

Zygmunt KUBIAK
Politechnika Poznańska

REALIZACJA PROSTEJ SIECI MIEJSCOWEJ CAN

Streszczenie. W referacie przedstawiono realizację prostej sieci CAN zaprojektowanej do celów edukacyjnych. Omówiono warstwę fizyczną, warstwę łącza danych oraz warstwę aplikacji. Węzły realizują takie funkcje, jak kontrola czasu, proste jednobitowe wyjścia, programowane wejścia-wyjścia cyfrowe, wejścia i wyjścia analogowe, pomiar temperatury i cyfrowe wyświetlanie informacji. Wszystkie węzły zbudowane zostały z wykorzystaniem autonomicznego kontrolera sieci CAN (SJA1000), który realizuje podstawowe funkcje warstwy łącza danych.

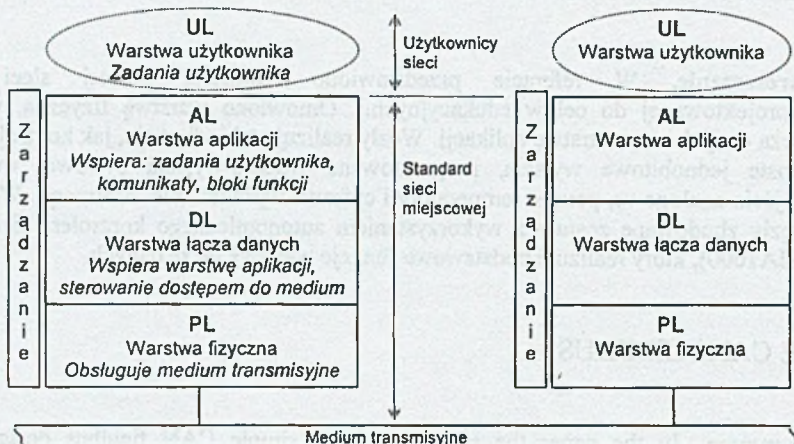
SIMPLE CAN FIELDBUS

Summary. In the paper the realisation of a simple CAN fieldbus designed for educational purposes is presented. The physical layer, the data link layer and the application layer are discussed. The set of exemplary functions of the nodes consists of time control, simple one-bit outputs, programmable digital input-output, analog input and output, temperature measurement and digital display. All nodes have been built using autonomous CAN network controller (SJA1000) that performs basic functions of the data link layer.

1. Wprowadzenie do sieci CAN

Cechą nowych rozwiązań systemów sterowania jest rozproszona struktura układów wejścia/wyjścia, tzw. sieć miejscowa (ang. fieldbus), łącząca inteligentne węzły. Model sieci miejscowej (rys. 1) zasadniczo obejmuje tylko trzy warstwy (1, 2 i 7) znanego modelu odniesienia OSI, opracowanego dla sieci rozległych. Ponadto sieci miejscowe z założenia są sieciami czasu rzeczywistego, a ponieważ przeznaczone są do pracy w warunkach przemysłowych, muszą charakteryzować się zwiększoną odpornością na zakłócenia.

Dla potrzeb pojazdów w firmie Robert Bosch GmbH specjalnie opracowana została sieć CAN (ang. Controller Area Network). Jest to sieć o konfiguracji magistralowej typu M-M (ang. multi-master), to znaczy, że poszczególne węzły sieci mogą jednocześnie żądać dostępu do magistrali. W roku 1992 sieć CAN została zatwierdzona przez ISO jako standard międzynarodowy w normach ISO 11898 oraz ISO 11519-2. Standard CAN dotyczy głównie warstwy drugiej (DL) oraz częściowo warstwy pierwszej (PL), przy czym pierwszy z dokumentów odnosi się do rozwiązań o dużych prędkościach transmisji a drugi - małych.



Rys. 1. Model sieci miejscowej

Fig. 1. The fieldbus model

Ponieważ standard się sprawdził i zyskał bardzo dużą popularność, powstało wiele innych standardów definiujących pozostałe warstwy modelu sieci miejscowej.

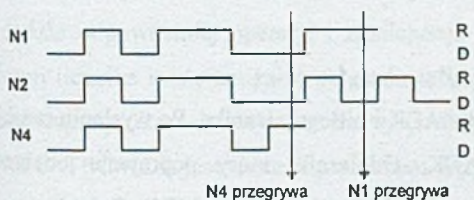
Warstwa fizyczna dotyczy technicznych aspektów przesyłania danych w postaci strumienia bitów. Opracowano wiele rozwiązań bazujących na standardzie CAN, które precyzują założenia dotyczące między innymi tej warstwy. Dla sieci CAN najczęściej stosowanym medium transmisyjnym jest para skręconych przewodów, tworząca magistralę dla dołączanych węzłów. Jedna z jej linii oznaczana jest CAN-H, a druga CAN-L. Magistrala z obu stron powinna być zakończona dopasowującymi impedancjami (terminatory - ok. 120Ω). Wyjścia układów sterujących magistralą mogą znajdować się w trzech stanach logicznych: stan wysokiej impedancji, stan dominujący i stan recesywny. Stan dominujący oznacza, że różnica napięć CAN-H - CAN-L wynosi nie mniej niż 0,9V, a dla stanu recesywnego napięcie różnicowe nie przekracza 0,5V. W przypadku jednoczesnego podania bitu dominującego i bitu recesywnego przez nadajniki dwóch różnych węzłów na magistrali

ustali się poziom dominujący. Opracowanych zostało wiele monolitycznych układów nadawczo-odbiorczych realizujących styk z fizyczną magistralą, jak: CF150B (Bosch), MTC3054 (Alcatel Mietec), PCA82C250, PCA82C251, PCA82C252, TJA1053 (Philips), TLE6252G (Siemens), Si9200EY, B10011S (Temic Siliconix), SN75LBC031 (Texas) itd.

Strumień bitów w sieci CAN kodowany jest zgodnie z metodą NRZ. Oznacza to, że może wystąpić wielobitowy stan dominujący lub recesywny, a to utrudnia poprawną synchronizację bitową poszczególnych węzłów. W związku z tym przyjęta została zasada tzw. *szpikowania bitami*, która oznacza, że po każdym pięciu jednakowych bitach następuje wstawienie bitu o przeciwnej polaryzacji.

Czas trwania pojedynczego bitu podzielony został na cztery segmenty, których długość jest całkowitą wielokrotnością *kwantu czasu* t_q . Podstawowym celem tego podziału jest ustalenie właściwego momentu próbkowania bitu. W każdym węźle musi istnieć możliwość ustawiania czasu trwania bitu od 8 do 25 kwantów czasu. W CiA (ang. CAN in Automation) przygotowano specyfikację wiążącą prędkość transmisji, długość magistrali z liczbą kwantów czasu na bit i położeniem momentu odczytu.

Warstwa łącza danych zgodnie ze specyfikacją Boscha dzieli się na dwie podwarstwy: obiektową i transferową. Do zadań podwarstwy obiektowej należą filtracja wiadomości oraz obsługa wiadomości i statusu. Podwarstwa transferu zajmuje się przygotowaniem ramki wiadomości, zatwierdzaniem i rozpakowaniem wiadomości, detekcją i sygnalizacją błędów, arbitrażem oraz prędkością transmisji i zależnościami czasowymi. CAN jest protokołem typu M-M (ang. multi-master), to znaczy, że poszczególne węzły sieci



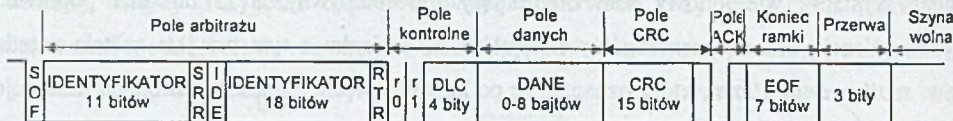
Rys. 2. Arbitraż między węzłami N1, N2 i N4

Fig. 2. Arbitration between N1, N2 and N4 nodes

mogą jednocześnie żądać dostępu do magistrali. Taka sytuacja wymaga arbitrażu. Przyjęto modyfikację metody CSMA/CD (ang. Carrier Sense Multiple Access with Collision Detect), polegającą na tzw. nieniszczącym arbitrażu, który oznacza, że w przypadku wystąpienia kolizji dostęp do magistrali uzyskuje wiadomość o najwyższym priorytecie.

Wiadomości przekazywane są za pomocą ramek w formacie standardowym (CAN 2.0A) lub w formacie rozszerzonym (CAN 2.0B). Różnią się identyfikatorem - 11 lub 29 bitów (rys. 3), co pozwala na rozróżnienie bardzo dużej liczby wiadomości. Ograniczenie

wynika jedynie z warstwy fizycznej. W praktyce znaczna długość identyfikatorów pozwala na kodowanie dodatkowych informacji dotyczących wiadomości lub węzłów (typ wiadomości, rodzaj sensora czy np. kod producenta). W sieci CAN nie ma adresowania konkretnych węzłów, a każda wysłana wiadomość może być odebrana przez dowolny moduł. Akceptacja danych zależy wyłącznie od ustawienia układu wyboru w odbiorniku węzła.



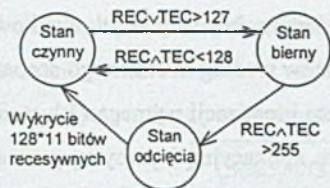
Rys. 3. Ramka formatu rozszerzonego (CAN 2.0B)

Fig. 3. Extended frame format (CAN 2.0B)

Ramka rozpoczyna się od bitu dominującego SOF. Dalej następuje pole arbitrażu, które jest wykorzystywane przy rozstrzygnięciu ewentualnych konfliktów z dostępem do sieci oraz służy do identyfikacji wiadomości w sieci. Bit RTR rozróżnia ramkę z danymi (bit dominujący) od ramki zdalnej (bit recesywny). Format ramki określany jest przez bit IDE (bit dominujący - ramka standardowa, bit recesywny - ramka rozszerzona). Bit SRR wysyłany jest jako recesywny. Bity rezerwowe r0 i r1 są ustawione jako dominujące. Pole kontrolne definiuje liczbę bajtów danych. Zawartość DCL w zakresie 0..7 bezpośrednio równa jest liczbie bajtów, natomiast wartość 8..15 oznacza 8 bajtów. To umożliwi kodowanie dodatkowych informacji. Szczególne znaczenie ze względu na wiarygodność danych ma pole CRC. Bit 16 kończący to pole (ogranicznik) jest recesywny. Kontrola dla pól od SOF do pola danych (jeśli występuje) włącznie, realizowana jest za pomocą kodu cyklicznego o następującym wielomianie generującym:

$$g(x) = x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1$$

Pole potwierdzenia (ACK) zawiera 2 bity: ACK i bit ogranicznika. Po wysłaniu ramki nadajnik wysyła 2 recesywne bity pola ACK. Odbiornik, który poprawnie odebrał wiadomość, odpowiada w tym samym czasie dominującym bitem ACK. Dzięki temu nadajnik uzyskuje potwierdzenie, że przynajmniej jeden węzeł poprawnie odebrał wiadomość. Pole EOF składa się z 7 recesywnych bitów kończących ramkę. W ramce standardowej (CAN 2.0A) nie występują pola SRR, r1 oraz 18 bitów rozszerzenia identyfikatora. Odbiornik może zażądać potrzebnej wiadomości wysyłając tzw. ramkę zdalną. W tym przypadku identyfikator określa źródło danych. W odpowiedzi źródło powinno wysłać żadaną wiadomość. Ważnym zadaniem warstwy łącza danych jest obsługa błędów. Gdy



Rys. 4. Stany węzłów przy obsłudze błędów

Fig. 4. Nodes states with respect to error handling

Wykrywane są następujące typy błędów: stanu bitu, szpikowania bitami, cyklicznej kontroli nadmiarowej (CRC), potwierdzenia (ACK), formatu ramki. Węzły przeprowadzają kontrolę stanu bitów. Błąd stanu bitu występuje, jeśli nadajnik wyśle bit dominujący, a odbierze recesywny albo wyśle recesywny, a odbierze dominujący. Wyjątkiem od tej zasady jest sekwencja arbitrażu. Błąd szpikowania zachodzi wtedy, gdy w ciągu bitów między SOF a ogranicznikiem CRC pojawi się więcej niż 5 takich samych bitów - odbiornik stwierdza naruszenie zasady szpikowania. Odbiornik na podstawie ciągu CRC sprawdza czy w ramce nie nastąpiło przekłamanie. Mimo znacznej odległości Hamminga równej 6 kod wykorzystywany jest wyłącznie do wykrywania błędów a nie do korekcji. Odbiornik, który poprawnie odbierze dane, ustawia bit ACK jako dominujący. W ten sposób nadajnik uzyskuje potwierdzenie poprawnej transmisji; w przeciwnym przypadku nadajnik wysyła ramkę błędu. W ramce występują bity, które mają znane wartości. Zmiana tych bitów oznacza błąd formatu ramki. W celu odróżnienia błędów chwilowych od trwałych kontrolery CAN wyposażone są w dwa liczniki: licznik błędów odbioru (REC - Receive Error Counter) oraz licznik błędów nadawania (TEC - Transmit Error Counter). Liczniki zwiększają swoją zawartość przy błędzie odpowiedniej operacji i zmniejszają ją przy poprawnej operacji. W zależności od stanu licznika może zmieniać się stan węzła. Węzły mogą znajdować się w trzech stanach: stan czynny (error active), stan bierny (error passive) i stan odcięcia (bus off). Stan czynny ($REC=TEC<128$) jest normalnym stanem węzłów - są w pełni aktywne. Gdy sieć działa bez zakłóceń, $REC=TEC=0$. W stanie pasywnym ($127<REC\vee TEC<255$) węzły nadal nadają i odbierają wiadomości. Różnica polega na tym, że w przypadku błędu węzeł czynny wysyła czynną ramkę błędu (z czynną flagą błędu - 6 bitów dominujących), a węzeł bierny wysyła bierną ramkę błędu (z bierną flagą błędu - 6 bitów recesywnych). Przekroczenie zawartości $REC\vee TEC>255$ powoduje przejście węzła do stanu odcięcia. Zależności między stanami ilustruje rys. 4.

którykolwiek z węzłów wykryje błąd, generowana jest specjalna ramka błędu, zawierająca flagę błędu (6 bitów dominujących). Bity dominujące flagi błędu powodują nadpisanie uszkodzonej ramki i wymuszają powtórzenie transmisji. Każdy z węzłów przeprowadza kontrolę bitów.

Warstwa aplikacji dotyczy bezpośredniego świadczenia usług użytkownikowi. W przypadku sieci miejscowych oznacza to wysłanie rozkazów według określonego scenariusza w celu pozyskiwania informacji z nadzorowanego obiektu i realizacji wymaganych sterowań. Na bazie CAN powstały standardy definiujące warstwę aplikacyjną i precyzujące warstwę fizyczną. Przykładowo SDS (ang. Smart Distributed System) opracowany został w firmie Honeywell, znanego producenta systemów nadzorczych. Standard SAE J1850 przeznaczony jest do zastosowań w samochodach osobowych, a SAE J1939 - dla ciężarówek, autobusów i ciężkich pojazdów specjalnych. W automatyzacji przemysłu popularność zyskały również DeviceNet, CANopen i CAN Kingdom.

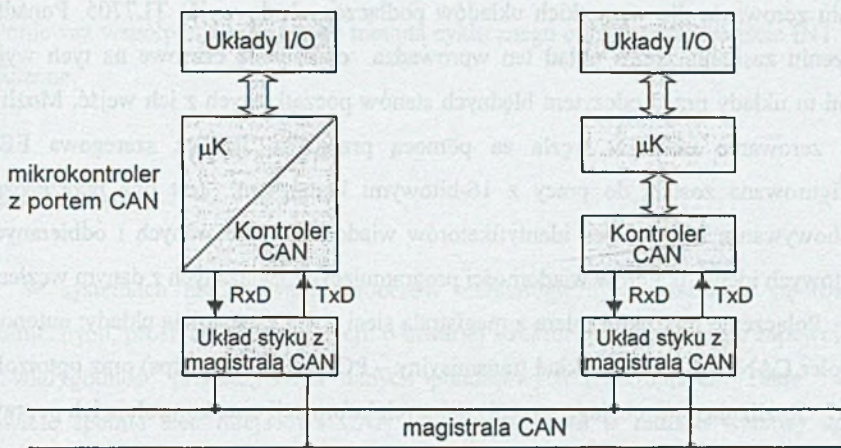
Duża popularność standardu CAN przyczyniła się do opracowania wielu kontrolerów sieci CAN. Realizują one zadania warstwy łącza danych oraz w różnym stopniu warstwy aplikacyjnej. Powstało kilka architektur kontrolerów. BasicCAN mają sprzętowo rozwiązaną logikę niezbędną do tworzenia i weryfikacji zależnego od protokołu strumienia danych. Sterowniki te zapewniają filtrację ograniczoną do 8 najstarszych bitów identyfikatora. FullCAN zawiera pamięć danych (obiektów CAN) oraz dłuższe rejestry. PeliCAN - dodatkowe możliwości, jak: monitorowanie magistrali CAN bez potwierdzania i sygnalizacji błędów, obsługa błędów pod przerwaniem, ustawianie/odczyt liczników błędów itd.

2. Realizacja modelu sprzętowego sieci CAN

Kontrolery możemy podzielić na dwie podstawowe grupy: specyficzne - realizujące wyłącznie zadania CAN oraz uniwersalne, bazujące na znanych mikrokontrolerach wyposażonych w kanały CAN. W pierwszej grupie występuje kilkanaście układów. Do najpopularniejszych należą SJA1000 (Philips), AN82257 (Intel), a także SAE81C91 (Siemens). Druga grupa układów jest znacznie bardziej liczna. Praktycznie każdy z producentów mikroprocesorów ma w swojej ofercie co najmniej jeden uniwersalny mikrokontroler z kanałami CAN. Przykłady: PIC16C185 (Microchip), 8X196CA (Intel), M37632MF (Mitsubishi), SABC164 (Siemens) czy P87C592 (Philips).

Na rys. 5 przedstawiono dwa ogólne rozwiązania węzłów sieci CAN. Jedno wykorzystuje uniwersalny mikrokontroler z dostępnymi portami CAN. Jest to rozwiązanie szczególnie korzystne dla dużej serii węzłów. W konstrukcjach jednostkowych, prototypowych lepsza jest druga konfiguracja, oparta na specjalizowanych kontrolerach CAN

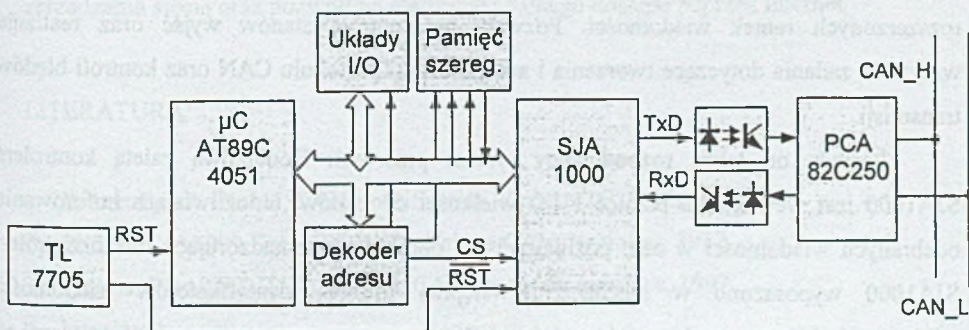
i dowolnych, uniwersalnych mikrokontrolerach realizujących zadania warstwy aplikacyjnej. W systemie omawianym w referacie przyjęto drugie rozwiązanie, pozwalające na zastosowanie dowolnego mikrokontrolera, znanego od strony programowania.



Rys. 5. Typowe struktury węzłów sieci CAN

Fig. 5. Typical structures of CAN fieldbus nodes

Głównymi elementami każdego węzła sieci (rys.6) są: mikrokontroler AT89C4051 (Atmel) oraz kontroler CAN typu SJA1000 (Philips). Port P1 mikrokontrolera wykorzystywany jest jako dwukierunkowa, przełączana szyna adresowa/danych. Linie portu P3 sterują transmisją na tej szynie oraz poprzez dekodery adresów (74ALS138) służą do wyboru pozostałych aktywnych układów węzła (SJA1000, pamięć EEPROM – NM93C56N oraz opcjonalny układ I/O). Poprzez port P3 obsługiwana jest również transmisja szeregową z pamięcią EEPROM i cyfrowym przetwornikiem temperatury DS1620. Do połączenia z zewnętrznymi układami I/O zastosowano 20-stykowe złącze, na które wyprowadzono sygnały magistrali oraz zasilanie.

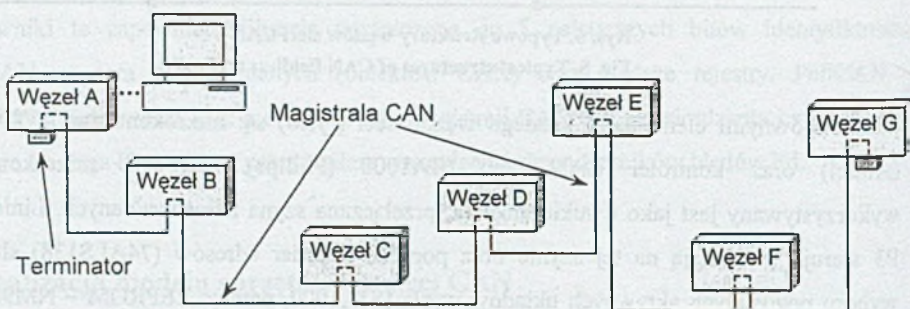


Rys. 6. Schemat blokowy węzła sieci CAN

Fig. 6. Block diagram of CAN fieldbus node

Węzeł zasilany jest napięciem +5V, którego wartość jest sprawdzana przez układ kontroli napięcia TL7705. Obniżenie napięcia poniżej 4,65V powoduje wygenerowanie sygnału zerowania dla wszystkich układów podłączonych do wyjść TL7705. Ponadto przy załączeniu zasilania węzła układ ten wprowadza opóźnienie czasowe na tych wyjściach. Chroni to układy przed odczytem błędnych stanów początkowych z ich wejść. Możliwe jest także zerowanie układów węzła za pomocą przycisku. Pamięć szeregową EEPROM skonfigurowana została do pracy z 16-bitowymi komórkami. Jest ona przeznaczona do przechowywania 29-bitowych identyfikatorów wiadomości nadawanych i odbieranych oraz 11-bitowych identyfikatorów wiadomości programujących związanych z danym węzłem.

Połączenie mikrokontrolera z magistralą sieci CAN zapewniają układy: autonomiczny kontroler CAN – SJA1000, układ transmisyjny – PCA82C250 (Philips) oraz optoizolatory – 6N137. Realizują one obsługę przepływających informacji oraz kontrolują ich poprawność.



Rys. 7. Sieć CAN - poprawne łączenie węzłów

Fig. 7. CAN fieldbus - correct connection of nodes

Wprowadzona separacja galwaniczna zabezpiecza pozostałe układy węzła przed różnicami potencjałów i przepięciami z magistrali. Układ SJA1000 umożliwia obsługę standardowych i rozszerzonych ramek wiadomości. Pozwala na kontrolę stanów wyjść oraz realizuje wszystkie zadania dotyczące tworzenia i analizy ramki protokołu CAN oraz kontroli błędów transmisji.

Posiada on także rozbudowany system przerwań. Dodatkową zaletą kontrolera SJA1000 jest wewnętrzna pamięć FIFO wielkości 64 bajtów, umożliwiająca buforowanie odebranych wiadomości w celu późniejszej ich obróbki przez nadzorujący mikrokontroler. SJA1000 wyposażono w mechanizm wstępnej filtracji identyfikatorów wiadomości odbieranych, który pozwala na odrzucenie wiadomości nie wykorzystywanych przez węzeł co wyraźnie zmniejsza obciążenie mikrokontrolera. Układ może pracować w najbardziej

rozbudowanym trybie, tzn. PeliCAN, który został zastosowany w omawianej aplikacji. Z dostępnych dwóch kanałów CAN wykorzystano tylko jeden. Do komunikacji z mikrokontrolerem służy 8-bitowa szyna danych oraz cztery linie sterujące: WR, RD, ALE i CS. Ponieważ współpraca odbywa się metodą cyklicznego odpytywania, wyjście INT nie jest przyłączone.

3. Zakończenie

W systemach nadzorowania procesów technologicznych poszukuje się rozwiązań ekonomicznych, prostych w eksploatacji, o otwartej strukturze i jednocześnie zapewniających dużą wiarygodność przekazywania danych pomiarowych i sterujących. Takie warunki całkowicie spełnia sieć miejscowa CAN i jej rozwinięcia w ramach warstwy aplikacji. Szeroko dostępna dokumentacja związana z CAN, dostępne i tanie układy transmisyjne, kontrolery CAN oraz mikrokontrolery z kanałami CAN decydują o popularności sieci. Według CiA w roku 2000 sprzedano ponad 100 mln kontrolerów CAN, a na rok 2003 przewidywane jest potrojenie tej liczby. Największy wzrost jest zauważalny w grupach uniwersalnych mikrokontrolerów 16- i 32 - bitowych z kanałami CAN. Praktycznie każdy producent sterowników programowalnych PLC oferuje urządzenia typu most lub brama do współpracy z siecią CAN. W tym kontekście sprzętowy model sieci pozwalający na znaczną ingerencję w parametry i jego strukturę posiada zauważalne walory edukacyjne. Przewidywana jest rozbudowa tego zestawu o kolejne węzły, a dalej oprogramowanie styku szeregowego, co umożliwi nadzór i monitorowanie sieci z komputera PC. Zestaw ma być uzupełniony również o kartę styku CAN do komputera PC, co zwiększy możliwości zarządzania siecią oraz pozwoli na realizację zdalnego dostępu poprzez Internet.

LITERATURA

1. Robert Bosch GmbH, CAN Specification Version 2.0, Stuttgart 1991.
2. Philips, SJA1000 Data Sheet, Stand-alone CAN controller, 1997.
3. Philips, PCA82C250 Data Sheet, CAN controller interface, 1997.
4. <http://www.c@n.com>

Recenzent: Dr inż. Ryszard Jakuszewski

Abstract

The CAN (Controller Area Network) fieldbus (CAN) was developed by Robert Bosch GmbH for use in automotive industry as a solution for networking in distributed real-time systems. The efficacy of this solution made this standard very popular in many other domains. CAN is a serial bus with multi master capabilities, it means that several network nodes can request bus access simultaneously. The arbitration process has been defined in the network protocol. In the paper the realization of a simple CAN fieldbus designed for educational purposes is presented. The physical layer, the data link layer and the application layer have been discussed. The network is an autonomous system but the possibility of coupling it with higher supervising layer exists too. Nodes carry out such functions as: time control, simple one-bit outputs, programmable digital input-output, analog input and output, temperature measurement and digital display. All nodes have been built using autonomous CAN controller (SJA1000) that performs basic functions of the data link layer. The AMTEL AT89C4051 controller accomplishes tasks of the application layer. It has a built-in ROM of flash type that enables numerous and easy modifications of node program. The PCA82C250 circuit enables direct access to the bus. Each module contains serial access EPROM memory, where, among others, sent and received messages identifiers are stored. The bitrate for the model is 1 Mb/s. For didactic purposes the possibility of slowing the system bitrate down to 12,5 kb/s is added. It enables data transmission for longer distances and simplifies network monitoring.