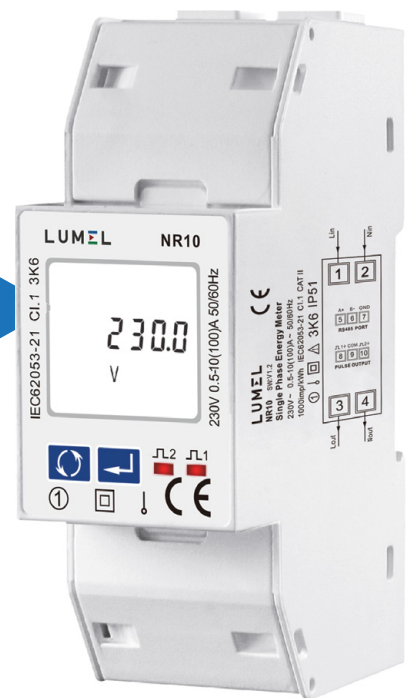


WIELOFUNKCYJNY MIERNIK MOCY

NR10



INSTRUKCJA OBSŁUGI
PROTOKÓŁ MODBUS



Spis treści

1. Implementacja protokołu NR10
 - 1.1 Przegląd protokołu Modbus
 - 1.2 Rejestr wejściowy
 - 1.2.1 Rejestry wejściowe NR10
 - 1.3 Rejestry protokołu Modbus i konfiguracja miernika cyfrowego:
 - 1.3.1 Parametry rejestru protokołu MODBUS
- 2 Informacje ogólne o RS485
 - 2.1 Half Duplex
 - 2.2 Podłączanie instrumentów
 - 2.3 Zaciski A i B
 - 2.4 Rozwiązywanie problemów
- 3 Informacje ogólne o protokole MODBUS
 - 3.1 Format komunikatów protokołu MODBUS
 - 3.2 Tryby transmisji szeregowej
 - 3.3 Kontrola czasu komunikatów protokołu MODBUS (Tryb RTU)
 - 3.4 Sposób szeregowej transmisji znaków
 - 3.5 Metody sprawdzania błędów
 - 3.5.1 Sprawdzanie parzystości
 - 3.5.2 Sprawdzanie CRC
 - 3.6 Kody funkcji
 - 3.7 Format zmiennoprzecinkowy IEEE
 - 3.8 Obsługiwane polecenia protokołu MODBUS
 - 3.8.1 Odczyt rejestrów wejściowych
 - 3.9 Rejestry do zapisu i odczytu
 - 3.9.1 Odczyt rejestrów do odczytu
 - 3.9.2 Zapis rejestrów do zapisu i odczytu
 - 3.10 Odpowiedź na wyjątek
 - 3.11 Kody wyjątków
 - 3.11.1 Tabela kodów wyjątków
 - 3.12 Diagnostyka

1. Implementacja protokołu Modbus inteligentnego miernika NR10

1.1 Przegląd protokołu Modbus

Ten rozdział zawiera podstawowe informacje dotyczące połączenia inteligentnego miernika z siecią protokołu Modbus. Jeśli wymagane są dodatkowe informacje lub szczegóły dotyczące inteligentnej implementacji, należy zapoznać się z rozdziałem 2 i 3 niniejszego dokumentu.

RS485 to dwukierunkowa, pełna lub półdupleksowa struktura magistrali komunikacyjnej składająca się z pojedynczego urządzenia master i co najmniej jednego urządzenia slave. Maksymalna liczba urządzeń slave może się jednak znacznie różnić w zależności od systemu; większość producentów ogranicza maksymalną liczbę urządzeń slave między 16 a 32. Większość sygnałów RS485 działa prawidłowo na prądzie podkładu Bias DC 5 woltów. Sygnały są nadawane i odbierane naprzemiennie. Oznacza to, że każda linia działa odwrotnie od drugiej, a każda z nich również zależy do drugiej z elektrycznego punktu widzenia. Odbiornik analizuje różnicę, a nie bezwzględną wartość napięcia dwóch sygnałów. Jest to określone jako „Bias linii” i jest krytyczne w aplikacjach RS-485. Różnica większa niż 0,3 V jest ogólnie akceptowana jako ważna, ale w zależności od systemu może wynosić nawet 0,7 V. Wartości bezwzględne poniżej są uważane za „niezdefiniowane” lub „szare” i mogą skutkować wysokim lub niskim odczytem przez odbiornik. W wielu niez izolowanych aplikacjach, oprócz 2 linii danych, zapewnione jest połączenie „masy sygnałowej” lub „osłony”, jednak nie jest to konieczne, ponieważ sygnały odnoszą się do siebie nawzajem, a nie do masy bezwzględnej. Służy to jedynie jako punkt uziemienia dla ekranowania kabla komunikacyjnego. Ważne jest, aby pamiętać, że w niez izolowanych aplikacjach uziemienie ekranu powinno być podłączone tylko na jednym końcu magistrali. Jeśli masy są połączone ze sobą w systemach niez izolowanych, nawet niewielka różnica napięcia między absolutnymi punktami uziemienia spowoduje powstanie „pętli uziemienia”, która może spowodować poważne uszkodzenie sprzętu. Należy zauważyć, że w odizolowanych urządzeniach powyższe stwierdzenie nie ma zastosowania, ponieważ nie ma bezpośredniego połączenia między punktami uziemienia, jednak we wszystkich aplikacjach należy zastosować odpowiednie uziemienie ekranu kabla, aby wyeliminować i / lub zmniejszyć zakłócenia elektryczne.

Nasz NR10 oferuje opcję komunikacji RS485 do bezpośredniego połączenia z SCADA lub innymi systemami komunikacyjnymi przy użyciu protokołu Modbus RTU. Protokół Modbus ustala format zapytania urządzenia master, umieszczając w nim adres urządzenia, kod funkcji definiujący żądaną akcję, wszelkie dane do wysłania oraz pole sprawdzania błędów. Komunikat odpowiedzi urządzenia slave jest również konstruowany przy użyciu protokołu Modbus. Zawiera pola potwierdzające podjęte działania, wszelkie dane do zwrócenia i pole sprawdzania błędów. NR10 nie odpowie jeśli wystąpi błąd w odbiorze wiadomości. Jeśli NR10 nie może wykonać żądanej akcji, wygeneruje komunikat o błędzie i wyśle go jako odpowiedź.

Interfejsem elektrycznym jest 2-przewodowy RS485 wykorzystujący 2 zaciski śrubowe. Połączenie należy wykonać za pomocą kabla ekranowanego typu skrętka (zwykle Belden o rozmiarze 22, 8761 lub równoważny). Wszystkie połączenia „A” i „B” są połączone szeregowo. Topologia linii może, ale nie musi, wymagać rezystorów dopasowujących w zależności od rodzaju i długości użytego kabla. Topologia pętli (pierścienia) nie wymaga rezystora dopasowującego. Impedancja rezystora dopasowującego powinna odpowiadać impedancji kabla i znajdować się na obu końcach linii. Oba końce kabla powinny być zakończone rezystorem 120 Ω (min. 0,25 W). W sieci RS485 maksymalna dozwolona długość wynosi 1200 metrów (3900 stóp). Można podłączyć maksymalnie 32 węzły elektryczne, w tym sterownik. Adres każdego NR10 można ustawić na dowolną wartość od 1 do 247. Tryb rozgłoszeniowy (adres 0) nie jest obsługiwany.

Format każdego bajtu w trybie RTU to:

System kodowania:	8 bitów na bajt
Format danych:	4 bajty (2 rejestry) na parametr. Format zmiennoprzecinkowy (zgodnie z IEEE 754) Najpierw najbardziej znaczący rejestr (domyślnie). Ustawienie domyślne może zostać zmienione, jeśli jest to wymagane - patrz Rejestr do odczytu i zapisu, parametr „Kolejność rejestrów”.
Pole sprawdzania błędów:	2 bajtowa cykliczna kontrola nadmiarowa (CRC)
Ramki:	1 bit startu 8 bitów danych, najpierw wysłany jest najmniej znaczący bit 1 bit na parzystość / nieparzystość (lub brak parzystości) 1 bit stopu, jeśli używana jest parzystość; 1 lub 2 bity przy braku parzystości

Kodowanie danych

Wszystkie wartości danych w mierniku NR10 są przesyłane jako 32-bitowe liczby zmiennoprzecinkowe IEEE754 (wejście i wyjście), dlatego każda wartość miernika NR10 jest przesyłana za pomocą dwóch rejestrów protokołu Modbus. Wszystkie żądania odczytu rejestru i żądania zapisu danych muszą określać parzystą liczbę rejestrów. Próby odczytu/zapisu nieparzystej liczby rejestrów powodują, że NR10 zwraca komunikat wyjątku protokołu Modbus. Jednakże, dla kompatybilności z niektórymi systemami SCADA, NR10 odpowie na każdy pojedynczy rejestr wejściowy lub rejestr do odczytu/zapisu odczytany z wartością określoną dla typu instrumentu.

NR10 może przesłać maksymalnie 40 wartości w jednej transakcji; dlatego maksymalna liczba rejestrów, których może dotyczyć zapytanie wynosi 80. Przekroczenie tego limitu powoduje, że NR10 wygeneruje odpowiedzi o wyjątku.

Prędkość transmisji danych można wybrać między 2400, 4800, 9600, 19200, 38400.

1.2 Rejestr wejściowy

Rejestry wejściowe służą do wskazania aktualnych wartości zmierzonych i obliczonych wielkości elektrycznych. Każdy parametr jest przechowywany w dwóch kolejnych rejestrach 16 bitowych. W poniższej tabeli przedstawiono adres rejestru 3X oraz wartości bajtów adresu w obrębie komunikatu. Litera A (√) w kolumnie wskazuje, że parametr jest obowiązujący dla konkretnego systemu okablowania. Każdy parametr z krzyżykiem (X) zwróci wartość zero. Każdy parametr jest przechowywany w rejestrach 3X. Kod funkcji protokołu Modbus 04 służy do uzyskania dostępu do wszystkich parametrów.

Na przykład, aby zażądać:	Amps 1	Adres początkowy	=0006
		Liczba rej.	=0002
	Amps 2	Adres początkowy	=0008
		Liczba rej.	=0002

Każde żądanie danych musi być ograniczone do maksymalnie 40 parametrów. Przekroczenie limitu 40 parametrów spowoduje wygenerowanie kodu wyjątku protokołu Modbus.

1.2.1 Rejestry wejściowe NR10

Adres (Rejestr)	Rejestr wejściowy NR10 Parametr		Szesnastkowy adres początkowy protokołu Modbus	
	Opis	Jednostki	Hi Byte	Lo Byte
30085	Całkowita moc demand układu	W	00	54
30087	Maksymalna całkowita moc demand układu	W	00	56
30089	Bieżąca dodatnia moc demand układu	W	00	58
30091	Maksymalna dodatnia moc demand układu	W	00	5A
30093	Bieżąca zwrotna moc demand układu	W	00	5C
30095	Maksymalna zwrotna moc demand układu	W	00	5E
30259	Prąd demand	A	01	02
30001	Napięcie między fazą a neutralnym	V	00	00
30007	Prąd	A	00	06
30013	Moc czynna	W	00	0C
30019	Moc pozorna	VA	00	12
30025	Moc bierna	VAr	00	18
30031	Współczynnik mocy	Brak	00	1E
30037	Kąt fazowy	stopnie	00	24
30071	Częstotliwość	Hz	00	46
30073	Energia czynna pobierana	kWh	00	48
30075	Energia czynna oddawana	kWh	00	4A
30077	Energia bierna pobierana	kVArh	00	4C
30265	Maksymalny prąd Demand	A	01	08
30079	Energia bierna oddawana	kVArh	00	4E
30343	Energia czynna całkowita	kWh	01	56
30345	Energia bierna całkowita	kVArh	01	58

1.3 Rejestry protokołu Modbus i konfiguracja miernika cyfrowego:

Rejestry do zapisu i odczytu służą do przechowywania i wyświetlania ustawień konfiguracji urządzenia. Wszystkie rejestry do zapisu i odczytu niewymienione w poniższej tabeli należy traktować jako zastrzeżone dla użytku producenta i nie należy podejmować prób modyfikacji ich wartości.

Parametry rejestru do zapisu i odczytu można przeglądać lub zmieniać za pomocą protokołu Modbus. Każdy parametr jest przechowywany w dwóch kolejnych rejestrach 4X. Kod funkcji 03 protokołu Modbus jest używany do odczytu parametru, a kod funkcji 16 jest używany do zapisu. Zapis tylko jednego parametru na komunikat.

1.3.1 Parametry rejestru protokołu MODBUS

Adres Rejestr	Parametr	Szesnastkowy adres początkowy protokołu Modbus		Prawidłowy zakres	Tryb
		Bajt Górny	Bajt Dolny		
40013	Relay Pulse Width	00	0A	Zapis okresu pracy przełącznika w milisekundach: 60, 100 lub 200, domyślnie 200.	r/w
40019	Network Parity Stop	00	12	Zapis bitów stopu/parzystości portu sieciowego dla protokołu MODBUS, gdzie: 0 = Jeden bit stopu i brak parzystości, domyślnie. 1 = Jeden bit stopu i parzystość. 2 = Jeden bit stopu i nieparzystość. 3 = Dwa bity stopu i brak parzystości. Aby zastosować, wymagane jest ponowne uruchomienie.	r/w
40021	Network Node	00	14	Zapis adresu węzła portu sieciowego: 1 do 247 dla protokołu MODBUS, domyślnie 1. Aby zastosować, wymagane jest ponowne uruchomienie. Należy zauważyć, że adresy węzłów MODBUS Protocol i Johnson Controls można zmienić za pomocą menu konfiguracji wyświetlacza.	r/w
40029	Network Baud Rate	00	1C	Zapis szybkości transmisji portu sieciowego dla protokołu MODBUS, gdzie: 0 = 2400. 1 = 4800. 2 = 9600, domyślna. 3 = 19200. 4 = 38400. Aby zastosować, wymagane jest ponowne uruchomienie.	r/w
462721	Demand Interval, Slide Time, Automatic Scroll Display Interval (Scroll Time), Backlight Time	F5	00	Format danych: BCD min-min-s-min Scroll Time=0: wyświetlacz nie jest przewijany automatycznie Backlight Time=0: Podświetlenie jest zawsze włączone.	r/w
463761	System Power	F9	10	Domyślny format: Hex (16 bit) 0000: 0,001 kWh (kVArh) /pob (domyślnie) 0001: 0,01 kWh (kVArh) /pob 0002: 0,1 kWh (kVArh) /pob 0003: 1 kWh (kVArh) /pob	r/w

463776	Measurement Mode	F9	20	Format danych: Hex (16 bit) 0001: Mode 1 (Całkowita = Pobierana) 0002: Mode 2 (Całkowita = Pobierana + Oddawana) 0003: Mode 3 (Całkowita = Pobierana - Oddawana)	r/w
363792	Pulse Output & LED Indicator Mode	F9	30	Format danych: Hex (16 bit) 0000: Energia pobierana i oddawana, dioda LED miga w przypadku energii pobieranej i oddawanej 0001: Energia pobierana, dioda LED miga tylko w przypadku energii pobieranej 0002: Energia oddawana, dioda LED miga tylko w przypadku energii oddawanej	ro

2. Informacje ogólne o RS485

Poniższe informacje dotyczą NR10 i mogą być pomocne w przypadku zastosowania sieci mieszanej. RS485 lub EIA (Electronic Industries Association) RS485 to zbalansowany, półduplexowy system transmisji umożliwiający transmisję do 1,2 km. Poniższa tabela podsumowuje standard RS-485:

PARAMETR	
Tryb działania	Różnicowy
Liczba sterowników i odbiorników	32 sterowniki, 32 odbiorniki
Maksymalna długość kabla	1 200 m
Maksymalna szybkość transmisji danych	10 M baud
Maksymalne napięcie równoległe	12 V do -7 V
Minimalne poziomy wyjściowe sterownika (z obciążeniem)	+/- 1,5 V
Minimalne poziomy wyjściowe sterownika (bez obciążenia)	+/- 6 V
Obciążenie sterownika	Minimum 60 Ω
Ograniczenie prądu zwarcia wyjścia sterownika	150 mA dla ziemi, 250 mA dla 12 V 250 mA dla -7 V
Minimalna oporność wejściowa odbiornika	12 kΩ
Czułość odbiornika	+/- 200 mV

Dalsze informacje dotyczące RS485 można uzyskać z EIA lub od producentów urządzeń RS485, na przykład Texas Instruments lub Maxim Semiconductors. Lista ta nie jest wyczerpująca.

2.1 Half Duplex

Półduplex to system, w którym jeden lub więcej nadajników (rozmówców) może komunikować się z jednym lub większą liczbą odbiorników (słuchaczy), przy czym tylko jeden nadajnik jest aktywny jednocześnie. Na przykład „rozmowa” rozpoczyna się od zadawania pytania, osoba, która zadała pytanie, będzie słuchać, dopóki nie otrzyma odpowiedzi, lub dopóki nie zdecyduje, że osoba, której zadano pytanie, nie odpowie.

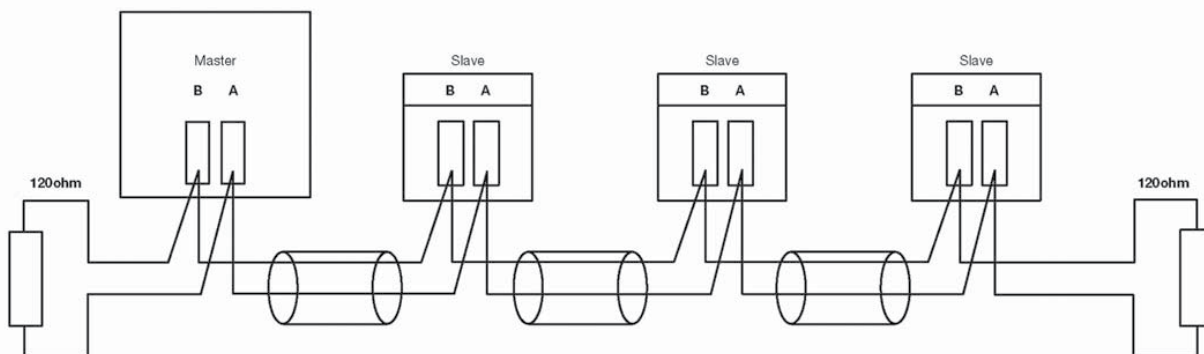
W sieci 485 urządzenie master rozpocznie „rozmowę” przesyłając „zapytanie” do określonego urządzenia slave, a następnie urządzenie master będzie oczekiwało na odpowiedź od urządzenia slave. Jeśli urządzenie slave nie odpowie w określonym czasie (ustawionym w urządzeniu master za pomocą oprogramowania sterującego) to urządzenie master zrezygnuje z „rozmowy”.

2.2 Podłączanie instrumentów

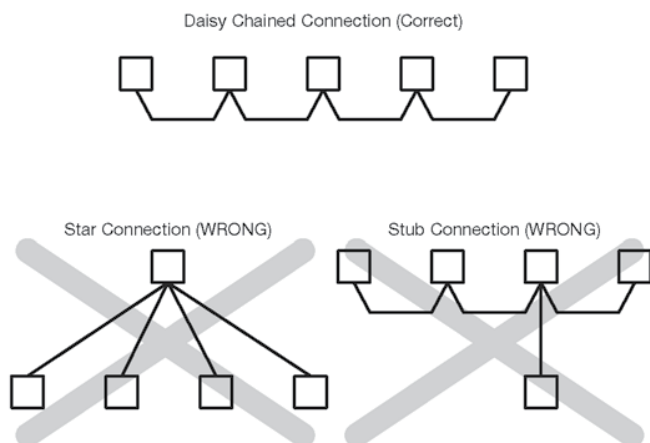
W przypadku podłączenia sieci RS485 do komputera PC należy zachować ostrożność, jeśli rozważamy użycie konwerterów RS232 na 485 wraz z adapterem USB na RS485. Należy rozważyć podłączenie konwertera RS232 na RS485 bezpośrednio do odpowiedniego gniazda RS232 na PC, albo użycie konwertera USB na RS485 lub, w przypadku komputerów stacjonarnych, odpowiedniej wtyczki do karty RS485. (Wiele konwerterów 232: 485 pobiera energię z gniazda RS232. W przypadku korzystania z adaptera USB na RS232, adapter może nie mieć wystarczającej mocy, aby uruchomić konwerter 232: 485).

Należy użyć ekranowanej skrętki dwużyłowej. Aby uzyskać optymalną wydajność w przypadku długich przewodów lub w środowisku, w którym występują zakłócenia, konieczne może być użycie przewodu specjalnie zaprojektowanego dla RS485. Wszystkie zaciski „A” powinny być połączone razem za pomocą jednego przewodu skrętki, wszystkie zaciski „B” powinny być połączone razem za pomocą drugiego przewodu w parze.

Zalecany jest kabel Belden 9841 (pojedyncza para) lub 9842 (dwie pary), albo podobny o impedancji 120 Ω. Oba końce kabla powinny być zakończone rezystorem 120 Ω, 0,25 W (lub większym). Uwaga: Diagram pokazuje tylko topologię okablowania. Należy zawsze przestrzegać oznaczenia zacisku według schematu na bocznej stronie miernika NR10.

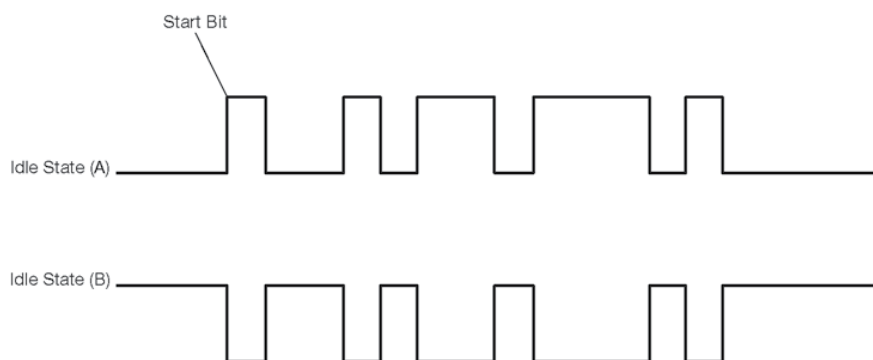


Do każdego zacisku nie mogą być podłączone więcej niż dwa przewody, co zapewnia, że używana jest konfiguracja „Daisy Chain” (szeregowo) lub „straight line” (bezpośrednio). Podłączenie typu „Star” (na gwiazdę) lub sieć z „Stubs (Tees)” (otwarte) nie jest zalecana, ponieważ odbicia w obrębie kabla mogą spowodować uszkodzenie danych.



2.3 Zaciski A i B

Połączenia A i B z miernikiem cyfrowym NR10 mogą być identyfikowane przez obecne w nich sygnały przy aktywnej magistrali RS485:

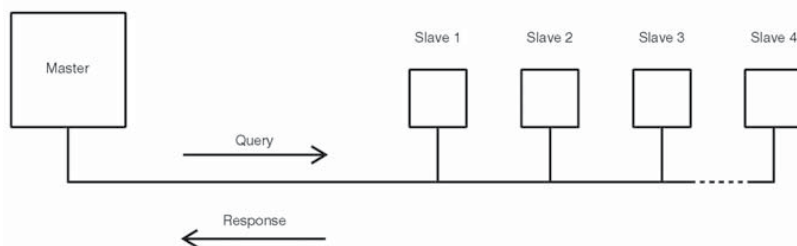


2.4 Rozwiązywanie problemów

- Należy zacząć od prostej sieci, jednego urządzenia master i jednego urządzenia slave. Miernik cyfrowy NR10 znacznie to ułatwia, ponieważ sieć może pozostać niezmienną, a poszczególne urządzenia są odłączane poprzez odłączenie połączenia RS485 z tyłu urządzenia.
- Sprawdź, czy sieć jest prawidłowo połączona. To znaczy, czy wszystkie „A” są połączone ze sobą i wszystkie „B” są połączone ze sobą. Potwierdź, że dane „przesyłane” na RS485 nie są wysyłane z powrotem do komputera liniami RS232. (Ta funkcja jest czasem opcją podłączenia w konwerterze). Zdarza się, że wiele pakietów wykorzystywanych przez komputery nie działa poprawnie, gdy odbierają one odbicie przesyłanego komunikatu. Oprogramowanie komputerowe SpecView i PCView z konwerterem RS232 na RS485 mają tę funkcję.
- Potwierdź, że adres urządzenia jest taki sam jak adres oczekiwany przez urządzenie master.
- Jeśli do „sieci” jest podłączone jedno urządzenie, ale nie więcej niż jedno, należy sprawdzić, czy każde urządzenie ma unikalny adres.
- Każde żądanie danych musi być ograniczone do maksymalnie 40 parametrów. Naruszenie tego wymogu wpłynie na wydajność urządzenia i może skutkować czasem reakcji niezgodnym ze specyfikacją.
- Sprawdź, czy tryb protokołu MODBUS (RTU lub ASCII) i parametry szeregowo (szybkość transmisji, liczba bitów danych, liczba bitów stopu i parzystość) są takie same dla wszystkich urządzeń w sieci.
- Sprawdź, czy urządzenie master żąda zmiennych zmiennoprzecinkowych (par rejestrów umieszczonych na granicach liczb zmiennoprzecinkowych) i nie rozdziela zmiennych zmiennoprzecinkowych.
- Sprawdź, czy kolejność bajtów liczb zmiennoprzecinkowych oczekiwana przez urządzenie master jest taka sama, jak w przypadku NR10. (Pakiety PCView i Citect mogą korzystać z wielu formatów, w tym obsługiwanych przez miernik).
- Jeśli to możliwe, należy użyć drugiego konwertera RS232 na RS485 i podłączyć go między magistralą RS485 a dodatkowym komputerem z zainstalowanym pakietem oprogramowania, który może wyświetlać dane magistrali. Sprawdź, czy przesyłane żądania są prawidłowe.

3. Informacje ogólne o protokole MODBUS

Komunikacja w sieci protokołu MODBUS jest inicjowana (uruchamiana) przez urządzenie master wysyłające zapytanie do urządzenia slave. Urządzenie slave w sposób ciągły monitoruje zapytania przesyłane do niego przez sieć i wykonuje żadaną akcję wysyłając równocześnie odpowiedź zwrótną do urządzenia master. Tylko urządzenie master może inicjować zapytanie.



W protokole MODBUS urządzenie master może zaadresować poszczególne urządzenia slave lub może zainicjować komunikat rozgłoszeniowy dla wszystkich urządzeń slave za pomocą specjalnego adresu „Broadcast”. Miernik cyfrowy NR10 nie obsługuje adresu rozgłoszeniowego.

3.1 Format komunikatów protokołu MODBUS

Protokół MODBUS definiuje format zapytania urządzenia master i odpowiedzi urządzenia slave.

Zapytanie zawiera adres urządzenia (lub adres rozgłoszeniowy), kod funkcji definiujący żadaną akcję, wszelkie dane do wysłania oraz pole sprawdzania błędów.

Odpowiedź zawiera pola potwierdzające podjęte działania, informacje zwrótne i pole sprawdzania błędów. Jeśli wystąpił błąd w odbiorze komunikatu, komunikat jest ignorowany, jeśli urządzenie slave nie może wykonać żadanej akcji, to generuje i wysyła odpowiedź w postaci komunikatu o błędzie. Funkcje protokołu MODBUS używane przez mierniki cyfrowe NR10 kopiują wartości rejestrów 16-bitowych między urządzeniami master i slave. Jednak dane wykorzystywane przez miernik cyfrowy NR10 są generowane w 32-bitowym formacie liczb zmiennoprzecinkowych IEEE 754. Zatem każdy parametr urządzenia jest koncepcyjnie przechowywany w dwóch sąsiednich rejestrach protokołu MODBUS. Zapytanie

Poniższy przykład ilustruje żądanie pojedynczego zmiennoprzecinkowego parametru, tj. dwóch 16-bitowych rejestrów protokołu Modbus.

Pierwszy bajt

Slave Address	Function Code	Start Address (Hi)	Start Address (Lo)	Number of Points (Hi)	Number of Points (Lo)	Error Check (Lo)	Error Check (Hi)
---------------	---------------	--------------------	--------------------	-----------------------	-----------------------	------------------	------------------

Adres slave: 8-bitowa wartość reprezentująca adresowane urządzenie slave (1 do 247), wartość 0 jest zarezerwowana dla adresu rozgłoszeniowego. Mierniki cyfrowe nie obsługują adresu rozgłoszeniowego.

Kod funkcji: 8-bitowa wartość informująca adresowane urządzenie slave, jakie działanie ma zostać wykonane. (3, 4, 8 lub 16 dotyczą miernika cyfrowego)

Adres początkowy (Hi): Górne (najbardziej znaczące) osiem bitów 16-bitowej liczby określającej adres początkowy wymaganych danych.

Adres początkowy (Lo): Dolne (najmniej znaczące) osiem bitów 16-bitowej liczby określającej adres początkowy wymaganych danych. Musi to być liczba parzysta ponieważ rejestry są używane parami i zaczynają się od zera.

Liczba punktów (Hi): Górne (najbardziej znaczące) osiem bitów 16-bitowej liczby określającej liczbę żądanych rejestrów.

Liczba punktów (Lo): Dolne (najmniej znaczące) osiem bitów 16-bitowej liczby określającej liczbę żądanych rejestrów. Musi to być liczba parzysta ponieważ rejestry są używane parami.

Sprawdzanie błędów (Lo): Dolne (najmniej znaczące) osiem bitów 16-bitowej liczby reprezentującej wartość sprawdzenia błędu.

Sprawdzanie błędów (Hi): Górne (najbardziej znaczące) osiem bitów 16-bitowej liczby reprezentującej wartość sprawdzenia błędu.

Odpowiedź

Poniższy przykład ilustruje normalną odpowiedź na pojedynczego zmiennoprzecinkowego parametru, tj. dwóch 16-bitowych rejestrów protokołu Modbus.

Pierwszy bajt

Slave Address	Function Code	Byte Count	First Register (Hi)	First Register (Lo)	Second Register (Hi)	Second Register (Lo)	Error Check (Lo)	Error Check (Hi)
---------------	---------------	------------	---------------------	---------------------	----------------------	----------------------	------------------	------------------

Adres slave: 8-bitowa wartość reprezentująca adres urządzenia slave, które odpowiada.

Kod funkcji: 8-bitowa wartość, która, gdy jest kopią kodu funkcji w zapytaniu, wskazuje, że urządzenie slave rozpoznało zapytanie i odpowiedziało. (Patrz także Odpowiedź na wyjątek).

Liczba bajtów: 8-bitowa wartość wskazująca liczbę bajtów danych zawartych w tej odpowiedzi

Pierwszy rejestr (Hi)*: Górne (najbardziej znaczące) osiem bitów 16-bitowej liczby reprezentującej pierwszy rejestr żądany w zapytaniu.

Pierwszy rejestr (Lo)*: Dolne (najmniej znaczące) osiem bitów 16-bitowej liczby reprezentującej pierwszy rejestr żądany w zapytaniu.

Drugi rejestr (Hi)*: Górne (najbardziej znaczące) osiem bitów 16-bitowej liczby reprezentującej

drugi rejestr żądany w zapytaniu.

Drugi rejestr (Lo)*: Dolne (najmniej znaczące) osiem bitów 16-bitowej liczby reprezentującej drugi rejestr żądany w zapytaniu.

Sprawdzanie błędów (Lo): Dolne (najmniej znaczące) osiem bitów 16-bitowej liczby reprezentującej wartość

sprawdzenia błędu.

Sprawdzanie błędów (Hi): Górne (najbardziej znaczące) osiem bitów 16-bitowej liczby reprezentującej wartość

sprawdzenia błędu.

*Te cztery bajty razem dają wartość żadanego parametru zmiennoprzecinkowego.

Odpowiedź na wyjątek

W przypadku wykrycia błędu w treści zapytania (z wyłączeniem błędów parzystości i niedopasowania sprawdzania błędów), do urządzenia master zostanie wysłana odpowiedź o błędzie (zwana odpowiedzią na wyjątek). Odpowiedź na wyjątek jest identyfikowana przez kod funkcji będący kopią kodu funkcji zapytania, ale z ustawionym najbardziej znaczącym bitem. Dane zawarte w odpowiedzi na wyjątek to jednobajtowy kod błędu.

Pierwszy bajt

Slave Address	Function Code	Error Code	Error Check (Lo)	Error Check (Hi)
---------------	---------------	------------	------------------	------------------

Adres slave: 8-bitowa wartość reprezentująca adres urządzenia slave, które odpowiada.

Kod funkcji: 8-bitowa wartość, która jest kodem funkcji w zapytaniu zsumowana logicznie z 80 hex, co oznacza, że urządzenie slave nie rozpoznaje zapytania lub nie może wykonać akcji Żądania.

Kod błędu: 8-bitowa wartość wskazująca charakter wykrytego wyjątku. (Patrz „Tabela kodów wyjątków” poniżej).

Sprawdzanie błędów (Lo): Dolne (najmniej znaczące) osiem bitów 16-bitowej liczby reprezentującej wartość sprawdzenia błędu.

Sprawdzanie błędów (Hi): Górne (najbardziej znaczące) osiem bitów 16-bitowej liczby reprezentującej wartość sprawdzenia błędu.

3.2 Tryby transmisji szeregowej

Istnieją dwa tryby transmisji szeregowej protokołu MODBUS: ASCII i RTU. Miernik cyfrowy NR10 nie obsługuje trybu ASCII.

W trybie RTU (Remote Terminal Unit) każdy 8-bitowy bajt jest używany w pełnym zakresie binarnym i nie jest ograniczony do znaków ASCII, jak w trybie ASCII. Większa gęstość danych umożliwia lepszą przepustowość danych przy tej samej szybkości transmisji, jednak każdy komunikat musi być przesyłany w ciągłym strumieniu. Jest mało prawdopodobne, aby stanowiło to problem dla nowoczesnych urządzeń komunikacyjnych.

System kodowania: pełny 8-bitowy plik binarny na bajt. W tym dokumencie wartość każdego bajtu będzie wyświetlana jako dwa znaki szesnastkowe, każdy w zakresie 0-9 lub A-F.

Protokół linii: 1 bit startu, po którym następuje 8 bitów danych. 8 bitów danych jest najpierw wysyłanych z najmniej znaczącym bitem.

Opcja użytkownika Parzystości Brak parzystości i 2 Bity stopu

1 Bity stopu: Brak parzystości i 1 Bit stopu
Parzystość i 1 Bit stopu
Nieparzystość i 1 Bit stopu.

Opcja użytkownika szybkości transmisji 4800; 9600; 19200; 38400

Szybkość: Mierniki cyfrowe nie obsługują szybkości transmisji 38400 natomiast oferują szybkość 2400

Szybkość transmisji, parzystość i bity stopu muszą być tak wybrane, aby odpowiadały ustawieniom urządzenia master.

3.3 Kontrola czasu komunikatów protokołu MODBUS (Tryb RTU)

Komunikat protokołu MODBUS ma zdefiniowane punkty początkowe i końcowe. Urządzenia odbierające rozpoznają początek komunikatu, odczytują „Adres Slave”, aby określić, czy są one adresowane i wiedząc, kiedy komunikat zostanie zakończony, mogą użyć bajtów kontroli błędów i bitów parzystości, aby potwierdzić integralność komunikatu. Jeśli kontrola błędów lub parzystość nie powiedzie się, to komunikat jest odrzucony.

W trybie RTU komunikaty zaczynają się od cichego interwału o długości co najmniej znak razy 3,5.

Następnie przesyłany jest pierwszy bajt komunikatu, adres urządzenia.

Urządzenia master i slave monitorują sieć w sposób ciągły, w tym w okresach „cichych”. Gdy zostanie odebrany pierwszy bajt (bajt adresu), każde urządzenie sprawdza czy jest adresowanym urządzeniem. Jeśli urządzenie stwierdzi, że jest adresowanym urządzeniem, rejestruje cały komunikat i postępuje zgodnie z nim, jeśli nie jest adresowane, kontynuuje monitorowanie następnego komunikatu.

Po ostatnim przesłanym bajcie, cichy interwał o długości co najmniej 3,5 znaku oznacza koniec komunikatu. Nowy komunikat może rozpocząć się po tym interwale.

W wersjach miernika NR10 1000 i 2000 wymagany jest cichy interwał minimum 60 ms, aby zagwarantować poprawne odebranie następnego żądania.

Cały komunikat musi być przesyłany jako ciągły strumień. Jeśli przed zakończeniem komunikatu wystąpi cichy interwał o długości ponad 1,5 raza czasu znaku, urządzenie odbierające ignoruje niekompletny komunikat i zakłada, że następny bajt będzie bajtem adresu nowego komunikatu.

Podobnie, jeśli nowy komunikat zaczyna się wcześniej niż po czasie równym 3,5 raza czasu znaku po poprzednim komunikacie, urządzenie odbierające może uznać to za kontynuację poprzedniego komunikatu. Spowoduje to błąd, ponieważ wartość w końcowym polu CRC nie będzie ważna dla połączonych komunikatów.

3.4 Sposób szeregowej transmisji znaków

Gdy komunikaty są przesyłane w standardowych sieciach szeregowych protokołu MODBUS, każdy bajt jest wysyłany w tej kolejności (od lewej do prawej):

Znak transmisji = bit startu + bajt danych + bit parzystości + 1 bit stopu (łącznie 11 bitów):

Najmniej znaczący bit (LSB)

Najbardziej znaczący bit (MSB)

Start	1	2	3	4	5	6	7	8	Parzystość	Stop
-------	---	---	---	---	---	---	---	---	------------	------

Znak transmisji = bit startu + bajt danych + 2 bity stopu (łącznie 11 bitów):

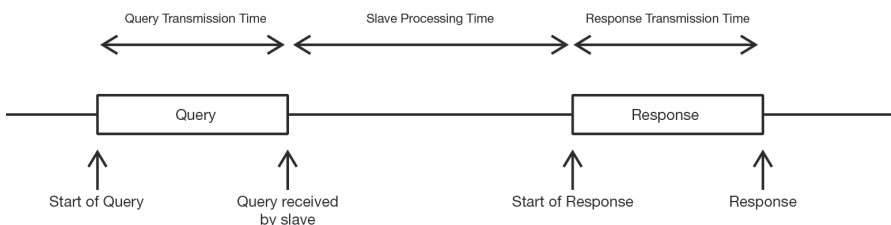
Start	1	2	3	4	5	6	7	8	Parzystość	Stop
-------	---	---	---	---	---	---	---	---	------------	------

Miernik cyfrowy NR10 dodatkowo obsługuje Brak parzystości, Jeden bit stopu.

Znak transmisji = bit startu + bajt danych + 1 bit stopu (łącznie 10 bitów):

Start	1	2	3	4	5	6	7	8	Parzystość	Stop
-------	---	---	---	---	---	---	---	---	------------	------

Urządzenie master jest konfigurowane przez użytkownika w taki sposób, aby czekało na określony czas interwału. Urządzenie master zaczeka przez ten okres czasu, zanim zdecyduje, że urządzenie slave nie odpowie i że transakcja powinna zostać przerwana. Należy zachować ostrożność podczas określania limitu czasu zgodnie ze specyfikacją urządzenia master oraz slave. Urządzenie slave może zdefiniować „czas odpowiedzi” jako okres od otrzymania ostatniego bitu zapytania do przesłania pierwszego bitu odpowiedzi. Urządzenie master może zdefiniować „czas odpowiedzi” jako okres pomiędzy przesłaniem pierwszego bitu zapytania do otrzymania ostatniego bitu odpowiedzi. Czas transmisji komunikatu, który jest funkcją szybkości transmisji, musi być uwzględniony w obliczaniu limitu czasu.



3.5 Metody sprawdzania błędów

Standardowe sieci szeregowe protokołu MODBUS wykorzystują dwa procesy sprawdzania błędów, wspomniane powyżej bajty kontroli błędów sprawdzają integralność komunikatu, podczas gdy kontrola parzystości (parzystość lub nieparzystość) może być zastosowana do każdego bajtu w komunikacie

3.5.1 Sprawdzanie parzystości

Jeśli włączone jest sprawdzanie parzystości - poprzez wybranie Parzystość lub Nieparzystość -ilość „1” zostanie zliczona w części danych każdego znaku transmisji. Bit parzystości zostanie następnie ustawiony na 0 lub 1, aby uzyskać parzystą lub nieparzystą sumę „1”.

Należy zauważyć, że sprawdzanie parzystości może wykryć błąd tylko wtedy, gdy nieparzysta liczba bitów zostanie odebrana lub upuszczona w znaku transmisji podczas transmisji, jeśli na przykład dwie 1 są uszkodzone do 0, kontrola parzystości nie wykryje błędu.

Jeśli określono Brak kontroli parzystości, bit parzystości nie jest przesyłany i nie można wykonać kontroli parzystości. Ponadto, jeśli określono Brak kontroli parzystości i wybrano jeden bit stopu, znak transmisji jest skutecznie skracany o jeden bit.

3.5.2 Sprawdzanie CRC

Bajty kontroli błędów komunikatów protokołu MODBUS zawierają wartość cyklicznej kontroli nadmiarowej (CRC), która jest używana do sprawdzania zawartości całego komunikatu. Aby zapewnić zgodność z protokołem MODBUS, bajty kontroli błędów muszą być zawsze obecne, nie ma możliwości wyłączenia tego.

Bajty kontroli błędów reprezentują 16-bitową wartość binarną obliczoną przez urządzenie nadawcze. Urządzenie odbierające musi ponownie obliczyć CRC podczas odbierania komunikatu i porównać obliczoną wartość z wartością odebraną w bajtach kontroli błędów. Jeśli te dwie wartości nie są równe, komunikat powinien zostać odrzucony.

Obliczanie sprawdzania błędów rozpoczyna się przez wstępne załadowanie 16-bitowego rejestru do wszystkich 1 (tj. Hex (FFFF)), a każdy kolejny 8-bitowy bajt komunikatu jest stosowany do bieżącej zawartości rejestru. Uwaga: tylko osiem bitów danych w każdym znaku transmisji jest wykorzystywanych do generowania CRC, bity startu, bity stopu i bit parzystości, jeśli jest używany, nie są uwzględniane w bajtach kontroli błędów.

Podczas generowania bajtów kontroli błędów każdy 8-bitowy bajt komunikatu jest rozłączony z dolną połową 16-bitowego rejestru. Rejestr jest następnie przesuwany osiem razy w kierunku najmniej znaczącego bitu (LSB), z zerem w pozycji najbardziej znaczącego bitu (MSB). Po każdej zmianie LSB przed przesunięciem jest wyodrębniany i badany. Jeśli LSB był 1, w rejestrze wystąpiła wtedy alternatywa rozłączna z ustaloną, stałą wartością. Jeśli LSB ma wartość 0, nie ma alternatywy rozłącznej.

Proces ten powtarza się aż do wykonania wszystkich ośmiu [przesunięć. Po ostatnim przesunięciu następny 8-bitowy bajt komunikatu jest alternatywą rozłączną z dolną połową 16-bitowego rejestru, a proces się powtarza. Końcowa zawartość rejestru, po zastosowaniu wszystkich bajtów komunikatu, jest wartością sprawdzenia błędu. W następującym pseudo kodzie „Słowo błędu” to 16-bitowa wartość reprezentująca wartości sprawdzania błędów.

```
BEGIN
  Error Word = Hex (FFFF)
  FOR Each byte in message
    Error Word = Error Word XOR byte in message
  FOR Each bit in byte
    LSB = Error Word AND Hex (0001)
    IF LSB = 1 THEN Error Word = Error Word - 1
    Error Word = Error Word / 2
    IF LSB = 1 THEN Error Word = Error Word XOR Hex (A001)
  NEXT bit in byte
NEXT Byte in message
END
```

3.6 Kody funkcji

Część kodu funkcji komunikatu protokołu MODBUS określa działanie, które ma podjąć urządzenie slave. Miernik cyfrowy NR10 obsługuje następujące kody funkcji

Kod	Nazwa protokołu MODBUS	Opis
03	Odczyt rejestrów do odczytu	Odczyt zawartość lokalizacji do odczytu / zapisu (odniesienia 4X)
04	Odczyt rejestrów wejściowych	Odczyt zawartość lokalizacji tylko do odczytu (odniesienia 3X)
08	Diagnostyka	Obsługiwana jest tylko podfunkcją zero. Odpowiedzią jest element danych zapytania bez zmian.
15	Wstępne ustawienie wielu rejestrów	Ustawianie zawartości lokalizacji do odczytu/zapisu (odniesienia 4X)

3.7 Format zmiennoprzecinkowy IEEE

Protokół MODBUS definiuje 16-bitowe „Rejestry” dla zmiennych danych. 16-bitowa liczba byłaby zbyt restrykcyjna, na przykład dla parametrów energii, ponieważ maksymalny zakres 16-bitowej liczby wynosi 65535.

Istnieje jednak szereg metod, które zostały przyjęte w celu przewyższenia tego ograniczenia. Mierniki cyfrowe NR10 wykorzystują dwa kolejne rejestry do reprezentowania liczby zmiennoprzecinkowej, skutecznie rozszerzając zakres do +/- 1x1037.

Wartości odczytywane przez mierniki cyfrowe NR10 mogą być używane bezpośrednio bez konieczności „skalowania” wartości, np. jednostkami dla parametrów napięcia są wolty, jednostkami dla parametrów mocy są waty itp.

Czym jest liczba zmiennoprzecinkowa?

Liczba zmiennoprzecinkowa jest liczbą składającą się z dwóch części, mantysy i wykładnika i jest zapisana w formie 1,234 x 105. Mantysa (w tym przykładzie 1,234) musi mieć punkt dziesiętny przesunięty w prawo, z liczbą miejsc określoną przez wykładnik (w tym przykładzie 5 miejsc), tj. 1,234x 105 = 123400. Jeśli wykładnik jest ujemny, kropka dziesiętna jest przesuwana w lewo.

Czym jest liczba zmiennoprzecinkowa formatu IEEE 754?

Liczba zmiennoprzecinkowa IEEE 754 jest binarnym odpowiednikiem liczby zmiennoprzecinkowej punktu dziesiętnego wskazanej powyżej. Główna różnica polega na tym, że najbardziej znaczący bit mantysy jest zawsze ustawiony na 1 i dlatego nie jest potrzebny do reprezentacji liczby. Proces, w którym najbardziej znaczący bit jest ustawiony na 1, nazywany jest normalizacją, mantysa jest zatem określana jako „mantysa normalna”. Podczas normalizacji bity w mantysie są przesuwane w lewo, podczas gdy wykładnik jest zmniejszany, aż najbardziej znaczący bit mantysy wynosi jeden. W szczególnym przypadku, gdy liczba wynosi zero, mantysa i wykładnik są równe zero.

Bity w formacie IEEE 754 mają następujące znaczenie:

Data Hi Reg, Hi Byte.	Data Hi Reg, Lo Byte.	Data Lo Reg, Hi Byte.	Data Lo Reg, Lo Byte.
SEEE EEEE	EMMM MMMM	MMMM MMMM	MMMM MMMM

Gdzie:

S reprezentuje bit znaku, gdzie wartość 1 jest ujemna a wartość 0 dodatnia

E jest wykładnikiem 8-bitowym z przesunięciem 127, tj. wykładnikiem zera jest wartość 127, wykładnikiem 1 jest wartość 128 itd.

M to 23-bitowa normalna mantysa. 24. bit ma zawsze wartość 1 i dlatego nie jest zapisywany.

Używając powyższego formatu liczba zmiennoprzecinkowa 240,5 jest reprezentowana jako 43708000 hex:

Data Hi Reg, Hi Byte.	Data Hi Reg, Lo Byte.	Data Lo Reg, Hi Byte.	Data Lo Reg, Lo Byte.
43	70	80	00

Poniższy przykład pokazuje sposób konwersji liczby zmiennoprzecinkowej IEEE 754 z postaci szesnastkowej na postać dziesiętną. W tym przykładzie użyta zostanie wartość 240,5 podana powyżej.

Należy zauważyć, że reprezentacja zapisu liczby zmiennoprzecinkowej nie jest formatem intuicyjnym. Aby przekonwertować tę wartość na dziesiętną, bity powinny być rozdzielone w sposób podany powyżej w tabeli formatowania zapisu liczb zmiennoprzecinkowych.

Na przykład:

Data Hi Reg, Hi Byte.	Data Hi Reg, Lo Byte.	Data Lo Reg, Hi Byte.	Data Lo Reg, Lo Byte.
0100 0011	0111 0000	1000 0000	0000 0000

Na tej podstawie można określić następujące informacje.

- Bit znaku wynosi 0, wskazując liczbę dodatnią.
- Wartość wykładnika wynosi 10000110 binarnie lub 134 dziesiętnie. Odejmowanie 127 od 134 daje 7, co jest rzeczywistym wykładnikiem.
- Mantysa jest liczbą binarną 111000010000000000000000

Na lewo od mantysy znajduje się domyślny punkt binarny, który zawsze poprzedzony jest przez 1. Ten bit nie jest przechowywany w szesnastkowej reprezentacji liczby zmiennoprzecinkowej. Dodanie 1 i punktu binarnego na początku mantysy daje następujący wynik:

1,111000010000000000000000

Teraz dostosowujemy mantysę do wykładnika. Ujemny wykładnik przesuwa punkt binarny w lewo. Dodatni wykładnik przesuwa punkt binarny w prawo. Ponieważ wykładnik wynosi 7, mantysa jest dostosowywana w następujący sposób:

11110000,100000000000000000

Otrzymujemy binarną liczbę zmiennoprzecinkową. Bity binarne, które znajdują się na lewo od punktu binarnego reprezentują wykładnik 2 odpowiadający ich pozycji. Na przykład: 11110000 reprezentuje $(1 \times 2^7) + (1 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (0 \times 2^0) = 240$.

Bity binarne, które znajdują się na prawo od punktu binarnego, również reprezentują wykładnik 2 odpowiadający ich pozycji. Ponieważ cyfry znajdują się na prawo od punktu binarnego, wykładniki są ujemne. Na przykład: $,100$ reprezentuje $(1 \times 2^{-1}) + (0 \times 2^{-2}) + (0 \times 2^{-3}) + \dots$ co równa się $0,5$.

Dodanie tych dwóch liczb i odniesienie do bitu znaku daje liczbę +240,5.

Dla każdej żądanej wartości zmiennoprzecinkowej należy zażądać dwóch rejestrów protokołu MODBUS (cztery bajty). Otrzymane polecenie i znaczenie tych czterech bajtów dla NR10 pokazano poniżej:

Data Hi Reg, Hi Byte.	Data Hi Reg, Lo Byte.	Data Lo Reg, Hi Byte.	Data Lo Reg, Lo Byte.
-----------------------	-----------------------	-----------------------	-----------------------

3.8 Obsługiwane polecenia protokołu MODBUS

Mierniki cyfrowe NR10 obsługują komendy protokołu MODBUS RTU „Odczyt rejestrów wejściowych” (rejstry 3X), „Odczyt rejestrów do zapisu/odczytu” (rejstry 4X) oraz „Wstępne ustawienie wielu rejestrów” (rejstry zapisu 4X). Wszystkie wartości zapisane i zwrócone są w formacie zmiennoprzecinkowym IEEE 754 z najbardziej znaczącym rejestrem jako pierwszym.

3.8.1 Odczyt rejestrów wejściowych

Kod 04 protokołu MODBUS odczytuje zawartość rejestrów 3X.

Przykład

Następujące zapytanie zażąda „Volts 1” z urządzenia o adresie węzła 1:

Nazwa pola	Przykład (Hex)
Adres slave	01
Funkcja	04
Starting Address High	00
Starting Address Low	00
Number of Points High	00
Number of Points Low	02
Error Check Low	71
Error Check High	CB

Uwaga: Dane muszą być żądane w parach rejestrów, tzn. „Adres początkowy” i „Liczba punktów” muszą być liczbami parzystymi, aby zażądać zmiennej zmiennoprzecinkowej. Jeśli „Adres początkowy” lub „Liczba punktów” jest nieparzysta, zapytanie znajdzie się w środku zmiennej zmiennoprzecinkowej, a wynikiem będzie komunikat o błędzie. Odpowiedź zwraca zawartość Volts 1 jako 230,2. Zobacz również „Odpowiedź na wyjątek” poniżej.

Nazwa pola	Przykład (Hex)
Adres slave	01
Funkcja	04
Byte Count	04
Data, High Reg, High Byte	43
Data, High Reg, Low Byte	66
Data, Low Reg, High Byte	33
Data, Low Reg, Low Byte	34
Error Check Low	1B
Error Check High	38

3.9 Rejestry do zapisu i odczytu

3.9.1 Odczyt rejestrów do odczytu

Kod 03 protokołu MODBUS odczytuje zawartość rejestrów 4X.

Przykład

Następujące zapytanie zażąda dominującego „Czas Demand”:

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	03
Starting Address High	00
Starting Address Low	00
Number of Points High	00
Number of Points Low	02
Error Check Low	C4
Error Check High	0B

Uwaga: Dane muszą być żądane w parach rejestrów, tzn. „Adres początkowy” i „Liczba punktów” muszą być liczbami parzystymi, aby zażądać zmiennej zmiennoprzecinkowej. Jeśli „Adres początkowy” lub „Liczba punktów” jest nieparzysta, zapytanie znajdzie się w środku zmiennej zmiennoprzecinkowej, a wynikiem będzie komunikat o błędzie.

Odpowiedź zwraca zawartość Czas Demand jako 1; zobacz również „Odpowiedź na wyjątek” poniżej.

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	03
Byte Count	04
Data, High Reg, High Byte	3F
Data, High Reg, Low Byte	80
Data, Low Reg, High Byte	00
Data, Low Reg, Low Byte	00
Error Check Low	F7
Error Check High	CF

3.9.2 Zapis rejestrów do zapisu i odczytu

Kod 10 protokołu MODBUS (16 miejsc dziesiętnych) zapisuje zawartość rejestrów 4X.

Przykład

Następujące zapytanie ustawi Okres Demand na 60, co skutecznie zresetuje Czas Demand:

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	10
Starting Address High	00
Starting Address Low	02
Number of Registers High	00
Number of Registers Low	02
Byte Count	04
Data, High Reg, High Byte	42
Data, High Reg, Low Byte	70
Data, Low Reg, High Byte	00
Data, Low Reg, Low Byte	00
Error Check Low	67
Error Check High	D5

Uwaga: Dane muszą być zapisane w parach rejestrów, tzn. „Adres początkowy” i „Liczba punktów” muszą być liczbami parzystymi, aby zapisać zmienną zmiennoprzecinkową. Jeśli „Adres początkowy” lub „Liczba punktów” jest nieparzysta, zapytanie znajdzie się w środku zmiennej zmiennoprzecinkowej, a wynikiem będzie komunikat o błędzie. Zwykle na jedno zapytanie można zapisać tylko jedną wartość zmiennoprzecinkową.

Dana odpowiedź wskazuje, że zapis zakończył się powodzeniem. Zobacz też „Odpowiedź na wyjątek” poniżej.

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	10
Starting Address High	00
Starting Address Low	02
Number of Registers High	00
Number of Registers Low	02
Error Check Low	E0
Error Check High	08

3.10 Odpowiedź na wyjątek

W powyższym przykładzie „Rejestr do zapisu/odczytu”, gdyby urządzenie slave nie obsługiwało tej funkcji, wysłałoby odpowiedź na wyjątek, jak pokazano poniżej. Kod funkcji wyjątku jest oryginalnym kodem funkcji zapytania z ustawieniem MSB, tj. ma on wartość 80 hex logicznie zsumowaną. Kod wyjątku wskazuje powód wyjątku. Urządzenie slave nie odpowie w ogóle, jeśli wystąpi błąd z parzystością lub CRC zapytania. Jeśli jednak urządzenie slave nie może przetworzyć zapytania, odpowie na wyjątek. W tym przypadku kod 01 oznacza, że żądana funkcja nie jest obsługiwana przez urządzenie slave.

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	10 OR 80 = 90
Exception Code	01
Error Check Low	8D
Error Check High	C0

3.11 Kody wyjątków

3.11.1 Tabela kodów wyjątków

Mierniki cyfrowe obsługują następujące kody funkcji:

Exception Code	Nazwa protokołu MODBUS	Opis
01	Illegal Function	Kod funkcji nie jest obsługiwany przez miernik
02	Illegal Data Address	Próba adresowanie, aby uzyskać dostęp do nieprawidłowego adresu lub próba odczytania lub zapisania części wartości
03	Illegal Data Value	Próba ustawienia zmiennej zmiennoprzecinkowej na niewłaściwą
05	Slave Device Failure	Wystąpił błąd, gdy urządzenie próbowało zapisać aktualizację do konfiguracji

3.12 Diagnostyka

Kod 08 protokołu MODBUS zapewnia szereg podfunkcji diagnostycznych. Mierniki cyfrowe NR10 obsługują tylko funkcję „Zwrot danych zapytania” (podfunkcja 0).

Przykład

Następujące zapytanie wyśle zapytanie diagnostyczne „zwróć dane zapytania” z elementami danych ustawionymi na Hex (AA) i Hex (55) i oczekuje, że zostaną one zwrócone w odpowiedzi:

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	08
Sub-Function High	00
Sub-Function Low	00
Data Byte 1	AA
Data Byte 2	55
Error Check Low	5E
Error Check High	94

Uwaga: Za pomocą tej funkcji należy wysłać dokładnie jeden rejestr danych (dwa bajty).

Następująca odpowiedź wskazuje poprawną odpowiedź na zapytanie, tj. takie same bajty jak zapytanie.

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	08
Sub-Function High	00
Sub-Function Low	00
Data Byte 1	AA
Data Byte 2	55
Error Check Low	5E
Error Check High	94



LUMEL

LUMEL S.A.

ul. Słubicka 4, 65-127 Zielona Góra, Poland
tel.: +48 68 45 75 100, fax +48 68 45 75 508
www.lumel.com.pl

Informacja techniczna:

tel.: (68) 45 75 140, 45 75 141, 45 75 142, 45 75 145, 45 75 146
e-mail: sprzedaz@lumel.com.pl

Realizacja zamówień:

tel.: (68) 45 75 150, 45 75 151, 45 75 152, 45 75 153, 45 75 154, 45 75 155
fax.: (68) 32 55 650

Pracownia systemów automatyki:

tel.: (68) 45 75 145, 45 75 145

Wzorcowanie:

tel.: (68) 45 75 163
e-mail: laboratorium@lumel.com.pl