messages for which there is no currently defined response code number. New response codes will be integrated into future updates to the TAP specification.

Remote Entry Device Implementors should contact PCIA or its Internet Web site (http://www.pcia.com) for the most current list of response code numbers.

Appendix B
ASCII Code Table

MS	0	1	2	3	4	5	6	7
LS CHAR CHAR	000	001	010	011	100	101	110	111
0 0000	NUL	DLE	SP Space	0	@ 'At' Sign	P	` Single Left Quote	p
1 0001	SOH	DC1	! Exclamation Point	1	A	Q	a	q
2 0010	STX	DC2	" Double Quote	2	B	R	b	r
3 0011	ETX	DC3	# Pound or Number Sign	3	C	S	c	s
4 0100	EOT	DC4	$ Dollar Sign	4	D	T	d	t
5 0101	ENQ	NAK	% Percent Sign	5	E	U	e	u
6 0110	ACK	SYN	& Ampersand	6	F	V	f	v
7 0111	BEL	ETB	' Apostrophe	7	G	W	g	w
8 1000	BS	CAN	(Open Parenthesis	8	H	X	h	x
9 1001	HT	EM	) Close Parenthesis	9	I	Y	i	y
A 1010	LF	SUB	* Asterisk	: Colon	J	Z	j	z
B 1011	VT	ESC	+ Plus Sign	; Semi-Colon	K	[Open Bracket	k	{ Left Curly Brace
C 1100	FF	FS	, Comma	< Less Than	L	\ Back Slash	l	| Vertical Bar
D 1101	CR	GS	- Minus Sign	= Equal Sign	M	] Close Bracket	m	} Right Curly Brace
E 1110	SO	RS	. Period	> Greater Than	N	^ Carrot	n	~ Tilde Accent
F 1111	SI	US	/ Forward Slash	? Question Mark	O	_ Under Score	o	DEL

Appendix C
Sample Session (Automatic Mode)

The following represents a typical call flow in sending one page request into a paging network.

	Remote Entry Device	**Paging Terminal**	
1)	Dials paging terminal		
		Modem Answers	
2)	Modem Connects		
3)	<CR>		
4)		ID=	
5a)	<ESC>PG1<CR>		
6)		110 1.7<CR>	*
		Thank you for calling the PCIA<CR>	*
		<ACK><CR>	
7)		<ESC>[p<CR>	
8)	<STX>123<CR>ABC<CR><ETX>17;<CR>		
		211 Page accepted <CR>	*
		<ACK><CR>	
9)	<EOT><CR>		
10a)		115 Thank you for calling <CR>	*
10b)		<ESC><EOT><CR>	
11)		Drops Carrier	
	Drops Carrier		

Note: The numeric response codes shown (110, 211 and 115) are returned only from paging terminals which are operating at TAP revision 1.6 or higher.

* Prior to version 1.6 all returned message sequences are optional and numeric codes were not defined as part of the specification.