AP4118 Rail Tag Programmer User Guide

TransCore
8600 Jefferson Street NE
Albuquerque, New Mexico 87113

January 2017

P/N 16-4118-001 Rev A

For further information, contact:

TransCore
8600 Jefferson Street NE
Albuquerque, NM 87113 USA

Technical Support

Phone: (505) 856-8007
Web: transcore.com/rfidsupport

WARNING TO USERS IN THE UNITED STATES

FCC RADIO FREQUENCY INTERFERENCE STATEMENT
47 CFR §15.105(a)

NOTE: This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency (RF) energy and may cause harmful interference to radio communications if not installed and used in accordance with the instruction manual. Operating this equipment in a residential area is likely to cause harmful interference, in which case, depending on the regulations in effect, the users may be required to correct the interference at their own expense.

NO UNAUTHORIZED MODIFICATIONS
47 CFR §15.21

CAUTION: This equipment may not be modified, altered, or changed in any way without permission from TransCore, LP. Unauthorized modification may void the equipment authorization from the FCC and will void the TransCore warranty.

USE OF SHIELDED CABLES IS REQUIRED
47 CFR §15.27(a)

NOTE: Shielded cables must be used with this equipment to comply with FCC regulations.

TransCore, LP
USA

1 Before You Begin

2 Programmer Basics

3 Configuring the AP4118 Rail Tag Programmer

4 Operating the Programmer

A Tag Programmer Specifications

B RS–232 Connector Pin-outs

C TransCore 6-Bit ASCII Format

D 4-Bit ASCII Hex Codes

E Programmer Response Codes

F Security Characters

List of Figures

List of Tables

Chapter 1

Before You Begin

The AP4118 Rail Tag Programmer User Guide provides information necessary for installing, operating, and troubleshooting the AP4118 Rail Tag Programmer.

Purpose of This Guide

This *AP4118 Rail Tag Programmer User Guide* discusses the proper setup and operation of the AP4118 Rail Tag Programmer. This guide also provides procedures useful in troubleshooting any problems encountered after installation.

Intended Audience

The *AP4118 Rail Tag Programmer User Guide* is designed to be used by technical personnel responsible for setting up and operating the AP4118 Rail Tag Programmer and related software.

Related Documents

AP4118 Rail Tag Programming Host Software User Guide

Guide Topics

This document presents the following information:

Chapter 1 – Before You Begin	Describes the purpose, intended audience, guide topics, related documentation, and document conventions
Chapter 2 – Setting Up the Programmer	Describes how to set up the tag programmer, prevent electrostatic discharge damage to tags, and connect the programmer to a host computer system
Chapter 3 – Permissions	Defines permissions and instructs how to configure the AP4118 Rail Tag Programmer using a PT5780 or PT5785 Permissions Tag
Chapter 4 – Operating the Programmer	Describes how to program a tag and how to verify or read the data. This chapter also lists the RS–232 command set and describes the security character operation
Chapter 5 – Troubleshooting Procedures	Lists problems that may be encountered during tag programmer operation and suggested solutions

Appendix A – Programmer Specifications	Provides the programmer’s physical and environmental specifications
Appendix B – RS–232 Connector Pin-outs	Describes the cable interface configuration to a host computer
Appendix C – TransCore 6-Bit ASCII Format	Lists TransCore’s 6-bit ASCII character codes
Appendix D – 4-Bit ASCII Hex Codes	Lists TransCore’s 4-bit hexadecimal formats
Appendix E – Programmer Response Codes	Lists the tag programmer response codes
Appendix F – Security Characters	Lists the 6-bit security character codes

Typographical Conventions

Table 1 lists the conventions used in this manual.

Table 1 Typographical Conventions

Convention	Indication
WARNING	This procedure might cause harm to the equipment and/or the user.
CAUTION	Concerns about a procedure
`Code`	Code, including keywords and variables within text and as separate paragraphs, and user-defined program elements within text appear in courier typeface.
Dialog Box Title	Title of a dialog box as it appears on screen
Function	Starts with the characters, G4, which are in mixed case with no underscores, and include parentheses after the name, as in G4FunctionName().
Menu Item	Appears on a menu. Capitalization follows the interface.
Note	Auxiliary information that further clarifies the current discussion. These important points require the user’s attention. The paragraph is in italics and the word Note is in boldface.
NUL	Zero-value ASCII character or a zero-value byte

Table 1 Typographical Conventions (continued)

Convention	Indication
NULL	Zero-value pointers. Null-terminated string refers to strings of printable ASCII characters with a zero-value byte placed in memory directly after the last printable character of the string.

Chapter 2

Programmer Basics

This chapter lists the factory-supplied and optional equipment for the AP4118 programmer, and describes how to prevent electrostatic discharge (ESD) damage to tags, connect to a host computer, and install optional TransCore tag programming software to the host computer.

Factory Supplied Equipment

The programmer is shipped with the items listed here. When unpacking your order, make sure all of the items are included:

- AP4118 Rail Tag Programmer
- Programmer Accessory Kit (P/N 19094-01) includes:
    - Power transformer and cable to convert AC voltage to 12V DC
    - Power cord
    - Programmer-to-PC RS–232 connecting cable
    - Static-suppression wrist strap with cord
    - TransCore USB flash drive with:
      – AP4118 Tag Programmer Host Software
      – AP4118 Programmer User Guide

If any of the above items are missing from your order, contact your TransCore distributor.
If you did not purchase TransCore P/N 14-4118-002 and you want to program the AT5549 End-of-Train Tag, you will also need an AT5549 End-of-Train Tag Programmer Adapter (TransCore P/N 20-4118-001), which is required to seat the AT5549 tag during programming.

To use the AP4118 programmer you need either a host system, such as a personal computer (PC), or a communications terminal to operate the programmer. TransCore does not supply this equipment for the AP4118 Rail Tag Programmer.

Preventing Electrostatic Discharge Damage

Caution
TransCore tags are sensitive to ESD, and precautions are necessary to ensure proper tag programming operation. Use the grounded ESD static-suppression wrist strap when using the programmer. Additional protection from ESD is recommended where practical.

Static is generated by friction. Some causes of static include the following:

- Shoes moving across a carpeted or plastic floor
- Hot air blowing into a room from a hot-air duct
- Rubbing tags together
- Rubbing tags across a table top
- Wearing certain types of clothing

ESD may cause significant damage to the tag and will adversely affect a tag's operating performance. Typical symptoms of ESD include the following:

- Inability to program the tag
- Greatly reduced operating range
- Tag operating failure

Antistatic Workstation

In a factory or fixed indoor environment, position the programmer in an antistatic workstation. Well-designed workstations use a system of multiple precautions against ESD. Figure 1 shows an antistatic workstation.

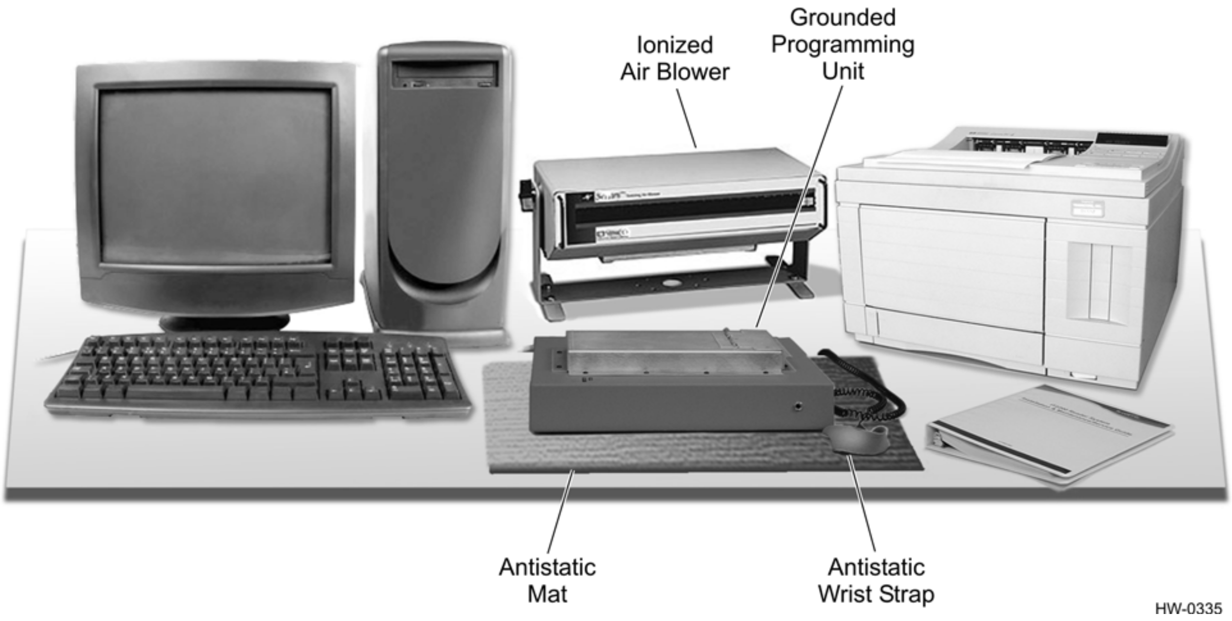


Figure 1 Programming Workstation with ESD Controls

Contact TransCore Technical Support if you have any questions concerning the suitability of your programming station or environment.

You can take several precautions to prevent ESD from affecting programming operations. The most effective precautions are listed below. The potential for ESD damage decreases when two or more precautions are used together.

Using the ESD Wrist Strap

Caution

You should always attach the ESD wrist strap before beginning programming operations in any setting.

1. Be sure the programmer is connected to an earth ground.
2. Connect the banana plug of the ESD wrist strap to the ESD jack on the front of the unit.
3. Attach the ESD strap to your wrist by adjusting the hook and loop material.

Antistatic Mat

The antistatic work mat on which your programmer sits should be connected to earth ground through the standard 1 M-Ohm resistance that should be provided with the mat. The mat also should be clean and free of any electrical path to direct earth ground.

Ionized Air Blower

You may choose to set up an ionized air blower at your workstation to provide additional ESD protection. Ionized air blowers neutralize tags from static build-up before programming.

ESD-Safe Tags Container

Keep tags in the original box on an antistatic mat or within an ESD-safe container until they are programmed. Grounded metallic containers are best for inhibiting ESD.

Connecting the Host Device

The programmer requires connection to a host device — a host computer or communications terminal — for normal operation. Figure 2 shows the connector from the programmer to the host.

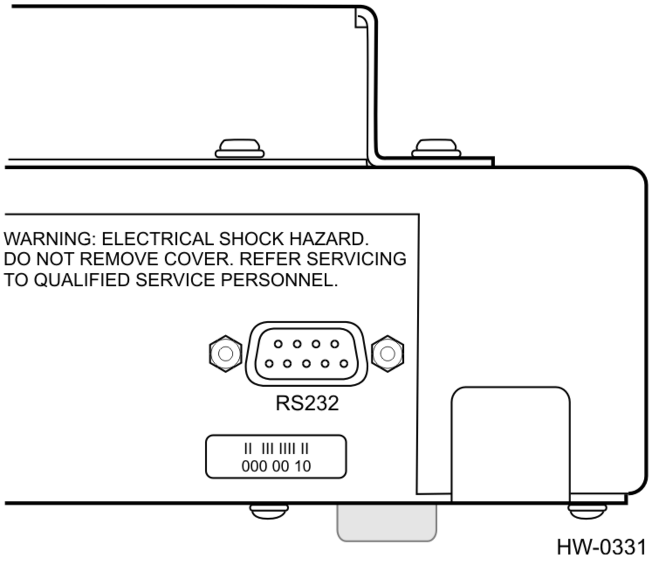

WARNING: ELECTRICAL SHOCK HAZARD.
DO NOT REMOVE COVER. REFER SERVICING
TO QUALIFIED SERVICE PERSONNEL.
RS232
000 00 10
HW-0331

Figure 2 Serial Interface Connector

If the host system is a PC or a communications terminal, then you do not need to use a null modem.

The tag programmer's PC interface complies with the RS–232 standard for data terminal equipment and uses the protocol settings listed in Table 2. Refer to Appendix B for specifications of the programmer's RS–232 pin-outs.

Table 2 PC Interface Protocol Settings

Protocol	Setting
Data Rate	9600 bps
Data Bits	8
Parity	None
Stop Bits	1
Software Flow Control	None
Hardware Flow Control	None

To connect the programmer to the host computer

1. Connect the RS–232 cable from the unit to the host computer system.

2. Switch on power to the programmer. You should see the POWER then READY LEDs illuminate. When the READY LED illuminates, you can begin using the tag programmer. The tag programmer has factory default settings installed to allow tag reads only.

   To configure the tag programmer for your tag requirements, you will need to use either a PT5780 or PT5785 Permissions Tag[1] to program specific operational and security information into the programmer. Chapter 3, “Configuring the AP4118 Rail Tag Programmer,” instructs the user in configuring the tag programmer using this tag.

   If the READY LED does not illuminate or if the host system cannot communicate with the tag programmer, refer to Chapter 5, “Troubleshooting.”

AP4118 Host Software

The AP4118 Tag Programmer Host Software can be found on the provided TransCore USB Drive. The host software is not necessary to use the AP4118 Rail Tag Programmer. Many customers prefer to use their specific host software to communicate with the tag programmer.

To Install the Host Software

1. Switch on the host computer.

2. Insert the TransCore USB flash drive into a USB slot on the host computer.

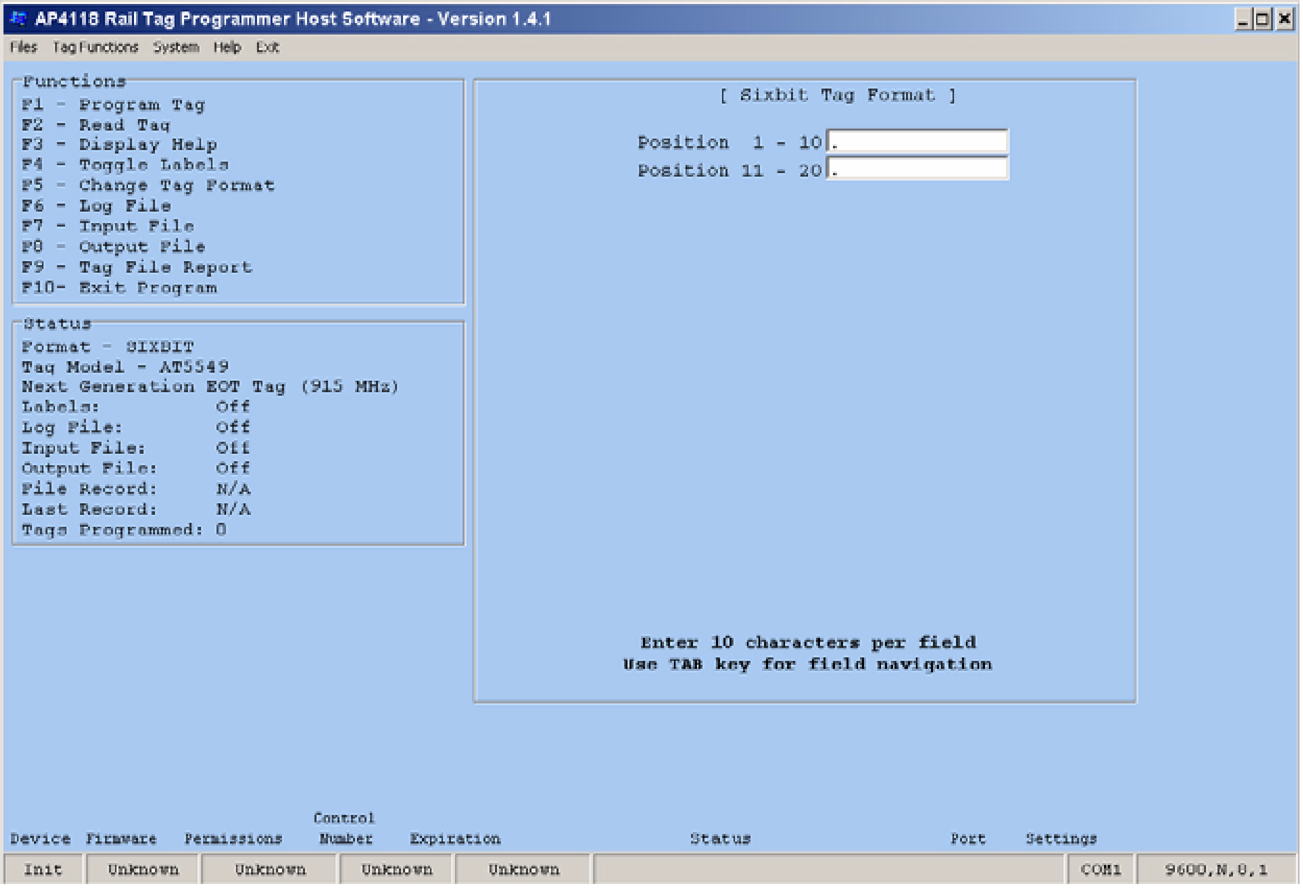
3. Open the USB drive folder on the host computer and locate the AP4118 Tag Programmer Host Software.

4. Click and drag the host software to the desired location on the host computer to begin the download.

5. Double-click on the AP4118 icon or select **Start > Programs** > **AP4118** > **AP4118** to launch the tag programming software.

[1] Use the PT5780 tag if configuring the AP4118 programmer for 902 to 928 MHz frequency band tags and the PT5785 tag if configuring the AP4118 programmer for 860 to 880 MHz frequency band tags.

Figure 3 shows the host software main screen.



```
Functions
F1 - Program Tag
F2 - Read Tag
F3 - Display Help
F4 - Toggle Labels
F5 - Change Tag Format
F6 - Log File
F7 - Input File
F8 - Output File
F9 - Tag File Report
F10- Exit Program

Status
Format - SIXBIT
Tag Model - AT5549
Next Generation EOT Tag (915 MHz)
Labels:           Off
Log File:         Off
Input File:       Off
Output File:      Off
File Record:      N/A
Last Record:      N/A
Tags Programmed: 0
```

```
[ Sixbit Tag Format ]

Position  1 - 10 .
Position 11 - 20 .

Enter 10 characters per field
Use TAB key for field navigation
```

Device	Firmware	Permissions	Control Number	Expiration	Status	Port	Settings
Init	Unknown	Unknown	Unknown	Unknown		COM1	9600,N,8,1

Figure 3 Tag Programming Host Software Main Screen

For complete instructions to install and operate the tag programming host software, refer to *AP4118 Rail Tag Programmer Host Software User Guide*.

Chapter 3

Configuring the AP4118 Rail Tag Programmer

This chapter briefly defines permissions and instructs how to configure the AP4118 Rail Tag Programmer using an PT5780 or PT5785 Permissions Tag.

Permissions

An AP4118 Rail Tag Programmer requires specific setup data before it can be used to program tags. Configuration of the programmer is done using a permissions tag. The permissions tag transfers parameters, such as permissions expiration date, programming session duration, programmer control number, and application tag permissions (read/write/authentication access rights, and ownership matching) to the rail tag programmer via a RF signal. Initial tag programmer configuration can be performed at the factory or at the customer site using a permissions tag.

Note: *Use the PT5780 tag if configuring the AP4118 programmer for 902 to 928 MHz frequency band tags and the PT5785 tag if configuring the AP4118 programmer for 860 to 880 MHz frequency band tags.*

PT5780 and PT5785 Permissions Tags

TransCore's permissions tag (Figure 4) is designed specifically to configure the AP4118 Rail Tag Programmer. During programmer configuration, the permissions tag data is validated by the programmer and, if valid, stored in the AP4118 tag programmer's nonvolatile memory.

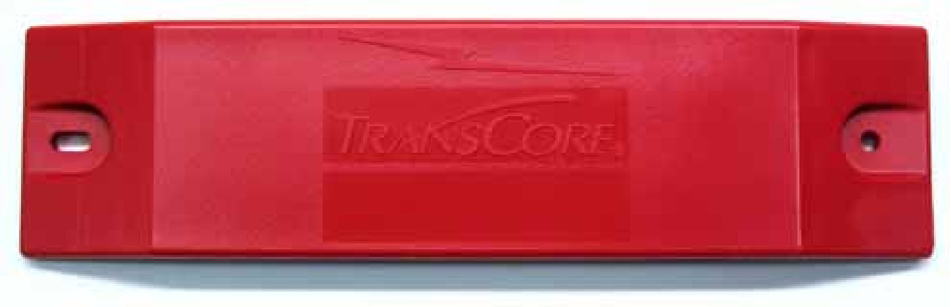


Figure 4 Permissions Tag

Tag programmer configuration information transferred from the permissions tag to the tag programmer can include:

- Rail tag owner values
- Rail tag ATA frame security character values
- Rail tag read and write access, by page
- Rail tag programmer control number (CN) matching
- Rail tag programmer lifetime and session dates/durations
- Rail tag programmer control/configuration
- ATA mode programming permissions
- Generic/non-AAR programming enable/disable

A permissions tag can be used once during a programmer's expected life cycle, or may be required on an ongoing basis during programmer use. Also, a different permissions tag can be used at any time in the programmer's lifetime to update or reconfigure the programmer. In some cases, multiple permissions tags for a single programmer may be required at a given site to enable tag programming by multiple owners, or to selectively enable or disable various programmer features.[1]

Ordering a Permissions Tag

Permissions tags may be obtained from TransCore Order Administration. An order form is available online at www.transcore.com/downloads.html under the USER and INSTALLATION MANUALS > Miscellaneous menu. Select PT5780/PT5785 Tag Order Form from the Miscellaneous pull-down list.

AP4118 Rail Tag Programmer Control Number and Serial Number

The CN is an eight-character alphanumeric string and is printed on a label on the rear panel of the AP4118 Rail Tag Programmer (e.g., CN: 43213131). The Serial Number is printed on the same label (e.g., Serial No. 545445456).

Permissions Tag Expiration Date

The expiration of the permissions tag can vary according to customer requirements. Depending on customer requirements, a tag permissions period may range from two years to an indefinite programming period. The expiration date, which is laser etched into the tag face, should coincide with the expiration or termination of the tag programming certification or authorization. The expiration date appears in the following

[1] Use the PT5780 tag if configuring the AP4118 programmer for 902 to 928 MHz frequency band tags and the PT5785 tag if configuring the AP4118 programmer for 860 to 880 MHz frequency band tags.

date format EXP: DD/MM/YYYY, for instance, 22/11/2009.

Once validated, the permissions tag settings are stored in the tag programmer. Nonvolatile permissions tag settings, including duration/expiration settings, will survive power cycles, resets, soft or warm boot, and flash application code download operations.

Configuring the Rail Tag Programmer Using the Permissions Tag

This section instructs how to configure the AP4118 Rail Tag Programmer using the permissions tag. For complete instructions on the operation of an AP4118 Rail Tag Programmer, refer to "Request Control Number (#43<cr>)" on page 24.

To use a Permissions Tag.

1. Switch on the tag programmer and wait for the READY LED to light (Figure 5).

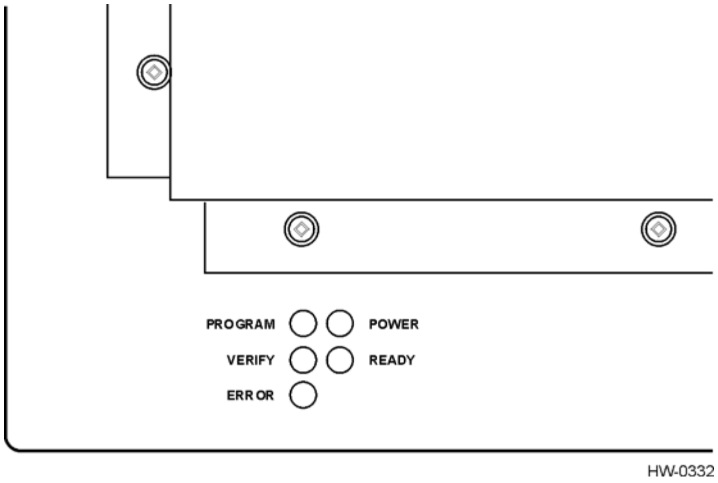


Figure 5 AP4118 Tag Programmer LEDs

2. Insert the permissions tag into the tag holder and close the door (Figure 6).

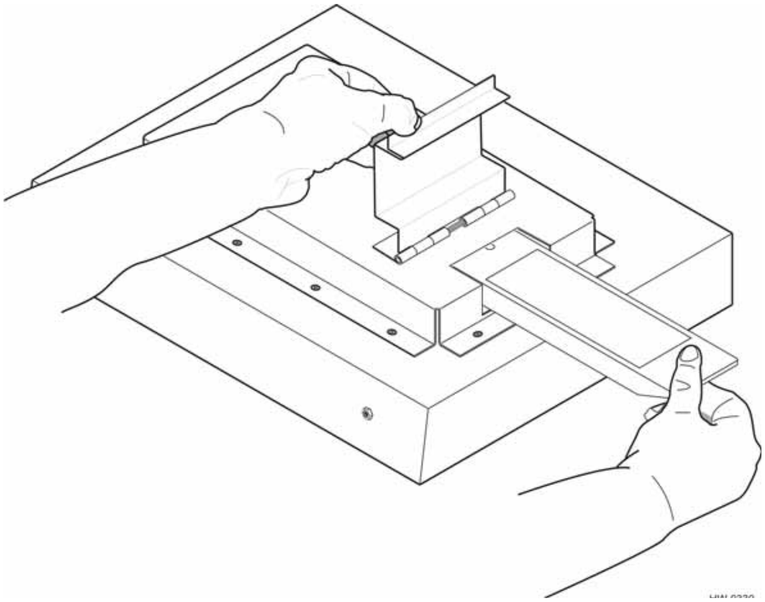


Figure 6 Permissions Tag Being Slid into Tag Holder

3. Wait 10 seconds.

4. Remove the permissions tag from the tag holder.

5. Insert a rail tag into the tag holder and close the door.

6. Start the tag programmer application, for instance, TransCore's host software program or terminal emulation software program, and test the programmer permissions by writing to and/or reading from the rail tag. See "RS–232 Command Set" on page 18 for programming and verifying tags.

   If the AP4118 tag programmer did not accept the permissions tag, you will not be able to program or read rail tags. Check the permissions tag expiration date (etched into the tag face) to ensure the tag is valid. You can enter the #44 Request Expiration Date to retrieve the permissions expiration. See "Request Expiration Date (#44<cr>)" on page 24 for more information. Finally, compare the permissions tag CN to the tag programmer CN (located on label affixed to back of tag programmer) to ensure they match.

Note: *If the permissions tag has not expired and its CN matches the tag programmer CN, contact TransCore Technical Support.*

Permissions Tag/Programmer Control Number Matching

The AP4118 tag programmer checks the permission tag's CN against its own CN prior to implementing the permissions tag settings. If the permissions tag CN value does not match or is invalid, then the permissions tag settings are discarded. If the permissions tag CN value matches, the tag programmer implements the permissions tag data.

AAR Tag Programming

Note: *This information applies to AAR-compliant tag programming, with the permissions tag settings as required to support AAR-compliant tag programming.*

AAR-compliant programming requires validation and evaluation of some of the incoming data, as described in the following section.

AAR User Data Type Detection and Format Delimiting

The AP4118 tag programmer supports only two basic AAR programming modes:

- programming of rail tag as an AAR-compliant rail tag, or
- programming of rail tag as an ASCII tag, according to the AAR Standard for Automatic Equipment Identification (S-918) definition.

The programmer evaluates the incoming data as being in one of the two programming mode categories, or responds with an error condition and prevents improper tag programming. Upon receiving a legacy AP4110 Tag Programmer-compatible command, the tag programmer evaluates the incoming user data to determine the data format code (DFC) value as specified in AAR Specification S-918. Table 3 lists the supported DFC values.

If the DFC is any value not listed in Table 3, the AP4118 creates an error condition and #ERROR response, and tag programming stops.

Table 3 Data Format Code, AP4118 Handling

6-Bit Binary DFC Value	Value Definition (per AAR Standard S-918)	AP4118 Handling
000000	6-bit ASCII format	Allow programming; security character enforcement required. Tag type handling not required.
110011	AAR-compliant: other; non-dynamic, non-passive alarm; AAR/ATA/ISO	Allow programming; enforce tag type and security characters. (This is the predominant DFC value for a rail tag.)
110100	AAR rail: toll road	Allow programming; enforce tag type and security characters

AAR Rail Tag Security Character Handling

Upon receiving a legacy AP4110-compatible tag programming command, the AP4118 programmer checks the incoming security characters embedded in the user data. If the incoming characters are secure, the AP4118 programmer then checks the secure characters against the security characters in the permissions tag. The data remains the same if a match is made; otherwise, an immediate data compare error occurs. This error is due to security characters being replaced by space characters.

Non-secure characters can be written to the tag without look-up or comparison. Regardless of the incoming programming command format (6-bit or 4-bit), the security characters reside in bits 106 through 117 of the actual tag ATA frame data. The AP4118 programmer treats the security field as two 6-bit characters. For reference, see Table C-1 of this user guide for the entire 6-bit ASCII character set.

Generic/Non-AAR Tag Programming

Note: *This information applies to programming of non-AAR-compliant tags, with the permissions tag settings factory-programmed to support this programming mode.*

In this programming mode, the programmer does not enforce all of the restrictions described in "AAR Tag Programming" on page 14. This mode permits nearly unrestricted programming of all user data bits in the ATA/AAR frame data space in the rail tag, excluding the security characters. This mode's programming functions are summarized here.

- No rail tag owner lookup or matching; all associated data in the permissions tag are ignored
- No security character lookup or matching in this mode, only non-secure characters (see Table C-1 of this user guide) may be programmed; ignore all associated data in the permissions tag

AP4118 Programmer CN and Permissions Tag ID Programming

During each tag programming operation, for both ATA and non-ATA data and addresses, the rail tag should be programmed with adequate information to allow tracing of the most recent tag programming operation to a specific programmer and/or permissions tag. Page 3 (hex address 0x18) of the rail tag is allocated for storing this information. Table 4 lists the page 3 layout in the rail tag.

Table 4 Rail Tag Page 3 Layout

BYTE	0	1	2	3	4	5	6	7
Rail Tag Data Address	0x18	0x19	0x1A	0x1B	0x1C	0x1D	0x1E	0x1F
Rail Tag Data Values	Least-significant 4 bytes of the PERMISSIONS TAG's unique ID.				Programmer CN (32 bits, 4 bytes) (DWORD)			

Safeguarding the Permissions Information

Once you have configured the AP4118 tag programmer, keep the permissions tag and/or programming certification or authorization available for future reference.

Chapter 4

Operating the Programmer

This chapter describes how to program a tag and how to verify or read data from a tag. This chapter also lists the RS–232 command set and describes the security character operation.

Before You Program Tags

Before you can begin programming, you must set up the programmer station as described in Chapter 2, “Programmer Basics.” The programmer must be connected to a host system using a customer-supplied programming software or TransCore AP4118 Rail Tag Programmer (TP) Host Software, or to a communications terminal. You will need to manually enter the RS–232 control commands if you are using a communications terminal.

You also need to configure the programmer using a permissions tag as described in Chapter 3, “Configuring the AP4118 Rail Tag Programmer.”

Chapter 2, “Programmer Basics,” provides instructions for installing the TP software. The accompanying *AP4118 Rail Tag Programmer Host Software User Guide* provides detailed instructions for operating the TP software.

Note: *If you are using the TransCore TP software on your host computer, do not use the RS–232 commands listed in this chapter.*

To operate the programmer using a communications terminal or a personal computer (PC) that does not use TransCore’s TP software, follow the steps in Programming a Tag and/or Verifying Tag Data sections of this chapter, as required to your task. Use the programmer commands listed in the “RS–232 Command Set” in this chapter.

The tag programmer is powered by 12V DC power supply converter that plugs into a standard 110V AC, 50-60 Hz electrical outlet. Refer to Chapter 2, “Programmer Basics,” or to Appendix A, “Tag Programmer Specifications,” for additional information.

Caution

TransCore tags are sensitive to ESD, and precautions are necessary to ensure proper tag programming operation. Use the grounded ESD static-suppression wrist strap when using the programmer. Additional protection from ESD is recommended where practical. Refer to Chapter 2, “Programmer Basics,” for recommended practices to protect the tags against ESD damage.

Indicator Lights

Figure 7 shows the indicator lights used by the programmer during operation.

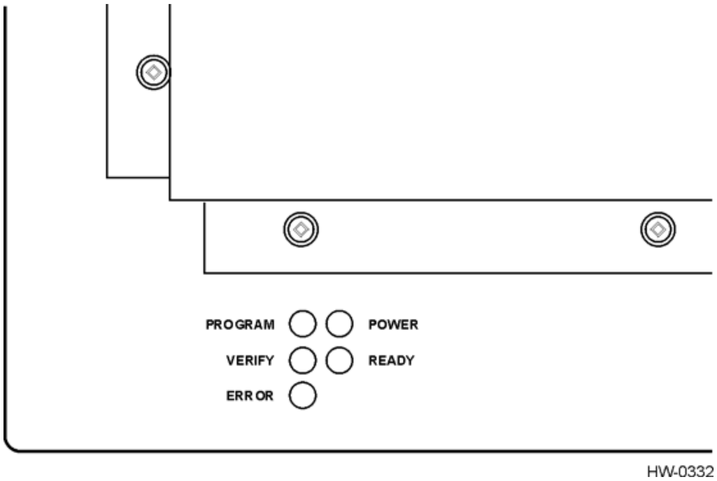


Figure 7 AP4118 Rail Tag Programmer Indicator Lights

PROGRAM	The tag is being programmed.
VERIFY	The programmer has read and verified the tag identification (ID) code.
ERROR	An error has occurred while programming the current tag.
POWER	The programmer is receiving power.
READY	The programmer is ready to program tags.

Sign-On

You should expect to see the following sign-on message on the screen of the host system or a communications terminal upon power-up or after you send the programmer command #30:

#Model AP4118 Ver X.XX
#Copyright 2005 TransCore

where X.XX is the current software version.

Command Execution Protocol

The programmer sends an appropriate acknowledgment to the host device upon the successful or unsuccessful execution of any command received.

All programmer commands are in the following format:

```
#abcxxxxxx...xxxxxx<cr>
```

where

#	start of command character. All commands must start with this character.
abc	the command
xxx...	user data
<cr>	carriage return

If command execution is successful, the programmer sends the response

```
#Done<cr>
```

If command execution is unsuccessful, the programmer sends the response

```
#Error<cr>
```

RS–232 Command Set

Programmer commands are listed below. Use these commands by following the format described in the Command Execution Protocol section above.

Note: *You have the option of placing the tag into the tag holder before issuing the command or you can enter the command and place the tag into the tag holder within 10 seconds. If you do not place the tag in the holder and close the door within 10 seconds, the programmer will abort this command. See “Programming a Tag” on page 24 for tag insertion instructions.*

Enable Tone Generator (#120<cr>)

The default state of the audible tone generator is tone-enabled. The tone generator creates audible responses described as a beep, triple beep, and solid tone. This command re-enables the programmer’s audible response capability after it has been disabled by the #121 command.

Response:

```
#Done<cr>
```

Disable Tone Generator (#121<cr>)

This command disables the programmer’s audible response capability. The default state of the audible tone generator is tone-enabled.

Response:

```
#Done<cr>
```

Program Tag Using TransCore 6-Bit ASCII Format (#200xxx...<cr>)

The programmer programs the user data (xxx...) into the tag using the TransCore 6-bit ASCII format. This format requires 20 data characters. Refer to Appendix C, “TransCore 6-Bit ASCII Format,” for a description of this format.

During execution of this command, a series of beeps indicates that the programmer is waiting for a tag to be placed in the holder. If you do not place the tag in the tag holder and close the door within 10 seconds, the programmer will abort this command. See “Programming a Tag” on page 24 for tag insertion instructions.

Response:

```
#xxxxxxxxxxxxxxxxxxxx<cr>
#Done<cr>
```

or

```
#Error<cr>
```

Program Tag Using 4-Bit ASCII Hex Format (#201xxx...<cr>)

The programmer programs the user data (xxx...) into the tag using the 4-bit ASCII hexadecimal (hex) format. This format requires 30 characters of data. Refer to Appendix D, “4-Bit ASCII Hex Codes,” for a description of this ASCII hex format.

During execution of this command, a series of beeps indicates that the programmer is waiting for a tag to be placed in the holder. If you do not place the tag in the tag holder and close the door within 10 seconds, the programmer will abort this command. See “Programming a Tag” on page 24 for tag insertion instructions.

Response:

```
#xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx<cr>
#Done<cr>
```

or

```
#Error<cr>
```

Read/Verify Tag Data Using TransCore 6-Bit ASCII Format (#210<cr>)

The programmer reads the data from the tag, puts it into the TransCore 6-bit ASCII format, and sends it to the host device. Refer to Appendix C for a description of this 6-bit ASCII format.

During execution of this command, a series of beeps indicates that the programmer is waiting for a tag to be placed in the holder. If you do not place the tag in the holder and close the door within 10 seconds, the programmer will abort this command. See "Programming a Tag" on page 24 for tag insertion instructions.

Response:

```
#xxxxxxxxxxxxxxxxxxxx<cr>
#Done<cr>
```

or

```
#Error<cr>
```

Read/Verify Tag Data Using 4-Bit ASCII Hex Format (#211<cr>)

The programmer reads the data from the tag, puts it into the 4-bit ASCII hex format, and sends it to the host device. Refer to Appendix D, "4-Bit ASCII Hex Codes," for this ASCII hex format.

During execution of this command, a series of beeps indicates that the programmer is waiting for a tag to be placed in the holder. If you do not place the tag in the tag holder and close the door within 10 seconds, the programmer will abort this command. See "Programming a Tag" on page 24 for tag insertion instructions.

Response:

```
#xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx<cr>
#Done<cr>
```

or

```
#Error<cr>
```

Write a Page of Data Using Hex Format (#220yyxxxx…<cr>)

The programmer programs the user hex data (xxxxxxxxxxxxxxxx), 16 hex characters, starting at hex address yy. The starting address (yy) must be on an 8-byte page boundary. For example, the starting address will be 00, 08, 10, 18.

If the tag is not already in the tag holder, a series of beeps indicates that the programmer is waiting for a tag to be placed in the holder. If you do not place the tag in the tag holder and close the door within 10 seconds, the programmer will abort this command. See "Programming a Tag" on page 24 for tag insertion instructions.

The programmer will not program the data if the programmer does not have permission to write to the memory area of the tag that is requested by the user. Each programmer will be assigned permissions at the factory or will use a permissions tag that must be placed in to the programmer at pre-specified intervals.

Response:

```
#xxxxxxxxxxxxxxxx<cr>
#Done<cr>
```

or

```
#Error<cr>
```

Write a Byte of Data Using Hex Format (#221yyxx<cr>)

The programmer programs the user hex data (xx), 2 hex characters, at hex address yy.

If the tag is not already in the tag holder, a series of beeps indicates that the programmer is waiting for a tag to be placed in the holder. If you do not place the tag in the holder and close the door within 10 seconds, the programmer will abort this command. See “Programming a Tag” on page 24 for tag insertion instructions.

The programmer will not program the data if the programmer does not have permission to write to the memory area of the tag that is requested by the user. Each programmer will be assigned permissions at the factory or through the use of a permissions tag that must be inserted into the programmer at pre-specified intervals.

Response:

```
#xx<cr>
#Done<cr>
```

or

```
#Error<cr>
```

Read a Page of Data Using Hex Format (#230yy<cr>)

The programmer reads hex data (xxxxxxxxxxxxxxxx), 16 hex characters, starting at hex address yy. The starting address (yy) must be on an 8-byte page boundary. For example, the starting address will be 00, 08, 10, 18, …

If the tag is not already in the tag holder, a series of beeps indicates that the programmer is waiting for a tag to be placed in the holder. If you do not place the tag in the holder and close the door within 10 seconds, the programmer will abort this command. See “Programming a Tag” on page 24 for tag insertion instructions.

The programmer will not read the data if the programmer does not have permission to read the memory area of the tag that is requested by the user. Each programmer will be assigned permissions at the factory, or through the use of a permissions tag that must be placed in to the programmer at pre-specified intervals.

Response:

```
#xxxxxxxxxxxxxxxx<cr>
#Done<cr>
```

or

```
#Error<cr>
```

Read a Byte of Data Using Hex Format (#231yy<cr>)

The programmer reads a byte of hex data (xx), 2 hex characters, from hex address yy. If the tag is not already in the tag holder, a series of beeps indicates that the programmer is waiting for a tag to be placed in the holder. If you do not place the tag in the holder and close the door within 10 seconds, the programmer will abort this command. See "Programming a Tag" on page 24 for tag insertion instructions.

The programmer will not read the data if the programmer does not have permission to read the memory area of the tag that is requested by the user. Each programmer will be assigned permissions at the factory or through the use of a permissions tag that must be placed in to the programmer at pre-specified intervals.

Response:

```
#xx<cr>
#Done<cr>
```

or

```
#Error<cr>
```

System Reset (#30<cr>)

The command resets the programmer to its original power-on state with the self-test. System reset takes approximately eight seconds.

Response (system sign-on message):

#Model AP4118 Ver X.XX
#Copyright 2005 TransCore

where X.XX is the current software version

or

```
#Error<cr>
```

Display System Status (#40<cr>)

This command requests a two-digit status response code from the programmer indicating the status of the last operation performed. Refer to Appendix E, "Programmer Response Codes," for a description of the programmer response codes that refer to system status.

Response:

```
#xx<cr>
```

where XX is the two-digit response code. The code 00 indicates that the programmer performed the last operation without error.

Display Tag Door Switch Status (#42<cr>)

This command requests a two-digit status response from the programmer indicating the status of its tag door switch.

Response:

```
#00<cr>door open
#01<cr>door closed
```

Request Control Number (#43<cr>)

This command requests the AP4118 tag programmer control number.

Response:

```
#xxxxxxxx
```

where

```
xxxxxxxx is hex data (control number)
```

Request Expiration Date (#44<cr>)

This command requests the tag programmer permissions expiration date.

Response:

`#MMDDYYYY` or `None`

Request Programming Mode (#45<cr>)

This command requests the current AP4118 programming mode, based on the permissions tag.

Response:

`#AAR` = AAR S-918 compatible

`#GEN` = Generic, non-AAR programming format

Programming a Tag

Note: *You have the option of first placing at tag into the tag holder before issuing the command or you can enter the command and place the tag into the tag holder within 10 seconds. If you are programming an AT5549 End-of-Train Tag, place the tag into the programmer adapter (P/N 20-4118-001), with the tag's reflective side up, before sliding the tag into the tag holder. If you do not place the tag in the tag holder and close the door within 10 seconds, the programmer will abort this command.*

To program a tag

1. Send the programmer the appropriate command along with the user data you want to program onto the tag. Use command #200 if you are using the TransCore 6-bit format for your data, command #201 for the 4-bit hex data format, command #220 if you are writing a page of SeGo data to the tag, and command #221 if you are writing a byte of SeGo data to the tag.

2. When the programmer begins its 10-second timer, indicated by an audible beep once every second, slide the tag into the tag holder (Figure 8) and close the door.

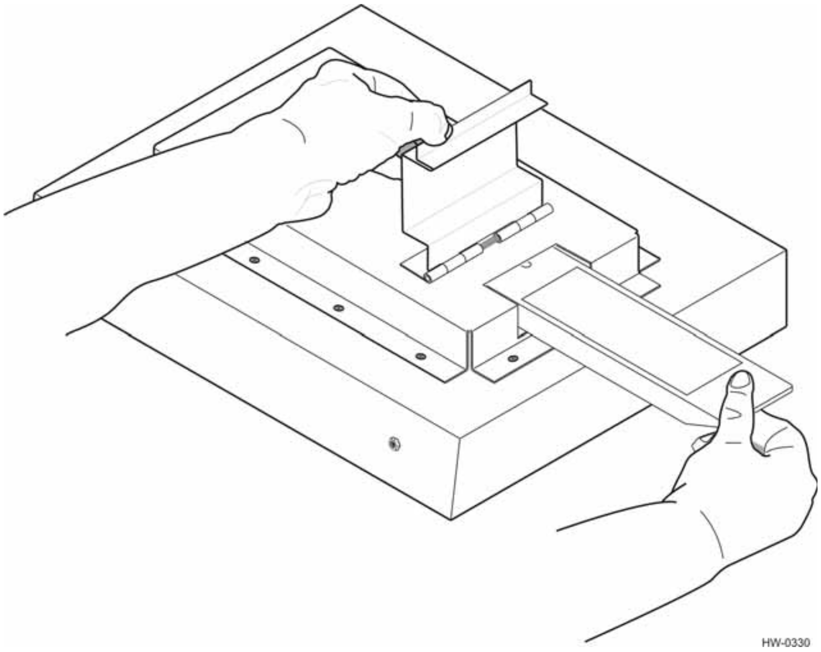


Figure 8 Tag Being Slid into Tag Holder

3. Or place the AT5549 End-of-Train Tag reflective side up into programmer adapter, slide into the tag holder (Figure 9), and close the door.

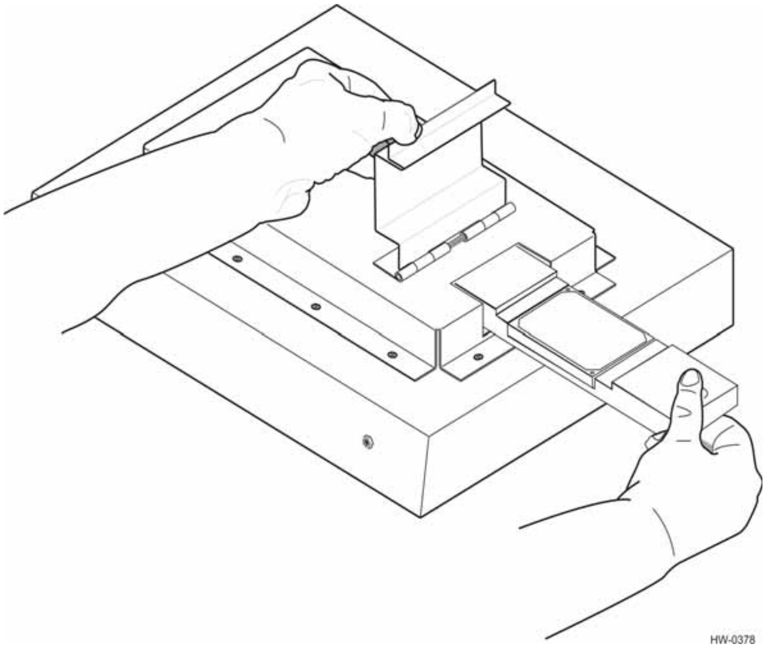


Figure 9 Programmer Adapter and AT5549 EOT Tag Being Slid into Tag Holder

If the tag is placed before or during the 10-second period, the programmer programs the user data, verifies the user data, and sends a response to the host system. If the tag is not placed during the 10-second period, the programmer aborts the command, sends the #Error<cr> response to the host, and emits a continuous audible tone.

When the tag is successfully programmed, the programmer responds with a triple beep and with the following response:

```
xxx...<cr>
#Done<cr>
```

where xxx... is the tag data read from the tag and is encoded in the specified format.

If the tag programming process fails, the programmer responds with a continuous audible tone and with #Error<cr>.

If you receive this response, repeat this sequence beginning at step 1. Also refer to Table 5 on page 28 for troubleshooting remedies.

Verifying Tag Data

To verify the data on a tag

1. Send the programmer the appropriate command. Use command #210 if you are using the TransCore 6-bit format for your data, command #211 for the 4-bit hex

data format, command #230 if you are verifying a page of SeGo data in the tag, or command #231 if you are verifying a byte of SeGo data in the tag.

2. When the programmer begins its 10-second timer indicated by an audible beep once every second, slide the proper tag (using the AT5549 End-of-Train Tag programmer adapter) into the tag holder and close the door. (Figure 8 and Figure 9).

   If you place the tag during the 10-second period, the programmer reads the tag data, checks for data integrity, and sends the tag data and a #Done response to the host system. If you do not slide the tag into the holder during the 10-second period, the programmer aborts the command and sends the #Error<cr> response to the host.

If the verification process is not successful, repeat this sequence beginning at step 1. Also refer to Table 5 on page 28 for troubleshooting remedies.

Security Characters

The tag security character function prevents the use of unauthorized tags in TransCore electronic identification systems. All AP4118 Rail Tag Programmers have the capability, when enabled by settings in the PT5780 or PT5785 Permissions Tag, to program specific, predefined security characters into a tag. This capability must be specified in an authorized TransCore order administration requisition. Security characters are available for American Trucking Associations and International Standards Organization tag data formats.

When enabled, the security characters, which are specific 6-bit patterns designated for security coding purposes, will be two specific characters from the list of security characters in Appendix F or one character each from Appendix F, “Security Characters,” and Appendix C, “TransCore 6-Bit ASCII Format.”

When a tag programmer has tag security enabled, all tags programmed with secure characters will contain the assigned security characters in the designated positions of the tag data. Security characters comprise the second half of the 26th character through the first half of the 29th character when data is in the 4-bit hex format. The security characters are the 18th and 19th of the 20 six-bit ASCII characters in the tag ATA frame.

The programmer places security characters into their associated tag data locations, via the permissions tag, when security characters in the tag data match the internal security characters stored in the programmer.

The programmer will not store in a tag any security codes that differ from those stored in the programmer’s internal memory. If you attempt to program the tag with security characters that do not match the internal security characters of the programmer (i.e., incorrect or improperly obtained codes), the programmer automatically programs the space character into both security character locations.

Chapter 5

Troubleshooting

Table 5 lists some problems that you may encounter when using the AP4118 Rail Tag Programmer, along with suggestions for troubleshooting each problem.

If a problem persists, call TransCore Technical Support.

Table 5 Troubleshooting Problems and Solutions

Problem	Suggested Solutions
READY LED does not illuminate at power-on.	• Check power transformer. • Replace the internal 5-amp, 250-volt Fast Blo fuse.
System powers on but does not sign on.	• Verify RS–232 connection at the programmer and at the host. • Verify that host baud rate is set to 9600.
Programmer beeper is not heard during tag programming/verify sequence, or system continually beeps until time-out.	• Verify that the door is fully closed. • Verify that the optical switch that is activated by the tag door is clean and is clear of any dirt buildup.
Tag programmer reads tags but programming sequence fails repeatedly.	• Enter command #40 to retrieve the response code then use the programmer response codes list in Appendix E, “Programmer Response Codes,” to identify the problem. • Improve electrostatic discharge protection in programming area. • Tag programmer permissions has expired or programming session has timed out. Enter command #44 to retrieve the tag programmer permissions expiration date. Also, review the expiration date etched into the PT5780 or PT5785 Permissions Tag and/or review the permissions certificate for the tag programmer.
Requested *verify* data does not match programmed data.	• Check for security code violations. • Check programmed data for accuracy.
System resets intermittently.	• Check for damage to DC power connection. • Check for a loose AC power connection.

Appendix A

Tag Programmer Specifications

Tag Programmer Specifications

This appendix lists the AP4118 Rail Tag Programmer physical and environmental specifications.

Table A-1 lists the AP4118 Rail Tag Programmer parameters and specifications.

Table A-1 AP4118 Rail Tag Programmer Physical and Environmental Specifications

Parameter	Specification
Size (W x H x D)	14.3 x 3.2 x 11.5 in (36.3 x 8.1 x 29.2 cm)
Weight	6.8 lb (3.1 kg)
Input Voltage	12V DC
Power Consumption	12 W
Excessive Current Protection	5-amp, 250-volt Fast Blo fuse
Operating Temperature	32°F to 122°F (0°C to 50°C)
Storage Temperature	-4°F to 185°F (-20°C to 85°C)
Humidity	95% noncondensing @77°F to 122°F (25°C to 50°C)
Enclosure	Dustproof
Operational Vibration	1.04 G_{rms}, 5-500 Hz, power spectral density-uniform 0.0022 G^2/Hz, 1 hour per axis
Shock	4 $G_{zero\text{-}to\text{-}peak}$ by 11 ms half-sine duration in all 3 axes
PC-to-Programmer Cable	RS–232 (data terminal equipment)
RF Power	Tag holder: 50 mW
FCC Classification	Part 15, site license not required

Environmental

The tag programmer is designed to operate in typical office environment conditions. The tag programmer performs to the specifications listed in Table A-1 and is ready to program with a warm-up time of not more than three minutes.

Appendix B

RS–232 Connector Pin-outs

RS–232 Connector Pin-outs

Table B-1 identifies the pin-outs of the programmer’s DB9 rear panel RS–232 connector.

Note: *The signal input/output is listed from the tag programmer side.*

Table B-1 RS–232 Connector Pin-outs

Pin	Input/Output	Function
1	N/A	No connection
2	Out	Transmit
3	In	Receive
4	N/A	No connection
5	In/Out	Signal ground
6	N/A	No connection
7	In	Clear to send (CTS)
8	Out	Request to send (RTS)
9	N/A	No connection

Appendix C

TransCore 6-Bit ASCII Format

TransCore 6-Bit ASCII Format

In TransCore's 6-bit ASCII format, each ASCII data character is represented by a certain 6-bit pattern. These bit patterns are placed in specific groups of six consecutive bits within the tag. The ASCII characters and their corresponding 6-bit codes are listed in Table C-1. TransCore's 6-bit ASCII codes are produced by subtracting 00100000 from the character's 8-bit ASCII value.

Table C-1 6-Bit ASCII Codes

Character	6-Bit Code	Character	6-Bit Code
(space)	000000	G	100111
*	001010	H	101000
-	001101	I	101001
.	001110	J	101010
/	001111	K	101011
0	010000	L	101100
1	010001	M	101101
2	010010	N	101110
3	010011	O	101111
4	010100	P	110000
5	010101	Q	110001
6	010110	R	110010
7	010111	S	110011
8	011000	T	110100
9	011001	U	110101
A	100001	V	110110
B	100010	W	110111
C	100011	X	111000
D	100100	Y	111001
E	100101	Z	111010
F	100110		

Appendix D

4-Bit ASCII Hex Codes

4-Bit ASCII Hex Codes

In TransCore's 4-bit hex format, each ASCII character is represented by a certain 4-bit pattern. These bit patterns are placed in specific groups of four consecutive bits within the tag. The ASCII characters and their corresponding 4-bit hex codes are listed in Table D-1.

Table D-1 4-Bit ASCII Hex Codes

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

Programmer Response Codes

The programmer returns one of these codes in response to command #40, Display System Status. Table E-1 lists the response codes.

Table E-1 Programmer Response Codes

Code	Response	Description
00	Operation successful/ no error	The last operation performed by the programmer was executed without errors.
01	Illegal command	Invalid command sent to the programmer.
02	Invalid tag data	The user data sent to the programmer was in the incorrect format (such as an illegal hex character).
03	Presence time-out on program cycle	The programmer did not recognize the placement of a tag in the tag holder during the 10-second period prior to a programming sequence.
04	Presence time-out on verify cycle	The programmer did not recognize the placement of a tag in the tag holder during the 10-second period prior to a verification sequence.
05, 06, 07	N/A	Reserved
08	Data compare error	During a programming sequence, the data read from the tag did not correspond to the user data that was written to the tag.
09	Locked tag	The programmer has detected a lock condition of the tag. The tag data is stored permanently and cannot be changed.
0A	No frame marker	The programmer could not decipher a frame marker from the tag. Either the tag is not in the tag holder, is damaged, or the RF is not being generated and programming and/or verifying the tag data can not occur.
0B	Bad tag parity	The checksum bits read from the tag did not correspond to the checksum that the programmer calculated from the tag data.
0C	Security code operation	The AP4118 has detected an attempt to write illegal security characters to the security character positions of the tag’s ATA frame data. The AP4118 has then coded the SPACE character in the positions, therefore altering the user’s original tag data.
0D	Invalid permissions	This programmer does not have permission to write data to the tag address that the user requested.

Appendix F

Security Characters

Security Characters

You can enter security characters into a TransCore AT5118 Tag if your programmer has been configured at the factory to perform the security character function. These ASCII security characters are encoded in the Amtech 6-bit data encoding scheme and are listed in Table F-1.

Table F-1 ASCII Security Characters

Character	6-Bit Code
!	000001
"	000010
#	000011
$	000100
%	000101
&	000110
'	000111
(	001000
)	001001
+	001011
,	001100
:	011010
;	011011
<	011100
=	011101
>	011110
?	011111
@	100000
[	111011
\	111100
]	111101
^	111110
_	111111