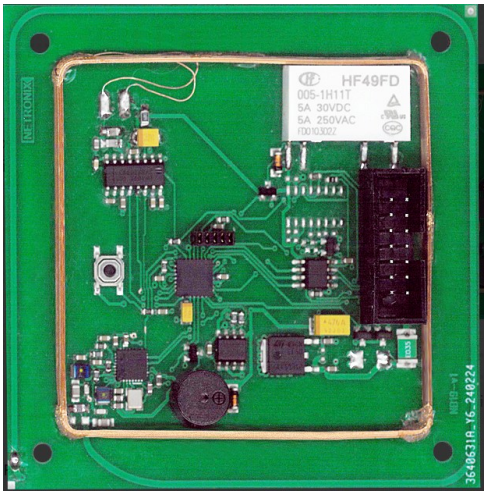
DOKUMENTACJA TECHNICZNA

CZYTNIK RFID

CTU-S5R



1 WPROWADZENIE 5
2 DANE TECHNICZNE 6
3 KONFIGURACJA ZA POMOCĄ WBUDOWANEGO PRZYCISKU 8
4 KONFIGURACJA ZA POMOCĄ NARZĘDZIA ACCESSCONFIG 9
5 INTERFEJSY 10
5.1 Interfejs RS232 / RS485/ UART 10
5.1.1 Protokół komunikacyjny dla interfejsu RS232 / RS485/ UART 10
5.1.2 Przebiegi sygnałów dla szeregowej transmisji danych RS232 oraz UART 10
5.2 Interfejs I²C 10
5.2.1 Algorytm wymiany danych 11
5.2.2 Zależności czasowe 12
5.3 Interfejs 1WIRE. 12
5.4 Interfejs WIEGAND. 13
6 KOMENDY DOSTĘPNE DLA INTERFEJSU RS232/UART/RS485 - PROTOKÓŁ NETRONIX 14
6.1 Zarządzanie kluczami 14
6.1.1 Zapis klucza do dynamicznej pamięci kluczy 14
6.1.2 Zapis klucza do statycznej pamięci kluczy 14
6.2 Rozkazy komunikacji z transponderami 14
6.2.1 Załączanie i wyłączanie pola czytnika 14
6.2.2 Wyselekcjonowanie jednego transpondera Mifare z wielu 15
6.2.3 Odczyt OSTATIO ODCZYTANEGO ID TRANSPONDERA 16
6.2.4 Logowanie do wybranego sektora transpondera za pomocą Dynamicznego Bufora Klucza 16
6.2.5 Logowanie do wybranego sektora transpondera za pomocą Statycznego Bufora Kluczy 17
6.2.6 Odczyt zawartości bloku transpondera 17
6.2.7 Zapis zaartości bloku transpondera 17
6.2.8 Kopiowanie zawartości bloku transpondera do innego bloku 17
6.2.9 Zapis wartości do bloku transpondera 18
6.2.10 Odczyt wartości z bloku transpondera 18
6.2.11 Zwiększenie wartości zawartej w bloku transpondera 18
6.2.12 Zmniejszanie wartości zawartej w bloku transpondera 19
6.2.13 Uśpienie transpondera będącego w polu 19
6.2.14 Zapis zawartości strony w Mifare UL 19
6.2.15 Odczyt zawartości stron w Mifare UL 20
6.2.16 Uwierzytelnienie dla transpondera Ultralight C 20
6.3 Rozkazy do komunikacji z transponderami Mifare Plus 20
6.3.1 Rozkazy poziomu SL0 20
6.3.1.1 Write Perso –inicjalizacja karty 20
6.3.1.2 Commit Perso – przejście do następnego poziomu SL 20
6.3.2 Rozkazy poziomu SL1 21
6.3.2.1 Uwierzytelnienie SL1 21
6.3.2.2 Przejście na wyższy poziom SL/ sprawdzenie oryginalności transpondera 21
6.3.3 Rozkazy poziomu SL3 21
6.3.3.1 Wprowadzenie transpondera w tryb ISO14443-4 21
6.3.3.2 Logowanie do sektora 22
6.3.3.3 Odczyt zawartości bloku transpondera 22
6.3.3.4 Zapis zawartości bloku transpondera 22
6.3.4 Czasy trwania operacji dla Mifare Plus 23
6.4 Obsługa transponderów DESFire, DESFire EV1 23
6.4.1 Autoryzacja, logowanie do aktualnie wyselekcjonowanej aplikacji 23
6.4.2 Zmiana ustawień klucza Master aktualnie wybranej aplikacji 23
6.4.3 Zmiana klucza 24
6.4.4 Tworzenie aplikacji 24
6.4.5 Usuwanie aplikacji 25

6.4.6 Pobieranie listy aplikacji ... 25
6.4.7 Wybór aplikacji ... 25
6.4.8 Formatowanie transpondera ... 26
6.4.9 Inicjalizacja protokołu transmisji z transponderami DESFire (ISO14443-4) ... 26
6.4.10 Pobieranie listy plików aktualnie wybranej aplikacji ... 26
6.4.11 Pobieranie właściwości pliku ... 26
6.4.12 Tworzenie plików typu *Standard Data Files* ... 27
6.4.13 Tworzenie plików typu *Backup Data Files* ... 27
6.4.14 Tworzenie plików typu Linear/Cyclic Record Files ... 28
6.4.15 Usuwanie pliku ... 28
6.4.16 Zmiana ustawień pliku ... 28
6.4.17 Odczyt danych z pliku typu *Std/Back Data File* ... 29
6.4.18 Zapis danych do pliku typu *Std/Back Data File* ... 29
6.4.19 Zapis rekordu do pliku typu *Record Data File* ... 29
6.4.20 Odczyt rekordu z pliku typu *Record Data File* ... 30
6.4.21 Czyszczenie plików typu *Record Data File* ... 30
6.4.22 Komenda potwierdzająca - *DesCommit* ... 30
6.4.23 Deselekcja transpondera ... 30
6.5 Transmisja danych I-Block protokołu T=CL ISO14443-4 ... 31
6.6 Obsługa transponderów rodziny I-CODE SLI ... 31
6.6.1 Odczyt numeru ID transpondera I-CODE SLI ... 31
6.6.2 Odczyt strony transpondera SLI ... 31
6.6.3 Zapis zawartości strony w SLI ... 32
6.7 Odczyt numeru ID transpondera ICLASS ... 32
6.8 Wejścia i wyjścia elektryczne ... 32
6.8.1 Zapis stanu wyjścia ... 32
6.8.2 Odczyt stanu wejścia ... 32
6.8.3 Odczyt konfiguracji dowolnego portu ... 35
6.9 Hasło dostępu ... 36
6.9.1 Logowanie do czytnika ... 36
6.9.2 Zmiana hasła ... 36
6.9.3 Wylogowanie z czytnika ... 37
6.10 Obsługa wewnętrznej pamięci transponderów ... 37
6.10.1 Odczyt numeru transpondera z pamięci ... 37
6.10.2 Zapis numeru transpondera do pamięci ... 37
6.11 Obsługa wbudowanej kontroli dostępu ... 37
6.11.1 Zapis konfiguracji kontroli dostępu ... 38
6.11.2 Odczyt konfiguracji kontroli dostępu ... 38
6.11.3 Zapis konfiguracji automatu ... 38
6.11.4 Odczyt konfiguracji automatu ... 40
6.11.5 Ustawienie daty I czasu ... 40
6.11.6 Odczytanie daty i czasu ... 40
6.12 Konfiguracja interface'u szeregowego RS-232/485 ... 41
6.12.1 Zapis konfiguracji interfejsu szeregowego ... 41
6.12.2 Odczyt konfiguracji interfejsu szeregowego ... 41
6.12.3 Konfiguracja rejestratora zdarzeń ... 41
6.12.4 Odczyt konfiguracji rejestratora zdarzeń ... 42
6.12.5 Odczyt liczników związanych z pamięcią zdarzeń ... 42
6.12.6 Odczyt zdarzeń ... 43
6.13 Rozkazy pozostałe ... 43
6.13.1 Zmiana głośności buzzera ... 43
6.13.2 Zdalny reset czytnika ... 44
6.13.3 Tryb sleep ... 44
6.13.4 Odczyt wersji oprogramowania czytnika ... 44
6.14 Znaczenie kodów operacji w ramkach odpowiedzi ... 45
7 PROTOKÓŁ MODBUS RTU ... 46
7.1 Obsługiwane funkcje protokołu MODBUS ... 46

7.2 Adresy MODBUS 46
7.2.1 Adresy do odczytu ID karty 46
7.2.2 Adresy do odczytu/zapisu konfiguracji autoreadera 46
Adresy do konfiguracji GPIO1 46
7.2.3 Adresy do konfiguracji GPIO2 47
7.2.4 Adresy do konfiguracji przekaźnika 48
7.2.5 Adresy do konfiguracji buzzera 48
7.3 Enkapsulacja protokołu Netronix w protokole MODBUS RTU 49
7.3.1 Schemat postępowania 49
7.3.2 Przykład użycia – odczyt wersji oprogramowania. 49
8 MECHANIZM MASTERID 51
9 MECHANIZM USERID 52
10 CZYSZCZENIE PAMIĘCI KART I POWRÓT DO USTAWIEŃ FABRYCZNYCH 53
11 BOOTLOADER – ZMIANA FIRMWARE URZĄDZENIA 54
12 PRZYKŁAD PRACY Z TRANSPONDEREMI 56
12.1 Przykład pracy z transponderem S50,S70 56
12.2 Przykład pracy z transponderami Desfire 56

1 WPROWADZENIE

Czytnik serii CTU-S jest OEM'owym czytnikiem kart RFID działających na częstotliwości znamionowej 13,56MHz oraz 125kHz.

Posiada on następującą funkcjonalność:

- Obsługuje transpondery:
  - 13.56MHz: Mifare S50, Mifare S70, Mifare Ultra Light, Mifare DesFire, I-CODE SLI, iCLASS,
  - 125kHz: EM4102, Hitag 1/s, HID PROX II, Hitag 2
  - NFC ID - unikalne ID z telefonu z aplikacją Android NFC ID
  - USER ID – odczyt bezpiecznego ID z pamięci transponderów Mifare Plus, Desfire, Classic
- Wbudowana antena
- Pamięć kart z wbudowanym sterownikiem rygla
- Różnorodność interfejsów komunikacyjnych w zależności od wersji (tabela poniżej)
- Adresowalność na szynie RS-485
- Wbudowany przekaźnik, buzzer
- Wbudowany przycisk konfiguracyjny/powrotu do ustawień fabrycznych
- Konfigurowalne dwustanowe wejścia/wyjścia
- Konfigurowanie zachowania buzzera, przekaźnika
- Sterowanie dwustanowymi wyjściami
- Odczytywanie dwustanowych wejść
- Możliwość konfigurowania formatu wysyłanego numeru ID
- Transmisja zabezpieczona hasłem
- Aktualizacja oprogramowania poprzez interfejs komunikacyjny
- Dostępna wersja (CTU-S5RL) z trybem *sleep*

Rodzina modeli CTU-S

Typ modułu	GPIO	Pamięć kart	Pamięć zdarzeń	Przekaźnik	Napięcie zasilania	INTERFEJSY							
						RS-232	RS-485	RS-232TTL	SPI	I2C	WIEGAND	1WIRE	UART TTL
CTU-S2R	2	1000	3400	✓	5-16	✓							
CTU-S5R*	2	1000	3400	✓	5-16		✓	✓	✓	✓	✓	✓	✓
CTU-S5L	1	1000	3400	✓	5			✓	✓	✓	✓	✓	✓

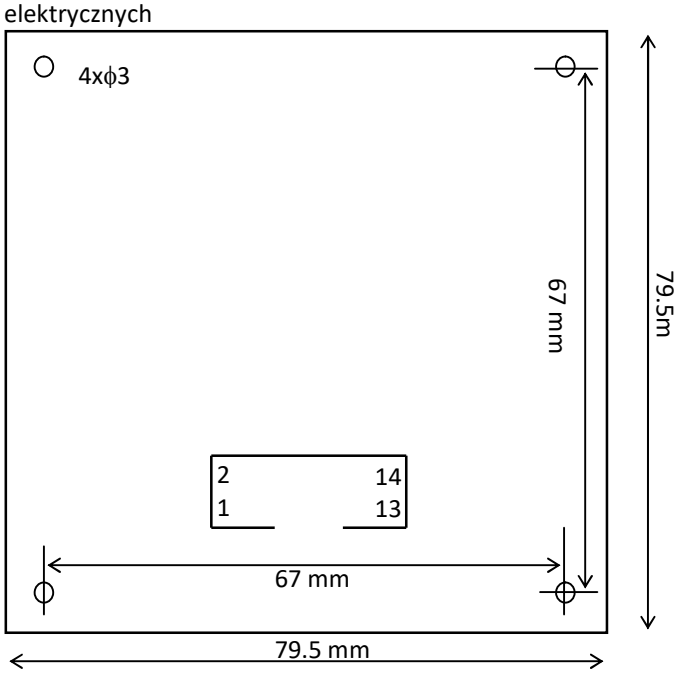
* - wersja produkowana standardowo, pozostałe wersje na zamówienie

2 DANE TECHNICZNE

Obsługiwana funkcjonalność w zależności od typu transpondera / karty:		
Typ karty	**Odczyt numeru ID**	**Pełny zapis i odczyt bloków pamięci**
MIFARE® Classic S50,S70, Ultralight	TAK	TAK
MIFARE® DesFire	TAK	TAK
I-CODE SLI	TAK	TAK
Mifare PLUS	TAK	SL0,SL1,SL3
IClass	TAK (tylko CSN)	NIE
EM4102	TAK	-
Q5	TAK	TAK
Hitag-1/s	TAK	TAK (nieszyfrowany)

Parametry modułu CTU-Sxx	
Napięcie zasilania	5-16 V (CTU-S5R / CTU-S2R) 5V (CTU-S5L)
Maksymalny pobór prądu	170 mA
Pobór prądu w trybie sleep (CTU-S5RL)	16uA
Znamionowa częstotliwość RF pracy modułu	13,56 MHz
Temperatura pracy	-20°c - +65°c
Dopuszczalny prąd przekaźnika	3A
Dopuszczalne napięcie przekaźnika	30V
Odległość odczytu	do 15 cm
Maksymalny prąd wyjść GPIO	20mA
Maksymalne napięcie wejść GPIO	5V
Parametry transmisji RS232/RS485/RSTTL/UART TTL	2400, 4800, 9600, 19200, 38400, 57600, 115200 b/s, 8 bitów danych, 1 bit stopu, bez bitu parzystości Protokół Netronix lub MODBUS RTU
Adres na magistrali I^2C	0xC0

Wymiary, opis wyprowadzeń elektrycznych



Rys.2 Widok od góry

Nr pinu	Opis wyprowadzenia złącza wannowego
1	RS232RX, RS485B, RSTTL_RX, UART_RX, 1WIRE, MOSI, SDA, WIEGAND1
2	RS232TX, RS485A, RSTTL_TX, UART_TX, MISO
3	SCK, SCL,WIEGAND0
4	CS
5	NC
6	GND
7	VCC
8	GPIO 1
9	GPIO 2 / SLEEP(CTU-S5L)
10	GND
11	NC
12	NC
13	RELAY 1
14	RELAY 2

3 KONFIGURACJA ZA POMOCĄ WBUDOWANEGO PRZYCISKU

Na płycie umieszczony został przycisk, który pełni dwie funkcje:

1. Powrót do ustawień fabrycznych – przytrzymanie przycisku przez co najmniej 8 sekund
2. Wybór interfejsu wg schematu poniżej:

KROK	**Ilość wciśnięć:**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	MENU1 wybór transpondera	-	Mifare ISO14443A	EM4102 Unique	HID PROX II	iCLASS CSN	ISO15693 I-CODE	HITAG 1/S	HITAG 2	ISO14443 B	Wszystkie
2	podwójny, potwierdzający sygnał buzzera										
3	MENU2 wybór interfejsu*	-	RS232 TTL	UART TTL	WIEGAND **	DALLAS ALL	DALLAS SINGLE	RS485	I2C	SPI	WIEGAND26-SLAM CONFIG
4	Potrójny, potwierdzający sygnał buzzera										

* - typ interfejsu uzależniony jest od typu modułu

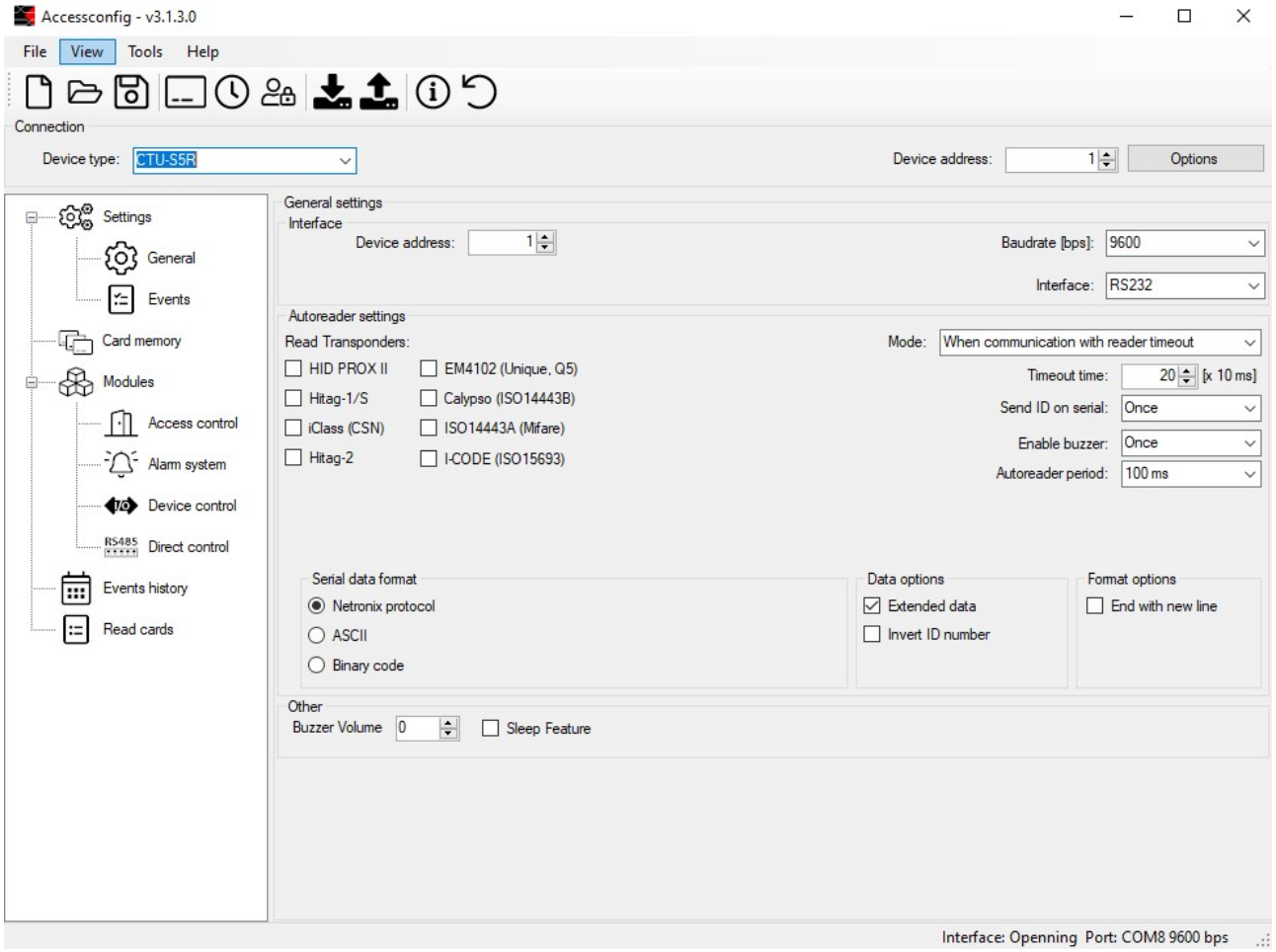
** - domyślnie jest to Wiegand 37. Zmiana parametrów jest możliwa z użyciem narzędzia AccessConfig

Przykład:

Aby ustawić interfejs Wiegand oraz transponder Unique, należy:
-wcisnąć dość szybko przycisk 3 razy,
-poczekać na podwójny sygnał buzzera,
-wcisnąć dość szybko przycisk 4 razy,
-poczekać na potrójny sygnał buzzera

4 KONFIGURACJA ZA POMOCĄ NARZĘDZIA ACCESSCONFIG

Czytnik można skonfigurować z użyciem narzędzia AccessConfig dostępnego na naszej stronie. Komunikacja może odbywać się portem RS232, UART lub RS485 w zależności od wybranej przyciskiem konfiguracji. Domyślnie aktywnym interfejsem jest RS232.



5 INTERFEJSY

5.1 INTERFEJS RS232 / RS485/ UART

5.1.1 PROTOKÓŁ KOMUNIKACYJNY DLA INTERFEJSU RS232 / RS485/ UART

Przy komunikacji poprzez interfejs RS232 / RS485/ UART wykorzystywany jest protokół Netronix lub protokół MODBUS RTU.

W niniejszej dokumentacji opis protokołu RS-232/485 ograniczony został do opisu rozkazów i odpowiedzi oraz ich parametrów. Nagłówek oraz suma kontrolna CRC występuje zawsze i jest zgodna z pełną dokumentacją "Protokół Netronix" dostępną na stronie www.netronix.pl/. Domyślne ustawienia parametry komunikacji to 9600,8 bitów, 1bit stopu, bez bitu parzystości. Prędkość transmisji zmienić można komendą C_SetInterfaceConfig opisaną w dalszej części dokumentacji.

Ramka rozkazu:

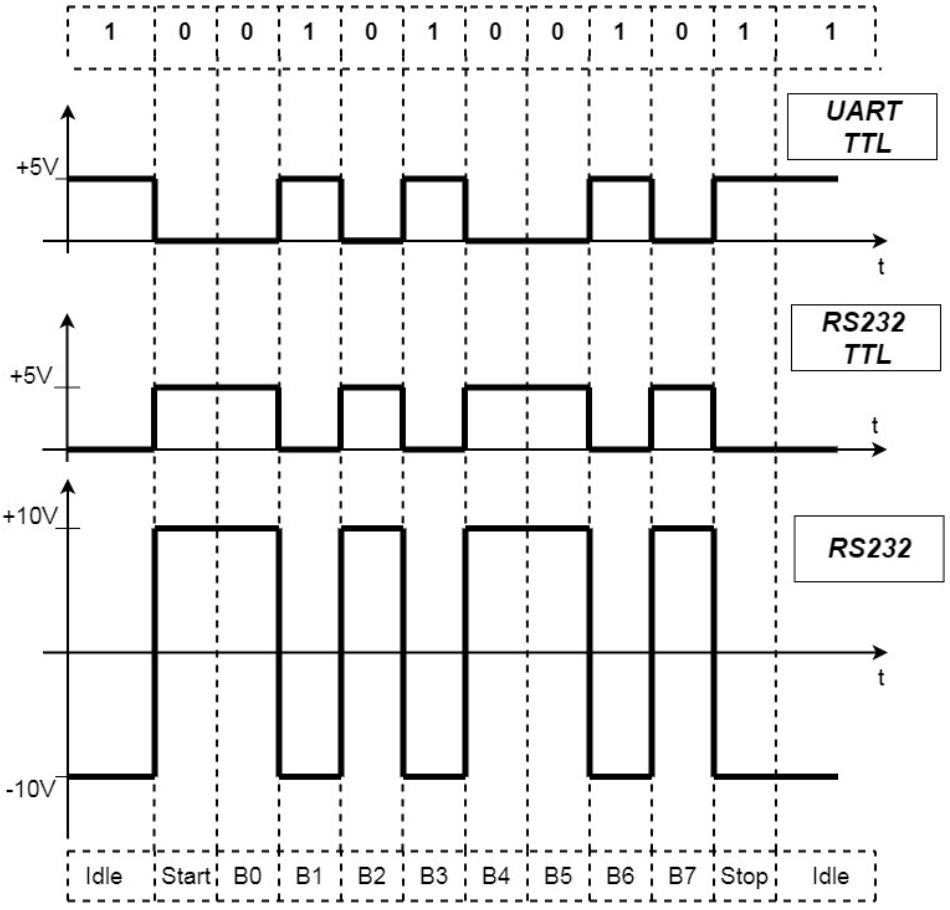
Adres	Długość	C_NazwaRozkazu	Parametry_rozkazu1…n	CRC

Ramka odpowiedzi:

Adres	Długość	C_NazwaRozkazu +1	Parametry_odpowiedzi1…m	KodOperacji	CRC

Pracę z protokołem RS przetestować można za pomocą narzędziowego, darmowego oprogramowania „**FRAMER**".

5.1.2 PRZEBIEGI SYGNAŁÓW DLA SZEREGOWEJ TRANSMISJI DANYCH RS232 ORAZ UART



Rys 3. Przebiegi sygnałów dla wyjść interfejsów UART oraz RS232

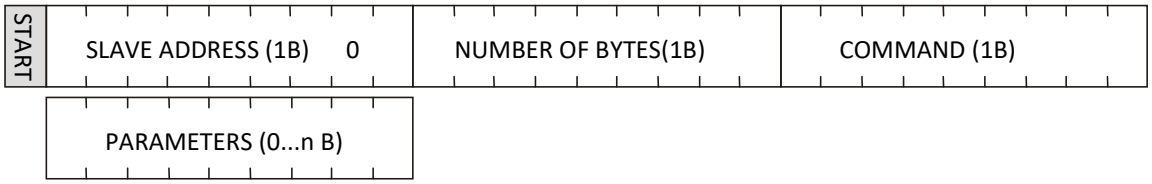
5.2 INTERFEJS I^2C

5.2.1 ALGORYTM WYMIANY DANYCH

Po skonfigurowaniu zgodnie z punktem 4, moduł CTU-S5R działa w trybie interfejsu I2C następujących sekwencjach:

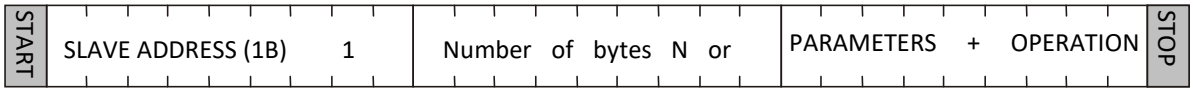
1. Master (urządzenie zewnętrzne) zapisuje rozkaz wraz z ewentualnymi parametrami w slave'ie (moduł CTU)
2. Wykonywany jest rozkaz. (natychmiast po odebraniu zadeklarowanej w ramce ilości bajtów wysyłanych)
3. Master dokonuje odczytu odpowiedzi i jej parametrów oraz kodu operacji. W przypadku otrzymania bajtu zajętości 0xCB, należy ponawiać próbę odczytu odpowiedzi po około 1ms (komendy związane z odczytem/zapisem do transponderów mogą trwać do 100ms)

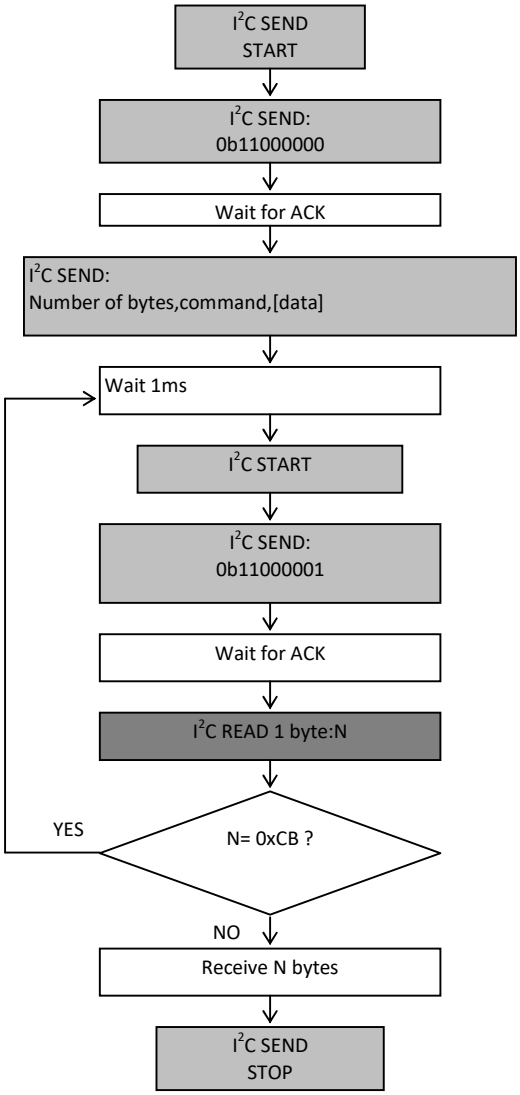
Do modułu CTU zapisujemy pytanie-rozkaz:



Pole „number of bytes" musi zawierać informację o ilości bajtów wysyłanych bezpośrednio po nim, czyli suma pół "command" i „parameters"

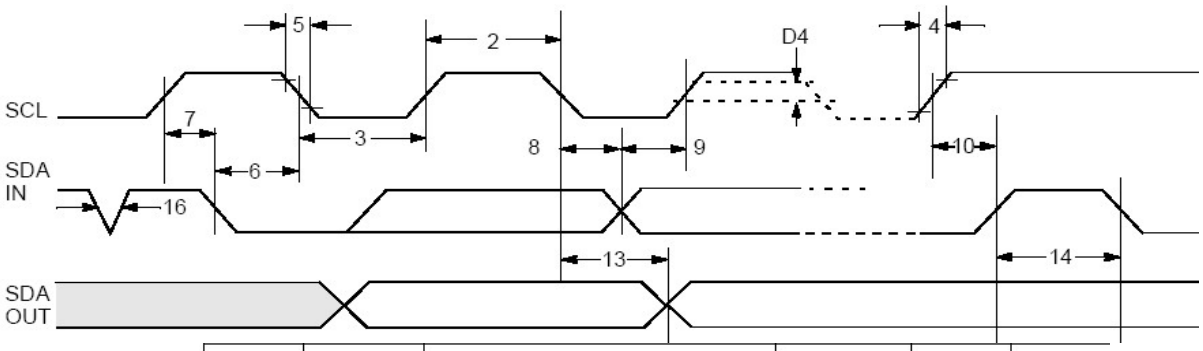
Następnie otrzymujemy:





5.2.2 ZALEŻNOŚCI CZASOWE

Moduł wysyła i odbiera dane z częstotliwością zegara do 400KHz z uwzględnieniem zależności czasowych przedstawionych poniżej.



Param. No.	Sym.	Characteristic	Min.	Max.	Units
1	FCLK	Clock Frequency	— —	400 100	kHz
2	THIGH	Clock High Time	600 4000	— —	ns
3	TLOW	Clock Low Time	1300 4700	— —	ns
4	TR	SDA and SCL Rise Time **(Note 1)**	— —	300 1000	ns
5	TF	SDA and SCL Fall Time	—	300	ns
6	THD:STA	Start Condition Hold Time	600 4000	— —	ns
7	TSU:STA	Start Condition Setup Time	600 4700	— —	ns
8	THD:DAT	Data Input Hold Time	0	—	ns
9	TSU:DAT	Data Input Setup Time	100 250	— —	ns
10	TSU:STO	Stop Condition Setup Time	600 4000	— —	ns
11	TSU:WP	WP Setup Time	600 4000	— —	ns
12	THD:WP	WP Hold Time	1300 4700	— —	ns
13	TAA	Output Valid from Clock **(Note 2)**	— —	900 3500	ns
14	TBUF	Bus free time: Time the bus must be free before a new transmission can start	1300 4700	— —	ns
15	TOF	Output Fall Time from VIH Minimum to VIL Maximum	20+0.1CB —	250 250	ns

Note2: Czytnik przetrzymuje w stanie niskim pierwszy impuls zegara każdego wysyłanego bajtu do czasu wystawienia prawidłowego stanu na linii SDA

5.3 INTERFEJS 1WIRE.

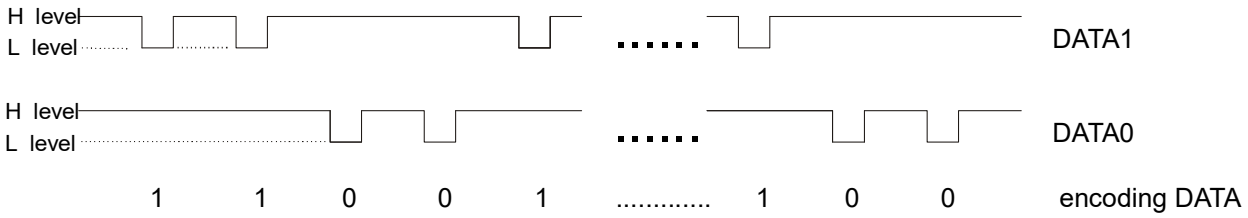
Po skonfigurowaniu urządzenia do pracy w trybie 1WIRE, czytnik emuluje zachowanie „pastylki" Dallas serii DS1990. Tak długo jak karta będzie w polu, tak długo czytnik będzie wystawiał unikalny numer na magistrali 1WIRE. Czas między kolejnymi odczytani transpondera wynosi 150ms, więc impulsy *presense* powinny występować nie rzadziej niż raz na 150ms. Format wysyłanego ID ma postać:

Kod rodziny	ID	Adres	CRC_DAL
01	ID1…ID5	01	XX

5.4 INTERFEJS WIEGAND.

Czytnik po skonfigurowaniu do pracy w trybie *WIEGAND* wysyła unikalny numer ID przeczytanej karty zgodnie z protokołem Wiegand 37 o parametrach:

Czas trwania impulsu (poziom L)..... 100us
Odstęp pomiędzy impulsami (poziom H)..... 1ms



6 KOMENDY DOSTĘPNE DLA INTERFEJSU RS232/UART/RS485 - PROTOKÓŁ NETRONIX

6.1 ZARZĄDZANIE KLUCZAMI

Zarządzanie kluczami sprowadza się do zapisu kluczy do wewnętrznej pamięci kluczy. Kluczy tych w celach bezpieczeństwa nie można odczytać. W celu utrzymania najwyższego bezpieczeństwa danych istnieje pewna poprawna filozofia pracy z kluczami. Polega ona na zapisie kluczy przez jednostki lub osoby posiadające najwyższy stopień zaufania. Taki zapis odbywa się tylko raz lub bardzo rzadko. Praca czytnika w konkretnej aplikacji polega nie na używaniu klucza wprost, ale na wywoływaniu odpowiedniego numeru klucza w celu zalogowania się do sektora. W ten sposób w konkretnej aplikacji klucz w zasadzie nie pojawia się na magistrali danych.
Dodatkowo użytkownik powinien zadbać, aby klucz miał odpowiednie prawa dostępu do sektorów. Realizuje się to poprzez proces inicjalizacji kart, gdzie zapisuje się do kart nowe tajne klucze wraz z odpowiednimi prawami dostępu przydzielonymi tym kluczom.
Każdemu sektorowi transpondera przyporządkowany jest klucz A i klucz B. Komendy C_LoadKeyToSKB oraz C_LoadKeyToDKB zapisują klucze do pamięci czytnika bez informacji jakiego rodzaju jest to klucz (A czy B). Użytkownik podczas logowania do sektora musi podać jako parametr 0xAA lub 0xBB jeżeli chce aby wywołany klucz był traktowany jako A lub jako B.

6.1.1 ZAPIS KLUCZA DO DYNAMICZNEJ PAMIĘCI KLUCZY

Pamięć dynamiczna charakteryzuje się samoczynnym kasowaniem jej zawartości w przypadku zaniku zasilania. Jej zawartość można wielokrotnie nadpisywać.

Ramka rozkazu:

nagłówek	C_LoadKeyToDKB	Key1…6	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_LoadKeyToDKB	Zapis klucza do dynamicznej pamięci kluczy	0x14
Key1…6	6 bajtowy klucz	dowolne

Ramka odpowiedzi:

nagłówek	C_LoadKeyToDKB +1		KodOperacji	CRC

6.1.2 ZAPIS KLUCZA DO STATYCZNEJ PAMIĘCI KLUCZY

Pamięć statyczna charakteryzuje się nie kasowaniem jej zawartości w przypadku zaniku zasilania. Jej zawartość można wielokrotnie nadpisywać.

Ramka rozkazu:

nagłówek	C_LoadKeyToSKB	Key1…6, KeyNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_LoadKeyToSKB	Zapis klucza do statycznej pamięci kluczy	0x16
Key1…6	6 bajtowy klucz	dowolne
KeyNo	Numer klucza. W czytniku można zapisać do 32 różnych kluczy.	0x00…0x1f

Ramka odpowiedzi:

nagłówek	C_LoadKeyToSKB +1		KodOperacji	CRC

6.2 ROZKAZY KOMUNIKACJI Z TRANSPONDERAMI

6.2.1 ZAŁĄCZANIE I WYŁĄCZANIE POLA CZYTNIKA

Ramka rozkazu:

nagłówek	C_TurnOnAntennaPower	State	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_TurnOnAntennaPower	Załączanie i wyłączanie pola czytnika, kontrola mechanizmu AutoReader*	0x10
State	stan załączenia pola/AutoReader'a	0x00 – wyłączanie pola/AR 0x01 – załączanie pola/AR

*Tylko gdy Autoreder jest w trybie włączenia na stałe

Uwaga, domyślnie czytnik ma włączony mechanizm AutoReader pracujący w przypadku, kiedy do czytnika nie są wysyłane komendy. Dlatego wywołanie komendy np. wyłączenia pola, spowoduje wyłączenie pola tak długo jak do czytnika będą wysyłane komendy. Po czasie *AofflineTime* (domyślnie 2s), kontrole nad polem przejmie AutoReader.

Ramka odpowiedzi:

nagłówek	C_TurnOnAntennaPower +1		KodOperacji	CRC

6.2.2 WYSELEKCJONOWANIE JEDNEGO TRANSPONDERA MIFARE Z WIELU

Ramka rozkazu:

nagłówek	C_Select	RequestType	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_Select	Wyselekcjonowanie jednego transpondera z wielu	0x12
RequestType	sposób selekcjonowania transpondera	0x00 - Standardowe selekcjonowanie transponderów z grupy tych nie będących w uśpieniu 0x01 - Selekcjonowanie transponderów z grupy wszystkich będących w polu czytnika.

Ramka odpowiedzi:

nagłówek	C_Select +1	ColNo, CardType, ID1…….IDn	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	znaczenie
ColNo	Ilość kolizji podczas selekcjonowania jednego transpondera. Liczba ta może świadczyć ile nie uśpionych transponderów jednocześnie jest w polu.	
CardType	Typ wyselekcjonowanego transpondera	Dla Select podstawowego: 0x50 – S50 0x70 – S70 0x10 – UltraLight 0xdf – DesFire 0xCA – iClass 0xE0 – I-CODE SLI 0x01 – EM4102 0x03 – Hitag-1/S 0x04 – HID PROX II 0x05 – Hitag-2
ID1…IDn	Unikalny numer transpondera	ID1 – LSB, IDn – MSB

6.2.3 ODCZYT OSTATNIO ODCZYTANEGO ID TRANSPONDERA

Rozkaz umożliwia odczytanie zbuforowanego w pamięci ostatnio 'widzianego' ID transponder lub aktualnie znajdującego się w polu. Po wywołaniu tej komendy bufor jest czyszczony, chyba, że transponder nadal znajduje się w polu.

Jeśli ten rozkaz jest wykorzystywany do odczytu ID poprzez cykliczne jego wywoływanie, należy wyłączyć automatyczne wysyłanie ID mechanizmem Autoreader (komenda C_SetAutoreaderConfig, parametr ASerial), aby nie doprowadzić do kolizji danych na magistrali szeregowej.

Ramka rozkazu:

nagłówek	C_GetHistID		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ GetHistID	Odczyt numeru ID	0x08

Ramka odpowiedzi:

nagłówek	C_ GetHistID +1	TagType, ID1...IDn	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	znaczenie
TagType	Typ wyselekcjonowanego transpondera	0x50 – S50 0x70 – S70 0x10 – UltraLight 0xDF – DesFire 0xCA – iClass 0xE0 – I-CODE SLI 0x01 – EM4102 0x03 – Hitag-1/S 0x04 – HID PROX II 0x05 – Hitag-2
ID1…IDn	Unikalny numer transpondera	ID1 – LSB, IDn – MSB

6.2.4 LOGOWANIE DO WYBRANEGO SEKTORA TRANSPONDERA ZA POMOCĄ DYNAMICZNEGO BUFORA KLUCZA

Aby logowanie zakończyło się powodzeniem konieczne jest po każdym załączeniu czytnika, ponowne załadowanie Dynamicznego Bufora Klucza.

Ramka rozkazu:

nagłówek	C_LoginWithDKB	SectorNo, KeyType, DKNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_LoginWithDKB	Logowanie do sektora	0x18
SectorNo	Numer sektora transpondera do którego użytkownik chce się zalogować	**NumeracjaBlokówISektorów
KeyType	Typ klucza, jaki zawarty jest w wewnętrznym Dynamicznym Buforze Klucza	0xAA –klucz typu A 0xBB – klucz typu B
DKNo	Numer dynamicznego klucza	0x00

Ramka odpowiedzi:

nagłówek	C_LoginWithDKB +1		KodOperacji	CRC

6.2.5 LOGOWANIE DO WYBRANEGO SEKTORA TRANSPONDERA ZA POMOCĄ STATYCZNEGO BUFORA KLUCZY

Aby logowanie zakończyło się powodzeniem konieczne jest wcześniejsze załadowanie Statycznego Bufora Kluczy.

Ramka rozkazu:

nagłówek	C_LoginWithSKB	SectorNo, KeyType, SKNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_LoginWithSKB	Logowanie do sektora	0x1a
SectorNo	Numer sektora transpondera do którego użytkownik chce się zalogować	**NumeracjaBlokówISektorów
KeyType	Typ klucza, jaki zawarty jest w wewnętrznym Dynamicznym Buforze Klucza	0xAA –klucz typu A 0xBB – klucz typu B
SKNo	Numer statycznego klucza	0x00…0x1F

Ramka odpowiedzi:

nagłówek	C_LoginWithSKB +1		KodOperacji	CRC

6.2.6 ODCZYT ZAWARTOŚCI BLOKU TRANSPONDERA

Ramka rozkazu:

nagłówek	C_ReadBlock	BlockNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ReadBlock	Odczyt zawartości bloku transpondera	0x1e
BlockNo	Numer bloku w ramach danego sektora	**NumeracjaBlokówISektorów

Ramka odpowiedzi:

nagłówek	C_ReadBlock +1	Data1….. Data16	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
Data1…. Data16	Dane odczytane z bloku transpondera	

6.2.7 ZAPIS ZAARTOŚCI BLOKU TRANSPONDERA

Ramka rozkazu:

nagłówek	C_WriteBlock	BlockNo, Data1….. Data116	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_WriteBlock	Zapis zawartości bloku transpondera	0x1c
BlockNo	Numer bloku w ramach danego sektora	**NumeracjaBlokówISektorów
Data1…. Data16	Dane jakie mają być zapisane w bloku transpondera	dowolne

Ramka odpowiedzi:

nagłówek	C_WriteBlock +1		KodOperacji	CRC

6.2.8 KOPIOWANIE ZAWARTOŚCI BLOKU TRANSPONDERA DO INNEGO BLOKU

Ramka rozkazu:

nagłówek	C_CopyBlock	SourceBlockNo, TargetBlockNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_CopyBlock	Kopiowanie zawartości bloku transpondera do innego bloku	0x60
SourceBlockNo	źródłowy blok	**NumeracjaBlokówISektorów
TargetBlockNo	docelowy blok dla danych	

Ramka odpowiedzi:

nagłówek	C_CopyBlock +1		KodOperacji	CRC

6.2.9 ZAPIS WARTOŚCI DO BLOKU TRANSPONDERA

Ramka rozkazu:

nagłówek	C_WriteValue	BlockNo, BackupBlockNo,Value1...4,	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_WriteValue	Zapis wartości do bloku transpondera	0x34
BlockNo	Numer bloku w ramach danego sektora, w którym Wartość będzie zapisana	**NumeracjaBlokówISektorów
BackupBlockNo	Deklarowany numer bloku zawierający kopię Wartości. BackupBlockNo nie ma to istotnego znaczenia dla działania systemu a użytkownik sam może/powinien zobić kopię Wartości.	**NumeracjaBlokówISektorów
Value1...4	Wartość zapisywana do bloku transpondera	dowolne

Ramka odpowiedzi:

nagłówek	C_WriteValue +1		KodOperacji	CRC

6.2.10 ODCZYT WARTOŚCI Z BLOKU TRANSPONDERA

Ramka rozkazu:

nagłówek	C_ReadValue	BlockNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ReadValue	Odczyt wartości z bloku transpondera	0x36
BlockNo	Numer bloku w ramach danego sektora, z którego Wartość będzie odczytana	**NumeracjaBlokówISektorów

Ramka odpowiedzi:

nagłówek	C_ReadValue+1	Value1...4, BackupBlockNo	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
Value1...4	Wartość odczytana z bloku transpondera	
BackupBlockNo	Numer bloku który może zawierać kopię Wartości	**NumeracjaBlokówISektorów

6.2.11 ZWIĘKSZENIE WARTOŚCI ZAWARTEJ W BLOKU TRANSPONDERA

Aby wykonanie rozkazu przyniosło poprawne rezultaty w deklarowanym bloku dane muszą mieć format „Wartości”.

Ramka rozkazu:

nagłówek	C_IncrementValue	BlockNo, Value1...4	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_IncrementValue	Zwiększenie wartości zawartej w bloku transpondera	0x30
BlockNo	Numer bloku w ramach danego sektora, w którym Wartość będzie modyfikowana	**NumeracjaBlokówISektorów
Value1...4	wartość dodawana do istniejącej rzeczywistej wartości bloku transpondera	

Ramka odpowiedzi:

nagłówek	C_IncrementValue +1		KodOperacji	CRC

6.2.12 ZMNIEJSZANIE WARTOŚCI ZAWARTEJ W BLOKU TRANSPONDERA

Aby wykonanie rozkazu przyniosło poprawne rezultaty w deklarowanym bloku dane muszą mieć format „Wartości".

Ramka rozkazu:

nagłówek	C_DecrementValue	BlockNo, Value1...4	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DecrementValue	Zmniejszanie wartości zawartej w bloku transpondera	0x32
BlockNo	Numer bloku w ramach danego sektora, w którym Wartość będzie modyfikowana	**NumeracjaBlokówISektorów
Value1...4	wartość odejmowana od istniejącej rzeczywistej wartości bloku transpondera	dowolna

Ramka odpowiedzi:

nagłówek	C_DecrementValue+1		KodOperacji	CRC

6.2.13 UŚPIENIE TRANSPONDERA BĘDĄCEGO W POLU

Aby uśpić transponder, musi być on wcześniej wyselekcjonowany.

Ramka rozkazu:

nagłówek	C_Halt		CRC

Ramka odpowiedzi:

nagłówek	C_Halt+1		KodOperacji	CRC

6.2.14 ZAPIS ZAWARTOŚCI STRONY W MIFARE UL

Ramka rozkazu:

nagłówek	C_WritePage4B	PageAdr, Data1...4	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_WritePage4B	Zapis zawartości strony w Mifare UL	0x26
PageAdr	Numer strony w transponderze	0x00…0x0f
Data1...4	Dane jakie mają być zapisane	dowolne

Ramka odpowiedzi:

nagłówek	C_WritePage4B +1		KodOperacji	CRC

6.2.15 ODCZYT ZAWARTOŚCI STRON W MIFARE UL

Ramka rozkazu:

nagłówek	C_ReadPage16B	PageAdr	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ReadPage16B	Odczyt zawartości stron w Mifare UL	0x28
PageAdr	Adres strony począwszy od której powinien rozpocząć się odczyt 4 kolejnych stron. Jeżeli PageAdr>0x???? to nastąpi odczyt stron znajdujących się na początku pamięci.	0x00…0x0f

Ramka odpowiedzi:

nagłówek	C_ReadPage16B +1	Data1…16	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
Data1…16	Odczytane dane z 4 kolejnych stron.	dowolne

6.2.16 UWIERZYTELNIENIE DLA TRANSPONDERA ULTRALIGHT C

Uwaga! Uwierzytelnienie jest możliwe tylko po uprzednim zapisaniu kluczy w pamięci transpondera.

Ramka rozkazu:

nagłówek	C_ULC_Auth	KeyIdx	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ULC_Auth		0x3C
KeyIdx	Indeks klucza zapisanego w czytniku	0x00…0x1f

Ramka odpowiedzi:

nagłówek	C_ ULC_Auth +1		KodOperacji	CRC

6.3 ROZKAZY DO KOMUNIKACJI Z TRANSPONDERAMI MIFARE PLUS

6.3.1 ROZKAZY POZIOMU SL0

6.3.1.1 WRITE PERSO –INICJALIZACJA KARTY

Ramka rozkazu:

nagłówek	C_MfPlusCMD	0xA8, AdrH, AdrL, Data{0..15}	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_MfPlusCMD	Komenda obsługi MFPlus	0x3A
0xA8	Podkomenda ‘Write Perso’	
AdrH, AdrL	Dwubajtwtowy numer bloku lub klucza do zapisania	Zgodnie z dokumentacją Transpondera MFPLUS
Data{0..15}	Klucz lub dane do zapisania	dowolne

Ramka odpowiedzi:

nagłówek	C_ MfPlusCMD +1		KodOperacji	CRC

6.3.1.2 COMMIT PERSO – PRZEJŚCIE DO NASTĘPNEGO POZIOMU SL

Ramka rozkazu:

nagłówek	C_MfPlusCMD	0xAA	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_MfPlusCMD	Komenda obsługi MFPlus	0x3A
0xAA	Podkomenda 'Commit Perso'	

Ramka odpowiedzi:

nagłówek	C_ MfPlusCMD +1		KodOperacji	CRC

6.3.2 ROZKAZY POZIOMU SL1

W tym poziomie transponder Mifare Plus jest kompatybilny z transponderem Mifare Classic. Dostępne są wszystkie komendy związane z obsługą Mifare Classic, dodatkowo wprowadzona została funkcjonalność uwierzytelniania AES.

6.3.2.1 UWIERZYTELNIENIE SL1

Ramka rozkazu:

nagłówek	C_MfPlusCMD	0x10, KeyIdx	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_MfPlusCMD	Komenda obsługi MFPlus	0x3A
0x10	Podkomenda 'Authentication SL1'	
KeyIdx	Indeks klucza AES zapisanego w czytniku	0x00-0x1F

Ramka odpowiedzi:

nagłówek	C_ MfPlusCMD +1		KodOperacji	CRC

6.3.2.2 PRZEJŚCIE NA WYŻSZY POZIOM SL/ SPRAWDZENIE ORYGINALNOŚCI TRANSPONDERA

Przejście na wyższy poziom SL lub sprawdzenie oryginalności następuje po poprawnej autoryzacji AES z odpowiednim identyfikatorem klucza.

Ramka rozkazu:

nagłówek	C_MfPlusCMD	0x70, AdrH, AdrL, KeyIdx	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_MfPlusCMD	Komenda obsługi MFPlus	0x3A
0x70	Podkomenda 'First Auth'	
AdrH, AdrL	Dwubajtwtowy numer bloku lub klucza do zapisania	0x9002 – przejście do poziomu SL2 0x9003 – przejście do poziomu SL3 0x8000 – sprawdzenie oryginalności transpondera
KeyIdx	Indeks klucza AES zapisanego w czytniku	0x00-0x1F

Ramka odpowiedzi:

nagłówek	C_ MfPlusCMD +1		KodOperacji	CRC

6.3.3 ROZKAZY POZIOMU SL3

6.3.3.1 WPROWADZENIE TRANSPONDERA W TRYB ISO14443-4

Każda komenda związana z SL3 musi być poprzedzona jednorazowym wprowadzeniem transpondera w tryb zgodności z ISO14443-4

Ramka rozkazu:

nagłówek	C_Init_ISO14443-4	CID	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_Init_ISO14443-4		0x3E
CID	Identyfikator CID	0x00

Ramka odpowiedzi:

nagłówek	C_Init_ISO14443-4+1		KodOperacji	CRC

6.3.3.2 LOGOWANIE DO SEKTORA

Ramka rozkazu:

nagłówek	C_MfPlusCMD	0x1A, Sector, KeyType, KeyIdx	CRC

Gdzie:

Nazwa parametru	**Opis parametru**	**Zakres wartości**
C_MfPlusCMD	Komenda obsługi MFPlus	0x3A
0x1A	Podkomenda 'sector login'	
Sector	Numer sektora	0x00-0x1f – karta Plus 2K 0x00-0x27 – karta Plus 4k
KeyType	Typ klucza	0xAA – klucz A 0xBB – klucz B
KeyIdx	Indeks klucza AES zapisanego w czytniku	0x00-0x1F

Ramka odpowiedzi:

nagłówek	C_ MfPlusCMD +1		KodOperacji	CRC

6.3.3.3 ODCZYT ZAWARTOŚCI BLOKU TRANSPONDERA

Ramka rozkazu:

nagłówek	C_ MfPlusCMD	read_cmd, block	CRC

Gdzie:

Nazwa parametru	**Opis parametru**	**Zakres wartości**
C_MfPlusCMD	Komenda obsługi MFPlus	0x3A
read_cmd	Typ procedury odczytu:	0x30-0x33
block	Numer bloku do odczytu	0-3 dla sektorów<32 0-15 dla sektorów>32

cmd.	**MAC on command**	**MAC on resonse**	**Plain /encrypted**
0x30	Yes	No	Encrypted*
0x31	Yes	Yes	Encrypted*
0x32	Yes	No	Plan
0x33	Yes	Yes	Plan

*tylko transpondery Plus X

Ramka odpowiedzi:

nagłówek	C_ MfPlusCMD +1	Data1….. Data16	KodOperacji	CRC

Gdzie:

Nazwa parametru	**Opis parametru**	**Zakres wartości**
Data1…. Data16	Dane odczytane z bloku transpondera	

6.3.3.4 ZAPIS ZAWARTOŚCI BLOKU TRANSPONDERA

Ramka rozkazu:

nagłówek	C_ MfPlusCMD	write_cmd, block, data0..data15	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_MfPlusCMD	Komenda obsługi MFPlus	0x3A
write _cmd	Typ procedury zapisu:	0xA0-0xA3
block	Numer bloku do odczytu	0-3 dla sektorów<32 0-15 dla sektorów>32
data0..data15	Dane do zapisu bloku transpondera	

cmd.	MAC on command	MAC on resonse	Plain /encrypted
0xA0	Yes	No	Encrypted*
0xA1	Yes	Yes	Encrypted*
0xA2	Yes	No	Plain
0xA3	Yes	Yes	Plain

*tylko transpondery Plus X

Ramka odpowiedzi:

nagłówek	C_ MfPlusCMD +1		KodOperacji	CRC

6.3.4 CZASY TRWANIA OPERACJI DLA MIFARE PLUS

Poniższe zestawienie określa czas trwania poszczególnych operacji, liczony od momentu zakończenia wysyłania ramki komendy (RS) do momentu rozpoczęcia wysyłania ramki odpowiedzi(RS)

Operacja	Rezultat poprawny [ms]	Rezultat niepoprawny [ms]
SELECT	14	12
LOGIN SL3	25	100
READ BLOCK	10	100
WRITE BLOCK	13	100

6.4 OBSŁUGA TRANSPONDERÓW DESFIRE, DESFIRE EV1

6.4.1 AUTORYZACJA, LOGOWANIE DO AKTUALNIE WYSELEKCJONOWANEJ APLIKACJI

Ramka rozkazu:

nagłówek	C_DesAuth (0x42)	KeyNo{0..0x10}, KeyIdx, AuthType	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesAuth	Komenda autoryzacji	0x42
KeyNo	Numer klucza w odniesieniu do transpondera	0x00..0x10
KeyIdx	Indeks klucza AES zapisanego w czytniku	0x00-0x1F
AuthType	Typ autoryzacji : 0x0A – DES 0x3A – 3DES 0xAA – AES	0x0A, 0x3A, 0xAA

Ramka odpowiedzi:

nagłówek	C_DesAuth +1		KodOperacji	CRC

6.4.2 ZMIANA USTAWIEŃ KLUCZA MASTER AKTUALNIE WYBRANEJ APLIKACJI

Ramka rozkazu:

nagłówek	C_DesChangeKeySett (0x44)	KeySettings	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości

C_DesChangeKeySett	Komenda zmiany ustawień klucza	0x44
KeySettings	Bajt konfiguracyjny	0x00..0x0f

Ramka odpowiedzi:

nagłówek	C_DesChangeKeySett+1		KodOperacji	CRC

Struktura bajtu konfiguracyjnego *KeySettings*:

Bit	Znaczenie
0	0 – klucz PICC Master key jest niemodyfikowalny
	1* – klucz PICC Master key jest modyfikowalny
1	0 – wywołanie funkcji C_DesGetAppIDs wymaga autoryzacji z użyciem PICC Master key
	1* – wywołanie funkcji C_DesGetAppIDs nie wymaga autoryzacji
2	0 – utworzenie/usunięcie aplikacji wymaga autoryzacji z użyciem PICC Master key
	1* -utworzenie nowej aplikacji nie wymaga autoryzacji, usunięcie aplikacji wymaga autoryzacji kluczem danej aplikacji lub PICC Master key
3	0 – zmiana konfiguracji klucza PICC Master Key jest niemożliwa
	1* - zmiana konfiguracji klucza PICC Master Key dozwolona w przypadku autoryzacji z użyciem tego klucza
4	RFU – 0
5	RFU – 0
6	RFU – 0
7	RFU – 0

* - ustawienie domyślne

6.4.3 ZMIANA KLUCZA

Ramka rozkazu:

nagłówek	C_DesChangeKey (0x46)	KeyNo, NewEESavedKey,[PrevEESavedKey]	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesChangeKey	Komenda zmiany klucza	0x46
KeyNo	Numer klucza do zmiany	0x00..0x0D
NewEESavedKey	Indeks nowego klucza zapisanego w pamięci czytnika	0x00..0x13
PrevEESavedKey	• Jeśli zmieniany klucz nie jest tym, którym nastąpiła aktualna autoryzacja, podajemy indeks aktualnego klucza, który będzie zmieniany • Jeśli zmieniany klucz jest tym samym, którym nastąpiła aktualna autoryzacja, parametr ten pozostawiamy pusty	0x00..0x13

Ramka odpowiedzi:

nagłówek	C_DesChangeKey+1		KodOperacji	CRC

6.4.4 TWORZENIE APLIKACJI

Ramka rozkazu:

nagłówek	C_DesCreateApp (0x48)	AId1..3,KeySettings1, KeySettings2	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesCreateApp	Komenda tworzenia aplikacji	0x48
AId1..3	3-bajtowy identyfikator aplikacji	0x00..0xFF
KeySettings1	Bajt konfiguracyjny (patrz poniżej)	0x00..0x0F
KeySettings2	Bit3..bit0: Liczba kluczy przypisanych do danej aplikacji Bit7..Bit6: 00 – autoryzacja DES dla całej aplikacji 10- autoryzacja AES dla całej aplikacji	0x00..0x0D

Ramka odpowiedzi:

nagłówek	C_DesCreateApp +1		KodOperacji	CRC

Struktura bajtu konfiguracyjnego *KeySettings*:

Bit	Znaczenie
0	0 – klucz Application Master key jest niemodyfikowalny
	1* – klucz Application Master key jest modyfikowalny, wymaga autoryzacji z użyciem dotychczasowego klucza AppMasterKey
1	0 – wywołanie funkcji C_DesGetAppIDs wymaga autoryzacji z użyciem PICC Master key
	1* – wywołanie funkcji C_DesGetAppIDs nie wymaga autoryzacji
2	0 – utworzenie/usunięcie pliku wymaga autoryzacji z użyciem AppMasterKey
	1* -utworzenie/usunięcie pliku nie wymaga autoryzacji z użyciem AppMasterKey
3	0 – zmiana konfiguracji klucza Application Master Key jest niemożliwa
	1* - zmiana konfiguracji klucza Application Master Key dozwolona w przypadku autoryzacji z użyciem tego klucza
4	Bit7-Bit4: określają prawa do zmian parametrów klucza
5	0x0*:Klucz Master aplikacji jest niezbędny do zmiany ustawień kluczy
6	0x1-0xD : autoryzacja przy pomocy klucza z tym indeksem jest konieczna do zmiany ustawień kluczy
7	0xE :zmiana ustawień klucza wymaga autoryzacji z użyciem tego samego klucza

* - ustawienie domyślne

6.4.5 USUWANIE APLIKACJI

Ramka rozkazu:

nagłówek	C_DesDeleteApp (0x4a)	AId1..3	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesDeleteApp	Komenda usuwania aplikacji	0x4a
AId1..3	3-bajtowy identyfikator aplikacji	0x00..0xFF

Ramka odpowiedzi:

nagłówek	C_DesCreateApp +1		KodOperacji	CRC

6.4.6 POBIERANIE LISTY APLIKACJI

Ramka rozkazu:

nagłówek	C_DesGetAppIDs (0x4c)		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesGetAppIDs	Komenda pobierania listy aplikacji	0x4c

Ramka odpowiedzi:

nagłówek	C_DesGetAppIDs +1	N*{Aid3,Aid2,Aid1}	KodOperacji	CRC

Zwracana jest lista numerów Aid, aktualnie istniejących aplikacji

6.4.7 WYBÓR APLIKACJI

Ramka rozkazu:

nagłówek	C_DesSelectApp (0x4e)	Aid1..3	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesSelectApp	Komenda wyboru aplikacji	0x4e
Aid1..3	3 bajtowy identyfikator aplikacji	0x00-0xff

Ramka odpowiedzi:

nagłówek	C_DesSelectApp+1		KodOperacji	CRC

6.4.8 FORMATOWANIE TRANSPONDERA

Ramka rozkazu:

nagłówek	C_DesFormatPICC (0x60)		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesFormatPICC	Komenda formatowania transpondera	0x60

Wykonanie tej komendy wymaga autoryzacji z użyciem klucza PICC Master key

Ramka odpowiedzi:

nagłówek	C_DesFormatPICC +1		KodOperacji	CRC

6.4.9 INICJALIZACJA PROTOKOŁU TRANSMISJI Z TRANSPONDERAMI DESFIRE (ISO14443-4)

Ramka rozkazu:

nagłówek	C_DesInitProtocol (0x3e)	CID	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesInitProtocol	Inicjalizacja ISO1444-4	0x3E
CID	Logiczny numer wyselekcjonowanego transpondera	0x00-0x0E

Komenda ta musi wystąpić bezpośrednio po wyselekcjonowaniu transpondera komendą C_Select. Obecna wersja czytnika pozwala na pracę z jednym transponderem Desfire jednocześnie. Numer logiczny CID nie ma aktualnie znaczenia, zaleca się podawanie numeru 0

Ramka odpowiedzi:

nagłówek	C_DesInitProtocol +1		KodOperacji	CRC

6.4.10 POBIERANIE LISTY PLIKÓW AKTUALNIE WYBRANEJ APLIKACJI

Ramka rozkazu:

nagłówek	C_DesGetFileIDs (0x64)		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesGetFileIDs	Komenda pobierania listy plików	0x64

Ramka odpowiedzi:

nagłówek	C_DesGetAppIDs +1	N*FileNo	KodOperacji	CRC

Zwracana jest lista numerów plików aktualnie istniejących w wybranej aplikacji

6.4.11 POBIERANIE WŁAŚCIWOŚCI PLIKU

Ramka rozkazu:

nagłówek	C_DesGetFileSett (0x66)	FileNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesGetFileSett	Komenda pobierania właściwości pliku	0x66
FileNo	Identyfikator pliku	0x00-0x0f

Ramka odpowiedzi:

nagłówek	C_DesGetAppIDs +1	File params...	KodOperacji	CRC

W zależności od typu pliku zwracana jest informacja w następującym formacie:

- Dla plików Standard Data Files i Backup Data Files

1 byte	1 byte	2 bytes		3 bytes	
File type	Comm. Sett.	Access right		File size	
		LSB	MSB	LSB	MSB

- Dla plików *Value Files* (ten typ aktualnie nie jest zaimplementowany)

1 byte	1 byte	2 bytes		4 bytes		4 bytes		4 bytes		1 byte
File type	Comm. Sett.	Access right		Lower limit		Upper limit		Limited credit value		Limited credit enable
		LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB	

- Dla plików Linear/Cyclic record files

1 byte	1 byte	2 bytes		3 bytes		3 bytes		3 bytes	
File type	Comm. Sett.	Access right		Record size		Maximum number of records		Current number of records	
		LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

6.4.12 TWORZENIE PLIKÓW TYPU *STANDARD DATA FILES*

Ramka rozkazu:

nagłówek	C_DesCreateSTDataFile (0x68)	FileNo,ComSett,AccRight1..2,FileSize1..3	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesCreateSTDataFile	Komenda tworzenia pliku STD	0x68
FileNo	Identyfikator pliku	0..0x0F
ComSett	Typ transmisji: 0x01 – nieszyfrowana 0x03 – szyfrowana DES	0x00,0x03
AccRight1..2	Prawa dostępu do pliku, patrz tabela poniżej	0x00..0xff
FileSize1..3	3 bajtowa wielkość pliku w bajtach, w kolejności LSB..MSB	0x00-0xff

Bajty określające prawa dostępu:

15	12	11	8	7	4	3	0
Read Access		Write Access		Read & Write Access		Change Right Access	
MBS	1st byte					2nd byte	LSB

Dwa bajty praw dostępu podzielone są na 4 pola 4 bitowe. Każde pole może zawierać wartości z przedziału 0x0 – 0xF

- Wartości z przedziału 0x0 – 0xD określają numer klucza, który będzie miał prawa do wykonania danej operacji,
- Wartość 0xE oznacza, że dana operacja nie wymaga autoryzacji
- Wartość 0xF oznacza, że nie ma dostępu do danej operacji, bez względu na użyty klucz

Ramka odpowiedzi:

nagłówek	C_DesCreateSTDataFile +1		KodOperacji	CRC

6.4.13 TWORZENIE PLIKÓW TYPU *BACKUP DATA FILES*

Ramka rozkazu:

nagłówek	C_DesCreateBACKDataFile (0x6a)	FileNo,ComSett,AccRight1..2,FileSize1..3	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesCreateBACKDataFile	Komenda tworzenia pliku BACKUP	0x6a

FileNo	Identyfikator pliku	0..0x07
ComSett	Typ transmisji: 0x01 – nieszyfrowana 0x03 – szyfrowana DES	0x00,0x03
AccRight1..2	Prawa dostępu do pliku	0x00..0xff
FileSize1..3	3 bajtowa wielkość pliku w bajtach w kolejności LSB..MSB	0x00-0xff

Ramka odpowiedzi:

nagłówek	C_DesCreateBACKDataFile +1		KodOperacji	CRC

Prawa dostępu określa się identycznie jak w przypadku plików *Standard Data Files*

Zapis pliku typu *Backup Data file* musi zakończyć się wydaniem komendy C_DesCommit.

6.4.14 TWORZENIE PLIKÓW TYPU LINEAR/CYCLIC RECORD FILES

Ramka rozkazu:

nagłówek	C_DesCreateRecordFile (0x6c)	FileNo, ComSett, AccRight1..2, RecSize1..3, RecNumb1..3, Cy/Li{0x0C,0x01}	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesCreateRecordFile	Komenda tworzenia pliku typu *Record File*	0x6c
FileNo	Identyfikator pliku	0..0x0F
ComSett	Typ transmisji: 0x01 – nieszyfrowana 0x03 – szyfrowana DES	0x00,0x03
AccRight1..2	Prawa dostępu do pliku	0x00..0xff
RecSize1..3	3 bajtowy rozmiar rekordu w bajtach, w kolejności LSB..MSB	0x00-0xff
RecNumb1..3	3 bajtowy parametr określający ilość rekordów, kolejność LSB..MSB	
Cy/Li	0x0c- typ cykliczny 0x0l – typ liniowy	0x0C,0x01

Ramka odpowiedzi:

nagłówek	C_DesCreateRecordFile+1		KodOperacji	CRC

Prawa dostępu określa się identycznie jak w przypadku plików *Standard Data Files*

6.4.15 USUWANIE PLIKU

Ramka rozkazu:

nagłówek	C_DesDeleteFile (0x6e)	FileNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesDeleteFile	Komenda usuwania pliku	0x6e
FileNo	Identyfikator pliku	0x00..0x0F

Ramka odpowiedzi:

nagłówek	C_DesDeleteFile+1		KodOperacji	CRC

6.4.16 ZMIANA USTAWIEŃ PLIKU

Ramka rozkazu:

nagłówek	C_DesChangeFileSett (0x80)	FileNo, ComSett, AccRight1..2	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości

C_DesChangeFileSett	Komenda zmiany właściwości pliku	0x80
FileNo	Identyfikator pliku	0..0x0F
ComSett	Typ transmisji: 0x01 – nieszyfrowana 0x03 – szyfrowana DES	0x00,0x03
AccRight1..2	Prawa dostępu do pliku	0x00..0xff

Ramka odpowiedzi:

nagłówek	C_DesChangeFileSett+1		KodOperacji	CRC

Prawa dostępu określa się identycznie jak w przypadku tworzenia plików *Standard Data Files*

6.4.17 ODCZYT DANYCH Z PLIKU TYPU *STD/BACK DATA FILE*

Ramka rozkazu:

nagłówek	C_DesReadData (0x82)	FileNo, Offset1..3, Length1..3	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesReadData	Komenda odczytu z pliku	0x82
FileNo	Identyfikator pliku	0..0x0F
Offset1..3	3 bajtowy parametr określający miejsce od którego zaczynamy czytać plik, kolejność LSB..MSB	0x00-0xFF
Length1..3	3 bajtowy parametr określający ilość bajtów, które chcemy odczytać, kolejność LSB..MSB (jednorazowo odczytać można do 58 bajtów)	0x00-0x3A

Ramka odpowiedzi:

nagłówek	C_DesReadData +1	n Bytes	KodOperacji	CRC

6.4.18 ZAPIS DANYCH DO PLIKU TYPU *STD/BACK DATA FILE*

Ramka rozkazu:

nagłówek	C_DesWriteData (0x84)	FileNo, Offset1..3,Data1..58	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesWriteData	Komenda zapisu do pliku	0x84
FileNo	Identyfikator pliku	0..0x0F
Offset1..3	3 bajtowy parametr określający miejsce od którego zaczynamy zapisywać, kolejność LSB..MSB	0x00-0xFF
Data1..58	Dane, które zamierzamy zapisać do pliku, (jednorazowo zapisać można do 58bajtów)	0x00-0xFF

Ramka odpowiedzi:

nagłówek	C_DesWriteData+1		KodOperacji	CRC

6.4.19 ZAPIS REKORDU DO PLIKU TYPU *RECORD DATA FILE*

Ramka rozkazu:

nagłówek	C_DesWriteRecord (0x86)	FileNo, Offset1..3,Data1..58	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesWriteRecord	Komenda zapisu rekordu	0x86
FileNo	Identyfikator pliku	0..0x0F
Offset1..3	3 bajtowy parametr określający miejsce od którego zaczynamy zapisywać, kolejność LSB..MSB (wartość ta musi być mniejsza niż wielkość pojedynczego rekordu)	0x00-0xFF
Data1..58	Dane, które zamierzamy zapisać do pliku,	0x00-0xFF

(jednorazowo zapisać można do 58bajtów, suma tej wartości oraz offsetu musi być mniejsza niż wielkość pojedynczego rekordu)

Ramka odpowiedzi:

nagłówek	C_DesWriteRecord+1		KodOperacji	CRC

Uwaga: Zapis rekordu do pliku typu *Record File* musi zakończyć się wydaniem komendy C_DesCommit.

6.4.20 ODCZYT REKORDU Z PLIKU TYPU *RECORD DATA FILE*

Ramka rozkazu:

nagłówek	C_DesReadRecord (0x88)	FileNo, WhichRecord1..3,NoOfRecords1..3	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesReadRecord	Komenda odczytu rekordu	0x88
FileNo	Identyfikator pliku	0..0x0F
WhichRecord1..3	3 bajtowy parametr określający rekord od którego zaczynamy czytać, kolejność LSB..MSB	0x00-0xFF
NoOfRecords1..3	3 bajtowy parametr określający ilość rekordów do przeczytania, kolejność LSB..MSB	0x00-0xFF

Ramka odpowiedzi:

nagłówek	C_DesReadRecord +1	Record data...	KodOperacji	CRC

Ilość odczytanych danych nie może być większa niż 58 bajtów, stąd należy zachować zasadę: {NoOfRecords1..3}*rozmiar_rekordu < 58bytes

6.4.21 CZYSZCZENIE PLIKÓW TYPU *RECORD DATA FILE*

Ramka rozkazu:

nagłówek	C_DesClearRecordFile (0x8a)	FileNo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesClearRecordFile	Komenda czyszczenia pliku rekordowego	0x8a
FileNo	Identyfikator pliku	0..0x0F

Ramka odpowiedzi:

nagłówek	C_DesClearRecordFile+1		KodOperacji	CRC

Uwaga: Operacja ta musi zakończyć się wydaniem komendy C_DesCommit.

6.4.22 KOMENDA POTWIERDZAJĄCA - *DESCOMMIT*

Ramka rozkazu:

nagłówek	C_DesCommit (0x8c)		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesCommit	Komenda potwierdzenia	0x8c

Ramka odpowiedzi:

nagłówek	C_DesCommit+1		KodOperacji	CRC

6.4.23 DESELEKCJA TRANSPONDERA

Ramka rozkazu:

nagłówek	C_DesDeselect (0x8e)		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_DesDeselect	Komenda de-selekcjonująca transponder	0x8e

Ramka odpowiedzi:

nagłówek	C_DesDeselect+1		KodOperacji	CRC

6.5 TRANSMISJA DANYCH I-BLOCK PROTOKOŁU T=CL ISO14443-4

Komenda ta umożliwia wysłanie danych do transpondera w trybie ISO14443-4, jednocześnie zwraca informacje z transpondera. Przed wykonaniem tej komendy konieczne jest przejście w tryb ISO14443-4 za pomocą komendy C_Init_ISO14443-4.

Ramka rozkazu:

nagłówek	C_TranscIBlock	data	CRC

Gdzie:

Nazwa parametru	**Opis parametru**	**Zakres wartości**
C_TranscIBlock		0xC8
data	Dane pakietu I-Block	dowolne

Ramka odpowiedzi:

nagłówek	C_TranscIBlock+1	data	KodOperacji	CRC

6.6 OBSŁUGA TRANSPONDERÓW RODZINY I-CODE SLI

6.6.1 ODCZYT NUMERU ID TRANSPONDERA I-CODE SLI

Ramka rozkazu:

nagłówek	C_Inventory		CRC

Gdzie:

Nazwa parametru	**Opis parametru**	**Zakres wartości**
C_Inventory	Odczyt numeru ID	0x04

Ramka odpowiedzi:

nagłówek	C_Inventory +1	0,CardType,ID1...ID8	KodOperacji	CRC

6.6.2 ODCZYT STRONY TRANSPONDERA SLI

Ramka rozkazu:

nagłówek	C_SLIReadPage	PageAdr	CRC

Gdzie:

Nazwa parametru	**Opis parametru**	**Zakres wartości**
C_ SLIReadPage	Odczyt zawartości stron w Mifare UL	0x2C
PageAdr	Adres strony zgodny z obsługiwanym typem transpondera	

Ramka odpowiedzi:

nagłówek	C_ SLIReadPage +1	Data1…4	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
Data1…4	Odczytane dane.	dowolne

6.6.3 ZAPIS ZAWARTOŚCI STRONY W SLI

Ramka rozkazu:

nagłówek	C_SLIWritePage	PageAdr, Data1...4	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ SLIWritePage	Zapis zawartości strony w SLI	0x2E
PageAdr	Numer strony w transponderze	
Data1...4	Dane jakie mają być zapisane	dowolne

Ramka odpowiedzi:

nagłówek	C_ SLIWritePage +1		KodOperacji	CRC

6.7 ODCZYT NUMERU ID TRANSPONDERA ICLASS

Ramka rozkazu:

nagłówek	C_ICLASSInventory		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ICLASSInventory	Odczyt numeru ID	0x0A

Ramka odpowiedzi:

nagłówek	C_ICLASSInventory +1	0,CardType,ID1...ID8	KodOperacji	CRC

*możliwe jest odczytanie tylko numeru CSN

6.8 WEJŚCIA I WYJŚCIA ELEKTRYCZNE

Czytnik posiada konfigurowalne wejścia/wyjścia.

6.8.1 ZAPIS STANU WYJŚCIA

Ramka rozkazu:

nagłówek	C_WriteOutputs	IONo, State	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_WriteOutputs	Zapis stanu wyjścia	0x70
IONo	Numer portu IO. Port powinien być skonfigurowany jako wyjście	0x00-0x03
State	Żądany stan wyjścia	0x00 lub 0x01

Ramka odpowiedzi:

nagłówek	C_WriteOutputs +1		KodOperacji	CRC

6.8.2 ODCZYT STANU WEJŚCIA

Ramka rozkazu:

nagłówek	C_ReadInputs	IONo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ReadInputs	Odczyt stanu wejścia	0x72
IONo	Numer portu IO. Powinien on być skonfigurowany jako wejście.	0x0..0x1

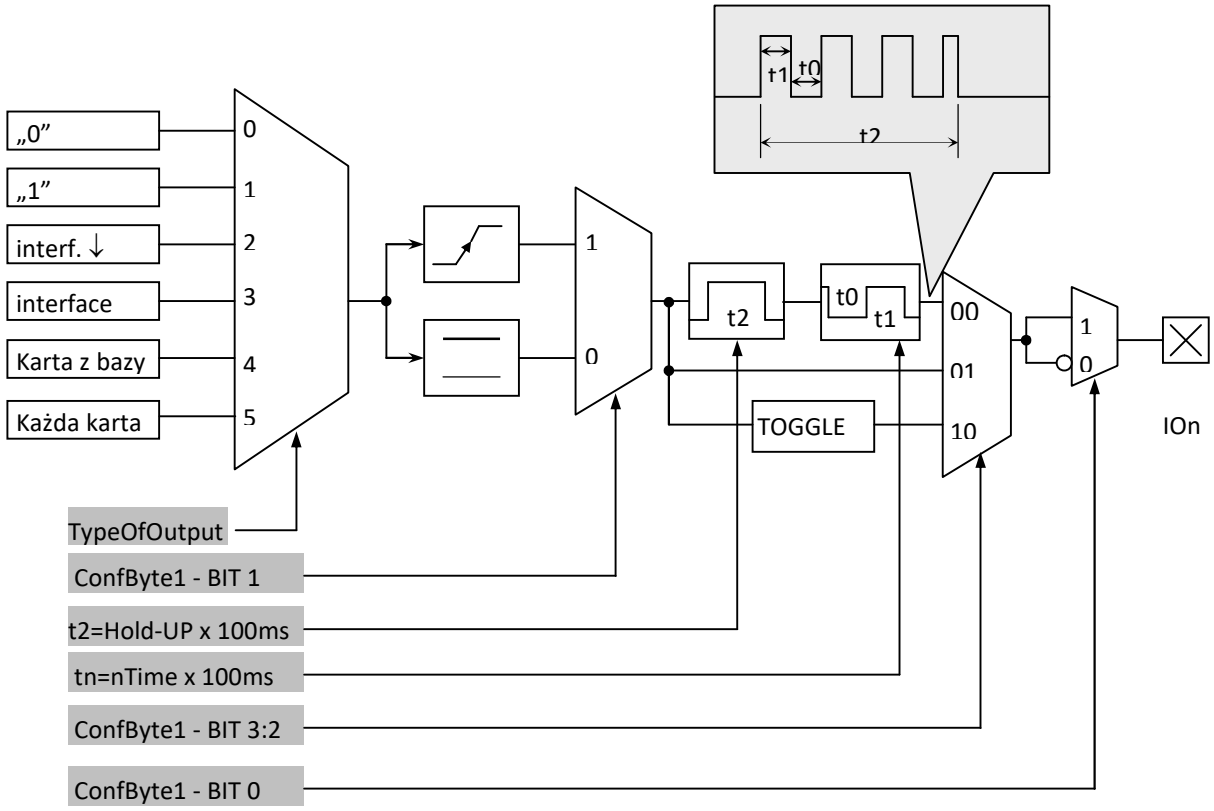
Ramka odpowiedzi:

nagłówek	C_ReadInputs +1	State,[COUNTER]	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
State	Odczytany stan wejścia	
Counter	Stan licznika dla wejścia typu licznikowego	

Zapis konfiguracji dowolnego portu



Ramka rozkazu:

C_SetIOConfig	IONo, IOConfigData1…n

Jeżeli Konfigurujemy port jako wyjście to parametry IOConfigData1…n mają postać:
Dir, ConfByte1, TypeOfOutput, Hold-UP, 0Time, 1Time

Gdzie:

Nazwa parametru	**Opis parametru**	**Zakres wartości**
C_SetIOConfig	Zapis konfiguracji dowolnego portu	0x50
IONo	Numer portu IO, który ma być skonfigurowany	0x0..0x3
Dir	kierunek portu	0x00 – wyjście
ConfByte1	Jeden bajt w którym: BIT0 określa typ wyjścia jako Normalnie otwarte lub Normalnie Zamknięte. BIT 1 określa sposób reakcji danego wyjścia jako reagujące na zmianę pobudzenia (reagujące na zbocze) lub reagujące na stan pobudzenia (reagujące na stan). BIT3:2 określa sposób zachowania wyjścia w stosunku do stanu sygnału wyzwolenia	ConfByte1 Bit 0 0-Normalnie Zamknięte 1-Normalnie Otwarte ConfByte1 Bit 1 0-reaguje na poziom 1-reaguje na zbocze ConfByte1 Bit 3:2 00 – generator fali prostokątnej 01-bezpośrednio 10 – zmiana stanu wyjścia
TypeOfOutput	źródło sygnału sterującego	0x00 – wyłączone na stałe 0x01 – załączone na stałe 0x02 – sterowane poprzez interface szeregowy automatycznie powracające do zera 0x03 - sterowane poprzez interface szeregowy 0x04 – RFU

		0x05 – ustawiane w przypadku przyłożenia do czytnika dowolnej karty
Podtrzymanie	Czas podtrzymania stanu załączenia po ustaniu pobudzenia. Czas ten wyrażony jest jako: Podtrzymanie x 100ms Podczas trwania czasu „Podtrzymanie" można skonfigurować wyjście potrafiące generować falę prostokątną. Czas jedynki i czas zera ustawiany jest następnymi parametrami:	
0Time	czas logicznego zera	
1Time	czas logicznej jedynki	

Jeżeli konfigurujemy port jako wejście to parametry IOConfigData1…n mają postać:
Dir, Triger, TypeOfInput, RFU1, RFU2, RFU3

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_SetIOConfig	Zapis konfiguracji dowolnego portu	0x50
IONo	Numer portu IO, który ma być skonfigurowany	0x0..0x3
Dir	kierunek portu.	0x01 – wejście
Triger	Bajt określający sposób wyzwalania wejścia	0x00 niezanegowane 0x01 zanegowane 0x02 reakcja na zbocze rosnące 0x03 reakcja na zbocze opadające
TypeOfInput	Typ wejścia: Standardowe – dostajemy informacje o stanie wejścia uwzględniając ustawienie Triger Licznikowe – jednobajtowy licznik zlicza ilość zboczy, które pojawiły się na wejściu. Licznik jest odczytywany i kasowany komendą C_ReadInputs	0x00 „0" na stałe 0x01 „1" na stałe 0x02 wejście standardowe 0x04 wejście licznikowe
RFU1-RFU3	Zarezerwowane	0x00

Nie wszystkie porty CTU-Sxx mają dowolny kierunek.
W celu poprawnej konfiguracji należy dla danego portu podać poprawny kierunek.

Spis istniejących portów, którymi można sterować w CTU-S		
Numer portu	kierunek	Opis
0	wejście/wyjście	GPIO1
1	wejście/wyjście	GPIO2
2	wyjście	PRZEKAŹNIK
3	wyjście	BUZZER

Ramka odpowiedzi:

nagłówek	C_SetIOConfig +1		KodOperacji	CRC

6.8.3 ODCZYT KONFIGURACJI DOWOLNEGO PORTU

Ramka rozkazu:

nagłówek	C_GetIOConfig	IONo	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_GetIOConfig	Odczyt konfiguracji dowolnego portu	0x52

IONo	Numer portu IO, który którego konfiguracja ma być odczytana	0x00…0x03

Ramka odpowiedzi:

nagłówek	C_GetIOConfig +1	IOConfigData1…n	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
IOConfigData1…n	ma postać taką samą jak przy zapisie konfiguracji	

6.9 HASŁO DOSTĘPU

6.9.1 LOGOWANIE DO CZYTNIKA

Ramka rozkazu:

nagłówek	C_LoginUser	Data1…n, 0x0	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_LoginUser	Logowanie do czytnika	0xb2
Data1…n	jest dowolnym łańcuchem bajtów	Dowolne z zakresu 0x01…0xff. Długość łańcucha może wynosić od 0 do 8 bajtów
0x00	Zero kończące string	0x00

Ramka odpowiedzi:

nagłówek	C_LoginUser +1		KodOperacji	CRC

6.9.2 ZMIANA HASŁA

Ramka rozkazu:

nagłówek	C_ChangeLoginUser	Data1…n, 0x0	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ChangeLoginUser	Zmiana hasła	0xb4
Data1…n	jest dowolnym łańcuchem bajtów który będzie obowiązującym hasłem dostępu.	Dowolne z zakresu 0x01…0xff. Długość łańcucha może wynosić od 0 do 8 bajtów
0x00	Zero kończące string	0x00

Jeżeli Data1=0x00 to czytnik nie będzie chroniony hasłem. W dowolnym momencie można ustalić nowe hasło, tak aby czytnik był chroniony hasłem.

Ramka odpowiedzi:

nagłówek	C_ChangeLoginUser+1		KodOperacji	CRC

6.9.3 WYLOGOWANIE Z CZYTNIKA

Rozkaz ten dezaktualizuje podane ostatnio hasło.

Ramka rozkazu:

nagłówek	C_LogoutUser		CRC

Nazwa parametru	Opis parametru	Zakres wartości
C_LogoutUser	Wylogowanie z czytnika	0xd6

Ramka odpowiedzi:

nagłówek	C_LogoutUser +1		KodOperacji	CRC

6.10 OBSŁUGA WEWNĘTRZNEJ PAMIĘCI TRANSPONDERÓW

6.10.1 ODCZYT NUMERU TRANSPONDERA Z PAMIĘCI

Ramka rozkazu:

nagłówek	C_CardMemoryRead	AdrL, AdrH	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_CardMemoryRead	Odczyt numeru transpondera z pamięci	0x20
AdrL, AdrH	odpowiednio młodszy i starszy bajt adresu	0-999*

*Adresy od 0 do 989 to adresy kart użytkowników. Adresy 990-999 to adresy kart Master.

Ramka odpowiedzi:

nagłówek	C_CardMemoryRead	+1	ID1(L)….ID5(H), Right	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
ID1(L)….ID5(H)	5 bajtów numeru transpondera	
Right	prawa dostępu dla danego transpondera	0x01

6.10.2 ZAPIS NUMERU TRANSPONDERA DO PAMIĘCI

Ramka rozkazu:

nagłówek	C_CardMemoryWrite	AdrL, AdrH, ID1(L)….ID5(H), Right	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_CardMemoryWrite	Zapis numeru transpondera do pamięci	0x22
AdrL, AdrH	odpowiednio młodszy i starszy bajt adresu	0-999*
ID1(L)….ID5(H)	5 bajtów numeru transpondera	Dowolne 5 bajtów
Right	prawa dostępu lub funkcja pełniona przez transponder	0x01

*Adresy od 0 do 989 to adresy kart użytkowników. Adresy 990-999 to adresy kart Master.

Ramka odpowiedzi:

nagłówek	C_CardMemoryWrite	+1		KodOperacji	CRC

6.11 OBSŁUGA WBUDOWANEJ KONTROLI DOSTĘPU

6.11.1 ZAPIS KONFIGURACJI KONTROLI DOSTĘPU

Ramka rozkazu:

nagłówek	C_AccesControllConfigWrite	Mode	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_AccesControllConfigWrite	Zapis konfiguracji kontroli dostępu	0x74
Mode	Tryb pracy modułu kontroli dostępu	0x00 – moduł wyłączony 0x01 – moduł załączony

Ramka odpowiedzi:

nagłówek	C_AccesControllConfigWrite+1		KodOperacji	CRC

6.11.2 ODCZYT KONFIGURACJI KONTROLI DOSTĘPU

Ramka rozkazu:

nagłówek	C_AccesControllConfigRead		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_ AccesControllConfigRead	Odczyt konfiguracji kontroli dostępu	0x76

Ramka odpowiedzi:

nagłówek	C_AccesControllConfigRead+1	Mode	KodOperacji	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
Mode	Tryb pracy modułu kontroli dostępu	0x00 – moduł wyłączony 0x01 – moduł załączony

6.11.3 ZAPIS KONFIGURACJI AUTOMATU

Rozkaz ten konfiguruje sposób pracy automatu odczytującego unikalny numer transpondera UID.

Opisywany czytnik daje możliwość chwilowego zawieszania pracy automatu w przypadku wystąpienia poprawnej transmisji na łączu RS.

Jeżeli czytnik będzie pracował w trybie mieszanym, tzn.
- uruchomiony jest automat odczytów UID, oraz:
- urządzenie nadrzędne (komputer, sterownik) komunikuje się z czytnikiem albo za pomocą czytnika z transponderami to:
konieczne jest odpowiednie skonfigurowanie czytnika tak aby w przypadku transmisji z czytnikiem lub z transponderem automat odczytów zawieszał swoją pracę.

Ramka rozkazu:

Nagłówek	C_SetAutoReaderConfig	ATrig_Period, AOfflineTime, Aserial, AMode, Abuzz, AMulti	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_SetAutoReaderConfig **0x58**	Zapis konfiguracji automatu	0x58
ATrig_Period	4 młodsze bity: Definiuje kiedy automat odczytów UID ma pracować, 4 starsze bity: ustala okres odczytów	Bity: 0-4 0-automat wyłączony na stałe 1-automat załączony na stałe 2=załącza się automatycznie gdy brak

<table>
<tr><td></td><td></td><td>transmisji na RS/USB przez czas dłuższy niż AOfflineTime
3= załącza się automatycznie gdy brak wywołań rozkazów komunikacji z transponderem przez czas dłuższy niż AOfflineTime

Bity:4-7 – liczba n, okres prób odczytów:
Dla n=0: domyślnie 250ms,
Dla n>0: okres = 2^{n-1} * 50ms</td></tr>
<tr><td>AOfflineTime</td><td>Czas braku transmisji na RS/USB
T= AofflineTime * [100 ms]
Brak transmisji może dotyczyć dowolnych rozkazów (Atrig=2), lub rozkazów komunikacji z transponderem (Atrig=3).

Rozkazy komunikacji z transponderem to:
C_TurnOnAntennaPower
C_Select
C_LoginWithDKB
C_LoginWithSKB)
C_ReadBlock
C_WriteBlock
C_CopyBlock
C_WritePage4B
C_ReadPage16B
C_IncrementValue
C_DecrementValue
C_WriteValue
C_ReadValue
C_Halt</td><td>0x00…0xff</td></tr>
<tr><td>ASerial</td><td>Automatyczne wysyłanie numeru transpondera UID po automatycznym odczytaniu go z transpondera</td><td>0-nigdy
1-tylko za pierwszym przyłożeniem transpondera
2-wysyła wszystkie</td></tr>
<tr><td>AMode</td><td>Wybór formatu wysyłanego numeru

MSB LSB
<table><tr><td>R</td><td>R</td><td>R</td><td>CR</td><td>M</td><td>E</td><td>I</td><td>A</td></tr></table></td><td><table><tr><td>R</td><td>Zarezerwowane, zawsze 0</td></tr><tr><td>CR=1</td><td>numer kończy się znakiem końca linii CR+LF</td></tr><tr><td>M=1</td><td>numer zaczyna się znakiem”M”</td></tr><tr><td>E=1</td><td>informacja rozszerzona o ilość kart w polu oraz typ</td></tr><tr><td>I=1</td><td>Numer w odwrotnej kolejności</td></tr><tr><td>A=1</td><td>Numer wysyłany w formacie ASCII</td></tr><tr><td>A=0</td><td>Numer wysyłany w formacie Nertonix</td></tr></table></td></tr>
<tr><td>ABuzz</td><td>Automatyczne sygnalizowanie odczytu za pomocą buzzera po automatycznym odczytaniu UID’u z transpondera.</td><td>0-nigdy
1-tylko za pierwszym przyłożeniem transpondera
2-sygnalizuje wszystkie</td></tr>
<tr><td>AMulti</td><td>Wybór skanowanych transponderów
MSB LSB
<table><tr><td>H2</td><td>HD</td><td>H1</td><td>U</td><td>S</td><td>I</td><td>B</td><td>M</td></tr></table></td><td>M-transpondery z rodziny Mifare/NFC
B- ISO14443B
I – IClass (CSN)
S – I-CODE (ISO15693)
U – Unique/EM4102/Q5
H1 – Hitag-1/S
HD – HID Prox II
H2 – Hitag-2</td></tr>
</table>

Ramka odpowiedzi:

nagłówek	C_ SetAutoReaderConfig +1		KodOperacji	CRC

6.11.4 ODCZYT KONFIGURACJI AUTOMATU

Ramka rozkazu:

nagłówek	C_ GetAutoReaderConfig		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_GetAutoReaderConfig	Odczyt konfiguracji automatu	0x5a

Ramka odpowiedzi:

nagłówek	C_ GetAutoReaderConfig +1	ATrig, AOfflineTime, ASerial, AMode, Abuzz, AMulti	KodOperacji	CRC

Gdzie:
Znaczenie parametrów odpowiedzi jest identyczne jak opisane wcześniej.

6.11.5 USTAWIENIE DATY I CZASU

Poniższe ustawienia nie mają wpływu na pracę czytnika.

Ramka rozkazu:

nagłówek	C_SetRtc	Year, Month, Day, Hour, Minute, Second	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_SetRtc	Ustawienie daty i czasu	0xb8
Year	rok	0…99
Month	miesiąc	1…12
Day	dzień miesiąca	1…31
Hour	godzina	0…23
Minute	minuta	0…59
Second	sekunda	0…59

Ramka odpowiedzi:

nagłówek	C_SetRtc +1		KodOperacji	CRC

6.11.6 ODCZYTANIE DATY I CZASU

Ramka rozkazu:

nagłówek	C_GetRtc		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_GetRtc	Odczytanie daty i czasu	0xb6

Ramka odpowiedzi:

nagłówek	C_GetRtc+1	Year, Month, Day, Hour, Minute, Second	KodOperacji	CRC

Gdzie:
Znaczenie parametrów odpowiedzi jest identyczne jak opisane wcześniej.

6.12 KONFIGURACJA INTERFACE'U SZEREGOWEGO RS-232/485

6.12.1 ZAPIS KONFIGURACJI INTERFEJSU SZEREGOWEGO

Rozkaz:

nagłówek	C_SetInterfaceConfig	Mode, Adr, Baudrate	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_SetInterfaceConfig	zapis konfiguracji interfejsu szeregowego	0x54
Mode		0x01
Adr	Adres na magistrali RS-485	0x01…0xfe
Bodrate	Prędkość danych na magistrali RS-232/485	0x01=2400 b/s 0x02=4800 b/s 0x03=9600 b/s 0x04=19200 b/s 0x05=38400 b/s 0x06=57600 b/s 0x07=115200 b/s

Odpowiedź:

	C_SetInterfaceConfig +1		KodOperacji	CRC

6.12.2 ODCZYT KONFIGURACJI INTERFEJSU SZEREGOWEGO

Rozkaz:

	C_ GetInterfaceConfig		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_GetInterfaceConfig	odczyt konfiguracji interfejsu szeregowego	0x56

Odpowiedź:

	C_ GetInterfaceConfig +1	Mode, Adr, Baudrate	KodOperacji	CRC

Gdzie:
Znaczenie parametrów odpowiedzi jest identyczne jak opisane wcześniej.
Zarządzenie zdarzeniami
Czytniki z serii CTU-S5R posiadają pamięć zdarzeń o pojemności 3400 rekordów. Źródłem zdarzenia może być operacja związana z kartą lub zmiana stanu na wejściach czytnika. Czytniki nie posiadają zegara RTC z podtrzymaniem bateryjnym. Po zaniku napięcia zasilania, zegar ustawiany jest na domyślą wartość: 1 styczeń 2000r., godz 00:00:00. Licznik zdarzeń zostaje wyzerowany.

6.12.3 KONFIGURACJA REJESTRATORA ZDARZEŃ

Ramka rozkazu:

nagłówek	C_SetEventTrig	CardTrig, In4Trig, In3Trig, In2Trig, In1Trig	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_SetEventTrig	Konfiguracja maskowania zdarzeń	0x7C
CardTrig	Maskowanie zdarzeń związanych z kartą (patrz niżej)	0x00 - 0xFF
In1Trig-In4Trig	Maskowanie zdarzeń związanych z wejściami (patrz niżej)	0x00-0xFF

Ramka odpowiedzi:

nagłówek	C_SetEventTrig+1		KodOperacji	CRC

- Bajt maskowania zdarzeń pochodzących „od karty"

Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
Reserve	Brak pamięci	Usunięcie karty	Dodanie karty	Reserve	Karta Master	Karta z poza bazy	Karta z bazy

Np. bajt 0x25(00100101) oznacza, że zapisywane będą zdarzenia gdy:

- odczytana zostanie karta występująca w bazie,
- odczytana zostanie karta zapisana jako master,
- nastąpiło usunięcie karty z bazy

- Bajty maskowania zdarzeń pochodzących od zmiany stanu na wejściach

Bajt	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
In1Trig	IO[3]F	IO[3]R	IO[2]F	IO[2]R	IO[1]F	IO[1]R	IO[0]F	IO[0]R
In2Trig	IO[7]F	IO[7]R	IO[6]F	IO[6]R	IO[5]F	IO[5]R	IO[4]F	IO[4]R
In3Trig	IO[11]F	IO[11]R	IO[10]F	IO[10]R	IO[9]F	IO[9]R	IO[8]F	IO[8]R
In4Trig	IO[15]F	IO[15]R	IO[14]F	IO[14]R	IO[13]F	IO[13]R	IO[12]F	IO[12]R

Bity IO[n]R oznaczają reakcję na zbocze rosnące wejścia **n,**
Bity IO[n]F oznaczają reakcję na zbocze opadające wejścia **n**

Np. ciąg bajtów konfiguracyjnych In4Trig-In1Trig: **0x00,0x31,0x40,0x08**, spowoduje, że zdarzenia będą zapisywały się w przypadku:

- Pojawienia się dowolnej zmiany stanu na wejściu o indeksie 10
- Pojawienia się zbocza rosnącego na wejściu o indeksie 8
- Pojawienia się zbocza rosnącego na wejściu o indeksie 7
- Pojawienia się zbocza opadającego na wejściu o indeksie 1

Podczas konfigurowania wyzwoleń zdarzeń należy stwierdzić, które z portów są skonfigurowane jako wejścia. Nie powinno się konfigurować zdarzeń dla tych IO, które są wyjściami.

Dla zagwarantowania poprawności zapisu zdarzenia, czas pomiędzy kolejnymi wyzwoleniami musi wynosić >20ms.

6.12.4 ODCZYT KONFIGURACJI REJESTRATORA ZDARZEŃ

Ramka rozkazu:

nagłówek	C_GetEventTrig		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_GetEventTrig	Odczyt konfiguracji rejestratora zdarzeń	0x7E

Ramka odpowiedzi:

nagłówek	C_GetEventTrig+1	CardTrig, In4Trig, In3Trig, In2Trig, In1Trig	KodOperacji	CRC

Bajty odpowiedzi (CardTrig, In4Trig, In3Trig, In2Trig, In1Trig) odpowiadają, bajtom z punktu 10.1.

6.12.5 ODCZYT LICZNIKÓW ZWIĄZANYCH Z PAMIĘCIĄ ZDARZEŃ

Ramka rozkazu:

nagłówek	C_GetEventParam		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_GetEventParam	Odczyt konfiguracji rejestratora zdarzeń	0x78

Ramka odpowiedzi:

nagłówek	C_GetEventParam+1	CapL, CapH, PointerL, PointerH, TotB3,TotB2,TotB1,TotB0	KodOperacji	CRC

CapH:CapL – dwu bajtowa wartość określająca pojemność pamięci zdarzeń
PointerH:PointerL – dwu bajtowa wartość będąca wskaźnikiem do pierwszego wolnego zdarzenia
TotB3:TotB2:TotB1:TotB0 – cztero bajtowa wartość określająca ilość zdarzeń zarejestrowanych od momentu wyzerowania licznika. TotB3 jest najmłodszym bajtem.
Zdarzenia zapisują się kolejno od indeksu 0 do indeksu Cap-1. W momencie gdy zapełniona zostanie pamięć, licznik „przekręca" się i nadpisywane zostają najstarsze wpisy.

Przykład:
Jeśli komendą C_GetEventParam odczytaliśmy, że pojemność pamięci zdarzeń wynosi 4400 wpisów, całkowita liczba zapisanych zdarzeń wynosi 5678, chcąc np. odczytać zdarzenie nr 5660, indeks interesującego nas zdarzenia będzie wynosił 5660-4400-1=1259.
Jeśli chcemy odczytać ostatnie zdarzenie, możemy skorzystać z wartości Pointer. Indeksem ostatniego zdarzenia będzie Pointer-1

6.12.6 ODCZYT ZDARZEŃ

Ramka rozkazu:

nagłówek	C_GetEvent	EvNoL, EvNoH	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_GetEvent	Odczyt zdarzenia	0x7a
EvNoL,EvNoH	Młodzy i starszy bajt indeksu zdarzenia	

Ramka odpowiedzi:

nagłówek	C_GetEvent+1	RR,MM,DD,gg,mm,ss,typ,B1,B2,B3,B4,B5	KodOperacji	CRC

RR,MM,DD – rok, miesiąc, dzień zdarzenia
gg,mm,ss - godzina, minuta, sekunda zdarzenia
typ - typ zdarzenia:

W zależności od wartości 8mego bitu bajtu „typ", wyróżnia się 2 przyporządkowania:

Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1
0 - karta	Brak pamięci	Usunięta	Dodana	reserved	Master	Z poza bazy	Z bazy
1 -wejścia	reserved	reserved	reserved	N4	N2	N1	N0

N4:N0 – numer wejścia w postaci binarnej, od którego pochodziło wyzwolenie zdarzenia.

- Jeśli dane zdarzenie pochodziło od karty, bajty B1-B5 zawierają numer ID karty.

B1	B2	B3	B4	B5
UID1	UID2	UID3	UID4	UID5 (Unique)

- Jeśli zdarzenie pochodzi od zmiany wejścia, bajty B1-B5 zawierają informacje o stanie wejść w formacie:

B1				B2				B3				B4				B5
IO0	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	IO9	IO10	IO11	IO12	IO13	IO14	IO15	Res

6.13 ROZKAZY POZOSTAŁE

6.13.1 ZMIANA GŁOŚNOŚCI BUZZERA

Rozkaz ten powoduje zmianę głośności sygnału dźwiękowego. Wprowadzona wartość będzie zapisana w nieulotnej pamięci EEPROM.

Ramka rozkazu:

nagłówek	C_BuzzerConfig		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_BuzzerConfig	Zmiana poziomu głośności buzzera	0x00-0x0F

Ramka odpowiedzi:

nagłówek	C_BuzzerConfig +1		KodOperacji	CRC

6.13.2 ZDALNY RESET CZYTNIKA

Ramka rozkazu:

nagłówek	C_Reset		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_Reset	Zdalny reset czytnika	0xd0

Ramka odpowiedzi:

nagłówek	C_Reset +1		KodOperacji	CRC

6.13.3 TRYB SLEEP

Wersja CTU-S5L może pracować w trybie uśpienia pobierając niski prąd. Funkcjonalność ta jest domyślnie wyłączona i należy ją jednorazowo włączyć komendą C_SetSleepFeature. Po jej włączeniu, wyprowadzenie IO2 staje się wejściem sterującym dla trybu uśpienia. Stan niski na IO2 wprowadza moduł w niski pobór prądu. W celu uzyskania minimalnego poboru prądu, port IO1 oraz wejścia komunikacyjne (RX/CS/SCK) powinny być spolaryzowane do masy lub +5V.

Ramka rozkazu:

nagłówek	C_SetSleepFeature	enable{0,1}	CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_SetSleepFeature	Włączenie funkcjonalności SLEEP	0x5C
enable	0-wyłączenie 1-włączenie	0,1

Ramka odpowiedzi:

nagłówek	C_SetSleepFeature +1		KodOperacji	CRC

6.13.4 ODCZYT WERSJI OPROGRAMOWANIA CZYTNIKA

Ramka rozkazu:

nagłówek	C_FirmwareVersion		CRC

Gdzie:

Nazwa parametru	Opis parametru	Zakres wartości
C_FirmwareVersion	Odczyt wersji oprogramowania czytnika	0xfe

Ramka odpowiedzi:

nagłówek	C_FirmwareVersion+1	Data1…..n	KodOperacji	CRC

Gdzie

Data1…n jest ciągiem znaków zapisanych w postaci kodów ASCII.

6.14 ZNACZENIE KODÓW OPERACJI W RAMKACH ODPOWIEDZI

Nazwa kodu operacji	Opis	Wartość
OC_Error	Błąd	0x00
OC_ParityError	Błąd parzystości	0x01
OC_RangeError	Błąd zakresu parametru	0x02
OC_LengthError	Błąd ilości danych	0x03
OC_ParameterError	Błąd parametru	0x04
OC_Busy	Chwilowa zajętość wewnętrznych modułów	0x05
OC_NoACKFromSlave	Brak wewnętrznej komunikacji	0x22
OC_CommandUnknown	Nieznana komenda	0x07
OC_WrongPassword	Złe hasło lub ostatnie hasło uległo przeterminowaniu czyli miał miejsce automatyczny LogOut.	0x09
OC_NoCard	Brak transpondera	0x0a
OC_BadFormat	Zły format danych.	0x18
OC_FrameError	Błąd transmisji. Może on świadczyć o istniejących zakłóceniach.	0x19
OC_NoAnswer	Brak odpowiedzi z transpondera	0x1E
OC_TimeOut	Przekroczony czas operacji. Może on świadczyć o braku transpondera w polu czytnika	0x16
OC_Successful	Operacja zakończona poprawnie	0xff
Kody operacji związane z transponderami DESFIRE		
OC_DesNoChanges	Operacja Commit nie przyniosła zmian	0x0c
OC_DesOutOfEEprom	Brak pamięci eeprom	0x0e
OC_DesIllegalCommand	Niedozwolona komenda	0x1c
OC_DesIntegrityError	Błąd CRC/ transmisji z kartą	0x1e
OC_DesNoSuchKey	Nieprawidłowy numer klucza	0x40
OC_DesLengthError	Nieprawidłowa długość komendy	0x7e
OC_DesPermisionDenied	Brak uprawnień do wykonania danej operacji	0x9d
OC_DesParameterError	Błąd parametru komendy	0x9e
OC_DesApplNotFound	Brak aplikacji o wybranych Aid	0xa0
OC_DesApplIntegrError	Błąd aplikacji, aplikacja zostaje zablokowana	0xa1
OC_DesAuthError	Błąd autoryzacji / niepoprawny klucz	0xae
OC_DesBoundaryError	Zapis/odczyt z rekordu wykroczył poza wielkość	0xbe
OC_DesPICCIntegError	Wewnętrzny błąd transpondera, zostaje zablokowany	0xc1
OC_DesCountError	Przekroczony limit 28 aplikacji	0xce
OC_DesDuplicateError	Aplikacja/Plik o tym identyfikatorze już istnieje	0xde
OC_DesEepromError	Błąd podczas zapisu/odczytu do pamięci EEPROM	0xee
OC_DesFileNotFound	Plik o tym identyfikatorze nie istnieje	0xf0
OC_DesFileIntegrError	Nieodwracalny błąd pliku, plik zostaje zablokowany	0xf1

7 PROTOKÓŁ MODBUS RTU

UWAGA:
Przy korzystaniu z protokołu MODBUS należy zmienić konfigurację urządzenia tak, aby samo nie wysyłało odczytanego ID. W tym celu należy do rejestru o adresie 1022 (ASerial) zapisać wartość 0.

7.1 OBSŁUGIWANE FUNKCJE PROTOKOŁU MODBUS

Funkcja	Opis
0x01	Read Coils
0x02	Read Discrete Inputs
0x03	Read Holding Regs
0x04	Read Input Regs
0x05	Write Single Coil
0x06	Write Single Reg
0x10	Write Multiple Regs

7.2 ADRESY MODBUS

7.2.1 ADRESY DO ODCZYTU ID KARTY

Adres	Typ	R/W	Opis
995	Holding Reg	R/W	Czas od ostatniego odczytu, po jakim ID zapisane w rejestrach 1000…1007 zostanie wyzerowane (x100ms)
996	Holding Reg	R/W	Rejestr jest ustawiany na wartość 1, gdy zostanie odczytany nowy transponder
997	Holding Reg	R	B<15:8> - Typ transpondera B<7:0> - Ilość kolizji
998	Holding Reg	R	Długość ID
999	Holding Reg	R	Licznik czasu od ostatniego odczytu (x100ms)
1000	Holding Reg	R	Kod transpondera [0]
1001	Holding Reg	R	Kod transpondera [1]
1002	Holding Reg	R	Kod transpondera [2]
1003	Holding Reg	R	Kod transpondera [3]
1004	Holding Reg	R	Kod transpondera [4]
1005	Holding Reg	R	Kod transpondera [5]
1006	Holding Reg	R	Kod transpondera [6]
1007	Holding Reg	R	Kod transpondera [7]

7.2.2 ADRESY DO ODCZYTU/ZAPISU KONFIGURACJI AUTOREADERA

Adres	Typ	R/W	Opis
1020	Holding Reg	R/W	ATrig
1021	Holding Reg	R/W	AOfflineTimer
1022	Holding Reg	R/W	ASerial
1023	Holding Reg	R/W	AMode
1024	Holding Reg	R/W	ABuzz
1025	Holding Reg	R/W	AMulti

Znaczenie rejestrów takie same jak dla parametrów komendy C_SetAutoreaderConfig

ADRESY DO KONFIGURACJI GPIO1

Adres	Typ	R/W	Opis
1100	Holding Reg	R/W	Dir – określa kierunek portu, Dir=0 – Wyjście Dir=1 - Wejście

1101	Holding Reg	R/W	Dla **Dir=0** (wyjście) **NormalOpen**	Dla **Dir=1** (wejście) **Triger** – wartość określająca sposób wyzwalania wejścia 0x0000 – niezanegowane 0x0001 – zanegowane 0x0002 – reakcja na zbocze narastające 0x0003 – reakcja na zbocze opadające
1102	Holding Reg	R/W	Dla **Dir=0** (wyjście) **ChangeState**	Dla **Dir=1** (wejście) **TypeOfInput** – typ wejścia 0x0000 – „0" na stałe 0x0001 – „1" na stałe 0x0002 – wejście standardowe 0x0003 – wejście licznikowe
1103	Holding Reg	R/W	Dla **Dir=0** (wyjście) **OtpMode**	Dla **Dir=1** (wejście) Opoznienie
1104	Holding Reg	R/W	Dla **Dir=0** (wyjście) TypeOfOutput –źródło sygnału sterującego 0x0000 – wyłączone na stałe 0x0001 – załączone na stałe 0x0002 – sterowane poprzez interfejs szeregowy, automatycznie powracające do zera 0x0003 – sterowane poprzez interf szeregowy 0x0004 – ustawiane w przypadku przyłożenia do czytnika karty, znajdującej się w wewnętrznej bazie. 0x05 – ustawiane w przypadku przyłożenia do czytnika dowolnej karty	
1105	Holding Reg	R/W	Dla **Dir=0** (wyjście) **Time** – długość generowanego przebiegu (x100ms). Zakres wartości: 0-255	
1106	Holding Reg	R/W	Dla **Dir=0** (wyjście) **Time0** – czas logicznego zera (x100ms) Zakres wartości: 0-255.	
1107	Holding Reg	R/W	Dla **Dir=0** (wyjście) **Time1** – czas logicznej jedynki (x100ms) Zakres wartości: 0-255.	

7.2.3 ADRESY DO KONFIGURACJI GPIO2

Adres	Typ	R/W	Opis	
1110	Holding Reg	R/W	Dir – określa kierunek portu, Dir=0 – Wyjście Dir=1 - Wejście	
1111	Holding Reg	R/W		Dla **Dir=1** (wejście) **Triger** – wartość określająca sposób wyzwalania wejścia 0x0000 – niezanegowane 0x0001 – zanegowane 0x0002 – reakcja na zbocze narastające 0x0003 – reakcja na zbocze opadające
1112	Holding Reg	R/W		Dla **Dir=1** (wejście) **TypeOfInput** – typ wejścia 0x0000 – „0" na stałe 0x0001 – „1" na stałe 0x0002 – wejście standardowe 0x0003 – wejście licznikowe
1113	Holding Reg	R/W		Dla **Dir=1** (wejście) Opoznienie
1114	Holding Reg	R/W	Dla **Dir=0** (wyjście) TypeOfOutput –źródło sygnału sterującego	

			0x0000 – wyłączone na stałe 0x0001 – załączone na stałe 0x0002 – sterowane poprzez interfejs szeregowy, automatycznie powracające do zera 0x0003 – sterowane poprzez interf szeregowy 0x0004 – ustawiane w przypadku przyłożenia do czytnika karty, znajdującej się w wewnętrznej bazie. 0x0005 – ustawiane w przypadku przyłożenia do czytnika dowolnej karty
1115	Holding Reg	R/W	Dla **Dir=0** (wyjście) **Time** – długość generowanego przebiegu (x100ms). Zakres wartości: 0-255
1116	Holding Reg	R/W	Dla **Dir=0** (wyjście) **Time0** – czas logicznego zera (x100ms) Zakres wartości: 0-255.
1117	Holding Reg	R/W	Dla **Dir=0** (wyjście) **Time1** – czas logicznej jedynki (x100ms) Zakres wartości: 0-255.

7.2.4 ADRESY DO KONFIGURACJI PRZEKAŹNIKA

Adres	Typ	R/W	Opis
1120	Holding Reg	R/W	Dir – określa kierunek portu, Dir=0 – Wyjście
1121	Holding Reg	R/W	
1122	Holding Reg	R/W	
1123	Holding Reg	R/W	
1124	Holding Reg	R/W	TypeOfOutput –źródło sygnału sterującego 0x0000 – wyłączone na stałe 0x0001 – załączone na stałe 0x0002 – sterowane poprzez interfejs szeregowy, automatycznie powracające do zera 0x0003 – sterowane poprzez interf szeregowy 0x0004 – ustawiane w przypadku przyłożenia do czytnika karty, znajdującej się w wewnętrznej bazie. 0x0005 – ustawiane w przypadku przyłożenia do czytnika dowolnej karty
1125	Holding Reg	R/W	**Time** – długość generowanego przebiegu (x100ms). Zakres wartości: 0-255
1126	Holding Reg	R/W	**Time0** – czas logicznego zera (x100ms) Zakres wartości: 0-255.
1127	Holding Reg	R/W	**Time1** – czas logicznej jedynki (x100ms) Zakres wartości: 0-255.

7.2.5 ADRESY DO KONFIGURACJI BUZZERA

Adres	Typ	R/W	Opis
1130	Holding Reg	R/W	Dir – określa kierunek portu, Dir=0 – Wyjście
1131	Holding Reg	R/W	
1132	Holding Reg	R/W	
1133	Holding Reg	R/W	
1134	Holding Reg	R/W	TypeOfOutput –źródło sygnału sterującego 0x0000 – wyłączone na stałe 0x0001 – załączone na stałe 0x0002 – sterowane poprzez interfejs szeregowy, automatycznie powracające do zera 0x0003 – sterowane poprzez interf szeregowy 0x0004 – ustawiane w przypadku przyłożenia do czytnika karty, znajdującej się w wewnętrznej bazie. 0x0005 – ustawiane w przypadku przyłożenia do czytnika dowolnej karty

1135	Holding Reg	R/W	**Time** – długość generowanego przebiegu (x100ms). Zakres wartości: 0-255
1136	Holding Reg	R/W	**Time0** – czas logicznego zera (x100ms) Zakres wartości: 0-255.
1137	Holding Reg	R/W	**Time1** – czas logicznej jedynki (x100ms) Zakres wartości: 0-255.

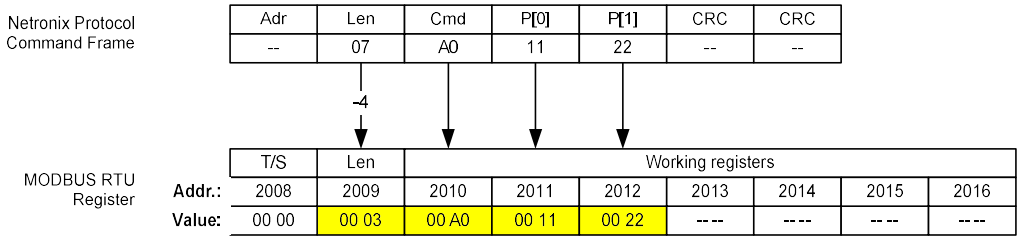
7.3 ENKAPSULACJA PROTOKOŁU NETRONIX W PROTOKOLE MODBUS RTU

Dowolną komendę z protokołu Netronix można wykonać wykorzystując odpowiednie rejestry z protokołu MODBUS.

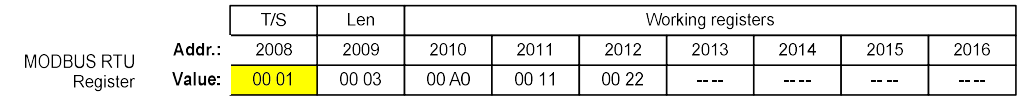
Adres	Typ	R/W	Nazwa	Opis
2008	Holding Reg	R/W	Trigger/Status	Rejestr ten służy do wyzwolenia przetwarzania komendy oraz do sprawdzania statusu przetwarzania. Dozwolone wartości: 0x0000 – Moduł w trybie Idle 0x0001 – Wyzwolenie przetwarzania 0x00EE – Błąd 0x00FF – Komenda wykonana. Odpowiedz znajduje się w rejestrach roboczych.
2009	Holding Reg	R/W	Len	Rejestr ten zawiera długość zapisanej komendy / długość odpowiedzi (ilość rejestrów zapisanych/do odczytania)
2010-2073	Holding Reg	R/W	Working registers	Rejestry te służą do zapisania komendy / odczytania odpowiedzi. Jeden rejestr przechowuje wartość 1 bajta komendy/odpowiedzi z ramki Netronix

7.3.1 SCHEMAT POSTĘPOWANIA

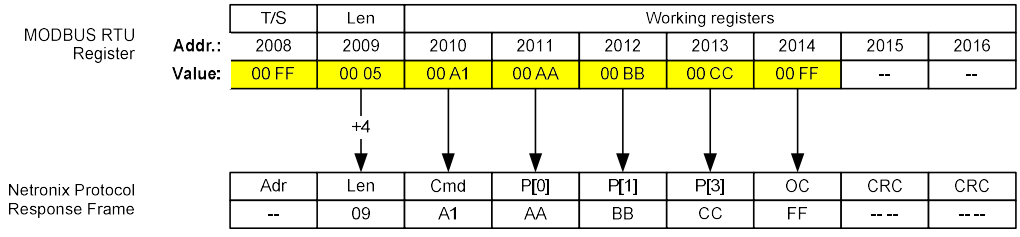
1. Zapisać do rejestrów MODBUS komendę z protokołu Netronix według poniższego schematu:



2. Zapisać do rejestru Trigger/Status wartość 0x0001.



3. Odczytywać rejestr Trigger/Status, aż pojawi się w nim wartość 0x00FF. Wartość 0x00FF oznacza, że odpowiedz jest już gotowa i można ją odczytać z rejestrów MODBUS.



7.3.2 PRZYKŁAD UŻYCIA – ODCZYT WERSJI OPROGRAMOWANIA.

Założenia:
Adres logiczny czytnika (w protokole Netronix/Modbus RTU) – 0x01.

1. Należy określić jak powinna wyglądać ramka w protokole Netronix. Istotne jest pole Cmd oraz ewentualne parametry. Cała ramka, dla komendy odczytu wersji firmware wygląda następująco:

Adr	Len	Cmd	CRC	
0x01	0x05	0xFE	0xC6	0x14

Cmd = 0xFE
Param – brak

2. Do rejestru ***Len*** należy wpisać ilość danych, a do rejestrów ***Working registers*** należy wpisać kod komendy (i ewentualne parametry):
   RTU Tx > 01 10 07D8 0002 04 0001 00FE 0925
   RTU Rx > 01 10 07D8 0002 C087

   Len = 0001
   WorkingRegister[0] = 00FE

3. Do rejestru ***Trigger/Status*** należy wpisać wartość 0x0001. Spowoduje to wykonanie komendy zapisanej w rejestrach ***Working Registers***.
   RTU Tx > 01 06 07D7 0001 F946
   RTU Rx > 01 06 07D7 0001 F946

   Trigger/Status=0001

4. Następnie należy odczytywać rejestr ***Trigger/Status***, aż do momentu odczytania wartości 0x00FF. Wartość 0x00FF oznacza że komenda została wykonana i w rejestrach ***Working Registers*** znajduje się odpowiedz.
   RTU Tx > 01 03 07D7 0001 3546
   RTU Rx > 01 03 02 00FF F804

   Trigger/Status=00FF

5. Następnie należy odczytać wartość rejestru ***Len***. Rejestr ten zawiera informację na temat ilości rejestrów, w których zapisana jest odpowiedź
   RTU Tx > 01 03 07D8 0001 0545
   RTU Rx > 01 03 02 0011 7848

   Len=0011

6. W ostatnim kroku, należy odczytać Len pierwszych rejestrów roboczych.
   RTU Tx > 01 03 07D9 0011 5549
   RTU Rx > 01 03 22
   00FF
   004D 0057 002D 0052 0037 002D 0056 0033 002E 0032 002E 0041 0031 002E 0035
   00FF
   8E C6

   00FF – Kod komendy+1 (odpowiedź na C_FirmwareVersion)
   004D 0057 002D … 0031 002E 0035 – zwrócone wersja firmware – „MW-R7-v3.2.A.1.5”
   00FF – Kod operacji (sukces)

8 MECHANIZM MASTERID

Mechanizm ten polega na istnieniu możliwości szybkiego dodawania/usuwania kart użytkowników do/z pamięci czytnika za pomocą „karty master".

Jeżeli chcemy zarejestrować kartę jako „kartę master" to należy najpierw dokonać wyczyszczenia pamięci kart za pomocą powrotu do ustawień fabrycznych. Po wyczyszczeniu pamięci należy w dowolnym momencie przyłożyć do modułu wybraną kartę. Karta ta automatycznie staje się „kartą master". Karty master nie można usunąć ani dodać za pomocą innej karty.

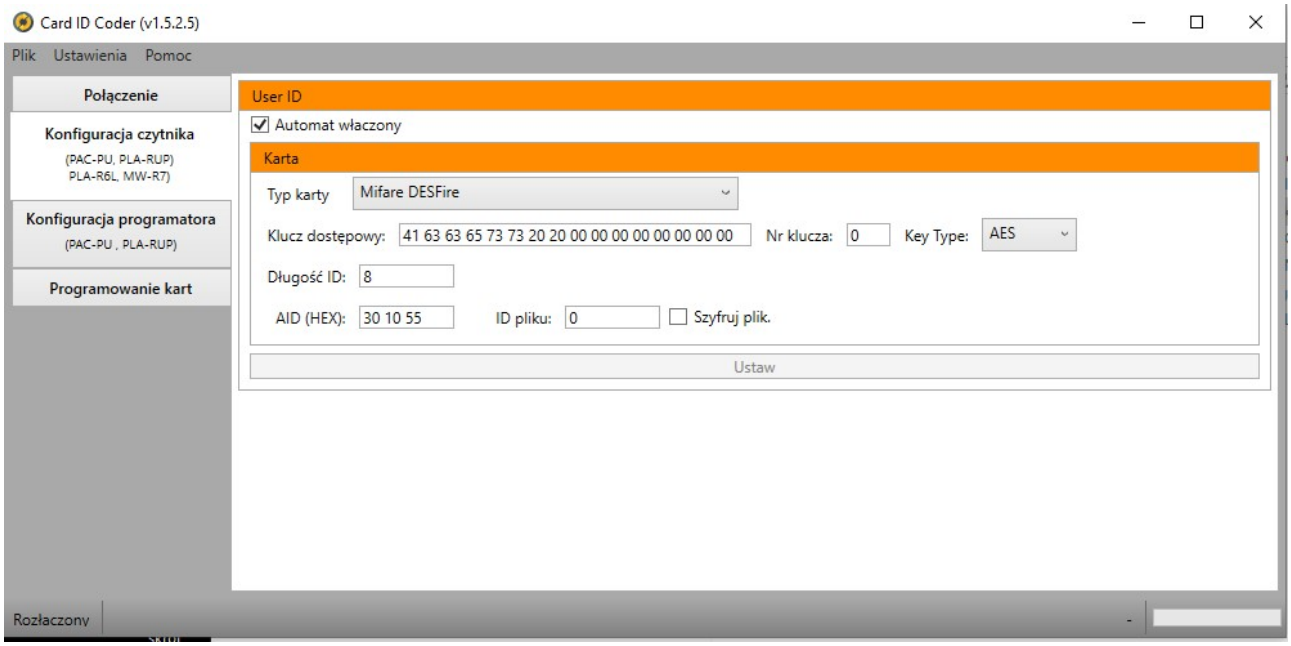
Jeżeli chcemy zarejestrować kartę jako „kartę użytkownika" to należy najpierw przyłożyć do czytnika „kartę master" a następnie, w ciągu ok. 5 sekund, przyłożyć rejestrowaną kartę.

Jeżeli chcemy usunąć z pamięci „kartę użytkownika" to należy najpierw przyłożyć do czytnika „kartę master" a następnie, w ciągu ok. 5 sekund, przyłożyć usuwaną kartę.

Po przyłożeniu do czytnika „karty użytkownika" czytnik uruchamia wyjście elektryczne zaprogramowane jako sterowane wewnętrznym mechanizmem kontroli dostępu.

9 MECHANIZM USERID

Mechanizm ten umożliwia odczyt zaszyfrowanych danych z sektorów/plików zaawansowanych transponderów Mifare Plus, Desfire, Classic.
Konfiguracja tego trybu odbywa się narzędziem CardIDCoder.



10 CZYSZCZENIE PAMIĘCI KART I POWRÓT DO USTAWIEŃ FABRYCZNYCH

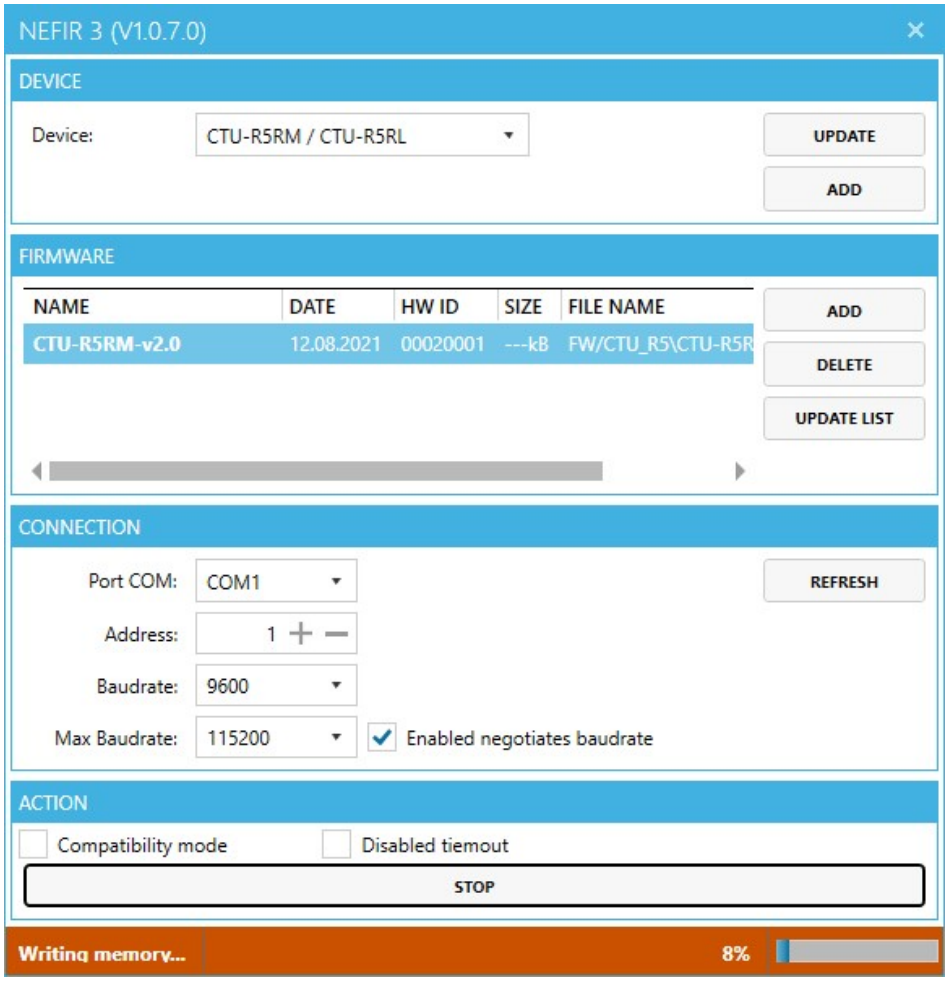
Aby powrócić do ustawień fabrycznych należy na czas ok. 8 sekund przycisnąć przycisk powrotu do ustawień fabrycznych. Podczas powrotu do ustawień fabrycznych ustawiane są na stałe następujące parametry czytnika:

Nazwa parametru lub funkcjonalność	**Wartość lub ustawienie**	
Interfejs		
Interfejs RS232	Adres: 0x01	
	Prędkość: 0x03	9600bps
Interfejs RS485	Adres: 0x01	
	Prędkość: 0x03	9600bps
Interfejs 1-Wire	Rodzina : 0x01	
	Adres : 0x00	
Interfejs Wiegand	Ilość bitów : 37	
Interfejs I^2C	Adres : 0xC0	
Odczyt transponderów		
AAutoreader	Trigger : 0x01	Zawsze włączony
	Timeout : 0x14	2s
	ASerial : 0x01	Za pierwszym przyłożeniem
	AMode : 0x04	Informacja rozszerzona o ilość kart w polu oraz typ karty
	ABuzzer : 0x01	Za pierwszym przyłożeniem
	AMulti : 0x11	MIFARE + EM4102
Karta Master	Brak karty Master w pamięci kart	
Cała wewnętrzna pamięć transponderów wraz z kartą Master	0xff 0xff 0xff 0xff 0xff - pamięć wyczyszczona	
Wejścia/Wyjścia		
Port 0 – GPIO1	Wejście dowolnego przeznaczenia	
Port 1 – GPIO2	Wejście dowolnego przeznaczenia	
Port 2 – przekaźnik	Załączenie elektrozamka	
Port 3 – buzzer	Sygnalizacja załączenia elektrozamka	
Głośność buzzera	0x05	
Zdarzenia		
Konfiguracja zdarzeń	Zapis zdarzeń nieaktywny	
Hasło		
Hasło dostępu	brak	
Funkcjonalność SLEEP	wyłączone	

11 BOOTLOADER – ZMIANA FIRMWARE URZĄDZENIA

W celu wgrania do urządzenia nowego firmware, należy postępować według poniższej procedury:

1. Podłączyć urządzenie do portu szeregowego RS232 w komputerze
2. Otworzyć program NEFIR3.exe
3. W sekcji DEVICE wybrać odpowiednie urządzenie. Jeżeli urządzenia nie ma na liście, to należy wcisnąć przycisk UPDATE.
4. W sekcji FIRMWARE należy wybrać wersję firmware, którą chcemy zaprogramować urządzenie. Wciśnięcie przycisku UPDATE LIST powoduje pobranie z serwera najnowszych wersji plików.
5. W sekcji CONNECTION należy:
    a. Wybrać właściwy port COM.
    b. Ustawić Baudrate na prędkość transmisji z jaką pracuje czytnik (standardowo 9600)
    c. Opcjonalnie zaznaczyć Enabled negotiates baudrate i w polu Max Baudrate wybrać prędkość 115200. Takie ustawienia powodują przyśpieszenie procesu przeładowywania.
6. W sekcji ACTION należy wcisnąć przycisk START. Po wciśnięciu przycisku rozpocznie się proces wgrywania nowego firmware do urządzenia.



Rysunek: Widok okna programu podczas przeładowywania firmware

12 PRZYKŁAD PRACY Z TRANSPONDEREMI

12.1 PRZYKŁAD PRACY Z TRANSPONDEREM S50,S70

Po poprawnym podłączeniu czytnika i nawiązaniu obustronnej komunikacji pomiędzy nim a komputerem nadrzędnym można przystąpić do operacji odczytu i zapisu pamięci transpondera.
Poniższe operacje zakładają, że czytnik posiada ustawienia fabryczne oraz, że użyta karta S50 posiada ustawienia fabryczne czyli pełne prawa dostępu i oba klucze 0xff ff ff ff ff ff.

Ponieważ podczas ręcznych prób czas pomiędzy kolejnymi rozkazami wysyłanymi po RS jest stosunkowo duży i osiąga od kilku sekund do kilku minut to należy wyłączyć wewnętrzny automat odczytów UID.
Należy to zrobić za pomocą rozkazu :
SetAutoReaderConfig z parametrami 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 .

Aby dokonać odczytu transpondera, najpierw należy załadować klucz do pamięci kluczy.
Załadujmy więc klucz do SKB za pomocą
C_LoadKeyToSKB, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00

Załączamy pole.
TurnOnAntennaPower, 0x01
Przykładamy transponder do czytnika

Selekcjonujemy transponder
C_Select, 0x00

Logujemy się np. do sektora 3.
C_LoginWithSKB, 0x03, 0xAA, 0x00

Odczytajmy zawartość 2-go bloku w 3-cim sektorze.
C_ReadBlock, 0x02

O ile wszystkie Kody Operacji w ramkach odpowiedzi były OC_Successful to otrzymane wartości są danymi odczytanymi z bloku.

12.2 PRZYKŁAD PRACY Z TRANSPONDERAMI DESFIRE

Po poprawnym podłączeniu czytnika i nawiązaniu obustronnej komunikacji pomiędzy nim a komputerem nadrzędnym można przystąpić do operacji odczytu i zapisu pamięci transpondera.
Poniższe operacje zakładają, że czytnik posiada ustawienia fabryczne oraz, że użyta karta Desfire posiada ustawienia fabryczne czyli pełne prawa dostępu, a klucz PICC Master key ma wartość 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00.
Wynikiem tego przykładu jest utworzenie nowej aplikacji, zmiana standardowego klucza aplikacji, utworzenie pliku z danymi, zapisanie a następnie odczyt danych z pliku.

Ponieważ podczas ręcznych prób czas pomiędzy kolejnymi rozkazami wysyłanymi po RS jest stosunkowo duży i osiąga od kilku sekund do kilku minut to należy wyłączyć wewnętrzny automat odczytów UID.
Należy to zrobić za pomocą rozkazu :

1. SetAutoReaderConfig 0x00, 0x00, 0x00, 0x00, 0x00, 0x00.

Aby dokonać odczytu transpondera, najpierw należy załadować klucze do pamięci kluczy.
Ładujemy więc standardowy klucz transponderów desfire na pozycję np.„3” pamięci czytnika, a na pozycję 4 ładujemy sobie własny klucz, który nadamy nowej aplikacji:

2. C_DesSaveKey 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00

3. C_DesSaveKey 0x04, 0x01, 0x02, 0x03, 0x04, 0x0a, 0x0b, 0x0c, 0x0d

Załączamy pole.

4. C_TurnOnAntennaPower 0x01

Przykładamy transponder do czytnika, Selekcjonujemy transponder

5. C_Select 0x00

Inicjujemy protokół wymiany danych ISO, z numerem logicznym transpondera 0

6. C_DesInitProtocol 0x00

Dokonujemy autoryzacji z użyciem klucza „0" czyli PICC Master key, klucz ten zapisany jest w pamięci czytnika pod indeksem „3"

7. C_DesAuth 0x00,0x03

Tworzymy aplikację o numerze identyfikacyjnym np. 0x30, 0x10, 0x55, o domyślnych ustawieniach klucza ApplicationMasterKey, z rezerwacją miejsca na 4 klucze

8. C_DesCreateApp 0x30,0x10,0x55,0x0F,0x04

Zmieniamy domyślny, nowo utworzony klucz ApplicationMasterKey na ten, który mamy zapisany w czytniku na pozycji 4. W związku z tym selekcjonujemy nową aplikację:

9. C_DesSelectApp 0x30,0x10,0x55

Logujemy się do aplikacji z użyciem klucza Application Master Key, a następnie zmieniamy go po czym ponownie logujemy z użyciem nowego klucza

10. C_DesAuth 0x00,0x03
11. C_DesChangeKey 0x00,0x04
12. C_DesAuth 0x00,0x04

Tworzymy standardowy plik z danymi, z pełnymi prawami dostępu dla Application Master Key, oraz prawami odczytu dla klucza „3". Plik będzie miał indeks „2", nieszyfrowaną wymianę danych oraz wielkość 1500 bajtów

13. C_DesCreateSTDataFile 0x02,0x00,0x30,0x00,0xDC,0x05,0x00

Dokonujemy teraz zapisu danych do utworzonego właśnie pliku od pozycji 0

14. C_DesWriteData 0x02,0x00,0x00,0x00, $TuSaNaszeDaneDoZapisu

Odczytujemy 21 bajtów właśnie zapisanych danych

15. C_DesReadData 0x02,0x00,0x00,0x00, 0x15,0x00,0x00