



# Biblioteki pakietu STEP 7 Basic 10.5 do RFID (protokół 3964R)

*Systemy RFID powoli zajmują dominującą pozycję na rynku identyfikacji bezstykowej. Coraz częściej czytniki tego typu są także integrowane w systemach sterowania automatycznego. Dzięki nim automatyzowane są procesy logistyczne, mogą powstawać także niemal bezobsługowe magazyny, często spotykamy się z nimi także w codziennym życiu (sklepach, na basenach, parkach rozrywki itp.). Szybką integrację systemów RFID umożliwiają m.in. takie rozwiązania jak przedstawiona w artykule biblioteka dla pakietu STEP 7.*

Biblioteka "rfid" zajmuje 12,6 kB w pamięci głównej sterownika S7-1200. W celu umożliwienia komunikacji z urządzeniem RFID funkcja "rfid\_run" musi być wywoływana cyklicznie. Blok danych "rfid" zawiera wszystkie zmienne potrzebne do działania funkcji "rfid\_run". Bloki danych grupy "\_com\_db" są zarezerwowane dla konfiguracji i pracy modułu komunikacyjnego RS232.

Blok programowy "rfid\_run" zawiera wszystkie funkcje odnoszące się do konfiguracji i inicjalizacji sterownika SIMATIC S7-1200 oraz komunikacji z urządzeniem RF300. Funkcja jest podzielona na cztery procedury INIT, RESET, RCV i WRITE. Wykaz zakresów ciągów kroków (*step chain*) zestawiono w **tabeli 1 (rysunek 1)**.

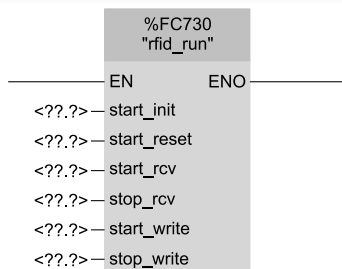
Na etapie konfiguracji programu STEP 7 Basic 10.5 moduł komunikacyjny RS232 musi być umieszczony w kieszeni dla modułów urządzenia S7-1200, chociaż sama konfiguracja jest wykonywana wyłącznie z użyciem biblioteki.

Procedury RCV i WRITE (**tabela 2**) znajdują się po uruchomieniu w stanie oczekiwania wewnątrz pętli. Gdy transponder RF300 znajdzie się w zasięgu pola, rozpoczyna się wykonywanie odpowiedniej procedury, a następnie program powraca do pętli oczekiwania (licznik kroków „301” lub „401”).

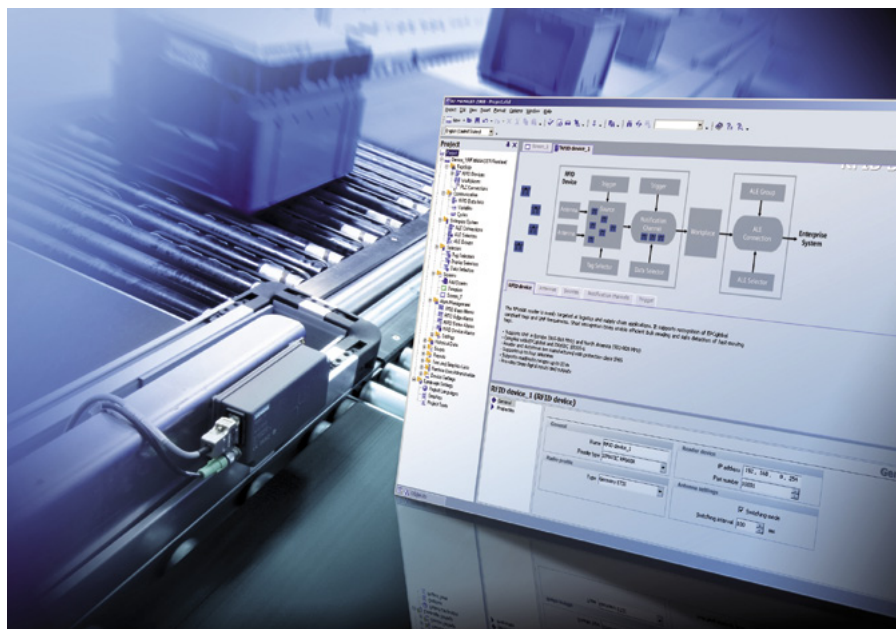
Schemat blokowy programu z **rysunku 2** ilustruje zasadę start-stop działania procedur zawartych w funkcji "rfid\_run". Na etapie inicjalizacji każdy krok jest monitorowany. Jeżeli któryś z kroków nie może zostać wykonany, pojawia się odpowiedni komentarz w słowie statusu. Proces

**Tabela 1. Wykaz zakresów ciągów kroków (step chain)**

Procedura	Dane wejściowe	Zakres kroków
INIT	start_init	100 do 199
RESET	start_reset	200 do 299
RCV	start_rcv stop_rcv	300 do 399
WRITE	start_write stop_write	400 do 499

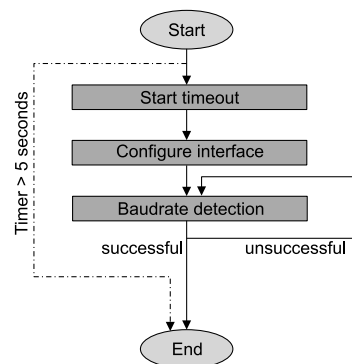


Rysunek 1.



**Tabela 2. Wykaz procedur z ich warunkami uruchomienia i zatrzymania**

Procedura	Warunek startu	Warunek stopu
INIT	Dodatknie zbocze na wejściu cyfrowym start_init Żadna inna procedura nie jest wykonywana	Pomyślnie wykonanie Błędne wykonanie
RESET	Dodatknie zbocze na wejściu cyfrowym start_reset Żadna inna procedura nie jest wykonywana	Pomyślnie wykonanie Błędne wykonanie
RCV	Dodatknie zbocze na wejściu cyfrowym start_rcv Żadna inna procedura nie jest wykonywana	Błędne wykonanie Dodatknie zbocze na wejściu cyfrowym stop_rcv
WRITE	Dodatknie zbocze na wejściu cyfrowym start_write Żadna inna procedura nie jest wykonywana	Błędne wykonanie Dodatknie zbocze na wejściu cyfrowym stop_write



Rysunek 3.

inicjalizacji jest wówczas zatrzymywany. Jeżeli powyższy ciąg kroków kończy się, z pozytywnym lub negatywnym wynikiem, po upływie

pięciu sekund, wówczas proces zostaje anulowany (rysunek 3). Ten przypadek może wystąpić na przykład w razie braku połączenia między sterownikiem S7-1200 i czytnikiem RF300.

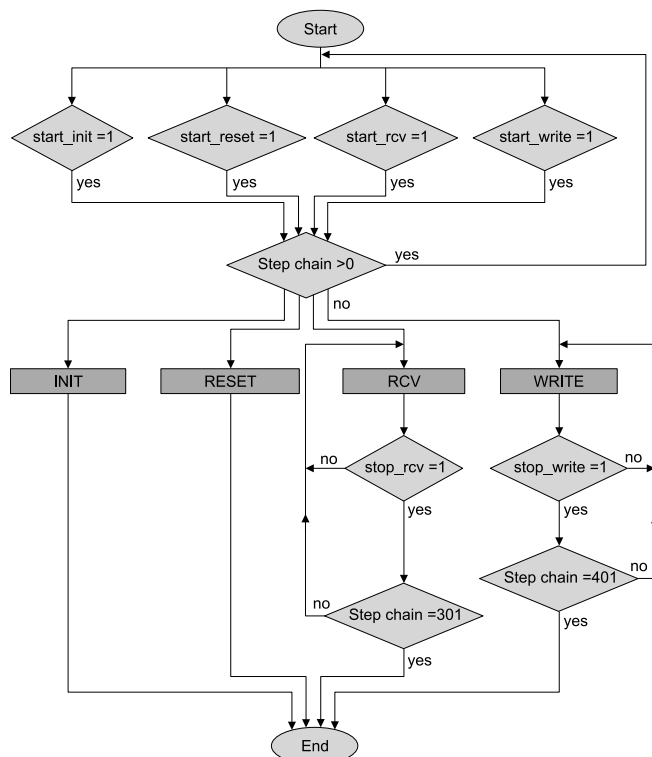
Moduł komunikacyjny RS232 sterownika S7-1200 po zakończeniu procedury „INIT” jest przygotowany do obsługi protokołu 3964R. Po wykonaniu tej procedury moduł zostaje skonfigurowany jak pokazano w tabeli 3.

**Tabela 3. Konfiguracja modułu komunikacyjnego**

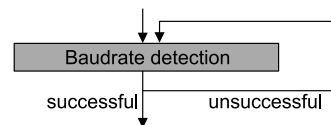
Nr	Parametr	Wpis
1.	Protokół komunikacyjny	Protokół punkt-punkt
2.	Szybkość transmisji	19,2 kbitów/s
3.	Liczba bitów danych	8 bitów danych na znak
4.	Parzystość	Nieparzysta liczba jedynek
5.	Bity stopu	Jeden bit stopu

Oprócz konfiguracji interfejsu komunikacyjnego, procedura detekcji nadchodzących danych jest sparametryzowana. Konfiguracja jest obsługiwana za pośrednictwem bloku danych „RCV\_CFG\_CONDITION[DB732]” (tabela 4).

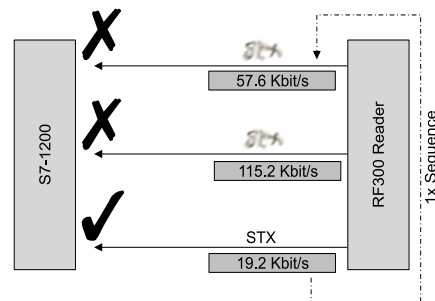
Ze względu na to, że czas odpowiedzi czytnika RF300 jest rzędu milisekund, następny odbierany znak musi



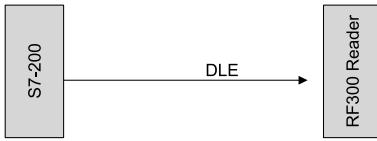
Rysunek 2.



Rysunek 4.

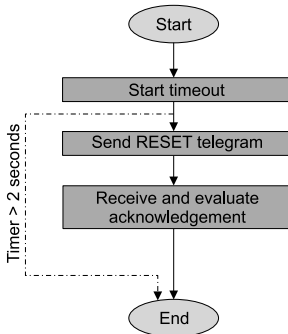


Rysunek 5.



Rysunek 6.

W przeciwieństwie do rozwiązania stosowanego w S7-200, rozwiązanie przyjęte w sterowniku S7-1200 jest oparte na dwukierunkowej transmisji szeregowej. Oznacza to, że możliwa jest praca „with presence message”. Można wówczas zaprojektować dynamiczny proces identyfikacji.



Rysunek 7.

zostać rozpoznany jako nowy komunikat (patrz tabela 4, punkty 1 i 2) natychmiast po zakończeniu procedury odbioru (patrz tabela 4, pkt. 10).

Po włączeniu zasilania czytnika RF300 jest wysyłany znak sterujący STX (hex: 02) z różnymi szybkościami transmisji. Sekwencja ta jest powtarzana do chwili, gdy partner tego połączenia, w tym przypadku sterownik S7-1200, odpowie na wysłany znak sterujący STX w odpowiednim czasie (rysunek 4). Ponieważ moduł komunikacyjny sterownika S7-1200 jest skonfigurowany, poprzez procedurę "INIT", do pracy z szybkością 19,2 kb/s, bezbłędny odbiór znaku STX nastąpi tylko w trakcie odpowiedniej sekwencji (19,2 kb/s + sygnały RS232) wysyłanej przez czytnik RF300. Wszystkie inne znaki zostają uznane za niewłaściwe i odrzucone przez procedurę inicjalizacji (rysunek 5). Po wykryciu przez sterownik S7-1200 znaku STX, natychmiast jest wysyłane do czytnika RF300 potwierdzenie w postaci znaku sterującego DLE (hex: 10) – rysunek 6.

Po wykonaniu tego kroku procedura inicjalizacji zostaje zakończona.

W trakcie procedury RESET następuje wysłanie telegramu do czytnika RF300, po czym zostaje sprawdzona otrzymana odpowiedź. Telegram RESET musi być koniecznie wysłany do czytnika RF300 przed wszystkimi innymi telegramami. Zatem procedurę RESET należy wywołać bezpośrednio po procedurze "INIT". W dalszym przebiegu pracy można także używać telegramów RESET do kasowania błędnych stanów czytnika RF300. Telegram ten można wysłać do czytnika RF300 w każdej chwili.

Podczas tej procedury monitorowany jest każdy krok pod względem poprawności odpowiedzi. Jeżeli jakiś krok nie może zostać wykonany, powoduje to wpisanie odpowiedniego ko-

Tabela 4. Konfiguracja bloku danych

Nr	Parametr	Wpis
1.	Warunek startowy	Dowolny znak (po wykryciu czasu nieaktywności)
2.	Czas oczekiwania linii zanim można rozpocząć odbiór	0 odstępów bitowych
3.	Znak startu	bez znaczenia
4.	Znak stopu	bez znaczenia
5.	Warunek zakończenia	Przekroczenie czasu w granicach sekwencji kroków
6.	Maksymalna długość	bez znaczenia
7.	Długość pola komunikatu (N, LENGTHSIZE, LENGTHM)	bez znaczenia
8.	Przekroczenie czasu odpowiedzi	bez znaczenia
9.	Przekroczenie czasu komunikatu	bez znaczenia
10.	Przekroczenie czasu w ramach sekwencji kroków	12 odstępów bitowych

Tabela 5. Anulowanie procedury RCV

Aktywna procedura	Wynik
Procedura RCV	Za pomocą procedury RCV jest poprawnie wykrywany komunikat presence, po czym można wykonać następujące kroki procedury RCV.
Inna procedura	Jeżeli sterownik S7-1200 nie jest w stanie wykonywania procedury RCV, komunikat presence nie może zostać wykryty i potwierdzony. Powoduje to wygenerowanie przez czytnik komunikatu o błędzie. Konieczne jest wyzerowanie czytnika RF300 (telegram RESET).

mentarza do słowa statusu. Następuje wówczas anulowanie procedury RESET. Jeżeli powyższy ciąg kroków nie zakończy się, to proces zostaje anulowany po upływie limitu czasu równego dwie sekundy (rysunek 7).

Rozkaz RESET powoduje sprowadzenie czytnika RF300 do ściśle zdefiniowanego stanu. Parametry transmisji wyznaczają sposób działania systemowego (tryb pracy) czytnika RF300.

Stosowanie domyślnych ustawień bloków bibliotecznych powoduje ustawienie "operation with presence message". Poza tym zostaje zgaszona dioda LED sygnalizująca błąd. Stosowany jest tryb RF300 "telegram mode". Tryb ten może być wybrany przez użytkownika w zależności od użytego transpondera RF300. Funkcjonalność bloków bibliotecznych zostaje zachowana także w trybie ISO.

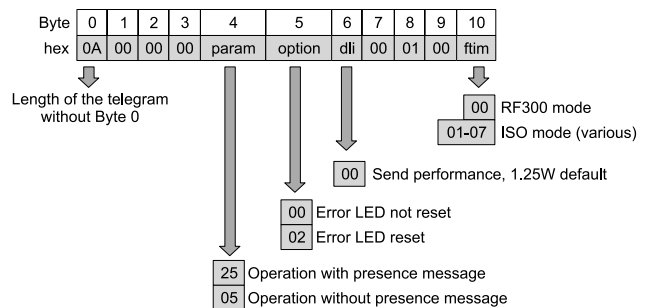
Bezpośrednio po odebraniu telegramu RESET czytnik RF300 zwraca potwierdzenie do modułu komunikacyjnego RS232. Po uznaniu potwierdzenia za poprawne, procedura zostaje zakończona.

W trakcie procedury RCV następuje wysłanie telegramu do czytnika RF300, a następnie zostaje sprawdzona otrzymana odpowiedź (rysunek 9).

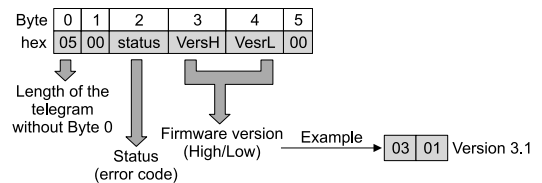
Podczas tej procedury monitorowany jest każdy krok pod względem poprawności odpowiedzi. Jeżeli jakiś krok nie może zostać wykonany, powoduje to wpisanie odpowiedniego komentarza do słowa statusu. Następuje wów-

czas anulowanie procedury RCV (rysunek 10, tabela 5).

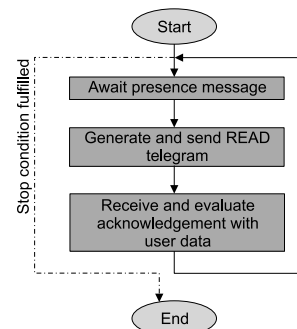
Powyższy ciąg kroków nie kończy się w wyniku pomyślnego wykonania, ale pozostaje na określonej pozycji w ciągu kroków



Rysunek 8.



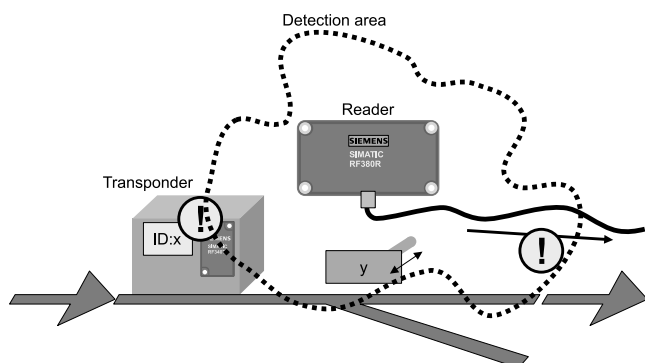
Rysunek 9.



Rysunek 10.

Tabela 6. Zmienne wejściowe i wyjściowe funkcji „rfid\_run”

Nr	Nazwa	Kierunek przekazywania	Typ danych	Opis/uwagi
1.	EN	IN(FC)	BOOL	Aktywuje podprogram
2.	start_init	IN(FC)	BOOL	Uruchamia proces inicjalizacji interfejsu oraz wykrywania szybkości transmisji. Funkcja "rfid_run" wykrywa tylko dodatnie zbocze.
3.	start_reset	IN(FC)	BOOL	Uruchamia proces wysyłania telegramu RESET. Funkcja "rfid_run" wykrywa tylko dodatnie zbocze.
4.	start_rcv	IN(FC)	BOOL	Uruchamia proces odczytu danych z transpondera RF300 za pomocą czytnika RF300. Funkcja "rfid_run" wykrywa tylko dodatnie zbocze.
5.	stop_rcv	IN(FC)	BOOL	Kończy procedurę RCV jeżeli nie jest obsługiwany żaden transponder. Funkcja "rfid_run" wykrywa dodatni sygnał w ciągu kilku cykli zegarowych CPU do chwili wystąpienia warunków zakończenia procedury RCV. Powinna być resetowana tylko przez program użytkownika po pomyślnym zakończeniu procedury RCV (należy sprawdzić słowo statusu). Po poprawnym zakończeniu procedury RCV zmienna "return_done" przyjmuje wartość "TRUE".
6.	start_write	IN (FC)	BOOL	Uruchamia proces zapisu do transpondera RF300 za pośrednictwem czytnika RF300. Funkcja "rfid_run" wykrywa tylko dodatnie zbocze.
7.	stop_write	IN (FC)	BOOL	Zatrzymuje procedurę WRITE jeżeli nie trwa zapis żadnego transpondera RF300. Funkcja "rfid_run" wykrywa dodatni sygnał w ciągu kilku cykli zegarowych CPU do chwili wystąpienia warunków zakończenia procedury WRITE. Po pomyślnym zakończeniu procedury WRITE zmienna "return_done" przyjmuje wartość "TRUE".
8.	cmd_length	IN (DB)	BYTE	Obowiązuje w połączeniu z procedurami „RCV” i „WRITE”. Określa długość danych do zapisu lub odczytu. Maksymalna wartość: 248. Wartość zabroniona: 0.
9.	cmd_address	IN (DB)	INT	Obowiązuje w połączeniu z procedurami „RCV” i „WRITE”. Określa adres początkowy (hex) danych do zapisu lub odczytu. Zależy od zastosowanego transpondera RF300. Więcej informacji: RF300 System Manual rozdz. 6.2, ID: 21738946 patrz rysunek 1.16
10.	cmd_data_to_write	IN (DB)	STRING	Obowiązuje w połączeniu z procedurą „WRITE”. Określa dane do zapisu do transpondera RF300. Maksymalna długość: 248.
11.	return_done	OUT (DB)	BOOL	Specyfikuje bezbłędny przebieg dowolnej procedury funkcji „rfid_run”. Początkowo (po starcie CPU) ma wartość logiczną "TRUE". Ma wartość "TRUE" (jeżeli ostatnia procedura zakończyła się poprawnie) aż do startu nowej procedury.
12.	return_error	OUT (DB)	BOOL	Specyfikuje bezbłędny przebieg dowolnej procedury funkcji „rfid_run”. Ma wartość "TRUE" (jeżeli ostatnia procedura zakończyła się niepoprawnie) aż do startu nowej procedury. Musi być zawsze badana razem ze zmienną "return_status".
13.	return_rcv_ok	OUT (DB)	BOOL	Specyfikuje pomyślny odczyt danych z transpondera RF300. Obowiązuje zawsze w połączeniu ze zmienną "return_transponder_content". Przyjmuje wartość "TRUE" dla pewnych okresów CPU i powinna być kontrolowana za pośrednictwem wykrywania zbocza.
14.	return_write_ok	OUT (DB)	BOOL	Specyfikuje pomyślny zapis danych do transpondera RF300. Przyjmuje wartość "TRUE" dla pewnych cykli CPU i powinna być kontrolowana za pośrednictwem wykrywania zbocza.
15.	return_status	OUT (DB)	Word	Może być używana po wystąpieniu błędów do określenia źródła błędu. Obowiązuje zawsze w połączeniu ze zmienną "return_error". Jest to wartość licznika ciągu kroków "step_cnt" z bloku danych "[DB730]" pomnożona przez 10. Podaje bezpośrednią informację o kroku powodującym powstanie błędu w funkcji "rfid_run". Jest opisana dalej w rozdziale 4.
16.	Return_transponder_content	OUT(DB)	STRING	Specyfikuje dane odczytane z transpondera RF300. Obowiązuje zawsze w połączeniu ze zmienną "return_rcv_ok". Zawiera po prostu kopię danych użytkownika bez znaków sterujących. Należy ją przetwarzać natychmiast po uaktywnieniu wyjścia "return_rcv_ok".



Rysunek 11.

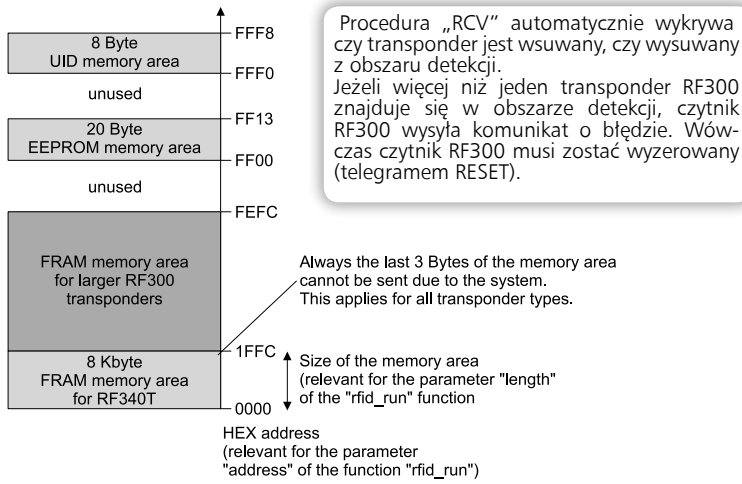
w celu powtórnego wykonania w razie konieczności. Jeżeli transponder RF300 pojawia się w obszarze detekcji czytnika RF300, następuje samoczynne wysłanie komunikatu presencie do sterownika S7-1200 (rysunek 11).

Procedura RCV umożliwia odczytanie z transpondera danych

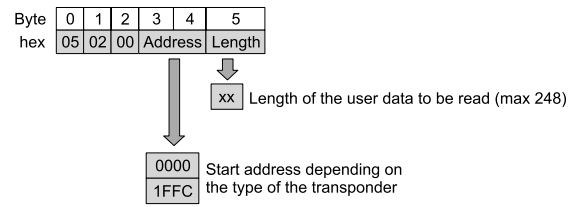
użytkownika. Na przykład transponder typu RF340 ma pamięć FRAM o pojemności 8 kB oraz pamięć EEPROM o pojemności 20 bajtów (mapa pamięci na rysunku 12).

Za pomocą telegramu READ dane są odczytywane z transpondera. Przekazywane parametry (adres początkowy i długość)

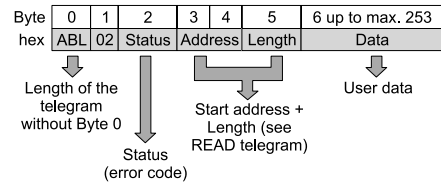
Podczas wykrywania szybkości transmisji czytnik RF380R zmienia zarówno szybkość, jak również typ interfejsu. Różne prędkości są testowane w standardach RS422 oraz RS232.



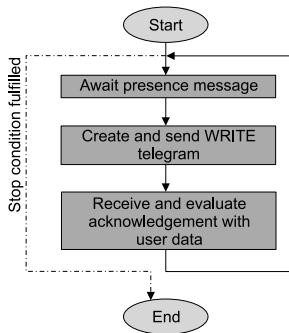
Rysunek 12.



Rysunek 13.



Rysunek 14.



Rysunek 15.

określają jakie dane i z jakiego obszaru pamięci mają zostać odczytane (rysunek 13).

Czytnik RF300 reaguje natychmiast na telegram READ potwierdzeniem zawierającym żądane dane użytkownika (rysunek 14). Po sprawdzeniu ważności potwierdzenia, dane użytkownika są gotowe do pobrania i wpisania do oddzielnej zmiennej. Zarówno znaki sterujące, jak i informacje o statusie zostają usunięte. Na tym kończy się wykonywanie procedury i ciąg kroków pozostaje w pętli oczekiwania (patrz "step chain = 301" na rysunku 1.2).

Za pośrednictwem procedury WRITE następuje wysłanie telegramu do czytnika RF300 i następnie przetworzenie otrzymanego potwierdzenia (rysunek 15). Podczas tej procedury monitorowany jest każdy krok pod względem poprawności odpowiedzi. Jeżeli jakiś krok nie może zostać wykonany, powoduje to wpisanie odpowiedniego komentarza do słowa statusu. Następuje wówczas anulowanie procedury WRITE.

Powyższy ciąg kroków nie kończy się w wyniku pomyślnego wykonania, ale pozostaje na określonej pozycji w ciągu kroków w celu powtórnego wykonania w razie konieczności.

Dane użytkownika można wpisać do określonego obszaru pamięci transpondera RF300 za pomocą telegramu WRITE. Przesyłane parametry adres początkowy i długość określają miejsce w pamięci transpondera, które ma być wypełnione danymi użytkownika (rysunek 16).

Tab. 7. „rfid[DB730]" → "return\_status"

Wartość statusu	Znaczenie
1000	Przekroczony timeout w procedurze „INIT”.
2000	Przekroczony timeout w procedurze „RESET”.
2050	Niepomyślny przebieg uzgodnienia 3964R. Brak gotowości czytnika RF300 do odbioru telegramu RESET.
2090	Nie odebrano potwierdzenia 3964R. Telegram RESET od urządzenia S7-1200 został rozpoznany przez czytnik RF300 jako nieważny i odrzucony.
2120	Nie odebrane żądanie transmisji 3964R. Żądanie wysłania potwierdzenia telegramu dla telegramu RESET nie zostało wysłane przez czytnik RF300.
2160	Potwierdzenie telegramu wysłane przez czytnik RF300 nie zostało rozpoznane przez urządzenie S7-1200 jako ważne.
3080	Telegram presence uznany za błędny.
3130	Niepomyślne początkowe uzgodnienie 3964R. Czytnik RF300 nie gotowy do odbioru telegramu READ.
3200	Nieodebrane żądanie nadawania 3964R. Czytnik RF300 nie wysyła żądania wysłania telegramu odpowiedzi na telegram READ.
3240	Telegram odpowiedzi na telegram READ wysłany przez czytnik RF300 nie został rozpoznany przez urządzenie S7-1200 jako ważny.
4080	Telegram presence uznany za błędny.
4130	Niepomyślny przebieg uzgodnienia 3964R. Brak gotowości czytnika RF300 do odbioru telegramu WRITE.
4200	Nie odebrano potwierdzenia 3964R. Telegram WRITE od urządzenia S7-1200 został rozpoznany przez czytnik RF300 jako nieważny i odrzucony.
4230	Telegram odpowiedzi na telegram WRITE wysłany przez czytnik RF300 nie został rozpoznany przez urządzenie S7-1200 jako ważny.

Czytnik RF300 odpowiada natychmiast na telegram WRITE wysyłając potwierdzenie (rysunek 17). Po sprawdzeniu ważności potwierdzenia kończy się wykonywanie procedury i ciąg kroków pozostaje w pętli oczekiwania (patrz "step chain = 401" na rysunku 2).

**Interfejs biblioteki**

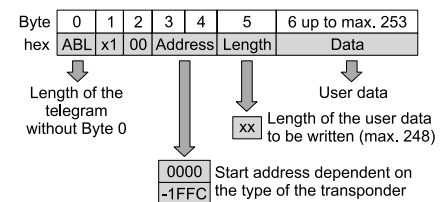
W tabeli 6 przedstawiono najważniejsze zmienne wejściowe i wyjściowe funkcji „rfid\_run”. Dzieli się one na zmienne wewnętrzne funkcji, tymczasowe, oraz zmienne globalne w bloku danych.

Każda procedura zawarta w funkcji "rfid\_run" podaje informacje o przyczynie błędu powodującego niepoprawne zakończenie tej procedury, za pomocą sumarycznego słowa statusu w bloku danych "rfid[DB730]". Adresem symbolicznym zmiennej jest "return\_status" (tabela 7). Słowo statusu jest wyprowadzane w postaci liczby dziesiętnej i ma znaczenie podane w poniższej tabeli. Te wartości statusu

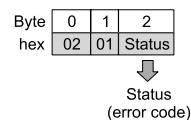
nie mają nic wspólnego z numerami błędów wewnętrznych urządzenia RFID.

**Tomasz Starak**

Opracowano na podstawie dokumentacji firmy Siemens.



Rysunek 16.



Rysunek 17.