

10. Lekcja 9: CAN

10.1. Rys historyczny

Sieć CAN (ang. Controller Area Network) jest międzynarodowym otwartym¹ standardem komunikacyjnym przeznaczonym głównie do stosowania w zadaniach sterowania i pomiarów w technice samochodowej oraz w przemyśle elektromaszynowym. Prace związane z opracowaniem sieci CAN zostały rozpoczęte w 1983 roku przez firmę Bosch. Do prac nad projektem nowej sieci dołączyła wkrótce firma Mercedes-Benz oraz firma Intel. W 1986 roku sieć CAN została oficjalnie zaprezentowana w Detroit na kongresie SAE (Society of Automotive Engineers) pod nazwą: Automotive Serial Controller Area Network. W połowie lipca 1987 roku firma Intel dostarczyła pierwszy zintegrowany układ kontrolera sieci CAN.

10.2. Standaryzacja

We wczesnych latach 90-tych firma Bosch przedłożyła specyfikację protokołu CAN (w wersji 2.0) w celu jej międzynarodowej standaryzacji. W wyniku dyskusji uwzględniającej między innymi sieć VAN (*Vehicle Area Network*) w listopadzie 1993 roku opublikowana została pierwsza norma dotycząca sieci CAN (ISO 11898). W wyniku dalszych prac rozwojowych unormowano sieć CAN tolerującą uszkodzenia (ang. *fault-tolerant*), która została opisana w specyfikacji ISO 11519-2. W roku 1995 norma ISO 11898 została rozszerzona o dodatek opisujący wykorzystanie 29-bitowego identyfikatora CAN.

Aby zapewnić zgodność implementacji sieci CAN, opracowano standardowe testy zgodności opublikowane w normie ISO 16845. W wyniku prowadzonych dalszych prac dokonano weryfikacji pierwotnego brzmienia normy ISO 11898. Obecnie obowiązują następujące normy:

- ISO 11898-1 opisuje warstwę danych
- ISO 11898-2 opisuje warstwę fizyczną
- ISO 11898-3 opisuje warstwę fizyczną odporną na uszkodzenia
- ISO 11992 opisuje interfejs dla samochodów ciężarowych i przyczep

10.3 CAN – charakterystyka ogólna

CAN jest lokalną siecią komunikacyjną typu *fieldbus*, a więc przeznaczoną głównie do zastosowań i realizacji zadań tzw. warstwy obiektowej modelu procesu produkcji i zarządzania. W warstwie obiektowej zlokalizowane są głównie urządzenia pomiarowe i wykonawcze. Ze względu na ekonomikę implementacji - podobnie zresztą jak w większości innych sieci typu *fieldbus* – stosowana jest komunikacja szeregową o maksymalnej prędkości transmisji równej 1Mb/s. Należy zaznaczyć, że prędkość transmisji jest funkcją długości sieci. Ogólnie im większa długość sieci tym mniejsza dopuszczalna prędkość transmisji. Na przykład dla sieci CAN o topologii magistrali maksymalna prędkość transmisji spada do ok. 20kb/s dla sieci o długości 1000m.

Typowymi topologiami sieci CAN są topologie: magistrali lub łańcucha. Nie są natomiast stosowane topologie: rozgałęzionego drzewa, pierścienia, gwiazdy czy siatki.

¹ Specyfikacje: ISO 11898, ISO 11519-2, ISO 11992

Sieć CAN złożona jest z urządzeń sieciowych zwanych węzłami. Węzły połączone są wzajemnie wspólną magistralą. Magistrala umożliwia wymianę informacji w sieci (dwie linie danych) i ewentualne zasilanie urządzeń sieciowych (2 dodatkowe linie zasilania). Sposób wymiany informacji w sieci CAN ma zupełnie odmienny charakter w stosunku do sposobu właściwego dla sieci typu *master-slave* (np. MODBUS RTU czy AS-i). Sieć CAN jest siecią typu rozgłoszeniowego. Oznacza to, że każdy węzeł ma prawo inicjacji transakcji sieciowych. W tym wypadku informacje przesyłane w wyniku realizacji transakcji sieciowych nazywane są wiadomościami. W przypadku sieci typu *master-slave* prawo do inicjacji transakcji sieciowych przysługiwało wyłącznie jednostce lub jednostkom nadrzędnym. Wynika stąd, że w sieci CAN nie ma wyróżnionej jednostki nadrzędnej, lub wszystkie węzły mogą (przynajmniej teoretycznie) taką rolę pełnić. Z tego powodu sieć CAN zaliczana jest do klasy sieci typu *multimaster*. Należy jednak wyraźnie zaznaczyć, że sieć CAN nie wyklucza możliwości wymiany informacji zgodnej z modelem sieci *master-slave*. Zależy to wyłącznie od implementacji warstwy aplikacyjnej. Międzynarodowe normy, o których była mowa w p. 10.2 definiują wyłącznie dwie pierwsze warstwy modelu komunikacyjnego ISO/OSI i nie dotyczą tej warstwy. Na przykład międzynarodowa organizacja CiA (ang. **CAN in Automation**) czuwająca obecnie nad rozwojem sieci zdefiniowała implementację tej warstwy o nazwie CanOpen, a firma Allen-Bradley produkująca sprzęt głównie do zastosowań elektroenergetycznych zdefiniowała implementację o nazwie DeviceNet.

Prawo inicjacji transferów sieciowych przez wszystkie węzły sieci wprowadza możliwość wystąpienia konfliktów (kolizji) informacyjnych na magistrali. Kolizje te mają miejsce wówczas, gdy dwa lub więcej węzłów sieciowych zainicjują jednocześnie transakcje sieciowe. Kolizje sieciowe rozstrzygane są w wyniku realizacji odpowiednich procedur zwanych procedurami arbitrażowymi. W przypadku sieci typu *master-slave* arbitraż nie jest konieczny, ponieważ organizacja wymiany informacji w takiej sieci nie pozwala na występowanie kolizji. Jeśli kolizje takie wystąpią w wyniku np. zakłóceń, to są rozstrzygane np. przez mechanizmy retransmisyjne realizowane w warstwie aplikacyjnej. W przypadku sieci o charakterze rozgłoszeniowym konflikty rozstrzygane są zasadniczo na dwa sposoby.

Pierwszy sposób (CSMA/CD²) polega na unieważnieniu bieżącej transakcji i przydzieleniu węzłom będącym w konflikcie losowych czasów oczekiwania na możliwość reinicjacji unieważnionej transakcji. Istnieje wówczas stosunkowo wysokie prawdopodobieństwo uniknięcia ponownego konfliktu. Niemniej jednak, istnieje również prawdopodobieństwo powstania konfliktu z informacjami generowanymi przez nowe węzły sieci w czasie reinicjalizacji transakcji wcześniej skonfliktowanych węzłów. Taki sposób rozwiązywania konfliktów nazywamy arbitrażem niszczącym, a sieć wykorzystująca ten sposób arbitrażu nie może być uznana za sieć deterministyczną.

Drugi sposób (CSMA/CA³) rozwiązywania konfliktów, zwany arbitrażem nieniszczącym polega na jednoznacznym rozstrzygnięciu w początkowym okresie trwania konfliktu, które z urządzeń komunikacyjnych będących w konflikcie uzyska

² CSMA/CD (ang. **C**arrier **S**ense **M**ultiple **A**ccess / **w**ith **C**ollision **D**etect) = protokół wielodostępu CSMA ze śledzeniem stanu dostępności medium transmisyjnego i wykrywaniem kolizji

³ CSMA /CA (ang. **C**arrier **S**ense **M**ultiple **A**ccess /**w**ith **C**ollision **A**voidance) = protokół wielodostępu do magistrali ze śledzeniem stanu nośnika i unikaniem kolizji

prawo do dalszej transmisji, a które urządzenia muszą wycofać żądanie rozpowszechniania własnych wiadomości. Taki sposób rozwiązywania konfliktów sprzyja zwiększeniu sprawności informacyjnej sieci, a także determinizmowi sieci. Sieć CAN korzysta z metody arbitrażu nieniszczącego.

Istotną cechą sieci rozgłoszeniowych jest zmiana sposobu sygnalizacji źródła pochodzenia informacji. W przypadku sieci typu monomaster każdemu urządzeniu podrzędnemu nadawany jest unikalny adres. Jednostka nadrzędna zwraca się z odpowiednim rozkazem do jednostki podporządkowanej wskazując jej adres. Jednostka nadrzędna musi zatem prowadzić listę adresów urządzeń podporządkowanych. Urządzenie podporządkowane odpowiada na zapytanie lub rozkaz jednostki nadrzędnej umieszczając w ramce odpowiedzi swój własny adres jako wskaźnik źródła pochodzenia informacji. Ponieważ w sieci CAN wszystkie urządzenia sieciowe mają prawo inicjowania transakcji sieciowych, to ważne jest nie to, która jednostka jest źródłem informacji, ale sama informacja jako taka. Dla porządku informacjom przypisano odpowiednie identyfikatory. Identyfikatory te często są mylnie utożsamiane z adresami. Co do zasady, identyfikatory wiadomości w sieci CAN muszą być unikalne. Identyfikatorem jest opatrywana każda wiadomość. Odgrywa ona rolę priorytetu wiadomości. Im niższa wartość binarna identyfikatora, tym wyższy jest priorytet wysyłanej informacji. Procedura arbitrażu jest realizowana w czasie transmisji identyfikatora. Prawo kontynuacji transakcji sieciowej uzyskuje węzeł generujący wiadomości o najniższej wartości binarnej identyfikatora. Ze względów technicznych dla skutecznej realizacji procedury arbitrażu konieczne jest stosowanie w CAN trybu transmisji synchronicznej.

Konstrukcja ramki danych CAN ma w zasadzie organizację bitową. Oznacza to, że podstawowymi elementami składowymi ramki CAN są pola bitowe. Od tej zasady istnieje jednak wyjątek – jest nim pole danych o organizacji bajtowej. Maksymalna długość pola danych w sieci CAN jest ograniczona do zaledwie 8 bajtów. Jest to znacząco mniej w porównaniu do innych sieci. Na przykład w przypadku sieci MODBUS RTU maksymalna długość pola danych wynosi 256 bajtów, w sieci PROFIBUS PA, DP i FMS – 244 bajty. Należy stąd wnioskować, że sieć CAN jest siecią przeznaczoną do efektywnego (organizacja bitowa), przesyłania stosunkowo krótkich informacji (0..8 bajtów).

Obecnie, w skali globalnej około 40 producentów układów scalonych dostarcza gotowe układy realizujące sprzętowo wymagania specyfikacji CAN na poziomach: warstwy fizycznej i warstwy wymiany danych. Pozwala to na skoncentrowanie uwagi konstruktorów węzłów CAN na implementacji warstwy aplikacyjnej modelu ISO/OSI. W związku z tym, praktycznie jedynymi zadaniami użytkownika jest realizacja przepływu strumieni informacji. Wszystkie inne niezbędne procedury związane z realizacją protokołu komunikacyjnego (takie jak np. kompozycja ramki, detekcja błędów czy retransmisja błędnych ramek) są przeprowadzane automatycznie przez sprzęt bez ingerencji użytkownika.

Podsumowując, podstawowymi cechami sieci CAN są:

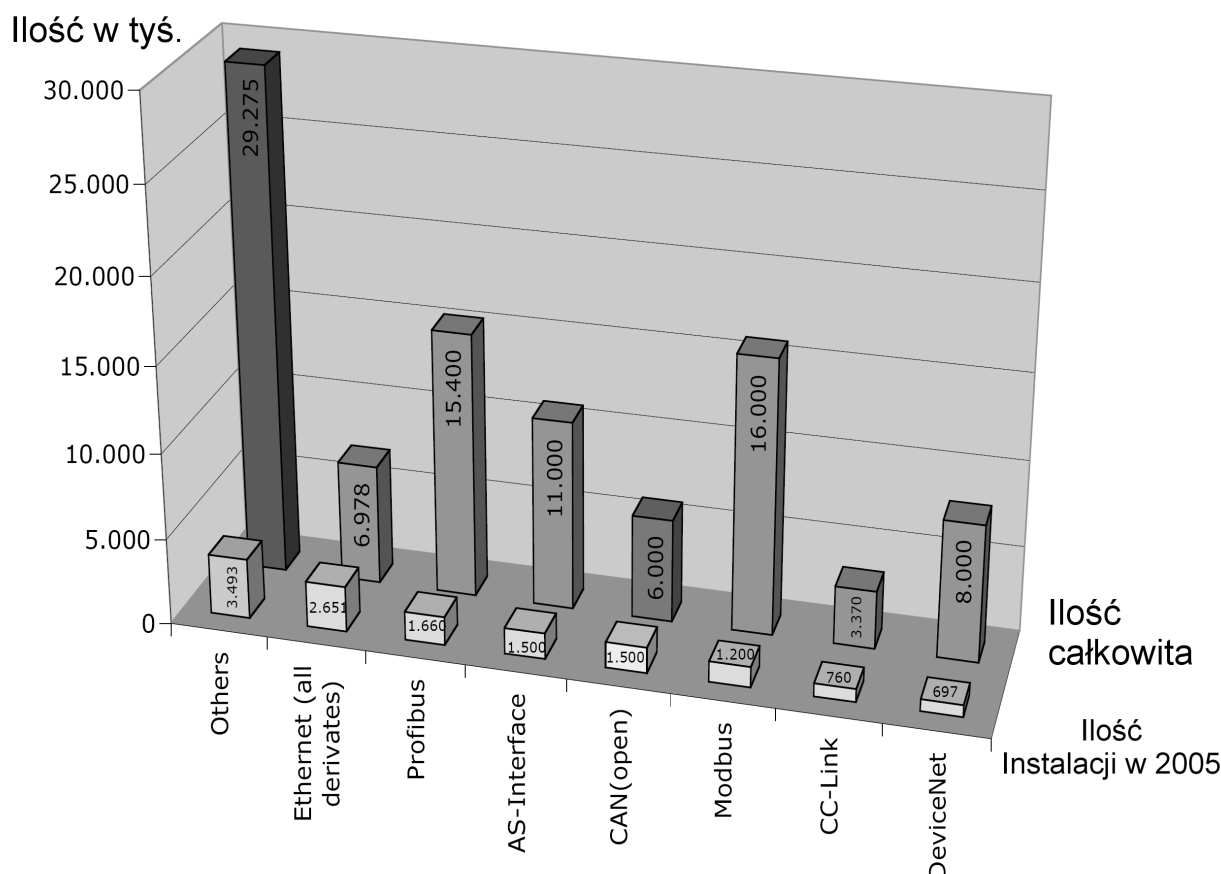
- organizacja typu multimaster. Organizacja tego typu pozwala na budowę sieci o właściwościach redundancyjnych, w której awaria jednego z węzłów nie powoduje blokady pracy sieci całej sieci,

- ❑ komunikacja rozgłoszeniowa (ang. *broadcast*). Każda wiadomość pojawiająca się w sieci jest dostępna dla wszystkich urządzeń sieciowych. To urządzenia sieciowe decydują o wykorzystaniu tej wiadomości lub decydują czy wiadomość jest skierowana do nich,
- ❑ zaawansowane metody detekcji błędów. Metody te gwarantują spójność i poprawność transmitowanych i odbieranych wiadomości.

10.4 Rozwój sieci CAN

Ze względu na otwarty charakter sieci oczywistą stała się potrzeba powołania organizacji, która będzie czuwać nad jej rozwojem. W 1992 powstała organizacja zrzeszająca użytkowników sieci CAN o nazwie CiA (ang. *CAN in Automation*). Głównym celem organizacji jest promowanie i rozwój sieci. Obecnie organizacja zrzesza około 460 użytkowników i producentów sieci CAN.

Na zlecenie organizacji CiA firma IMS Research wykonała w 2007 roku badania liczby zainstalowanych węzłów różnych rodzajów sieci. Na rys. 1 uwidoczniony jest wynik tej analizy.



Rys. 10.1 Liczba zainstalowanych węzłów różnych systemów sieciowych w samym 2005 roku i ich ogólna liczba według stanu z 2005 roku

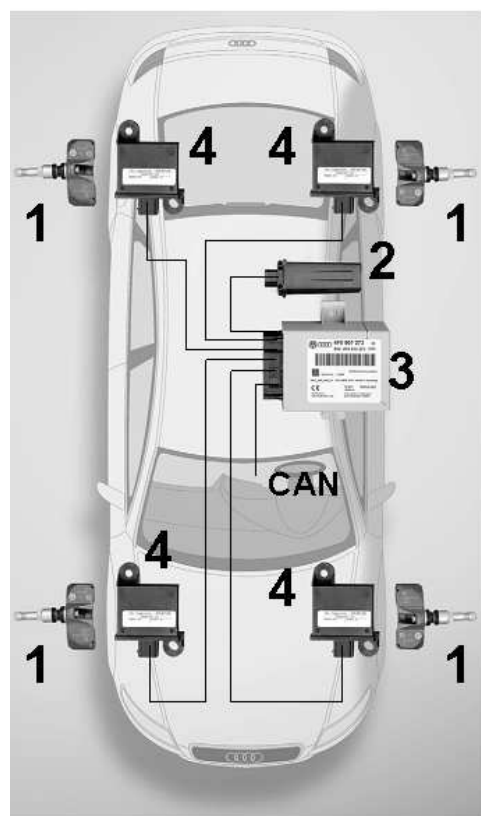
Wykres uwidacznia całkowitą ilość instalacji w skali globalnej bez podziału na poszczególne dziedziny przemysłu. Sieci wykorzystujące specyfikację CAN

(CANopen, DeviceNet) zajmują znaczące miejsce w ogólnej liczbie zainstalowanych węzłów. Liczba zainstalowanych węzłów sieci wykorzystujących CAN w 2005 roku jest tylko nieco ustępuje liczbie węzłów sieci Ethernet. Głównym powodem ekspansji sieci CAN jest jej powszechne zastosowanie w przemyśle samochodowym.

10.5 Przykład zastosowania sieci CAN

Utrzymywanie ciśnienia powietrza w oponach na odpowiednim poziomie jest ważnym elementem wpływającym na bezpieczeństwo jazdy, komfort podróży, zużycie opon oraz zużycie paliwa. Firma Beru stworzyła system kontroli ciśnienia w oponach, który został dość powszechnie zaaprobowany przez producentów samochodów. System oferuje ciągłą kontrolę ciśnienia w oponach zarówno podczas postoju jak i w czasie jazdy. Ponadto system umożliwia:

- wczesne ostrzeganie przed zbytnim spadkiem ciśnienia
- wspomaganie doboru ciśnienia podczas pompowania
- kontrolę parametrów ciśnienia podczas diagnostyki oraz napraw auta



- 1 – czujniki ciśnienia montowane w kołach
- 2 – odbiornik sygnału radiowego
- 3 – jednostka centralna
- 4 – przetworniki pomiarowe

Rys. 10.2 Ilustracja systemu kontroli ciśnienia w oponach

Czujnik ciśnienia wbudowany jest w zawór znajdujący się w kole (1). Zasilany jest bateryjnie a transmisja odbywa się drogą radiową. Dla zapewnienia jednoznaczności pomiarów każde koło, w tym również koło zapasowe posiada swój własny identyfikator. Monitorowane jest aktualne ciśnienie w oponie oraz jego spadki. Możliwe jest zaprogramowanie ciśnienia nominalnego dla danego auta (może tego dokonywać producent auta lub jego użytkownik). W skład zestawu wchodzi 4 zawory, 4 układy elektroniczne montowane w kole, 4 przetworniki pomiarowe,

odbiornik sygnału radiowego oraz jednostka centralna. Jednostka centralna jest jednocześnie węzłem sieci CAN, co umożliwia jej bezpośrednią współpracę z samochodową siecią CAN.

10.6. Sieć CAN a model odniesienia ISO/OSI

Podobnie jak większość współczesnych systemów komunikacyjnych, w tym również sieć CAN, została zaprojektowana zgodnie z wzorcowym modelem sieci otwartych zdefiniowanym przez ISO⁴. Model ISO/OSI opisuje sposób wymiany informacji w sieci. Definiuje 7 warstw protokołu oraz ich wzajemne relacje (logiczne, elektryczne, fizyczne). Co do zasady, kolejne warstwy modelu mogą komunikować się wzajemnie wyłącznie w układzie pionowym. Oznacza to, że warstwy świadczą usługi komunikacyjne wyłącznie w stosunku do swoich najbliższych warstw sąsiednich.

Numer warstwy	Nazwa warstwy	Realizacja w sieci CAN
7	Warstwa aplikacji	CANopen, DeviceNet, J1939
6	Warstwa prezentacji	Nie zdefiniowane
5	Warstwa sesji	
4	Warstwa transportowa	
3	Warstwa sieciowa	
2	Warstwa danych	Zgodna z ISO 11898-1
1	Warstwa fizyczna	Zgodna z: ISO 11898-2, ISO 11898-3, SAE J2411, ISO 11992

Rys. 10.3 Specyfikacja CAN w stosunku do modelu odniesienia ISO/OSI

W sieci CAN, podobnie jak dla większości sieci typu LAN, definiowane są tylko 3 warstwy modelu (rys 10.3). Należą do nich: warstwa fizyczna (1), warstwa danych zwana również warstwą łącza danych (2) oraz warstwa aplikacji (7). Pozostałe warstwy z modelu nie zostały zdefiniowane. Warstwy 1 i 2 zostały ujęte w odpowiednich międzynarodowych uregulowaniach standaryzacyjnych.

Warstwa fizyczna

Warstwa ta definiuje mechaniczne i elektryczne aspekty fizycznego dostępu do realnej sieci komunikacyjnej. Fizyczny kanał komunikacyjny zwany jest również kanałem podkładowym lub medium komunikacyjnym. W warstwie tej zdefiniowane są sposoby kodowania i dekodowania informacji oraz zastosowana technika synchronizacji sieci. Przedmiotem opisu warstwy fizycznej jest również specyfikacja standardowych rodzajów złącz.

W przypadku sieci CAN informacja kodowana jest w postaci napięciowego sygnału różnicowego pomiędzy dwoma przewodami elektrycznymi sieci. Medium fizycznym w

⁴ Model sieci otwartej ISO/OSI (ang. *Open System Interconetion*) jest zgodny z normą międzynarodową ISO7498 i często jest nazywany modelem ISO/OSI.

tym przypadku jest para skręconych przewodów o odpowiedniej impedancji charakterystycznej zakończonych dwoma rezystorami zwierającymi tzw. terminatorami. Rezystancja terminatorów jest dobrana, aby możliwe było optymalne tłumienie odbić w magistrali.

Warstwa łączy danych

Warstwa ta definiuje strukturę logiczną ramki komunikacyjnej. Warstwa ta jest odpowiedzialna za odbiór i wysyłanie informacji z i do warstwy fizycznej oraz za komunikację z warstwą aplikacji. W przypadku sieci CAN zadania warstwy łączy danych są realizowane sprzętowo. W celu sprzętowej realizacji zadań warstwy łączy danych zaprojektowano odpowiednie układy scalone zwane kontrolerami CAN. Kontrolery CAN są również integrowane z systemami mikroprocesorowymi w postaci specjalizowanych modułów. Kontroler CAN realizuje dostęp do medium komunikacyjnego za pośrednictwem sprzętowego sterownika spełniającego wymagania stawiane w warstwie fizycznej. Dzięki temu zapewniono prostą, taną i wygodną implementację sieci. Podobne podejście zastosowano między innymi w sieci AS-i. W wielu innych sieciach zadania warstwy drugiej modelu ISO/OSI realizowane są programowo (np. w sieci MODBUS RTU czy w sieci PROFIBUS PA).

Warstwa aplikacji

Specyfikacja CAN zawarta w odpowiednich normach opisujących warstwy pierwszą i drugą nie definiuje szeregu aspektów istotnych z punktu widzenia praktycznej realizacji sieci. Należą do nich między innymi takie aspekty jak: ustalenie schematu adresowania węzłów, kontrola przepływu informacji, ustalanie prędkości transmisji w sieci i procedur startowych, definicji sposobów przesyłania danych o objętości przekraczających objętość jednej ramki komunikacyjnej itp. Zagadnienia te są realizowane i rozstrzygane przez protokoły warstwy 7 modelu OSI zwane też protokołami warstwy wyższej HLP (ang. *High Level Protocol*).

Do najbardziej popularnych protokołów warstwy wyższej HLP należą:

- CANopen – protokół utworzony, spopularyzowany i rozwijany przez organizację CiA. Zyskał największą popularność w automatyce przemysłowej i w przemyśle budowy maszyn.
- DeviceNet – opracowany przez firmę Allen-Bradley(Rockwell). Obecnie jest rozwijany jest przez organizację ODVA (*Open DeviceNet Vendors Association*). Wykorzystywany jest w automatyce przemysłowej, i elektroenergetyce,
- CAN Kingdom – opracowany przez Szwedzką firmę Kvaser AB. Wykorzystywany jest głównie w zadaniach sterowania maszyn,
- J1939 – opracowany i rozwijany przez stowarzyszenie SAE (*Society of Automotive Engineers*). Stosowany jest w automatyzacji aut osobowych i ciężarowych oraz maszyn budowlanych.

10.7. Podstawowe wiadomości o sieci CAN

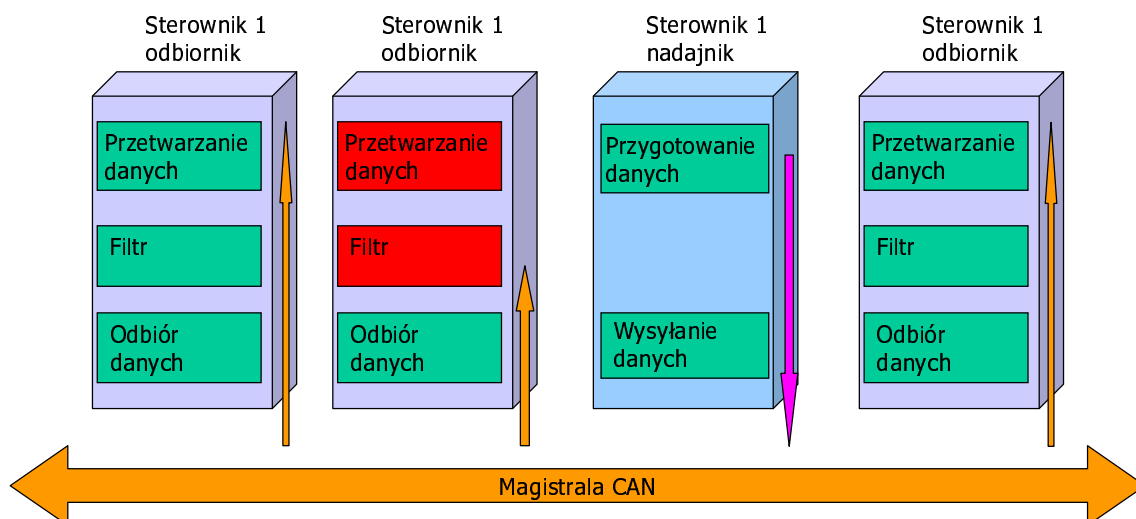
Komunikacja w sieci oparta jest na modelu rozgłoszeniowym (ang. *broadcast*). W modelu tym istotna jest wiadomość i jej treść, a nie źródło jej pochodzenia. Każdej wiadomości przypisywany jest identyfikator. Identyfikator musi być bezwzględnie unikalny w skali całej sieci. Identyfikatory pozwalają jednocześnie na:

- ❑ zachowanie porządku w sieci,
- ❑ wskazanie stopnia ważności wiadomości (prioritety wiadomości),
- ❑ realizację procedury arbitrażu.

Jest jasne, że jeśli identyfikatory wiadomości będą miały charakter unikalny w skali globalnej to zapewnić to może bardzo prostą obsługę i rozbudowę istniejącej sieci. Dzięki temu, w prosty sposób można dodawać nowe węzły do magistrali bez konieczności głęboko idących ingerencji w jej ustawienia. Także wymiana uszkodzonego węzła sieci jednostki nie powoduje konieczności jakichkolwiek zmian w oprogramowaniu czy też ustawieniach sprzętowych.

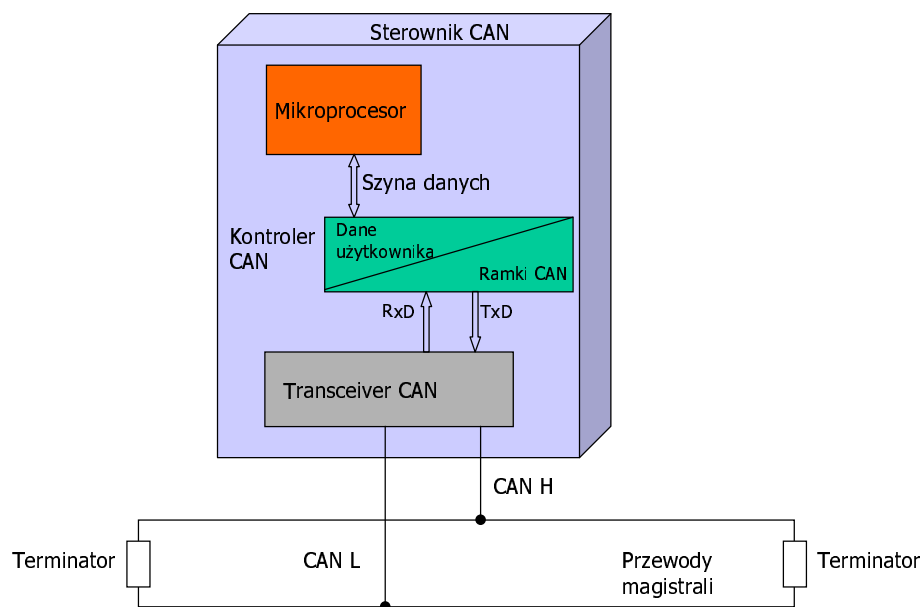
10.7.1 Przepływ informacji w sieci

Każda wiadomość rozsyłana w sieci jest dostępna dla każdego węzła w sieci, w tym oczywiście także dla węzła, który jest jej źródłem. Każdy węzeł, przy pomocy odpowiedniego filtra (maski), może decydować o przyjęciu wiadomości i przekazaniu jej do dalszego przetwarzania lub o jej odrzuceniu. Akceptacja wiadomości realizowana jest na podstawie analizy zawartości identyfikatora wiadomości. Urządzenia wysyłające wiadomości akceptują je jednocześnie niezależnie od identyfikatora. Schemat przepływu informacji w sieci przedstawiono na rys. 10.4.



Rys. 10.4 Przepływ informacji w obrębie magistrali

10.7.2 . Przepływ informacji w obrębie węzła



Rys. 10.5 Przepływ informacji w obrębie węzła

Przepływ informacji w obrębie węzła sieci CAN przedstawiono na rys. 10.5. Kontroler CAN oraz sterownik magistrali (transceiver) realizują zadania warstwy 1 i warstwy drugiej modelu ISO/OSI. Wyróżniona jednostka mikroprocesorowa realizuje programowo zadania warstwy aplikacyjnej. Przepływ informacji w węźle jest dwukierunkowy realizowany równolegle na poziomie wyższym i szeregowo na poziomie niższym. Należy jednak zaznaczyć, że przepływ informacji w kanale podkładowym (w magistrali CAN) odbywa się dwukierunkowo ale w trybie komunikacji szeregowej naprzemiennej.

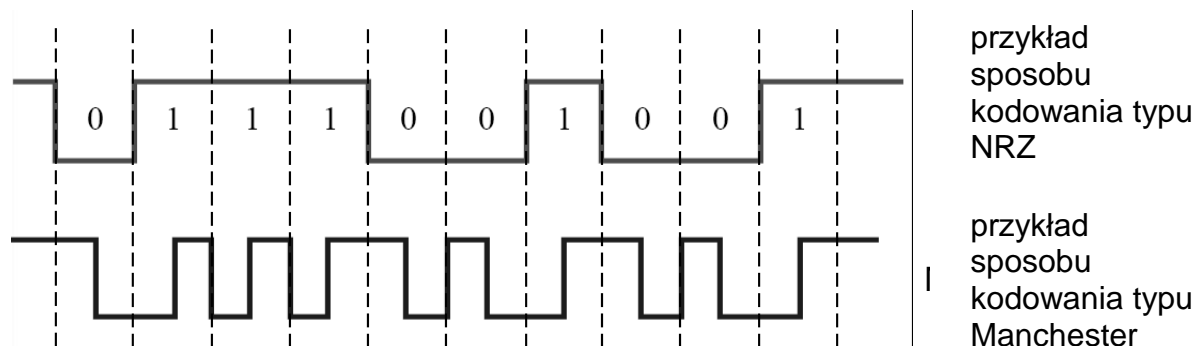
10.8 Warstwa fizyczna

Warstwa fizyczna opisuje między innymi rodzaj medium transmisyjnego, rodzaje okablowania, poziomy napięć, sposoby przyporządkowanie stanów logicznych, sposób synchronizacji, prędkość transmisji, itp.

10.8.1. Technika kodowania informacji

Magistrala CAN jest magistralą dwukierunkową i synchroniczną. Zastosowano w niej sposób kodowania informacji metodą NRZ (ang. **Non Return to Zero**). W metodzie tej stan logiczny jest reprezentowany przez stały poziom sygnału fizycznego. Ten sposób kodowania jest bardziej efektywny niż metodą Manchester (rys. 10.6). W metodzie kodowania typu Manchester stany logiczne kodowane są nie tyle poziomami sygnałów fizycznych, co przez zmiany (zbocza) tych sygnałów. Istotną wadą techniki kodowania NRZ jest możliwość utraty synchronizmu sieci, w przypadku przesyłania długich ciągów bitowych o tej samej wartości. W CAN problem ten rozwiązano w ten sposób, że w przypadku występowania ciągów bitowych składających się z samych zer lub samych jedynek i jednocześnie dłuższych niż 5 bitów następuje automatyczna krótkotrwała zmiana stanu na przeciwny (technika: *bit*

stuffing). Zmiana ta pozwala na zachowanie synchronizacji węzłów. Dodatkowe bity synchronizujące są usuwane przez kontroler CAN odbiornika wiadomości. Z punktu widzenia warstwy aplikacyjnej technika dodawania i usuwania bitów synchronizacyjnych jest całkowicie przezroczysta.



Rys. 10.6 Porównanie sposobów kodowania typu NRZ i typu Manchester

10.8.2. Taktowanie i synchronizacja

Wykorzystanie transmisji synchronicznej pozwala na znacznie zwiększenie przepływności binarnej w kanale podkładowym w stosunku do transmisji asynchronicznej. Wymaga to jednak zastosowania określonych metod zapewniających synchronizację procesów komunikacyjnych. W przypadku transmisji asynchronicznej, synchronizacja jest realizowana przez bit startu dołączany do każdego transmitowanego znaku. W przypadku komunikacji synchronicznej w sieci CAN stosowany jest tylko jeden bit startu na początku całej ramki komunikacyjnej. Zatem zapewnienie właściwego odbioru transmitowanej wiadomości wymaga właściwej synchronizacji nadajnika wiadomości z jej odbiornikami w czasie trwania transmisji całej wiadomości.

10.8.3. Zależność prędkości transmisji od długości magistrali

Prędkość transmisji w sieci CAN zależy od właściwości elektrycznych samego kanału podkładowego i wejściowych parametrów elektrycznych węzłów sieciowych. Istotnym parametrem sieci umożliwiającym określenie maksymalnej prędkości transmisji jest czas propagacji sygnału w sieci pomiędzy dwoma najbardziej od siebie oddalonymi węzłami sieci. Czas propagacji uwzględnia:

- ❑ czas potrzebny na przesłanie sygnału pomiędzy najbardziej oddalonymi węzłami sieci,
- ❑ czas opóźnienia w układach nadajnika i odbiornika,
- ❑ czas potrzebny na uzyskanie synchronizmu,
- ❑ czas niezbędny na powrót sygnału.

Wynika stąd, że im dłuższa magistrala, tym większa jej pojemność i indukcyjność oraz większe opóźnienie i tłumienie sygnału. W związku z tym wzrasta czas propagacji i zmniejsza się dopuszczalna prędkość komunikacji w sieci. Należy zaznaczyć, że dla prawidłowego działania sieci konieczne jest przyjęcie jednej wspólnej prędkości komunikacji dla wszystkich węzłów sieci. W tym przypadku prędkość sieci ustalana jest w oparciu o zasadę analizy najgorszego przypadku.

10.8.4. Kanał podkładowy transmisji danych

Aby medium fizyczne mogło być użyte do transmisji wiadomości zgodnych ze specyfikacją protokołu CAN musi posiadać zdolność reprezentowania stanu dominującego i recesywnego. Jak dotąd udało się to uzyskać za pomocą medium elektrycznego i optycznego. Stanem dominującym jest taki stan logiczny magistrali, który ma w stosunku do stanu recesywnego tę przewagę, że ich jednoczesna zaistnienie w tym samym medium fizycznym spowoduje powstanie stanu dominującego. W przypadku medium elektrycznego jest to stan 0 logicznego.

Dla medium elektrycznego poziomy napięcia różnicowego oraz stany dominujące i recesywne zostały zdefiniowane w normach: ISO 11898-2, ISO 11898-3, SAE J2411 i ISO 11992. Dla medium optycznego poziom recesywny jest reprezentowany przez brak sygnału świetlnego „dark” zaś poziom dominujący reprezentuje sygnał świetlny „light”.

Najczęściej używanym kanałem fizycznym transmisji w sieciach CAN jest symetryczna skręcona, ekranowana para przewodów. W pojazdach samochodowych stosowany jest pojedynczy przewód. Prowadzone są też badania nad możliwością komunikacji CAN z wykorzystaniem linii zasilających urządzenia (na wzór sieci AS-i).

Dobór medium fizycznego wymaga uwzględnienia takich parametrów jak: czas propagacji sygnału, rezystancja, przekrój poprzeczny przewodów czy spadek napięcia na przewodach. W procesie doboru medium obliczana jest średnica przewodu elektrycznego sieci tak aby nie został przekroczony maksymalny dopuszczalnego spadku napięcia pomiędzy najbardziej oddalonymi węzłami. Obliczenia te uwzględniają nie tylko rezystancje przewodów sieci, ale również rezystancje wszystkich podłączonych odbiorników. Maksymalny spadek napięcia w sieci musi być na tyle mały by zapewnić prawidłowy poziom sygnału w każdym węźle.

10.8.5. Topologia sieci

W celu eliminowania skutków odbicia sygnałów elektrycznych od końca linii, stosowane są tzw. terminatory. Terminatory zwierają obie linie sieci na jej obu końcach. Zasadniczą topologią sieci CAN jest topologia magistrali. Magistrala posiada linię główną i relatywnie krótkie odgałęzienia. Najkorzystniejszą relację pomiędzy prędkością komunikacji, a długością magistrali można uzyskać w topologii sieci jak najbardziej zbliżonej do prostoliniowej zakończonej terminatorami. Taka topologia nazywana jest również topologią łańcuchową.

Podobnie jak w innych sieciach także i w sieciach CAN stosowane są urządzenia pozwalające na wydłużenie zasięgu sieci, oraz umożliwiają integrację sieci CAN z innymi sieciami. Do urządzeń tych należą: wzmacniacze (ang. *repeaters*), mostki (ang. *bridges*) oraz bramy (ang. *gateways*).

Wzmacniacz (wzmacniak)

Zadaniem wzmacniacza jest regeneracja sygnału elektrycznego tłumionego i zniekształconego w medium komunikacyjnym. Wzmacniacz umożliwia zwiększenie zasięgu tej samej sieci. Z punktu widzenia przepływu informacji jest urządzeniem

całkowicie pasywnym. Należy pamiętać, że wzmacniacz jest dodatkowym źródłem opóźnienia w propagacji sygnału. Wzmacniacz łączy sieci na poziomie warstwy 1 modelu ISO/OSI.

Mostek

Mostek łączy (zespala) odseparowane informacyjnie sieci na poziomie warstwy drugiej modelu ISO/OSI. Istotne w tym przypadku jest, aby identyfikatory wiadomości w obu sieciach były globalnie unikalne. Mostek jest elementem aktywnym sieci i może sterować przepływem informacji pomiędzy zespolonymi sieciami.

Brama

Brama łączy sieci o różnych rodzajach protokołów komunikacyjnych na poziomie warstwy 7 modelu ISO/OSI (warstwa aplikacji). Brama jest, podobnie jak mostek, elementem aktywnym, którego zadaniem jest translacja protokołów komunikacyjnych.

10.8.6. Dostęp do magistrali

Bezpośredni dostęp do medium fizycznego (magistrali dwuprzewodowej) realizowany jest przy pomocy specjalnych układów zwanych sterownikami magistrali lub transceiverami⁵. Układ sterownika składa się z nadajnika oraz odbiornika. Sterownik w czasie nadawania wiadomości spełnia następujące zadania:

- ❑ dokonuje translacji poziomów napięć do poziomów zdefiniowanych dla stanów: recesywnego i dominującego
- ❑ zapewnia odpowiednią pojemność wyjściową układu,
- ❑ chroni układ nadajnika przed przeciążeniem prądowym i napięciowym,
- ❑ ogranicza poziom emisji elektromagnetycznej.

W czasie odbioru sterownik:

- ❑ zapewnia odpowiedni poziom sygnału w stanie recesywnym
- ❑ chroni układ komparatora wejściowego przed przepięciami na magistrali

Dodatkowo odbiornik wykrywa usterki na magistrali takie jak przerwanie linii, zwarcie przewodów, zwarcie do masy itd. Sterownik pozwala również na izolację galwaniczną węzła sieci CAN od magistrali.

10.8.7. Unormowania w zakresie warstwy fizycznej

ISO 11898-2 (Sieć CAN o dużej prędkości)

Jest to najczęściej spotykany standard specyfikujący wymagania w stosunku do warstwy fizycznej. Zgodnie z tym standardem maksymalna możliwa do uzyskania prędkość transmisji wynosi 1Mbit/s przy długości magistrali nie dłuższej niż 40m. Do transmisji wykorzystywana jest magistrala dwuprzewodowa. Jeden przewód oznaczany jest symbolem CAN_H, drugi jest oznaczany symbolem CAN_L. Liczba węzłów jest ograniczona dopuszczalnym obciążeniem magistrali. Impedancja falowa przewodów wynosi 120 Ω. Zakres dopuszczalnych napięć względem wspólnego

⁵ transceiver jest neologizmem powstałym ze złożenia słów angielskich transmitter i receiver oznaczających odpowiednio nadajnik i odbiornik.

potencjału odniesienia zawiera się w przedziale od $-2V$ do $+7V$. Nominalny czas propagacji sygnału w sieci nie powinien przekraczać $5ns/m$.

ISO 11898-3 (sieć CAN tolerująca uszkodzenia)

Ten standard jest najbardziej popularny w zastosowaniach w technice samochodowej. Ponieważ z założenia jest to sieć o krótkim zasięgu, to problem odbić sygnałów od końców przewodów nie jest istotny. Możliwa jest więc praca tej sieci zgodnej z tą specyfikacją bez konieczności stosowania terminatorów.

Pozwala to na użycie sterowników magistrali niskiej mocy, a co za tym idzie obniżenie poboru mocy. Specyfikacja ta dopuszcza stosowanie dowolnej topologii sieci. Sieć ma zdolności komunikacyjne nawet wówczas, gdy uszkodzeniu ulegnie jedna z linii magistrali. Norma ISO 11898-3 ogranicza maksymalną prędkość transmisji do $125kb/s$. Określa również zależność dopuszczalnej długości magistrali jest zależna prędkości transmisji i obciążenia sieci. Zakres dopuszczalnych napięć względem wspólnego potencjału odniesienia zawiera się w przedziale od $-2V$ do $+7V$. Dopuszczalne jest stosowanie do 32 węzłów w jednej sieci.

SAE J2411 (jednoprzewodowa sieć CAN)

Standard SAE J2411 definiuje sieci o niskich wymaganiach dotyczących zarówno prędkości transmisji jak i maksymalnej długości magistrali. Komunikacja odbywa się tylko przy użyciu jednego przewodu z prędkością $33,3kb/s$ (dla celów diagnostycznych dostępna jest też prędkość $83,3kb/s$). Maksymalna liczba urządzeń sieciowych wynosi 32. Standard ten stosowany jest zwłaszcza do sterowania układami elektronicznymi służącymi do podniesienia komfortu jazdy w samochodach. Medium transmisyjnym jest jeden nie ekranowany przewód. Topologia sieci jest dowolna. Standard dopuszcza komunikację pomiędzy aktywnymi węzłami w sytuacji, gdy pozostałe pozostają w stanie uśpienia.

ISO 11992 (sieć CAN typu punkt-punkt)

Standard ten definiuje połączenie typu punkt-punkt np. pomiędzy ciągnikiem a przyczepą. W przypadku gdy jeden ciągnik posiada dwie przyczepy, to stosowane jest połączenie łańcuchowe (ang. *daisy-chain*). Nominalna prędkość transmisji w tej sieci jest ograniczona do $125 kbit/s$ przy maksymalnej dopuszczalnej długości magistrali o topologii łańcuchowej równej $40 m$. Standard opisuje sposoby zarządzania błędami oraz definiuje napięcie zasilania ($12VDC$ lub $24VDC$). Jako medium transmisyjne stosowana jest para nie ekranowanych przewodów.

Pozostałe standardy

Magistrala CAN wykorzystująca światłowody jako medium transmisyjne nie została jak dotąd objęta żadnym standardem. Głównym powodem są występujące problemy z silnym tłumieniem sygnału i stratami sygnału na złączach. Zalety sieci wykorzystujących światłowody polegają między innymi na tym, że są one odporne na zewnętrzne zakłócenia elektromagnetyczne oraz same ich nie emitują. Pozwala to na zastosowanie sieci światłowodowych w środowiskach o wysokim poziomie zakłóceń elektromagnetycznych. Całkowite odseparowanie galwaniczne pozwala również na stosowanie tego rodzaju sieci w środowiskach zagrożonych wybuchem.

10.9. Warstwa łącza danych

10.9.1 Wprowadzenie

W celu osiągnięcia przejrzystości i elastyczności w projektowaniu sieci, w warstwie łącza danych w specyfikacji CAN zostały wyróżnione dwie podwarstwy zgodne z modelem ISO/OSI.

- ❑ Podwarstwa logicznego sterowania komunikacją LLC (*Logical Link Control*)
- ❑ Podwarstwa sterowania dostępem do warstwy fizycznej MAC (*Medium Access Control*)

Warstwa łącza danych
LLC Filtrowanie wiadomości Informacja o przeciążeniach Zarządzanie odtwarzaniem inf.
MAC Przygotowanie danych Kodowanie ramki (Stuffing / Destuffing) Wykrywanie błędów Sygnalizacja błędów Sygnalizacja potwierdzenia odbioru
Warstwa fizyczna
Kodowanie / Dekodowanie bitów Taktowanie Synchronizacja
Charakterystyki nadajnika / odbiornika

Zadania podwarstwy LLC są następujące:

- ❑ wspomaganie transmisji danych,
- ❑ filtrowanie wiadomości,
- ❑ zarządzanie powrotem do pracy w sieci (RECOVERY MANAGEMENT) oraz
- ❑ informowanie w przypadku wystąpienia przeciążenia.

Podwarstwa MAC odpowiada głównie za realizację procesu transmisji danych, a w szczególności za:

- ❑ podział danych na ramki,
- ❑ prowadzenie arbitrażu,
- ❑ kontrolę błędów,
- ❑ sygnalizację błędów,
- ❑ ograniczanie wpływu uszkodzeń na pracę sieci.

Na poziomie tej podwarstwy podejmowane są decyzje o możliwości podjęcia nowej transmisji lub odbioru. Również tu zdefiniowane są pewne wymagania dotyczące warunków taktowania procesu komunikacyjnego.

Rys. 10.7 Podstawowe funkcje realizowane przez podwarstwy LLC i MAC

10.9.2 Podstawowe definicje

10.9.2.1. Wiadomość

Wiadomość jest podstawowym obiektem wymiany informacji. Wiadomość jest generowana przez węzły sieci zgodnie ze specyfikacją warstwy łącza danych. Tworzona jest zgodnie z określonymi zasadami i ustalonym formatem. Wiadomość jest ograniczona co do swojej długości. Wiadomości mogą być transmitowane wówczas, gdy magistrala komunikacyjna nie jest zajęta.

10.9.2.2. Zarządzanie przepływem informacji

Węzły w sieci CAN nie posiadają i nie przechowują żadnych informacji dotyczących konfiguracji sieci. Jest kilka istotnych konsekwencji takiego stanu rzeczy.

Elastyczność

Sieć CAN może być rozbudowywana o nowe węzły bez konieczności dokonywania jakichkolwiek zmian w oprogramowaniu czy też konfiguracji sprzętowej żadnego z istniejących już węzłów. Nie są konieczne również zmiany w warstwie aplikacyjnej.

Selektywność

Każda wiadomość posiada IDENTYFIKATOR. Identyfikator odwołuje się do stopnia istotności wiadomości. Identyfikator nie zawiera informacji dotyczących miejsca (adresu) przeznaczenia wiadomości. Dzięki temu każdy węzeł sieci może sam decydować (przy użyciu funkcji filtrowania wiadomości) czy transmitowana wiadomość jest mu przydatna, czy też nie.

Tryb pracy równoległej węzłów (multicast)

Dzięki zastosowaniu mechanizmu filtrowania wiadomości, dowolna liczba węzłów może jednocześnie odbierać, przetwarzać i wykonywać polecenia zawarte w tej samej wiadomości.

Gwarancja zgodności danych

Specyfikacja sieci CAN gwarantuje, że wewnątrz sieci wiadomość będzie zaakceptowana przez wszystkie węzły albo nie zaakceptuje jej żaden z nich. Taką właściwość udało się uzyskać dzięki wykorzystaniu trybu MULTICAST oraz odpowiedniemu zarządzaniu błędami.

10.9.2.3 Prędkość transmisji

Prędkość transmisji w sieciach CAN może być różna. Prędkość ta musi być jednak identyczna dla wszystkich węzłów w obrębie tej samej sieci.

10.9.2.4 Priorytet wiadomości

IDENTYFIKATOR określa priorytet wiadomości. Priorytet wiadomości jest istotny w początkowej fazie wysyłania wiadomości, ponieważ w tej fazie rozgrywany jest arbitraż. W przypadku zaistnienia konfliktu w sieci będzie rozpowszechniona wiadomość o najwyższym priorytecie.

10.9.2.5 Żądanie transmisji danych

W przypadku, gdy określony węzeł sieci potrzebuje wiadomości (danych) z innego węzła sieci, to może wysłać odpowiednio sformatowaną wiadomość zwaną żądaniem transmisji lub ramką żądania transmisji (REMOTE FRAME). W odpowiedzi generowana jest wiadomość z żadaną informacją w postaci ramki danych (DATA FRAME). Identyfikatory obu tych wiadomości są identyczne.

10.9.2.6 Tryb MULTIMASTER

Każdy węzeł ma prawo rozpocząć transmisję wiadomości, o ile medium transmisyjne jest niezajęte. Dostęp do magistrali uzyskuje węzeł generujący wiadomość o najwyższym priorytecie.

10.9.2.7 Arbitraż

Jeżeli dwa lub więcej węzłów sieci rozpocznie transmisję wiadomości w tym samym momencie, to wystąpi konflikt dostępu do medium. Konflikt ten jest rozstrzygany w procesie nieniszczącego arbitrażu. W przypadku jednoczesnego rozpoczęcia transmisji ramek wiadomości o tym samym identyfikatorze (ramki danych i żądania transmisji) arbitraż wygrywa ramka danych. Podczas arbitrażu, każdy nadajnik każdego węzła dla każdego wysyłanego bitu porównuje wartość stanu logicznego wysyłanej wiadomości ze stanem logicznym odbieranym z sieci. Jeżeli obie wartości są identyczne, to węzeł może kontynuować nadawanie. W przypadku przeciwnym węzeł automatycznie zawiesza tryb nadawania i przechodzi w tryb odbioru informacji. Arbitraż rozgrywany jest wyłącznie w czasie trwania transmisji identyfikatorów wiadomości.

10.9.2.8 Bezpieczeństwo przesyłanych informacji

Każdy węzeł posiada zaimplementowane procedury służące do detekcji, sygnalizacji i kontroli błędów transmisji.

10.9.2.9 Kontrola błędów transmisji

Kontrola błędów transmisji jest wieloetapowa. Służą jej:

- porównanie stanu logicznego wysyłanego bitu ze stanem logicznym w medium,
- generowanie cyklicznej redundancyjnej sumy kontrolnej CRC (**Cyclic Redundancy Check**),
- technika kodowania bitów synchronizacyjnych (*bit stuffing*),
- kontrola integralności ramki.

10.9.2.10 Skuteczność mechanizmu detekcji błędów

Mechanizm detekcji błędów ma następujące właściwości:

- wykrywa wszystkie błędy globalne,
- wykrywa wszystkie błędy lokalne nadajnika,
- wykrywanych jest do 5 przypadkowo zaistniałych błędów w wiadomości,
- wykrywane są błędy ciągu danych o długości mniejszej niż 15 bitów,
- wykrywana jest każda nieparzysta liczba błędnych bitów.

Prawdopodobieństwo p nie wykrycia uszkodzonej ramki jest mniejsze niż

$$p = x \cdot 4,7 \cdot 10^{-11}$$

gdzie: x jest stopą błędu wiadomości.

10.9.2.11 Sygnalizacja błędów i czas powrotu do normalnej pracy

Zadaniem węzła, który wykryje uszkodzoną wiadomość jest natychmiastowe zasygnalizowanie takiego zdarzenia. Uszkodzona wiadomość jest anulowana

i automatycznie transmitowana ponownie. Maksymalny interwał czasowy upływający pomiędzy wykryciem błędu i ponownym rozpoczęciem transmisji jest równy czasowi transmisji 31 bitów (o ile nie pojawią się kolejne błędy).

10.9.2.12 Automatyczne wykluczanie uszkodzeń trwałych (Fault Confinement)

Węzły sieci CAN mają możliwość odróżnienia krótkich zakłóceń od uszkodzeń trwałych. Uszkodzone węzły mogą być wyłączane.

10.9.2.13 Połączenia

Szeregowa sieć CAN pozwala na dołączenie teoretycznie nieograniczonej liczby urządzeń. Praktycznie, liczba dołączonych urządzeń jest limitowana przez dopuszczalne czasy opóźnienia sygnału w sieci oraz dopuszczalne obciążenie prądowe trancieverów.

10.9.2.14 Wartości/Poziomy sygnału magistrali

Sygnał CAN może przyjmować tylko jeden z dwóch komplementarnych stanów logicznych: dominujący lub recesywny. W przypadku jednoczesnej transmisji bitu recesywnego i dominującego stan logiczny magistrali przyjmuje wartość bitu dominującego. Parametry techniczne (np. poziomy napięcie) reprezentujące stany logiczne nie są przedmiotem ogólnej specyfikacji i są specyficzne dla wybranej warstwy fizycznej.

10.9.2.15 Sygnalizacja potwierdzenia odbioru ACK (Acknowledgement)

Każdy odbiornik wiadomości sprawdza jej zawartość i potwierdza jej zgodność lub sygnalizuje błąd.

10.9.2.16 Tryb uśpienia węzła (Sleep Mode / Wake-up)

W celu zmniejszenia zużycia energii, węzły sieci CAN mogą przechodzić w stan uśpienia. W tym stanie zostaje zawieszona zarówno komunikacja z medium transmisyjnym jak i wewnętrzna aktywność węzła. Węzeł może być wyprowadzony ze stanu uśpienia w wyniku wystąpienia określonej aktywności w sieci lub też w wyniku działania układu sterowania węzła. Bezpośrednio po wyjściu ze stanu uśpienia, warstwa MAC rozpoczyna proces przygotowania węzła do pracy w sieci. W czasie przygotowania warstwa MAC oczekuje najpierw na ustabilizowanie się generatora podstawy czasu węzła, a następnie dokonuje synchronizacji węzła z magistralą sprawdzając 11 kolejnych bitów recesywnych. Dopiero po realizacji tych operacji następuje logiczne dołączenie węzła i definitywne zakończenie trybu uśpienia węzła.

10.9.3 Transmisja wiadomości

10.9.3.1 Format ramki

Rozróżniane są dwa formaty ramki różniące się między sobą polem IDENTYFIKATORA (*IDENTIFIER*).

- ❑ Ramka z identyfikatorem 11 bitowym – oznacza ramkę standardową
- ❑ Ramka z identyfikatorem 29 bitowym - oznacza ramkę rozszerzoną

10.9.3.2 Rodzaje ramek

Rozróżniane są cztery rodzaje ramek transmitujących wiadomości:

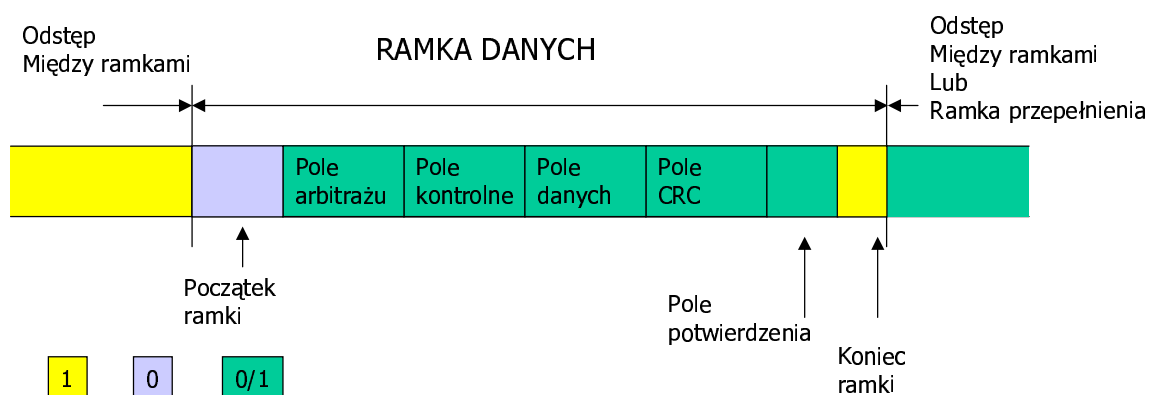
- ❑ Ramka danych (*DATA FRAME*) przeznaczona do transmisji wiadomości przez węzeł nadawczy do węzłów odbiorczych,
- ❑ Ramka żądania transmisji (*REMOTE FRAME*) przeznaczona do zainicjowania transmisji ramki danych z węzła o wskazanym przez nią identyfikatorze,
- ❑ Ramka sygnalizacji błędu (*ERROR FRAME*) przeznaczona do sygnalizacji błędu. Może być transmitowana przez każdy węzeł sieci, który dokonał detekcji błędu transmisji,
- ❑ Ramka przepełnienia (*OVERLOAD FRAME*) przeznaczona do generacji w sieci dodatkowego opóźnienia (zwłoki czasowej) pomiędzy kolejnymi ramkami danych lub ramkami żądania transmisji.

Ramka danych oraz ramka żądania transmisji mogą występować w obu formatach ramek, a więc zarówno w przypadku formatu standardowego jak i w przypadku formatu rozszerzonego. Pomiedzy ramkami transmisyjnymi wprowadzane są przerwy czasowe (*INTERFRAME SPACE*).

10.9.3.3 Ramka danych (*DATA FRAME*)

Ramka danych składa się z siedmiu pól:

- | | |
|-------------------------|-------------------|
| 1. Początek ramki | START OF FRAME |
| 2. Pole arbitrażu | ARBITRATION FIELD |
| 3. Pole kontrolne | CONTROL FIELD |
| 4. Pole danych | DATA FIELD |
| 5. Pole sumy kontrolnej | CRC FIELD |
| 6. Pole potwierdzenia | ACK FIELD |
| 7. Koniec ramki | END OF FRAME |



Rys. 10.8 Ramka danych

10.9.3.4 Początek ramki SOF (START OF FRAME)

Początek ramki występuje zarówno w przypadku transmisji ramki standardowej i rozszerzonej. SOF oznacza początek ramki danych i ramki żądania transmisji. SOF jest realizowany w postaci pojedynczego stanu dominującego.

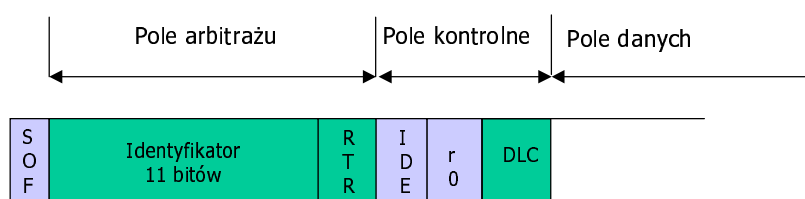
Dowolny węzeł sieci może rozpocząć transmisję tylko wówczas, gdy magistrala znajduje się w stanie czuwania. Wszystkie węzły sieci muszą zsynchronizować się z narastającym zboczem sygnału SOF tego węzła, który jako pierwszy rozpoczyna proces nadawania ramki danych lub ramki żądania transmisji.

10.9.3.5 Pole arbitrażu (ARBITRATION FIELD)

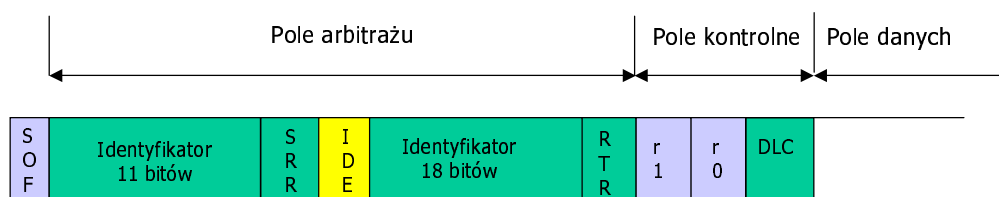
Format pola arbitrażu jest zależny od formatu ramki (standardowa czy rozszerzona).

- W transmisji ramki w formacie standardowym pole to składa się z 11 bitowego *identyfikatora* oraz bitu zwanego *RTR-BIT*. Bity identyfikatora oznaczane są kolejno *ID-28...ID-18*,
- W transmisji ramki w formacie rozszerzonym pole to składa się z 29 bitowego *identyfikatora* oraz bitów zwanych *SRR-BIT*, *IDE-BIT* i *RTR-BIT*. Kolejne bity identyfikatora oznaczane są *ID-28...ID-0*.

Standardowy format ramki



Rozszerzony format ramki



Rys. 10.9 Pole arbitrażu w ramce w formacie standardowym i rozszerzonym

10.9.3.6 Identyfikator

□ RAMKA STANDARDOWA

Długość identyfikatora w ramce standardowej wynosi 11 bitów i odpowiada tak zwanej bazie (ang. *base ID*) w ramce rozszerzonej. Transmisja identyfikatora rozpoczyna się począwszy od bitu ID-28 i kończy się na bicie ID-18. ID-18 jest najmniej znaczącym bitem LSB (ang. **Least Significant Bit**).

Uwaga: Identyfikator nie może przyjmować wartości recesywnych jednocześnie na siedmiu najbardziej znaczących bitach (ID-28 .. ID-22).

□ RAMKA ROZSZERZONA

Ramka rozszerzona składa się z 29 bitów. Podzielona jest na następujące pola

- Identyfikator podstawowy (ang. *base ID*) – 11 bitów
- Identyfikator rozszerzony (ang. *extended ID*) – 18 bitów

Identyfikator podstawowy odpowiada identyfikatorowi ramki w formacie standardowym. W ramce rozszerzonej decyduje o podstawowym priorytecie wiadomości.

Identyfikator rozszerzony zawiera pozostałe 18 bitów, transmitowanych w kolejności od ID-17 do ID-0.

W ramce standardowej po IDENTYFIKATORZE występuje zawsze bit RTR. Bit RTR występuje zarówno w ramce standardowej jak i rozszerzonej.

Bit RTR (ang. **Remote Transmission Request** – zdalne żądanie transmisji)

- W ramce danych bit RTR musi być *dominujący*
- W ramce zdalnego żądania transmisji REMOTE FRAME bit RTR musi być *recesywny*

W przypadku ramki rozszerzonej najpierw transmitowane jest *Base ID*, a po nim występują dwa bity: SRR (ang. **Substitute Remote Request**) oraz IDE (ang. **Identifier Extension Bit**).

Bit SRR (ramka rozszerzona)

Bit SRR jest bitem recesywnym. W ramce rozszerzonej transmitowany jest na pozycji bitu RTR ramki standardowej, dlatego też nazywany jest jego substytutem. Pozwala to na uniknięcie kolizji w przypadku, gdy ramka standardowa i rozszerzona mają identyczną bazę *Base ID*. W tym przypadku w ramka standardowa wygrywa arbitraż (RTR jest bitem dominującym).

Bit IDE

Dodatkowy bit identyfikatora IDE w zależności od rodzaju ramki znajduje się w:

- polu arbitrażu, w przypadku ramki rozszerzonej, lub
- polu kontrolnym w przypadku ramki standardowej.

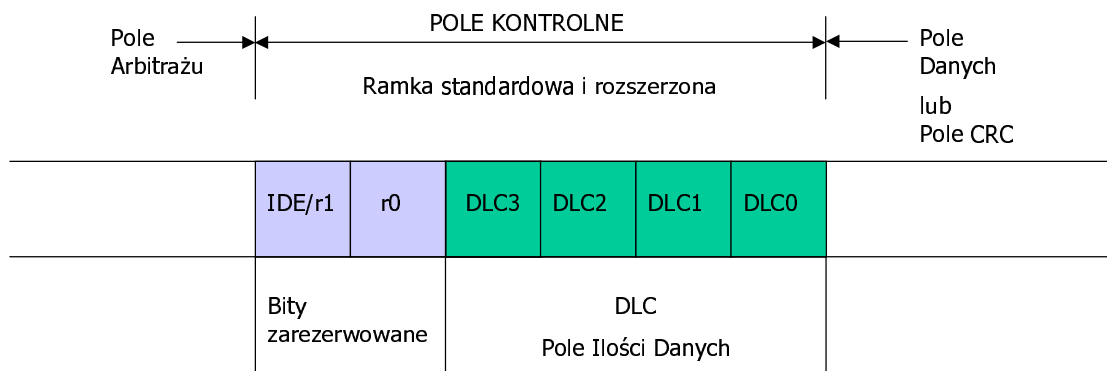
Bit IDE w ramce standardowej jest „dominujący” zaś w rozszerzonej „recesywny”.

10.9.3.7 Pole kontrolne (CONTROL FIELD)

Pole kontrolne występuje zarówno w ramce rozszerzonej jak i w standardowej. Pole to składa się z sześciu bitów (rys. 10.10).

- W ramce standardowej pole to zawiera kod długości pola danych DLC (ang. **Data Length Code**), bit IDE (dominujący) oraz bit rezerwowy r0.
- W ramce rozszerzonej pole to zawiera kod długości pola danych DLC, dwa rezerwowe bity r1 i r0. Podczas nadawania obydwa bity rezerwowe powinny być

dominujące. Jakkolwiek odbiornik wiadomości musi zaakceptować dowolną ich kombinację.



Rys. 10.10 Pole kontrolne

Kod długości pola danych *DLC* jest czterobitowym polem określającym liczbę bajtów występujących w *polu danych* (ang. *DATA FIELD*). Sposób kodowania liczby bajtów w *DLC* podano w tabeli 10.1.

Tabela 10.1 Sposób kodowania liczby bajtów w polu kontrolnym

Liczba bajtów danych	Wartości poszczególnych bitów DLC			
	DLC3	DLC2	DLC1	DLC0:
0	d	d	d	d
1	d	d	d	r
2	d	d	r	d
3	d	d	r	r
4	d	r	d	d
5	d	r	d	r
6	d	r	r	d
7	d	r	r	r
8	r	d	d	d

Oznaczenia:

d - bit dominujący

r – bit recesywny

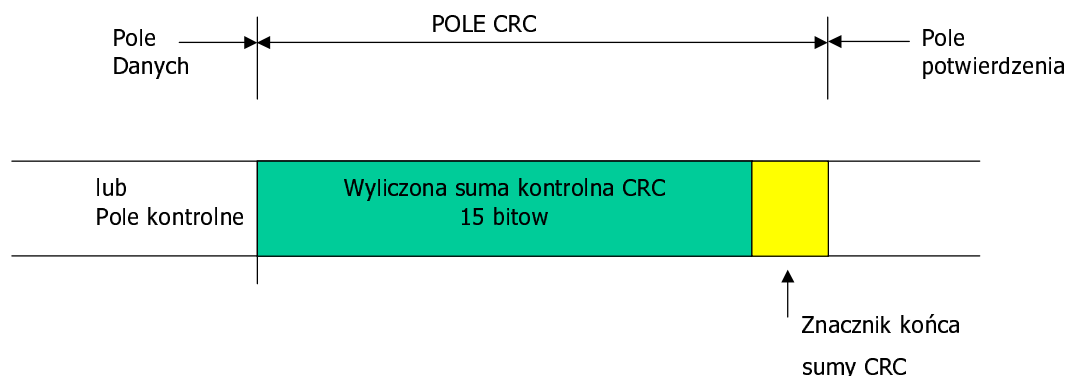
10.9.3.8 Pole danych (*DATA FIELD*)

Pole danych (ang. *DATA FIELD*) występuje zarówno w ramce standardowej jak i w rozszerzonej. POLE DANYCH zawiera informacje, które są transmitowane w RAMCE DANYCH. Może zawierać od 0 do 8 bajtów. Każdy bajt liczy 8 bitów. Najbardziej znaczący bit MSB (ang. ***Most Significant Bit***) w każdym bajcie jest nadawany jako pierwszy.

10.9.3.9 Pole sumy kontrolnej (*CRC FIELD*)

Pole sumy kontrolnej CRC występuje zarówno w ramce standardowej jak i w rozszerzonej. Pole to zawiera wyliczoną sumę kontrolną w części: *CRC SEQUENCE*

oraz bit końca sumy kontrolnej CRC DELIMITER (rys. 10.11) zwany również znacznikiem końca sumy CRC.



Rys. 10.11. Pole sumy kontrolnej CRC

Algorytm wyznaczania wartości sumy kontrolnej jest algorytmem cyklicznej kontroli nadmiarowej CRC przeznaczonym dla ramek liczących nie więcej niż 127 bitów. W celu wyliczenia wartości CRC zdefiniowano najpierw wielomian generacyjny. Następnie tworzony jest wielomian składający się utworzonego z kolejnych bitów nadawanej ramki tzn. składający się z: SOF, ARBITRATION FIELD, CONTROL FIELD, DATA FIELD oraz dodatkowych piętnastu bitów o wartości zerowej. Bity dodatkowe uzupełniające wtrącone w procesie nadawania informacji (*bit stuffing*) nie są brane pod uwagę. Wielomian ten jest dzielony (operacja dzielenia modulo 2) przez wielomian generacyjny:

$$x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + x^1 \quad (2)$$

W wyniku podziału obu wielomianów uzyskiwana jest wartość CRC SEQUENCE. Do implementacji można zastosować algorytm podany poniżej.

Algorytm wyznaczania CRC

Niech $CRC_RG(14:0)$ oznacza rejestr przechowujący wartość sumy kontrolnej CRC, natomiast $NXTBIT$ oznacza kolejny bit transmitowanego ciągu bitów począwszy od SOF i skończywszy na DATA FIELD. Suma CRC obliczana jest następująco:

```

CRC_RG = 0;
REPEAT
    CRCNXT = NXTBIT EXOR CRC(14)
    CRC_RG(14:1) = CRC_RG(13:0);
    CRC_RG(0) = 0;
    IF CRCNXT THEN
        CRC_RG(14:0) = CRC_RG(14:0) EXOR (4599hex);
    ENDIF
UNTIL (zostanie osiągnięte CRC SEQUENCE lub wystąpi jeden z błędów transmisji)

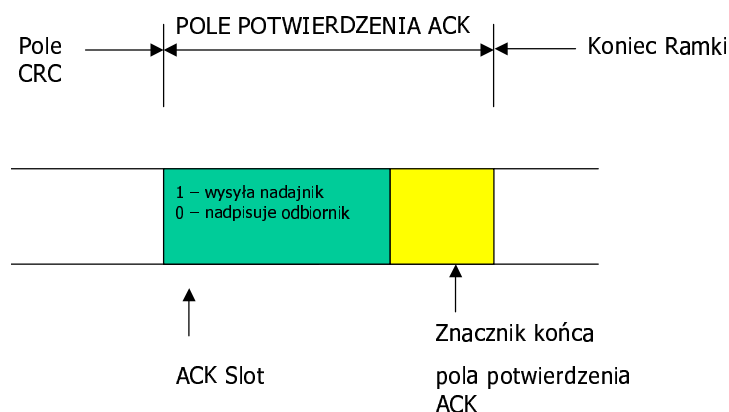
```

Po transmisji lub odebraniu ostatniego bitu pola danych DATA FIELD rejestr CRC_RG zawiera sekwencję CRC SEQUENCE.

Znacznik końca pola CRC (ang. *CRC DELIMITER*) występuje zarówno w ramce standardowej jak i rozszerzonej. Znacznik końca pola CRC jest transmitowany po sekwencji CRC SEQUENCE i ma zawsze wartość bitu recesywnego.

10.9.3.10 Pole potwierdzenia ACK (*ACK FIELD*)

Pole potwierdzenia odebrania wiadomości ACK FIELD występuje zarówno w ramce zwykłej jak i rozszerzonej. Pole ACK FIELD składa się z dwóch bitów (rys. 10.12). Pole to składa się bitu potwierdzenia ACK SLOT oraz bitu znacznika końca pola ACK ACK DELIMITER. W polu ACK FIELD węzeł nadający transmituje dwa bity recesywne. Węzeł odbiorczy, który odbierze prawidłową ramkę potwierdza odbiór wysyłając bit dominujący w przedziale czasu przeznaczonym na ACK SLOT.



Rys. 10.12 Pole potwierdzenia ACK

ACK DELIMITER

ACK DELIMITER jest znacznikiem końca dwubitowego pola ACK FIELD. Przyjmuje stan recesywny. W rezultacie, okno ACK SLOT jest otoczone przez dwa bity recesywne (CRC DELIMITER oraz ACK DELIMITER).

10.9.3.11 Koniec ramki (*END OF FRAME*)

Koniec ramki EOF (ang. **END OF FRAME**) występuje w ramce standardowej oraz w ramce rozszerzonej. Każda ramka (zarówno ramka danych, jak i ramka żądania transmisji) jest zakończona polem EOF składającym się z sekwencji 7 bitów recesywnych.

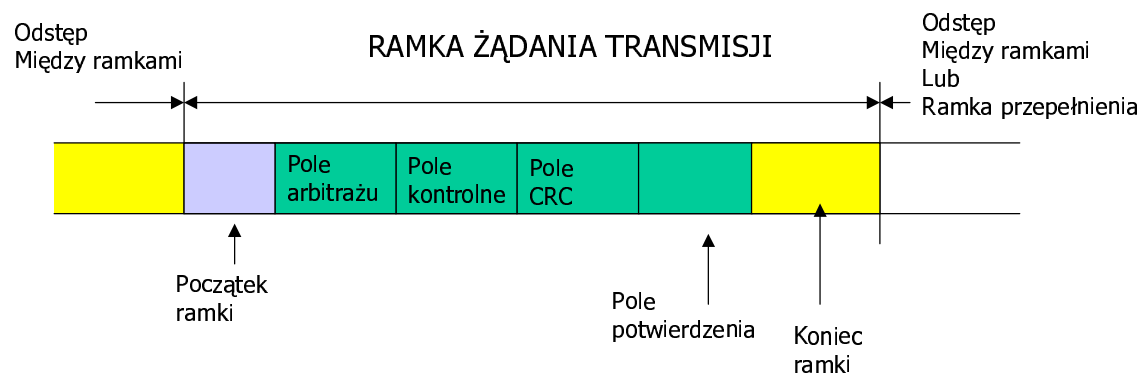
10.9.4 Ramka żądania transmisji (*REMOTE FRAME*)

Węzły sieci pracujące w trybie odbiorczym mogą inicjować transfery niezbędnych im danych pochodzących z innych węzłów wysyłając *Ramkę Żądania Transmisji* (ang. *REMOTE FRAME*). Ramka żądania transmisji występuje zarówno w trybie transmisji ramki standardowej jak również w trybie transmisji ramki rozszerzonej. W obydwu przypadkach składa się z 6 różnych pól bitowych (rys. 14):

1. Pole początku ramki – START OF FRAME
2. Pole arbitrażu – ARBITRATION FIELD
3. Pole kontrolne – CONTROL FIELD
4. Pole sumy kontrolnej – CRC FIELD

5. Pole potwierdzenia odbioru – ACK FIELD
6. Pole końca ramki – END OF FRAME.

W przeciwieństwie do *ramki danych*, bit RTR w ramce żądania transmisji przyjmuje stan recesywny. Nie występuje również pole danych DATA FIELD i to bez względu na wartość jaką przyjmuje pole *DLC* (dopuszczalne są wartości od 0 do 8). W tym przypadku wartość pola *DLC* odpowiada *DLC* ramki danych, która będzie odpowiedzią na żądanie transmisji.



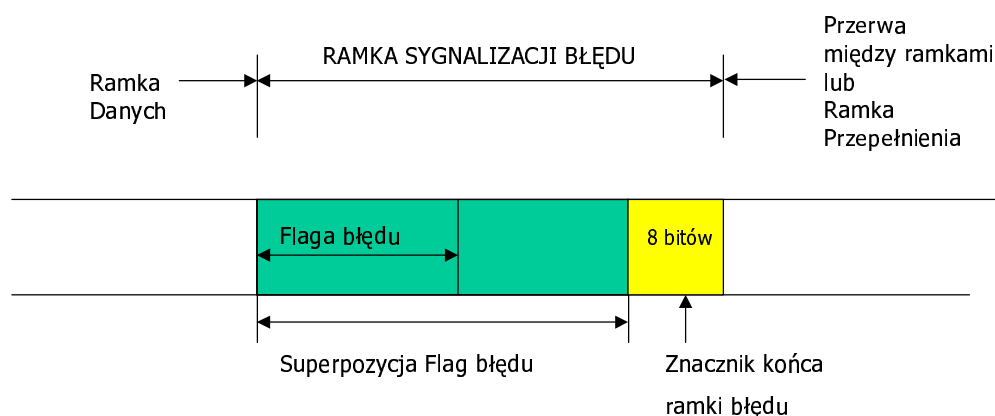
Rys. 10.13 Ramka żądania transmisji

O tym czy ramka jest *ramką danych* czy też *ramką żądania transmisji* decyduje wartość bitu RTR, który jest:

- ❑ recesywny – w ramce zdalnego żądania transmisji (REMOTE FRAME),
- ❑ dominujący – w ramce danych (DATA FRAME).

10.9.5 Ramka sygnalizacji błędu (ERROR FRAME)

Ramka sygnalizacji błędu (ang. *ERROR FRAME*) (rys. 10.14) składa się z dwóch pól. Pierwsze z nich powstaje jako superpozycja znaczników błędu ERROR FLAGS pochodzących z różnych węzłów. Drugie pole jest znacznikiem końca ramki (ang. *ERROR DELIMITER*).



Rys. 10.14 Ramka sygnalizacji błędu transmisji

W celu poprawnego zakończenia ramki błędu *ERROR FRAME*, węzeł pracujący w trybie *error passive* wymaga aby magistrala znajdowała się w trybie czuwania

przez czas potrzebny na transmisję przynajmniej 3 bitów, co ma np. miejsce w przypadku wystąpienia lokalnego błędu w tym węźle. Z tego powodu sieć CAN nie powinna nigdy pracować przy 100% obciążeniu.

Flaga błędu (ERROR FLAG)

Rozróżniane są 2 rodzaje flag błędów:

1. aktywna (ACTIVE ERROR FLAG) – składająca się z sześciu kolejnych bitów dominujących.
2. pasywna (PASSIVE ERROR FLAG) – składająca się z sześciu kolejnych bitów recesywnych (chyba że zostaną one zdominowane przez dominujący bit innego węzła).

Węzeł pracujący jako *error active* sygnalizuje wykrycie błędu wysyłając aktywną flagę błędu. Flaga błędu narusza reguły kodowania danych (*bit stuffing*) stosowane dla wszystkich pól począwszy od SOF do CRC DELIMITER lub też niszczy ustaloną postać ACK FIELD lub EOF. W wyniku takiego działania wszystkie węzły sieci wykrywają błąd i rozpoczynają transmisję ERROR FLAGS. Stąd właśnie pole ERROR FLAG jest superpozycją bitów dominujących pochodzących z różnych węzłów. Faktyczna długość tej sekwencji bitów waha się w przedziale <6..12>.

Węzeł pracujący w trybie *error passive* sygnalizuje wykrycie błędu transmitując PASSIVE ERROR FLAG. Flaga ta uznawana jest za kompletną wtedy, gdy urządzenie transmitujące ją wyśle i jednocześnie odczyta 6 kolejnych bitów o tej samej wartości.

Znacznik końca ramki (ERROR DELIMITER)

Znacznik końca ramki składa się z 8 kolejnych bitów recesywnych. Po transmisji flagi błędu ERROR FLAG, każde urządzenie transmituje bit recesywny i jednocześnie monitoruje magistralę czy taki bit na niej się pojawił, a następnie transmituje pozostałe 7 bitów recesywnych.

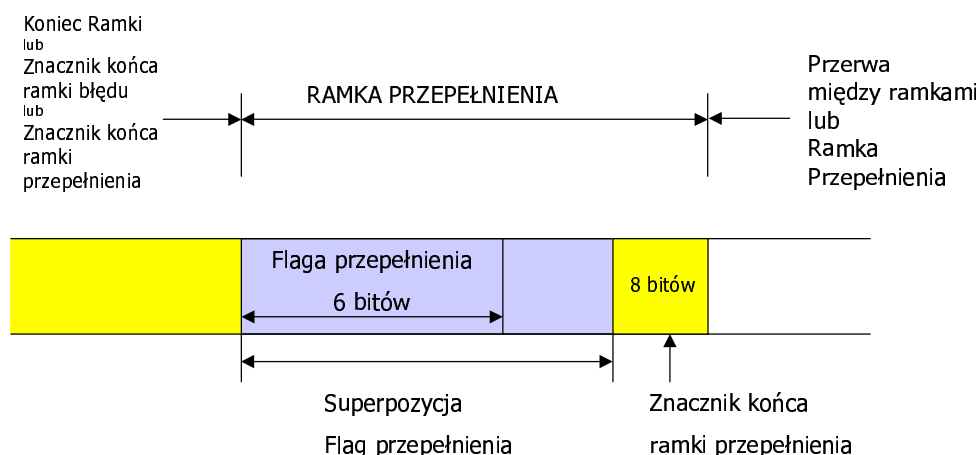
10.9.6 Ramka przepełnienia (OVERLOAD FRAME)

Ramka przepełnienia (OVERLOAD FRAME) składa się z dwóch pól: OVERLOAD FLAG oraz OVERLOAD DELIMITER (rys. 10.15). Rozróżniane są trzy przypadki przepełnienia prowadzące do wysłania OVERLOAD FLAG:

1. stan, w którym węzeł odbierający dane wymaga czasu przed odebraniem kolejnej ramki danych lub ramki żądania transmisji,
2. stan, w którym wykryty zostanie bit dominujący w pierwszym i drugim bicie INTERMISSION,
3. jeżeli węzeł pracujący w sieci CAN zarejestruje w ramce ERROR DELIMITER lub też OVERLOAD DELIMITER ósmy bit jako dominujący i rozpocznie transmisję ramki przepełnienia OVERLOAD FRAME zamiast ramki błędu ERROR FRAME. Wówczas nie zwiększy się także wartość licznika błędów.

W przypadku wystąpienia sytuacji opisanej w punkcie 1 rozpoczęcie transmisji ramki przepełnienia jest możliwe tylko z pierwszym bitem oczekiwanej przerwy (INTERMISSION). W pozostałych dwóch przypadkach transmisja rozpoczyna się w z przerwą jednobitową po zarejestrowaniu bitu dominującego.

Dopuszczalne jest wygenerowanie, co najwyżej 2 ramek przepełnienia w celu opóźnienia transmisji kolejnych ramek danych lub ramki żądania transmisji.



Rys. 10.15 Ramka przepełnienia

OVERLOAD FLAG

Flaga przepełnienia składa się z sześciu bitów dominujących i jest zgodna z ACTIVE ERROR FLAG. Kolejność w jakiej w ramce przepełnienia transmitowane są bity niszczy ustalony schemat pola przerwy INTERMISSION. W konsekwencji tego wszystkie węzły sieci również rejestrują warunek przepełnienia i rozpoczynają transmisje flagi przepełnienia. Wykrycie bitu dominującego jako trzeciego bitu podczas przerwy (INTERMISSION) interpretowane jest jako start (SOF) nadawania nowej ramki.

UWAGI:

Kontrolery, które mają zaimplementowany moduł CAN w wersji protokołu 1.0 lub 1.1 w odmienny sposób interpretują trzeci bit w polu przerwy (INTERMISSION). W przypadku wykrycia bitu dominującego następuje przyjęcie pierwszego bitu z sekwencji sześciu, jako bitu startu nowej ramki SOF. Szósty bit dominujący naruszy reguły kodowania bitów i spowoduje wystąpienie błędu.

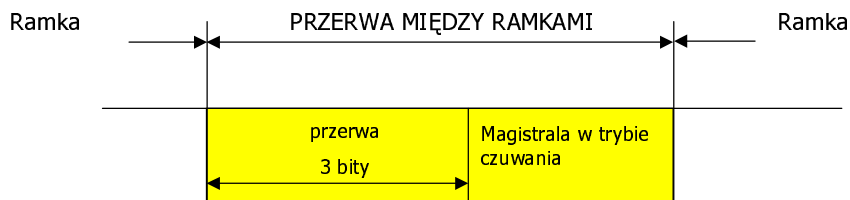
Znacznik końca ramki przepełnienia (ang. *OVERLOAD DELIMITER*) składa się z 8 bitów recesywnych. Znacznik końca ramki przepełnienia ma taką samą budowę jak znacznik końca ramki błędu ERROR DELIMITER. Po wysłaniu flagi przepełnienia OVERLOAD FLAG, węzły monitorują sieć dopóki nie zarejestrują zmiany bitu dominującego na recesywny. Od tej chwili wszystkie węzły sieciowe kończą nadawanie flagi przepełnienia OVERLOAD FLAG i jednocześnie rozpoczynają transmisję siedmiu kolejnych bitów recesywnych.

10.9.7 Przerwa między ramkami (INTERFRAME SPACING)

Ramki danych DATA FRAME oraz ramki żądania transmisji REMOTE FRAME są oddzielane od poprzedzających je (bez względu na rodzaj ramki poprzedzającej) ciągiem bitów zwanym INTERFRAME SPACE. Przerwa ta nie występuje przed ramkami przepełnienia oraz błędów (OVERLOAD oraz ERROR FRAME). Nie oddziela też ona występujących po sobie kolejno dwóch ramek przepełnienia OVERLOAD

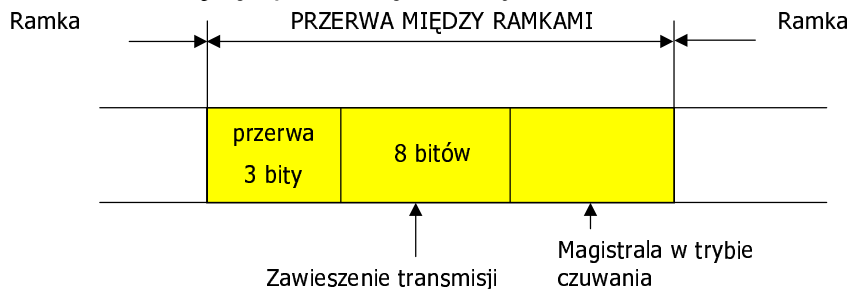
FRAME (w przypadku, gdy taka transmisja wystąpi). Przerwa między ramkami (ang. *INTERFRAME SPACE*) składa się z pola przerwy (ang. *INTERMISSION*) oraz pola czuwania (ang. *BUS IDLE*) (rys. 10.16) i pola zawieszenia transmisji (ang. *SUSPEND TRANSMISSION*) dla węzłów pracujących w trybie pasywnym (*error passive*).

Dla węzłów, które nie pracują w trybie *error passive* lub były odbiorcą poprzedniej wiadomości obowiązuje przerwa jak na rys. 10.16.



Rys. 10.16 Przerwa pomiędzy ramkami I

Dla węzłów, które pracują w trybie *error passive* i były nadawcą poprzedniej wiadomości obowiązuje przerwa jak na rys. 10.17.



Rys. 10.17 Przerwa między ramkami II

INTERMISSION

Pole przerwy *INTERMISSION* składa się z kolejnych 3 bitów recesywnych. Jedyną akcją, jaka może być podjęta w czasie trwania przerwy *INTERMISSION* jest sygnalizacja przepełnienia *OVERLOAD*. Rozpoczęcie transmisji danych lub żądania transmisji, jest w takiej sytuacji niedozwolone.

Uwaga: Jeżeli węzeł sieci CAN, który posiada wiadomość oczekującą na wysłanie, wykryje bit dominujący na trzeciej pozycji pola *INTERMISSION*, to zinterpretuje go jako SOF i rozpocznie nadawanie poczynając od pierwszego bitu pola *INDETIFIER* (z pominięciem bitu SOF).

Tryb czuwania magistrali (BUS IDLE)

Czas przebywania sieci w trybie czuwania nie jest możliwy do zdefiniowania. Jeżeli węzły wykryją, że sieć jest w stanie czuwania, to mogą w każdej chwili rozpocząć transmisję. Wiadomość, która oczekuje na wysłanie, jest transmitowana bezpośrednio po wystąpieniu przerwy *INTERMISSION*.

Zawieszenie transmisji (SUSPEND TRANSMISSION)

Węzeł, który pracuje w trybie *error passive*, po wysłaniu wiadomości, wysyła dodatkowo osiem bitów recesywnych zaraz po wystąpieniu przerwy *INTERMISSION*, a przed rozpoczęciem kolejnej transmisji lub wykryciu trybu czuwania w sieci. Jeżeli

w tym czasie jakiś węzeł rozpocznie nadawanie, to węzeł pracujący w trybie *error passive* rozpocznie odbiór tej wiadomości.

10.10 Wymagania stawiane budowie ramki

Opisany powyżej format ramki standardowej jest zgodny ze specyfikacją CAN 1.2. Ponieważ pojawiła się również specyfikacja ramki rozszerzonej, to ze względu na zachowanie zgodności wstecznej wymaga się aby:

- każdy nowy kontroler CAN wspierał bezwzględnie standardowy format ramki,
- każdy nowy kontroler CAN mógł odbierać wiadomości formatu rozszerzonego,
- ramki rozszerzone nie były uznawane za błędne z powodu swojego formatu,

Nie jest jednak wymagane, aby nowe kontrolery CAN wspierały ramki rozszerzone.

10.11 Nadajnik i odbiornik

NADAJNIK

Nadajnik jest węzłem tworzącym wiadomość. Węzeł pozostaje nadajnikiem tak długo jak długo sieć nie przejdzie w tryb czuwania lub też przegra arbitraż.

ODBIORNIK

Odbiornik jest węzłem, który nie jest nadajnikiem wiadomości, a sieć nie znajduje się w trybie czuwania.

10.12 Filtrowanie wiadomości

Wiadomości filtrowane są w oparciu o pełny identyfikator wiadomości. Dzięki zastosowaniu opcjonalnego rejestru maskującego możliwe jest skonfigurowanie operacji filtrowania w taki sposób, żeby przepuszczane były wszystkie wiadomości lub wybrane grupy wiadomości. W przypadku gdy w węźle zostanie zaimplementowany rejestr maskujący, to wymaga się, aby każdy bit tego rejestru był w pełni programowalny (musi istnieć możliwość zablokowania lub odblokowania operacji filtrowania). Długość rejestru maski powinna odpowiadać całemu identyfikatorowi bądź tylko jego fragmentom.

10.13 Klasyfikacja wiadomości

W zależności od aktualnego trybu pracy węzła (nadajnik czy odbiornik) moment w którym wiadomość zostaje uznana za poprawną jest różny.

Nadajnik

Nadajnik uznaje wiadomość za poprawną, jeżeli nie zostanie zarejestrowany żaden błąd do momentu transmisji EOF. Jeżeli wiadomość zostanie uznana za błędną, to automatycznie zostanie przeprowadzona retransmisja wiadomości. Retransmisja musi być przeprowadzona natychmiast po przejściu sieci wejdzie w tryb czuwania, tak aby umożliwić oczekującej wiadomości rywalizację o dostęp do sieci z pozostałymi wiadomościami.

Odbiornik

Odbiornik uznaje wiadomość za poprawną, jeżeli nie zarejestruje żadnego błędu aż do przedostatniego bitu pola EOF. Ostatni bit pola EOF nie jest brany pod uwagę.

10.14 Kodowanie ciągu bitów

Kodowanie ciągu bitów

Ciąg obejmujący bity pól: SOF, ARBITRATION FIELD, CONTROL FIELD, DATA FIELD, oraz CRC FIELD jest kodowany metodą zwaną *bit stuffing*. Metoda ta polega na tym, że jeżeli podczas transmisji zostanie wykryta sekwencja 5 kolejnych bitów o takiej samej wartości logicznej, to wstawiany jest dodatkowy bit przeciwnej wartości logicznej. Pozostałe pola wchodzące w skład ramek DATA FRAME i REMOTE FRAME (CRC DELIMITER, ACK FIELD oraz EOF) nie podlegają tej metodzie kodowania. ERROR FRAME oraz OVERLOAD FRAME mają stałą i zdefiniowaną strukturę i w związku z tym nie podlegają procedurze kodowania metodą *bit stuffing*. Do kodowania poszczególnych bitów w warstwie fizycznej stosowana jest metoda NRZ (ang. *Non-Return-to-Zero*). Została ona zilustrowana na rys. 10.6.

10.15 Zarządzanie błędami (ERROR HANDLING)

10.15.3 Wykrywanie błędów

Rozróżnia się 5 różnych rodzajów błędów, które nie wykluczają się wzajemnie:

□ Błąd pojedynczego bitu (BIT ERROR)

Węzeł, który nadaje wiadomość jednocześnie ją monitoruje. Błąd BIT ERROR występuje w przypadku, gdy wartość bitu odczytana z sieci jest inna niż aktualnie transmitowana. Wyjątek stanowią sytuacje, w których transmitowany jest bit recesywny w polu ARBITRATION FIELD oraz podczas trwania ACK SLOT. W tych przypadkach nie jest sygnalizowany błąd, gdy zostanie odczytany bit dominujący. Nadajnik, który wykryje bit dominujący podczas transmisji PASSIVE ERROR FLAG nie interpretuje tego jako BIT ERROR.

□ Błąd kodowania (STUFF ERROR)

Błąd kodowania występuje, gdy podczas transmisji pola ramki, które powinno być kodowane metodą *bit stuffing* w sieci zostanie wykrytych 6 kolejnych bitów o takiej samej wartości logicznej.

□ Błąd sumy kontrolnej (CRC ERROR)

Błąd ten występuje, gdy pole sumy kontrolnej CRC obliczone na podstawie odebranej wiadomości przez odbiornik, nie zgadza się z polem CRC dołączonym do wiadomości przez nadajnik.

□ Błąd formatu ramki (FORM ERROR)

Błąd ten występuje w przypadku, gdy naruszona zostanie ustalony format pola bitowego transmitowanej ramki (sytuacja w której odbiornik wykryje bit dominujący w miejscu ostatniego bitu pola EOF nie jest traktowana jako błąd formatu ramki)

❑ Błąd potwierdzenia ramki (ACKNOWLEDGEMENT ERROR)

Błąd ten wykrywany jest przez nadajnik w przypadku braku odczytania bitu dominującego podczas transmisji pola ACK SLOT.

10.15.4 Sygnalizacja błędów (ERROR SIGNALING)

Węzeł dołączony do sieci CAN sygnalizuje wykrycie błędu transmitując flagę błędu ERROR FLAG. Dla urządzeń pracujących w trybie *error active* jest to ACTIVE ERROR FLAG i odpowiednio dla węzłów pracujących w trybie *error passive* jest to PASSIVE ERROR FLAG.

Dla błędów typu:

- BIT ERROR
- STUFF ERROR
- FORM ERROR
- ACKNOWLEDGEMENT ERROR

Flaga błędu ERROR FLAG jest transmitowana począwszy od kolejnego bitu następującego po jego wykryciu.

W przypadku wykrycia błędu sumy kontrolnej CRC, transmisja flagi błędu rozpoczyna się począwszy od bitu następującego po transmisji sekcji ACK DELIMITER, chyba że rozpoczęta została już transmisja flagi błędu wynikająca z innego warunku.

10.16 Ograniczanie wpływu uszkodzeń (FAULT CONFINEMENT)

Z uwagi na sposób obsługi błędów, węzeł sieci CAN może znajdować się w jednym z 3 trybów pracy:

- ❑ *error active*,
- ❑ *error passive*,
- ❑ *bus off*,

Węzeł pracujący w trybie *error active* może brać udział w komunikacji w sieci, a w przypadku wykrycia błędu wysyłać ACTIVE ERROR FLAG.

Węzeł pracujący w trybie *error passive* może brać udział w komunikacji w sieci, ale w przypadku wykrycia błędu może wysyłać tylko PASSIVE ERROR FLAG. Ponadto po zakończeniu transmisji węzeł pracujący w tym trybie rozpoczyna okres oczekiwania przed rozpoczęciem kolejnej transmisji (zobacz SUSPEND TRANSMISION).

Węzeł pracujący w trybie *bus off* nie może mieć żadnego wpływu na działanie sieci. Tryb taki występuje np. w przypadku wyłączenia (zablokowania) trancivera (sterownika magistrali) węzła.

W celu kontroli i ograniczania błędów, każdy węzeł CAN posiada dwa liczniki:

1. licznik błędów transmisji (TRANSMIT ERROR COUNT)
2. licznik błędów odbioru (RECEIVE ERROR COUNT)

Stany tych liczników są zmieniane wg następujących reguł:

1. W przypadku wykrycia błędu przez *odbiornik*, wartość rejestru licznika błędów odbioru jest zwiększana o 1, poza przypadkami wykrycia błędu pojedynczego bitu BIT ERROR podczas transmisji ACTIVE ERROR FLAG bądź OVERLOAD FLAG.
2. Jeżeli *odbiornik* wykryje bit dominujący jako pierwszy po wysłaniu flagi błędu ERROR FLAG, wartość rejestru licznika błędów odbioru przyrasta o 8.
3. Jeśli *nadajnik* wysłał flagę błędu ERROR FLAG, to oznacza, że jednocześnie stan licznika błędów transmisji ulegnie zwiększeniu o 8.
4. Jeżeli *nadajnik* wykryje błąd pojedynczego bitu podczas wysyłania ACTIVE ERROR FLAG lub OVERLOAD FLAG, to wartość licznika błędów transmisji ulegnie zwiększeniu o 8.
5. Jeżeli *odbiornik* wykryje błąd pojedynczego bitu podczas wysyłania ACTIVE ERROR FLAG lub OVERLOAD FLAG, to wartość licznika błędów odbioru ulegnie zwiększeniu o 8.
6. Każdy węzeł toleruje wystąpienie do 7 kolejnych bitów dominujących po wysłaniu ACTIVE i PASSIVE ERROR FLAG lub OVERLOAD FLAG. W przypadku wykrycia 14-stego kolejnego bitu dominującego (dla ACTIVE ERROR FLAG lub OVERLOAD FLAG) lub po wykryciu 8-go kolejnego bitu dominującego występującego po PASSIVE ERROR FLAG, oraz po każdej dodatkowej sekwencji ośmiu kolejnych bitów dominujących każdy *nadajnik* zwiększa wartość swojego rejestru licznika błędów transmisji o 8. Analogicznie *odbiornik* o 8 zwiększa wartość swojego licznika błędów odbioru.
7. Po każdej udanej transmisji wiadomości (odczytanie potwierdzenia ACK oraz brak wystąpienia błędów aż do końca EOF) wartość licznika błędów transmisji jest zmniejszana o 1 chyba że już wynosi 0.
8. Po udanym odbiorze wiadomości (bez wystąpienia błędów aż do ACK SLOT oraz po udanym nadaniu bitu potwierdzenia ACK) wartość licznika błędów odbioru jest zmniejszana o 1, jeśli jego wartość mieści się w przedziale od 1 do 127. Jeżeli jego wartość wynosi 0, to pozostaje 0, a gdy była większa niż 127 zostaje ona ustawiona na wartość z przedziału 119-127.
9. Węzeł pracuje w trybie *error passive*, jeżeli wartość jego licznika błędów transmisji jest równa lub przekroczy 128 lub, gdy wartość jego licznika błędów odbioru jest równa lub przekroczy 128. Błąd, za przyczyną którego węzeł przechodzi w tryb *error passive* powoduje wygenerowanie ACTIVE ERROR FLAG.
10. Węzeł pracuje w trybie wyłączonej magistrali *bus off*, kiedy wartość rejestru jego licznika błędów transmisji jest większa niż 256.
11. Węzeł może przejść z trybu *error passive* do *error active*, gdy oba jego liczniki (błędów transmisji oraz błędów odbioru) mają wartości mniejsze lub równe 127.
12. Układ pracujący w trybie *bus off* może ponownie przejść do trybu *error active* z wartością liczników błędów równą 0 po zarejestrowaniu 128 wystąpień 11 bitowych ciągów bitów recesywnych na magistrali.

Uwagi:

- Jeżeli wartości rejestrów liczników błędów przekraczają 96, to oznacza, że występują silne zakłócenia w sieci. Powinno to być jednocześnie sygnałem do przeprowadzenia dokładnego monitorowania magistrali w celu lokalizacji źródeł problemu.

- Wyjątki, w których licznik błędów transmisji nie ulega zmianie
 - Jeżeli *nadajnik* pracuje w trybie *error passive* i rozpozna błąd potwierdzenia ramki (ACKNOWLEDGEMENT ERROR) polegający na braku dominującego bitu ACK oraz nie wykryje bitu dominującego podczas nadawania własnej flagi PASSIVE ERROR FLAG.
 - W przypadku, gdy nadajnik wysyła flagę błędu ERROR FLAG spowodowaną błędem kodowania STUFF ERROR podczas arbitrażu.
- Jeżeli podczas uruchamiania sieci tylko 1 węzeł jest w trybie *online* i transmituje jakąkolwiek wiadomość, to nie otrzyma on żadnego potwierdzenia. W związku z tym wykryje błąd i powtórzy transmisję. W takiej sytuacji może on przejść do trybu *error passive*, ale nie jest dozwolone jego przejście do trybu *bus off*.

10.17 Częstotliwość generatora podstawy czasu

Maksymalna tolerancja częstotliwości generatora podstawy czasu jest równa 1,58%. W związku z tym możliwe jest stosowanie rezonatora ceramicznego taktującego węzeł CAN dla prędkości transmisji do 125 kb/s.

W przypadku sieci CAN pracującej z pełną prędkością wymagane jest zastosowanie oscylatora kwarcowego. Węzły CAN zgodne ze specyfikacjami 1.0 i 1.1 pracujące wspólnie w jednej sieci muszą być wyposażone w oscylator kwarcowy.

Węzły CAN zgodne ze specyfikacjami 1.2 lub późniejszymi mogą być wyposażone w oscylator ceramiczny.

10.18 Wymagania dotyczące taktowania (BIT TIMING REQUIREMENTS)

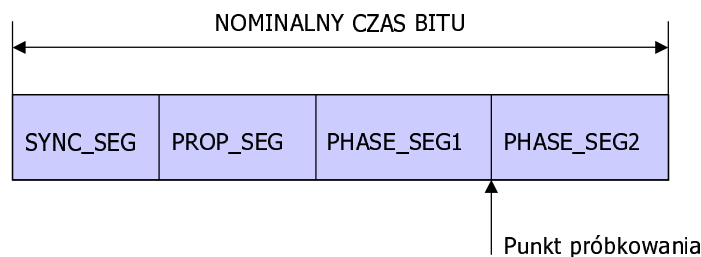
Nominalna prędkość transmisji N (NOMINAL BIT RATE) jest liczbą bitów transmitowanych w ciągu sekundy przy założeniu wykorzystania idealnego nadajnika oraz braku desynchronizacji.

Okres czas trwania transmisji pojedynczego bitu T (BIT TIME) jest definiowany jako:

$$T = \frac{1}{N} \quad (3)$$

Nominalny czas trwania transmisji pojedynczego bitu można podzielić teoretycznie na kilka nie zachodzących na siebie segmentów czasowych (rys. 10.18):

- segment synchronizacji (SYNC_SEG)
- segment propagacji (PROP_SEG)
- segment fazy 1 (PHASE_SEG1)
- segment fazy 2 (PHASE_SEG2)



Rys. 10.18 Nominalny czas bitu

SYNC_SEG

Segment SYNC_SEG wykorzystywany jest do synchronizowania różnych węzłów podłączonych do magistrali. W tym segmencie powinno znajdować się zbocze sygnału kodującego przesyłany bit.

PROP_SEG

W tym segmencie następuje buforowanie skutków wystąpienia fizycznych opóźnień występujących w medium. Czas opóźnienia sygnału w medium jest równy podwojonej sumie czasu propagacji sygnału na magistrali, opóźnienia w układzie wejściowym i opóźnieniu w układzie wyjściowym tranceivera.

PHASE_SEG1, PHASE_SEG2

Segmenty te wykorzystywane są do kompensacji błędów synchronizacji. Mogą się one być wydłużane lub skracane w wyniku operacji resynchronizacji.

Punkt próbkowania (SAMPLE POINT)

Chwila (punkt) próbkowania jest momentem czasowym, w którym jest odczytywany poziom elektryczny sygnału medium, i w którym przypisywana jest wartość logiczna temu poziomowi. Chwila próbkowania występuje na końcu segmentu PHASE_SEG1.

Czas przetwarzania informacji

Czas przetwarzania informacji jest czasem liczonym od momentu próbkowania i koniecznym na analizę poziomu odbieranego bitu.

Kwant czasu τ_q – TIME QUANTUM

Kwant czasu jest elementarną jednostką czasu pochodną względem okresu generatora podstawy czasu. Istnieje programowalny przelicznik o całkowitych wartościach z przedziału 1 do 32. Począwszy od minimalnego czasu kwantu, czas ten może mieć długości:

$$\tau_q = m \cdot \tau_{\min} \quad (4)$$

gdzie: τ_{\min} - minimalny czas trwania kwantu
 m - multiplikator (1..32).

Czasy trwania segmentów czasowych wynoszą odpowiednio:

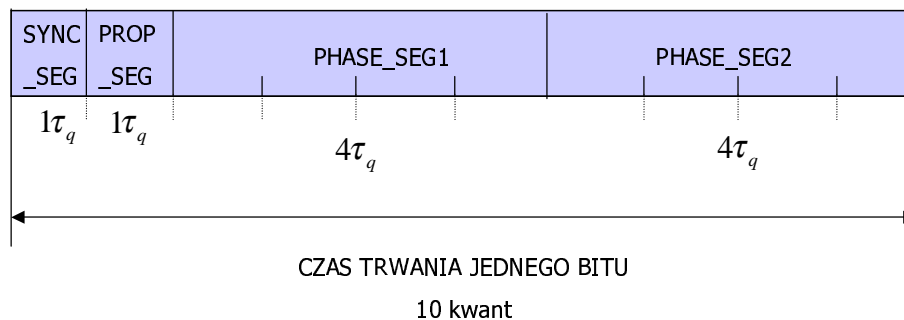
- SYNC_SEG – 1 τ_q
- PROP_SEG – jest programowalny w zakresie 1-8 τ_q
- PHASE_SEG1 – jest programowalny w zakresie 1-8 τ_q

- PHASE_SEG2 – równy jest maksymalnej długości czasu PHASE_SEG1 oraz czasu przetwarzania informacji
- Czas przetwarzania informacji – jest krótszy lub równy dwukrotnej wartości τ_q

Całkowita liczba kwantów w okresie bitu musi być programowalna w zakresie co najmniej od 8 do 25.

Częstą i celową praktyką jest konstruowanie węzłów CAN w taki sposób, aby nie było konieczności stosowania różnych generatorów dla potrzeb układu komunikacji i jednostki centralnej CPU węzła. W celu uzyskania odpowiedniej częstotliwości generatora podstawy czasu dla celów komunikacyjnych, niezbędna jest w tym przypadku programowa możliwość zmiany czasu trwania transmisji bitu (*bit timing*).

Dopasowanie chwili próbkowania w sieci, w której wszystkie węzły mają programowalne czasy transmisji bitu nie jest problemem. W przypadku braku możliwości programowania czasu trwania kwantu, czas trwania transmisji jednego bitu węzłów CAN powinien być zgodny z definicją przedstawioną na rys. 10.19.



Rys. 10.19 Podział bitu na segmenty

Synchronizacja sprzętowa - HARD SYNCHRONIZATION

W sieci CAN stosowana jest transmisja synchroniczna. Oznacza to, że rozpoczęcie odbioru informacji w odbiorniku jest synchronizowane z nadajnikiem. Odbiornik czeka z wygenerowaniem segmentu synchronizacyjnego SYNC_SEG tak długo jak nie nadejdzie odpowiednie zbocze synchronizujące generowane przez nadajnik.

RESYNCHRONIZATION JUMP WIDTH

W wyniku wystąpienia ponownej synchronizacji okres PHASE_SEG1 może się wydłużyć lub też PHASE_SEG2 może ulec skróceniu. Całkowita zmiana długości trwania segmentu fazy PHASE BUFFER SEGMENT jest ograniczona od góry przez parametr RESYNCHRONIZATION JUMP WIDTH.

Wartość *RESYNCHRONIZATION JUMP WIDTH* powinna być programowalna w zakresie od 1 do min. 4 PHASE_SEG1. Do celów synchronizacji wykorzystywane są zbocza odbieranego sygnału. Ponieważ maksymalna liczba bitów o takiej samej wartości występujących w trakcie transmisji jest ograniczona to istnieje możliwość realizacji resynchronizacji węzła także w trakcie odbioru informacji. Maksymalna odległość czasowa pomiędzy kolejnymi zboczami synchronizującymi przejściami, które mogą być wykorzystane do ponownej synchronizacji wynosi 29 okresów transmisji jednego bitu.

Błąd fazy zbocza sygnału – PHASE ERROR

Błąd fazowy zbocza odbieranego sygnału wyznaczany jest względem segmentu synchronizacji SYNC_SEG i mierzony jest w kwantach czasu. Znak błędu fazowego określany jest w następujący sposób:

- $e=0$ jeżeli zbocze leży wewnątrz SYNC_SEG,
- $e>0$ jeżeli zbocze leży przed punktem próbkowania SAMPLE POINT,
- $e<0$ jeżeli zbocze leży za punktem próbkowania danego bitu.

Ponowna synchronizacja

W przypadku, gdy wartość błędu fazy PHASE ERROR zbocza sygnału, który wywołuje ponowną synchronizację jest mniejsza lub równa programowalnej wartości RESYNCHRONIZATION JUMP WIDTH to w efekcie mamy do czynienia z sytuacją podobną do synchronizacji sprzętowej.

Jeżeli wielkość błędu fazy jest większa niż RESYNCHRONIZATION JUMP WIDTH

- oraz, gdy błąd fazy jest dodatni, wtedy PHASE_SEG1 jest wydłużany o wartość równą RESYNCHRONIZATION JUMP WIDTH,
- oraz, gdy błąd fazy jest ujemny, wtedy PHASE_SEG2 jest skracany do wartości równej RESYNCHRONIZATION JUMP WIDTH

Zasady synchronizacji

1. Dozwolona jest tylko jedna synchronizacja w czasie trwania transmisji danego bitu.
2. Zbocze sygnału może być użyte do synchronizacji tylko wówczas, gdy wartość sygnału kodującego stan logiczny bitu zarejestrowana w poprzedzającym punkcie próbkowania jest różna od wartości sygnału zaraz po wystąpieniu tego zbocza.
3. W stanie czuwania medium, synchronizacja sprzętowa jest realizowana przez każde zbocze polegające na zmianie stanu z recesywnego na dominujący. Pozostałe zbocza występujące przy zmianach stanów medium z recesywnego do dominującego mogą być wykorzystane do ponownej synchronizacji, gdy spełniają punkty 1 i 2 z wyjątkiem gdy: węzeł nadający stan dominujący nie będzie przeprowadzał ponownej synchronizacji w wyniku dodatniego błędu fazy.

10.19. Warstwa aplikacyjna

10.19 CANopen – wprowadzenie

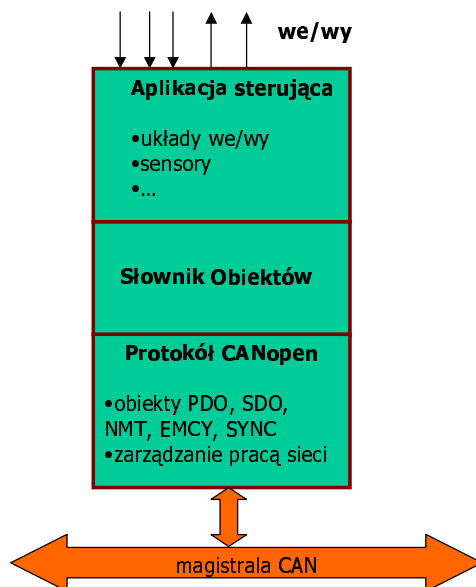
CANopen jest jednym z protokołów komunikacji warstwy wyższej wykorzystujących specyfikację sieci CAN. Został on zaprojektowany z myślą o zastosowaniach do automatyzacji przemysłowych procesów wytwórczych. Z biegiem czasu CANopen znalazł zastosowanie w wielu innych dziedzinach w tym także w budownictwie i transporcie.

Prace nad tym protokołem prowadzone były początkowo pod nadzorem firmy Bosch. Od 1995 roku rozwojem CANopen zajmuje się organizacja CiA (ang. CAN in Automation). Specyfikacje CiA obejmują warstwę aplikacji, profile komunikacyjne, strukturę urządzeń programowalnych, profile sprzętowe, zalecenia dotyczące okablowania oraz złącz. Aktualna wersja protokołu CANopen (CiA DS. 301) została objęta standardem i normą EN50325-4.

10.19.1 Model urządzenia

Urządzenie pracujące zgodnie z protokołem CANopen przedstawiane jest zazwyczaj w postaci modelu złożonego z trzech modułów:

- ❑ modułu odpowiedzialnego za komunikację (protokół CANopen),
- ❑ modułu zawierającego wiedzę o urządzeniu (słownik obiektów),
- ❑ aplikacji odpowiedzialnej za poprawną pracę urządzenia.



Rys. 10.20 Schemat modelu urządzenia

10.19.2 Słownik obiektów

Słownik obiektów jest najważniejszym elementem modelu urządzenia. Zawiera niezbędne informacje potrzebne do poprawnej pracy urządzenia i daje dostęp do:

- ❑ definicji wykorzystywanych typów danych w urządzeniu,
- ❑ parametrów komunikacji,
- ❑ danych i konfiguracji aplikacji.

W tab. 10.3 przedstawiono strukturę słownika obiektów.

Tab. 10.3 Struktura słownika obiektów

Indeks (hex)	Obiekt
0000	Wolny
0001 – 001F	Statyczne typy danych
0020 – 003F	Złożone typy danych
0040 – 005F	Charakterystyczne dane producenta
0060 – 007F	Charakterystyczne dane urządzenia (statyczne)
0080 – 009F	Charakterystyczne dane urządzenia (złożone)
00A0 – 0FFF	Zarezerwowany
1000 – 1FFF	Profil komunikacyjny
2000 – 5FFF	Profil charakteryzujący producenta
6000 – 9FFF	Znormalizowany profil urządzenia
A000 – FFFF	Zarezerwowany

Słownik obiektów może zawierać maksymalnie 65536 obiektów. Każdy obiekt jest adresowalny przez tzw. indeks. Zastosowane adresowania 16 bitowego pozwala na dostęp do prostych zmiennych. W przypadku adresowania złożonych struktur danych takich jak rekordy czy tablice, indeks odwołuje się wyłącznie do adresu danej struktury. Dostęp do obiektów złożonych możliwy jest dzięki zastosowaniu *subindeksu*. W takim przypadku *indeks* jest wskaźnikiem złożonej struktury danych, a *subindeks* wskazuje jej elementy.

10.19.3 Model komunikacji

Protokół CANopen wykorzystuje różne modele komunikacji pomiędzy urządzeniami w sieci. Wykorzystywane są różne obiekty komunikacji i usług, a także różne typy obiektów wyzwalania transmisji. Stosowana jest zarówno transmisja w trybie synchronicznym jak i asynchronicznym.

W przypadku transmisji w trybie synchronicznym nadawany jest specjalny obiekt synchronizacji (obiekt SYNC), który określa momenty, w których możliwa jest wymiana wiadomości.

Transmisja asynchroniczna pozwala na transmisję wiadomości w dowolnym momencie. W takim przypadku transmisja wyzwalana jest odpowiednio zdefiniowanym zdarzeniem lub chwilą czasową.

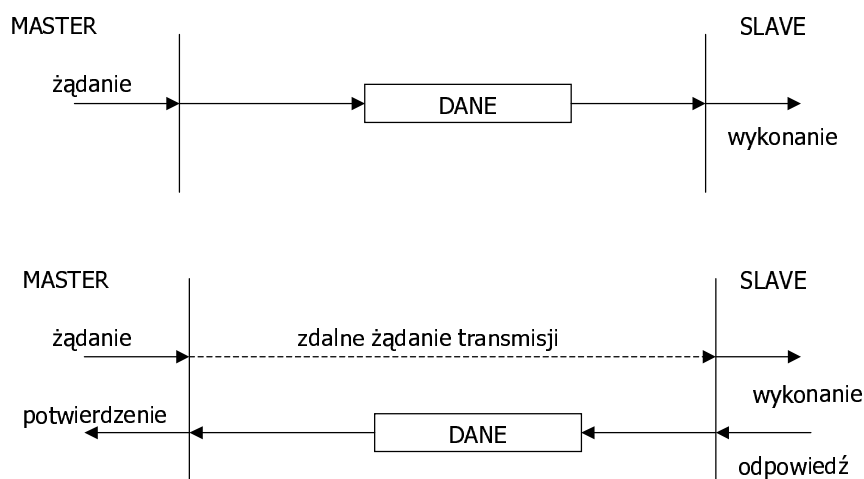
Niezależnie od tego wyróżniane są trzy następujące schematy współpracy urządzeń w sieci z protokołem CANopen:

- master-slave,
- klient-serwer,
- producent-konsument.

10.19.4 Współpraca typu master-slave

W przypadku schematu współpracy urządzeń typu master-slave, w każdym momencie czasowym występuje w sieci tylko i wyłącznie jedno urządzenie

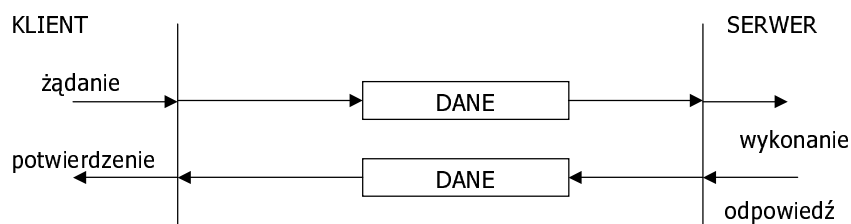
nadrzędne typu master. Pozostałe urządzenia traktowane są jako urządzenia podporządkowane. Urządzenie typu master wysyła żądanie do jednego lub wielu urządzeń podporządkowanych. Urządzenie podporządkowane odpowiada na żądanie urządzenia typu master. Ten model współpracy urządzeń jest podobny do modelu stosowanego np. w sieci MODBUS RTU (rozdział 5).



Rys. 10.21 Model komunikacji typu master-slave

10.19.5 Współpraca typu klient - serwer

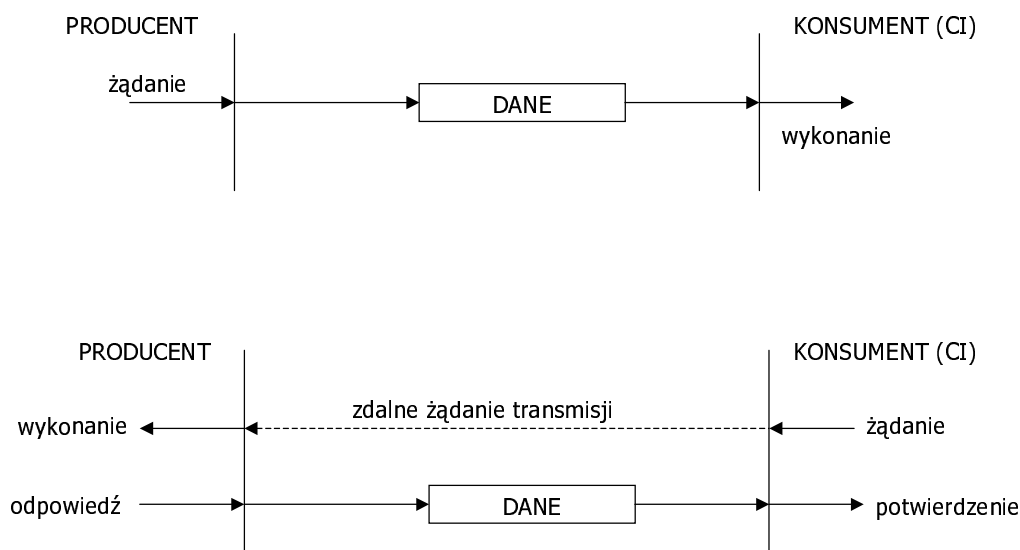
Ten model współpracy urządzeń polega na komunikacji pomiędzy jednym klientem i jednym serwerem. Klient wysyła do serwera żądanie wykonania określonej funkcji. Po jej wykonaniu, serwer wysyła odpowiedź potwierdzającą do klienta.



Rys. 10.22 Model komunikacji typu klient - serwer

10.19.6 Współpraca typu producent - konsument

Współpraca typu producent - konsument nazywana jest również współpracą typu *push-pull*. W tego typu komunikacji bierze udział jeden producent i od 0 do n konsumentów. Model *push* charakteryzuje się brakiem konieczności potwierdzenia odebrania wiadomości. W modelu *pull* komunikacja inicjowana jest przez konsumenta, a producent odpowiada. W tym przypadku potwierdzeniem odebrania żądania jest wysłanie odpowiedzi przez producenta.



Rys. 10.23 Model komunikacji typu producent - konsument

10.19.7 Obiekty komunikacji

Protokół CANopen na poziomie warstwy aplikacji korzysta z różnych sposobów wymiany informacji. Stosowane do tego tzw. obiekty komunikacyjne, które w zależności oferują różnego rodzaju funkcje i usługi takie jak np.: transmisja danych procesowych, zarządzanie i monitorowanie pracy sieci czy zdalna konfiguracja węzłów sieci.

Wszystkie obiekty i usługi oferowane przez CANopen działają przy założeniu poprawnej pracy sieci w warstwie łącza danych i poprawnej implementacji warstwy aplikacyjnej. Specyfikacją protokołu CANopen nie obejmuje przypadków, w których to założenie nie jest spełnione.

10.19.8 Obiekt PDO (Process Data Object)

Obiekty PDO służą do dynamicznej wymiany danych procesowych między urządzeniami sieci. Obiekty te mają wysoki priorytet i mogą zawierać do 8 bajtów danych w jednej wiadomości. Stosowana jest tu komunikacja typu producent-konsument.

Rozróżnia się dwa rodzaje obiektów PDO. Jeden służy do nadawania TPDO (ang. **Transmit-PDO**), a drugi do odbioru wiadomości RPDO (ang. **Receive-PDO**). Urządzenie wspierające transmisję obiektów TPDO pełni funkcję producenta natomiast, urządzenie wspierające odbiór obiektów RPDO pełni rolę konsumenta. Zazwyczaj urządzenia pracujące zgodnie z protokołem CANopen wspierają oba te tryby pracy. Odpowiednie parametry komunikacyjne umieszczone są w słowniku obiektów danego urządzenia.

10.19.9 Rodzaje trybów komunikacyjnych

Rozróżnić można następujące rodzaje komunikacji:

- asynchroniczna – transmisja może rozpocząć się w dowolnym momencie czasowym i może być:

- wyzwalana zdarzeniem – transmisja następuje po wystąpieniu określonego zdarzenia np. po zmianie wartości wejściowej,
- wyzwalana czasem – transmisja następuje po upływie określonego czasu,
- inicjowana zdalnym żądaniem transmisji – transmisja inicjowana jest przez żądanie transmisji wysłane przez konsumenta PDO.
- Synchroniczna – transmisja może nastąpić tylko w określonym przedziale czasowym, po wystąpieniu obiektu synchronizującego SYNC nadawanego przez jedno z urządzeń w sieci i może być:
 - cykliczna – transmisja odbywa się okresowo po każdym n -tym obiekcie SYNC (n jest zmiennym parametrem)
 - acykliczna – transmisja następuje po wystąpieniu obiektu SYNC, ale wcześniej musi być ona:
 - wyzwolona zdarzeniem lub
 - wyzwolona zdalnym żądaniem transmisji.

10.19.10 Typy wymiany danych

Z obiektem PDO skojarzone są dwa tryby wymiany danych – zapis i odczyt.

Tryb: zapis PDO

Producent PDO wysyła w obiekcie PDO dane. Dane te są odbierane przez 0 lub n węzłów. Konsument PDO nie wysyła żadnego potwierdzenia odbioru. Ten typ wymiany danych podobny jest do transmisji rozgłoszeniowej (broadcast). Jest to model wymiany danych typu *push*.

Tryb: odczyt PDO

W tym trybie jeden lub więcej konsumentów PDO wysyła żądanie zdalnej transmisji do producenta PDO. W odpowiedzi producent przesyła żądane dane. Ich odbiorcą może być od 0 do n konsumentów PDO. Jest to model wymiany danych typu *pull*.

10.19.11 Obiekt SDO (Service Data Object)

Obiekty SDO umożliwiają zapis i odczyt danych bezpośrednio do i ze słownika obiektów. Obiekty SDO przesyłane są zazwyczaj asynchronicznie z priorytetem niższym niż obiekty PDO, gdyż prędkość wymiany danych nie ma tu tak wielkiego znaczenia.

W tego typu komunikacji stosowany jest model wymiany danych typu punkt do punktu (ang. *peer to peer*), a więc zgodny ze specyfikacją klient-serwer. Klientem SDO jest w tym przypadku węzeł żądający wymiany danych (odczyt lub zapis z lub do słownika obiektów) natomiast serwerem SDO jest aplikacja udostępniająca swój słownik obiektów. Urządzeniem inicjującym wymianę danych (tworzącym tzw. kanał komunikacyjny) jest zawsze klient. Tego typu transmisja wymaga potwierdzenia, dlatego każde połączenie wymaga użycia dwóch wiadomości CAN (żądanie klienta i odpowiedź serwera). W zależności od liczby danych przeznaczonych do przesłania, stosowane są dwa typy transmisji:

- segmentowy (standardowy) – podstawowy typ transmisji,
- blokowy (opcjonalny) – wykorzystywany przy transmisji dużych bloków danych.

W obydwu typach transmisji stosowane są różne procedury wymiany danych.

10.19.12 Transmisja segmentowa

Z punktu widzenia klienta SDO transmisję segmentową można podzielić na:

- Odczyt SDO
 - inicjacja odczytu SDO,
 - odczyt danych SDO.
- Zapis SDO
 - inicjacja zapisu SDO,
 - zapis danych SDO.
- Anulowanie transmisji.

W transmisji segmentowej w jednym segmencie transmitowane może być maksymalnie 7 bajtów danych. Gdy liczba bajtów danych jest większa, dokonywany jest ich podział na segmenty.

Jeżeli liczba danych nie przekracza 4 bajtów, to dane te mogą być zawarte już w ramce inicjującej transmisję. Jest to tzw. transfer przyspieszony (ang. *expedited transfer*).

10.19.13 Transmisja blokowa

Z punktu widzenia klienta SDO transmisję blokową możemy podzielić na:

- Odczyt blokowy SDO
 - inicjacja odczytu bloku,
 - odczyt bloku,
 - zakończenie odczytu bloku.
- Zapis blokowy SDO
 - inicjacja zapisu bloku,
 - zapis bloku,
 - zakończenie zapisu bloku.
- Anulowanie transmisji.
-

10.19.14 Obiekt synchronizujący SYNC

W celu synchronizacji pracy elementów sieci CANopen stosuje się rozgłoszeniową transmisję obiektów typu SYNC. Obiekt SYNC jest transmitowany w regularnych odstępach czasu przez jeden z węzłów sieci zwany producentem SYNC. Okres transmisji jest określony przez parametr zwany okresem cyklu komunikacji (ang. *Communication Cycle Period*) i zapisany jest w słowniku obiektów urządzenia pod adresem 1006h. Parametr ten może być zmieniany w podczas procedury uruchamiania urządzenia.

Aby zapewnić obiektowi SYNC możliwie szybki dostęp do sieci nadawany jest mu odpowiednio wysoki priorytet. Obiekt SYNC nie zawiera żadnych danych. Transmisja odbywa się zgodnie z modelem producent - konsument. Producent w zdefiniowanych odstępach czasu wysyła obiekt SYNC do sieci. Wszystkie urządzenia (konsumenty SYNC), które tego potrzebują mogą być odbiorcą tych obiektów i za ich pomocą synchronizować swoją pracę z pozostałymi węzłami sieci.

10.19.15 Obiekt typu Time Stamp

Obiekt typu Time Stamp służy do przesyłania informacji o aktualnym czasie. Jest to transmisja rozgłoszeniowa. Obiekt o długości 6 bajtów z wysokim priorytetem wysyłany jest do sieci CAN przez producenta. Wysyłana jest liczba milisekund jakie upłynęły od północy oraz liczba dni począwszy od 1 stycznia 1984 roku.

10.19.16 Obiekt awaryjny EMCY

Obiekt typu EMCY (ang. *emergency*) służy do sygnalizowania wystąpienia awarii w urządzeniu. W przypadku zaistnienia awarii nadawana jest jedna ramka CAN z 8 bajtami danych. Posiada ona wysoki priorytet. Dane zawierają kod błędu. Obiekt EMCY implementowany jest opcjonalnie.

Wystąpienie błędu sygnalizowane jest tylko raz. Kody błędów są zawarte w słowniku obiektów urządzenia. Poza standardowymi kodami błędów istnieje możliwość definiowania własnych kodów przez poszczególnych producentów.

10.19.17 Obiekt NMT

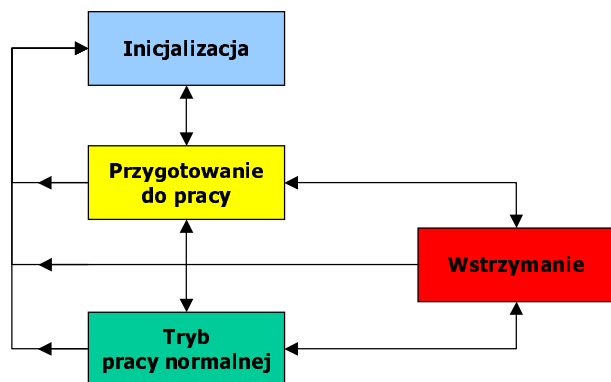
Zarządzanie siecią CANopen (ang. *Network Management*) polega na nadzorowaniu poszczególnych węzłów sieci i oparte jest na współpracy typu master - slave. Do zarządzania stosuje się obiekty NMT. Za ich pomocą możliwa jest inicjalizacja pracy urządzenia, jego uruchomienie, monitorowanie, czy też zatrzymanie.

Każde urządzenie pracujące w sieci CANopen musi umożliwiać pracę w trybie *slave* NMT. Urządzenia typu *slave* adresowane są za pomocą identyfikatorów węzłów NODE-ID. W sieci musi być też zdefiniowany jeden moduł zarządzający pracujący jako master. Zarządzanie siecią skupia się głównie na dwóch zadaniach. Jedno z nich to zarządzanie trybem pracy poszczególnych węzłów i sieci. Drugie to nadzór nad urządzeniem polegające głównie na kontroli jego aktywności.

10.19.18 Zarządzanie trybem pracy

Każde urządzenia w sieci CANopen może znajdować się jednym z 4 stanów:

1. STOPPED – wstrzymania pracy,
2. Pre-operational - gotowości do pracy,
3. Operational – pracy normalnej,
4. Initializing – inicjalizacji.



Rys. 10.24 Zależności między trybami pracy

Urządzenie master NMT odpowiedzialny jest za sterowanie trybem pracy sieci i jej poszczególnych urządzeń. Obiekt NMT steruje procesem uruchomienia sieci oraz zarządza trybem pracy poszczególnych węzłów.

Do komunikacji wykorzystywany jest obiekt NMT o długości 2 bajtów, z których pierwszy zawiera adres węzła, a drugi polecenie trybu pracy węzła, do którego ma on przejść. Możliwe jest też wysłanie poleceń do wszystkich węzłów jednocześnie.

10.19.19 Nadzór

W specyfikacji protokołu CANopen wyróżnione są 2 rodzaje usług umożliwiających ciągłe nadzorowanie pracy sieci. Są nimi: dozór węzłów (starszy sposób) oraz nowszy i zalecany w nowych implementacjach w postaci tzw. sygnału *heartbeat*.

Dozór aplikacji nad węzłami pracującymi w trybie *slave* realizowany przez odpytywanie (wysłanie zapytania o tryb pracy) w ustalonych odstępach czasu. Jeżeli urządzenie typu *slave* nie odpowie w określonym czasie, to sygnalizowane jest to w aplikacji sterującej pracą urządzenia typu *master*. Jeżeli urządzenie *slave* nie zostanie zapytane o stan w przeciągu określonego czasu, to również zostanie zasygnalizowane to aplikacji nim zarządzającej.

Działanie usługi *heartbeat* polega na cyklicznym wysyłaniu informacji o trybie swojej pracy do sieci. Jest to typowy przykład współpracy sieciowej typu producent - konsument. Producent wysyła sygnał potwierdzający jego obecność w sieci w regularnych odstępach czasu. Odbiorcami mogą być wszystkie urządzenia w sieci. Jeżeli w określonym czasie konsument nie otrzyma obiektu *heartbeat* z konkretnego węzła, to generowane jest zdarzenie informujące o tym lokalną aplikację zarządzającą pracą konsumenta.

10.19.20 Obiekt Boot-Up

Obiekt typu *Boot-up* transmitowany jest do sieci w celu przesłania informacji, że obiekt który ją wysłał osiągnął stan gotowości do pracy (pre-operational). Dzieje się tak zawsze po starcie lub restarcie urządzenia. Obiekt typu *Boot-up* ma ten sam identyfikator co obiekt *heartbeat*, ale nie przenosi żadnych danych.