

PolCarReader – Czytnik Ethernet-owy RFID z PoE



Charakterystyka sterownika

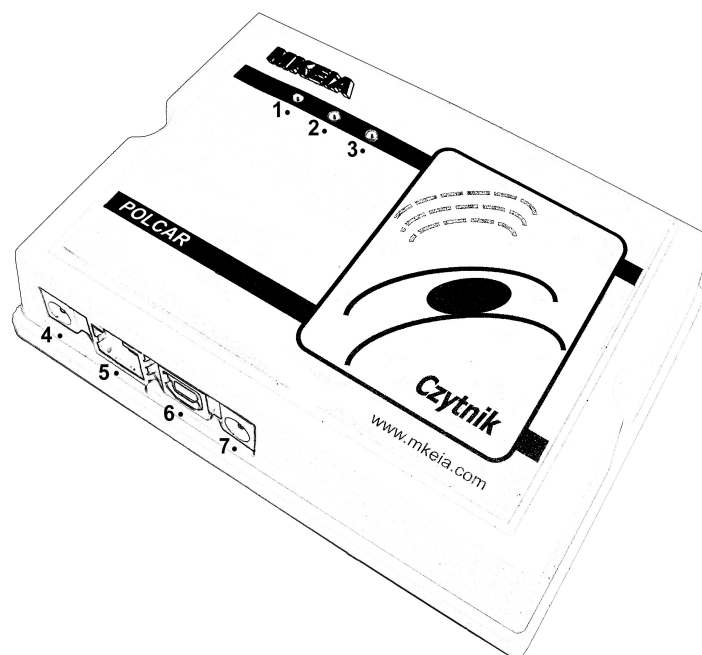
- Obsługa transponderów typu Unique EM-410x
- Komunikacja przez Ethernet
- Obsługa protokołu TCP/IP
- Obsługa protokołu ICMP
- Konfiguracja przez Telnet
- Prosta obsługa
- Zasilanie PoE lub 12V-24V 200mA (opcjonalnie)
- PoE 802.3af standard
- Temperatura pracy od 0°C do 40°C

PolCarReader jest układem czytnika RFID zdolnym do komunikacji za pomocą interfejsu sieciowego Ethernet lub USB. Urządzenie może czytać karty zgodne ze układem EM-410x Unique pracującym w paśmie 125kHz. Głównym zadaniem czytnika jest sterowanie dostępem do pomieszczeń (elektrozamkiem). PolCarReader po przyłożeniu karty-transpondera łączy się z serwerem pod wcześniej zaprogramowany adres IP oraz port. Aplikacja serwerowa (stworzona już samodzielnie przez kupującego) po połączeniu się sterownika sprawdza w bazie danych, czy kod karty odczytany przez odpowiedni czytnik pozwala na wejście do obiektu. W przypadku powodzenia jest wysyłana komenda z serwera do czytnika z żądaniem otwarcia zamka (na czas wcześniej zaprogramowany) i zasygnalizowaniem faktu diodami LED lub sygnalizatorem dźwiękowym. W przeciwnym przypadku aplikacja wysyła odpowiedni rozkaz sterujący diodami LED lub sygnalizatorem dźwiękowym, który informuje o braku dostępu do obiektu. Konfiguracja ustawień sterownika jak na przykład adres IP, maska, brama, IP serwera, port serwera, login, hasło, czas otwarcia zamka itp. jest realizowana w dwojaki sposób: przy pomocy połączenia autoryzowanego Telnet lub przez port USB (wirtualny port COM), wykorzystując dowolny terminal. W sterowniku został zaimplementowany protokół ICMP, który pozwala użytkownikowi na odpytanie kontrolne urządzenia. Dodatkowo czytnik PolCarReader może służyć jako standardowy czytnik USB - przy odłączonym kablu Ethernet-owym. W tym trybie pracy urządzenie wysyła kod karty na wirtualny port szeregowy z jednoczesnym wydaniem krótkiego sygnału dźwiękowego. Czytnik PolCarReader jest produkowany w dwóch wersjach tj. z zasilaniem bezpośrednio z sieci Ethernet przez PoE lub w wersji prostszej zasilany z zewnętrznego źródła 12V. Czytnik w wersji z PoE również może być zasilany ze źródła zewnętrznego. Obudowa urządzenia o wymiarach 128x94x28mm jest wykonana z plastiku w kolorze jasnoszarym i jest możliwy jej montaż na ścianie przy pomocy dwóch wkrętów. W wersji bez PoE złącza są wyprowadzone z boku obudowy, a w wersji z PoE od spodu obudowy.

Instrukcja obsługi czytnika PolCarReader

1. Sposób podłączenia i zasilania czytnika

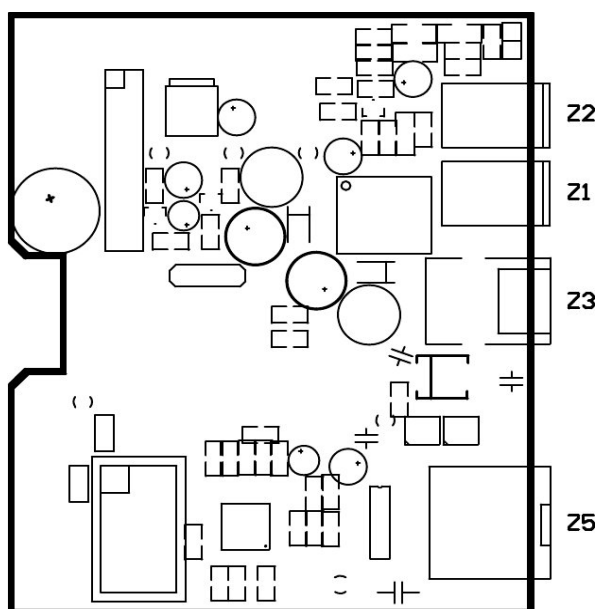
Czytnik jest wykonywany w dwóch wersjach – z zasilaniem z PoE i bez. W przypadku sterownika bez zasilania z PoE złącza są umieszczone z boku obudowy jak to jest pokazane na rysunku 1. Sterownik posiada cztery złącza. Licząc od lewej strony przy orientacji obudowy tak aby diody LED były w górnej jej części, pierwsze złącze to zasilanie 10V-24V (na bolcu +), drugie to złącze Ethernet-owe, trzecie USB, a czwarte do obsługi zamku (+ na bolcu, a masa na zewnątrz). Złącze do obsługi zamku normalnie otwarte – stan wysokiej impedancji, a podczas wysterowania podanie na + napięcia zasilania na bolec. Do złącza zasilania i wyjścia do sterowania elektrozamkiem należy użyć wtyków o wymiarach 5,5/2,1 mm.



Rysunek 1 - Wyprośadzenie złącz czytnika i sygnalizacja LED

1	Dioda LED żółta	5	Złącze Ethernet-owe
2	Dioda LED zielona	6	Złącze USB typu B
3	Dioda LED czerwona	7	Wyjście do sterowania elektrozamkiem
4	Zasilanie 10-24V		

W wersji z zasilaniem bezpośrednio po kablu komunikacyjnym PoE złącza są umieszczone wewnątrz obudowy z rozmieszczeniem przedstawionym na rysunku 2. Układ może być zasilony z zewnętrznego źródła w przypadku braku sieci z obsługą zasilania po PoE.



Rysunek 2 - Widok płytki czytelnika wewnątrz obudowy

Z1	Złącze zasilania 12V-24V (opcjonalnie)	Z3	Złącze USB typu B
Z2	Złącze wyjścia do sterowania elektrozamkiem	Z5	Złącze RJ45 Ethernetowe

2. Opis i działanie

2.1 Sposoby komunikacji

Komunikacja ze sterownikiem jest realizowana na trzy sposoby:

a) Komunikacja USB. Od strony PC port USB jest widziany jako wirtualny port COM (sterownik FTDI) pracujący z prędkością transmisji 115200b/s bez bitów parzystości i jeden bit stopu. Jest ona wykorzystywana do konfiguracji ustawień sterownika i wysyłania kodu transpondera (bezpośrednio wyrzucany kod po przyłożeniu karty) w okresie czasu nie częściej niż 1s.

Uwaga: Ustawianie parametrów sterownika przez USB jest możliwe po zwarciu zwory przy czerwonej diodzie LED na płycie PCB od strony złączy. Zworka dla nowego urządzenia jest normalnie zwarta. Pierwsza konfiguracja powinna być dokonana przez USB.

b) Komunikacja przez Ethernet – Telnet. Telnet służy podobnie jak komunikacja po USB do konfiguracji ustawień sterownika. Logowanie do Telnetu – login: **admin**, hasło: **1234**. Aby zalogować się po telnetcie wpisujemy komendę w wierszu poleceń *telnet adres_ip*.

c) Komunikacja przez Ethernet – Aplikacja Główna. Po przyłożeniu transpondera do sterownika, sterownik łączy się z serwerem pod wcześniej ustalony adres i port. Opis protokołu komunikacji Aplikacji głównej jest opisany dalej.

2.2 Uruchomienie i konfiguracja

Sterownik uruchamiamy podpinając go do źródła zasilania prądu stałego o napięciu 7V-24V i wydajności prądowej minimum 200mA (+ dodatkowa wydajność prądowa na elektrozamek) do pierwszego złącza od lewej (koło złącza Ethernet). Po załączeniu zasilania dioda LED powinny się zapalić na ułamek sekundy po czym zgasnąć. Jeśli urządzenie nie jest podpięte do sieci Ethernet dioda żółta miga.

Pierwszą konfigurację należy wykonać po USB (wirtualny port COM) wykorzystując dowolny terminal np. *putty* pracujący najlepiej w standardzie VT100. Dane konfiguracyjne są zapisywane w pamięci EEPROM, a niektóre ustawienia (ip,maska,brama) uaktualniane po restarcie urządzenia. Polecenia konfiguracyjne po połączeniu można wyświetlić wpisując ? lub polecenie *help*. Znaczenie poleceń konfiguracyjnych:

- *setipaddr IP_ADR* – ustawianie adresu IP sterownika
- *setnetmask IP_MASK* – ustawianie maski

- *setgateway IP_GATEWAY* – ustawianie bramy
- *setmacaddr MAC_ADR* – ustawianie adresu MAC
- *setlogin LOGIN* – ustawianie loginu do Telnetu
- *setpasswd PASSWD* – ustawianie hasła do logowania do Telnetu
- *setdelay DELAY* – ustawienie czasu włączenia zamka w ms
- *settings* – wyświetlenie ustawień z EEPROM (nie bieżących)
- *opengate* – komenda otwarcia zamka na wcześniej ustalony czas
- *setrfidserver IP_ADR* – adres IP serwera, z którym ma się łączyć sterownik
- *setrfidport PORT* – port, na który ma się łączyć z serwerem sterownik
- *setbuzzer BEEP* – ustawianie czasu załączenia sygnalizatora dźwiękowego w ms
- *reboot* – restart sterownika i aktualizacja ustawień
- *exit* – wyjście z ustawień przy połączeniu przez Telnet

2.3 Sygnalizacja podstawowa LED

Diody LED są obsługiwane od strony serwera wysyłając rozkazy po nawiązanym połączeniu. Nie mniej jednak przed nawiązaniem połączenia dioda żółta jest wykorzystana do sygnalizacji braku połączenia kablowego (link) – dioda żółta miga oraz dioda czerwona do sygnalizacji próby łączenia sterownika z serwerem po przyłożeniu karty.

2.4 Protokół aplikacji głównej

Wszystkie dane liczby są w formacie LITTLE-ENDIAN (tak jak ma PC oraz ARM). Aplikacja posługuje się bezpośrednio protokołem binarnym. Po przyłożeniu transpondera aplikacja łączy się z serwerem i od razu wysyła ramkę (16 bajtów) z odczytanym kodem transpondera w następującej postaci:

CMD, MAC, TIMESTAMP-UINT32, TAG, gdzie:

CMD – stała wartość 0x01 - Komenda nowy tag dostępny (1 bajt)

MAC – MAC adres jako identyfikator urządzenia (6 bajtów)

TIMESTAMP-UINT32 - (4 bajty) typ uint32_t (unsigned int) w formacie Little Endian - Znacznik czasowy odczytania transpondera, oznacza ile milisekund od włączenia urządzenia odczytano zadany tag

TAG - odczytany tag (5 bajtów)

Po wysłaniu ramki z kodem transpondera przez sterownik aplikacja gotowa jest do przyjmowania komend, po odczekaniu nie mniej niż 200ms od otrzymania ramki od sterownika, możemy wysyłać rozkazy. Po wysłaniu danego rozkazu musimy oczekiwać na dwubajtową odpowiedź z ustawionym najstarszym bitem *CMD* oraz kodem błędu. Do dyspozycji mamy następujące rozkazy:

CMD:

0x02 - otwórz zamek

0x03 - steruj diodami LED

0x04 - NOP komenda pusta w celu możliwości podtrzymywania połączenia

0x05 - Rozkaz rozłączenia

a) Otwórz zamek(0x02)

Komenda powoduje otwarcie zamka na wcześniej ustalony czas w konfiguracji

Komenda ma postać (2 bajty): 0x02, 0x00

Odpowiedź (2 bajty) 0x82, *ERROR_CODE*

b) Steruj diodami LED i sygnalizatorem dźwiękowym(0x03)

Komenda ma postać (14 bajtów):

CMD, LED-MASK-UINT32, TYPE, TIME-UINT32 gdzie,

CMD (1bajt-unsigned char) – 0x03

LED-MASK-UINT32 – (4 bajty unsigned int) – Maska LED na której ma być wykonana operacja u nas są to trzy diody więc można ustawić bity 0,1,2 oraz sygnalizator dźwiękowy bit 3.

TYPE – (1 bajt) – rodzaj komendy, gdzie

0x01 - Włącz wybrane LED-y lub sygnalizator w sposób ciągły (nawet po zakończeniu

połączenia)

0x02 - Wyłącz wybrane LED-y i sygnalizator

0x03 - Błyśnij (1 raz) wybranymi LED-ami lub załącz sygnalizator dźwiękowy przez określony czas kolejnym parametrem (stosować ten tryb gdy nie chcemy, by pozostały włączone diody lub sygnalizator po zerwaniu połączenia)

0x04 - Mrugaj cyklicznie wybranymi LED-ami

TIME-UINT32 – (4 bajty) – czas wyrażony w milisekundach, ważne tylko dla komend 0x03,0x04; wówczas określa długość błysku lub jego okres, dla komend załącz i wyłącz wartość jest ignorowana.

TOUT-UNIT32 – (4 bajty) – czas całkowity mrugania diod w trybie mrugaj cyklicznie (0x04) w pozostałych trybach ignorowany

Odpowiedz na rozkaz (2 bajty)

0x83, *ERROR_CODE*

c) Podtrzymaj połączenie *keep_alive* (0x04)

Rozkaz ten nic nie robi, służy tylko do podtrzymywania aktywnej sesji, normalnie aplikacja po 30 sekundach bezczynności rozłącza się (standardowo dla protokołu TCP), *keep_alive* pozwala przedłużyć przetrzymywanie aplikacji w stanie połączenia.

Rozkaz ma postać: 0x04, 0x00

Odpowiedz na rozkaz (2 bajty)

0x84, *ERROR_CODE*

d) Rozłączenie połączenia (0x05)

Komenda powoduje rozłączenie i zakończenie bieżącej sesji, aby sesja została odczytana za bezbłędną rozłączenie musi nastąpić w wyniku wykonania tej komendy

Komenda ma postać: 0x05, 0x00

Odpowiedz na rozkaz (2 bajty)

0x85, *ERROR_CODE*

Kody błędów – *ERROR_CODE*

RFID_RESPONSE_OK (0x00) – Wszystko OK

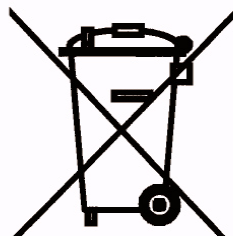
RFID_RESPONSE_ERR_EEPROM (bit 0) – Błąd pamięci EEPROM

RFID_RESPONSE_ERR_LEN (bit 1) – Błędna długość komendy

RFID_RESPONSE_ERR_ARG (bit 2) – Błędny argument komendy

Uwaga: Nie wolno wysyłać kolejnej komendy przed otrzymaniem odpowiedzi

2.5 Uwagi



Oznaczenie tym symbolem produktu oznacza, że nie należy go wyrzucać razem z innymi odpadami. Może to spowodować negatywne skutki dla środowiska i zdrowia ludzi. Urządzenie powinno zostać poddane przetworzeniu lub recyklingowi. Użytkownik jest odpowiedzialny za dostarczenie zużytego sprzętu do wyznaczonego punktu gromadzenia zużytych urządzeń elektrycznych i elektronicznych. Szczegółowe informacje na temat recyklingu można uzyskać u odpowiednich władz lokalnych, w przedsiębiorstwie zajmującym się usuwaniem odpadów lub w miejscu zakupu produktu.