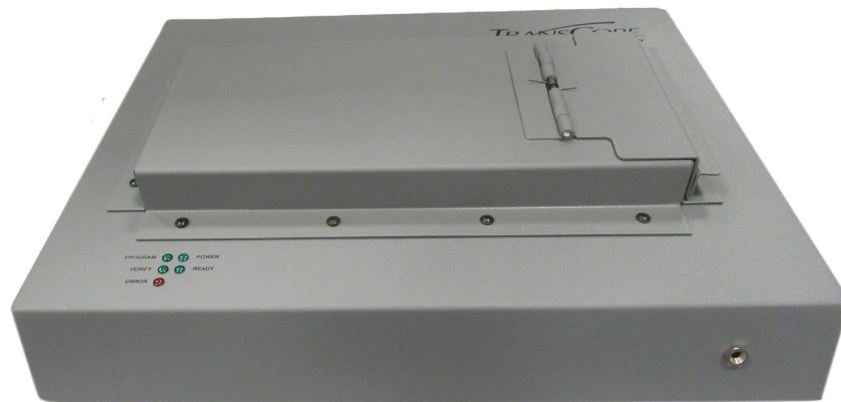# AP4119 Rail Tag Programmer

User Guide

The AP4119 Rail Tag Programmer User Guide provides information necessary for installing, operating, and troubleshooting the AP4119 Rail Tag Programmer.

Figure 1 AP4119 Rail Tag Programmer

This Guide describes the proper setup and operation of the AP4119 Rail Tag Programmer and provides procedures useful for troubleshooting problems encountered after installation.

The AP4119 Rail Tag Programmer User Guide is designed to be used by technical personnel responsible for setting up and operating the AP4119 Rail Tag Programmer and related software.

TRANSCORE®
Trusted Transportation Solutions

For further information, contact:

TransCore
8600 Jefferson Street NE
Albuquerque, NM 87113 USA

Web: www.transcore.com

TransCore Technical Support
Phone: (505) 856-8007

email: RFIDtechsupport@transcore.com

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

Table of Contents

List of Figures

List of Tables

Chapter 1 Programmer Basics

This chapter lists the factory-supplied and optional equipment for the AP4119 programmer, and describes how to prevent electrostatic discharge (ESD) damage to tags, connect to a host device, and install the optional tag programming software to the host computer.

Factory Supplied and Optional Equipment

The programmer is shipped with the items listed. Make sure all of the items are included when unpacking the order:

- AP4119 Rail Tag Programmer
- Programmer Accessory Kit P/N 19143-01 (Table 1)

Table 1 Accessory Kit P/N 19143-01

Accessory Kit Item	Qty (ea)
Power transformer and cable to convert AC voltage to 24 VDC	1
Power cord	1
Programmer-to-PC RS–232 connecting cable	1
Programmer-to-PC USB Type A Cable	1
Static-suppression wrist strap with cord	1
TransCore USB flash drive with AP4119 Tag Programmer Host Software	1
AP4119 Programmer User Guide	1

If any of the items in Table 1 are missing from the order, contact the TransCore distributor.

Note: *The AT5549 End-of-Train Tag requires a Programmer Adapter (P/N 20-4119-001) in order to seat the AT5549 tag during programming. This adapter is an optional purchase item.*

The AP4119 requires a host computer system such as a PC or communications terminal for operation. TransCore does not supply this equipment for the AP4119 Rail Tag Programmer.

Preventing Electrostatic Discharge Damage

Caution

⚠️ **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 tabletop
- 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

Anti-static Workstation

In a factory or fixed indoor environment, position the programmer in an anti-static workstation. Well-designed workstations use a system of multiple precautions against ESD. Figure 2 shows an anti-static workstation.
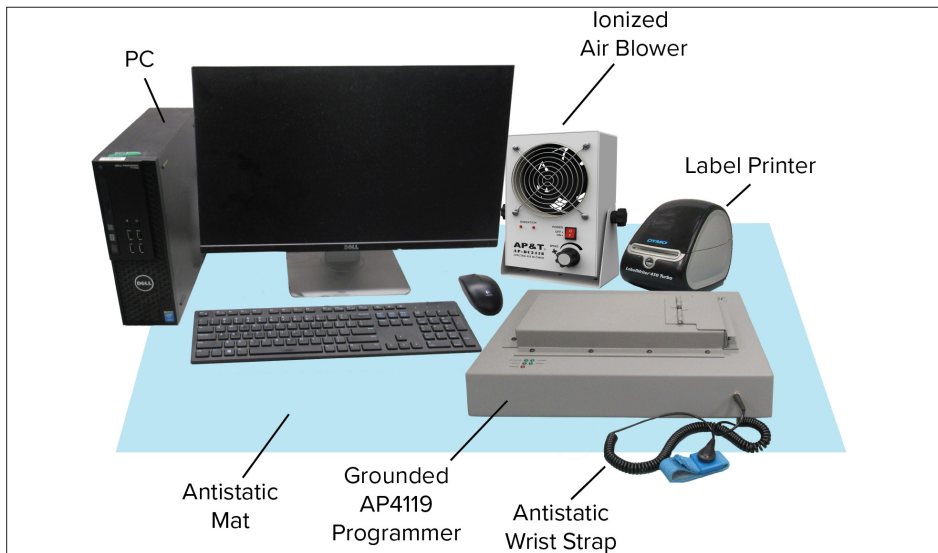
Figure 2 Programming Workstation with ESD Controls

Contact TransCore Technical Support with any questions concerning the suitability of the programming station or environment.

Take precautions to prevent ESD from affecting programming operations. A list of some of the most effective precautions follows. The potential for ESD damage decreases when two or more precautions are used together.

ESD Wrist Strap

Caution
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.

Antistatic Mat

The anti-static work mat on which the 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

An ionized air blower may be set up at the 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 anti-static mat or within an ESD-safe container until they are programmed. Grounded metallic containers are best for inhibiting ESD.

Connecting to a Host Device

The programmer requires connection to a host device — a host computer or communications terminal — for normal operation. Figure 3 shows the RS-232 and USB connectors from the programmer to the host.
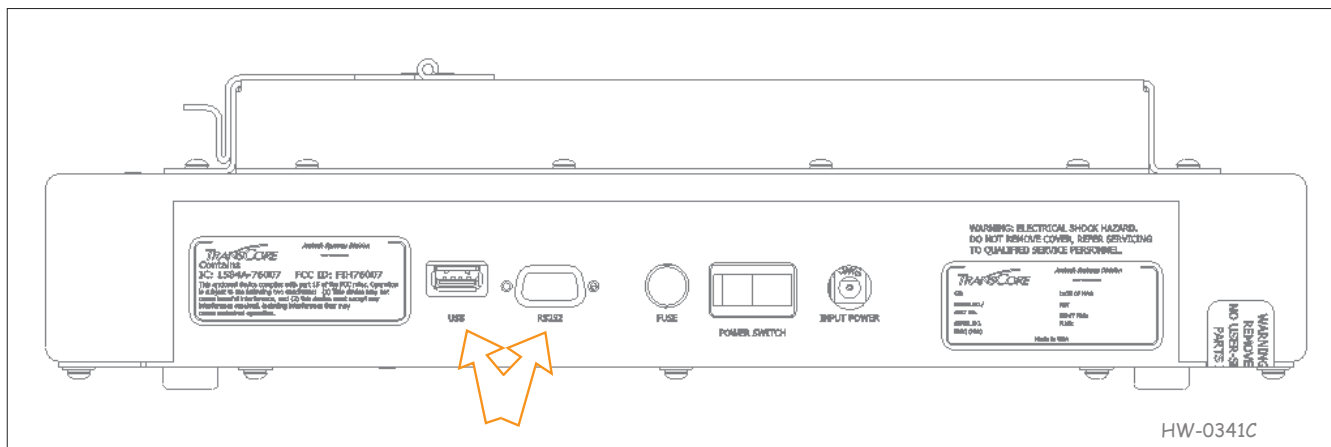
Figure 3 RS–232 and USB Connectors

For RS–232, if the host system is a PC or a communications terminal, a null modem is not needed.

The tag programmer's PC interface complies with the RS–232 and USB 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

Download USB Drivers

Before using the AP4119 programmer with a USB connection, the proper drivers may need to be installed on the host computer. The USB drivers can be installed during the AP4119 Host installation process ("Rail Tag Programmer Host Software" on page 1–11). Alternatively, the FTDI USB drivers can be manually installed by following the next steps:

1. Connect the USB cable from the USB port at the back of the AP4119 to one of the host computer's USB ports.

2. Allow the device driver to be detected and installed automatically. If the device driver fails to automatically load, disconnect the AP4119 from the host computer.

3. Locate and download the device driver setup executable file for your operating system from https://ftdichip.com/drivers/

Detailed installation guides can be found here: https://ftdichip.com/document/installation-guides/.

4. Launch the downloaded software and follow the steps to install the driver.

5. Reconnect the AP4119 and allow it to be detected by the operating system.

Note: *While any hyperlinks included in this section were valid at the time of publication, TransCore cannot guarantee their long term validity.*

Connect the programmer to the host computer

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

2. Switch on power to the programmer. The POWER then READY LEDs should illuminate. When the READY LED illuminates, 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, use a PT5780 permissions tag or License Certificate (requires AP4119 Host Software) to program specific operational and security information into the programmer. "Chapter 2 Configuring the Programmer" on page 2–13 instructs the user in configuring the tag programmer using the permissions tag.

   If the READY LED does not illuminate or if the host system cannot communicate with the tag programmer, refer to "Chapter 4 Troubleshooting" on page 4–31.

AP4119 Host Software

The AP4119 Tag Programmer Host Software can be found on the provided TransCore USB Drive. The host software is not necessary to use the AP4119 Rail Tag Programmer. Many customers prefer to use their specific host software to communicate with the tag programmer. However, the AP4119 Tag Programmer host software is required to download License Certificates to the AP4119 and to upgrade the AP4119 Rail Tag Programmer software. Refer to the AP4119 Rail Tag Programmer Host Software User Guide (16-4119-003) for detailed instructions.

Rail Tag Programmer Host Software

Install the rail tag programmer host software on the PC after properly connecting the hardware. The PC must have both a hard disk and a USB port.

1. Make sure the PC is running.

2. Insert the provided USB Flash drive into an available USB port.

3. Open the AP4119 Host folder and click on the **Install** icon (Figure 4). Follow the instructions to create a folder on the PC hard drive.
   The host software is installed on the PC in C:\Program Files\Transcore\AP4119 Host and an icon is added to the computer's desktop as well as adding a shortcut to the computer's **Start** Menu.
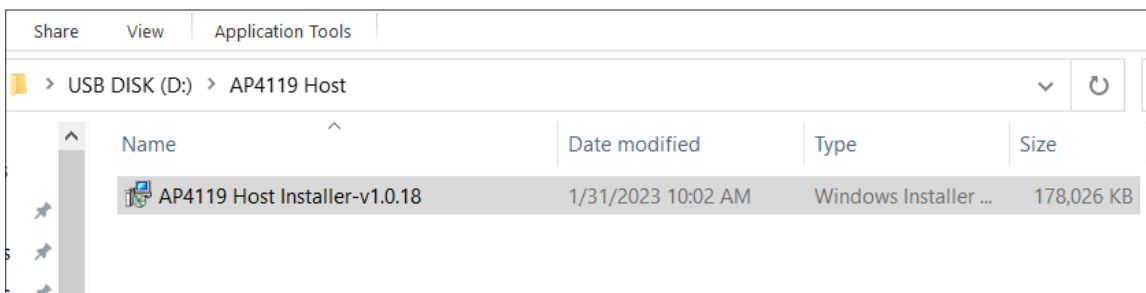
Figure 4 Host Software Installer

4. Install the USB and Label Printer Drivers (Figure 5).
   Drivers are located on the PC in C:\Program Files\Transcore\AP4119 Host\Drivers.
   The driver packages that need to be installed are:
   • CDM212364_Setup.exe (The AP4119 USB Port Driver)
   • DLS8Setup.8.7.2.exe (The DYMO® Label Printer Driver)
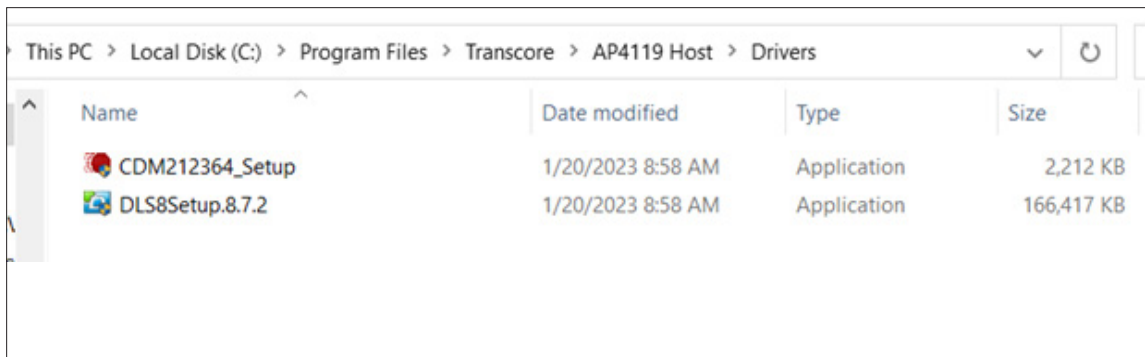
Figure 5 Label Printer and USB Drivers

5. Remove the USB flash drive from the computer USB port and reboot the PC.

Figure 6 shows the host software main screen.

For complete instructions on how to install and operate the tag programming host software, refer to the AP4119 Rail Tag Programmer Host Software User Guide.
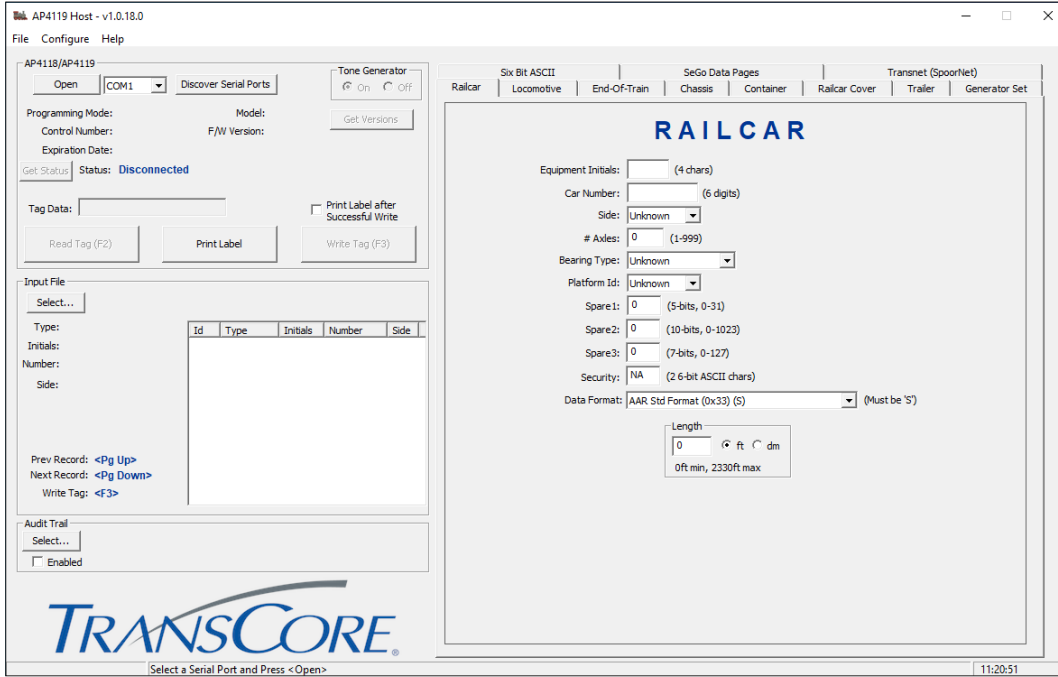
Figure 6 Tag Programming Host Software Screen

Chapter 2 Configuring the Programmer

This chapter briefly defines permissions and instructs how to configure the AP4119 Rail Tag Programmer using a PT5780 Permissions Tag.

Permissions

An AP4119 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 or a License Certificate (requires AP4119 Host Software). Refer to the Host Software User Guide for download instructions. The permissions tag transfers parameters, such as permissions expiration date and application tag permissions (read/write/ authentication access rights and ownership matching) to the rail tag programmer via an RF signal. Initial tag programmer configuration can be performed at the factory or at the customer site using a permissions tag.

PT5780 Permissions Tag

TransCore's permissions tag (Figure 7) is specifically designed to configure the AP4119 Rail Tag Programmer. During programmer configuration, the permissions tag data is validated by the programmer and, if valid, stored in the AP4119 tag programmer's nonvolatile memory.

Figure 7 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 dates
- 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.

Ordering a Permissions Tag

Contact TransCore Order Administration to obtain the appropriate Permissions Tag for your system requirements.

AP4119 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 AP4119 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 date format EXP: DD/MM/YYYY, e.g., 22/11/2021.

Once validated, the permissions tag settings are stored in the tag programmer. Nonvolatile permissions tag settings, including 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 AP4119 Rail Tag Programmer using the permissions tag.

To use a Permissions Tag

1. Switch on the tag programmer and wait for the READY LED to illuminate (Figure 8).
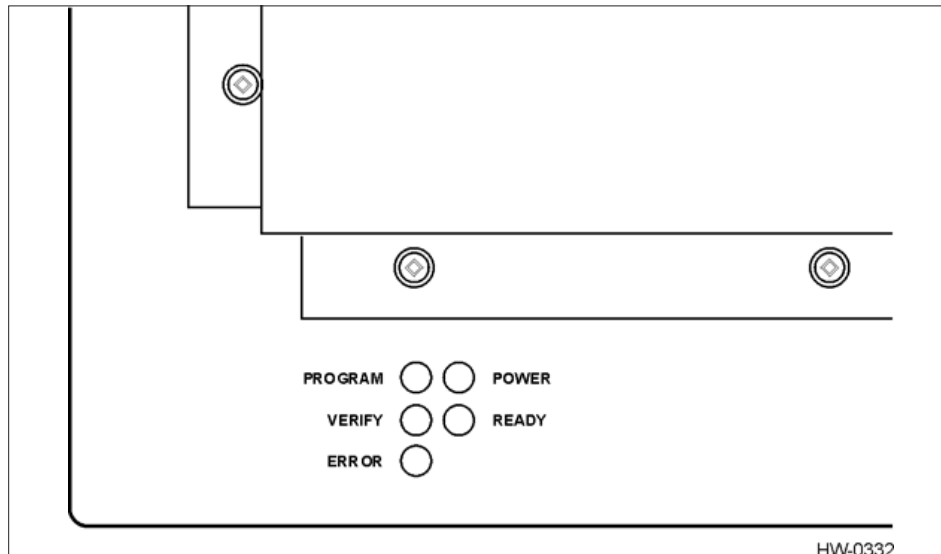
Figure 8 AP4119 Tag Programmer LEDs

2. Insert the permissions tag into the tag holder and close the door (Figure 9).
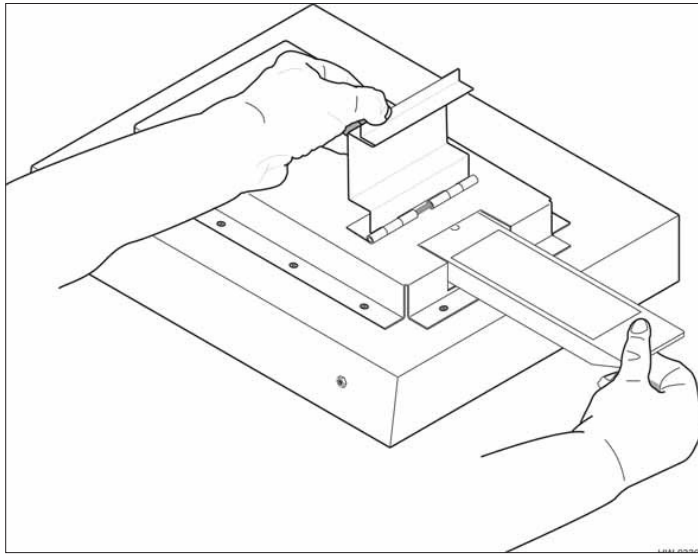
Figure 9 Permissions Tag Being Slid into Tag Holder

3. The Verify LED will illuminate. After the programmer has been properly configured, the programmer responds with 3 short beeps.

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, such as TransCore's host software program or a terminal emulation software program, and test the programmer permissions by writing to and/or reading from the rail tag. Refer to "Serial Command Set" on page 3–20 for programming and verifying tags.

    If the AP4119 tag programmer did not accept the permissions tag, you will not be able to program rail tags. Check the permissions tag expiration date (etched into the tag face) to ensure the tag is valid. Enter #44 Request Expiration Date to retrieve the permissions expiration. Refer to "#44<cr>   Request Expiration Date" on page 3–24 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 AP4119 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

This section 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.

AAR User Data Type Detection and Format Delimiting

The AP4119 tag programmer supports 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 AP4118 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 AP4119 creates an error condition and `#Error` response, and tag programming stops.

Table 3 Data format code, AP4119 handling

6-Bit Binary DFC Value	Value Definition (per AAR Standard S-918)	AP4119 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 AP4118-compatible tag programming command, the AP4119 programmer checks the incoming security characters embedded in the user data. If the incoming characters are secure, the AP4119 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 AP4119 programmer treats the security field as two 6-bit characters. Refer to "Table 10 6-Bit ASCII Codes" on page C–34 for the entire 6-bit ASCII character set.

TransCore Proprietary

Generic/Non-AAR Tag Programming

This section 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 2–16. 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 look up or matching; all associated data in the permissions tag are ignored
- No security character lookup or matching in this mode, only non-secure characters may be programmed (refer to "Table 10 6-Bit ASCII Codes" on page C–34); ignore all associated data in the permissions tag

AP4119 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 the AP4119 tag programmer is configured, keep the permissions tag and/or license certificate available for future reference.

Chapter 3 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 serial command set and describes the security character operation.

Before Programming Tags

Before beginning programming, set up the programmer station as described in "Chapter 1 Programmer Basics" on page 1–7. The programmer must be connected to a host system using a customer-supplied programming software or TransCore AP4119 Rail Tag Programmer (TP) host software, or to a communications terminal. Manually enter the serial control commands if a communications terminal will be used.

Configure the programmer using a permissions tag as described in "Chapter 2 Configuring the Programmer" on page 2–13.

"Chapter 1 Programmer Basics" on page 1–7 provides instructions for installing the TP software. The accompanying AP4119 Rail Tag Programmer Host Software User Guide provides detailed instructions for operating the TP software.

Note: *If the TransCore TP software on the host computer is being used, do not use the serial 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. Use the programmer commands listed in the "Serial Command Set" in this chapter.

The tag programmer is powered by 24 VDC power supply converter that plugs into a standard 110V AC, 50-60 Hz electrical outlet. Refer to "Chapter 1 Programmer Basics" or to Appendix A on page A–32 for additional information.

⚠ CAUTION

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 1 Programmer Basics" on page 1–7 *for recommended practices to protect the tags against ESD damage.*

Indicator Lights

Figure 10 shows the indicator lights used by the programmer during operation.
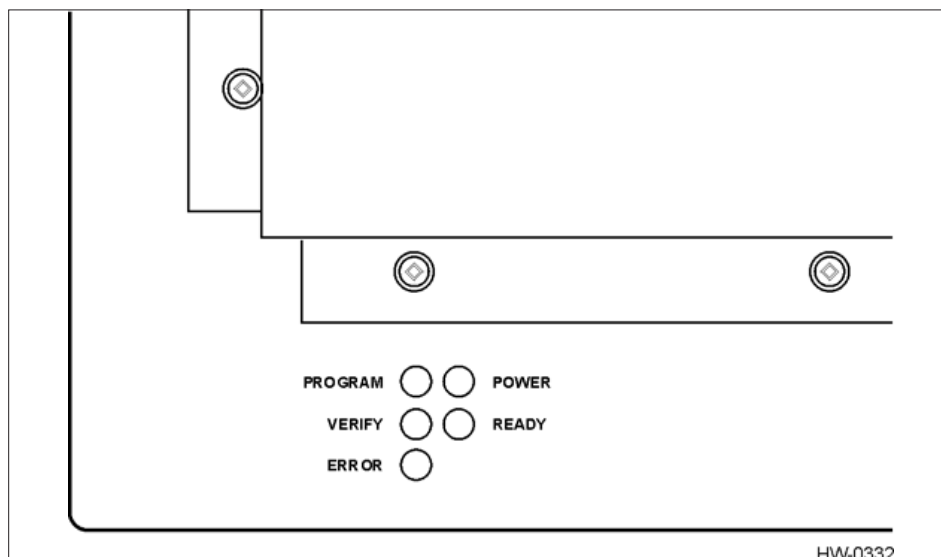
Figure 10 AP4119 Rail Tag Programmer Indicator Lights

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

Expect to see the following sign-on message on the screen of the host system or a communications terminal upon power-up or after sending the programmer command #30:

```
#Model AP4119 Ver X.XX
#Copyright 2022 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:

where

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

`#`	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>
```

Serial Command Set

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

Note: *The tag may be placed into the tag holder before issuing the command, or with the tag holder door open, enter the command and place the tag into the tag holder within 10 seconds. If the tag is not placed in the tag holder and the door is not closed within 10 seconds,the programmer will abort this command. See* "Programming a Tag" on page 3–25 *for tag insertion instructions.*

#120<cr>   Enable Tone Generator

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>`

#121<cr>   Disable Tone Generator

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

> Response:
>
> > `#Done<cr>`

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

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 6-Bit ASCII Format" on page C–34 for a description of this format.

During execution of this command, 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 the tag is not placed in the tag holder and the door is not closed within 10 seconds, the programmer will abort this command. See "Programming a Tag" on page 3–25 for tag insertion instructions.

Response:

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

or

```
#Error<cr>
```

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

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" on page D–35 for a description of this ASCII hex format.

During execution of this command, 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 the tag is not placed in the tag holder and the door is not closed within 10 seconds, the programmer will abort this command. See "Programming a Tag" for tag insertion instructions.

Response:

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

or

```
#Error<cr>
```

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

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, 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 the tag is not placed in the tag holder and the door is not closed within 10 seconds, the programmer will abort this command. See "Programming a Tag" for tag insertion instructions.

Response:

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

or

```
#Error<cr>
```

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

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 for this ASCII hex format.

During execution of this command, 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 the tag is not placed in the tag holder and the door is not closed within 10 seconds, the programmer will abort this command. See "Programming a Tag" for tag insertion instructions.

> Response:

```
#xxxxxxxxxxxxxxxxxxxxxxxxxxxxx<cr> #Done<cr>
```

> or

```
#Error<cr>
```

#220yyxxxx...<cr>   Write a Page of Data Using Hex Format

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 the tag is not placed in the tag holder and the door is not closed within 10 seconds, the programmer will abort this command. See "Programming a Tag" 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>
```

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

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 the tag is not placed in the tag holder and the door is not closed within 10 seconds, the programmer will abort this command. See "Programming a Tag" 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>
```

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

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 the tag is not placed in the tag holder and the door is not closed within 10 seconds, the programmer will abort this command. See "Programming a Tag" 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>
```

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

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 the tag is not placed in the tag holder and the door is not closed within 10 seconds, the programmer will abort this command. See "Programming a Tag" 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>
```

#30<cr>   System Reset

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

Response (system sign-on message):

```
#Model AP4119 Ver X.XX
#Copyright 2022 TransCore
```

where X.XX is the current software version

or

```
#Error<cr>
```

#40<cr>   Display System Status

This command requests a two-digit status response code from the programmer indicating the status of the last operation performed. Refer to Appendix D on page D–35 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.

#42<cr>   Display Tag Door Switch Status

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
```

#43<cr>   Request Control Number

This command requests the AP4119 tag programmer control number.

Response:

```
#xxxxxxxx
```

where

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

#44<cr>   Request Expiration Date

This command requests the tag programmer permissions expiration date.

Response:

```
#MM/DD/YYYY
```

or

```
#None
```

#45<cr>   Request Programming Mode

This command requests the current AP4119 programming mode, based on the permissions tag.

Response:

#AAR = AAR S-918 compatible

#GEN = Generic, non-AAR programming format

Programming a Tag

Note: *The tag may be placed into the tag holder before issuing the command, or with the tag holder door open, enter the command and place the tag into the tag holder within 10 seconds. If an AT5549 End-of-Train Tag is being programmed, place the tag into the programmer adapter (P/N 20-4119-001), with the tag's reflective side up, before sliding the tag into the tag holder. If the tag is not placed in the tag holder and the door is not closed 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 to be program on to the tag. Use command #200 if 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 and close the door (Figure 11), or place the AT5549 End-of-Train Tag reflective side up into programmer adapter, slide into the tag holder, and close the door (Figure 12).
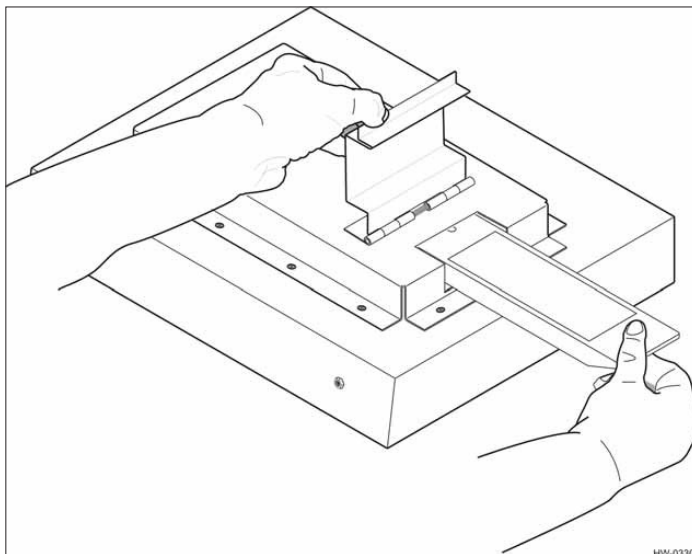
Figure 11 Tag Being Slid into Tag Holder
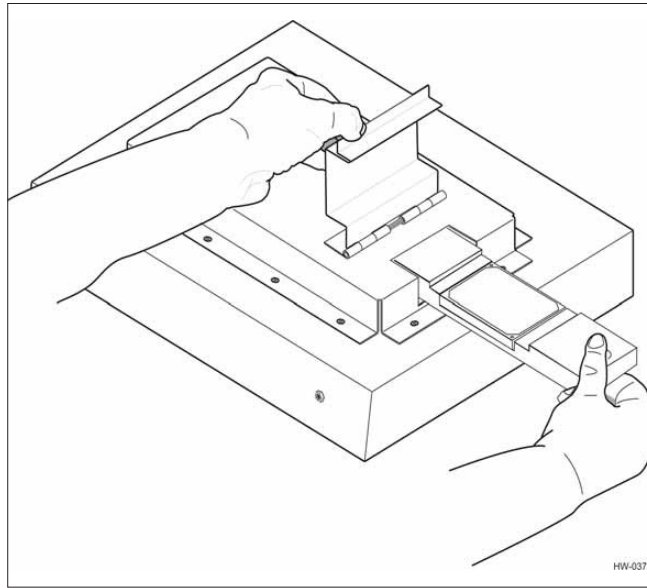
Figure 12 Programmer Adapter and AT5549 EOT Tag Being Slid into Tag Holder

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 this response is received, repeat this sequence beginning at Step 1. Refer to Table 7 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 11 and Figure 12).

If the tag is placed 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 the tag is not slid 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. Refer to Table 7 for troubleshooting remedies.

Security Characters

The tag security character function prevents the use of unauthorized tags in TransCore electronic identification systems. All AP4119 Rail Tag Programmers have the capability, when enabled by settings in the PT5780 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 C and Appendix F.

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 any security codes in a tag 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.

Regulatory Region

⚠ **WARNING**

Warning: ***This equipment can be set to use frequency ranges NOT authorized for use within the US/Canada and thus will NOT comply with FCC/IC frequency allocation requirements at those frequencies.***

Note: *Users in countries other than US/Canada should check with the appropriate local authorities for regulatory requirements.*

The AP4119 Rail Tag Programmer contains a Frequency Hopping Spread Spectrum (FHSS) transceiver module. The transceiver is pre-configured for different regulatory regions throughout the world. The region code is set with command `#642XX`. The `XX` setting is defined in Table 5.

Table 5 List of Regional Codes

Region	Country	XX=
1	United States, Argentina, Canada, Chile, Colombia, Costa Rica, Dominican Republic, Mexico, Panama, Peru, Puerto Rico, Uruguay, Venezuela	01
2	Australia	02
3	Brazil	03
4	China	04
5	Hong Kong	05
6	South Korea	06
7	Malaysia	07
8	New Zealand	08
9	Singapore, Thailand	09
10	South Africa	0A
11	Taiwan	0B
12	Vietnam	0C
13	India	0D
14	Jordan	0E
15	Philippines	0F
16	Indonesia	10
17	Dubai, Qatar	11

The frequency hop set for pre-configured regulatory regions are listed in Table 6.

Table 6 Region Codes and Operating Frequency Hop Set

Region	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Freq. Hop Set in MHz	902.75	920.25	902.25	920.75	920.25	917.2	919.25	921.75	920.25	915.6	922.25	920.25	865.2	865.2	918.25	923.25	868.25
	903.25	920.75	902.75	921	920.75	917.4	919.75	922.25	920.75	915.8	922.75	920.75	865.4	865.4	918.75	923.75	868.5
	903.75	921.25	903.25	921.25	921.25	917.6	920.25	922.75	921.25	916	923.25	921.25	865.6	865.6	919.25	924.25	868.75
	904.25	921.75	903.75	921.5	921.75	917.8	920.75	923.25	921.75	916.2	923.75	921.75	865.8	865.8	919.75	924.75	869
	904.75	922.25	904.25	921.75	922.25	918	921.25	923.75	922.25	916.4	924.25	922.25	866	866			871
	905.25	922.75	904.75	922	922.75	918.2	921.75	924.25	922.75	916.6	924.75	922.75	866.2	866.2			871.25
	905.75	923.25	905.25	922.25	923.25	918.4	922.25	924.75	923.25	916.8	925.25	923.25	866.4	866.4			871.5
	906.25	923.75	905.75	922.5	923.75	918.6	922.75	925.25	923.75	917	925.75	923.75	866.6	866.6			871.75
	906.75	924.25	906.25	922.75	924.25	918.8		925.75	924.25	917.2	926.25	924.25	866.8	866.8			872
	907.25	924.75	906.75	923	924.75	919		926.25	924.75	917.4	926.75	924.75		867			872.25
	907.75	925.25	907.25	923.25		919.2		926.75		917.6	927.25			867.2			872.5
	908.25	925.75	915.25	923.5		919.4		927.25		917.8	927.75			867.4			872.75
	908.75		915.75	923.75		919.6		927.75		918				867.6			873
	909.25		916.25	924		919.8				918.2				867.8			873.25
	909.75		916.75	924.25		920				918.4							873.5
	910.25		917.25			920.2				918.6							873.75
	910.75		917.75			920.4				918.8							874
	911.25		918.25			920.6											874.25
	911.75		918.75														874.5
	912.25		919.25														874.75
	912.75		919.75														875
	913.25		920.25														
	913.75		920.75														
	914.25		921.25														
	914.75		921.75														
	915.25		922.25														
	915.75		922.75														
	916.25		923.25														
	916.75		923.75														
	917.25		924.25														
	917.75		924.75														
	918.25		925.25														
	918.75		925.75														
	919.25		926.25														
	919.75		926.75														
	920.25		927.25														
	920.75																
	921.25																
	921.75																
	922.25																
	922.75																
	923.25																
	923.75																
	924.25																
	924.75																
	925.25																
	925.75																
	926.25																
	926.75																
	927.25																

Warning: Use these commands with extreme caution. It is possible to change the programmer compliance with the regional regulations.

Disclaimer: *If you plan to use the AP4119 in any country other than the United States or Canada, it shall be your sole responsibility to ensure the AP4119 complies with any regulatory requirements applicable to such country, including technical, performance, safety or certification requirements. TransCore is not responsible for any claims, actions, fines, penalties, damages, attorney fees or other injuries resulting from your breach of this section.*

Chapter 4 Troubleshooting

Table 7 lists some problems that may be encountered when using the AP4119 Rail Tag Programmer, along with suggestions for troubleshooting each problem.

If a problem persists, call TransCore Technical Support.

Table 7 Troubleshooting Problems and Solutions

Problem	Suggested Solutions
READY LED does not illuminate at power-on.	Check power transformer. Replace the internal 3-amp, 250-volt Fast-Blo fuse.
System powers on but does not sign on.	Verify RS–232 or USB 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.	Enter command #120 to enable the tone generator. 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 listed in Appendix D to identify the problem. Improve electrostatic discharge protection in programming area. Tag programmer permissions have expired. Enter command #44 to retrieve the tag programmer permissions expiration date. Also, review the expiration date etched into the PT5780 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

This appendix lists the AP4119 Rail Tag Programmer physical and environmental specifications.

The tag programmer is designed to operate in typical office environment conditions. The tag programmer performs to the specifications listed in Table 8 and is ready to program with a warm-up time of not more than three minutes.

Table 8 lists the AP4119 Rail Tag Programmer parameters and specifications.

Table 8 AP4119 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	6.10 lb (2.77 kg)
Input Voltage	24 V DC
Power Consumption	6 W
Excessive Current Protection	3-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	Dust-proof
Operational Vibration	1.04 G_{rms}, 5-500 Hz, power spectral density-uniform 0.0022 G2/Hz, 1 hour per axis
Shock	4 $G_{zero-to-peak}$ by 11 ms half-sine duration in all 3 axes
PC-to-Programmer Cable	RS–232 or Type A USB (data terminal equipment)
FCC Classification	Part 15, site license not required

TransCore Proprietary

Appendix B RS–232 Connector Pin-outs

Table 9 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 9 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 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 10. TransCore's 6-bit ASCII codes are produced by subtracting 00100000 from the character's 8-bit ASCII value.

Table 10 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

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 11.

Table 11 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

Appendix E Programmer Response Codes

The programmer returns one of these codes in response to command #40, Display System Status. Table 12 lists the response codes.

Table 12 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 AP4119 has detected an attempt to write illegal security characters to the security character positions of the tag's ATA frame data. The AP4119 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.

TransCore Proprietary

The codes and responses shown in Table 13 are related to encrypted permission file downloads.

Table 13 Permission File Download Responses

Code	Response
10	Wrong Packet Length
11	Non-Hex Character
12	Invalid Packet Number
13	Invalid Packet Sequence
14	Control Number Mismatch
15	Permission Date Expired
16	Invalid CRC

The codes and responses shown in Table 14 are related to Permissions Tag downloads.

Table 14 Permissions Tag Download Responses

Code	Response
18	Radio Write Fail
19	PTag Page Read Fail
1A	PTag ID CRC Fail
1B	PTag Get Sys Maint Fail
1C	PTag Load ID Sys Maint Fail
1D	PTag Write AES Reg Fail
1E	PTag Read AES Reg Fail
1F	PTag Auth Read Page Fail

Appendix F 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 15.

Table 15 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

Appendix G Commands

This appendix lists the serial command set for the AP4119.

Table 16 Serial Command Set

Number	Command Name	Response
#120<cr>	Enable Tone Generator	#Done<cr>
#121<cr>	Disable Tone Generator	#Done<cr>
#200xxx...<cr>	Program Tag Using TransCore 6-Bit ASCII Format	#xxxxxxxxxxxxxxxxxxxx<cr> #Done<cr> or #Error<cr>
#201xxx...<cr>	Program Tag Using 4-Bit ASCII Hex Format	#xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx<cr> #Done<cr> or #Error<cr>
#210<cr>	Read/Verify Tag Data Using TransCore 6-Bit ASCII Format	#xxxxxxxxxxxxxxxxxxxx<cr> #Done<cr> or #Error<cr>
#211<cr>	Read/Verify Tag Data Using 4-Bit ASCII Hex Format	#xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx<cr> #Done<cr> or #Error<cr>
#220yyxxxx...<cr>	Write a Page of Data Using Hex Format	#xxxxxxxxxxxxxxxx<cr> #Done<cr> or #Error<cr>
#221yyxx<cr>	Write a Byte of Data Using Hex Format	#xx<cr> #Done<cr> or #Error<cr>
#230yy<cr>	Read a Page of Data Using Hex Format	#xxxxxxxxxxxxxxxx<cr> #Done<cr> or #Error<cr>
#231yy<cr>	Read a Byte of Data Using Hex Format	#xx<cr> #Done<cr> or #Error<cr>
#30<cr>	System Reset	#Model AP4119 Ver X.XX #Copyright 2022 TransCore where X.XX is the current software version or #Error<cr>
#40<cr>	Display System Status	#xx<cr> where XX is the two-digit response code. The code 00 indicates that the programmer performed the last operation without error.
#42<cr>	Display Tag Door Switch Status	#00<cr>door open #01<cr>door closed
#43<cr>	Request Control Number	#xxxxxxxx where xxxxxxxx is hex data (control number)
#44<cr>	Request Expiration Date	#MM/DD/YYYY or #None
#45<cr>	Request Programming Mode	#AAR = AAR S-918 compatible #GEN = Generic, non-AAR programming format

For more information:

Sales Support
800.923.4824

Technical Support
505.856.8007

transcore.com

TRANSCORE®
Trusted Transportation Solutions