

## NMID30-1

Miernik energii sieci 1- i 3-faz. 1/5 A  
(Z CERTYFIKATEM MID)



## Instrukcja Modbus

### **Cechy**

- Certyfikat MID B+D
- Numer certyfikatu 0120/SGS0209
- Klasa B (kWh) Dyrektywa WE 2004/22/WE
- Certyfikowany dla wersji jedno i trójfazowej
- Certyfikowany dla kWh pobieranej / importowanej

## **I. Implementacja protokołu NMID30-1**

- 1.1 Przegląd protokołu Modbus
- 1.2 Rejestr wejściowy
  - 1.2.1 Rejestry wejściowe NMID30-1
- 1.3 Rejestry protokołu Modbus i konfiguracja licznika 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

## I.1 Przegląd protokołu Modbus

Ten rozdział zawiera podstawowe informacje dotyczące połączenia inteligentnego licznika z siecią protokołu Modbus. Jeśli wymagane są dodatkowe informacje lub szczegóły dotyczące inteligentnej implementacji, należy zapoznać się z rozdz. 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 idealnie 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 określane jako „Bias linii” i jest krytyczne w aplikacjach RS-485. Różnica większa niż 0,3 V jest ogólnie akceptowana jako ważna, ale może wynosić nawet 0,7 V, w zależności od systemu. 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 NMID30-1 oferuje opcję komunikacji RS485 do bezpośredniego połączenia z SCADA lub innymi systemami komunikacyjnymi przy użyciu protokołu Modbus RTU. Protokół Modbus ustala format zapytania 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. Jeśli wystąpi błąd w odbiorze wiadomości, NMID30-1 nie odpowie. Jeśli NMID30-1 nie może wykonać żądanej akcji, skonstruuje komunikat o błędzie i wyśle go jako odpowiedź.

Interfejsem elektrycznym jest 2-przewodowy RS485, poprzez 2 zaciski śrubowe. Połączenie należy wykonać za pomocą kabla ekranowanego typu skrętką (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. Kabel powinien być zakończony na każdym końcu rezystorem 120 omów (1/4 W min.). W sieci RS485 maksymalna dozwolona długość wynosi 1200 metrów (3900 stóp). Można podłączyć maksymalnie 32 węzły elektryczne, w tym sterownik. Adres każdego NMID30-1 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-bit 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 wysyłany 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 NMID30-1 są przesyłane jako 32-bitowe liczby zmiennoprzecinkowe IEEE754 (wejście i wyjście), dlatego każda wartość miernika NMID30-1 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 NMID30-1 zwraca komunikat wyjątku protokołu Modbus. Jednakże, dla kompatybilności z niektórymi systemami SCADA, NMID30-1 odpowie na każdy pojedynczy rejestr wejściowy lub rejestr do odczytu/zapisu odczytany z wartością określoną dla typu instrumentu.

NMID30-1 może przesłać maksymalnie 40 wartości w jednej transakcji; dlatego maksymalna liczba rejestrów, których może dotyczyć zapytanie wynosi 80. Przekroczenie tego limitu powoduje, że NMID30-1 wygeneruje odpowiedzi o wyjątku.

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

## I.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. 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 rejestrów	=0002
	Amps 2	Adres początkowy	=0008
		Liczba rejestrów	=0002

Każde żądanie danych musi być ograniczone do 40 parametrów lub mniej. Przekroczenie limitu 40 parametrów spowoduje wygenerowanie kodu wyjątku ModBus.

I.2.1 Rejestry wejściowe NMID30-I

Adres (Rejestr)	Numer parametru	Parametr rejestru wejściowego NMID30-I		Szesnastkowy adres początkowy protokołu Modbus		3Ø	3Ø	1Ø
		Opis	Jednostki	Bajt Górny	Bajt Dolny	4W	3W	2W
30001	1	Napięcie między fazą 1 a neutralnym	Volt	00	00	√	X	√
30003	2	Napięcie między fazą 2 a neutralnym	Volt	00	02	√	X	X
30005	3	Napięcie między fazą 3 a neutralnym	Volt	00	04	√	X	X
30007	4	Faza 1 prąd	Ampery	00	06	√	√	√
30009	5	Faza 2 prąd	Ampery	00	08	√	√	X
30011	6	Faza 3 prąd	Ampery	00	0A	√	√	X
30013	7	Faza 1 moc	waty	00	0C	√	X	√
30015	8	Faza 2 moc	waty	00	0E	√	X	√
30017	9	Faza 3 moc	waty	00	10	√	X	X
30019	10	Faza 1 Volt Amp	VoltAmp	00	12	√	X	√
30021	11	Faza 2 Volt Amp	VoltAmp	00	14	√	X	X
30023	12	Faza 3 Volt Amp	VoltAmp	00	16	√	X	X
30025	13	Faza 1 Volt Amp bierna	VAr	00	18	√	X	√
30027	14	Faza 2 Volt Amp bierna	VAr	00	1A	√	X	X
30029	15	Faza 3 Volt Amp bierna	VAr	00	1C	√	X	X
30031	16	Faza 1 wsp. mocy (1)	Brak	00	1E	√	X	√
30033	17	Faza 2 wsp. mocy (1)	Brak	00	20	√	X	X
30035	18	Faza 3 wsp. mocy (1)	Brak	00	22	√	X	X
30037	19	Faza 1 kąt fazowy	Stopnie	00	24	√	X	√
30039	20	Faza 2 kąt fazowy	Stopnie	00	26	√	X	X
30041	21	Faza 3 kąt fazowy	Stopnie	00	28	√	X	X
30043	22	Średnie napięcie między fazą a neutralnym	Volt	00	2A	√	X	X
30047	24	Średni prąd linii	Ampery	00	2E	√	√	√
30049	25	Suma prądów linii	Ampery	00	30	√	√	√
30053	27	Całkowita moc układu	waty	00	34	√	√	√
30057	29	Całkowita volt amp układu	VA	00	38	√	√	√
30061	31	Całkowita VAr układu	VAr	00	3C	√	√	√
30063	32	Całkowity wsp. mocy układu (1)	Brak	00	3E	√	√	√
30067	34	Całkowity kąt fazowy układu	Stopnie	00	42	√	√	√
30071	36	Częstotliwość napięć zasilania	Hz	00	46	√	√	√
30073	37	Pobierana Wh od ostatniego resetu (2).	kWh/MWh	00	48	√	√	√
30075	38	Oddawana Wh od ostatniego resetu (2).	kWh/MWh	00	4A	√	√	√
30077	39	Pobierana VArh od ostatniego resetu (2).	kVArh/MVArh	00	4C	√	√	√
30079	40	Oddawana VArh od ostatniego resetu (2).	kVArh/MVArh	00	4E	√	√	√
30081	41	VAh od ostatniego resetu (2).	kVAh/MVAh	00	50	√	√	√
30083	42	Ah od ostatniego resetu (3).	Ah/kAh	00	52	√	√	√
30085	43	Całkowita moc demand układu (4)	waty	00	54	√	√	√
30087	44	Maksymalna całkowita moc demand układu (4)	waty	00	56	√	√	√
30101	51	Całkowita VA demand układu	VA	00	64	√	√	√
30103	52	Maksymalna całkowita VA demand układu	VA	00	66	√	√	√
30105	53	Neutralny prąd demand	Ampery	00	68	√	X	X
30107	54	Maksymalny neutralny prąd demand	Ampery	00	6A	√	X	X
30201	101	Napięcie między L1 a L2	Volt	00	C8	√	√	X
30203	102	Napięcie między L2 a L3	Volt	00	CA	√	√	X
30205	103	Napięcie między L3 a L1	Volt	00	CC	√	√	X
30207	104	Średnie napięcie liniowe	Volt	00	CE	√	√	X
30225	113	Prąd neutralny	Ampery	00	E0	√	X	X
30235	118	Faza 1 L/N THD napięcia	%	00	EA	√	X	√
30237	119	Faza 2 L/N THD napięcia	%	00	EC	√	X	X
30239	120	Faza 3 L/N THD napięcia	%	00	EE	√	X	X
30241	121	Faza 1 THD prądu	%	00	F0	√	√	√
30243	122	Faza 2 THD prądu	%	00	F2	√	√	X
30245	123	Faza 3 THD prądu	%	00	F4	√	√	X

30249	125	THD średniego napięcia między fazą a neutralnym	%	00	F8	√	X	√
30251	126	THD średniego prądu linii	%	00	FA	√	√	√
30255	128	Całkowity wsp. mocy układu (5)	Stopnie	00	FE	√	√	√
30259	130	Faza 1 prąd demand	Ampery	01	02	√	√	√
30261	131	Faza 2 prąd demand	Ampery	01	04	√	√	X
30263	132	Faza 3 prąd demand	Ampery	01	06	√	√	X
30265	133	Faza 1 maksymalny prąd demand	Ampery	01	08	√	√	√
30267	134	Faza 2 maksymalny prąd demand	Ampery	01	0A	√	√	X
30269	135	Faza 3 maksymalny prąd demand	Ampery	01	0C	√	√	X
30335	168	THD napięcia między L1 a L2	%	01	4E	√	√	X
30337	169	THD napięcia między L2 a L3	%	01	50	√	√	X
30339	170	THD napięcia między L3 a L1	%	01	52	√	√	X
30341	171	THD średniego napięcia liniowego	%	01	54	√	√	X
30343	172	Energia czynna całkowita	kWh	01	56	√	√	√
30345	173	Energia bierna całkowita	kVarh	01	58	√	√	√

**Uwagi:**

1. Znak współczynnika mocy jest ustawiany tak aby wskazywać naturę obciążenia. Dodatni dla pojemnościowego i ujemny dla indukcyjnego.
2. Dostępna jest opcja użytkownika wyboru k lub M jako prefiksu energii.
3. Ta sama opcja użytkownika, jak w punkcie 2 powyżej, umożliwia dodanie przedrostka braku lub k dla Amperogodzin
4. Obliczenie sumy mocy demand dotyczy wyłącznie mocy pobieranej
5. Ujemny całkowity współczynnik mocy układu jest odwróconą wersją parametru 32, wielkość jest taka sama jak parametr 32.
6. Dostępna jest opcja użytkownika, aby wybrać Brak, k lub M dla prefiksu energii.

**I.3 Rejestry protokołu Modbus i konfiguracja licznika 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.

**I.3.1 Parametry rejestru protokołu MODBUS**

Rejestr adresów	Numer parametru	Parametr	Adres początkowy protokołu Modbus Hex		Prawidłowy zakres	Tryb
			Bajt Górny	Bajt Dolny		
40001	1	Demand Time	00	00	Odczyt minut do pierwszego obliczenia demand. Gdy czas Demand osiągnie okres Demand, wtedy wartości Demand są obowiązujące.	Ro
40003	2	Demand Period	00	02	Zapis okresu Demand: 0, 5, 8, 10, 15, 20, 30 lub 60 minut, domyślnie 60. Ustawienie okresu na 0 spowoduje, że demand pokaże bieżącą wartość parametru, a maks. demand pokaże maksymalną wartość parametru od ostatniego resetu demand.	r/w
40007	4	System Volts	00	06	Odczyt napięcia systemu, VLL dla 3F3P, VLN dla innych.	ro
40009	5	System Current	00	06	Zapis prądu układu, ograniczony od 1 do 9999A. Wymaga hasła, patrz parametr 13	ro
40011	6	System Type	00	08	Zapis typu układu: 3f4p = 3, 3f3p = 2 i 1f2p= 1 Wymaga hasła, patrz parametr 13	r/wp
40013	7	Relay Pulse Width	00	0A	Zapis okresu pracy przekaźnika w milisekundach 60, 100 lub 200, domyślnie 200.	r/wp

40015	8	Password Lock	00	OE	Zapis dowolnej wartości blokady hasła rejestrów chronionych Odczyt statusu blokady hasła: 0 = zablokowane. 1 = odblokowane. Odczyt zresetuje także limit czasu hasła do jednej minuty.	r/w
40019	10	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. Wymaga ponownego uruchomienia, aby było skuteczne.	r/w
40021	11	Network Node	00	14	Zapis adresu węzła portu sieciowego: 1 do 247 dla protokołu MODBUS, domyślnie 1. Wymaga ponownego uruchomienia, aby było skuteczne. Należy zauważyć, że adresy węzłów MODBUS Protocol i Johnson Controls można zmienić za pomocą menu konfiguracji wyświetlacza	r/w
40023	12	Pulse Divisor	00	16	Zapis indeks stałej impulsowej: $n = 2$ do $6$ w $Wh / 10^n$ , domyślnie 3.	r/w
40025	13	Password	00	18	Zapis hasła dostępu do chronionych rejestrów. Odczyt zero. Odczyt zresetuje także limit czasu hasła do jednej minuty. Domyślne hasło to 0000.	r/w
40029	15	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. Wymaga ponownego uruchomienia, aby było skuteczne. Typ wartości: float (32 bity)	r/w
40031	16	Energy Units Prefix	00	1E	Zapis prefiksu jednostki dla wartości energii wyjściowej. 0 = k, np. kWh, domyślnie. Ale Ah dla amperogodzin. 1 = M, np. MWh. Ale kAh dla amperogodzin.	r/w
40037	19	System Power	00	24	Odczyt całkowitej mocy układu, np. dla 3f4p daje napięcie układu x prąd układu x 3.	ro
40087	44	Pulse 1 Energy Type	00	56	Zapis parametru wejścia protokołu MODBUS dla wyjścia impulsowego 1: 1: energia czynna pobierana 2: energia czynna całkowita, 4: energia czynna oddawana 5: energia bierna pobierana, 6: energia bierna całkowita 8: energia bierna oddawana Typ wartości: float (32 bity)	r/w
40089	45	Pulse 2 Energy Type	00	58	Zapis parametru wejścia protokołu MODBUS dla wyjścia impulsowego 2: 1: energia czynna pobierana 2: energia czynna całkowita, 4: energia czynna oddawana 5: energia bierna pobierana, 6: energia bierna całkowita 8: energia bierna oddawana Typ wartości: float (32 bity)	r/w
461457	30729	Reset	F0	10	Reset wartości maksymalnych Demand Typ wartości: Hex (16 bit)	wo
464513	464513	Serial Number	FC	00	Numer seryjny Typ wartości: unsigned int (32 bit) Rejestr tylko do odczytu	ro

**Kolejność rejestrów** NMID30-1 otrzymuje lub wysyła liczby zmiennoprzecinkowe: - w kolejności bajtów 3210 - najstarszy jest wysyłany pierwszy.

Jest całkowicie możliwe, aby zmienić konfigurację NMID30-1 za pomocą protokołu Modbus master ogólnego przeznaczenia, ale często łatwiej jest korzystać z wyświetlacza NMID30-1, zwłaszcza w celu uzyskania dostępu chronionego hasłem.

#### Hasło

Niektóre z opisanych powyżej parametrów są chronione hasłem i dlatego wymagają wprowadzenia hasła w rejestrze haseł przed ich zmianą. Domyślne hasło to 1000. Po wprowadzeniu hasła przekroczy ono limit czasu w ciągu jednej minuty, chyba że zostanie odczytany rejestr hasła lub rejestr blokady hasła w celu zresetowania timera limitu czasu. Po wprowadzeniu wymaganych zmian w chronionych parametrach należy ponownie zastosować blokadę hasła poprzez

- a) Umożliwienie przekroczenia limitu czasu lub
- b) Zapis dowolnej wartości w rejestrze blokady hasła lub
- c) Włączenie i wyłączenie zasilania instrumentu.

## 2. Informacje ogólne o RS485

Poniższe informacje odnoszą się do NMID30-1 i zostały dołączone, aby pomóc w sytuacji gdy zaimplementowano sieć mieszaną. RS485 lub EIA (Electronic Industries Association) RS485 to zbalansowany, półdupleksowy 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 omów
Ograniczenie prądu zwarcia wyjścia sterownika	150 mA do Gnd, 250 mA do 12 V
Minimalna oporność wejściowa odbiornika	12 kohms
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

Półdupleks 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.

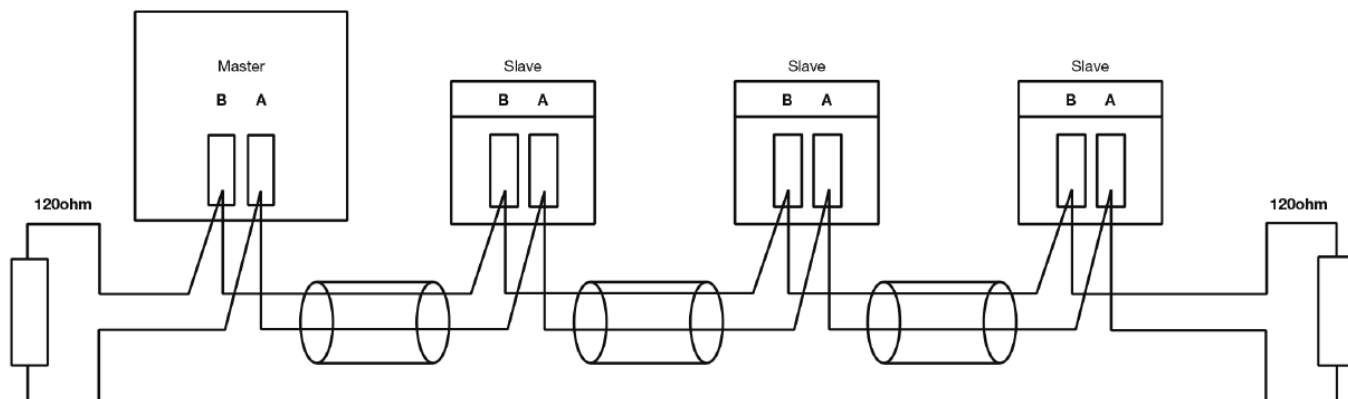
W sieci 485 „master” rozpocznie „rozmowę” z „zapytaniem” zaadresowanym do określonego „slave”, a następnie „master” będzie nasłuchiwał odpowiedzi „slave”. Jeśli „slave” nie odpowie w określonym czasie (ustawionym przez oprogramowanie sterujące w „master”), „master” zrezygnuje z „rozmowy”.

### 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. W przypadku dłuższych tras kablowych lub w otoczeniu zaszumionym, aby uzyskać optymalną wydajność, konieczne może być użycie kabla 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 omów. Kabel powinien być zakończony na każdym końcu rezystorem 120 ohm, ćwierć watomym (lub większym). Uwaga: Diagram pokazuje tylko topologię okablowania. Należy zawsze postępować zgodnie z identyfikacją zacisku na etykiecie produktu miernika Eastron Digital.



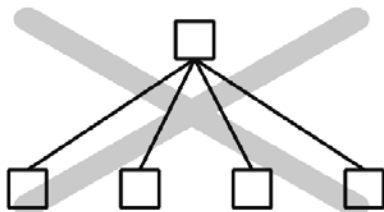


Do każdego zacisku nie mogą być podłączone więcej niż dwa przewody, co zapewnia, że używana jest konfiguracja „Daisy Chain” (szeregowo) lub „straight line” (bezpośrednio). Podłączenie typu „Star” (na gwiazdę) lub sieć z „Stubs (Tees)” (otwarte) nie jest zalecana, ponieważ odbicia w obrębie kabla mogą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

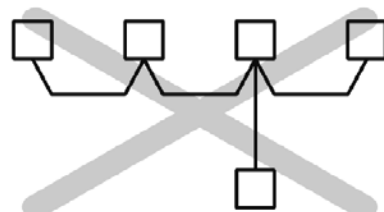
Podłączenie szeregowe (poprawne)



Podłączenie w gwiazdę (NIEWŁAŚCIWE)

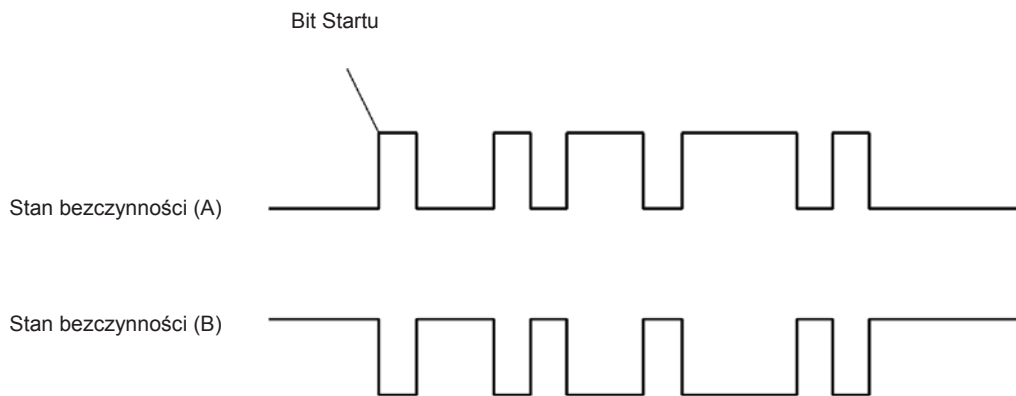


Podłączenie otwarte (NIEWŁAŚCIWE)



### 2.3 Zaciski A i B

Połączenia A i B z miernikiem cyfrowym NMID30-1 mogą być identyfikowane przez obecne w nich sygnały gdy istnieje aktywność na magistrali RS485:



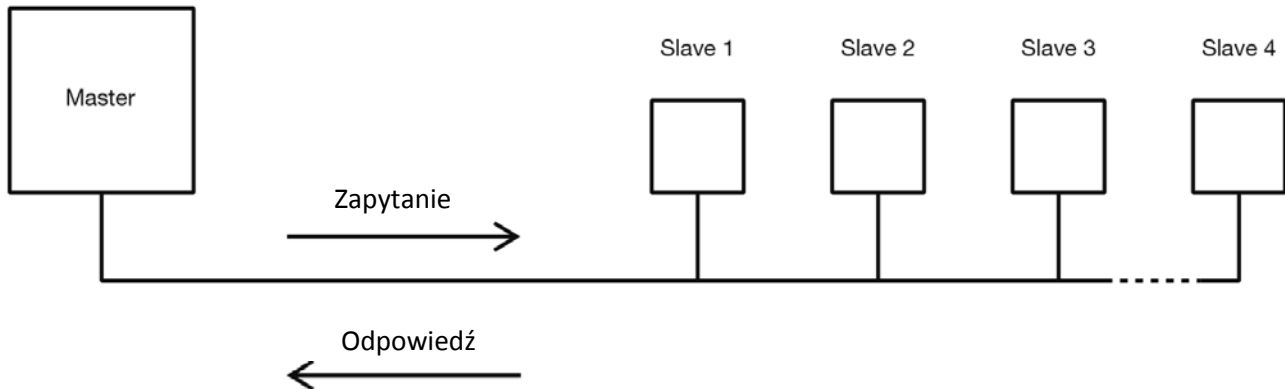
### 2.4 Rozwiązywanie problemów

- Należy zacząć od prostej sieci, jednego master i jednego slave. Dzięki miernikowi cyfrowemu NMID30-1 można to łatwo osiągnąć, ponieważ sieć może pozostać nienaruszona, podczas gdy poszczególne instrumenty są odłączane poprzez usunięcie 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). Wydaje się, że wiele pakietów opartych na komputerach PC nie działa dobrze, gdy odbierają one odbicie przesyłanego przez siebie komunikatu. Uważa się, że SpecView i PCView (oprogramowanie PC) z konwerterem RS232 na RS485 są wyposażone w tę funkcję.
- Potwierdź, że adres instrumentu jest taki sam jak adres oczekiwany przez „master”.
- Jeśli „sieć” działa z jednym instrumentem, ale nie więcej niż z jednym, sprawdź, czy każdy instrument ma unikalny adres.
- Każde żądanie danych musi być ograniczone do 40 parametrów lub mniej. Naruszenie tego wymogu wpłynie na wydajność instrumentu i może skutkować czasem reakcji przekraczającym specyfikację.
- Sprawdź, czy tryb protokołu MODBUS (RTU lub ASCII) i parametry szeregowe (szybkość transmisji, liczba bitów danych, liczba bitów stopu i parzystość) są takie same dla wszystkich urządzeń w sieci.
- Sprawdź, czy „master” żąda zmiennych zmiennoprzecinkowych (par rejestrów umieszczonych na granicach liczb zmiennoprzecinkowych) i nie rozdziela zmiennych zmiennoprzecinkowych.
- Sprawdź, czy kolejność bajtów liczb zmiennoprzecinkowych oczekiwana przez „master” jest taka sama, jak w przypadku produktów Eastron Digital. (Pakiety PCView i Citect mogą korzystać z wielu formatów, w tym obsługiwanych przez miernik Eastron Digital).
- Jeśli to możliwe, uzyskaj drugi konwerter RS232 na RS485 i podłącz go między magistralą RS485 a dodatkowym komputerem wyposażonym w pakiet oprogramowania, który może wyświetlać dane na



### 3. Informacje ogólne o protokole MODBUS

Komunikacja w sieci protokołu MODBUS jest inicjowana (uruchamiana) przez „Master” wysyłającego zapytanie do „Slave”. „Slave”, które stale monitoruje sieć pod kątem adresowanych do niego zapytań, odpowie, wykonując żądaną akcję i wysyłając odpowiedź z powrotem do „Master”. Tylko „Master” może zainicjować zapytanie.



W protokole MODBUS master może zaadresować poszczególne urządzenia slave lub za pomocą specjalnego adresu „Broadcast”, może zainicjować komunikat rozgłoszeniowy dla wszystkich urządzeń slave. Miernik cyfrowy NMID30-1 nie obsługuje adresu rozgłoszeniowego.

#### 3.1 Format komunikatów protokołu MODBUS

Protokół MODBUS definiuje format zapytania master i odpowiedzi slave.

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

Odpowiedź zawiera pola potwierdzające podjęte działania, wszelkie dane do zwrócenia i pole sprawdzania błędów. Jeśli wystąpił błąd w odbiorze komunikatu, komunikat jest ignorowany, jeśli urządzenie slave nie może wykonać żądanej akcji, to skonstruuje komunikat o błędzie i wyśle go jako odpowiedź. Funkcje protokołu MODBUS używane przez mierniki cyfrowe NMID30-1 kopiują wartości rejestrów 16-bitowych między urządzeniami master i slave. Jednak dane wykorzystywane przez miernik cyfrowy NMID30-1 są generowane w 32-bitowym formacie liczb zmiennoprzecinkowych IEEE 754. Zatem każdy parametr instrumentu 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 Ostatni Bajt

Adres slave	Kod funkcji	Adres początkowy (Hi)	Adres początkowy (Lo)	Liczba punktów (Hi)	Liczba punktów (Lo)	Sprawdzanie błędów (Lo)	Sprawdzanie błędów (Hi)
-------------	-------------	-----------------------	-----------------------	---------------------	---------------------	-------------------------	-------------------------

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

Kod funkcji: 8-bitowa wartość informująca adresowany slave, jakie działanie ma zostać wykonane. (3, 4, 8 lub 16 są ważne dla licznika 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. Ponieważ rejestry są używane parami i zaczynają się od zera, musi to być liczba parzysta.

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. Ponieważ rejestry są używane parami, musi to być liczba parzysta.

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 Ostatni Bajt

Adres slave	Kod funkcji	Liczba Bajtów	Pierwszy rejestr (Hi)	Pierwszy rejestr (Lo)	Drugi rejestr (Hi)	Drugi rejestr (Lo)	Sprawdzanie błędów (Lo)	Sprawdzanie błędów (Hi)
-------------	-------------	---------------	-----------------------	-----------------------	--------------------	--------------------	-------------------------	-------------------------

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

Kod funkcji: 8-bitowa wartość, która, gdy jest kopią kodu funkcji w zapytaniu, wskazuje, że slave rozpoznał zapytanie i odpowiedział. (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 żądanego 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

Ostatni Bajt

Adres slave	Kod funkcji	Kod błędu	Sprawdzanie błędów (Lo)	Sprawdzanie błędów (Hi)
-------------	-------------	-----------	-------------------------	-------------------------

Adres slave: 8-bitowa wartość reprezentująca adres slave, który 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

Zażądano.

Kod błędu: 8-bitowa wartość wskazująca charakter wykrytego wyjątku. (Zobacz „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.

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

reprezentującej wartość sprawdzenia błędu.

### 3.2 Tryby transmisji szeregowej

Istnieją dwa tryby transmisji szeregowej protokołu MODBUS: ASCII i RTU. Miernik cyfrowy NMID30-1 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 I 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: Liczniki cyfrowe nie obsługują 38400, ale zamiast tego oferują 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 odrzucany.

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 przesyłanym bajcie, cichy interwał o długości co najmniej 3,5 znaku oznacza koniec komunikatu. Nowy komunikat może rozpocząć się po tym interwale. W NMID30-1 1000 i 2000 wymagany jest cichy interwał minimum 60 ms, aby zagwarantować pomyślne 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)

Najmniej znaczący bit (LSB)



Najbardziej znaczący bit (MSB)



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

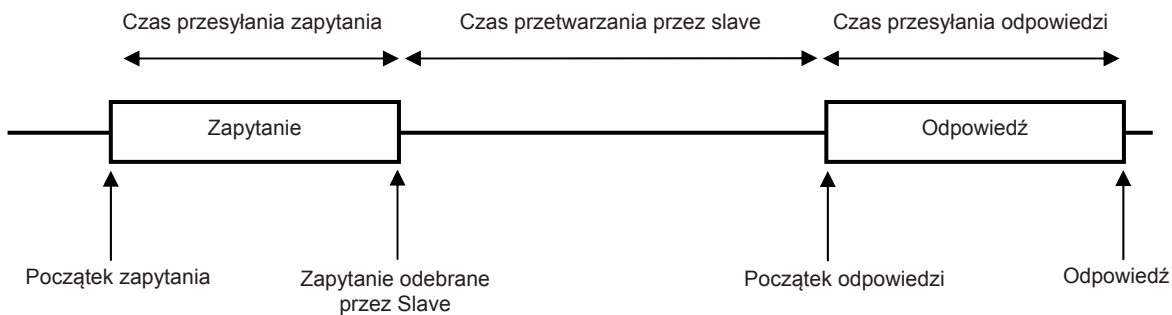
Znak transmisji = bit startu + bajt danych + 2 bit stopu (łącznie 11 bitów):

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

Cyfrowy miernik NMID30-1 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	Bieg
-------	---	---	---	---	---	---	---	---	------------	------

Master jest konfigurowany przez użytkownika, tak aby czekał na określony czas interwału. Master zaczeka przez ten okres czasu, zanim zdecyduje, że slave nie odpowie i że transakcja powinna zostać przerwana. Należy zachować ostrożność podczas określania limitu czasu zgodnie ze specyfikacją urządzenia master, jak i specyfikacją slave. Slave może zdefiniować „czas odpowiedzi” jako okres od otrzymania ostatniego bitu zapytania do przesłania pierwszego bitu odpowiedzi Master może zdefiniować „czas odpowiedzi” jako okres pomiędzy przesłaniem pierwszego bitu zapytania do otrzymania ostatniego bitu odpowiedzi. Można zauważyć, że 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 rozłączany 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 następnym 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 błędów. W następującym pseudo kodzie „Słowo błędów” 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 kody funkcji komunikatu protokołu MODBUS określa działanie, które ma podjąć urządzenie slave. Cyfrowy miernik NMID30-1 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 pod-funkcja zero. Odpowiedzią jest element danych zapytania bez zmian.
15	Wstępne ustawienie wielu rejestrów	Ustawienie 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 przezwyciężenia tego ograniczenia. Mierniki cyfrowe NMID30-1 wykorzystują dwa kolejne rejestry do reprezentowania liczby zmiennoprzecinkowej, skutecznie rozszerzając zakres do +/- 1x1037.

Wartości wytwarzane przez liczniki cyfrowe NMID30-1 mogą być używane bezpośrednio bez konieczności „skalowania” wartości, na przykład jednostkami dla parametrów napięcia są wolt, 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 „normalna mantysa”. 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.
SEE E	EMMM MMMM	MMMM MMMM	MMMM MMMM

Gdzie:

S reprezentuje bit znaku, gdzie 1 jest ujemne, a 0 dodatnie

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

M to 23-bitowa normalna mantysa. 24-ty bit jest zawsze 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 demonstruje, jak konwertować liczby zmiennoprzecinkowe IEEE 754 z ich postaci szesnastkowej na postać dziesiętną. W tym przykładzie użyjemy wartości 240.5 pokazanej powyżej. Należy zauważyć, że reprezentacja zapisu liczby zmiennoprzecinkowej nie jest formatem intuicyjnym. Aby przekonwertować tę wartość na dziesiętną, bity powinny być rozdzielone tak jak określono to 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 pojawia się jako liczba binarna 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 przesuwamy punkt binarny w lewo. Dodatni wykładnik przesuwamy punkt binarny w prawo. Ponieważ wykładnik wynosi 7, mantysa jest dostosowywana w następujący sposób: 11110000.100000000000000000

Wreszcie mamy 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 razem 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 mierników Eastron Digital 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 NMID30-1 obsługują komendy protokołu MODBUS RTU „Odczyt Rejestrów Wejściowych” (rejestry 3X), „Odczyt Rejestrów do Zapisu/Odczytu” (rejestry 4X) oraz „Wstępne ustawienie wielu rejestrów” (rejestry 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 instrumentu o adresie węzła 1:

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	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.

Następująca odpowiedź daje zawartość Volts 1 jako 230,2. Zobacz też „Odpowiedź na wyjątek” poniżej.

Nazwa pola	Przykład (Hex)
Slave Address	01
Function	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 „Czasu 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. Następująca odpowiedź zwraca zawartość Czasu Demand jako 1, ale zobacz także „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. Na ogół na jedno zapytanie można zapisać tylko jedną wartość zmiennoprzecinkową. Następująca 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

Gdyby slave w powyższym przykładzie „Rejestru do zapisu/odczytu” nie obsługiwał tej funkcji, odpowiedziałby odpowiedzią na wyjątek, jak pokazano poniżej. Kod funkcji wyjątku jest oryginalnym kodem funkcji z zapytania z ustawieniem MSB, tj. ma on 80 hex zsumowane logicznie ze sobą. Kod wyjątku wskazuje powód wyjątku. Slave nie odpowie w ogóle, jeśli wystąpi błąd z parzystością lub CRC zapytania. Jeśli jednak slave nie może przetworzyć zapytania, odpowie na wyjątek. W tym przypadku kod 01, żądana funkcja nie jest obsługiwana przez to 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

Cyfrowe mierniki obsługują następujące kody funkcji:

Kod wyjątku	Nazwa protokołu MODBUS	Opis
01	Illegal Function	Kod funkcji nie jest obsługiwany przez miernik
02	Illegal Data	Próba adresowanie, aby uzyskać dostęp do nieprawidłowego adresu lub próba odczytania lub zapisania części wartości zmiennoprzecinkowej
03	Illegal Data	Próba ustawienia zmiennej zmiennoprzecinkowej na niewłaściwą wartość
05	Slave Device	Błąd Wystąpił błąd, gdy przyrząd próbował zapisać aktualizację do konfiguracji

### 3.12 Diagnostyka

Kod 08 protokołu MODBUS zapewnia szereg podfunkcji diagnostycznych. Mierniki cyfrowe NMID30-1 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,

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