Sieci przemysłowe w praktyce, część 7 Modbus – podstawy

Ponad dwadzieścia lat temu (w 1979 roku) firma Modicon zaproponowała otwarty protokół sieciowy Modbus jako standard wymiany informacji w sieciach przemysłowych. Rozwiązanie to zostało przyjęte przez wielu producentów sprzętu stosowanego w przemyśle, dziś jest już uznanym standardem, implementowanym w większości komercyjnych urządzeń sieciowych.

Modbus jest przykładem znakowej komunikacji asynchronicznej. System transmisyjny składa się z jednostki nadrzędnej (*mastera*) oraz wielu jednostek podrzędnych (*slave*). Protokół umożliwia zaadresowanie 247 urządzeń. Adres 0 jest zarezerwowany do trybu rozgłaszania.

Transmisja polega na wykonaniu transakcji. Realizowana jest ona przez wysłanie przez jednostkę nadrzędną zapytania do urządzeń podrzędnych i odebraniu przez nią odpowiedzi. Istnieją dwa typy zapytań. Pierwsze – do pojedynczego urządzenia. Drugie, to rozgłoszenie do wszystkich urządzeń w sieci. Na zapytanie składa się numer realizowanej funkcji, blok danych, bity kontroli poprawności danych. Przykład ramki odczytującej dane:

<AA><FF><PPPP><NNNN><CCCC>,
gdzie:
AA – to adres urządzenia docelowego,
FF – to numer funkcji MODBUS,
PPPP – oznacza numer parametru (adres komórki),
NNNN – oznacza liczbę parametrów do odczytania,
CCCC – suma kontrolna.

Tab. 1. Transakcje realizowane w sieci Modbus

Numer funkcji	Realizowana funkcja	Rozgłoszenie
1	Odczyt stanu binarnego wejścia	NIE
2	Odczyt stanu binarnego wyjścia	NIE
3	Odczyt wielu słów	NIE
4	Odczyt rejestru wejściowego	NIE
5	Wymuszenie stanu 1 wyjścia binarnego	TAK
6	Zapis stanu rejestru	TAK
7	Funkcja specjalna	NIE
8	Test pętli	–
9	Programowanie 484	–
10	Odpytywanie (polling) 484	–
11	Licznik zdarzeń transmisji	NIE
12	Logowanie zdarzeń transmisji	NIE
13	Programowanie sterownika	–
14	Odpytywanie sterownika	–
15	Wymuszenie stanu słowa wyjść binarnych	TAK
16	Zapis wielu rejestrów	TAK
17	Raport slave ID	NIE
18	Programowanie 884/M84	–
19	Kasowanie połączenia	–
20	Odczyt globalnych referencji	NIE
21	Zapis globalnych referencji	NIE
22	Zapis 4x rejestru z maską	NIE
23	Zapis/odczyt 4x rejestrów	NIE
24	Odczyt kolejki FIFO	NIE

Po odebraniu zapytania, urządzenie podrzędne odpowiada, przesyłając potwierdzenie odebrania polecenia, dane oraz bity kontroli poprawności danych. W przypadku błędów przy transmisji zapytania odpowiedzią jest specjalny komunikat o błędzie. Jeśli pytanie miało charakter rozgłoszenia, urządzenia nie potwierdzają odbioru komunikatu. Schemat ramki odpowiedzi jest taki sam jak zapytania.

W systemie Modbus zaprojektowano dwa tryby transmisji:

– Pierwszy to tryb ASCII, w którym bajt informacji jest przedstawiony jako dwa znaki ASCII. Tryb ten zezwala na sekwencyjne przesyłanie znaków, przy czym czas między poszczególnymi znakami w ramce nie musi być jednakowy. Nie może jednak przekroczyć 1s. Przesyłane znaki są kodowane heksadecymalnie w postaci znaków ASCII 0...9 oraz A...F. Oprócz 7 bitów danych jednostka informacyjna zawiera trzy dodatkowe bity: startu, parzystości i stopu.
– Drugi tryb to RTU. W odróżnieniu od trybu ASCII dane są tutaj przesyłane jako paczki danych ośmiobitowych w ramach jednej transakcji.

Znaczniki początku i końca transmisji powinny trwać około 3,5 okresu trwania przesyłania danych. Jednocześnie przerwy między poszczególnymi częściami ramki nie powinny być dłuższe niż 1,5 takiego okresu. Jeśli przerwa będzie dłuższa, urządzenie odbierające dane stwierdzi błąd transmisji. W **tab. 1** zestawiono funkcje sieci Modbus zdefinio-

wane przez firmę Modicon. W praktyce wykorzystywanych jest tylko kilka z nich, pozostałe to funkcje działające jedynie w urządzeniach firmy Modicon.

Ważnym elementem dla protokołu Modbus jest możliwość sprawdzenia poprawności przesłnych danych w trakcie transakcji. Dla transmisji ASCII polem potwierdzającym poprawność przesłanych danych jest LCR (*Longitudinal Redundancy Check*). Zarówno urządzenie nadające, jak i odbierające oblicza LCR według następującego algorytmu:

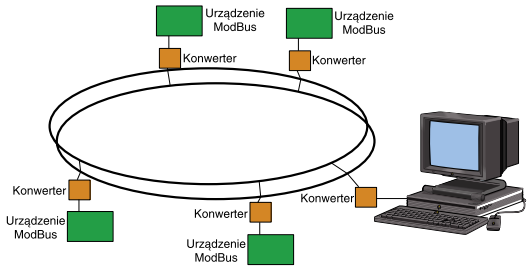
Krok 1. Dodać wszystkie słowa transmisji, podzielone w słowa 8-bitowe, wyłączając znak początku (:) i znaki końca CRLF. W operacji dodawania nie należy uwzględniać ewentualnych przeniesień.
Krok 2. Wynik dodawania odjąć od FFh.
Krok 3. Wykonać uzupełnienie do dwóch.

Inną metodą sprawdzenia poprawności jest obliczenie CRC (*Cyclic Redundancy Check*). W tym celu należy:

Krok 1. Nadać wartość początkową FFFFh.
Krok 2. Pobrać ośmiobitowe słowo danych i wykonać funkcję Ex-OR z młodszym bajtem danych CRC (CRC LO). Wynik zapisać w CRC.
Krok 3. Przesunąć zawartość rejestru CRC o 1 bit w prawo, najbardziej znaczący bit rejestru wyzerować.
Krok 4. Dla wartości 0 najmniej znaczącego bitu rejestru CRC przejść do kroku 3, dla wartości 1 wykonać Ex-OR rejestru CRC ze stałą A001h.
Krok 5. Powtarzać kroki 3 i 4 dla całego słowa danych.
Krok 6. Powtarzać kroki 2 do 5 dla wszystkich słów danych.
Krok 7. Zapisać gotową wartość CRC do przesyłanego komunikatu, zamieniając miejscami słowo mniej i bardziej znaczące.

W praktyce na szczęście nie ma potrzeby implementacji protokołu transmisji. Producenci sterowników przemysłowych i aparatury do nich udostępniają gotowe moduły, które wystarczy dodać do zestawu, lub serwery DDE, lub OPC, dzięki którym dane można odczytywać bezpośrednio z komputera.

Odmianą Modbusa spotykaną w przemyśle jest JBUS. Różni się on tylko tym, że ma odwrócone słowa: starszy i młodszy w danych dłuższych niż 8 bitów.



Rys. 1

Topologia sieci Modbus jest ściśle uzależniona od zastosowanej warstwy fizycznej sieci. Za pomocą konwerterów sieciowych możliwe jest łączenie różnych rodzajów medium, różnych topologii itd. Bardzo ciekawym rozwiązaniem jest połączenie wielu urządzeń MODBUS-owych za pomocą sieci światłowodowej. Dzięki temu system może być bardzo rozległy, a zastosowanie topologii pierścienia uodparnia sieć na awarie (na **rys. 1** pokazano przykład sieci Modus, w której urządzenia połączono światłowodem). Ponieważ protokół dopuszcza tylko jednego **mastera**, w sieci może występować tylko jeden komputer lub sterownik nadrzędny odpytujący urządzenia.

Adam Bieńkowski