



***MODBUS_TCPIP - drajwer protokołu
MODBUS_TCP/IP wg standardu OPEN
MODBUS/TCP
Podręcznik użytkownika***

Dok. Nr PLP4038
Wersja: 05-10-2005

ASKOM[®] i **asix[™]** to zastrzeżone znaki firmy ASKOM Sp. z o. o., Gliwice. Inne występujące w tekście znaki firmowe bądź towarowe są zastrzeżonymi znakami ich właścicieli.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną lub inną powoduje naruszenie praw autorskich niniejszej publikacji.

ASKOM Sp. z o. o. nie bierze żadnej odpowiedzialności za jakiegokolwiek szkody wynikłe z wykorzystywania zawartych w publikacji treści.

Copyright © 2005, ASKOM Sp. z o. o., Gliwice



ASKOM Sp. z o. o., ul. Józefa Sowińskiego 13, 44-121 Gliwice,
tel. +48 (0) 32 3018100, fax +48 (0) 32 3018101,
<http://www.askom.com.pl>, e-mail: office@askom.com.pl

1. MODBUS_TCPIP - drajwer protokołu MODBUS_TCP/IP wg standardu OPEN MODBUS/TCP

1.1. Przeznaczenie drajwera

Drajwer MODBUS TCPIP jest przeznaczony do wymiany danych pomiędzy systemem **asix** i innymi komputerami/urządzeniami za pomocą protokołu MODBUS, zrealizowanego w oparciu o sieć Ethernet z protokołem TCP/IP. Domyślnym trybem działania drajwera MODBUS_TCPIP jest tryb Open Modbus/TCP, opracowany na podstawie specyfikacji 'OPEN MODBUS/TCP Specification' Release 1.0, wydanej 29.03.1999 przez firmę Schneider Electric. Drajwer pozwala na jednoczesną realizację trybu SLAVE oraz trybu MASTER.

1.2. Tryb SLAVE

Tryb SLAVE polega na realizacji poleceń zapisu i odczytu zmiennych ASMEN'a nadesłanych z innych komputerów/urządzeń, pełniących w sieci MODBUS funkcję MASTER. Możliwa jest jednoczesna obsługa wielu komputerów pełniących funkcję MASTER.

Zmienne ASMEN'a są udostępniane w trybie SLAVE jako zmienne MODBUS'a należące do jednego z niżej wymienionych typów:

- CS (coil statuses),
- HR (holding registers),
- IR (input registers).

Deklaracje przypisania (mapowania) zmiennych ASMEN'a zmiennym MODBUS'a są umieszczane przez projektanta aplikacji w plikach tekstowych, które są odczytywane przez drajwer na etapie startu systemu **asix** – przykład na końcu rozdziału.

Dla odczytu zmiennych ASMEN'a zaimplementowano następujące funkcje protokołu MODBUS:

- Read Coil Statuses (funkcja 01),
- Read Holding Registers (funkcja 03),
- Read Input Registers (funkcja 04).

Dla zapisu zmiennych ASMEN'a zaimplementowano następujące funkcje protokołu MODBUS:

- Preset Single Coil (funkcja 05),
- Preset Single Register (funkcja 06),
- Preset Multiple Coils (funkcja 15),
- Preset Multiple Registers (funkcja 16).

1.2.1. Deklaracja kanału transmisji

Deklaracja kanału transmisji dla trybu SLAVE ma następującą postać:

nazwa_kanału=MODBUS_TCPIP, SLAVE, port [, numer]

gdzie:

MODBUS_TCPIP	- nazwa drajwera,
SLAVE	- realizacja trybu SLAVE,
port	- numer portu, w którym będzie następował nasłuch komputerów chcących dołączyć się do systemu asix w ramach realizowanej przez te komputery funkcji MASTER sieci MODBUS,
numer	- numer nadany komputerowi asix w sieci MODBUS (domyślnie 1).

UWAGA Można zadeklarować tylko jeden kanał transmisji realizujący tryb SLAVE.

1.2.2. Parametryzacja trybu SLAVE

Parametryzacja drajwera w systemie **asix** odbywa się przy użyciu osobnej sekcji w pliku ini aplikacji, o nazwie [MODBUS_TCPIP_SLAVE]. Przy użyciu tej sekcji można deklarować:

- listę klientów akceptowanych przez drajwer,
- mapowanie zmiennych MODBUS'a na zmienne ASMEN'a dla poszczególnych klientów,
- sposób przekazywania statusu dla poszczególnych klientów,
- sposób odświeżania zmiennych.



TYP_PROTOKOLU = <adres_IP>, OVATION

Znaczenie - obsługa stacji działających wg specyfikacji 'MODBUS RTU' wymaga użycia pozycji.

Parametr:

adres_IP - adres stacji pełniącej rolę mastera.



ADRES_IP_KLIENTA=adres_IP_klienta [,TYLKO_ODCZYT]

Znaczenie - do systemu **asix** mogą się dołączyć wyłącznie ci klienci, którzy posiadają zezwolenie na dostęp. Klienci są deklarowani przy użyciu pozycji ADRES_IP_KLIENTA, umieszczonej w sekcji MODBUS_TCPIP_SLAVE.

Wartość domyślna - domyślnie każdy klient ma możliwość odczytu i zapisu zmiennych systemu **asix**. Istnieje możliwość ograniczenia uprawnień klientów wyłącznie do odczytu zmiennych poprzez użycie opcji TYLKO_ODCZYT.

W takiej sytuacji próba zapisu jest kwitowana wysłaniem telegramu odpowiedzi typu *exception* z kodem 1, oznaczającym nielegalną funkcję.

Parametr:

adres_IP_klienta - adres IP klienta akceptowanego przez drajwer,

TYLKO_ODCZYT - opcja zezwalająca klientowi wyłącznie na odczyt danych.

Liczba deklaracji klientów jest nieograniczona.

PRZYKŁAD

Deklaracja klienta o adresie IP 10.10.10.84 z nieograniczonymi prawami dostępu do systemu **asix**:

ADRES_IP_KLIENTA=10.10.10.84



TIMEOUT_POLACZENIA=adres_IP_klienta,liczba

Znaczenie - dla każdego dołączonego klienta deklaruje się maksymalny czas, jaki może upłynąć pomiędzy kolejnymi zapytaniami ze strony klienta (tzw. timeout połączenia). Po przekroczeniu timeout'u połączenie z klientem jest zrywane. W przypadku klientów, dla których nie zostanie podana deklaracja time-out'u połączenia, domyślnie przyjmowana będzie wartość 5 minut.

Parametr:

adres_IP_klienta - adres IP klienta,

liczba - wartość timeout'u wyrażona w minutach.

PRZYKŁAD

Deklaracja timeout'u połączenia wynoszącego 10 minut dla klienta o adresie IP 10.10.10.84:

TIMEOUT_POLACZENIA=10.10.10.84, 10

Przekazywanie statusu zmiennych w osobnych rejestrach i cewkach



STATUS_REJESTROW=adres_IP_klienta



STATUS_CEWEK=adres_IP_klienta

Znaczenie - protokół MODBUS nie operuje pojęciem statusu w odniesieniu do przekazywanych wartości rejestrów oraz stanu cewek. Z tego powodu przekazywanie statusu można dołączyć jedynie w sposób sztuczny do standardowej przesyłki MODBUS'a. Przedmiotowy drajwer może przekazać status zmiennej **asix**'a, umieszczając go w kolejnym elemencie następującym za elementem zawierającym wartość zmiennej. Dla rejestrów jest to kolejny rejestr, dla cewek – kolejna cewka. W przypadku rejestrów przestrzeń dla przekazania statusu jest wystarczająca (16 bitów), dla cewek status musi być ograniczony do dwóch stanów (0 - dobry, 1 - zły).

Przykładowo, jeśli wartość zmiennej jest przekazywana jako rejestr 10, to status jest przekazywany jako rejestr 11. Jeśli stan cewki jest przekazywany na bicie 5, to status cewki jest przekazywany na bicie 6.

Użyta metoda pozwala przesłać wartość zmiennej oraz jej status w tym samym telegramie, zapewniając spójność danych.

Tryb przekazywania statusu jest deklarowany indywidualnie dla każdego klienta, oddzielnie dla rejestrów i cewek. Deklaracje umożliwiają:

- przekazywanie statusu rejestrów,
 - przekazywanie statusu cewek.
- Wartość domyślna - domyślnie status nie jest przekazywany ani dla rejestrów, ani dla cewek

Parametr:

- STATUS_REJESTROW* - przekazywanie statusu rejestrów,
- STATUS_CEWEK* - przekazywanie statusu cewek,
- adres_IP_klienta - adres IP klienta.

Przekazywanie statusu zmiennych bez użycia osobnych rejestrów lub cewek



STATUS_BLEDU_REJESTROW=adres_IP_klienta,liczba

Znaczenie

- w przypadku pracy bez przekazywania statusów w osobnych rejestrach lub cewkach pojawia się problem przekazania informacji o tym, że wartość zmiennej jest nieważna (np. błędy komunikacji ze źródłem danych). Przyjęto konwencję, w myśl której będzie możliwe deklarowanie wartości przekazywanej w przypadku niepoprawnego statusu dla rejestrów (domyślnie 0xffff).

Parametr:

- adres_IP_klienta* - adres IP klienta;
- liczba* - wartość 16-bitowa (HEX), przekazywana w przypadku niepoprawnego statusu zmiennej.

PRZYKŁAD

Deklaracja przekazywania do klienta 10.10.10.84 liczby 0x8000 w przypadku błędu sygnalizowanego w statusie zmiennej mapowanej na rejestry MODBUS'a:

STATUS_BLEDU_REJESTROW=10.10.10.84, 0x8000

UWAGA W przypadku cewek, ze względu na dwustanowy charakter zmiennej, nie ma możliwości przekazania informacji o statusie w taki sposób, aby odróżniała się ona od poprawnej wartości zmiennej. Z tego też względu należy przejąć jako zasadę, że w trybie pracy bez przekazywania statusów należy używać rejestry.

Deklaracje mapowania zmiennych MODBUS'a na zmienne ASMEN'a

Ze względu na to, że protokół MODBUS nie posługuje się nazwami zmiennych ale numerami rejestrów i cewek, konieczne jest przygotowanie plików konfiguracyjnych mapujących zmienne MODBUS'a na zmienne ASMEN'a.

Deklaracja mapowania zmiennej MODBUS'a na zmienną ASMEN'a ma następującą postać:

nazwa_ASIX'a, adres_MODBUS'a [, okres_wiarygodności]

gdzie:

<i>nazwa_ASIX'a</i>	- nazwa zmiennej procesowej systemu asix ; musi mieć swój odpowiednik wśród nazw zmiennych procesowych deklarowanych w plikach ASMEN'a;
<i>adres_MODBUS'a</i>	- typ oraz indeks zmiennej MODBUS'a, poprzez który będzie udostępniana wartość zmiennej procesowej; w zależności od typu zmiennej MODBUS'a jej wartość jest przekazywana poprzez jeden bit, jeden rejestr (zmienna typu SHORT lub USHORT) lub dwa kolejne rejestry (zmienna typu FLOAT, LONG lub ULONG);
<i>okres_wiarygodności</i>	- przedział czasu (w sekundach), w którym wartość zmiennej uznawana jest za poprawną. Jest on określany jako różnica czasu pomiędzy stemplem czasu zapytania od klienta i stemplem czasu aktualnie udostępnianej wartości zmiennej. Domyślnie 5 sekund.

Akceptowany format adresu MODBUS'a jest następujący:

HR<nr_rej> lub **IR**<nr_rej> lub **CS**<nr_cewki>

gdzie:

<i>nr_rej</i>	- liczba oznaczająca numer rejestru HR lub IR,
<i>nr_cewki</i>	- liczba oznaczająca numer cewki,

Liczba rejestrów MODBUS'a wykorzystywana do przesłania wartości zmiennej będzie ściśle skorelowana z typem zmiennej ASMEN'a:

- dla zmiennych typu WORD będzie to jeden rejestr lub jedna cewka,
- dla zmiennych typu INT16 będzie to jeden rejestr,
- dla zmiennych typu DWORD, LONG lub FLOAT będą to dwa rejestry;

lub w przypadku przesyłania statusów:

- dla zmiennych typu WORD będą to dwa rejestry lub dwie cewki,
- dla zmiennych typu INT16 będą to dwa rejestry,
- dla zmiennych typu DWORD, LONG lub FLOAT będą to trzy rejestry.

Poniżej podano sposób interpretacji zawartości rejestrów dla liczb FLOAT, LONG i ULONG MODBUS'a:

Pierwszy rejestr														Drugi rejestr																			
1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0	1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0
5	4	3	2	1	0												5	4	3	2	1	0											
Mantysa - młodsze bity														Z	Cecha																		

Rysunek 1. Sposób interpretacji zawartości rejestrów dla liczb FLOAT drajwera MODBUS.

Pierwszy rejestr														Drugi rejestr																			
1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0	1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0
5	4	3	2	1	0												5	4	3	2	1	0											
Z	Starsze bity														Młodsze bity																		

Rysunek 2. Sposób interpretacji zawartości rejestrów dla liczb LONG drajwera MODBUS.

Pierwszy rejestr														Drugi rejestr																			
1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0	1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0
5	4	3	2	1	0												5	4	3	2	1	0											
Starsze bity														Młodsze bity																			

Rysunek 3. Sposób interpretacji zawartości rejestrów dla liczb ULONG drajwera MODBUS.

PRZYKŁAD

W ASMEN'ie zmienne X_WORD, X_INT, X_DWORD, X_LONG, X_FLOAT są zdefiniowane następująco:

```
X_WORD,    ED120.2, KANAL1, 1, 1, NIC
X_INT,     ED130.2, KANAL1, 1, 1, NIC_INT
X_DWORD,   EL140.2, KANAL1, 1, 1, NIC_DW
X_LONG,    EL150.2, KANAL1, 1, 1, NIC_LONG
X_FLOAT,   EG160.2, KANAL1, 1, 1, NIC_FP
```

Mapowanie wartości tych zmiennych na ciągły obszar rejestrów zaczynający się od rejestru HR1 (bez przysyłania statusów) wygląda następująco:

```
X_WORD,    HR1
X_INT,     HR2
X_DWORD,   HR3
X_LONG,    HR5
X_FLOAT,   HR7
```

Mapowanie wartości tych zmiennych na ciągły obszar rejestrów zaczynający się od rejestru HR1 (z przysyłaniem statusów) wygląda następująco:

```
X_WORD,    HR1
X_INT,     HR3
X_DWORD,   HR5
X_LONG,    HR8
X_FLOAT,   HR11
```

Deklaracje mapowań zmiennych MODBUS'a na zmienne ASMEN'a są umieszczane w plikach tekstowych. Lokalizację plików z deklaracjami mapowań określa się przy użyciu pozycji:



PLIK_MAPOWAN=adres_IP_klienta,nazwa_pliku

Znaczenie - lokalizacja plików z deklaracjami mapowań.

Parametr:

adres_IP_klienta - adres IP klienta,

nazwa_pliku - nazwa pliku zawierającego deklaracje mapowań.

Powiązanie pliku z adresem IP klienta ma na celu umożliwienie poszczególnym klientom niezależnego sposobu mapowania.

Liczba pozycji z deklaracjami plików mapowań jest nieograniczona.

Odświeżanie zmiennych udostępnianych przez drajwer

Drajwer może stosować jedną z dwóch strategii odnośnie obsługi zmiennych eksportowanych z systemu asix.

Strategia I (domyślna) zakłada, że wszystkie zmienne, których nazwy zostały zadeklarowane w plikach z deklaracjami mapowań są cyklicznie odczytywane przez osobny wątek drajwera z cache'a ASMEN'a i umieszczane w ogólnodostępnym buforze drajwera. Wątki robocze obsługujące połączenia z klientami budują telegramy odpowiedzi bazując na zawartości tego bufora, bez konieczności odwoływania się do API ASMEN'a.

Okres odczytu z cache'a ASMEN'a do ogólnodostępnego bufora drajwera jest parametryzowany przy użyciu pozycji:



OKRES_AKTUALIZACJI_BUFORA_DANYCH=liczba

Znaczenie - pozwala zadeklarować okres odczytu z cache'a ASMEN'a do ogólnodostępnego bufora drajwera.
 Wartość domyślna - domyślnie okres odczytu wynosi 1 sekundę.
 Parametr:
liczba - okres odczytu z cache'a ASMEN'a w sekundach.

Strategia II (opcja) zakłada, że każdy wątek roboczy samodzielnie pobiera dane z cache'a ASMEN'a w reakcji na zapytania przesłane przez klienta. W efekcie wątek roboczy odczytuje z cache'a ASMEN'a wyłącznie te dane, które są w danej chwili żądane przez klienta.

Strategia II powinna być również aktywowana w przypadku, gdy system **asix** będzie korzystał z drajwera wyłącznie w celu importu danych z innego systemu. W takim trybie cykliczne odświeżanie zawartości buforów drajwera nie ma sensu.

Strategia II jest uaktywniana przy pomocy pozycji:



ODCZYT_NA_ZYCZENIE = TAK

Wartości zmiennych znajdujących się w cache'u ASMEN'a w chwili realizacji polecenia odczytu mogą być nieaktualne, ponieważ w okresie poprzedzającym zapytanie klienta odświeżanie przedmiotowych zmiennych mogło nie być aktywne (system **asix** został dopiero uruchomiony lub też żaden z klientów ASMEN'a nie był zainteresowany odświeżaniem tych zmiennych). Z tego względu wątek przygotowujący przesyłkę dla klienta będzie sprawdzał stempel czasu aktualnie udostępnianych wartości zmiennych. Jeśli wartość stempla czasu nie będzie się mieściła w zakresie *okresu_wiarygodności* określonym w deklaracji mapowania zmiennej, to wątek odczyta wartość zmiennej bezpośrednio z drajwera i dopiero wtedy odeśle odpowiedź do klienta.

Domyślnie drajwer realizuje strategię I z częstością aktualizacji zawartości ogólnodostępnego bufora wynoszącą 1 sekundę.



PLIK_LOGU=nazwa_pliku

Znaczenie - pozycja pozwala na zdefiniowanie pliku, do którego będą zapisywane wszystkie komunikaty diagnostyczne drajwera w trybie SLAVE oraz informacje o zawartości telegramów odebranych/wysłanych przez drajwer w tym trybie. Jeśli pozycja PLIK_LOGU nie definiuje pełnej ścieżki, to plik logu zostanie utworzony w bieżącej kartotece. Plik logu powinien być wykorzystywany jedynie w fazie uruchamiania systemu **asix**.
 Wartość domyślna - domyślnie nie tworzy się pliku logu.
 Parametr:
nazwa_pliku - nazwa pliku logu

**LOG_TELEGRAMOW =TAK/NIE**

- Znaczenie - pozycja pozwala na zapisywanie do pliku logu (zadeklarowanego przy użyciu pozycji PLIK_LOGU) zawartości telegramów przesyłanych pomiędzy drajwerem pracującym w trybie SLAVE i klientami sieciowymi. Zapis zawartości telegramów do pliku logu powinien być wykorzystywany jedynie w fazie uruchamiania systemu **asix**.
- Wartość domyślna - domyślnie drajwer nie wpisuje do pliku logu treści telegramów.

**ROZMIAR_PLIKU_LOGU=liczba**

- Znaczenie - pozycja pozwala określić rozmiar pliku logu w MB.
- Wartość domyślna - domyślnie pozycja przyjmuje, że plik logu ma rozmiar 1 MB.
- Parametr:
liczba - rozmiar pliku logu w MB.

1.3. Tryb MASTER

Tryb MASTER jest implementacją funkcji MASTER protokołu sieci MODBUS (w trybie RTU) opartej o Ethernet z protokołem TCP/IP.

Tryb MASTER ma zaimplementowane następujące typy danych:

- CS - Coil Statuses,
- IS - Input Statuses,
- HR - Holding Registers,
- IR - Input Registers,
- HRL - 2 kolejne Holding Registers traktowane jako podwójne słowo w formacie INTELA,
- HRF - 2 kolejne Holding Registers traktowane jako liczba zmiennoprzecinkowa w formacie INTELA,
- HRLM - 2 kolejne Holding Registers traktowane jako podwójne słowo w formacie MOTOROLI,
- HRFM - 2 kolejne Holding Registers traktowane jako liczba zmiennoprzecinkowa w formacie MOTOROLI,
- IRL - 2 kolejne Input Registers traktowane jako podwójne słowo w formacie INTELA,
- IRF - 2 kolejne Input Registers traktowane jako liczba zmiennoprzecinkowa w formacie INTELA,
- IRLM - 2 kolejne Input Registers traktowane jako podwójne słowo w formacie MOTOROLI,
- IRFM - 2 kolejne Input Registers traktowane jako liczba zmiennoprzecinkowa w formacie MOTOROLI.

PRZYKŁADY

- CS22 - Coil 22
- IS197 - Input 197
- HR118 - Holding Register 118

IR25 - Input Register 25

oraz następujące funkcje protokołu MODBUS:

Read Coil Statuses	(funkcja 01),
Read Input Statuses	(funkcja 02),
Read Holding Registers	(funkcja 03),
Read Input Registers	(funkcja 04),
Preset Single Coil	(funkcja 05),
Preset Single Register	(funkcja 06),
Preset Multiple Registers	(funkcja 16 ograniczona do zapisu pary rejestrów).

Drajwer MODBUS jest ładowany automatycznie jako DLL.

1.3.1. Delaracja kanału transmisji

Deklaracja kanału transmisji dla trybu MASTER ma następującą postać:

nazwa_kanału=MODBUS_TCPIP, MASTER, port, adres_IP, numer [, dwustan [, rejestr]]

gdzie:

<i>MODBUS_TCPIP</i>	- nazwa drajwera,
<i>MASTER</i>	- tryb MASTER,
<i>port</i>	- numer portu, przez który będzie realizowane połączenie z urządzeniem typu slave o numerze <i>numer</i> ,
<i>adres_IP</i>	- adres IP urządzenia,
<i>numer</i>	- numer urządzenia typu slave obsługiwanego w tym kanale (domyślnie 1),
<i>dwustan</i>	- max liczba dwustanów w jednym zapytaniu (domyślnie 32*8),
<i>rejestr</i>	- max liczba rejestrów w jednym zapytaniu (domyślnie 127).

Dla każdego urządzenia typu slave obsługiwanego w trybie MASTER będzie wymagana osobna deklaracja kanału transmisji.

1.3.2. Parametryzacja trybu MASTER

Parametryzacja trybu MASTER odbywa się przy użyciu osobnej sekcji o nazwie [**MODBUS_TCPIP_MASTER**]. Przy użyciu tej sekcji można deklarować:

- czas nawiązywania połączeń na etapie rozbiegu,
- timeout oczekiwania na odpowiedź z urządzenia typu slave,
- plik logu oraz jego rozmiar,
- log telegramów.

Deklaracja trybu pracy MODBUS RTU



TYP_PROTOKOLU = <adres_IP>, OVATION

Znaczenie - obsługa stacji działających wg specyfikacji 'MODBUS RTU' wymaga użycia niniejszej pozycji.

Parametr:

adres_IP - adres stacji pełniącej rolę slave'a.

Deklaracja czasu rozbiegu

Nawiązywanie połączeń ze slave'ami jest realizowane na etapie rozbiegu drajwera. Domyślny czas trwania rozbiegu to 3 sekundy. Może on być modyfikowany przy pomocy pozycji:

***CZAS_ROZBIEGU = liczba***

Znaczenie - czas rozbiegu drajwera, w czasie którego nawiązywane są połączenia ze slav'ami.
 Wartość domyślna - 3 s.
 Zdefiniowanie - ręczne.

***TIMEOUT_ODBIORU=adres_IP_klienta, liczba***

Znaczenie - dla każdego urządzenia typu slave określany jest maksymalny czas, jaki może upłynąć pomiędzy wysłaniem zapytania i otrzymaniem odpowiedzi (tzw. timeout odbioru). Po przekroczeniu timeout'u połączenie będzie zrywane (i nawiązywane ponownie). Wartość timeout'u jest określana indywidualnie dla każdego urządzenia typu slave.
 Wartość domyślna - w przypadku urządzeń typu slave, dla których nie zostanie podana deklaracja time-out'u odbioru, przyjmuje się domyślnie 5 sekund.

Parametry:

adres_IP_klienta - adres IP urządzenia typu slave,
liczba - wartość timeout'u wyrażona w sekundach.

***PLIK_LOGU=nazwa_pliku***

Znaczenie - pozycja pozwala na zdefiniowanie pliku, do którego będą zapisywane wszystkie komunikaty diagnostyczne drajwera generowane w trybie MASTER oraz informacje o zawartości telegramów wysłanych/odebranych przez drajwer w tym trybie. Jeśli pozycja PLIK_LOGU nie definiuje pełnej ścieżki, to plik logu zostanie utworzony w bieżącej kartotece. Plik logu powinien być wykorzystywany jedynie w fazie uruchamiania systemu **asix**.

Wartość domyślna - domyślnie nie tworzy się pliku logu.

Parametry:

nazwa_pliku - nazwa pliku logu.

***LOG_TELEGRAMOW =TAK/NIE***

Znaczenie - pozycja pozwala na zapisywanie do pliku logu (zadeklarowanego przy użyciu pozycji PLIK_LOGU) zawartości telegramów przesyłanych pomiędzy drajwerem pracującym w trybie MASTER i urządzeniami typu slave. Zapis zawartości telegramów do pliku logu powinien być wykorzystywany jedynie w fazie uruchamiania systemu **asix**.

Wartość domyślna - domyślnie drajwer nie wpisuje do pliku logu treści telegramów.

**ROZMIAR_PLIKU_LOGU=liczba**

- Znaczenie - pozycja pozwala określić rozmiar pliku logu w MB.
Wartość domyślna - domyślnie pozycja przyjmuje, że plik logu ma rozmiar 1 MB.
Parametr:
liczba - rozmiar pliku logu w MB.

2. Spis rysunków

<i>Rysunek 1. Sposób interpretacji zawartości rejestrów dla liczb FLOAT drajwera MODBUS.....</i>	<i>7</i>
<i>Rysunek 2. Sposób interpretacji zawartości rejestrów dla liczb LONG drajwera MODBUS.....</i>	<i>7</i>
<i>Rysunek 3. Sposób interpretacji zawartości rejestrów dla liczb ULONG drajwera MODBUS.....</i>	<i>7</i>

1. MODBUS_TCPIP - DRAJWER PROTOKOŁU MODBUS_TCP/IP WG STANDARDU OPEN	
MODBUS/TCP.....	3
1.1. PRZEZNACZENIE DRAJWERA	3
1.2. TRYB SLAVE.....	3
1.2.1. <i>Deklaracja kanału transmisji.....</i>	<i>4</i>
1.2.2. <i>Parametryzacja trybu SLAVE.....</i>	<i>4</i>
1.3. TRYB MASTER.....	10
1.3.1. <i>Delaracja kanału transmisji</i>	<i>11</i>
1.3.2. <i>Parametryzacja trybu MASTER.....</i>	<i>11</i>
2. SPIS RYSUNKÓW.....	15