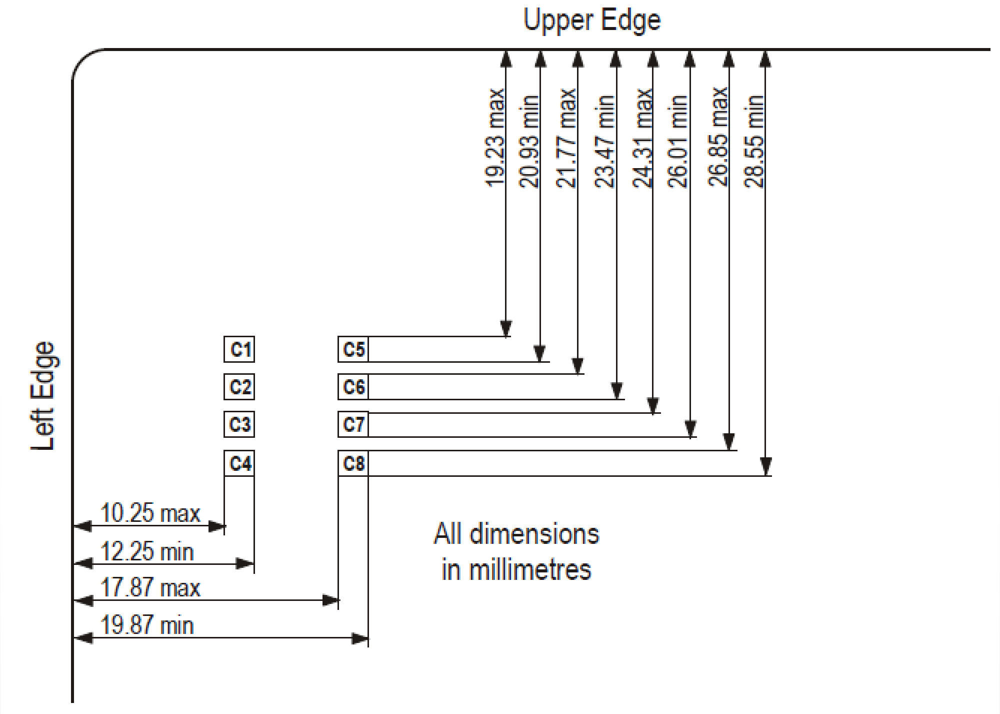
SoTech

Fun with Chip&PIN

Denis A Nicole

EMV electrical protocol

C1 Vcc (+5V, 55mA until Jan 2014)
C2 Reset (active low)
C3 Clock (1MHz to 5MHz)
C5 Ground
C7 Input/Output (1 bit = 372 clocks)



EMV low-level protocol

Prior to transmission of a character, the I/O line shall be in state H.

A character consists of 10 consecutive bits:

- 1 start bit in state L
- 8 bits, which comprise the data byte
- 1 even parity checking bit
- The start bit is detected by the receiving end by periodically sampling the I/O line. The sampling time should be less than or equal to 75 clocks.

It’s more-or-less low-voltage RS232 with a weak pull-up—you could use 9600 Baud if you set the clock right.

You can buy a cheap reader

- http://rsww.com part 619-9370: £24. I had to find the PC/SC driver (WindowsXP) in Brazil.
- Or you can make one: you need a 3.5795MHz crystal to get 9600Baud.
  Details later: the crystal only arrived yesterday.
- Simple software (Perl with GUI): G Scriptor: http://www.springcard.com/download/software.html

You need lots of specs

- ISO7816: go to the Uni library
- EMVco:
  http://www.emvco.com/specifications.aspx
- Visa, you want the
  Visa Integrated Circuit Card Specification
  https://partnernetwork.visa.com/vpn/global/category.do?userRegion=1&categoryId=43&documentId=77
  or
  www.scardsoft.com/documents/VISA/ICC_Card.pdf

SoTech

And an FRS

- http://www.lightbluetouchpaper.org/category/banking-security/

It’s then just a matter of chasing bytes

[Reset] You just type reset in the script window.
ATR: 3B 6E 00 00 00 31 C0 71 D6 65 A3 03 01 80 00 83 90 00
This means:
3B TS *Initial Byte* Active high logic, least significant byte first
6E T_0 *Format Byte* Y1=6 TB_1, TC_1 present. K=14 historical bytes
00 TB_1 deprecated Use of C6 contact
00 TC_1 extra delay between characters
00 First historical byte: last three bytes are status
The rest is in *compact header* format
31 C0 Card service data. Select application by full or partial DF
Use read record command; card has master file
71 D6 Use full or partial DF name or file ID
Use short EF identifier or record number
65 A3 03 01 80 00 Pre-issue data. I know not what, but not manufacturer
83 Life cycle status (What?)
90 00 Success. You’ll see lots of these

Let’s find what’s on the card

```
Sending: 00 A4 04 00 0E 31 50 41 59 2E 53 59 53 2E 44 44 46 30 31
Select 1PAY.SYS.DDF01 See EMV4.2 book 1 p138
Received: 61 2B
0x2B bytes of response still available.
Now we have to actually ask for the response
Sending: 00 C0 00 00 2B
Received: 6F 29 84 0E 31 50 41 59 2E 53 59 53 2E 44 44 46 30
31 A5 17 88 01 01 5F 2D 02 65 6E BF 0C 0C C5 0A FF
FF 3F 00 00 00 03 FF FF 03 90 00
We’re now into ASN.1 Tag-Length-Value (tag is 1 or 2 bytes)
6F 29 This is the File Control Information. 41 bytes follow
  84 0E Distinguished File Name “1PAY.SYS.DDF01”. We knew that
  A5 17 FCI Proprietary data
    88 01 01 What we really wanted. The Short File ID
    5F 2D 02 “EN” Language Preference
    BF 0C 0C 12 bytes of discretionary data I can’t parse
90 00 Success
```

Now we can read the file to find the apps

```
Sending: 00 B2 01 0C 00
Received: 6C 16
Wrong length: should be 0x16
We have to ask for the data
Sending: 00 B2 01 0C 16
Received: 70 14 61 12 4F 07 A0 00 00 00 03 10 10 50 04 56 49
53 41 87 01 01 90 00
70 14 Record template
  61 12 Application ID template
    4F 07 A0 00 00 00 03 10 10 Application ID: Visa credit or debit
    50 04 “VISA” Application Label
    87 01 01 Application priority
90 00 Success
```

There’s one more app on the card!

```
Sending: 00 B2 02 0C 00
Received: 6C 1A
Wrong length: should be 0x1A
Sending: 00 B2 02 0C 1A
Received: 70 18 61 16 4F 07 A0 00 00 00 03 80 02 50 0B 43 4F
4F 50 45 52 41 54 49 56 45 90 00
70 14 Record template
  61 12 Application ID template
    4F 07 A0 00 00 00 03 80 02 Application ID: Visa credit or debit
    50 0B “COOPERATIVE” Application Label
90 00 Success
Sending: 00 B2 03 0C 00
Received: 6A 83
Wrong parameter(s) P1-P2. Record not found.
```

Open the Visa app

```
Sending: 00 A4 04 00 07 A0 00 00 00 03 10 10
Received: 61 2A
0x2A bytes of response still available.
We have entered a new state on the card
Sending: 00 C0 00 00 2A
Received: 6F 28 84 07 A0 00 00 00 03 10 10 A5 1D 50 04 56 49
53 41 87 01 01 5F 2D 02 65 6E BF 0C 0C C5 0A 01 01
7F 51 47 00 03 0F FF 03 90 00
6F 28
  84 07 A0 00 00 00 03 10 10 We've seen all this before
  A5 1D
    50 04 "VISA"
    87 01 01
    5F 2D 02 "EN"
    BF 0C 0C ...
```

Check a PIN

```
Sending: 80 CA 9F 17 00 How many tries do I have left?
Received: 9F 17 01 03 90 00 Three. It might not tell you!

Sending: 00 20 00 80 08 24 00 00 FF FF FF FF FF Is it 0000?
Received: 63 C2 Nope; two tries left
State of non-volatile memory changed. Counter: 0x2

Sending: 00 20 00 80 08 24 00 01 FF FF FF FF FF 0001?
Received: 90 00 Yes

Sending: 80 CA 9F 17 04
Received: 9F 17 01 03 90 00 We’re back to three tries

If you brick your card, an ATM might check online and reset it.
Don’t blame me!
```

Get the processing options: files to read

```
Sending: 80 A8 00 00 02 83 00
  02 83 00 2 bytes if input data, the PDOL tag (83) and no content
Received: 61 10
0x10 bytes of response still available.
Sending: 00 C0 00 00 10
Received: 80 0E 5C 00 08 01 01 00 10 01 03 01 18 01 03 00 90 00
80 0E
  Application Interface Profile
  5C Supports SDA (bad!), cardholder verification,
     issuer authentication, perform terminal risk management
  00 Reserved for Future Use
  Application File Locator
  SFI<<3  1st record  total records  records in offline authentication
  08      01         01             00
  10      01         03             01
  18      01         03             00
90 00 Success
```

Then we read the seven records

```
Sending: 00 B2 01 0C 00
Read record 01 from SFI 1 0C = SFI<<3|4
Received: 6C 42
Wrong length: should be 0x42
Sending: 00 B2 01 0C 42 Try again with the right length
Received: I'm not going to show you this
70 40
  5F 20 0D Cardholder name
  57 13 Track two data
  9F 1F 18 Track one data
90 00 Success

All you need to clone a stripe!
```

What’s in the other records?

```
Sending: 00 B2 01 14 34
Received: I’m not going to show you this
70 32
  5F 25 05 YY MM DD Application effective date, BCD
  5F 24 03          Application expiry date
  5A 08    Application Primary Account Number, BCD
  5F 34 01 02 PAN sequence number
  9F 0D 05 Default Issuer action Code. What to do with Terminal
           Verification Results. Bitfields, 1= active
  9F 0E 05 Denial...
  9F 0F 05 Online...
90 00 Success
```

What’s in the other records?

```
Sending: 00 B2 02 14 32
Received: I’m not going to show you this
70 30
  8C 15 Card risk management data object list 1
  8D 17 ...                                   2
90 00 Success

Sending: 00 B2 03 14 2D
Received: I’m not going to show you this
70 2B
  9F 08 02 Application version number
  5F 30 02 Service Code
  4F 07 02 Application Usage Control
  5F 28 02 08 26 Application country code
  8E 10 Cardholder verification List
  9F 42 02 08 26 Application currency code
```

Cardholder Verification List?

```
00 00 00 00 Amount X (BCD in pence)
00 00 00 00 Amount Y
41 03 Use plaintext PIN on card, if supported by terminal (03), else
5E 03 Use paper signature, if supported by terminal, else
42 03 Use enciphered PIN online, if supported by terminal, else
1F 03 Don’t bother with any authentication, if supported...
      otherwise, fail.
```

What’s in the other records?

```
Sending: 00 B2 02 1C 2F
Received: I’m not going to show you this
70 2D
  9F 32 01 03 Issuer public key exponent. A popular RSA value
  8F 01 07 Certificate Authority Public Key Index
  92 24 Issuer public key remainder
        (? 288 bits: too small for a modulus)2
90 00 Success

Sending: 00 B2 03 1C 96
Received: I’m not going to show you this
70 81
  This is Signed Static Application Data, but the byte counts
  do not add up...
```