

CZAKI THERMO-PRODUCT

ul. 19 Kwietnia 58
05-090 Raszyn-Rybie
tel. (22) 7202302
fax. (22) 7202305
www.czaki.pl
handlowy@czaki.pl



PROGRAMOWALNY PRZETWORNIK POMIAROWY

TMD-20
(SX034)

INSTRUKCJA OBSŁUGI
GWARANCJA

Spis treści

1. Wprowadzenie	3
2. Opis zacisków przetwornika	4
2.1 Podłączanie interfejsu i zasilania	4
2.1 Podłączanie czujnika termorezystancyjnego	4
2.3 Podłączenie czujnika termoelektrycznego	4
2.4 Podłączenie sygnału prądowego lub napięciowego	4
2.5 Interfejs RS-485	5
3. ModBus - opis protokołu	6
4. ModBus - opis funkcji	9
4.1 Wykaz rejestrów przetwornika	13
5. Dane techniczne	14
5.1 Ustawienia fabryczne	15
6. Zawartość opakowania	15
7. Warunki gwarancji	16
8. Karta gwarancyjna	16

1. Wprowadzenie

TMD-20 jest mikroprocesorowym przetwornikiem pomiarowym umożliwiającym pomiar następujących sygnałów:

temperatury za pomocą czujników:

termorezystancyjnych: Pt100, Ni100

termoelektrycznych: T, J, K, N, S, R, B

napięć z zakresu $\pm 10V$

prądów z zakresu $\pm 25mA$

Przetwornik posiada interfejs RS-485, który umożliwia podłączenie wielu przetworników do wspólnej magistrali. Konfiguracja przetwornika oraz odczyt wyników pomiarowych odbywa się poprzez interfejs RS-485.

W celu zapewnienia bezkolizyjnej, odpornej na błędy transmisji danych - wykorzystano protokół komunikacyjny ModBus-RTU.

Podstawowe informacje o protokole Modbus podane są w punkcie 4 Instrukcji.

Przetwornik zapewnia separację galwaniczną wejść pomiarowych od zasilania i wyjścia cyfrowego.

Konstrukcja przetwornika TMD-20 umożliwia tworzenie rozproszonych systemów pomiarowych, zapewniając wysoką dokładność uzyskanych wyników.

2. Opis zacisków przetwornika

10 PWR (+24V DC)

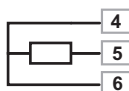
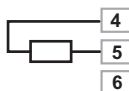
12 GND

7 A (D+)

8 B (D-)

9 GND

Rys. 2.1 Podłączenie zasilania oraz interfejsu RS-485



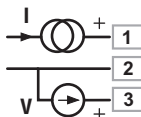
2-przewodowe

3-przewodowe

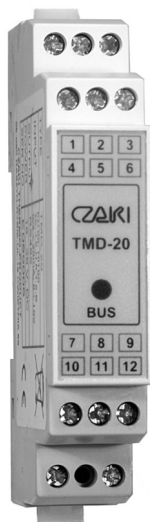
Rys. 2.2 Podłączenie czujnika termorezystancyjnego (wersja SX034)



Rys. 2.3 Podłączenie czujnika termoelektrycznego (termopary) (wersja SX034)



Rys. 2.4 Podłączenie sygnału napięciowego lub prądowego (wersja SX034)

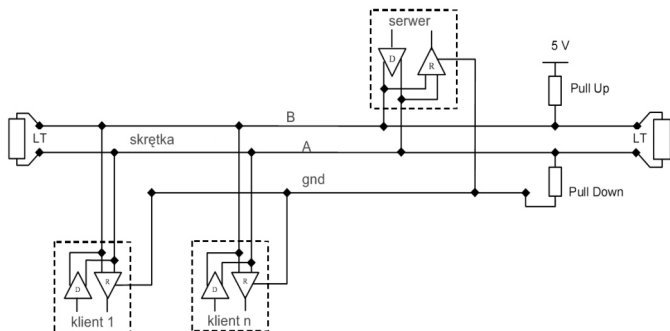


2.5 Interfejs RS-485

Zalecany kabel : ekranowana skrętka 24AWG o $Z > 1000\Omega$.

Kabel powinien być ekranowany. Ekran należy jednostronnie uziemić.

Uwaga - można stosować kable Ethernetowe kategorii 5 STP (kable określone normą EIA 568). Maksymalna długość magistrali na takich kablach wynosi 600m.



Rys. 2.5 Zalecany sposób podłączenia interfejsu RS-485 Modbus-RTU

Sugerowane kolory przewodów:

A	brązowy
B	żółty
GND	szary

Typowe układy nadawczo-odbiorcze umożliwiają podłączenie do 32 urządzeń RS485 na wspólnej magistrali. By zwiększyć ilość urządzeń stosuje się powielacze RS-485 (repeaters).

3. ModBus - opis

Standard ModBus definiuje protokół komunikacyjny, który umożliwia bezkolizyjną komunikację urządzeń w trybie klient/serwer.

Komunikacją w sieci ModBus zarządza urządzenie nadrzędne tj. serwer.

Pozostałe urządzenia są określane mianem klientów.

Wymiana danych w sieci ModBus polega na przesyłaniu komunikatów pomiędzy serwerem a poszczególnymi klientami, przy czym klient wysyła komunikat tylko w odpowiedzi na komunikat serwera.

Każdy klient musi mieć nadany unikatowy numer (z zakresu $1 \div 247$), który jest jego adresem w sieci. Serwer komunikuje się z wybranym klientem, podając jego adres w treści komunikatu.

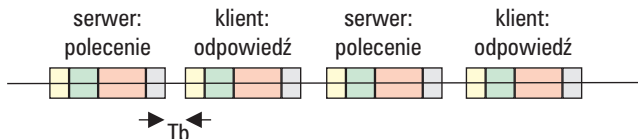
Poza adresem klienta, komunikat zawiera: kod polecenia, dane oraz sumę kontrolną komunikatu.

adres	kod funkcji	dane	CRC
1 bajt	1 bajt	0 ÷ 252 bajty	2 bajty

Rys. 3.1 Ramka komunikatu ModBus-RTU

Kod polecenia informuje klienta o akcji jakiej żąda od niego serwer.

Adresat po odczytaniu treści komunikatu wykonuje polecenie (np. przesyła wynik pomiaru) lub zwraca informację o błędzie - jeśli nie może wykonać danego polecenia.



T_b - przerwa w dostępie do magistrali : $\min 3.5 * \text{czas nadawania pojedynczego bajtu (11 bitów)}$

Rys. 3.2 Przesyłanie komunikatów po magistrali szeregowej

Każde urządzenie klienckie "obsługuje" pewną listę poleceń - zależnie od rodzaju i funkcjonalności urządzenia. Polecenia zazwyczaj są żądaniem zapisu lub odczytu rejestrów lub linii we/wy urządzenia.

Lista poleceń obsługiwanych przez dane urządzenie, oraz wykaz dostępnych rejestrów, podawana jest w dokumentacji urządzenia.

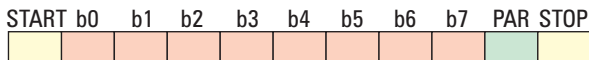
Suma kontrolna ramki

Każdy komunikat ModBus zakończony jest 16-bitową sumą kontrolną -CRC. W przetworniku TMD-10 wartość początkowa wielomianu służącego do wyliczenia CRC wynosi 0xA001 (heksadecymalnie).

Format danych

W protokole ModBus rejestry są podstawową "porcją" danych. Rejestr ma rozmiar dwóch bajtów. Komunikaty ModBus przesyłane są szeregowo - bajt po bajcie.

Bajt przesyłany jest od najmłodszego do najstarszego bitu. Każdy bajt danych poprzedzony jest bitem START, a zakończony bitem kontroli parzystości (PAR) oraz bitem STOP:



Kontrola parzystości.

Bit kontroli parzystości PAR ustawiany jest zależnie od wybranego trybu kontroli parzystości:

Tryb	parzysta	nieparzysta	ilość jedynek w bajcie
EVEN	PAR = 0	PAR = 1	
ODD	PAR = 1	PAR = 0	
NO	PAR = 1	PAR = 1	

przykład:

dla bajtu 0110 1010

mamy PAR = 0 (EVEN) lub PAR = 1 (ODD).

3.4 Reprezentacja danych

Przesyłanie rejestrów.

Przy przesyłaniu wartości rejestru jako pierwszy wysyłany jest młodszy bajt rejestru (LSB) a następnie bajt starszy (MSB).

np. 0x1234 wysyłany jest w postaci: 0x34 , 0x12

Przesyłanie liczby rzeczywistej:

W przetwornikach TMD liczby rzeczywiste zapisywane są w postaci 4 bajtowych wartości. Format zapisu jest zgodny z zapisem liczb zmiennoprzecinkowych pojedynczej precyzji (6 cyfr znaczących).

Liczba rzeczywista zapisywana jest w postaci:

$$x = (-1)^s * 2^{(e-127)} * (1.f)$$

gdzie $s = b31$, $e = (b2...b9)2$, and $f = b10b11...b32$.

Kolejność wysyłania :

Przesyłanie ciągów znaków

Pojedynczy znak ma rozmiar 1 bajtu. Stąd rejestr zawiera 2 znaki

Przesyłanie ciągu znaków wykonywane jest jak przesyłanie ciągu rejestrów: najpierw przesyłany jest młodszy bajt, następnie starszy.

Przykład: tekst 'TMD-20' przesyłany jest w kolejności:

'M.' '_ 'T' '_ 'D' '_ '0' '_ '2'

<i>Przykład</i>	<i>dana</i>	<i>kolejność bajtów</i>
0x1234	rejestr (2 bajty)	0x34 12
0x12345678	liczba rzeczywista (4 bajty)	0x34 12 78 56
'abcdef'	tekst (1 rejestr zawiera 2 znaki)	'badcfe'

4. Wykaz obsługiwanych funkcji ModBus

03	(0x03)	Odczyt rejestrów wewnętrznych
06	(0x06)	Zapis pojedynczego rejestru
08	(0x08)	Funkcje diagnostyczne:
10	(0x10)	Zapis rejestrów wewnętrznych

Funkcja 0x03 (Odczyt rejestrów)

Funkcja służy do odczytu wewnętrznych rejestrów urządzenia (np. wartości pomiarów, wartości parametrów itp.)

Polecenie serwera:

kod funkcji	1B	0x03
adres rejestru	2B	0x0000 ÷ 0xFFFF
ilość rejestrów	2B	1 ÷ 125

Odpowiedź klienta:

kod funkcji	1B	0x03
ilość bajtów	1B	2xN
wartość rejestrów	2B x N	

Zgłoszenie błędu:

kod błędu	0x83
kod wyjątku	1B
adres poza zakresem	0x02
dane poza zakresem	0x03

Funkcja 0x06 (Zapis pojedynczego rejestru)

Funkcja umożliwia zapis pojedynczego rejestru urządzenia (np. sieciowy adres przetwornika).

Polecenie serwera:

kod funkcji	1B	0x06
adres rejestru	2B	0x0000 ÷ 0xFFFF
wartość rejestru	2B	0x0000 ÷ 0xFFFF

Odpowiedź klienta:

kod funkcji	1B	0x06
adres rejestru	2B	0x0000 ÷ 0xFFFF
wartość rejestru	2B	0x0000 ÷ 0xFFFF

Zgłoszenie błędu:

kod błędu	1B	0x86
kod wyjątku	1B	
adres poza zakresem		0x02
dane poza zakresem		0x03

Funkcja 0x10 (Zapis do rejestrów)

Funkcja umożliwia zapis kilku rejestrów urządzenia (np. zapis liczby rzeczywistej lub opisu obiektu).

Polecenie serwera:

kod funkcji	1B	0x10
adres 1-go rejestru	2B	0x0000 ÷ 0xFFFF
ilość rejestrów (N)	2B	0x0001 ÷ 0x0078
ilość bajtów	1B	N * 2
wartości rejestrów	N*2B	0x0001 ÷ 0x0078

Odpowiedź klienta:

kod funkcji	1B	0x10
adres 1-go rejestru	2B	0x0000 ÷ 0xFFFF
ilość bajtów	2B	0x0001 ÷ 0x0078

Zgłoszenie błędu:

kod błędu	1B	0x90
kod wyjątku	1B	
adres poza zakresem		0x02
dane poza zakresem		0x03

Funkcja diagnostyczna 0x08

Funkcja służy do przeprowadzania diagnostyki przetwornika.

Przy wywołaniu jako parametr podawany jest kod podfunkcji

Polecenie serwera:

kod funkcji	1B	0x08
kod podfunkcji	2B	0x0000,0x0003,0x0002
dane N * 2B		

Odpowiedź klienta * (patrz opis podfunkcji 0x0004):

kod funkcji	1B	0x03
kod podfunkcji	2B	0x0001
dane	2B	0x0000

Subfunkcje diagnostyczne.

Echo

kod podfunkcji	2B	0x0000
dane	N*2B	

Przetwornik powtarza odebrane dane - komunikat wysłany przez serwer będzie identyczny z komunikatem wysłanym w odpowiedzi.

Reset interfejsu komunikacyjnego

kod podfunkcji	2B	0x0001
dane	2B	0x0000

Port szeregowy przetwornika zostaje zresetowany i ponownie zainicjalizowany. Jeśli przetwornik był ustawiony w tryb nasłuchu - nie wysyła odpowiedzi i wychodzi z trybu nasłuchu.

Jeśli przetwornik nie był w trybie nasłuchu - wysyła odpowiedź na polecenie - tak jak przy podfunkcji echo.

Zmiana ustawień parametrów komunikacyjnych (szybkość transmisji, kontrola parzystości) oraz zmiana adresu w sieci - zostaje wykonana po resecie interfejsu komunikacyjnego!

4.1 Wykaz rejestrów przetwornika TMD-20

Adres	Rozmiar	Opis	Typ danych
Rejestry informacyjne (tylko do odczytu)			
\$A0A	4R	typ przetwornika	tekst (8 znaków)
\$A01	1R	identyfikator typu	liczba heksadecymalna
\$A02	4R	wersja wykonania	tekst (8 znaków)
\$A06	4R	wersja programu	tekst (8 znaków)
\$A0E	8R	strona www producenta	tekst (16 znaków)
Rejestry konfiguracyjne (dozwolony odczytu i zapis)			
\$880	1R	adres urządzenia w sieci ModBus	liczba z zakresu 1 ÷ 247
\$882	1R	szybkość transmisji	= 1 dla szybkości 9600 b/s = 2 dla szybkości 19200 b/s
\$883	1R	kontrola parzystości	= 1 dla parzystości 'EVEN' = 2 dla parzystości 'ODD' = 3 dla parzystości 'NO'
\$884	1R	konfiguracja pomiarowa	= 0 ... 4 (tab)
\$885	2R	wzmocnienie	liczba rzeczywista
\$887	1R	stała czasowa filtru wyjściowego	liczba całkowita 0 ÷ 255
\$888	2R	offset	liczba rzeczywista
\$88A	8R	identyfikator obiektowy	tekst (16 znaków)
Sygnały pomiarowe (tylko do odczytu)			
\$804	2R	wartość sygnału	liczba rzeczywista

* Adresy rejestrów podane są w kodzie heksadecymalnym

Rozmiar danych podany jest jako wielokrotność rozmiaru pojedynczego rejestru (2 bajty)

4.1 Wykaz rejestrów przetwornika TMD-20

Konfiguracja pomiarowa (rejestr \$884)

wartość	Opis
\$00	wejście napięciowe $\pm 10V$
\$01	wejście prądowe $\pm 25mA$
\$02	wejście termorezystancyjne Pt100 podłączenie 2-przew.
\$03	wejście termorezystancyjne Pt100 podłączenie 3-przew.
\$04	wejście termorezystancyjne Ni100 podłączenie 2-przew.
\$05	wejście termorezystancyjne Ni100 podłączenie 3-przew.
\$06	wejście termoelektryczne FE-CuNi (termoelement J)
\$07	wejście termoelektryczne NiCr-NiAl (termoelement K)
\$08	wejście termoelektryczne NiCrSi-NiSi (termoelement N)
\$09	wejście termoelektryczne PtRh10-Pt (termoelement S)
\$0A	wejście termoelektryczne PtRh13-Pt (termoelement R)
\$0B	wejście termoelektryczne PtRh30-PtRh6 (termoelement B)
\$0C	wejście termoelektryczne Cu-CuNi (termoelement T)

5.Dane techniczne

Typ czujnika wejściowego
(PN-EN 60751+A2:1997)

Pt100, Ni100

Podłączenie czujnika

2, lub 3 przewodowe

Zakres pomiarowy oraz błąd przetwarzania

Pt100	-100 ... 850°C	/ 0,05%
Ni100	-60 ... 180°C	/ 0,05%
T	-100 ... 230°C	/ 0,10%
J	-100 ... 1000°C	/ 0,10%
K	-100 ... 1200°C	/ 0,10%
N	-100 ... 1300°C	/ 0,10%
S	200 ... 1600°C	/ 0,10%
R	200 ... 1600°C	/ 0,10%
B	400 ... 1800°C	/ 0,15%
prądowy	0 ... $\pm 25mA$	/ 0,10%
napięciowy	0 ... $\pm 10V$	/ 0,10%

5. Dane techniczne cd.

Dryft temperaturowy	< 0,01% wskazania / °C
Błąd kompensacji temperatury zimnych końców	1,0 °C
Prąd pomiarowy Pt100, Ni100	0,2 mA
Czas odpowiedzi	< 0,2s
Czas uśredniania pomiarów	< 0,3 ÷ 60s (ust fabr. 1 s)
Komunikacja	RS-485
Protokół komunikacyjny	MODBUS-RTU
Adres sieciowy	1 ... 247
Szybkość transmisji	9600, 19200 bit/s
Parametry transmisji	8E1, 8O1, 8N1 (8 bitów danych + bit kontroli parzystości + bit stop)
Separacja galwaniczna	500V AC
Zasilanie	12 ... 36 V DC / 0,2 W
Temperatura pracy	0 ... +60°C
Wilgotność	< 90% bez kondensacji
Wymiary (wys. x szer. x gł.) / ciężar	98 x 17,5 x 56,4mm / ~ 50g

5.1 Ustawienia fabryczne

Adres sieciowy	1
Szybkość transmisji	19200 bit/s
Parametry transmisji	8E1
Konfiguracja pomiarowa	wejście prądowe ±25mA
Stała czasowa filtra wyjściowego	2

6. Zawartość opakowania

- Przetwornik TMD-20.
- Instrukcja obsługi z kartą gwarancyjną.

