

# Spécifications Techniques

## «Logiciel Coupleur»

## HISTORIQUE

Version	Date	Auteur	Révision
1.0	12/12/11	V. THIVENT	Création
1.1	02/01/12	V. THIVENT	Amendement
1.1	20/07/12	V. THIVENT	Amendement
1.2	05/12/12	V. THIVENT	Amendement
1.3	11/12/12	V. THIVENT	Calypso
1.4	21/01/12	V. THIVENT	Commandes lecteurs autonomes

# SOMMAIRE

<b>HISTORIQUE.....</b>	<b>2</b>
<b>1 Introduction.....</b>	<b>6</b>
1.1 But .....	6
1.2 Glossaire .....	6
<b>2 Présentation des lecteurs.....</b>	<b>7</b>
<b>3 Présentation du fonctionnement.....</b>	<b>8</b>
<b>4 Commandes du système.....</b>	<b>9</b>
4.1 Define .....	9
4.1.1 Type de carte.....	10
4.1.2 Type de protocole .....	11
4.1.3 Etat des LED et du Buzzer.....	11
4.1.4 Conditions d'accès pour les cartes MIFARE Classic.....	12
4.2 Code d'erreur .....	13
4.3 Les commandes proprent au lecteur .....	14
4.3.1 OpenCOM1.....	14
4.3.2 CloseCOM1.....	14
4.3.3 GetErrorMessage.....	14
4.3.4 GetLibrary.....	15
4.3.5 Version.....	15
4.3.6 RF_Power_Control.....	16
4.3.7 RF_Config_Card_Mode.....	16
4.3.8 LEDBuzzer.....	17
4.3.9 LCD.....	17
4.4 Prise de contact .....	18
4.4.1 ISO14443_3_A_PollCard.....	18
4.4.2 ISO14443_3_A_PollCardWU.....	18
4.4.3 ISO14443_3_A_Halt.....	19
4.5 Gestion des SAM .....	20
4.5.1 SAM_Config_Card_Mode.....	20
4.5.2 SAM_SelectSlot.....	20
4.5.3 SAM_GetATR.....	21
4.6 MIFARE Ultralight .....	22
4.6.1 Mf_Classic_Read_Block.....	22
4.6.2 Mf_Classic_Write_Block.....	23
4.7 MIFARE Classic .....	24

4.7.1	Mf_Classic_LoadKey.....	25
4.7.2	Mf_Classic_Authenticate.....	25
4.7.3	Mf_Classic_Read_Block.....	26
4.7.4	Mf_Classic_Write_Block.....	27
4.7.5	Mf_Classic_Read_Sector.....	28
4.7.6	Mf_Classic_Write_Sector.....	29
4.7.7	Mf_Classic_UpdateAccessBlock.....	30
4.7.8	Mf_Classic_Read_Value.....	31
4.7.9	Mf_Classic_Write_Value.....	32
4.7.10	Mf_Classic_Increment_Value.....	33
4.7.11	Mf_Classic_Decrement_Value.....	34
4.7.12	Mf_Classic_Restore_Value.....	35
4.8	MIFARE DESFire .....	36
4.8.1	Mf_DESFire_AuthenticateHost.....	37
4.8.2	Mf_DESFire_CreateApplication.....	38
4.9	MIFARE SAM AV2 .....	39
4.10	ST SR176 et SRixK .....	39
4.11	ASK CTS et CTM .....	39
4.12	ISO15693 .....	39
4.13	Calypso .....	39
4.13.1	Calypso_SelectApplication.....	39
4.13.2	Calypso_AppendRecord.....	40
4.13.3	Calypso_Decrease.....	40
4.13.4	Calypso_DecreaseMultiple.....	41
4.13.5	Calypso_Increase.....	41
4.13.6	Calypso_IncreaseMultiple.....	41
4.13.7	Calypso_ReadRecord.....	41
4.13.8	Calypso_UpdateRecord.....	42
4.13.9	Calypso_WriteRecord.....	43
4.13.10	Calypso_OpenSecureSession1.....	43
4.13.11	Calypso_OpenSecureSession2.....	44
4.13.12	Calypso_OpenSecureSession3.....	45
4.13.13	Calypso_CloseSecureSession.....	46
4.13.14	Calypso_Invalidate.....	46
4.13.15	SAM_Calypso_SelectDiversifer.....	47
4.13.16	SAM_Calypso_GetChallenge.....	47
4.13.17	SAM_Calypso_DigestInitOld.....	48
4.13.18	SAM_Calypso_DigestUpdate.....	48
4.13.19	SAM_Calypso_DigestClose.....	48
4.13.20	SAM_Calypso_DigestAuthenticate.....	49
4.14	Lecteur autonome .....	50

4.14.1 AutoReader_SetRTC.....	52
4.14.2 AutoReader_AddUser.....	52
4.14.3 AutoReader_DeleteUser.....	53
4.14.4 AutoReader_ClearFullUser.....	53
4.14.5 AutoReader_LoadKeyMifare.....	54
4.14.6 AutoReader_AccessMifare.....	54
4.14.7 AutoReader_ComptMifare.....	55
4.14.8 AutoReader_AfficheValueMifare.....	56
4.14.9 AutoReader_TempoGache.....	56
4.14.10 AutoReader_BlackList.....	57
4.14.11 AutoReader_SendInfo.....	57
4.14.12 AutoReader_LCD1.....	58
4.14.13 AutoReader_LCD2A.....	58
4.14.14 AutoReader_LCD2B.....	59
4.14.15 AutoReader_LCD2C.....	59
<b>5 Contact.....</b>	<b>59</b>

# 1 Introduction

## 1.1 But







































































Ce document présente les spécifications techniques logicielles du lecteur.

## 1.2 Glossaire

UID	Identifiant Unique
EF	Fichier Elementaire
DF	Fichier Dédié

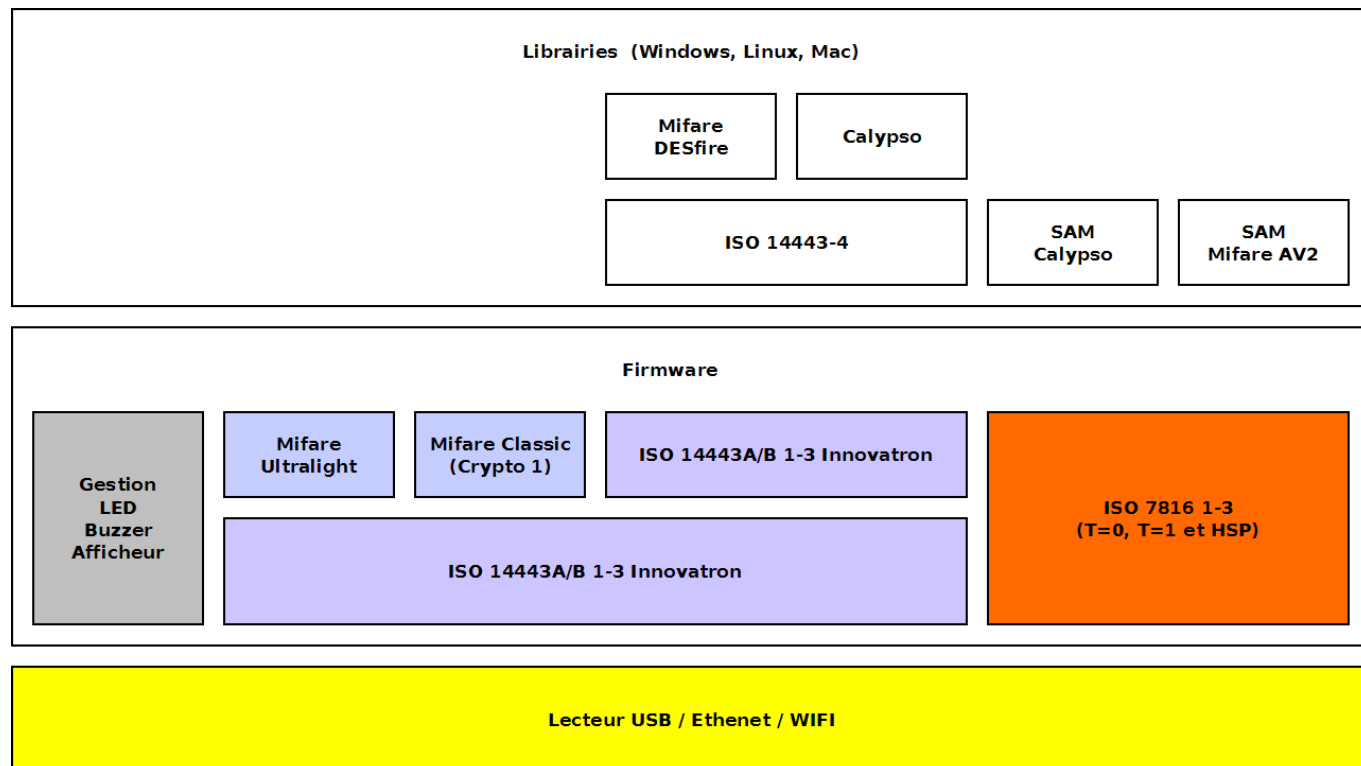
## 2 Présentation des lecteurs

ODALID propose différents types de lecteurs avec des fonctionnalités différentes :

	Clé USB NFC	Coupleur USB CDC	Coupleur USB CDC 2 SAM	Coupleur TCP/IP	Coupleur TCP/IP 2 SAM	Lecteur autonome TCP/IP
Interface Host	USB PC/SC	USB CDC (émulation série)	USB CDC (émulation série)	TCP/IP	TCP/IP	TCP/IP
LED						
Afficheur						
Buzzer						
Mifare Ultralight						
Mifare Classic						
Mifare DESfire						
Mifare DESfire SAM						
SR176						
SRixK						
ISO15693		Option		Option		
Calypso						
Calypso SAM						

### 3 Présentation du fonctionnement

Tous les lecteurs fonctionnent avec la même librairie. Pour chaque lecteur les commandes sont identiques.





## 4 Commandes du système

### 4.1 Define

Dans ce paragraphe sont définies les différentes valeurs suivantes :

#### 4.1.1 Type de carte

TypeA	Modulation de type ISO14443A
TypeB	Modulation de type ISO14443B
TypeCTS	Modulation de type CTS
TypeST	Modulation de type ST
TypeInno	Modulation de type Innovatron
Type15693	Modulation de type ISO15693

#### 4.1.2 Type de protocole

TypeISO_T0	Protocole T=0 (ISO7816)
TypeISO_T1	Protocole T=1 (ISO7816)
TypeHSP	Protocole HSP (Calypso)

#### 4.1.3 Etat des LED et du Buzzer

LED_ON	Active toutes les LED
LED_OFF	Désactive toutes les LED
LED_GREEN_OFF	Active toutes la LED Verte
LED_GREEN_ON	Désactive la LED Verte
LED_YELLOW_OFF	Active les la LED jaune
LED_YELLOW_ON	Désactive la LED jaune
LED_RED_OFF	Active les la LED rouge
LED_RED_ON	Désactive la LED rouge
BUZZER_OFF	Active le buzzer
BUZZER_ON	Désactive le buzzer

#### 4.1.4 Conditions d'accès pour les cartes MIFARE Classic

- Conditions d'accès pour les blocks de données (block 0 à 2)

	C1	C2	C3	Read	Write	Increment	Decrement, transfert, restore	
ACC_BLOCK_TRANSPORT	0	0	0	key A B	key A B	key A B	key A B	Transport Configuration
ACC_BLOCK_READWRITE	1	0	0	key A B	Key B	never	never	read/write block
ACC_BLOCK_VALUE	1	1	0	key A B	Key B	Key B	key A B	value block

- Conditions d'accès pour les blocks de sécurité (block 3)

	C1	C2	C3	KEYA		Access bits		KEYB	
				read	write	read	write	read	write
ACC_AUTH_TRANSPORT	0	0	1	never	key A	key A	key A	key A	key A
ACC_AUTH_NORMAL	0	1	1	never	key B	key A B	key B	never	key B

## 4.2 Code d'erreur

## 4.3 Les commandes proprent au lecteur

### 4.3.1 OpenCOM1

**NAME**

OpenCOM1

**DESCRIPTION**

Ouvre la connexion entre le lecteur et l'ordinateur

**INPUTS**

ReaderName \*Name : Information sur le lecteur

**RETURNS**

Status : 0 ou erreur

### 4.3.2 CloseCOM1

**NAME**

CloseCOM1

**DESCRIPTION**

Ferme la connexion entre le lecteur et l'ordinateur

**INPUTS**

ReaderName \*Name : Information sur le lecteur

**RETURNS**

Status : 0 ou erreur

### 4.3.3 GetErrorMessage

**NAME**

CloseCOM1

**DESCRIPTION**

Retourne le message d'erreur en chaîne de caractères

**INPUTS**

Status : Statut de l'erreur

**RETURNS**

Chaîne de caractères sur le message d'erreur

#### 4.3.4 GetLibrary

##### NAME

GetLibrary

##### DESCRIPTION

Retourne la version de la librairie

##### INPUTS

char \*recv : chaîne de caractères qui contient la version de la librairie  
uint16\_t \*len\_recv : taille de la chaîne de caractères

##### RETURNS

Status : 0 ou erreur

#### 4.3.5 Version

##### NAME

Version

##### DESCRIPTION

Retourne la version du lecteur

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
char \*version : chaîne de caractères qui contient la version du firmware  
uint8\_t \*serial : Numéro de série du lecteur sur 4 octets  
char \*stack : chaîne de caractères qui contient la version de la stack

##### RETURNS

Status : 0 ou erreur

### 4.3.6 RF\_Power\_Control

#### NAME

RF\_Power\_Control

#### DESCRIPTION

Active ou désactive le champ RF

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL RFOff : Active ou désactive le champ RF  
     TRUE → RF  
     FALSE → Pas de RF  
 uint8\_t Delay : indique le temps du lecteur dans cet état  
     0 → infini  
     10 → 10 ms (exemple)

#### RETURNS

Status : 0 ou erreur

### 4.3.7 RF\_Config\_Card\_Mode

#### NAME

RF\_Config\_Card\_Mode

#### DESCRIPTION

Sélectionne le type de modulation en fonction du type de carte avec laquelle on veut communiquer. (Cette commande doit être utilisée pour les commandes transparentes).

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 int8\_t Type : Type de carte à utiliser voir paragraphe Define

#### RETURNS

Status : 0 ou erreur



### 4.3.8 LEDBuzzer

#### NAME

LEDBuzzer

#### DESCRIPTION

Permet d'activer ou de désactiver les LED et/ou le Buzzer

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 int8\_t LEDBuzzer : Active ou désactive les LED et/ou le Buzzer voir paragraphe Define

#### RETURNS

Status : 0 ou erreur

### 4.3.9 LCD

#### NAME

LCD

#### DESCRIPTION

Permet d'afficher des informations à l'écran du lecteur

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 int8\_t line : ligne de l'écran  
 char \*send : chaîne de 16 caractères contenant le texte à afficher

#### RETURNS

Status : 0 ou erreur

## 4.4 Prise de contact

### 4.4.1 ISO14443\_3\_A\_PollCard

#### NAME

ISO14443\_3\_A\_PollCard

#### DESCRIPTION

Effectue la prise de contact de la carte selon la norme ISO14443A avec un Request standard (la carte ne répond pas après la commande Halt).

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t \*atq : 2 octets "Answer To Query"  
 uint8\_t \*sak : 1 octet "Select Acknowledge"  
 uint8\_t \*uid : 10 octets UID  
 uint16\_t \*uid\_len : longueur de l'UID

#### RETURNS

Status : 0 ou erreur

### 4.4.2 ISO14443\_3\_A\_PollCardWU

#### NAME

ISO14443\_3\_A\_PollCardWU

#### DESCRIPTION

Effectue la prise de contact de la carte selon la norme ISO14443A avec un Request All (la carte répond après la commande Halt).

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t \*atq : 2 octets "Answer To Query"  
 uint8\_t \*sak : 1 octet "Select Acknowledge"  
 uint8\_t \*uid : 10 octets UID  
 uint16\_t \*uid\_len : longueur de l'UID

#### RETURNS

Status : 0 ou erreur

### 4.4.3 ISO14443\_3\_A\_Halt

**NAME**

ISO14443\_3\_A\_Halt

**DESCRIPTION**

Désactive la carte par la commande HALT

**INPUTS**

ReaderName \*Name : Information sur le lecteur

**RETURNS**

Status : 0 ou erreur

## 4.5 Gestion des SAM

### 4.5.1 SAM\_Config\_Card\_Mode

**NAME**

SAM\_Config\_Card\_Mode

**DESCRIPTION**

Sélectionne le type de protocole

**INPUTS**

ReaderName \*Name : Information sur le lecteur  
uint8\_t Type : Type de protocole (voir define)

**RETURNS**

Status : 0 ou erreur

### 4.5.2 SAM\_SelectSlot

**NAME**

SAM\_SelectSlot

**DESCRIPTION**

Sélectionne le Slot du SAM

**INPUTS**

ReaderName \*Name : Information sur le lecteur  
uint8\_t Slot : Numero du slot

**RETURNS**

Status : 0 ou erreur

### 4.5.3 SAM\_GetATR

**NAME**

SAM\_GetATR

**DESCRIPTION**

Demande d'ATR au SAM

**INPUTS**

ReaderName \*Name : Information sur le lecteur  
uint8\_t \*atr : Reponses du SAM  
uint8\_t atrlen : longueur de la reponse

**RETURNS**

Status : 0 ou erreur

## 4.6 MIFARE Ultralight

Les commandes dédiées aux cartes MIFARE Ultralight sont celles utilisées pour les cartes MIFARE Classic, il faudra désactiver l'authentification (auth =FALSE).

### 4.6.1 Mf\_Classic\_Read\_Block

#### NAME

Mf\_Classic\_Read\_Block

#### DESCRIPTION

Lecture d'un block de la carte MIFARE ultralight

#### INPUTS

ReaderName *Name	: Information sur le lecteur
BOOL auth	: Authentification FALSE → pas d'authentification
uint8_t block	: Bloc de la carte MIFARE Ultralight
uint8_t *data	: Données (4 octets)
BOOL Auth_Key	: Clé d'authentification Clé A → TRUE Clé B → FALSE valeur ignorée si auth = FALSE
uint8_t key_index	: Index de la clé stocké dans le EEPROM du coupleur [0-15] valeur ignorée si auth = FALSE

#### RETURNS

Status : 0 ou erreur

## 4.6.2 Mf\_Classic\_Write\_Block

### NAME

Mf\_Classic\_Write\_Block

### DESCRIPTION

Écriture d'un block de la carte MIFARE Ultralight

### INPUTS

ReaderName *Name	: Information sur le lecteur
BOOL auth	: Authentification
	FALSE → pas d'authentification
uint8_t block	: Bloc de la carte MIFARE Ultralight
uint8_t *data	: Données à écrire (4 octets)
BOOL Auth_Key	: Clé d'authentification
	Clé A → TRUE
	Clé B → FALSE
	valeur ignoré si auth = FALSE
uint8_t key_index	: Index de la clé stocké dans le EEPROM du coupleur
	[0-15]
	valeur ignorée si auth = FALSE

### RETURNS

Status : 0 ou erreur

## 4.7 MIFARE Classic

Pour la carte MIFARE Classic deux méthodes d'authentification peuvent être utilisées :

- On peut authentifier la carte MIFARE Classic directement avec la commande `Mf_Classic_Authenticate` et la clé de 6 octets
- Ou en chargeant la clé dans l'EEPROM sécurisée du lecteur grâce à la commande `Mf_Classic_LoadKey`.

Les clé sont stockées comme suit dans l'EEPROM sécurisée du lecteur

Index	KeyA	KeyB
0	KeyA_0	KeyB_0
1	KeyA_1	KeyB_1
2	KeyA_2	KeyB_2
3	KeyA_3	KeyB_3
4	KeyA_4	KeyB_4
5	KeyA_5	KeyB_5
6	KeyA_6	KeyB_6
7	KeyA_7	KeyB_7
8	KeyA_8	KeyB_8
9	KeyA_9	KeyB_9
10	KeyA_10	KeyB_10
11	KeyA_11	KeyB_11
12	KeyA_12	KeyB_12
13	KeyA_13	KeyB_13
14	KeyA_14	KeyB_14
15	KeyA_15	KeyB_15

Lorsque les clé MIFARE Classic sont stockées dans l'EEPROM sécurisée on peut utiliser les commandes d'accès au fichier `Read`, `Write`, `Decrement`, `Increment` et `Restore` avec l'authentification automatique. Dans ce cas c'est le lecteur qui se charge d'authentifier la carte MIFARE Classic.

Les commandes `Decrement`, `Increment` et `Restore` ont directement intégré la commande `Transfer`.



### 4.7.1 Mf\_Classic\_LoadKey

#### NAME

Mf\_Classic\_LoadKey

#### DESCRIPTION

Chargement des clés MIFARE dans l'EEPROM du lecteur

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL Auth\_Key : clé d'authentification  
                   TRUE → KeyA  
                   FALSE → KeyB  
 uint8\_t \*key : clé sur 6 octet  
 uint8\_t key\_index : index de l'EEPROM (0 à 15)

#### RETURNS

Status : 0 ou erreur

### 4.7.2 Mf\_Classic\_Authenticate

#### NAME

Mf\_Classic\_Authenticate

#### DESCRIPTION

Phase d'authentification pour la carte MIFARE Classic

#### INPUTS

ReaderName \*Name : information sur le lecteur  
 BOOL Auth\_Key : clé d'authentification  
                   TRUE → KeyA  
                   FALSE → KeyB  
 BOOL internal\_Key : authentification via les clés contenues dans le lecteur  
                   TRUE → OK  
                   FALSE → utilisation de la chaîne Key  
 uint8\_t sector : secteur de la carte  
                   MIFARE Classic 1K [0-15]  
                   MIFARE Classic 4K[0-39]  
 uint8\_t \*key : clé MIFARE [6 octets] ignorée si internal\_Key = TRUE  
 uint8\_t key\_index I : index de la clé stocké en interne dans le coupleur  
                   [0-15] ignorée si internal\_Key = FALSE

#### RETURNS

Status : 0 ou erreur

### 4.7.3 Mf\_Classic\_Read\_Block

#### NAME

Mf\_Classic\_Read\_Block

#### DESCRIPTION

Lecture d'un block de la carte MIFARE Classic

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL auth : Authentification automatique  
                   TRUE → authentification automatique dans le lecteur  
                   FALSE → pas d'authentification automatique  
 uint8\_t block : Bloc de la carte MIFARE  
 uint8\_t \*data : Données lu (16 octets)  
 BOOL Auth\_Key : Clé d'authentification  
                   Clé A → TRUE  
                   Clé B → FALSE  
                   valeur ignorée si auth = FALSE  
 uint8\_t key\_index : Index de la clé stocké dans le EEPROM du coupleur  
                   [0-15]  
                   valeur ignorée si auth = FALSE

#### RETURNS

Status : 0 ou erreur

#### 4.7.4 Mf\_Classic\_Write\_Block

##### NAME

Mf\_Classic\_Write\_Block

##### DESCRIPTION

Écriture d'un block de la carte MIFARE Classic

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL auth : Authentification automatique  
     TRUE → authentification automatique dans le lecteur  
     FALSE → pas d'authentification automatique  
 uint8\_t block : Bloc de la carte MIFARE  
 uint8\_t \*data : Données à écrire (16 octets)  
 BOOL Auth\_Key : Clé d'authentification  
     Clé A → TRUE  
     Clé B → FALSE  
     valeur ignorée si auth = FALSE  
 uint8\_t key\_index : Index de la clé stocké dans le EEPROM du coupleur  
     [0-15]  
     valeur ignorée si auth = FALSE

##### RETURNS

Status : 0 ou erreur

## 4.7.5 Mf\_Classic\_Read\_Sector

### NAME

Mf\_Classic\_Read\_Sector

### DESCRIPTION

Lecture d'un Secteur de la carte MIFARE Classic

### INPUTS

ReaderName \*Name : Information sur le lecteur  
BOOL auth : Authentification automatique  
TRUE → authentification automatique dans le lecteur  
FALSE → pas d'authentification automatique  
uint8\_t block : Bloc de la carte MIFARE  
uint8\_t \*data : Données lu ici 48 octets  
BOOL Auth\_Key : Clé d'authentification  
Clé A → TRUE  
Clé B → FALSE  
valeur ignorée si auth = FALSE  
uint8\_t key\_index : Index de la clé stocké dans le EEPROM du coupleur  
[0-15]  
valeur ignorée si auth = FALSE

### RETURNS

Status : 0 ou erreur

## 4.7.6 Mf\_Classic\_Write\_Sector

### NAME

Mf\_Classic\_Write\_Sector

### DESCRIPTION

Ecriture d'un Secteur de la carte MIFARE Classic

### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL auth : Authentification automatique  
     TRUE → authentification automatique dans le lecteur  
     FALSE → pas d'authentification automatique  
 uint8\_t block : Bloc de la carte MIFARE Classic  
 uint8\_t \*data : Données à écrire ici 48 octets  
 BOOL Auth\_Key : Clé d'authentification  
     Clé A → TRUE  
     Clé B → FALSE  
     valeur ignorée si auth = FALSE  
 uint8\_t key\_index : Index de la clé stocké dans le EEPROM du coupleur  
     [0-15]  
     valeur ignorée si auth = FALSE

### RETURNS

Status : 0 ou erreur

## 4.7.7 Mf\_Classic\_UpdateAccessBlock

### NAME

Mf\_Classic\_UpdateAccessBlock

### DESCRIPTION

Mise à jour des conditions d'accès et des clés de sécurité

### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL auth : Authentification automatique  
     TRUE → authentification automatique dans le lecteur  
     FALSE → pas d'authentification automatique  
 uint8\_t sector : Secteur de la carte MIFARE Classic  
 uint8\_t old\_key\_index : Index de la clé stocké dans le EEPROM du coupleur  
     [0-15]  
     valeur ignorée si auth = FALSE  
 uint8\_t \*new\_key\_A : Nouvelle Clé A 6 octets  
 uint8\_t \*new\_key\_B : Nouvelle Clé B 6 octets  
 uint8\_t bc0 : Conditions d'accès pour le block 0  
 uint8\_t bc1 : Conditions d'accès pour le block 1  
 uint8\_t bc2 : Conditions d'accès pour le block 2  
 uint8\_t bc3 : Conditions d'accès pour le block 3  
 BOOL Auth\_Key : Clé d'authentification  
     Clé A → TRUE  
     Clé B → FALSE  
     valeur ignoré si auth = FALSE

### RETURNS

Status : 0 ou erreur

## 4.7.8 Mf\_Classic\_Read\_Value

### NAME

Mf\_Classic\_Read\_Value

### DESCRIPTION

Lecture de la valeur contenue dans le block de la carte MIFARE Classic

### INPUTS

ReaderName *Name	: Information sur le lecteur
BOOL auth	: Authentification automatique TRUE → authentification automatique dans le lecteur FALSE → pas d'authentification automatique
uint8_t block	: Bloc de la carte MIFARE Classic
uint32_t *value	: valeur lue
BOOL Auth_Key	: Clé d'authentification Clé A → TRUE Clé B → FALSE valeur ignorée si auth = FALSE
uint8_t key_index	: Index de la clé stocké dans le EEPROM du coupleur [0-15] valeur ignorée si auth = FALSE

### RETURNS

Status : 0 ou erreur

## 4.7.9 Mf\_Classic\_Write\_Value

### NAME

Mf\_Classic\_Write\_Sector

### DESCRIPTION

Écriture d'une valeur dans un block de la carte MIFARE Classic

### INPUTS

ReaderName *Name	: Information sur le lecteur
BOOL auth	: Authentification automatique TRUE → authentification automatique dans le lecteur FALSE → pas d'authentification automatique
uint8_t block	: Bloc de la carte MIFARE Classic
uint32_t value	: valeur à écrire
BOOL Auth_Key	: Clé d'authentification Clé A → TRUE Clé B → FALSE valeur ignorée si auth = FALSE
uint8_t key_index	: Index de la clé stocké dans le EEPROM du coupleur [0-15] valeur ignorée si auth = FALSE

### RETURNS

Status : 0 ou erreur



#### 4.7.10 Mf\_Classic\_Increment\_Value

##### NAME

Mf\_Classic\_Increment\_Value

##### DESCRIPTION

Incrément de la valeur du block de la carte MIFARE Classic

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL auth : Authentification automatique  
     TRUE → authentification automatique dans le lecteur  
     FALSE → pas d'authentification automatique  
 uint8\_t block : Bloc de la carte MIFARE Classic sur lequel sera effectué l'opération  
 uint8\_t trans\_block : Bloc dans lequel la valeur sera transmise  
 BOOL Auth\_Key : Clé d'authentification  
     Clé A → TRUE  
     Clé B → FALSE  
     valeur ignorée si auth = FALSE  
 uint8\_t key\_index : Index de la clé stocké dans le EEPROM du coupleur [0-15]  
     valeur ignorée si auth = FALSE

##### RETURNS

Status : 0 ou erreur

#### 4.7.11 Mf\_Classic\_Decrement\_Value

##### NAME

Mf\_Classic\_Decrement\_Value

##### DESCRIPTION

Décrémentation du block de la carte MIFARE Classic

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL auth : Authentification automatique  
                   TRUE → authentification automatique dans le lecteur  
                   FALSE → pas d'authentification automatique  
 uint8\_t block : Bloc de la carte MIFARE Classic sur lequel sera effectué l'opération  
 uint8\_t trans\_block : Bloc dans lequel la valeur sera transmise  
 BOOL Auth\_Key : Clé d'authentification  
                   Clé A → TRUE  
                   Clé B → FALSE  
                   valeur ignorée si auth = FALSE  
 uint8\_t key\_index : Index de la clé stocké dans le EEPROM du coupleur [0-15]  
                   valeur ignorée si auth = FALSE

##### RETURNS

Status : 0 ou erreur

## 4.7.12 Mf\_Classic\_Restore\_Value

### NAME

Mf\_Classic\_Restore\_Value

### DESCRIPTION

Restauration de la valeur

### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL auth : Authentification automatique  
     TRUE → authentification automatique dans le lecteur  
     FALSE → pas d'authentification automatique  
 uint8\_t block : Bloc de la carte MIFARE Classic sur lequel sera effectué l'opération  
 uint8\_t trans\_block : Bloc dans lequel la valeur sera transmise  
 BOOL Auth\_Key : Clé d'authentification  
     Clé A → TRUE  
     Clé B → FALSE  
     valeur ignorée si auth = FALSE  
 uint8\_t key\_index : Index de la clé stocké dans le EEPROM du coupleur [0-15]  
     valeur ignorée si auth = FALSE

### RETURNS

Status : 0 ou erreur

## 4.8 MIFARE DESFire

Les cartes DESFire possèdent deux générations les EV0 et EV1, voici ci-dessous un tableau qui différencie les deux types.

	<b>MIFARE DESFire EV0</b>	<b>MIFARE DESFire EV1</b>
<b>Fréquence</b>	13,56 Mhz	13,56 Mhz
<b>Débit</b>	106 kbit/s, 212 kbit/s, 424 kbit/s	106 kbit/s, 212 kbit/s, 424 kbit/s, <b>848 kbit/s</b>
<b>Protocole</b>	ISO/IEC 14443-4 Type A	ISO/IEC 14443-4 Type A
<b>UID</b>	7 octets	7 octets
	DES/3DES 56/112 bit key	DES/3DES 56/112/ <b>168</b> bit key <b>AES 128 bit key</b>

## 4.8.1 Mf\_DESFire\_AuthenticateHost

### NAME

Mf\_DESFire\_AuthenticateHost

### DESCRIPTION

Authentification

### INPUTS

ReaderName *Name	: Information sur le lecteur
BOOL auth	: Authentification automatique TRUE → authentification automatique dans le lecteur FALSE → pas d'authentification automatique
uint8_t block	: Bloc de la carte MIFARE Classic sur lequel sera effectué l'opération
uint8_t trans_block	: Bloc dans lequel la valeur sera transmise
BOOL Auth_Key	: Clé d'authentification Clé A → TRUE Clé B → FALSE valeur ignorée si auth = FALSE
uint8_t key_index	: Index de la clé stocké dans le EEPROM du coupleur [0-15] valeur ignorée si auth = FALSE

### RETURNS

Status : 0 ou erreur

## 4.8.2 Mf\_DESFire\_CreateApplication

### NAME

Mf\_DESFire\_CreateApplication

### DESCRIPTION

Créer une application dans la carte MIFARE DESFire

### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint32\_t aid : Authentification automatique  
                   TRUE → authentification automatique dans le lecteur  
                   FALSE → pas d'authentification automatique  
 uint8\_t Key1 : Bloc de la carte MIFARE Classic sur lequel sera effectué  
                   l'opération  
 uint8\_t Key1 : Bloc dans lequel la valeur sera transmise

### RETURNS

Status : 0 ou erreur

## 4.9 MIFARE SAM AV2

## 4.10 ST SR176 et SRixK

## 4.11 ASK CTS et CTM

## 4.12 ISO15693

## 4.13 Calypso

### 4.13.1 Calypso\_SelectApplication

#### NAME

Calypso\_SelectApplication

#### DESCRIPTION

Sélectionne l'application Calypso

#### INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Type	: Type de carte sans contact
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso
uint8_t *AID	: AID de l'application
uint8_t AIDlen	: longueur de l'AID
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso

#### RETURNS

Status : 0 ou erreur

### 4.13.2 Calypso\_AppendRecord

#### NAME

Calypso\_AppendRecord

#### DESCRIPTION

Ajoute un enregistrement à un EF circulaire

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t Type : Type de carte sans contact  
 CALYPSO\_INFO \*pInfo\_Card : structure d'information Calypso  
 uint8\_t \*AID : AID de l'application  
 uint8\_t AIDlen : longueur de l'AID  
 CALYPSO\_INFO \*pInfo\_Card : structure d'information Calypso

#### RETURNS

Status : 0 ou erreur

### 4.13.3 Calypso\_Decrease

#### NAME

Calypso\_Decrease

#### DESCRIPTION

Décrémente la valeur contenu dans le fichier compteur

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t Type : Type de carte sans contact  
 CALYPSO\_INFO \*pInfo\_Card : structure d'information Calypso  
 uint8\_t rec\_no : N° d'enregistrement à lire  
 uint8\_t sfi : Fichier à sélectionner  
 uint32\_t DecValue : Valeur à décrémente  
 uint32\_t \*NewValue : Nouvelle valeur

#### RETURNS

Status : 0 ou erreur



#### 4.13.4 Calypso\_DecreaseMultiple

#### 4.13.5 Calypso\_Increase

##### NAME

Calypso\_Increase

##### DESCRIPTION

Incrémente la valeur contenu dans le fichier compteur

##### INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Type	: Type de carte sans contact
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso
uint8_t rec_no	: N° d'enregistrementn à lire
uint8_t sfi	: Fichier à sélectionner
uint32_t IncValue	: Valeur à incrémenter
uint32_t *NewValue	: Nouvelle valeur

##### RETURNS

Status : 0 ou erreur

#### 4.13.6 Calypso\_IncreaseMultiple

#### 4.13.7 Calypso\_ReadRecord

##### NAME

Calypso\_ReadRecord

##### DESCRIPTION

Lecture d'un enregistrement dans un EF circulaire

##### INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Type	: Type de carte sans contact
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso
uint8_t rec_no	: N° d'enregistrementn à lire
uint8_t sfi	: Fichier à sélectionner
uint8_t *Data	: données à écrire
uint8_t *Datalen	: longueur des données à écrire

## RETURNS

Status : 0 ou erreur

### 4.13.8 Calypso\_UpdateRecord

#### NAME

Calypso\_UpdateRecord

#### DESCRIPTION

Lecture d'un enregistrement dans un EF circulaire

#### INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Type	: Type de carte sans contact
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso
uint8_t rec_no	: N° d'enregistrement à lire
uint8_t sfi	: Fichier à sélectionner
uint8_t rec_size	: nombre d'octet à lire
uint8_t *Data	: données lues
uint8_t *Datalen	: longueur des données lues

## RETURNS

Status : 0 ou erreur

### 4.13.9 Calypso\_WriteRecord

#### NAME

Calypso\_WriteRecord

#### DESCRIPTION

Ecriture d'un enregistrement dans un EF lineaire

#### INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Type	: Type de carte sans contact
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso
uint8_t rec_no	: N° d'enregistrementn à lire
uint8_t sfi	: Fichier à) sélectionner
uint8_t rec_size	: nombre d'octet à lire
uint8_t Data	: données à écrire
uint8_t Datalen	: longueur des données lues

#### RETURNS

Status : 0 ou erreur

### 4.13.10 Calypso\_OpenSecureSession1

#### NAME

Calypso\_OpenSecureSession1

#### DESCRIPTION

Ouverture d'une session sécurisé Calypso selon la révision 1

#### INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Type	: Type de carte sans contact
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso
uint8_t kvc	: Référence de la clé
uint8_t rec_no	: N° d'enregistrementn à lire
uint8_t sfi	: Fichier à) sélectionner
uint8_t *Sam_Challenge	: Challenge du SAM
uint8_t *Card_Challenge	: Challenfge de la carte
uint16_t *Resp	: Reponse de la carte (avec les octet d'etat SW1-SW2)
uint8_t *Resplen	: longueur de la réponse de la carte
uint16_t *Data	: reponse de la carte (si reponse);

uint8\_t \*Datalen : longueur de la réponse de la carte

## RETURNS

Status : 0 ou erreur

### 4.13.11 Calypso\_OpenSecureSession2

## NAME

Calypso\_OpenSecureSession2

## DESCRIPTION

Ouverture d'une session sécurisé Calypso selon la révision 2

## INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Type	: Type de carte sans contact
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso
uint8_t kvc	: Référence de la clé
uint8_t rec_no	: N° d'enregistrement à lire
uint8_t sfi	: Fichier à sélectionner
uint8_t *Sam_Challenge	: Challenge du SAM
uint8_t *Card_Challenge	: Challenge de la carte
uint8_t *Ratification	:
uint8_t *Resp	: Réponse de la carte (avec les octet d'état SW1-SW2)
uint16_t *Resplen	: longueur de la réponse de la carte
uint8_t *Data	: Réponse de la carte (si réponse);
uint16_t *Datalen	: longueur de la réponse de la carte
uint8_t *kvc_resp	: Référence de la clé de la carte

## RETURNS

Status : 0 ou erreur

### 4.13.12 Calypso\_OpenSecureSession3

#### NAME

Calypso\_OpenSecureSession3

#### DESCRIPTION

Ouverture d'une session sécurisé Calypso selon la révision 3

#### INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Type	: Type de carte sans contact
CALYPSO_INFO *pInfo_Card	: structure d'information Calypso
uint8_t kvc	: Référence de la clé
uint8_t rec_no	: N° d'enregistrementn à lire
uint8_t sfi	: Fichier à) sélectionner
uint8_t *Sam_Challenge	: Challenge du SAM
uint8_t *Card_Challenge	: Challenfge de la carte
uint8_t *Ratification	:
uint8_t *Resp	: Reponse de la carte (avec les octet d'etat SW1-SW2)
uint16_t *Resplen	: longueur de la réponse de la carte
uint8_t *Data	: Reponse de la carte (si reponse);
uint16_t *Datalen	: longueur de la réponse de la carte
uint8_t *kvc_resp	: Référence de la clé de la carte
uint8_t *kif_resp	: Référence de la clé d'identification de la carte de la carte

#### RETURNS

Status : 0 ou erreur

### 4.13.13 Calypso\_CloseSecureSession

#### NAME

Calypso\_CloseSecureSession

#### DESCRIPTION

Fermeture de la session sécurisé Calypso

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t Type : Type de carte sans contact  
 CALYPSO\_INFO \*pInfo\_Card : structure d'information Calypso  
 BOOL ratify\_now :  
 uint8\_t \*Sam\_Sign : Signature du SAM  
 uint8\_t \*Resp : Reponse de la carte (avec les octet d'etat SW1-SW2)  
 uint16\_t \*Resplen : longueur de la réponse de la carte

#### RETURNS

Status : 0 ou erreur

### 4.13.14 Calypso\_Invalidate

#### NAME

Calypso\_Invalidate

#### DESCRIPTION

Dévalides le DF

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t Type : Type de carte sans contact  
 CALYPSO\_INFO \*pInfo\_Card : structure d'information Calypso

#### RETURNS

Status : 0 ou erreur

#### 4.13.15 SAM\_Calypso\_SelectDiversifer

**NAME**

SAM\_Calypso\_SelectDiversifer

**DESCRIPTION**

Envoie au SAM le numero de serie de la carte

**INPUTS**

ReaderName \*Name : Information sur le lecteur  
uint8\_t \*UID : numéro de série de la carte

**RETURNS**

Status : 0 ou erreur

#### 4.13.16 SAM\_Calypso\_GetChallenge

**NAME**

SAM\_Calypso\_GetChallenge

**DESCRIPTION**

Récupere le challenge générer par le SAM

**INPUTS**

ReaderName \*Name : Information sur le lecteur  
uint8\_t Challenge : Challenge générer par le SAM

**RETURNS**

Status : 0 ou erreur

### 4.13.17 SAM\_Calypso\_DigestInitOld

#### NAME

SAM\_Calypso\_DigestInitOld

#### DESCRIPTION

Initialisatiuon du process de signature avec la réponse de la carte à la commande Calypso\_OpenSecureSession1, Calypso\_OpenSecureSession2 ou Calypso\_OpenSecureSession3

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t key : Numero de la clé  
 uint8\_t \*Data : Reponse de à la commande Calypso\_OpenSecureSession avec le Challenge  
 uint8\_t Datalen :longuerue des données

#### RETURNS

Status : 0 ou erreur

### 4.13.18 SAM\_Calypso\_DigestUpdate

#### NAME

SAM\_Calypso\_DigestUpdate

#### DESCRIPTION

envoie les commandes APDU au SAM

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t key : Numero de la clé  
 uint8\_t \*Data : commande APDU  
 uint8\_t Datalen : longueur des données

#### RETURNS

Status : 0 ou erreur

### 4.13.19 SAM\_Calypso\_DigestClose

#### NAME

SAM\_Calypso\_DigestClose

#### DESCRIPTION



Demande une signature au SAM

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
uint8\_t key : Numero de la clé  
uint8\_t \*SignHi : Signature du SAM

#### RETURNS

Status : 0 ou erreur

### 4.13.20 SAM\_Calypso\_DigestAuthenticate

#### NAME

SAM\_Calypso\_DigestAuthenticate

#### DESCRIPTION

Envoie la signature de la carte au SAM

#### INPUTS

ReaderName \*Name : Information sur le lecteur  
uint8\_t key : Numero de la clé  
uint8\_t \* SignLo : Signature de la carte

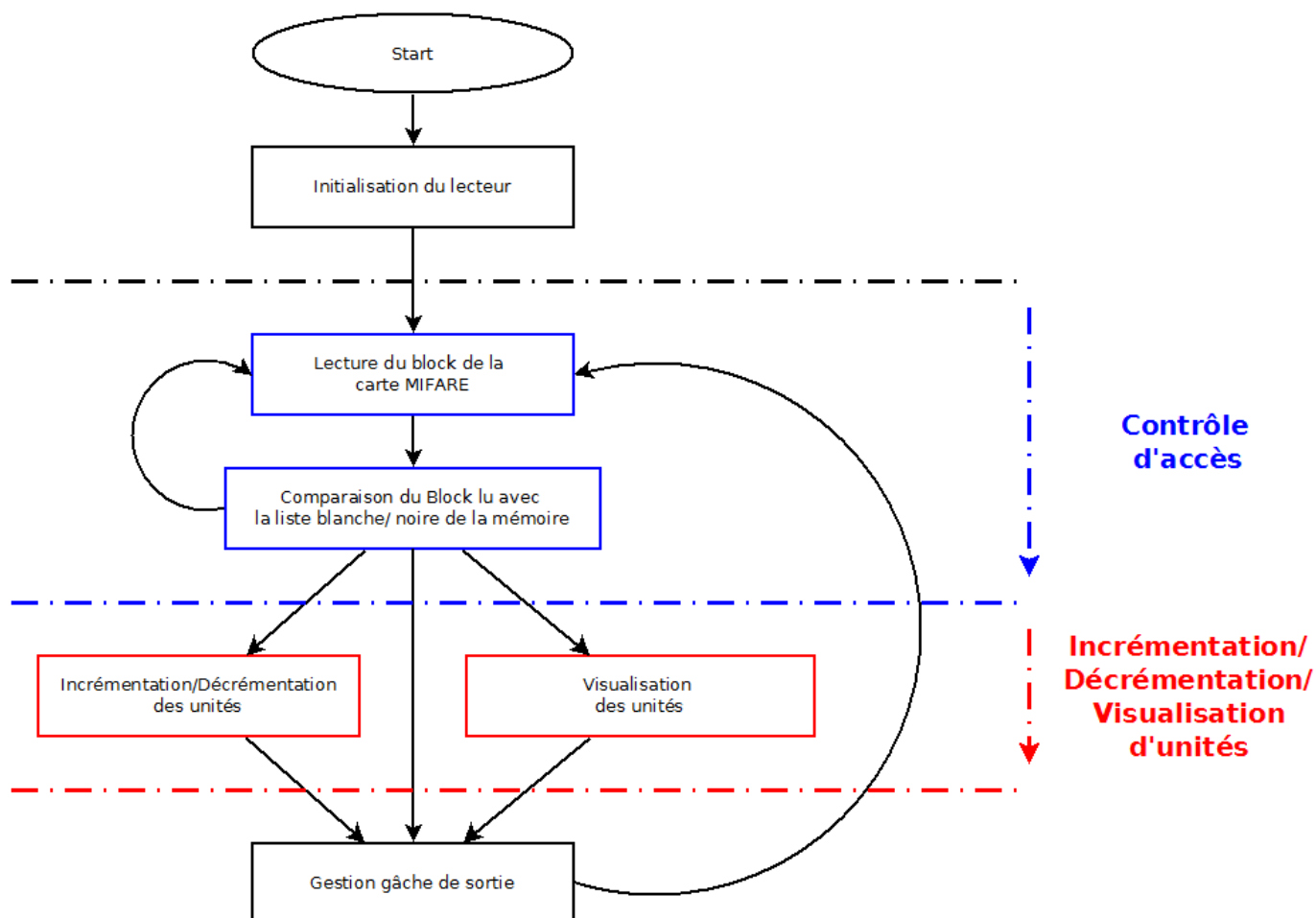
#### RETURNS

Status : 0 ou erreur

## 4.14 Lecteur autonome

En mode autonome le lecteur embarque une machine à état.

Le lecteur peut être configuré en contrôle d'accès, effectuer des manipulations sur des unités préprogrammées dans la carte MIFARE ou bien en pointeuse.



Le lecteur possède un afficheur 2 lignes. Par défaut le lecteur indique

- l'heure sur la première ligne;
- "Bonjour" sur la deuxième ligne
- "Autorisé" sur la deuxième ligne lorsqu'un badge est autorisé (en mode contrôle d'accès) ou le nombre d'unités restantes lorsqu'il est configuré en mode comptage décomptage et visualisation d'unités;
- "Non Autorisé" sur la deuxième ligne lorsqu'un badge est non autorisé

Les informations de pointage peuvent soit être stockées dans la mémoire interne puis être téléchargées, ou envoyées directement à un serveur sous la forme d'une trame XML



<trame><id>4348414E4C4F4E00</id><uid>CBE84496</uid><heure>21:12:09</heure><date>10/05/11</date><host>192.168.1.4</host></trame>

Spécifications Techniques«Logiciel Coupleur»		Page 51 on 60	
Lieu: DIJON / ODALID		Date : 21/01/2013	
Ref : 1.4		Ecrit par: Vincent THIVENT	

#### 4.14.1 AutoReader\_SetRTC

##### NAME

AutoReader\_SetRTC

##### DESCRIPTION

Mise à jour de l'heure

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL TimeHost : TRUE -> le lecteur prend automatiquement l'heure de l'ordinateur  
 FALSE -> configuration manuelle  
 uint8\_t Time[7] : Heure (7 caractères) (ignoré lorsque (TimeHost = TRUE));

SEC	MIN	HOURS	DAY	DATE	MONTH	YEAR
00-59	00-59	00-23	01-07	1-31	01-12	00-99

Ex: char Time[] = {0x00, 0x00, 0x14, 0x01, 0x01, 0x01, 0x11}; => Lundi 1 Janvier 2011 14:00:00

##### RETURNS

Status : 0 ou erreur

#### 4.14.2 AutoReader\_AddUser

##### NAME

AutoReader\_AddUser

##### DESCRIPTION

Ajout d'un utilisateur en mémoire

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t User[8] : Identifiant (8 caractères)

##### RETURNS

Status : 0 ou erreur

#### 4.14.3 AutoReader\_DeleteUser

**NAME**

AutoReader\_DeleteUser

**DESCRIPTION**

Suppression d'un utilisateur en mémoire

**INPUTS**

ReaderName \*Name : Information sur le lecteur  
uint8\_t User[8] : Identifiant UID (8 caractères)

**RETURNS**

Status : 0 ou erreur

#### 4.14.4 AutoReader\_ClearFullUser

**NAME**

AutoReader\_ClearFullUser

**DESCRIPTION**

Suppression de tous les UID contenus en mémoire

**INPUTS**

ReaderName \*Name : Information sur le lecteur

**RETURNS**

Status : 0 ou erreur

#### 4.14.5 AutoReader\_LoadKeyMifare

##### NAME

AutoReader\_LoadKeyMifare

##### DESCRIPTION

Chargement des clés Mifare dans l'EEPROM sécurisée du lecteur.

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL Auth\_Key : Clé A ou B (clé A -> TRUE et clé B -> FALSE)  
 uint8\_t Key[6] : Clé Mifare (6 octets) ex : {0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF}  
 uint8\_t key\_index : Adresse de stockage de la clé (0 – 15) dans le lecteur

##### RETURNS

Status : 0 ou erreur

#### 4.14.6 AutoReader\_AccessMifare

##### NAME

AutoReader\_AccessMifare

##### DESCRIPTION

Configuration du block de lecture MIFARE et de la clé d'authentification

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t Block : Adresse du secteur ou lire la clé (0 – 15)  
 uint8\_t Auth\_Key : Clé A ou B (clé A -> TRUE et clé B -> FALSE)  
 uint8\_t key\_index : Adresse de stockage de la clé (0 – 15) dans le lecteur

##### RETURNS

Status : 0 ou erreur

#### 4.14.7 AutoReader\_ComptMifare

##### NAME

AutoReader\_ComptMifare

##### DESCRIPTION

Configuration du block de lecture MIFARE et de la clé d'authentification qui contient le compteur

##### INPUTS

ReaderName *Name	: Information sur le lecteur
uint8_t Block	: Adresse du secteur ou lire la clé (0 – 15)
uint8_t Auth_Key	: Clé A ou B (clé A -> TRUE et clé B -> FALSE)
uint8_t key_index	: Adresse de stockage de la clé (0 – 15) dans le lecteur
BOOL Compt	: Incréméntation ou Décréméntation (Incréméntation -> TRUE et Décréméntation -> FALSE)
uint32_t Value	: Valeur à incrémenter ou décréémenter

##### RETURNS

Status : 0 ou erreur

#### 4.14.8 AutoReader\_AfficheValueMifare

##### NAME

AutoReader\_AfficheValueMifare

##### DESCRIPTION

Affichage du block de lecture MIFARE et de la clé d'authentification qui contient le compteur

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t Block : Adresse du secteur ou lire la clé (0 – 15)  
 uint8\_t Auth\_Key : Clé A ou B (clé A -> TRUE et clé B -> FALSE)  
 uint8\_t key\_index : Adresse de stockage de la clé (0 – 15) dans le lecteur

##### RETURNS

Status : 0 ou erreur

#### 4.14.9 AutoReader\_TempoGache

##### NAME

AutoReader\_TempoGache

##### DESCRIPTION

Temporisation de la gâche de sortie

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 uint8\_t Temps : Tempo de la gâche de sortie (Temps = 5 -> 5 x 500 ms = 2,5s)

##### RETURNS

Status : 0 ou erreur



#### 4.14.10 AutoReader\_BlackList

##### NAME

AutoReader\_BlackList

##### DESCRIPTION

Configuration du lecteur pour comparer le block de lecteur à une liste blanche ou une liste noire

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
uint8\_t Temps : Tempo de la gâche de sortie (Temps = 5 -> 5 x 500 ms = 2,5s)

##### RETURNS

Status : 0 ou erreur

#### 4.14.11 AutoReader\_SendInfo

##### NAME

AutoReader\_SendInfo

##### DESCRIPTION

Mise à jour de l'adresse IP du serveur

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
BOOL SendInfo : Envoi ou non envoi des informations aux serveurs (envoi -> TRUE et non envoi -> FALSE)  
uint8\_t \* IPServer : Adresse IP du serveur  
uint16\_t Port : Port du serveur

##### RETURNS

Status : 0 ou erreur

##### Format de donnée de l'envoi:

```
<trame><id>4348414E4C4F4E00</id><uid>CBE84496</uid><heure>21:12:09</heure>
><date>10/05/11</date><host>192.168.1.4</host></trame>
```

La balise

- <id></id> : data lue par le contrôle d'accès
- <uid></uid> : uid de la carte MIFARE
- <heure></heure> : heure de passage
- <date></date> : date

#### 4.14.12 AutoReader\_LCD1

##### NAME

AutoReader\_LCD1

##### DESCRIPTION

Affichage de la première ligne du lecteur

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL option : Option d'affichage (0 - > paramètre par défaut  
 "date et heure" 1 -> affichage des paramètres LCD)  
 uint8\_t Temps : Texte à afficher 16 caractères

##### RETURNS

Status : 0 ou erreur

#### 4.14.13 AutoReader\_LCD2A

##### NAME

AutoReader\_LCD2A

##### DESCRIPTION

Affichage permanent de la deuxième ligne du lecteur

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL option : Option d'affichage (0 - > paramètre par défaut  
 "Bonjour !" 1 -> affichage des paramètres LCD)  
 uint8\_t Temps : Texte à afficher 16 caractères

##### RETURNS

Status : 0 ou erreur

#### 4.14.14 AutoReader\_LCD2B

##### NAME

AutoReader\_LCD2B

##### DESCRIPTION

Affichage de la deuxième ligne du lecteur " autorisé"

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL option : Option d'affichage (0 - > paramètre par défaut " Autorisé !" 1 -> affichage des paramètres LCD)  
 uint8\_t Temps : Texte à afficher 16 caractères

##### RETURNS

Status : 0 ou erreur

#### 4.14.15 AutoReader\_LCD2C

##### NAME

AutoReader\_LCD2C

##### DESCRIPTION

Affichage de la deuxième ligne du lecteur " Non Autorisé"

##### INPUTS

ReaderName \*Name : Information sur le lecteur  
 BOOL option : Option d'affichage (0 - > paramètre par défaut "Non Autorisé !" 1 -> affichage des paramètres LCD)  
 uint8\_t Temps : Texte à afficher 16 caractères

##### RETURNS

Status : 0 ou erreur

## 5

## Contact

### SARL ODALID

Aile des Sciences de l'Ingénieur  
 9 Avenue Alain Savary  
 21000 Dijon  
 France

Spécifications Techniques«Logiciel Coupleur»		Page 59 on 60
Lieu: DIJON / ODALID	Date : 21/01/2013	
Ref : 1.4	Ecrit par:	Vincent THIVENT

Tel : +33 (0)9 72 12 90 98  
Fax : +33 (0)9 72 22 45 17  
Web : <http://odalid.com>  
Mail : [support@odalid.com](mailto:support@odalid.com)