

---

# WYBRANE PROBLEMY INŻYNIERSKIE

NUMER 2

INSTYTUT AUTOMATYZACJI PROCESÓW TECHNOLOGICZNYCH  
I ZINTEGROWANYCH SYSTEMÓW WYTWARZANIA

---

Krzysztof BOBA, Piotr MICHALSKI\*

Instytut Automatykacji Procesów Technologicznych i Zintegrowanych Systemów  
Wytwarzania, Wydział Mechaniczny Technologiczny, Politechnika Śląska, Gliwice,  
\*piotr.michalski@polsl.pl

## PROBLEM OBSŁUGI URZĄDZEŃ PRACUJĄCYCH W SIECI CAN NA PRZYKŁADZIE INKLINOMETRU

**Streszczenie:** Artykuł przedstawia ogólne właściwości sieci CAN, a także podstawową specyfikację protokołu CANopen. W opracowaniu wyszczególnione są cechy charakterystyczne tego protokołu oraz problemy jakie można napotkać podczas implementacji sieci tej klasy. Przedstawiono problem identyfikacji węzła i adresowania wiadomości procesowych, odczyt zawartych w nich informacji (tzw. PDO-mapping) na przykładzie inklinometru. Na koniec opisano sposób reprezentacji danych na wyświetlaczu pracującego w standardzie sieci CAN.

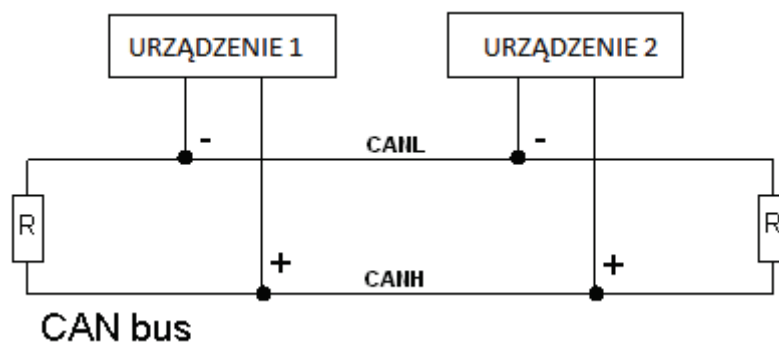
### 1. Wstęp

Opracowanie magistrali CAN ma swój początek w przemyśle samochodowym. Wraz z rozwojem pojazdów i elektroniki świat stanął przed dwoma problemami dotyczącymi rozwoju aut. Pierwszy dotyczył podniesienia komfortu użytkowników, poprzez zastosowanie np. elektrycznego podnoszenia szyb, elektrycznego sterowania lusterkami, podgrzewanych siedzeń, odbiorników GPS, klimatyzacji. Drugi dotyczył kwestii bezpieczeństwa wskutek zaostrzenia norm. Samochody zaczęto więc wyposażać w rozmaite układy elektroniczne: ABS, EBD, ESP, alarmy, czujniki ciśnienia w oponach [1]. Naturalnym skutkiem takich działań jest zwiększenie liczby urządzeń, jakie należy podłączyć do sieci, a więc elementów, które należy monitorować. Każdy obiekt musi posiadać swój unikalny identyfikator, pozwalający odróżnić go od innych obiektów. Zwiększenie liczby wysyłanych danych skutkuje koniecznością powstania coraz lepszych i bardziej niezawodnych mechanizmów ich skutecznej dystrybucji i adresowania. Nie może się zdarzyć, że wiadomość trafia do urządzenia, które nie potrafi jej zinterpretować, lub co gorsze, wprowadzić go w błąd i skutkować nieobliczalnym jego działaniem. W sieci CAN skonstruowano zasady, które niwelują te problemy.

## 2. Ogólna charakterystyka CAN i protokołu CANopen

W sieci CAN wszystkie urządzenia połączone są pojedynczą skrętką pary przewodów, zakończoną na końcach odpowiednimi impedancjami [1].

Podstawą standaryzacji każdej sieci komunikacyjnej jest siedmiowarstwowy model ISO/OSI (*ISO OSI Reference Model*) zapisany w standardzie ISO7498. Standard CAN używa tylko trzech warstw: fizycznej, łącza danych i warstwy aplikacji. Pozostałe warstwy są puste. Model ogólny CAN określa tylko w jaki sposób małe pakiety wiadomości są wysyłane.



Rys.1. Topologia magistrali CAN  
Fig.1. Topology of CAN main line

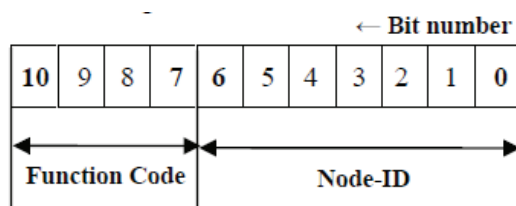
Nie rozwiązuje natomiast problemu przesyłania informacji, które np. są zbyt duże i nie mieszczą się w standardowym polu danych. Zagadnienia te pokrywane są przez protokoły, znajdujące się w siódmej warstwie modelu [2]. Najbardziej rozpowszechnione z nich to: CANopen, DeviceNet, CAN Kingdom oraz CAL.

W Pracowni Sensoryki i Sieci Przemysłowych Wydziału Mechanicznego Technologicznego Politechniki Śląskiej, w ramach pracy dyplomowej magisterskiej zaimplementowano sieć CAN. Oprogramowanie *CAN Studio* sterujące siecią, pracuje w protokole CANopen. Koncepcja tego protokołu opiera się na istnieniu słownika obiektu *OD (Object Dictionary)*, podobnie jak innych popularnych sieci, np. Profibus. *OD* protokołu CANopen jest zorganizowaną listą zawierającą informacje dotyczące urządzenia podłączonego do węzła sieci. Zawiera wszystkie parametry opisujące urządzenie i jego zachowanie w systemie. Każda pozycja tej listy jest adresowana za pomocą 16-bitowego indeksu i 8-bitowego podindeksu, zapisywanych najczęściej w kodzie heksadecymalnym. Każde urządzenie posiada własne *OD* [3]. Można je przeczytać w pliku z rozszerzeniem \*.eds. Głównym zadaniem protokołu CANopen jest oczywiście wymiana danych. Wymiana ta odbywa się bez dodatkowego protokołu i przebiega zgodnie z zasadą nadawca-odbiorca. Oznacza to, że wiadomość transmitowana przez nadawcę może być otrzymana przez jeden lub kilka węzłów (tzw. broadcasting). Wiadomość protokołu CANopen, która zawiera dane procesowe (np. temperaturę, ciśnienie) jest nazywana *PDO (Process Data Object)*. Wysyłane *PDO* oznacza się TxPDO, odbierane jako RxPDO. Nie ma ono określonego formatu: pole danych w ramce może zawierać od jednego do ośmiu, różnie rozmieszczonych bitów (np: jeden *PDO* może przesłać maksymalnie 64 wartości I/O lub 4 16-bitowe wartości analogowe) [3]. Ze względu na to, że wiadomość zawarta w *PDO* nie jest w żaden sposób zstandaryzowana, nie każda stacja może potrafić odczytać zapisane w niej dane. Potrzebna

jest zatem zarówno metoda identyfikacji wiadomości przeznaczonych do konkretnego odbiorcy (stworzenie pewnego identyfikatora) oraz możliwość jej późniejszego odcodowania.

### 3. Identyfikacja wiadomości

Wymiana informacji może być przeprowadzona na dwa sposoby [1]. Pierwszy przez odwołanie się do określonej stacji, czyli tzw. adresowanie stacji – dzięki podaniu adresu odbiornika, nadawca adresuje wiadomość. Drugi przez podanie określonej wiadomości – w tym trybie, zamiast stacji numeruje się wiadomość. Każda ramka otrzymuje swój identyfikator, który definiuje zawartość oraz priorytet [2]. Taka wiadomość jest przeznaczona do kilku stacji odbiorczych dołączonych do magistrali. Każda stacja musi zdecydować, czy przekazywana informacja jest dla niej istotna, czy nie [1]. W protokole CANopen istnieją 3 rodzaje identyfikatorów: *Node ID*, indeksowanie *OD* oraz *COB-ID* [4]. **Node ID** używany przy identyfikacji węzła sieci, dostępny jest w trybie przedoperacyjnym. Każde urządzenie podłączone do systemu posiada swój własny numer. Dopuszczalny zakres wartości to 1-127. Sposób wykorzystywany w mało skomplikowanych systemach w celu ograniczenia czasu poświęcanego na konfigurację [3]. Numer ten jest bezpośrednio związany z 11-bitowym identyfikatorem ramki. Konkretnie na jego wartość składa się siedem pierwszych bitów, po których następują 4 bity zależne od rodzaju wiadomości (tab. 1).



Rys.2. Struktura 11-bitowego identyfikatora w protokole CANopen  
Fig2. Structure of the 11-bit CAN-identifier in the CANopen

W standardzie ten predefiniowany system dystrybuje numery na m.in. cztery RxPDO, cztery TxPDO, jedno *SDO* oraz jeden *EMG* (sygnalizacja błędu) [3].

Tab. 1. Wartości Function Code w zależności od rodzaju wiadomości [5]  
Tab. 1. The function Code values depending on the message type [5]

OBIEKT	FUNCTION CODE
NMT Module Control	0000
SYNC	0001
TIME STAMP	0010
EMERGENCY	0001
first TxPDO	0011
first RxPDO	0100
fourth TxPDO	1001
fourth RxPDO	1010
SDO (transmit)	1011
SDO (receive)	1100

**Indeks i podindeks** słownika obiektu służy do odszukiwania pożądaných zmienných, którymi mogą być dane procesowe lub dane konfiguracyjne, wewnątrz węzła. Identyfikatory **COB-ID** służą głównie do identyfikacji specyficznych wiadomości o wysokim priorytecie oraz obsługi usług np. NMT [4] oraz do wysłania *PDO*. W tym przypadku istnieją dwa sposoby nadania numeru: poprzez użycie predefiniowanych wartości (tab. 2.) lub skonfigurowanie połączenia poprzez *SDO* [3]. Predefiniowane wartości zależą od numeru węzła do jakiego podłączone jest urządzenie. COB-ID powstaje poprzez dodanie tego numeru do wartości zależnej od *PDO*, np. 180 dla pierwszego TxPDO.

Tab. 2. Wartości COB-ID dla różnych typów PDO[5]  
Tab. 2. COB-ID values for different types of PDO [5]

USŁUGA	COB-ID	WIADOMOŚCI
PDO	180 + Node ID	first TxPDO
	200 + NodeID	first RxPDO
	280 + NodeID	second TxPDO
	300 + NodeID	second RxPDO
	380 + NodeID	third TxPDO
	400 + NodeID	third RxPDO
	480 + NodeID	fourth TxPDO
	500 + NodeID	fourth RxPDO
	...	...

Takie rozwiązanie gwarantuje uzyskanie wystarczającej liczby unikalnych identyfikatorów, nawet w najbardziej złożonych systemach, przy dużej liczbie urządzeń.

#### 4. PDO-mapping

Rozwiązanie problemu odcodowania wiadomości kryje się w tzw. *PDO-mappingu*. Opisuje on, jakie dane są wysyłane w danym *PDO*, w jaki sposób są rozmieszczone w wiadomości i jaki jest ich typ oraz długość. Instrukcje te są zapisane w słowniku obiektu (*OD*), dostępnym zarówno po stronie nadawcy jak i odbiorcy. W jaki sposób odczytywać mapowanie pokazano na przykładzie drugiego TxPDO. Zgodnie ze specyfikacją CANopen, PDO to znajduje się w indeksie 1A01.

Wydruk przedstawia fragment pliku \*.eds inklinometru IFM CR 2101. Najważniejsze dane zebrano w tab. 3.

```
[1A01]
SubNumber=0x02
ParameterName=TPDO 2 Mapping
ObjectType=9
ObjFlags=0x00000000

[1A01sub0]
ParameterName=number of mapped objects
ObjectType=7
```

```

AccessType=RO
DataType=0x0005
PDOMapping=0
ObjFlags=0x00000000
DefaultValue=0x01

[1A01sub1]
ParameterName=1 mapped object
ObjectType=7
AccessType=RO
DataType=0x0007
PDOMapping=0
ObjFlags=0x00000000
DefaultValue=0x20200010
-----
[2020]
ParameterName=slope cross (Y)
ObjectType=7
AccessType=RO
DataType=0x0003
PDOMapping=1
ObjFlags=0x00000000

```

*Tab. 3. Najważniejsze parametry indeksu 1A01*

*Tab. 3. The most important parameters of 1A01 index*

1A01 – second TxPDO		
PODINDEKS	WARTOŚĆ	ZNACZENIE
0	1	PDO zawiera jedną wartość
1	0x20200010	Indeks 2020 podindeks 00, składa się z 16 bitów

Indeks 1A01 w nazwie parametru informuje, że tutaj znajduje się sposób zmapowania drugiego TxPDO. 1A01 posiada pod sobą dwa podindeksy. Wartość w podindeksie 0 określa, ile pojedynczych wartości zawiera PDO. W tym przypadku jest to jedna wartość (nachylenie w osi Y). Podindeks 1 zawiera informację o tym gdzie znajduje się ta wartość i w jaki sposób jest zorganizowana. Pierwsze cztery cyfry określają numer indeksu, następne dwie: numer podindeksu, ostatnie dwie definiują zawartość 16 bitów. W nich zapisana jest wartość nachylenia. Biblioteka zawiera także jej znak oraz odchylenie w drugiej osi. Zapisane są w innych indeksach, a ich odczytanie jest analogiczne.

## 5. Obsługa wyświetlacza

Po wysłaniu wiadomości w których znajdują się zmierzone przez czujnik wartości procesu, odpowiednim jej zaadresowaniu do węzła odbiorczego, a następnie odkodowaniu zawartych informacji, konieczne może być ich wyświetlenie, w celu poinformowania użytkownika o przebiegu obserwowanego procesu. Wyświetlacz pracujący w sieci CAN jest traktowany

jak każdy inny węzeł. Potrzebne jest określenie jego numeru identyfikacyjnego, a także wczytanie pliku \*.eds, w którym znajdują się wszystkie parametry oraz opis jego zachowania. Każde urządzenie tego typu posiada własny język programowania (graficzny lub słowny) umożliwiający tworzenie tekstu, menu i podstron, a także grafiki i wykresów, które następnie pokazują się na ekranie. Urządzenie wykorzystywane w pracowni posiada środowisko graficzne *ecolog*. Niezależnie jednak od rodzaju języka, istnieją zawsze co najmniej 2 najważniejsze rodzaje zmiennych możliwych do wykorzystania: wewnętrzne i zewnętrzne. Zmiennymi wewnętrznymi są np. data lub godzina. Wprowadzenie ich do strony powoduje wyświetlenie np. aktualnego czasu. Drugi typ zmiennych oczekuje od urządzeń zewnętrznych podania wartości. Ten rodzaj jest wykorzystywany do reprezentacji danych przesyłanych w PDO. Zgodnie z tym co napisano powyżej, wartości są przechowywane pod odpowiednimi indeksami słownika obiektu. Jako zmienną wyświetlaną należy podać numer indeksu parametru, który chcemy przedstawić na ekranie.

## 6. Podsumowanie

Standard CANopen jest szeroko stosowany w automatyce przemysłowej. Pomysłowość jego działania w kwestiach tak fundamentalnych jak identyfikacja elementów czy sposób wysyłania danych wpływa na jego niezawodność. Jednocześnie w połączeniu z jego nadzwyczajną zdolnością wykrywania wielu błędów podczas transmisji danych i odpowiedniego reagowania na nie powodują, że uzyskano sieć bezpieczną przy jednocześnie dużej szybkości transmisji.

### Literatura

1. Magistrala CAN. „Elektronika Praktyczna” 2000.
2. Komputerowa sieć przemysłowa typu CAN. Instytut Informatyki, Zespół Przemysłowych Zastosowań Informatyki. Gliwice, 2005
3. Boterenbrood H.: CANopen high-level protocol for CAN-bus. Amsterdam, 2000
4. <http://www.ars-informatica.com>, 15.03.2011
5. <http://www.can-cia.org/>, 05.04.2011

## SERVICE OF DEVICES WORKING IN CAN NETWORK PROBLEM ON EXAMPLE OF INCLINOMETER

**Summary:** The article presents the general characteristics of the CAN network as well as the basic specification of the CANopen protocol. The paper lists the characteristics of this protocol and problems they may encounter during the implementation of this network. In the article a problem of node identification, addressing of processes message and read of contained information were presented. At the end authors described how to represent data on the display working in the CAN.