Application Notes

Monarch® 9855® RFID Printer

```
{F,1,A,R,E,600,400,"9800FMT"¦
T,1,15,V,190,100,1,1,2,2,B,L,0,0¦
C,220,135,0,1,1,1,B,L,0,0,"Ship From:"¦
B,2,12,F,250,130,1,2,75,5,L,0¦
X,3,24,0¦}

{B,1,N,1¦
1,"Dayton, Ohio"¦
2,"12345678901"¦
3,"0123456789ABCDEF12345678"¦}
```

Ship From:
Dayton, Ohio

Ship From:
Dayton, Ohio

Ship From:
Dayton, Ohio

AVERY DENNISON

Each product and program carries a respective written warranty, the only warranty on which the customer can rely. Avery Dennison Corp. reserves the right to make changes in the product, the programs, and their availability at any time and without notice. Although Avery Dennison Corp. has made every effort to provide complete and accurate information in this manual, Avery Dennison Corp. shall not be liable for any omissions or inaccuracies. Any update will be incorporated in a later edition of this manual.

WARNING

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 energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

CANADIAN D.O.C. WARNING

This digital apparatus does not exceed the Class A limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.
Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la classe A prescrites dans le Réglement sur le brouillage radioélectrique édicte par le ministère des Communications du Canada.

Trademarks

Monarch®, 9855®, and Avery Dennison® are trademarks of Avery Dennison Corporation.
EPCglobal, Inc.™ and Electronic Product Code™ (EPC) are trademarks of Uniform Code Council, Inc.
Uniform Code Council, Inc.® is a trademark of Uniform Code Council, Inc.

Avery Dennison Corporation
170 Monarch Lane
Miamisburg, OH 45342

TABLE OF CONTENTS

GETTING STARTED

1

Your Monarch® 9855® RFID (Radio Frequency Identification) printer has been engineered to program (encode) an RFID label (commonly called “RFID tags”) before the label’s format is printed. RFID tags contain an embedded RFID inlay (chip and antenna).

RFID is only available using die cut or black mark supplies; linerless supplies are not supported. The RFID printer is also capable of printing standard (non-RFID) supplies.

The printer supports Class 1 Generation 2 (C1Gen2) supplies. C1Gen2 is a type of transponder that allows read and write capability up to 496 bits (and 64 bits for some models) of data. The printer also accepts EPC data following the guidelines in the *EPCglobal™ Tag Data Standards* Specification, which conforms to the *EPC Radio-Frequency Identity Protocols Class 1 Generation 2 UHF RFID Protocol for Communications at 860-960MHz Standards* (RFID Air Interface protocol).

The RFID supplies are sensitive to static electricity and can be damaged by static electricity. Ground yourself by touching some metal, such as the printer’s metal base, before handling the supplies.

Note: Information in this document supercedes information in previous versions.

Using This Manual

This manual describes the software changes available in version 8.0.

The following table lists the contents of this manual.

Chapter		Contents
1	Getting Started	Information you should know before using the printer.
2	Defining the RFID Data Field	Using Monarch® Printer Control Language II (MPCLII) to create an RFID format. Also includes sample data streams.
3	Troubleshooting	Describes RFID errors that may occur.

Use the *RFID Quick Reference* (provided with your printer) for supply loading and maintenance information.

Refer to the *Monarch® RFID Printer Setup Utility* at http://www.servisource1.com/prnutil/rfidsetup/ for illustrations to determine which type of RFID supplies you are using and basic printer configuration information.

Audience

These *RFID Application Notes* are written for the System Administrator, who is creating formats and encoding (programming) RFID supplies for the 9855 printer.

RFID Terms to Know

Review these terms before you continue.

EPC The Electronic Product Code, which is a numbering standard for items, similar to the UPC code for bar coding. The EPC is divided into several sections: Header, Manager Number, Object Class, and Serial Number. One of the memory fields reserved for EPC programming. This memory is separate from the user memory and the amount of EPC memory varies with the tag types.

Inlay A type of media that contains a transponder and is converted for use in Monarch® RFID supplies (tags). Inlays can be made with different types of transponders.

Interrogator The electronics module that programs the RFID tags through the antenna.

Read Power Increase or decreases the strength of the RF field emitted by the printer's antenna to read the RFID tags.

RFID Data Field The Monarch® Printer Control Language II (MPCLII) data field containing the information to program into an RFID tag.

RF Field Area inside the printer where the RFID tag is programmed. The RF field area is controlled by the RFID power level and the antenna.
Note: The printer's antenna is located between the platen roller and supply guide inside a bracket.

RFID Reader An *optional* device that reads the RFID tags after they are programmed.

RFID Tags Supplies that contain an embedded programmable chip and antenna.

TID The Transponder Identification Number, which contains the chip type, features, and available custom commands supported for tag authentication.

Transponder The combination of the embedded programmable chip with an antenna on some type of media (film, paper, etc.).

User Memory One of the memory fields reserved for user programming. This memory is separate from the EPC memory and the amount of programmable user memory varies with the tag types.

Write Power Increase or decreases the strength of the RF field emitted by the printer's antenna to program the RFID tags.

RFID Considerations

- Printing over the RFID tag (or inlay) may cause printing irregularity.
- You may want to purchase an RFID Reader to verify the RFID tags after printing them.
- Do not use batch separators, which print a pinstripe pattern on a label; or skip index mode, which prints ONE format over multiple labels, since these features unnecessarily waste a label. A batch separator label is different from an overstrike label.

Designing an RFID Format

A format defines which fields appear and where the fields are printed on the label. This section provides some tips about designing formats on large-gap RFID supplies.

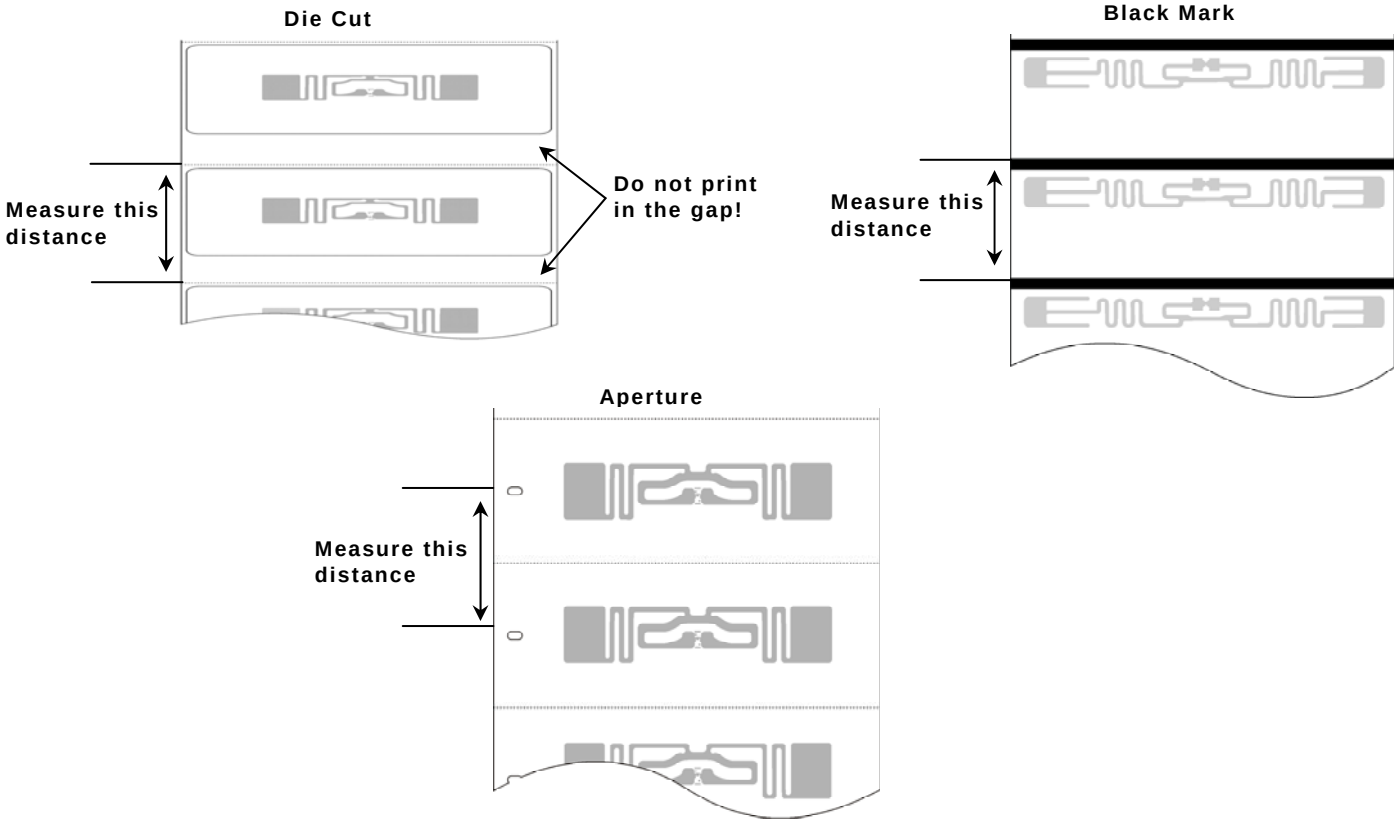
1. Determine your label length. Measure the supply according to the supply type:

   - Die cut supplies: Measure from the leading edge of one label to the leading edge of the next label.

     Note: The leading edge is the edge of the label that exits the printer first; regardless of how the format is designed on the label.

     If your RFID supplies require a supply position setting and corresponding print position setting, you must account for the die cut gap in the format header. To determine the supply/print positions for your inlay, refer to the *Monarch® RFID Printer Setup Utility*.

   - Black mark supplies: Measure from the top of one black mark to the top of the next black mark.
   - Aperture supplies: Measure from the top of one aperture hole to the top of the next aperture.



2. Enter the label length for parameter *f6. length* in the format header. For more information, refer to the *Packet Reference Manual*.

3. Determine your label width. Measure the width of the printable area of your supply. **Do not** include the liner (backing paper) in this dimension.

4. Enter the label width for parameter *f7. width* in the format header. For more information, refer to the *Packet Reference Manual*.

DEFINING THE RFID DATA FIELD

2

The Monarch® Printer Control Language II (MPCLII) RFID Data Field contains the information you want programmed into the RFID tag. The printer can accept EPC data following the guidelines in the *EPC™ Generation 1 Tag Data Standards* Specification or the *EPCglobal™ Tag Data Standards* Specification, which conforms to the *EPC Radio-Frequency Identity Protocols Class 1 Generation 2 UHF RFID Protocol for Communications at 860-960MHz Standards* (RFID Air Interface protocol). Refer to the EPC Specification to create valid EPC data.

This chapter details the syntax of the RFID Data Field and explains how to use Expanded Gen2 data.

The syntax of the RFID Data Field is similar to the standard non-printable text field format.

Syntax **`X,field#,#ofchar,data_type |`**

X1. *X* — RFID Data Field.

X2. *field#* — Unique number from **0-999** assigned to this field.

X3. *#ofchar* — This number must be equal to or greater than the total number of characters in the RFID Data Field. Range: **0-2710**. The amount varies according to your RFID data, protocol, and module type. For example, C1Gen2 can be up to 496 bits, **plus** the access password, lock code, etc.

Note: An error 715 occurs if the printer does not receive the correct amount of data. You can pad data to make sure you have the correct amount of bits. See "Using Option 30," for more information.

X4. *data_type* — Data type. Use **0** (ASCII Hex - ASCII representation of Hex), which uses characters **A** to **F** and **0** to **9**. For example, the letter A in ASCII Hex has a value of 41 and is represented in MPCL batch data as "414141414141414141414141".

Note: Do not use the ASCII file separator character (1C) in RFID batch data.

Example **`X,5,24,0 |`**

Defines an RFID Data Field (field #5) with exactly 24 ASCII Hex characters for a 96-bit RFID tag.

Sample Format

```
{F,1,A,R,E,600,400,"RDCI" |
L,V,500,115,90,85,3 |
L,V,298,245,90,102,3 |
C,568,8,0,2,2,2,B,L,0,0,"FROM:",0 |
C,568,125,0,2,2,2,B,L,0,0,"CARRIER:",0 |
C,387,8,0,2,1,1,B,L,0,0,"(420) SHIP TO POSTAL CODE",0 |
C,391,250,0,2,1,1,B,L,0,0,"APPOINTMENT NUMBER:",0 |
C,327,250,0,2,1,1,B,L,0,0,"ITEM:",0 |
C,190,8,0,2,1,1,B,L,0,0,"UPC SHIPPING CONTAINER CODE",0 |
T,1,15,V,529,220,0,2,2,2,B,L,0,0,0 |
B,3,13,V,311,28,8,4,50,8,L,0 |
B,4,14,V,17,60,50,5,130,8,L,0 |
X,5,24,0 |<-------------------------- RFID Data Field
T,6,20,V,415,270,0,50,15,15,B,L,0,2 |
R,4,5,1,16,1,0 |<---------------------- Copy Option from field 5 (RFID Data Field) to
…|}                                    field 6 (Text field)
```

Sample Batch Data

```
{B,1,N,1 |
1,"RFID TEST" |
3,"1005678" |
4,"67-90-32" |
5,"3123456789ABCDEF12345678" |
6,"ABCDEFG" |
…|}
```

RFID Data Field Batch Data in ASCII Hex

The RFID Data Field contains exactly 24 characters of data in ASCII Hex format for a 96-bit RFID tag.

Applying Options to the RFID Data Field

All the normal field options (copy, merge, pad, increment, etc.) can be applied to the RFID Data Field. However, certain restrictions may apply.

- When incrementing the RFID Data Field, ASCII Hex increments 0 to F (0123456789ABCDEF), then back to 0.
- When incrementing the RFID Data Field, the only valid value to increment is by **one**.
- If Option 60 (Increment Field) contains the character “D” to decrement, it is ignored and the field is instead incremented by one.
- Use caution when incrementing an RFID field if data is coming directly from your host because the field must be incremented in ASCII Hex.

Incrementing EPC Data

Using version 7.1 or greater software, the printer increments EPC data.

When incrementing EPC data, ASCII Hex increments: 0 to F (0123456789ABCDEF), then back to 0.

When incrementing EPC Data, the only valid value to change is by **one**.

Incrementing EPC Data Example

```
{F,1,A,R,E,400,400,""|
T,1,30,V,50,70,0,1,1,1,B,L,0,0|
R,60,I,1|
B,2,12,F,10,130,1,4,50,8,L,0|
R,60,I,1|
X,3,100,0|
R,60,I,1| }

{B,1,N,3|
1,"001"|
2,"123456789012"|
3,"123456781234567812345 61B~028~028~02812345678~02812345678~02800000~028"| }
```

RFID Data Field Incrementing Option

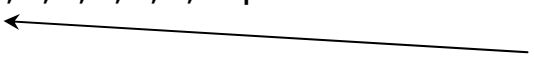
RFID Data Field Batch Data in ASCII Hex with EPC Data included

This example uses option 60 to increment the EPC RFID data by one.

Copy Option Example

```
{F,2,A,R,E,400,400,"ASCIIHEX" |
X,2,24,0 |
T,1,50,V,10,10,0,1,1,1,B,L,0,0 |
R,4,2,1,16,1,2 |}

{B,2,N,2 |
1,"313233343536373831323334" |
2,"313233343536373831323334" |}
```



Copy data from RFID Data field to text field

This example uses option 4 to copy data from the RFID Data Field and displays the data in text field 1. Note the data type is ASCII Hex, so the data in the RFID Data Field is in ASCII Hex format. This example uses a 96-bit RFID tag.

Using Option 5 (Define Data Entry Sources)

Use this option to read pre-programmed data in the RFID chip embedded within the supply. Using Option 5 stops the printer while reading each label; regardless of the print speed. If using a batch quantity greater than one, the data is read from each label.

Note: Using Option 5 to read pre-programmed RFID data is supported with the release of version 5.0 or greater printer's software.
The EM4122 protocol requires Option 5 to read the pre-programmed data.

When reading data, make sure the maximum number of characters in the field is equal to or greater than the number of characters being read. If not, the data may be incomplete.

Use option 6 (Upload Field Data) with Option 5 to upload the data from the RFID chip to a host.

Syntax **R,5,code |**

R1. *R* — Option Header.

R2. 5 — Option 5.

R3. *code* — Input code for the data in the field. Options:

- **H** Host
- **K** Keypad
- **N** No user input for this field
- **R** RFID (read data from the RFID chip). This is ignored on non-RFID printers.

Note: Option 5 re-images each label in the batch.

Example

```
T,2,10,V,250,50,0,1,1,1,B,C,0,0,0 |
R,5,R |
```

Reads the pre-programmed data from the RFID chip and saves that data into the text field.

Using Option 6 (Upload Field Data)

You can upload data from any field using Option 6. When uploading multiple fields of data, the data is comma separated.

Data is uploaded to the last port that received host data (serial, parallel, USB, or Ethernet) at the end of the batch or label, depending on the other options used as follows:

- When using a batch quantity of one without Option 5 or Option 60, data is uploaded at the end of the batch.
- When using a batch quantity greater than one with an Option 5 and/or Option 60, data is uploaded after each label.

Note: To upload data on a field that did not change, apply Option 60 to that field.

Syntax **R,6,device |**

R1. *R*	Option Header.
R2. *6*	Option 6.
R3. *device*	Last port that received host data. Use **H** (host).

Example
```
T,2,10,V,250,50,0,1,1,1,B,C,0,0,0 |
R,6,H |
```

Uploads the text field's data to a file.

Example
```
B,3,12,F,50,50,1,2,60,7,L,0 |
R,6,H |
R,60,I,0 |
```

Uploads the UPCA bar code field's data to a file and uploads data for each label in the batch.

Example
```
T,150,V,230,130,0,1,1,1,B,L,0,0 |
R,5,R |
R,6,H |
```

Reads the RFID data from the RFID chip embedded in the supply. Uploads the data to the last-used port.

Example
```
{F,1,A,R,E,600,400,"RDCI" |
…
X,5,24,0 |
T,6,20,V,415,270,0,50,15,15,B,L,0,2 |
R,4,5,1,16,1,0 |
R,6,H |
…|}
```

Copies data from field 5 (RFID Data Field) to field 6 (Text field). Uploads the data to the last-used port.

Sample Upload Packet

Example **R,5,R |**
R,6,H |

Returns the following in the upload packet:

313233343536373839303132

Pre-programmed data in the RFID chip.

Example **B,3,12,F,50,50,1,2,60,7,L,0 |**
R,6,H |

Returns the following in the upload packet:

123456789012

UPCA bar code data entered from the batch.

Example **R,5,R |**
R,6,H |
B,3,12,F,50,50,1,2,60,7,L,0 |
R,6,H |

Returns the following in the upload packet:

313233343536373839303132,123456789012

Pre-programmed data in the RFID chip and the UPCA bar code data entered from the batch.

Using Option 30 (Pad Data)

You can add characters to one side of a field to "pad" the field. Padding allows you to fill in the remaining spaces when the entered data does not fill an entire field.

If a variable length field is not completely filled with batch data, this option fills the remaining positions in the field with the character designated by Option 30.

Syntax **R,30,L/R,"character" |**

R1. *R*	Option Header.
R2. *30*	Option 30.
R3. *L/R*	Indicates type of padding **L** Pad field on left side **R** Pad field on right side
R4. *"character"*	Pad character must be within the **0** - **255** decimal range and enclosed inside quotation marks.

Note: The pad character must be in the same format specified in the RFID Data Field. See "Defining the RFID Data Field" for more information.

Example **R,30,L,"A" |**

Pads data with an "A" on the left side of the field.

Example

```
X,2,24,0 |
R,30,R,"0" |
```

Pads the data in the RFID Data Field with a "0" on the right side of the data. This example uses a 96-bit RFID tag.

Using Option 64 (Program AFI Field for UHF RFID)

Use Option 64 to program the AFI memory in the EPC memory bank (field) for UHF RFID tags.

Note: Do not apply a lock to the EPC memory bank since the AFI memory is part of the EPC memory bank. If the EPC memory bank is locked, the AFI memory cannot be programmed (the printer errors).

Syntax **R,64,field,"code",lock |**

R1. R	Option Header.
R2. 64	Option **64**.
R3. field	Memory field to program. Use **A** (AFI).
R4. code	Byte code. This is an 8-bit (1 byte) character to program, enclosed within quotation marks. Range: **0 - 255**. Use the ASCII character or the tilde followed by the equivalent three-character decimal value. For example, the letter A can be entered as A or ~041.
R5. lock	Lock code. Use **0** (no lock).

Example **X,5,16,0,0 |**
R,64,A,"A",0 |

Programs the character A into the AFI memory field. The data is not locked (0).

Example **X,5,16,0,0 |**
R,64,A,"~046",0 |

Programs the character F into the AFI memory field. The data is not locked (0).

Using Expanded Gen2 Data

With version 5.0 or greater software, we support Expanded C1Gen2 data, which is composed of five different fields:

- EPC Data
- User Memory
- Access Password
- Kill Password
- Lock Code

The printer also accepts EPC data following the guidelines in the *EPCglobal™ Tag Data Standards* Specification, which conforms to the *EPC Radio-Frequency Identity Protocols Class 1 Generation 2 UHF RFID Protocol for Communications at 860-960MHz Standards* (RFID Air Interface protocol). Refer to the EPC Specification to create valid EPC data.

Many different RFID supplies (tags) are available and the amount of programmable user memory varies with the chip embedded in the tag. Depending on your tag type, all memory fields may not be available. Refer to the *Monarch® RFID Printer Setup Utility* for details about the available user memory for each chip.

Use the access password to control when new data can be written to a field. The kill password sets a tag up to be inoperable. The lock code contains the locking method.

One of the four locking methods can be selected for each memory field (EPC, user memory, access password, and kill password). Depending on the locking method specified, the memory field may or may not be readable or writable.

There are four locking methods.

Value	EPC Lock Name	Description
0	No lock (unsecure)	The selected memory fields (EPC, user memory, access password and kill password) are readable and writable. The tag can be programmed multiple times.
1	Permalock (permanently unsecure)	Permanently locked in a readable and writable state. The tag can be programmed multiple times.
2	Password lock (secure)	Requires the access password to rewrite the selected memory fields. The tag can be programmed multiple times *with the password*.
3	Permalock and Password lock (permanently locked)	Never rewritable, but always readable. The tag can never be rewritten, once locked.

Note: The EPC field is always readable, no matter what locking method is assigned.

The following table describes the locking method for each memory field.

EPC Memory	Description
0	EPC is readable and writable.
1	EPC is permanently writable (can never be locked).
2	EPC is only writable with password, but is readable.
3	EPC is never rewritable, but is readable.
User Memory	**Description**
0	User memory is readable and writable.
1	User memory is permanently writable (can never be locked), but is readable.
2	User memory is only writable with password, but is readable.
3	User memory is never rewritable, but is readable.
Access Password	**Description**
0	Access password is readable and writable.
1	Access password is permanently writable (can never be locked).
2	Access password is never readable.
3	Access password is never readable or rewritable.
Kill Password	**Description**
0	Kill password is readable and writable.
1	Kill password is permanently writable (never locked), but is readable.
2	Kill password is only writable with password.
3	Kill password is never readable or rewritable.

To use the Expanded C1Gen2 data, you need to modify the RFID Field's batch data.

Syntax **field#,"EPC_data~028" |**
C,"User_Mem~028" |
C,"~028" |
C,"Acs_Pwd~028" |
C,"Kill_Pwd~028" |
C,"Lock_Code" |

field# — Identifies the RFID Data Field number for the following data.
Range: **1** - **999**.

"EPC_data~028" — EPC data. Enclose in quotation marks. To create EPC data, follow the guidelines in the *EPC Global Generation 1 Tag Data Standards Specification*. The printer can program up to 496-bits of data (124 ASCII Hex characters). The data must end with the ASCII field separator (decimal 028).

Note: Do not enter more characters than the available EPC data. For example, if the tag has 96-bits EPC data available, do not enter more than 96-bits of data for this field.

C,"User_Mem~028" — User Memory. Enclose in quotation marks. The amount of programmable user memory varies with the tag types. Refer to the *Monarch® RFID Printer Setup Utility* for details about the available user memory for each chip.
The printer can program up to 512-bits of data (128 ASCII Hex characters). The data must be ASCII Hex characters and end with the ASCII field separator (decimal 028).

C,"~028" — Identifies information to be appended. Reserved for future use. Only include the field separator (decimal 028) in this field.

C,"Acs_Pwd~028" — Access Password. This must be 8 ASCII Hex characters. No password is assigned if this field is left blank. The data must end with the ASCII field separator (decimal 028).

C,"Kill_Pwd~028" — Kill Password. This must be 8 ASCII Hex characters. No password is assigned if this field is left blank. The data must end with the ASCII field separator (decimal 028).

C,"Lock_Code" — Five-digit locking method for each field in this order:
EPC Data, User Memory, Reserved, Access password, and Kill password.
Use 0 for the reserved field.
Locking options:

- **0** No Lock
- **1** Permalock
- **2** Password lock
- **3** Permalock & password lock

Notes: Use only one locking method per field.
The printer is not capable of unlocking a field. Depending on the locking method used for each field, the EPC data *may* be programmable by sending the access password with the batch data. See the following examples.

Permalock Example

```
{F,1,A,R,E,400,400,"PERMLOCK" |
X,1,100,0 |}                          RFID Data Field

{B,1,N,1 |
1,"313233343536373831323334~028" |    EPC Data
C,"ABCDEF~028" |                      User Memory
C,"~028" |                            Reserved
C,"73737373~028" |                    Access Password
C,"CAD01234~028" |                    Kill Password
C,"11001" |}                          Lock Method for each field:
```

Parameter	Description	Data
One	EPC	1
Two	User Memory	1
Three	Reserved	0
Four	Access Password	0
Five	Kill Password	1

The EPC data is 313233343536373831323334, the user memory data is ABCDEF, the access password is 73737373, and the kill password is CAD01234. Selects permalock (1) as the locking method for the EPC, user memory, and kill field. Selects no lock (0) for the access password field. The EPC, user memory, and kill fields are **permanently** readable/writable.

Password Lock Example

```
{F,1,A,R,E,400,400,"PWDLOCK" |
X,1,100,0 |}                                    RFID Data Field

{B,1,N,1 |
1,"313233343536373831323334~028" |              EPC Data
C,"ABCDEFGH01234567UserMemorySample~028" |      User Memory
C,"~028" |                                      Reserved
C,"73737373~028" |                              Access Password
C,"CAD01234~028" |                              Kill Password
C,"22022" |}                                    Lock Method for each field
```

The EPC data is 313233343536373831323334, the user memory data is ABCDEFGH01234567UserMemorySample, the access password is 73737373, and the kill password is CAD01234. Selects password lock (2) as the locking method for the EPC, user memory, access, and kill fields. To change the EPC, user memory, or kill fields, the *access password must be sent* with the batch data.

Both Lock Example

```
{F,1,A,R,E,400,400,"BOTHLOCK" |
X,1,100,0 |}                                    RFID Data Field

{B,1,N,1 |
1,"31323334353637383132333 4~028" |             EPC Data
C,"ABCDEF ~028" |                               User Memory
C,"~028" |                                      Reserved
C,"73737373~028" |                              Access Password
C,"CAD01234~028" |                              Kill Password
C,"33033" |}                                    Lock Method for each field
```

The EPC data is 313233343536373831323334, the user memory data is ABCDEF, the access password is 73737373, and the kill password is CAD01234. Selects permalock and password lock (3) as the locking method for the EPC, user memory, access, and kill fields, which means these fields are never rewritable. The EPC data can only be programmed ONE time.

96-Bit RFID Data Field Examples

The following examples can be used for C1Gen2 (without locking). See “Using Expanded Gen2 Data” for more information about using the locking feature.

The 96-bit data that can be displayed using the printer’s Setup, RFID Menu for all these examples is the same: “313233343536373839303132.”

ASCII HEX Example

```
{F,2,A,R,E,400,400,"ASCIIHEX" |
T,1,50,V,10,10,0,1,1,1,B,L,0,0 |
X,2,24,0 |}

{B,2,N,1 |
1,"313233343536373839303132" |
2,"313233343536373839303132" |}
```

ASCII Hex data type is selected in the RFID Data Field and the batch data is entered as ASCII Hex.

SSCC96 Example

```
{F,45,A,R,E,600,400,"SSCC96" |
C,45,220,0,50,10,10,B,L,0,2,"®" |
C,75,385,0,50,40,30,B,L,0,2,"Monarch RFID" |
C,110,385,0,50,6,6,B,L,0,2,"SHIP TO RFID USER" |
C,110,150,0,50,6,6,B,L,0,2,"CARRIER" |
C,150,150,0,50,9,9,B,L,0,2,"PRO:" |
C,165,150,0,50,9,9,B,L,0,2,"B/L:" |
C,200,380,0,50,18,15,B,L,0,2,"AVERY DENNISON" |
C,235,380,0,50,14,12,B,L,0,2,"EMAIL: RFID@MONARCH.COM" |
C,255,380,0,50,14,12,B,L,0,2,"PHONE: 1 800 543-6650" |
C,275,345,0,50,6,6,B,L,0,2,"VENDOR STK NO:" |
C,275,110,0,50,6,6,B,L,0,2,"PACK/UNITS:" |
C,360,330,0,50,10,10,B,L,0,2,"EPC Pure Identity:" |
C,275,190,0,50,6,6,B,L,0,2,"COLOR:" |
C,315,110,0,50,6,6,B,L,0,2,"SIZE/STYLE:" |
C,415,390,0,50,12,12,B,L,0,2,"EPC#:" |
L,S,098,005,098,395,6,"" |
L,S,170,005,170,395,6,"" |
L,S,098,155,170,155,6,"" |
L,S,260,005,260,395,6,"" |
L,S,390,005,390,395,6,"" |
T,1,15,V,130,335,0,50,13,12,B,L,0,2 |
B,2,15,V,165,365,8,4,25,0,L,2 |
T,3,20,V,130,150,0,50,13,12,B,L,0,2 |
T,4,20,V,150,110,0,50,10,12,B,L,0,2 |
T,5,20,V,165,110,0,50,10,12,B,L,0,2 |
T,12,10,V,295,370,0,50,15,18,B,L,0,2 |
T,13,10,V,295,110,0,50,15,18,B,L,0,2 |

T,14,50,V,380,385,0,50,12,11,B,L,0,2 |   <-- Text Field with EPC Data
T,15,10,V,295,200,0,50,15,18,B,L,0,2 |
T,16,10,V,335,155,0,50,15,18,B,L,0,2 |
T,18,30,V,550,330,0,50,15,18,B,L,0,2 |
B,17,20,V,530,340,50,6,110,0,L,2 |

X,19,24 |   <-- RFID Data Field

T,20,40,V,415,330,0,50,12,13,B,L,0,2 |
R,4,19,1,40,1,1 |}
```

Batch Data

```
{B,45,N,1 |
1,"VENDOR USA" |
2,"42060512" |
3,"AVERY DENNISON" |
4,"0987764356" |
5,"0020545640" |
12,"0075687332" |
13,"3600" |

14,"urn:epc:tag:sscc-96:1.0028028.0000001235" |
15,"RED" |   <-- Text Field Batch Data
16,"48~"/TOUGH" |
17,"009280287586887" |
18,"0  09  28028  75688  7" |

19,"313401B5F0000004D3000000" |   <-- RFID Field Batch Data
20,"0" |}
```

SGTIN 96 Example

```
{F,46,A,R,E,600,400,"SGTIN96" |
C,45,220,0,50,10,10,B,L,0,2,"®" |
C,75,385,0,50,40,30,B,L,0,2,"Monarch RFID" |
C,110,385,0,50,6,6,B,L,0,2,"SHIP TO RFID USER" |
C,110,150,0,50,6,6,B,L,0,2,"CARRIER" |
C,150,150,0,50,9,9,B,L,0,2,"PRO:" |
C,165,150,0,50,9,9,B,L,0,2,"B/L:" |
C,200,380,0,50,18,15,B,L,0,2,"AVERY DENNISON" |
C,235,380,0,50,14,12,B,L,0,2,"EMAIL: RFID@MONARCH.COM" |
C,255,380,0,50,14,12,B,L,0,2,"PHONE: 1 800 543-6650" |
C,275,345,0,50,6,6,B,L,0,2,"VENDOR STK NO:" |
C,275,110,0,50,6,6,B,L,0,2,"PACK/UNITS:" |
C,360,330,0,50,10,10,B,L,0,2,"EPC Pure Identity:" |
C,275,190,0,50,6,6,B,L,0,2,"COLOR:" |
C,315,110,0,50,6,6,B,L,0,2,"SIZE/STYLE:" |
C,415,390,0,50,12,12,B,L,0,2,"EPC#:" |
L,S,098,005,098,395,6,"" |
L,S,170,005,170,395,6,"" |
L,S,098,155,170,155,6,"" |
L,S,260,005,260,395,6,"" |
L,S,390,005,390,395,6,"" |
T,1,15,V,130,335,0,50,13,12,B,L,0,2 |
B,2,15,V,165,365,8,4,25,0,L,2 |
T,3,20,V,130,150,0,50,13,12,B,L,0,2 |
T,4,20,V,150,110,0,50,10,12,B,L,0,2 |
T,5,20,V,165,110,0,50,10,12,B,L,0,2 |
T,12,10,V,295,370,0,50,15,18,B,L,0,2 |
T,13,10,V,295,110,0,50,15,18,B,L,0,2 |

T,14,50,V,380,385,0,50,12,11,B,L,0,2 |   <-- Text Field with EPC Data
T,15,10,V,295,200,0,50,15,18,B,L,0,2 |
T,16,10,V,335,155,0,50,15,18,B,L,0,2 |
T,18,30,V,550,330,0,50,15,18,B,L,0,2 |
B,17,20,V,530,340,50,6,110,0,L,2 |

X,19,24 |   <-- RFID Data Field

T,20,40,V,415,330,0,50,12,13,B,L,0,2 |
R,4,19,1,40,1,1 |}
```

Batch Data

```
{B,46,N,1 |
1,"VENDOR USA" |
2,"42060512" |
3,"AVERY DENNISON" |
4,"0987764356" |
5,"0020545640" |
12,"0075687332" |
13,"3600" |

14,"urn:epc:tag:sgtin-96:1.0028028.001234.2" |
15,"RED" |                                          <-- Text Field Batch Data
16,"48~"/TOUGH" |
17,"009280287586887" |
18,"0  09  28028  75688  7" |

19,"303401B5F001348000000002" |   <-- RFID Field Batch Data
20,"0" |}
```

Defining the RFID Setup Packet

Use the RFID setup packet (X) to set the protocol, write attempts, read power, write power, signal adjust, verify the programmed data, retry printing, and check for multiple RFID inlays in the printer's programmable RF field. Refer to the *Monarch® RFID Printer Setup Utility* for the read and write power settings based on your RFID supplies.

Syntax **{I,X,protocol,write_attem,read_power,write_power, signal_adjust,verify_write,retry_print,RF_inly_pos,singulate, mult_tg_ck | }**

X1. *X* — RFID Setup Packet.

X2. *protocol* — Select the UHF protocol. Use **0**.

X3. *write_attem* — The number of times the interrogator tries to program the RFID tag in the RF Field. Use **3**.

X4. *read_power* — Not used – enter **-7**.

X5. *write_power* — The amount of power to write (program) an RFID tag. The range is **-7** to **27** and the default is **-7**.

X6. *signal_adj* — The strength of the RF Field emitted by the printer's antenna. Use **2**.

X7. *verify_write* — The printer confirms the read value matches the programmed value after writing the RFID data. Options:

0 Disabled (default). The printer does not verify the programmed value.
1 Enabled. The printer verifies the programmed value.

X8. *retry_print* — The printer tries to reprint and program a label where a supply error occurred. The printer uses the same EPC data from the errored label on the next label. Options:

0 Enabled (default). The printer reprints and re-programs the EPC data from a label when a supply error occurs.
1 Disabled. The printer does not reprint and re-program the EPC data.

Note: *Retry_print* must be used with incrementing batches. If disabled, the printer does not reprint a label using the EPC data from the label with a supply error. For example, printing a batch of 100 labels prints 99 labels (or less) if a supply error occurs.

If multiple 728 errors occur, disable *retry_print*.

X9. *RF_inly_pos* — Not used – enter **0**.

X10. *singulate* — Not used – enter **0**.

X11. *mult_tg_ck* — Before printing, the printer checks for more than one RFID tag within the programmable RF Field's range. Options:

0 Check the first tag (default). The printer checks for multiple RFID tags in the field.
1 Check each tag. The printer checks for multiple RFID tags in the field before each tag in the batch.

Example **{I,X,0,3,10,10,2,1,0,0,0,1 | }**

Uses the default protocol (Class 1 Gen 2) and write attempts (3), sets the read and write power to 10, uses the default signal adjustment (2), the printer verifies the read value matches the programmed value, does not reprint a label if a supply error occurs, and checks for multiple RFID tags in the programmable field before each tag is printed.

TROUBLESHOOTING 3

Use this chapter as a reference for any RFID errors you may receive.

Problem	Action
Printer displays “Please Wait RFID Initialization.”	You sent an RFID batch before the printer communicated with the interrogator. The printer prints the RFID batch once communication with the interrogator is complete.
Printer displays “RFID Detection Not Complete.”	You tried to access the Setup, RFID Menu before the printer communicated with the interrogator. Press **Escape/Clear** until you see “Print Mode Ready.” Wait several seconds and then try to access the Setup, RFID Menu again.
Printer displays “Monarch Initializing.”	This message should flash briefly on the display when you turn on the RFID printer. If it does not disappear, turn off the printer; wait 15 seconds and then turn on the printer.
Printer does not read or program the RFID tag.	Make sure the following are set correctly for your inlay (RFID tag) type: Protocol (C1Gen2, EM4122, etc.), Read Value, Write Value, and Signal Adjust. An RFID module with hardware version 00000003, does not support the protocol, write attempts, read power or signal adjust. Refer to the *Monarch® RFID Printer Setup Utility* for more information. The EM4122 protocol requires Option 5 to read the pre-programmed RFID data. See “Using Option 5 (Define Data Entry Sources)” for more information.
No data provided for a UPC/EAN bar code.	The printer does not error if no batch data is sent for the bar code field. The bar code field is not printed.

RFID Errors

The printer does not recalibrate (feed a blank label) after any RFID error. Call Technical Support for any message not listed.

052 Data type in the RFID Data Field must be 0, 1, 2, or 3. See “Defining the RFID Data Field” for more information.

226 Rule Record Line xx. Upload device must be H (Host) for Option 6.

296 Protocol must be 0.

297 Write Attempts must be 3.

298 Read Power must be -7 to 27. See “Defining the RFID Setup Packet” for more information.

299 Write Power must be -7 to 27. See "Defining the RFID Setup Packet" for more information.

300 Signal Adjust must be 2. See "Defining the RFID Setup Packet" for more information.

307 Verify Write must be 0 (Disabled) or 1 (Enabled). See "Defining the RFID Setup Packet" for more information.

308 Retry Printing must be 0 (Disabled) or 1 (Enabled). See "Defining the RFID Setup Packet" for more information.

309 Singulate mode is invalid. Use **0**.

312 Multi Tag Check must be 0 (Check the first tag) or 1 (Check each tag). See "Defining the RFID Setup Packet" for more information.

313 Inlay Position is invalid. Use **0**.

612 Invalid Gen2 RFID Data length. See "Using Expanded Gen2 Data" for more information.

715 Invalid data length/data mismatch. The data in the RFID Data Field has an incorrect data length or there is a data type mismatch between selected data type and actual data entered. See "Defining the RFID Data Field" for more information on the data length and for selecting the appropriate data type for the data being entered. This error also occurs if any of the Expanded C1Gen2 fields are invalid. Check with your System Administrator about your format.

727 Duplicate TID fields. The printer detected two inlays within the programmable range with the same TID. One inlay is defective and another inlay may have the wrong data encoded – do not use the label that has an overstrike pattern. Check the Write power setting.

728 Multiple Tag Error. The printer detected multiple inlays within the programmable range. Re-check values from the *Monarch® RFID Printer Setup Utility*.

729 Invalid Tag Type. The chip (inlay) embedded in your RFID supply does not support serialized TID.

730 A memory allocation error occurred during initialization.

731 Backfeed error with a positive cut position defined for RFID. The tag is cut too short and cannot backfeed to the correct inlay encoding position. Check or modify the backfeed and cut position settings.

732 RFID Hardware Error. A non-RFID printer received a format containing an RFID Data Field. Resend the format to an RFID printer.

733 The RFID tag is not in the programmable field with short-feed length RFID supplies. Check supply loading. If you changed the supply, print or RFID position, make sure the RFID tag was not moved out of the programmable range. Refer to the *Monarch® RFID Printer Setup Utility* for more information.

734 The RFID tag is not programmed before the end of the label is reached. The program cycle fails. Resend the batch to the printer.

735 RFID module hardware error. The RFID module may need to be replaced or has become disconnected. Call Technical Support.

736 Invalid RFID command sent to the RFID module. The tag command is invalid.

737 Invalid RFID reader command sent to the RFID module. The reader setup command is invalid.

738 Insufficient RFID power to read the TID field or program the RFID tag. There is a reader or tag power level problem.

739 Missing specific tag. The printer cannot find the RFID tag identified by singulation. Check supply loading or the RFID tag may be lost or defective.

740 Command, hardware, inventory, or memory allocation error. There may be an RFID hardware or memory allocation error.

741 RFID tag missing. Tag not found in RF Field (area inside the printer where RFID tag is programmed). Check supply loading. If you made supply or print position settings, make sure the RFID tag was not moved out of the programmable range. Re-check values from the *Monarch® RFID Printer Setup Utility*.

742 Tag erase failed. The RFID tag was found in the RF Field, but could not be erased. If this error occurs consecutively, increase or decrease the write power setting by one. If the write power is set too high, you may affect the data programmed in adjacent labels.

743 Program tag failed. The RFID tag was found in the RF Field, but could not be programmed. If this error occurs consecutively, increase or decrease the write power setting by one. Re-check values from the *Monarch® RFID Printer Setup Utility*.

Note: If the Write Power is set too high, you may affect the data programmed in adjacent labels.

744 Tag locked fail. The RFID tag is unable to be programmed, because it is already locked. This is considered a bad RFID tag.

746 Lock tag fail. The RFID tag has not been locked to prevent reprogramming. This is considered a bad RFID tag.

747 Time out failure. Any RFID command (read, program, etc.) has failed to complete in the maximum amount of allowed time.

748 Invalid data length/data mismatch from RFID interrogator module. The RFID interrogator module found a problem with the data received from the printer. You may need a new RFID printer or RFID module.

749 RFID Verify Fail. The RFID verification process failed after writing (programming) the tag. This is considered a bad RFID tag.

INDEX

028028

Avery Dennison
170 Monarch Lane
Miamisburg, OH 45342

1-800-543-6650 (In the U.S.A.)
1-800-387-4740 (In Canada)
www.monarch.com

AVERY DENNISON