PHILIPS

mifare® *DESFire Functionality*

CAS - 2006

mifare® *DESFire* Functionality Agenda

- Introduction
- Main Characteristics & Block Diagram
- DESFire File System
  - Applications & Files
  - File Types
  - Key Management
  - Access Rights
  - Backup Management
  - Memory Mapping
- Typical Transaction Time
- Delivery Types & Development Tools

Interface:

- Contactless only
- Fully compliant to the ISO/IEC14443A (1-4)
- 7 bytes UID (“Double Size UID”)
- Operating distance up to 10cm
- Data transmission: 106 – 424 kBd
- Compatible to the Mifare Reader

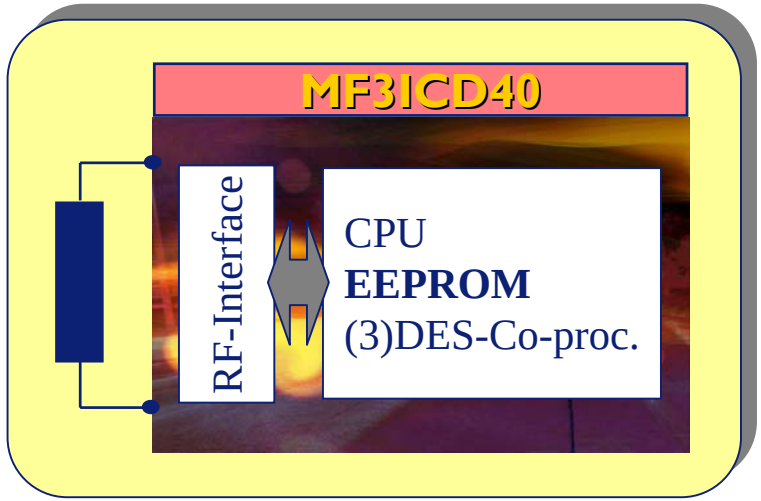
CPU & OS:

- Asynchronous CPU core
- (3) DES coprocessor
- Fixed Command Set
- No Customer ROM codes

„Buy the card and use it.“

NV Memory:

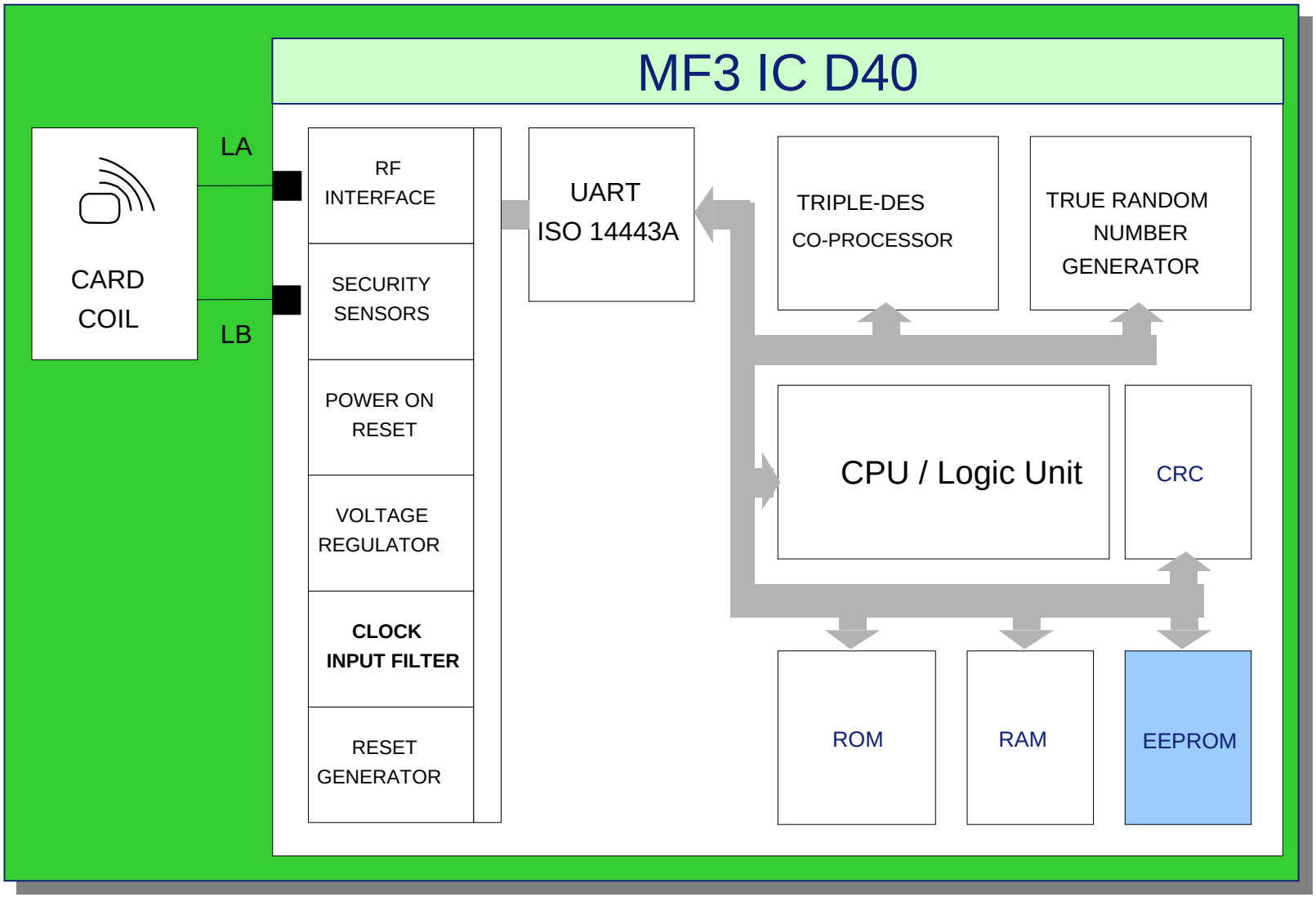
- 4 kByte EEPROM
- Erase + Write access: 1ms each
- R/W-Cycles: >100K
- Data retention: 10 years



File System:

- up to 28 application / card
- up to 16 files / application
- up to 14 keys / application
- 1 masterkey for card maintenance
- Plain, (3)DES encrypted, or MACed data transmission
- On-Chip Backup management

mifare® DESFire Block Diagram



The 4kByte EEPROM can be used for:

up to **28 different applications**

(like “sub-directories” on a HDD)

1	2	3	4	...	28

One MasterKey

(for card maintenance)

up to **16 files / application**

(like data-files within one “sub-directory”)

One MasterKey

(for application maintenance)

includes

up to **14 (3)DES keys / application**

(valid only within the “sub-directory”)

3 File types:

1) Data Files
2) Value Files
3) Record Files

Standard Data File (0x00)

File# 0x00 ... 0x0F

User File size: = 1 byte ... 4 kbyte

Required EEPROM size: = File Size*

General data file (e.g. card issuer data, card holder name)

$$EEPROM\,Size = INT\left[(StdData\,FileSize - 1)/32\right] * 32 + 32$$

Create Standard Data File

1	1	1	2	3
CMD	File #	Com Set	Access Rights	File Size

*Internally the NV-memory is allocated in blocks of 32 bytes.
(E.g. every file with a size of 1-32 bytes internally always uses 32 bytes.)

Backup Data File (0x01)

File# 0x00 ... 0x07

File size: = 1 Byte ... 2 kByte

Required EEPROM: = 2 x File size*

General data file (e.g. card issuer data, card holder name)

$$EEPROM\,Size = 2 \cdot INT\left[(BackupData\ \ FileSize - 1)/32\right] * 32 + 32$$

Create Backup Data File

1	1	1	2	3
CMD	File #	Com Set	Access Rights	File Size

*Internally the NV-memory is allocated in blocks of 32 bytes.
(E.g. every file with a size of 1-32 bytes internally always uses 32 bytes.)

Value File (0x02)

File# 0x00 ... 0x07

Required EEPROM size: = 32 Bytes

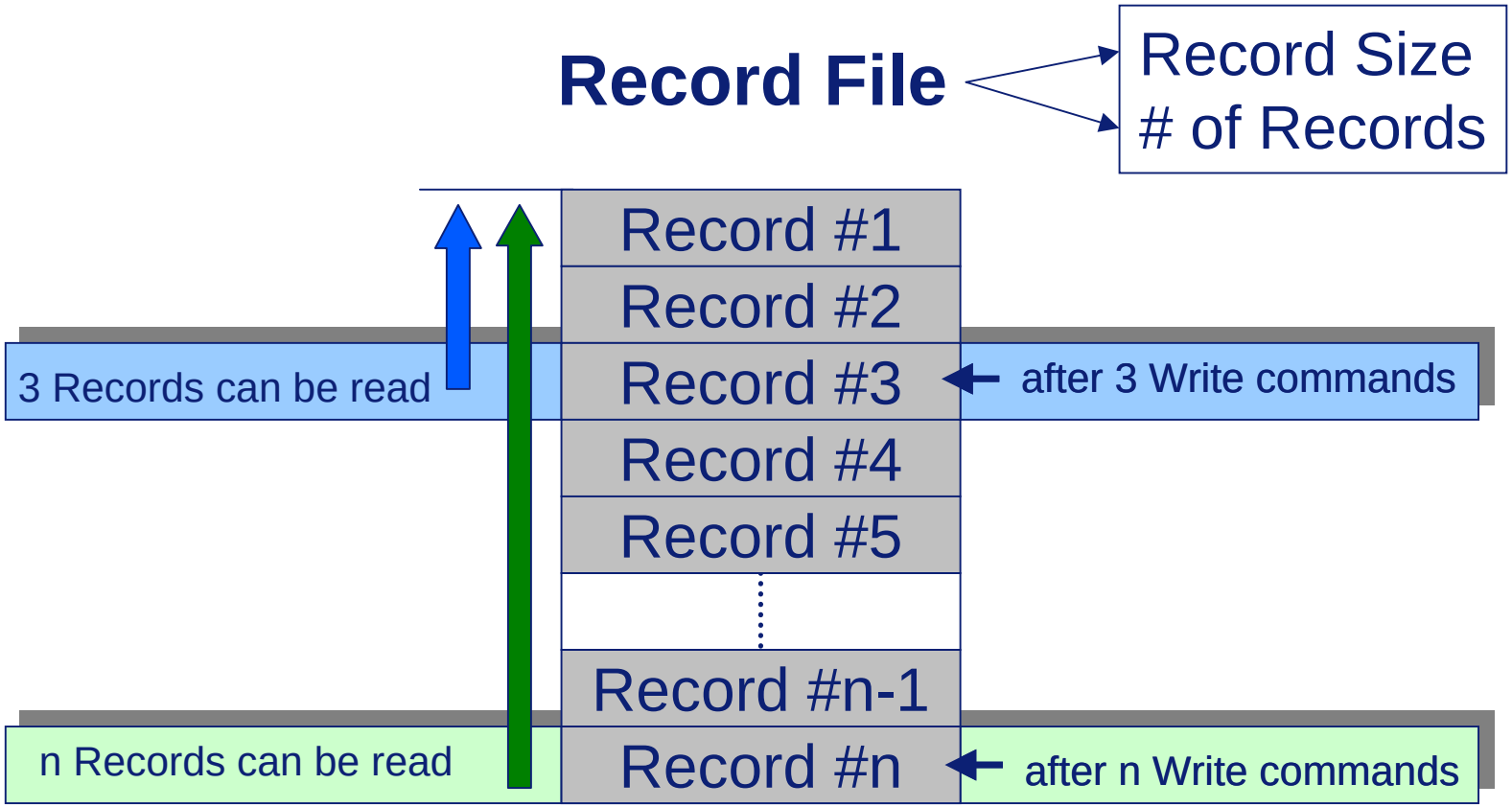
Value Range	= -16 777 215...+16 777 216 (4 Byte signed integer)
Lower Limit	= -16 777 215...+16 777 215
Upper Limit	= -16 777 214...+16 777 216 (Lower Limit < Upper Limit)

Limited Credit enabled (0x01) / disabled (0x00)

Create Value File

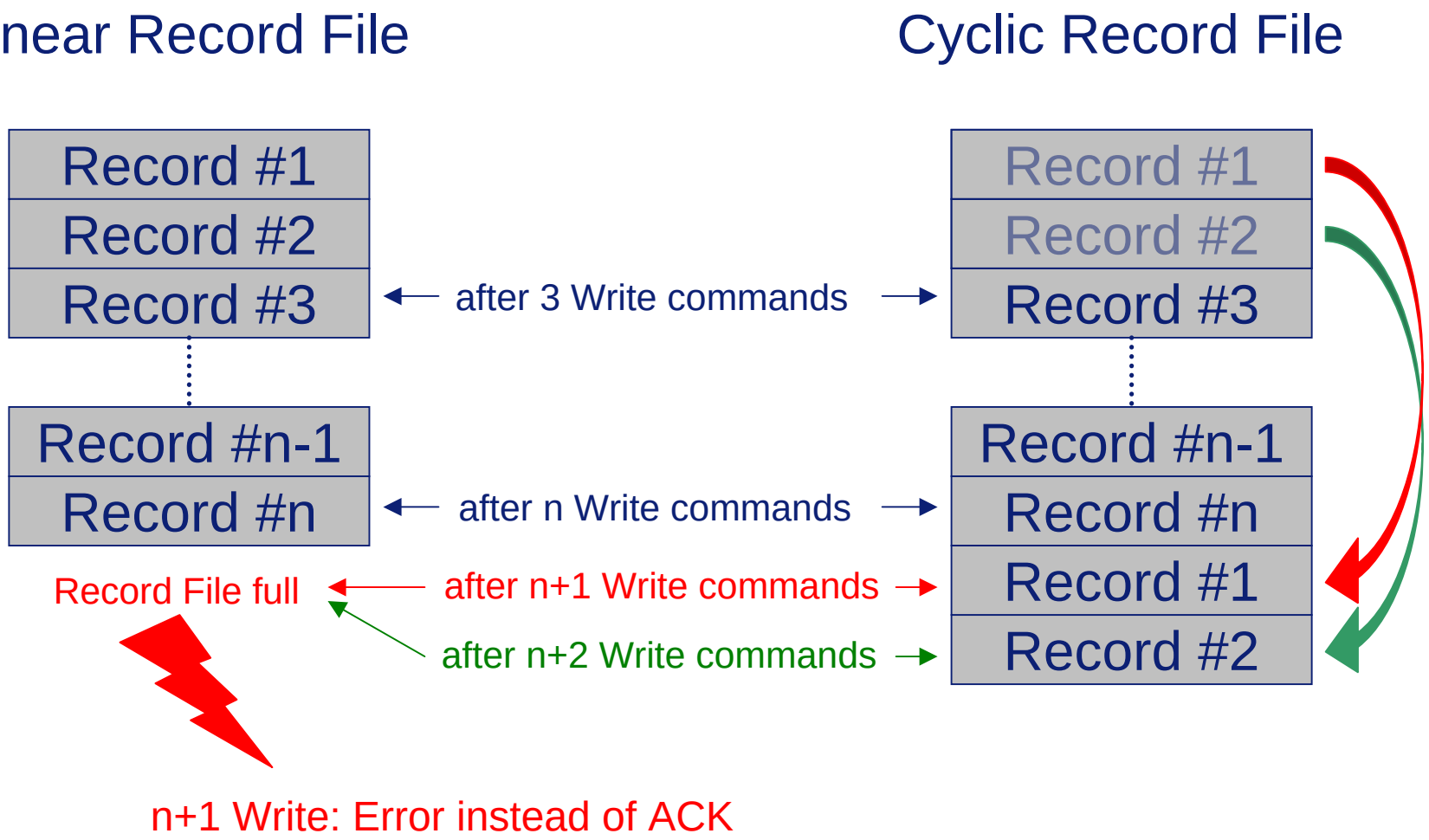
1	1	1	2	4	4	4	1
CMD	File #	Com Set	Access Rights	Lower Limit	Upper Limit	Value	Credit Limited enabled

mifare® *DESFire* File Types: Record File



- A Record File contains n Records.
- Each Record can be written once.
- The latest and all the previous Records can be read (at once).

mifare® *DESFire* File Types: Record File



Linear Record Files (0x03)

File# 0x00 ... 0x07

Required EEPROM size: = 32Bytes + Record Size • # of Records*

Record Size	= 0x00 00 01 ... 0xff ff ff (1 Byte - 4k Byte)
# of Records	= 0x00 00 01 ... 0xff ff ff

Create Record File

1	1	1	2	3	3
CMD	File #	Com Set	Access Rights	Record Size	Max. # of Records

* Internally the NV-memory is allocated in blocks of 32 bytes. (E.g. a Record File with 2 Records and a size of 10 Bytes/Record internally always uses 64 bytes.)

Cyclic Record Files (0x03)

File# 0x00 ... 0x07

Required EEPROM size: = 32Bytes + Record Size • # of Records*

Record Size	= 0x00 00 01 ... 0xff ff ff (1 Byte - 4k Byte)
# of Records	= 0x00 00 02 ... 0xff ff ff

Create Record File

1	1	1	2	3	3
CMD	File #	Com Set	Access Rights	Record Size	Max. # of Records

* Internally the NV-memory is allocated in blocks of 32 bytes. (E.g. a Record File with 2 Records and a size of 10 Bytes/Record internally always uses 64 bytes.)

DESFire data transmission:

Example Data: „Hello World“

Plain Data

Data										
48	65	6C	6C	6F	20	57	6F	72	6C	64

MACed* Data

Data											MAC			
48	65	6C	6C	6F	20	57	6F	72	6C	64	23	42	A1	2E

(3)DES enciphered

Data + 2Byte CRC -> filled up to n*8 -> (3)DES encrypted														
f2	45	2a	e0	50	56	3c	02	43	4e	63	ac	04	bb	21	26

Coding of the Communication Settings:

	b7	b6	b5	b4	b3	b2	b1	b0	Hex
Plain Data	0	0	0	0	0	0	x	0	0x00
MACed	0	0	0	0	0	0	0	1	0x01
(3)DES encrypted	0	0	0	0	0	0	1	1	0x03

*MAC: **M**essage **A**uthentication **C**ode

DES and 3DES keys are stored in 16 bytes strings.

DES: 1st half = 2nd half ➡ **Single DES**

00	11	22	33	44	55	66	77	00	11	22	33	44	55	66	77
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

(default)

3DES: 1st half ≠ 2nd half ➡ **Triple DES**

00	11	22	33	44	55	66	77	88	99	aa	bb	cc	dd	ee	ff
00	00	00	00	00	00	00	00	00	00	00	00	00	01	00	00

4 different Access Rights are stored for each file.

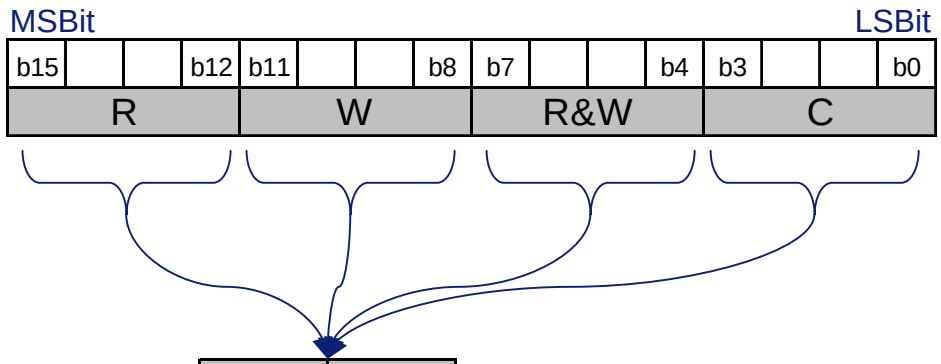
Access Rights are defined during creation of a file.

	Value File				all other files
R	Get Value	Debit			Read
W	Get Value	Debit	Limited Credit		Write
R&W	Get Value	Debit	Limited Credit	Credit	Read&Write
C	Change Config				

Key #0 always is the Masterkey
- on PICC level (if no application or AID 0x00 00 00 is selected)
- on Application level (if an Application is selected).

mifare® *DESFire Coding of* Access Rights

During creation of a file the Access Rights are defined with a 2-byte code:



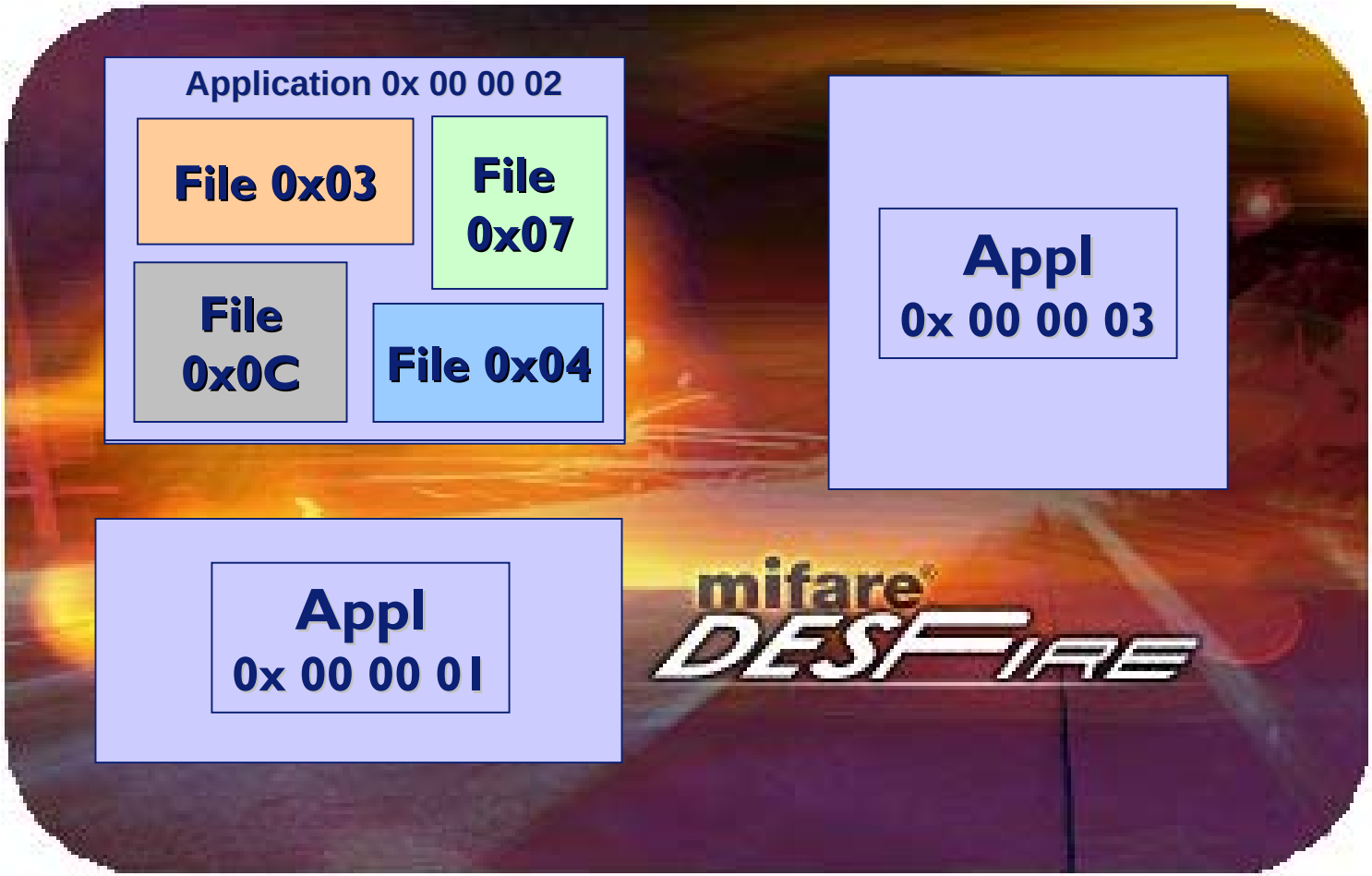
Hex	Key	
0x0	0	
0x1	1	
0x2	2	
0x3	3	
0x4	4	
0x5	5	
0x6	6	
0x7	7	
0x8	8	
0x9	9	
0xa	10	
0xb	11	
0xc	12	
0xd	13	
0xe	"free"	→ no authentication required
0xf	never	→ no access

Remark:
If a file is accessed without valid authentication but free access (0xe) is possible, the communication mode is forced to plain (through at least one relevant access right).

mifare® *DESFire* Application Example

Up to 28 different applications per card

Up to 16 different files per application



Application # 0x 00 00 02 contains 4 Files:

File 0x03:

Value File
lower limit: -10
upper limit: +2000
MACed Data

File 0x07:

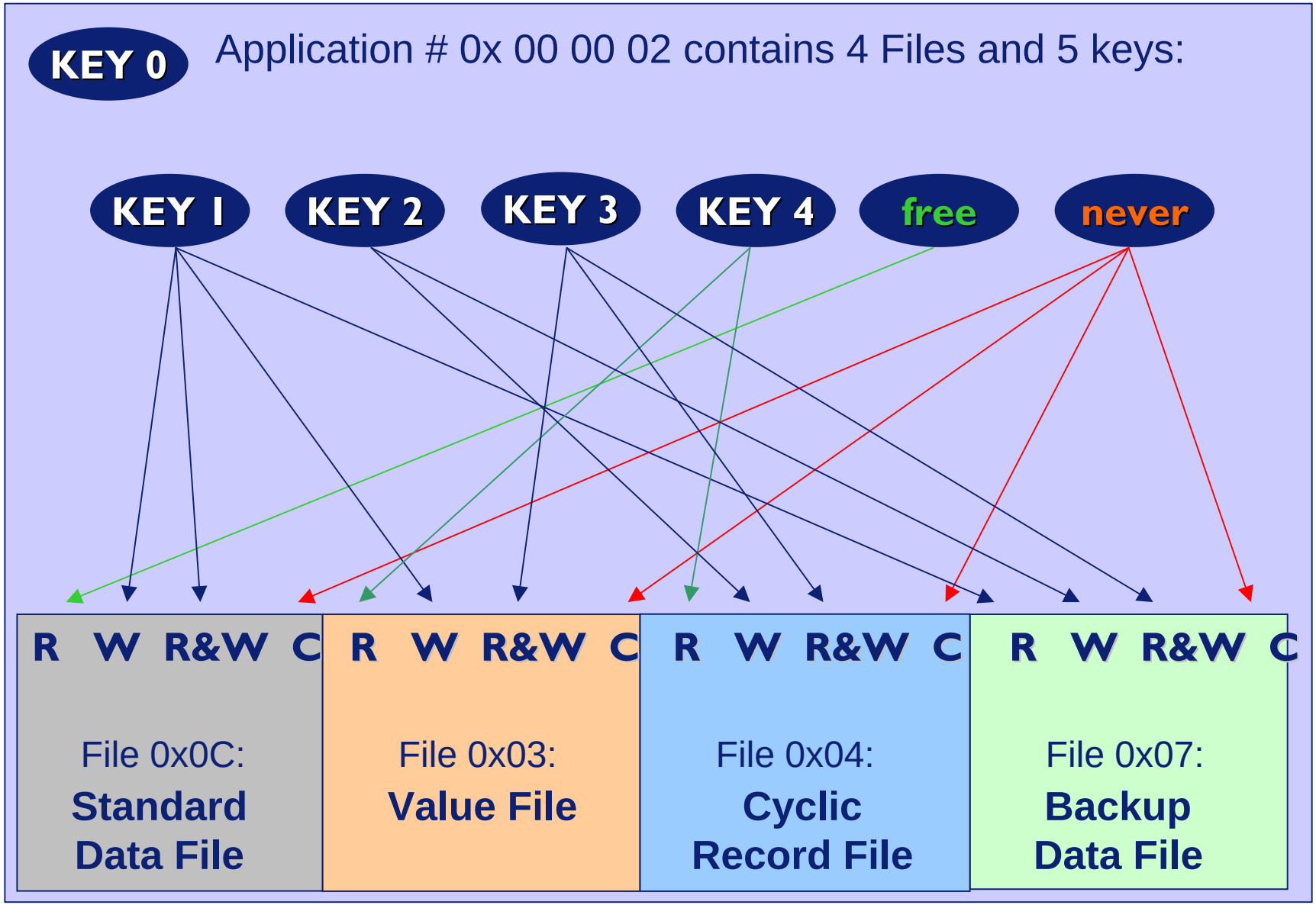
Backup Data File
File Size: 30 bytes
3DES encrypted Data

File 0x0C:

Standard Data File
File Size: 30 bytes
Plain Data

File 0x04:

Cyclic Record File
Record Size: 10 bytes
# of Records: 21
MACed Data


KEY 0
Application # 0x 00 00 02 contains 4 Files and 5 keys:
KEY 1
KEY 2
KEY 3
KEY 4
free
never
R W R&W C
File 0x0C:
Standard Data File
R W R&W C
File 0x03:
Value File
R W R&W C
File 0x04:
Cyclic Record File
R W R&W C
File 0x07:
Backup Data File

Transaction oriented approach

On application level, Multiple write commands can be issued.

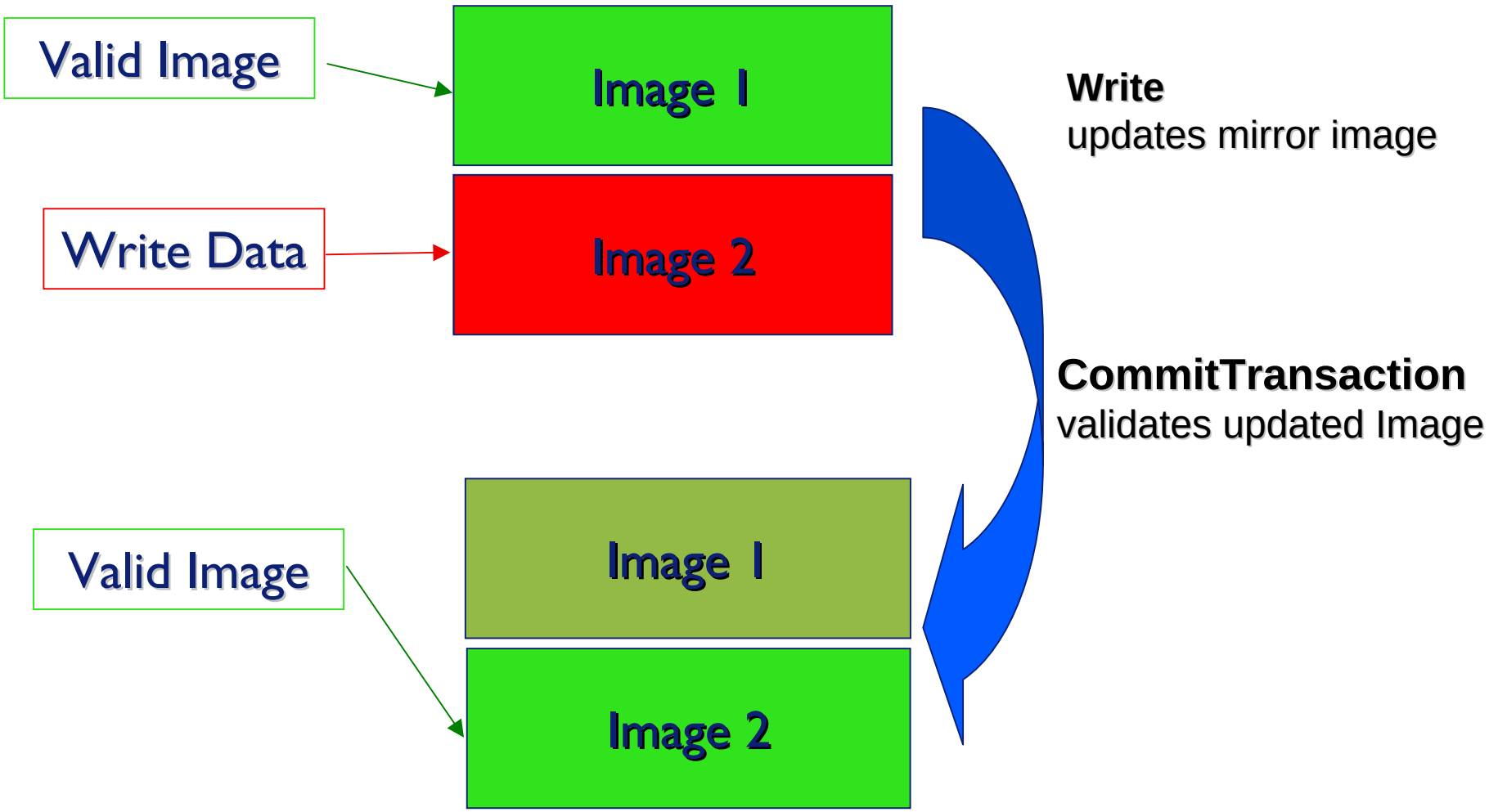
Completed transaction has to be validated by a CommitTransaction command.

If not validated or aborted, a full rollback of all writes happens.

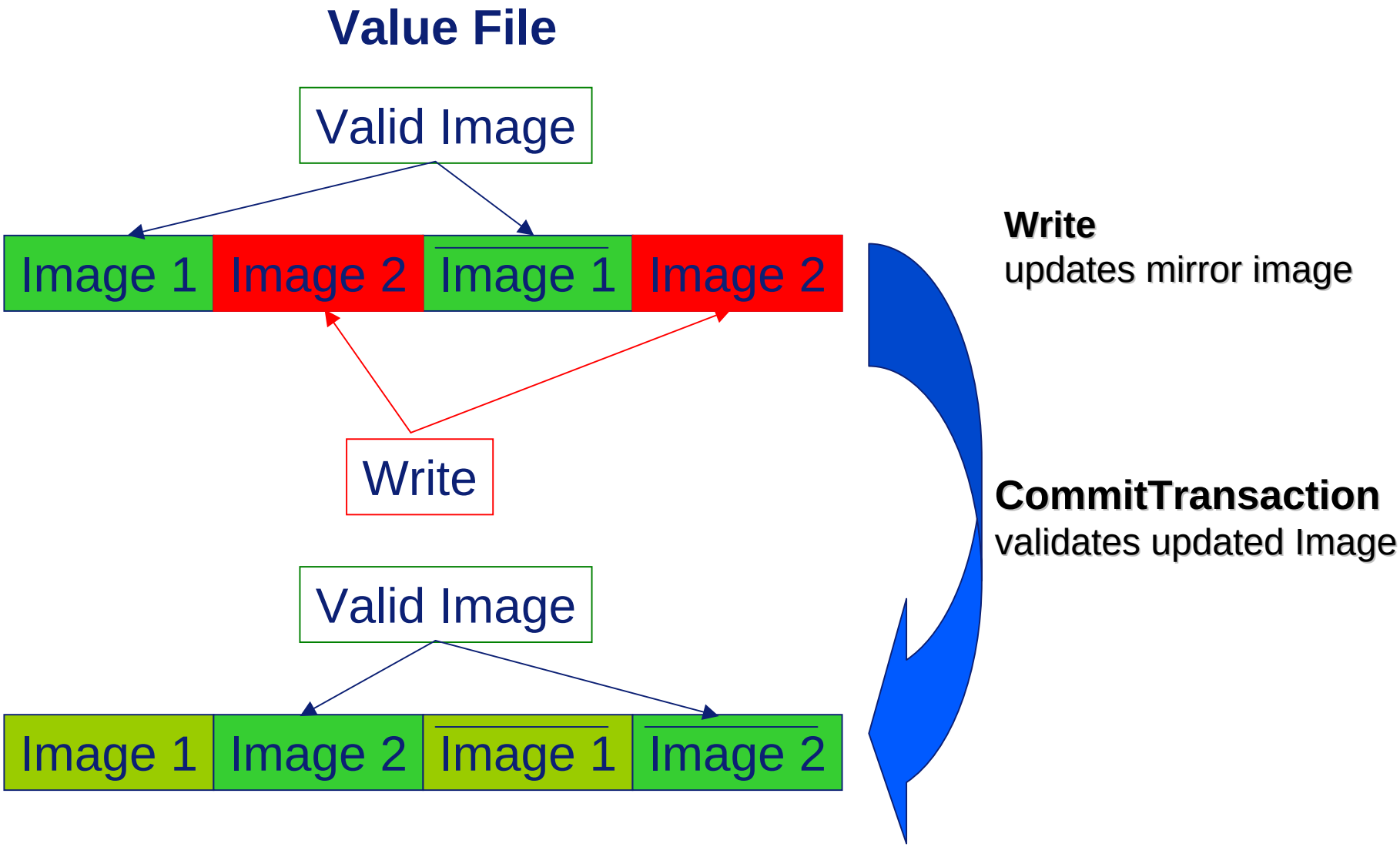
Either ALL writes are done or NO writes are done.

→ Application data is always consistent

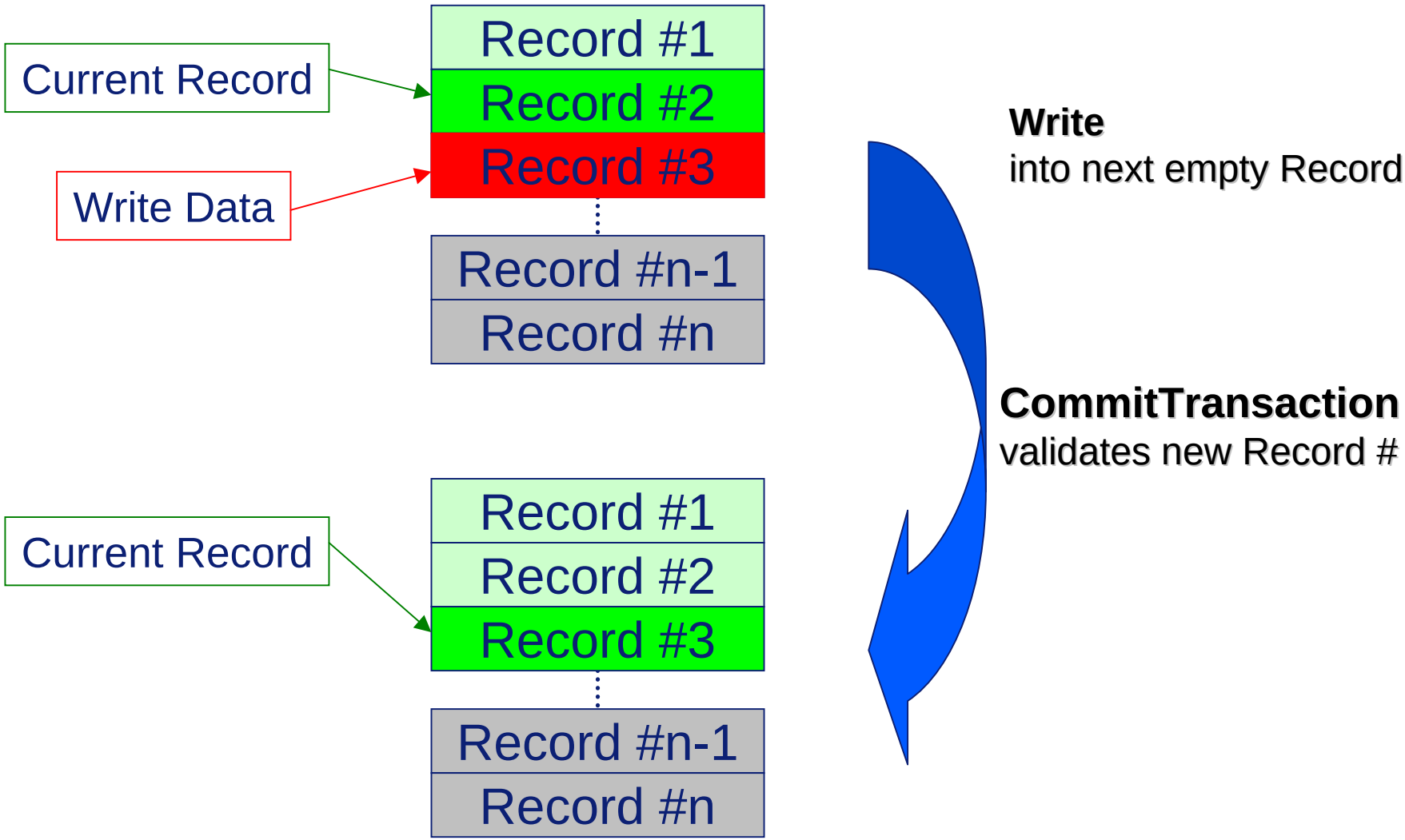
Backup Data File



*Data File Backup Management requires a Backup Data File.


Value File
Valid Image
Image 1
Image 2
Image 1
Image 2
Write
Valid Image
Image 1
Image 2
Image 1
Image 2
Write
updates mirror image
CommitTransaction
validates updated Image

Record File


Current Record
Write Data
Record #1
Record #2
Record #3
Record #n-1
Record #n
Write
into next empty Record
CommitTransaction
validates new Record #
Current Record
Record #1
Record #2
Record #3
Record #n-1
Record #n

The mifare® DES*Fire* EEPROM area is allocated in blocks of 32 bytes.

Blank Chip

The “blank” chip in delivery state uses 4 blocks for Manufacturer data and Administration.

Card Administration

The card administration requires 1 block per 4 created applications.
This memory is re-used after “Delete Application”.

Application

For each created application n blocks are required with:

$$n = 1 + \text{int}\left({}^{(keys+1)}/_{2} \right) blocks$$

number of keys	number of blocks
0	1
1	2
2	2
3	3
4	3
5	4
6	4
...	...

This memory cannot be re-used after “Delete Application”, but only after “FormatPICC”.

File Administration

Every 2nd file entry uses 1 block, beginning with 2nd generated file.

number of files	number of blocks
1	0
2	1
3	1
4	2
5	2
6	3
7	3
...	...

This memory is re-used after “Delete File”.

Data

The data of a **Standard Data File** requires n blocks with:

$$n = 1 + \mathrm{int}\left((filesize + 31) / 32 \right) blocks$$

	Standard Data File	Backup Data File
file size	number of blocks	number of blocks
1	1	2
2	1	2
…	1	2
32	1	2
33	2	4
34	2	4
…	2	4
64	2	4
65	3	6
…	…	…

The data of a **Backup Data File** requires 2x n blocks

The **Value Data File** requires 1 block, independent on value or limits.

The **Record File** requires n blocks with:

$$n = 1 + \mathrm{int}\left((recordsize \cdot number\_of\_records + 31) / 32 \right) blocks$$

mifare® *DESFire* Transaction Time

Command Sequence (typical transport transaction):

- –Establish protocol according to ISO 14443-4
  - •Application Selection
  - •mutual 3pass Authentication
  - •Read Standard Data File (48 bytes 3DES MACed)
  - •Read Backup Data File (48 bytes 3DES MACed)
  - •Read Value (12 bytes 3DES MACed)
  - •Read Record File (48 bytes 3DES MACed)
  - •Write to backup file (48 bytes 3DES MACed)
  - •Append record to record file (48 bytes 3DES MACed)
  - •Modify value file (12 bytes 3DES MACed)
  - •CommitTransaction
- –Deselect according to ISO 14443-4

Transaction Time 3DES MACed for Read 156 byte, Write 108 byte (incl. Backup):

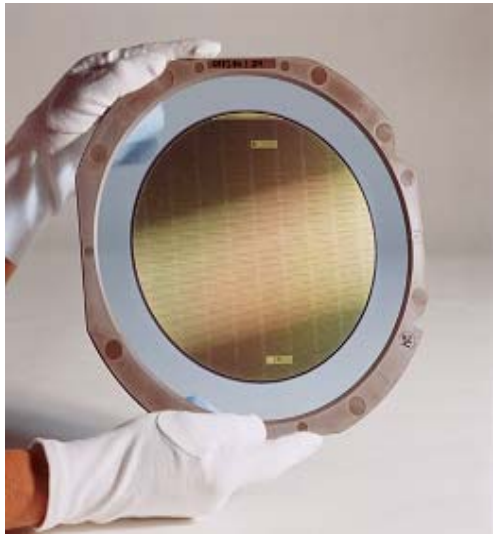
@ 106 kbaud: <130ms*

@ 212 kbaud: <110ms*

@ 424 kbaud: <100ms*

* Includes communication PCD - PICC, does NOT include reader data handling

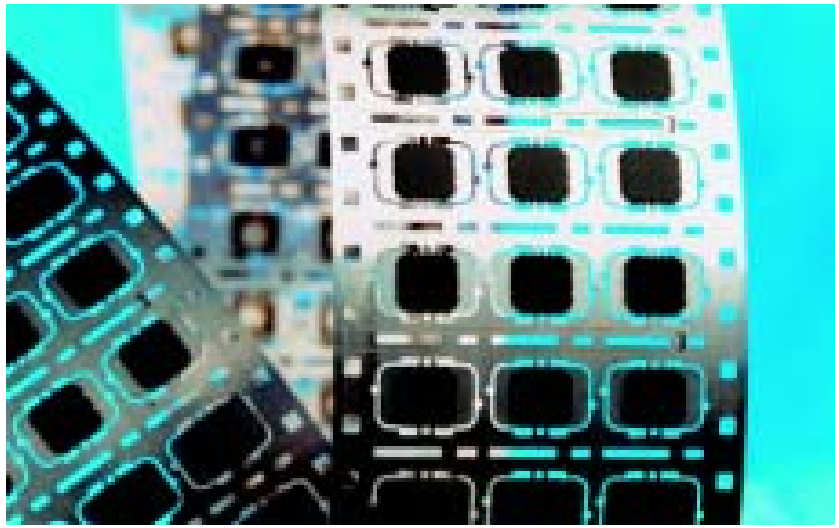
Sawn Wafer on FFC



150µm thickness

MF3ICD4001DW/V5

MOA4 Contactless Module



330µm thickness

MOA4: MF3MOD4001DV/4

MF EV70x

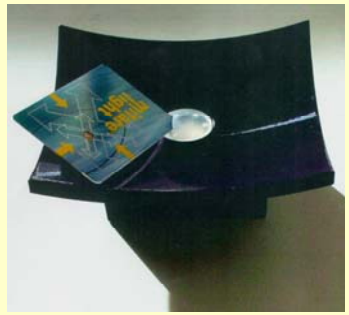
based on the Pegoda Reader, contains:

- USB Pegoda Reader (**RD70x**)
- Datasheets & Documents on a CD
- 5 Mifare Cards

+

- **mifare® *DESFire* Sample Cards**
- MF DESFire UI (Demo-SW)
- Debug Client SW
- C-library (incl. Source Code)

PEGODA Reader



Sample Cards

mifare® DESFire Facts

- Fully ISO 14443A compliant, up to part 4
- Unique 7 byte serial number ISO cascade level 2
- 4 KByte EEPROM, 1ms erase, 1ms program
- Fast Data Transfer, up to 424 Kbit/s
- Mutual Three Pass Authentication
- DES/3DES Data Encryption on RF-channel
- Data Authenticity by 4 byte 3DES MAC
- Flexible File System
- Up to 28 Applications per card
- Up to 14 3DES keys per Application, with key versioning
- Up to 16 Files per Application
- Automatic backup mechanism for all available file types


PHILIPS