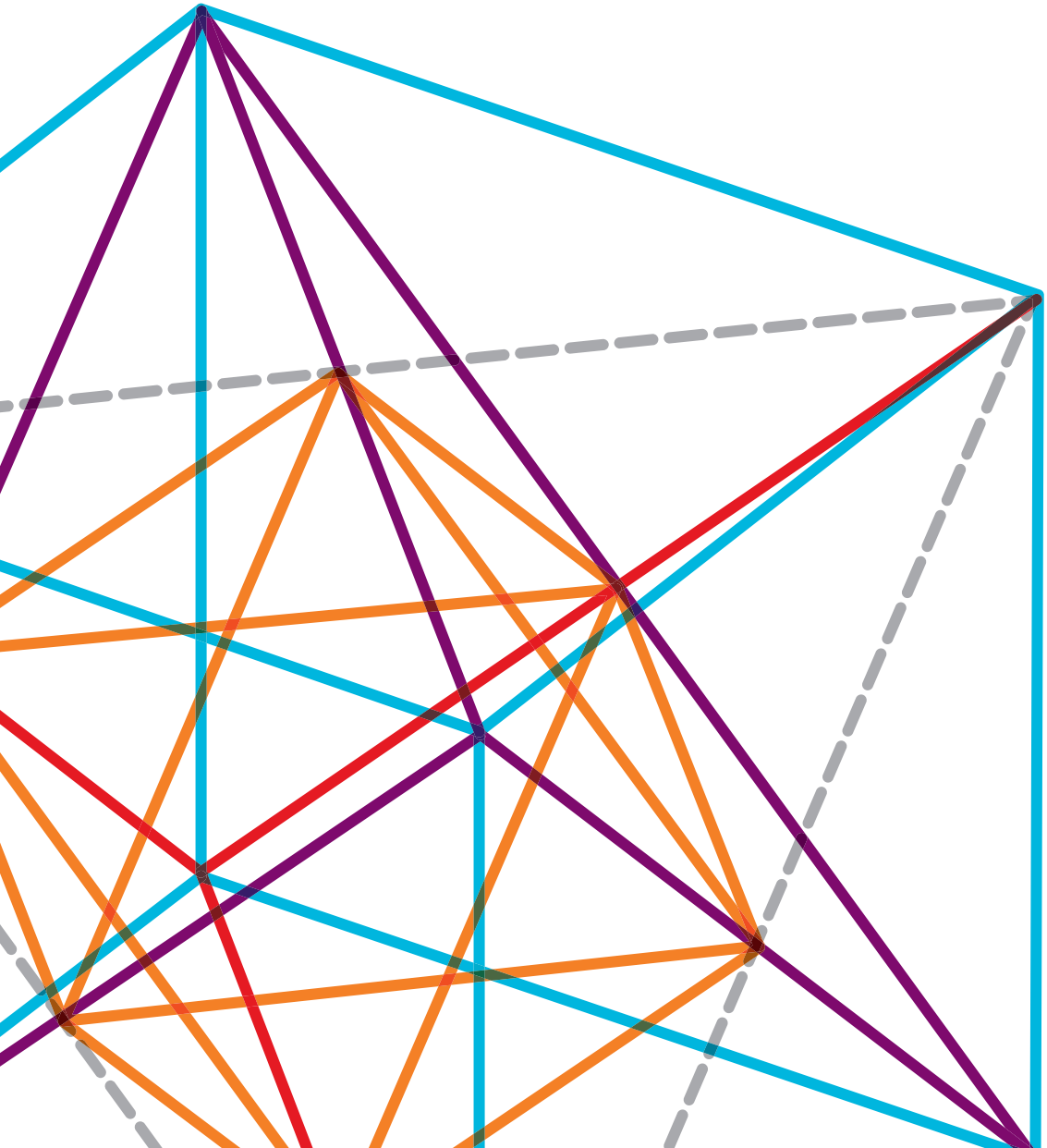
Magnetic Stripe Reader on ZQ500™ Series Printers

AppNote 2456936.909896

October 6, 2014

ZEBRA TECHNOLOGIES

SEE MORE. DO MORE.



INTRODUCTION

This document details the Magnetic Card Reader (MCR) accessory on the ZQ500 series, which can read magnetic stripe cards directly on the Zebra printer.

INSTALLATION

Text

The MCR features of the printer can only be used when the MCR accessory is installed on the unit. When successfully connected, the "mcr.revision" SetGetDo (SGD) command will be populated with the name of the connected MCR accessory.

CONFIGURING AND ACTIVATING THE MAGNETIC CARD READER

Configuration of the MCR reader is typically done at the time when the MCR is activated or "enabled". The MCR must be activated in order to swipe cards; otherwise it is "idle". The MCR is enabled by either:

- Setting the "mcr.enable" SGD command
- Using the MCR CPCL command

Both commands work with the same options detailed in the CPCL manual[1], with two notable exceptions.

- While the MCR CPCL command transmits the card data on the active interface[2] immediately when swiped, using mcr.enable requires the user to enter getvar mcr.query to get the card data.
- The ECHO option is not available on ZQ500 printers.

NOTES:

[1] See CPCL FOR LINK-OS ENABLED PRINTERS version 0.2, p290-295, "Magnetic Card Reading Commands"

[2] The active interface is the interface (USB, Ethernet, etc) for which the enable command was initiated on.

The above commands are used, in the most common cases, as follows:

SGD:

```
! U1 setvar "mcr.enable" "<timeout> <tracks to read> <MULTIPLE>
<ERRORS>"<CR><LF>
```

CPCL:

```
! U1 MCR <timeout> <tracks to read> <MULTIPLE> <ERRORS><CR><LF>
```

- Timeout – Time to wait for a card swipe, specified in eighths of a second; 0 for unlimited
- Tracks to read – From which tracks to read data: T1, T2 or T3 for track 1, track 2, or track 3
- MULTIPLE – Flag. When included, the printer will continue to read card swipes until cancelled (see cancel details below) or the specified timeout elapses. Otherwise, the printer will accept only one card swipe before returning the MCR to the idle state.
- ERRORS – Flag. When included, the printer will report errors.

Examples

Read all three tracks, with an unlimited timeout:

```
SGD: ! U1 setvar "mcr.enable" "0 T1 T2 T3"<CR><LF>

CPCL: ! U1 MCR 0 T1 T2 T3<CR><LF>
```

Read only track 2, with an unlimited timeout:

```
SGD: ! U1 setvar "mcr.enable" "0 T2"<CR><LF>

CPCL: ! U1 MCR 0 T2<CR><LF>
```

Read all three tracks, with a 5-second timeout:

```
SGD: ! U1 setvar "mcr.enable" "40 T1 T2 T3"<CR><LF>

CPCL: ! U1 MCR 40 T1 T2 T3<CR><LF>
```

Read all three tracks, with a 1-second timeout and error reporting:

```
SGD: ! U1 setvar "mcr.enable" "8 T1 T2 T3 ERRORS"<CR><LF>

CPCL: ! U1 MCR 8 T1 T2 T3 ERRORS<CR><LF>
```

Read all three tracks, with an unlimited timeout, until cancel command is issued:

```
SGD: ! U1 setvar "mcr.enable" "0 T1 T2 T3 MULTIPLE"<CR><LF>

CPCL: ! U1 MCR 0 T1 T2 T3 MULTIPLE<CR><LF>
```

RETURN THE MCR TO THE IDLE STATE

Once the card reader is enabled, it will remain so until any one of the following conditions is met:

- The specified timeout has elapsed
- A cancel command is issued, known as "cancelling" the card swipe
- The MCR is **not** in MULTIPLE mode and at least one requested track was read

A card swipe can be cancelled either by:

- Setting the "mcr.cancel" SetGetDo command
- Using the MCR-CAN command

Examples

```
SGD: ! U1 setvar "mcr.cancel" ""<CR><LF>

CPCL: ! U1 MCR-CAN<CR><LF>
```

Data Format

When performing a card swipe, the data is returned in the following format:

```
"<Track 1 Identifier>:<Track Data><CR><LF>
<Track 2 Identifier>:<Track Data><CR><LF>
…
<Track x Identifier>:<Track Data><CR><LF>
"
```

For example, the following swipe data contains tracks T1, T2, T3, E, EH1, EH2 and EKSN:

```
"T1:*4000*****4562^PUBLIC JR/JOHN Q.MR^****************************
<CR><LF>
T2:4000*****4562=****************<CR><LF>
T3:014000001234562==1230030003004020023000000021111194123000000000
123456789=56789832=0=000<CR><LF>
E:QjAJogdOevnDLd82UiuS8kMXezkXgNZfGqayHTsqhvjFBpa/
jIk6j1w0KmOULpSacb5GN9/q/6XL4BctMPtvGk/FWRlbBETdz+lf1YBB+AG2kt3jM7
+Yi3IRSm/HVNdKz2a0vXcmNs0=<CR><LF>
EH1:fYtlSq3cs3E8Um8cO7PdFyjJyag=<CR><LF>
EH2:NZHb49vJ0nEv2FQ/+QFu2sECHkI=<CR><LF>
EKSN:YplJAFYABEAAIg==<CR><LF>
"
```

NON-ENCRYPTED MODE

In non-encrypted mode, the card swipe will yield the following data:

Track	Data	Format
T1	Track 1 data	Plaintext
T2	Track 2 data	Plaintext
T3	Track 3 data	Plaintext

or

Track	Data	Format
Error	Error info	Plaintext

ENCRYPTED MODE

Encryption Scheme
Certain Zebra card readers support an encrypted reader mode, which encrypts card data according to the DUKPT standard[3]. Notably, the data on the card is not encrypted. The card reader accessory itself performs the encryption. The printer firmware does not have access to sensitive card data.

To decode this encrypted data, two pieces of information are required:
- The base encryption key
- The Key Serial Number, or KSN

The base key comes pre-programmed[4] in the reader unit and must be known to the user in order to decrypt the data[5]. The base key and KSN are combined to generate a unique (per transaction) *derived key*. The derived key is then used to decrypt the card data in accordance with the encryption scheme specified in mcr.crypt.algorithm[6].

Generating the derived key from the base key and KSN is a complicated process that is outside the scope of this document. There exist DUKPT libraries that can perform this operation.

NOTES:

[3] The DUKPT standard is beyond the scope of this document. See https://en.wikipedia.org/wiki/Derived_unique_key_per_transaction for more information.

[4] There is no way to retrieve the base key through the printer.

[5] Actual use cases of DUKPT don't require the user to know the base key. Decryption can be performed using the *derived key*. While the *derived key* is generated using the base key together with KSN, this key derivation process can be performed by a third-party that holds the base key before sending the derived key to the user. In such a case, the user can perform decryption with no knowledge of the base key, which is one of the design purposes of DUKPT. However, for testing purposes, it is perfectly reasonable for the user to hold the base key.

[6] Currently *3DES* is the only encryption algorithm supported.

Data Format

The card reader operates in one of two specific modes, *Level 3* and *Enhanced Level 3*, as programmed at the factory.

The following information applies only to ISO/ABA cards: In Level 3 mode, the card swipe will yield the following data:

Track	**Data**	**Format**
T1	Masked track 1 data	Plaintext
T2	Masked track 2 data	Plaintext
T3	Masked track 3 data	Plaintext
E	Encrypted track 1 and track 2 data	Base64
EH1	MD5 hash of track 1 data[7]	Base64
EH2	MD5 hash of track 2 data[7]	Base64
EKSN	Key Serial Number (KSN) for the given transaction	Base64

or

Track	**Data**	**Format**
Error	Error info	Plaintext

NOTES:

[7] The MD5 hash is a hash of the original binary data and not the masked plaintext.

In Enhanced Level 3 mode, the card swipe will yield the following data:

Track	Data	Format
T1	Masked track 1 data	Plaintext
T2	Masked track 2 data	Plaintext
E1	Encrypted track 1 data	Base64
E2	Encrypted track 2 data	Base64
E3	Encrypted track 3 data	Base64
EH1	MD5 hash of track 1 data[7]	Base64
EH2	MD5 hash of track 2 data[7]	Base64
EH3	MD5 hash of track 3 data[7]	Base64
EKSN	Key Serial Number (KSN) for the given transaction	Base64

or

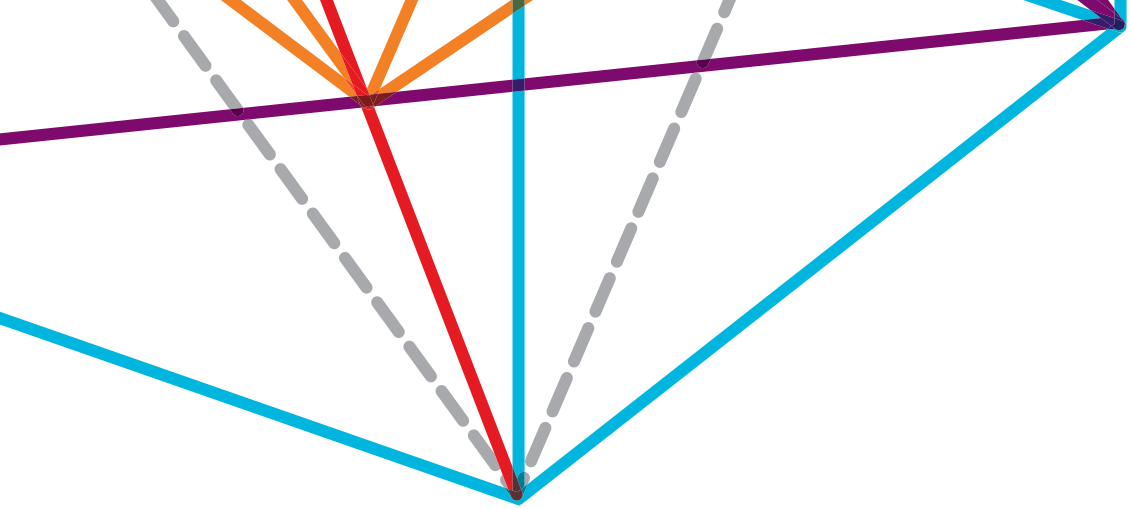
Track	Data	Format
Error	Error info	Plaintext

When decrypting track data in Enhanced Level 3, each track must be decrypted independently[8].

NOTES:

[7] The MD5 hash is a hash of the original binary data and not the masked plaintext.

[8] In other words, the internal CBC state of the cipher does not persist between tracks.



Corporate Headquarters
+1 800 423 0442
inquiry4@zebra.com

Asia-Pacific Headquarters
+65 6858 0722
apacchannelmarketing@zebra.com

EMEA Headquarters
+44 (0)1628 556000
mseurope@zebra.com

Latin America Headquarters
+1 847 955 2283
inquiry4@zebra.com

Other Locations / USA: California, Georgia, Illinois, Rhode Island, Texas, Wisconsin **Europe:** France, Germany, Italy, the Netherlands, Poland, Spain, Sweden, Turkey, United Kingdom **Asia Pacific:** Australia, China, Hong Kong, India, Indonesia, Japan, Malaysia, Philippines, Singapore, South Korea, Taiwan, Thailand, Vietnam **Latin America:** Argentina, Brazil, Colombia, Florida (LA Headquarters in USA), Mexico **Africa/Middle East:** Dubai, South Africa

(12/14)