



BT Mag

Bluetooth Magnetic Stripe Reader

USER MANUAL

80125501-001
22 January 2020

Revision History

Rev	Date	Description of Changes	By
A	8/20/2012	Initial release	JW
B	04/10/2013	Add the section 6 "Using the demo software" and section 7 card data output format	CH
C	01/22/2020	Font and layout facelift. Added card data use/error/success messages to Card Data Output Format.	CB

Table of Contents

1	INTRODUCTION	4
2	APPLICABLE DOCUMENTS	4
3	FEATURES AND BENEFITS	4
4	SPECIFICATIONS	5
5	OPERATION	7
6.	USING THE DEMO SOFTWARE	9
	6.1	<i>Android BTMag Demo</i> 9
	6.2	<i>IOS BTMag Demo</i> 12
7.	CARD DATA OUTPUT FORMAT	13
	7.1	<i>Unencrypted Data Output Format</i> 13
	7.2	<i>Encrypted Data Output Format</i> 13
	7.2.2	Enhanced Encryption Format 16
8.	OUTLINE DRAWING	19
APPENDIX A:		20

1. Introduction

The BT Mag is a handheld Magnetic Stripe reader that works with mobile devices or PCs with Bluetooth connectivity. It transfers card data via Bluetooth to POS applications in the host devices.

1.1. Applicable Documents

80125401-001 Rev.A	BT Mag Requirement Spec
ISO 7810	Identification cards, Physical characteristics 1 through 6
ISO 7811	Identification Cards - Track 1 through 3
ISO 4909	Magnetic stripe content for track 3
ISO 7812	Identification Cards – Identification for issuers Part 1 & 2 Identification Cards – Financial Transaction Cards AAMVA Specifications
ISO 7813	Driver’s License Standards - Most recent available 80101502-001 SPI Securehead manual

1.2. Features and Benefits

- Connects to any mobile device with Bluetooth capability
- Keychain holes for convenience
- Reads up to 3 tracks of card data
- Supports TDES and AES encryption using DUKPT Key Management
- Wireless range up to 30 feet
- Battery life: 4 hours active, 8 hours standby
- Micro-USB port for battery charging

2. Specifications

Interface

- Class 2 Bluetooth
- Can also be a standalone USB device for key injection when a Micro-B to A USB cable is connected
- Bluetooth is disabled during key injection or DFU communication

Magnetic stripe reader

- Meets ISO 7811 specification
- Supports AAMVA formats
- Support single, dual or triple tracks card
- Bi-directional swipe
- TDES, & AES Encryption
- Media Densities: 75 bpi through 210 bpi on all tracks, F2F Encoding Format
- Media Speed: 5 to 45 IPS
- Low Amplitude reading: >30 % @210 bpi, >40% @75 bpi

Batteries

- Rechargeable battery
- Battery life:
 - Up to 8 hours in standby mode
 - Up to 4 hours in active mode
 - Active mode is defined as 10+ swipes per hour
 - Charging through MicroUSB interface external charger
 - The unit is functional if Bluetooth connection is on while charging

LED

- One Dual-color Led to indicate Bluetooth/Charging status
- Blue indicates Bluetooth connection status
- Red indicates Charging
- Another Dual-color LED to indicate power/MSR read status
- Red indicates bad read
- Green on twice indicates good read
- Green blinking indicates power on/standby

Reliability

- Magnetic Head Life: 300,000 passes minimum
- Rail and Cover Life: 100,000 passes minimum
- MTBF: 300,000 POH

Electro-Static Discharges (ESD)

- 6kV contact, and 12kV air discharge

Environmental Temperature range

- Operating: 0 to 55 C (32 to 131 F) [non-condensing]
- Storage: -30 to 70 C (-22 to 158 F) [non-condensing]
- Relative humidity: Maximum 95% (non-condensing)

3. Operation

To power on or power off BT Mag, press and hold the power button for 5 seconds. Once the blue LED is blinking, the device is in pairing mode.

3.1. LED Definition

Event	BI-COLOR		BI-COLOR		Description
	GREEN	RED	BLUe	RED	
Power on	Flash 3 times ON (200ms) OFF (200ms)	OFF	OFF	X	Only applied when user power on the system or USB plug in.
Power off					Only applied when user shutdown the system or system detect low battery signal.
Power Standby/Sleep	ON(30ms) OFF(4970ms)	OFF	ON(30ms) OFF(4970 ms)	X	1.The reader will enter this mode when BT not in Pairing or Connected status
BTM Standby/Sleep					2. Short press the tact SW will force BT to search and build the Link again
MSR - Standby/Sleep					
Pairing	ON(30ms) OFF(4970ms)	OFF	ON(500ms) OFF(500ms)	X	After the reader is power on, that will into the pairing mode.
BTM Connected			ON (30ms) OFF (2970ms)		This mode indicates the MSR is connected with application software and waiting to accept the card swiping
Charging	X	X	X	ON	N/A
Charging Complete		X		OFF	N/A

Low Battery		ON (30ms)		X	The LED is blinking when Battery voltage is lower than 3.3V.
		OFF (2970ms)			The reader will shut down automatically when the battery voltage is lower than 3.2V
MSR - good read	Flash 2 times	OFF		X	Green indicates good read
	ON(500ms)		ON (30ms)		
	OFF(500ms)		OFF (2970ms)		
MSR - bad read	OFF	Flash 1 time		X	Red indicates bad read
		ON (500ms)	ON (30ms)		
		OFF (500ms)	OFF (2970ms)		

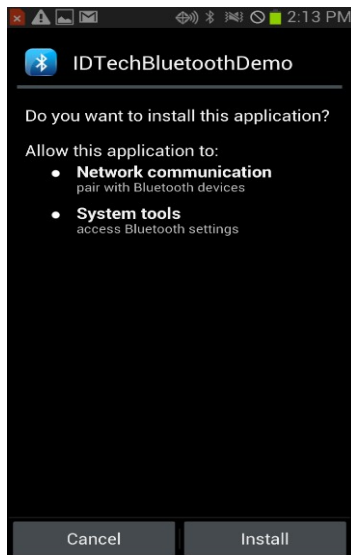
X: Not applicable

4. Using the demo software

4.1. Android BTMag Demo

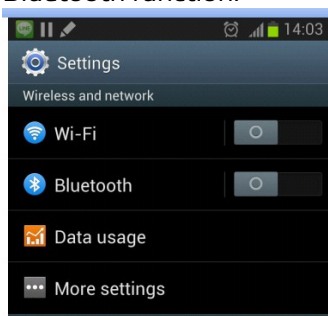
4.1.1. Install the demo

1. Copy the "BluetoothDemo.apk" file into the internal memory or SD card of the android phone or tablet.
2. Go to the file browser and click "BluetoothDemo.apk" to install.

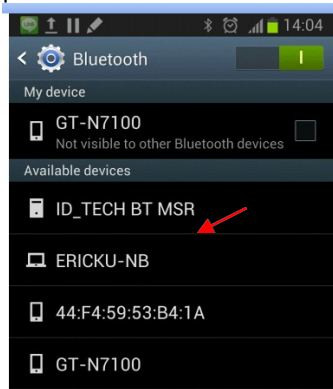


4.1.2. Pairing

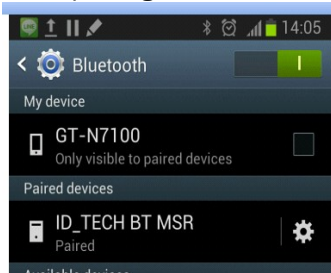
1. Long press power button (about 5 seconds) on the BTMag to power on. When the reader is turned on, the green LED will flash three times.
2. Go to the Settings of the phone. Find the Bluetooth setting button and turn on the Bluetooth function.



4. Press "ID_TECH BT MSR" for pairing. Some Android devices may require the password, it is "0000".



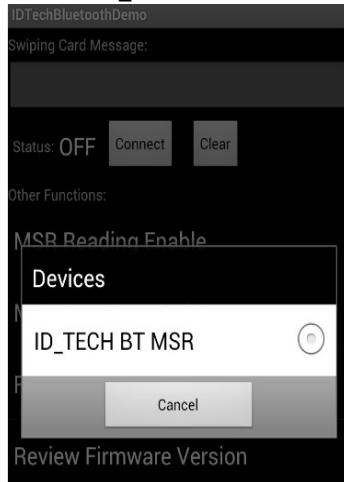
5. After pairing, it will be listed under the Paired devices.



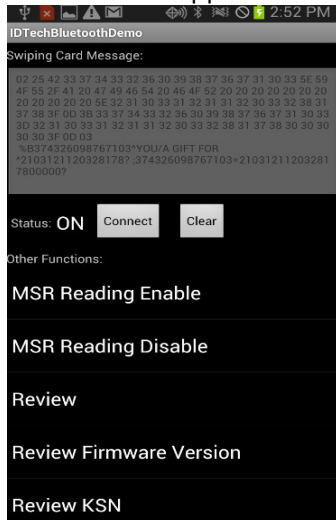
Note: The Pairing process should be completed within one minute after power on, otherwise BTMag will enter sleep mode (Green and Blue LED will flash together.) After the BTMag enters sleep mode, short-press the power button to wake the BTMag up from sleep mode and repeat the pairing process.

4.1.3. Connect to the Demo

1. Open the **IDTechBluetoothDemo** app.
2. Tap **Connect**, which brings up a prompt for selecting the device.
3. Select **ID_Tech BT MSR**. If the connection succeeds, it will show “connect success.”



4. Tap **MSR Reading Enable**; the status should be **ON**. A user can now swipe a card, and the card data appears in the text field at the top of the screen.



5. Tapping **Clear** clears the text in the window.

4.2. IOS BTMag Demo

4.2.1. Install the Demo

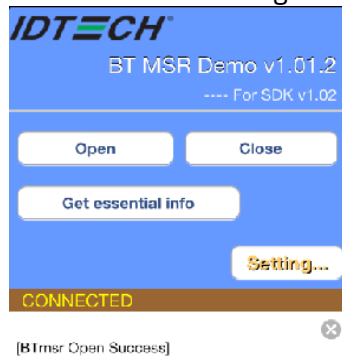
1. Install **BTmsrDemo.ipa** to the Apple device by iTunes or use XCode to build the source code directly.

4.2.2. Paring

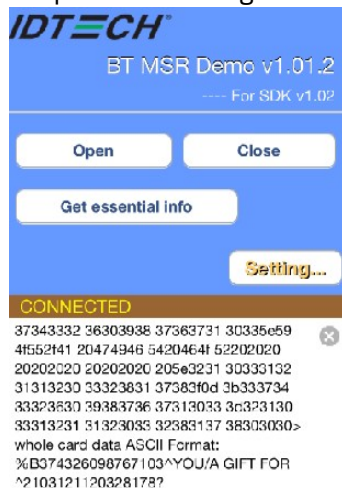
1. Go to Settings and turn on Bluetooth.
2. Select **ID_TECH BT MSR** to pair with. The password is **0000**.

4.2.3. Connect to the Demo

1. Tap **Open** to connect to the BT Mag device. The screen displays **BTmsr Open Success** on connecting successfully.



2. Swipe a card through the reader, the card data will be showed in the demo



3. To read or change the reader's settings, tap **Setting...**

5. Card Data Output Format

5.1. Unencrypted Data Output Format

5.1.1. Magnetic Track Basic Decoded Data Format

Track 1: <SS1><T1 Data><ES><Track Separator>

Track 2: <SS2><T2 Data><ES><Track Separator>

Track 3: <SS3><T3 Data><ES><Terminator>

Where:

- SS1 (start sentinel track 1) = % SS2
- (start sentinel track 2) = ;
- SS3 (start sentinel track 3) = ; for ISO, % for AAMVA ES (end sentinel all tracks) = ?
- Track Separator = Carriage Return
- Terminator = Carriage Return

For example:

```
%B4352378366824999^TFSTEST /THIRTYONE
```

```
^05102011000088200882000000?;4352378366824999=051020110000882?<CR>
```

Note that one of the following strings may proceed actual card data output:

- 02 45 48 03 0C: Card used successfully
- 02 45 48 03 0D: Card data error
- 02 45 4A 03 0E: Card data correct

5.2. Encrypted Data Output Format

5.2.1. Original Encryption Format

For ISO cards, both masked clear and encrypted data are sent, no clear data will be sent. For other cards, only clear data is sent.

Card data is sent out in format of:

```
<STX><LenL><LenH><Card Data><CheckLRC><Checksum><ETX>
```

- <STX> = 02h, <ETX> = 03h
- <LenL><LenH> is a two byte length of <Card Data>.
- <CheckLRC> is a one byte Exclusive-OR sum calculated for all <Card Data>.
- <Checksum> is a one byte Sum value calculated for all <Card data>.
- <Card Data> card data format is shown below.

5.2.2. ISO/ABA Data Output Format

- card encoding type (0: ISO/ABA, 4: for Raw Mode)
- track status (bit 0,1,2:T1,2,3 decode, bit 3,4,5:T1,2,3 sampling)
- track 1 unencrypted length (1 byte, 0 for no track1 data)
- track 2 unencrypted length (1 byte, 0 for no track2 data)
- track 3 unencrypted length (1 byte, 0 for no track3 data)
- track 1 masked (Omitted if in Raw mode)
- track 2 masked (Omitted if in Raw mode)
- track 3 data (Omitted if in Raw mode)
- track 1 encrypted (AES/TDES encrypted data)
- track 2 encrypted (AES/TDES encrypted data)
- track 3 encrypted (Only used in Raw mode)
- track 1 hashed (20 bytes SHA1-Xor)
- track 2 hashed (20 bytes SHA1-Xor)
- DUKPT serial number(10 bytes)

5.2.3. Non ISO/ABA Data Output Format

- card encoding type (1: AAMVA, 3: Others)
- track status (bit 0,1,2:T1,2,3 decode, bit 3,4,5:T1,2,3 sampling)
- track 1 length (1 byte, 0 for no track1 data)
- track 2 length (1 byte, 0 for no track2 data)
- track 3 length (1 byte, 0 for no track3 data)
- track 1 data
- track 2 data
- track 3 data

5.3. Description

5.3.1. Track 1, Track 2 and Track 3 Unencrypted Length

This one-byte value is the length of the original Track data. It indicates the number of bytes in the Track masked data field. It should be used to separate Track 1, Track 2 and Track 3 data after decrypting Track encrypted data field.

5.3.2. Track 3 Unencrypted Length

This one-byte value indicates the number of bytes in Track 3 data field.

5.3.3. Track 1 and Track 2 Masked

Track data masked with the MaskCharID (default is '*'). The first PrePANID (up to 6 for BIN, default is 4) and last PostPANID (up to 4, default is 4) characters can be in the clear (unencrypted).

5.3.4. Track 1, Track 2 and Track 3 Encrypted

This field is the encrypted Track data, using either TDES-CBC or AES-CBC with initial vector of 0. If the original data is not a multiple of 8 bytes for TDES or a multiple of 16 bytes for AES, the reader right pads the data with 0.

The key management scheme is DUKPT or Fixed key. For DUKPT, the key used for encrypting data is called the Data Key. Data Key is generated by first taking the DUKPT Derived Key exclusive or'ed with 0000000000FF0000 0000000000FF0000 to get the resulting intermediate variant key. The left side of the intermediate variant key is then TDES encrypted with the entire 16-byte variant as the key. After the same steps are performed for the right side of the key, combine the two key parts to create the Data Key.

5.3.5. Encrypted Data Length

Track 1 and Track 2 data are encrypted as a single block. In order to get the number of bytes for encrypted data field, we need to get Track 1 and Track 2 unencrypted length first. The field length is always a multiple of 8 bytes for TDES or multiple of 16 bytes for AES. This value will be zero if there was no data on both tracks or if there was an error decoding both tracks. Once the encrypted data is decrypted, all padding 0 need to be removed. The number of bytes of decoded track 1 data is indicated by track 1 unencrypted length field. The remaining bytes are track 2 data, the length of which is indicated by track 2 unencrypted length filed.

Track 1 and Track 2 Hashed

BTMag reader uses SHA-1 to generate hashed data for both track 1 and track 2 unencrypted data. It is 20 bytes long for each track. This is provided with two purposes in mind: One is for the host to ensure data integrity by comparing this field with a SHA-1 hash of the decrypted Track data, prevent unexpected noise in data transmission. The other purpose is to enable the host to store a token of card data for future use without keeping the sensitive card holder data. This token may be used for comparison with the stored hash data to determine if they are from the same card.

5.4. Enhanced Encryption Format

Card data is sent out in the following format

<STX><LenL><LenH><Card Data><CheckLRC><Checksum><ETX>

0	STX
1	Data Length low byte
2	Data Length high byte
3	Card Encode Type ¹
4	Track 1-3 Status ²
5	Track 1 data length
6	Track 2 data length
7	Track 3 data length
8	Clear/masked data sent status ³
9	Encrypted/Hash data sent status ⁴
10	Track 1 clear/mask data Track 2 clear/mask data Track 3 clear/mask data Track 1 encrypted data Track 2 encrypted data Track 3 encrypted data Session ID (8 bytes) (Security level 4 only) Track 1 hashed (20 bytes each) (if encrypted and hash track 1 allowed) Track 2 hashed (20 bytes each) (if encrypted and hash track 2 allowed) Track 3 hashed (20 bytes each) (if encrypted and hash track 3 allowed) KSN (10 bytes) CheckLRC Checksum ETX

Where <STX> = 02h, <ETX> = 03h

5.4.1. Note 1 : Card Encode Type

Card Type will be 8x for enhanced encryption format and 0x for original encryption format.

Value	Encode Type	Description
00h / 80h	ISO/ABA	format
01h / 81h	AAMVA	format
03h / 83h	Other	
04h / 84h	Raw;	un-decoded format

For Type 04 or 84 Raw data format, all tracks are encrypted and no mask data is sent. No track indicator '01', '02' or '03' in front of each track.

Track indicator '01','02' and '03' will still exist for non-encrypted mode.

5.4.2. Note 2: Track 1-3 status byte

Field 4:

Bit 0: 1— track 1 decoded data present

Bit 1: 1— track 2 decoded data present

Bit 2: 1— track 3 decoded data present

Bit 3: 1— track 1 sampling data present

Bit 4: 1— track 2 sampling data present

Bit 5: 1— track 3 sampling data present

Bit 6, 7 — Reserved for future use

5.4.3. Note 3: Clear/mask data sent status

Field 8 (Clear/mask data sent status) and field 9 (Encrypted/Hash data sent status) will only be sent out in enhanced encryption format.

Field 8: Clear/masked data sent status byte: Bit

0: 1 —track 1 clear/mask data present

Bit 1: 1— track 2 clear/mask data present

Bit 2: 1— track 3 clear/mask data present

Bit 3: 0— reserved for future use

Bit 4: 0— reserved for future use

Bit 5: 0— reserved for future use

5.4.4. Note 4: Encrypted/Hash data sent status

Field 9: Encrypted data sent status

Bit 0: 1— track 1 encrypted data present

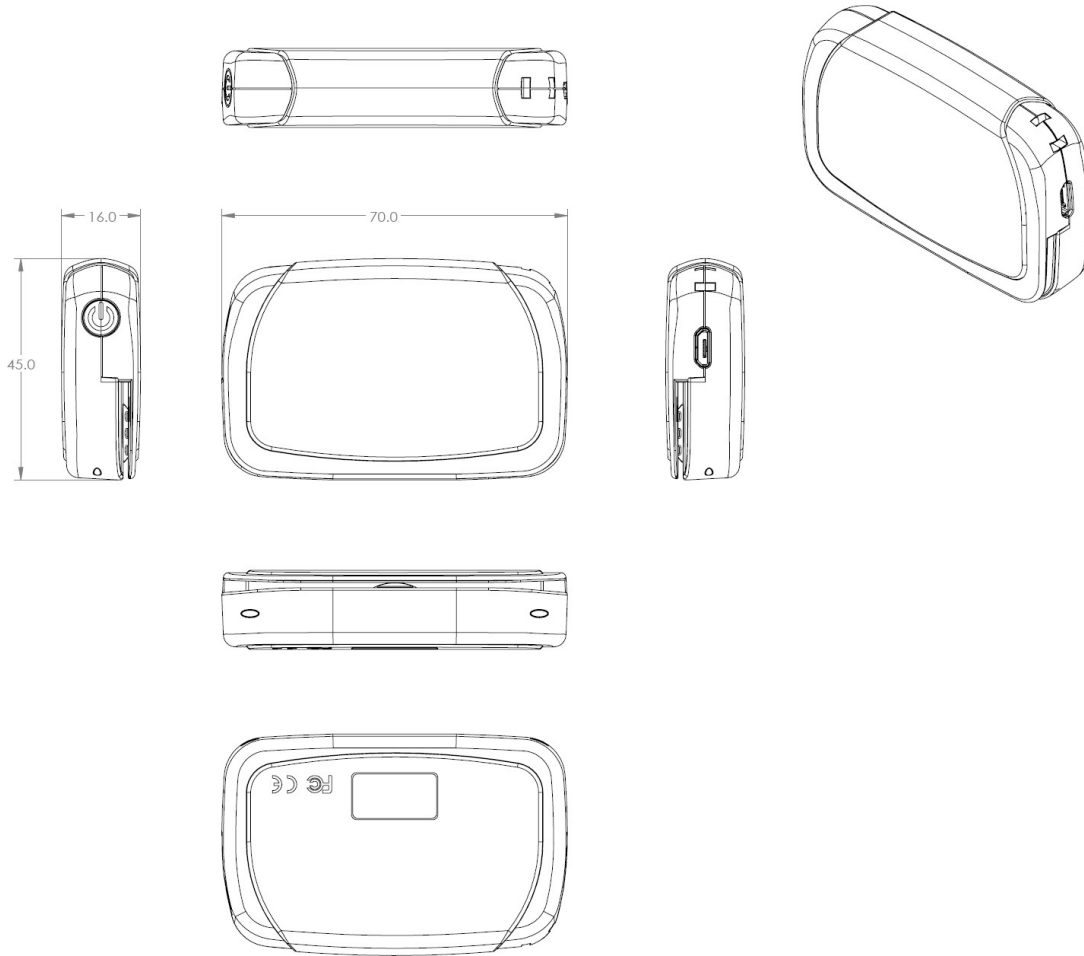
Bit 1: 1— track 2 encrypted data present

Bit 2: 1— track 3 encrypted data present

Bit 3: 1— track 1 hash data present
Bit 4: 1— track 2 hash data present
Bit 5: 1— track 3 hash data present
Bit 6: 1—session ID present
Bit 7: 1—KSN present

For the detailed encrypted data parsing example, please refer to Appendix A.

6. Outline Drawing



=0x70 (112 decimal)

863E9E3DA28E455B28F7736B77E47A64EDDA3BF03A06E44F31D1818C0BCD7A3
53FB1AD70EFD30FFC3DA08A4FBC9372E57E8B40848BAEAA3FE724B3550E2F4B
223E6BF264BEAE9E39142B648CDB51FB8DAF8EA5B63913D29419B67582FCCCE
9B372660F03668CC453216D9449C6B67EF3

Track 1 hashed 3418AC88F65E1DB7ED4D10973F99DFC8463FF6DF

Track 2 hashed 113B6226C4898A9D355057ECAF11A5598F02CA31

KSN 62994901190000000001

LRC, checksum and ETX 39 9F 03

Masked Data:

Track 1 data masked in ASCII:

%*4266*****9999^BUSH JR/GEORGE
W.MR^*****?*

Track 2 data masked in ASCII:

;4266*****9999=*****?*

Track 3 data unencrypted in ASCII:

;33333333376767607070776767633333333337676760707077676763333333333767
676070707767676333333333376767607072

Key Value: F8 2A 7A 0D 7C 67 46 F1 96 18 9A FB 54 2C 65 A3

KSN: 62 99 49 01 19 00 00 00 00 01

Decrypted Data in ASCII:

%B4266841088889999^BUSH JR/GEORGE
W.MR^0809101100001100000000046000000?!;4266841088889999=08091011000004
6?0;33333333376767607070776767633333333337676760707077676763333333333
76767607070776767633333333376767607072

Decrypted Data in Hex:

2542343236363834313038383838393939395E42555348204A522F47454F5247452057
2E4D525E3038303931303131303030303131303030303030303030303436303030303030
3F213B343236363834313038383838393939393D30383039313031313030303030303436
3F300000000000

Track2 masked data in ASCII

```
;4266*****9999=*****?*
```

In this example there is no Track 3 data either clear or masked (encrypted and hashed data is below)

Track 1 encrypted length 0x48 rounded up to multiple of 8 bytes = 0x48 (72 decimal)

```
DA7F2A52BD3F6DD8B96C50FC39C7E6AF22F06ED1F033BE0FB23D6BD33DC5A1
F808512F7AE18D47A60CC3F4559B1B093563BE7E07459072ABF8FAAB5338C6CC
8815FF87797AE3A7BE
```

Track 2 encrypted length 0x23 rounded up to multiple of 8 bytes =0x28 (40 decimal)

```
AB3B10A3FBC230FBFB941FAC9E82649981AE79F2632156E775A06AEDAF6F0
A184318C5209E55AD
```

Track 3 encrypted length 0x6B rounded up to 8 bytes =0x70 (64 decimal)

```
44A9CCF6A78AC240F791B63284E15B4019102BA6C505814B585816CA3C2D2F42
A99B1B9773EF1B116E005B7CD8681860D174E6AD316A0ECDBC687115FC89360A
EE7E430140A7B791589CCAADB6D6872B78433C3A25DA9DDAE83F12FEFAB530
CE405B701131D2FBAAD970248A45600093
```

Track 1 data hashed length 20 bytes 3418AC88F65E1DB7ED4D10973F99DFC8463FF6DF

Track 2 data hashed length 20 bytes 113B6226C4898A9D355057ECAF11A5598F02CA31

Track 3 data hashed length 20 bytes 688861C157C1CE2E0F72CE0F3BB598A614EAABB1

KSN length 10 bytes 62994901190000000002

LCR, check sum and ETX 06E203

Clear/Masked Data in ASCII:

```
Track 1: %*4266*****9999^BUSH JR/GEORGE
W.MR^*****?*
```

```
Track 2: ;4266*****9999=*****?*
```

```
Key Value: 1A 99 4C 3E 09 D9 AC EF 3E A9 BD 43 81 EF A3 34 KSN: 62 99 49
01 19 00 00 00 00 02
```

Decrypted Data:

Track 1 decrypted

```
%B4266841088889999^BUSH JR/GEORGE
W.MR^080910110000110000000046000000?!
```

Track 2 decrypted

```
;4266841088889999=080910110000046?0
```

Track 3 decrypted

;33333333337676760707077676763333333333767676070707767676333333333376767607077676763333333333767676070776767633333333337676760707?2

Track 1 decrypted data in hex including padding zeros (but there are no pad bytes here)

2542343236363834313038383838393939395E42555348204A522F47454F5247452057
2E4D525E3038303931303131303030303131303030303030303030303436303030303030
3F 21

Track 2 decrypted data in hex including padding zeros

3B343236363834313038383838393939393D3038303931303131303030303034363F30
0000000000

Track 3 decrypted data in hex including padding zeros

3B33333333333333333333333373637363736303730373037373637363736333333333333
3333333337363736373630373037303737363736373633333333333333333333373637
3637363037303730373736373637363333333333333333333337363736373630373037
3F320000000000