Automatyka Pojazdowa

SAMOCHODOWY TESTER DIAGNOSTYCZNY

Andrzej Brodzicki
Aleksadner Pasiut
Michał Trojnarski
Mateusz Wąsala

Spis treści

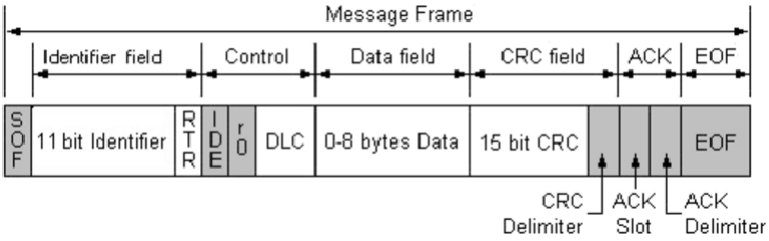
Cel projektu

Celem projektu było stworzenie aplikacji implementującej funkcjonalność samochodowego testera diagnostycznego. Aplikacja miała być napisana i przetestowana w środowisku CANoe, a następnie użyta do przeprowadzenia sesji diagnostycznej poprzez podłączenie do rzeczywistej sieci CAN przez konektor DSUB-9.

Opis procedury diagnostycznej

Diagnostyka przeprowadzana jest zgodnie z systemem diagnostyki pokładowej OBDII. Sposób wymiany informacji pomiędzy testowanym urządzeniem, a testerem diagnostycznym zdefiniowany jej przez protokół diagnotyczny (Keyword 2000 Protocol oparty na magistrali CAN i określony normą ISO-15765-3). Komunikacja między urządzeniami inicjowana jest przez tester diagnostyczny, który wysyła przez sieć komunikat z zapytaniem diagnostycznym. Zapytanie diagnostyczne składa się z informacji adresowej zawierającej adresy odbiorcy i nadawcy, identyfikatora oznaczającego wybraną funkcję oraz parametrów, które zależą od wybranej funkcji diagnostycznej. Odpowiedź sterownika transmitowana jest przez sieć do testera.

Zapytania diagnostyczne, jak i odpowiedzi, zrealizowane są w postaci ramek CAN. Zgodnie ze standardem 2.0A, ramka CAN składa się z 11-bitowego identyfikatora, 0-8 bajtów danych oraz innych pól bitowych. Identyfikator ramki definiował rodzaj informacji przenoszonej przez ramkę, czym rozróżniał informacje związane z sesją diagnostyczną (zapytania, odpowiedzi), od innych informacji przesyłanych po sieci CAN.



Rys. 1. Struktura ramki CAN w standardzie 2.0A

W przypadku zapytań, jak i odpowiedzi, dane zawsze miały rozmiar 64-bitów, czyli 8-bajtów. Zawarte w nich były parametry zależące od wybranej funkcji diagnostycznej (w przypadku zapytania), lub szczegóły odpowiedzi sterownika.

Realizacja funkcji (usług) diagnostycznych odbywa się w specjalnie określonym stanie pracy sterownika zwanym sesją diagnostyczną. Normalny stan pracy sterownika to wstrzymanie się od usług diagnostycznych. W praktyce uruchomienie sesji diagnostycznej realizowane jest poprzez cykliczne wysyłanie do sterowanika określonej ramki CAN. Zaprzestanie wysyłania tej wiadomości sprawia, że sterownik przełącza się z powrotem do trybu pracy zwykłej.

Kody błędu

Kody błędów, czyli tzw. DTC (ang. Diagnostic Trouble Code), zapisywane są w sterowniku jako 16-bitowe liczby, które w testerze diagnostycznym lub w dokumentacji przetwarzane są w 5-znakowe słowa alfanumeryczne. Każde takie słowo składa się z litery (P, C, B lub U) oraz 4 cyfr. Litera P w kodzie błędu oznacza, że usterka wystąpiła w układzie napędowym (ang. Powertrain), litera C będzie wskazywać na podwozie (ang. Chassis), B – nadwozie (ang. Body), U – system komunikacyjny (ang. Network). Pierwsza cyfra informuje czy kod błędu jest kodem standardowym czy też specyficznym dla danego producenta; druga cyfra z reguły określa podsystem, którego błąd bezpośrednio dotyczny; w dwóch pozostałych cyfrach umieszcza się bliższe informacje o występującej usterce. Kody błędów mogą być odczytane i skasowane za pomocą testera diagnostycznego. Jeżeli usterka zostanie usunięta, to odpowiadający jej kod błędu może zostać automatycznie skasowany z pamięci sterownika, jeżeli przez określony czas i w określonych warunkach ekspoatacyjnych usterka ponownie nie wystąpiła.

Litera	1. cyfra	2. cyfra	3. cyfra	4. cyfra

Rys. 2. Struktura kodu błędu

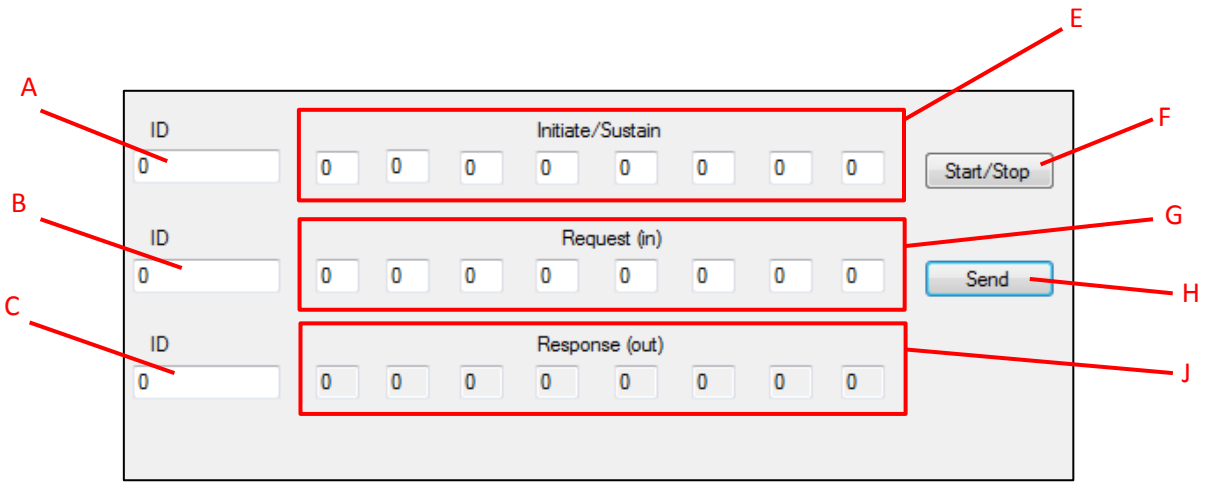
Aplikacja samochodowego testera diagnostycznego

Funkcjonalności aplikacji

Zaprojektowana aplikacja samochodowego testera diagnostycznego realizuje następujące funkcje:

- cyklicze wysyłanie wiadomości o identyfikatorze i danych zdefiniowanych przez użytkownika
- pojedyncze wysłanie wiadomości o identyfikatorze i danych zdefiniowanych przez użytkownika
- odbiór i wyświetlanie danych wiadomości o identyfikatorze określonym przez użytkownika

Interfejs graficzny



Rys.3. Interfejs graficzny aplikacji

Elementy interfejstu graficznego:

- trzy pola tekstowe heksadecymalne używane do określenia identyfikatorów:
  - wiadomości wysyłanej cyklicznie (A)
  - wiadomości wysyłanej jednorazowo (B)
  - wiadomości odbieranej (C)
- 24 heksadecymalne pola tekstowe, pogrupowane po osiem używane do:
  - określenia danych wiadomości wysyłanej cyklicznie (E)
  - określenia danych wiadomości wysyłanej jednorazowo (G)
  - wyświetlenia danych odbieranej wiadomości (J)
- przełącznik monostabilny włączający i wyłączający cykliczne wysyłanie wiadomości (F)
- przełącznik monostabilny wysyłający wiadomość jednorazową (H)

Testy aplikacji

Jeszcze w fazie projektowania aplikacja została przetestowana przy użyciu demonstracyjnej wersji programu CANoe. Oprócz kodu realizującego działanie aplikacji, napisany został kod symulujący działanie testowanego urządzenia. Wszystkie testy przebiegły poprawnie.

Testy na rzeczywistej sieci CAN zostały przeprowadzone 30.06.2016 r. w laboratorium Automatyki Pojazdowej. Komputer PC wyposażony w pełną wersję oprogramowania CAN został podłączony do modelu samochodu. Następnie uruchomiona została aplikacja i wysłane zostały określone żądania diagnostyczne. Tester poprawnie wysłał wiadomości i odebrał odpowiedzi.

	identyfikator	dane							
żadanie	0x252	05	A9	80	90	00	37	00	00
odpowiedź	0x552	80	90	00	37	00	00	00	00
żadanie	0x252	05	A9	80	90	00	34	00	00
odpowiedź	0x552	80	90	00	34	04	00	00	00
żadanie	0x252	05	A9	80	90	00	35	00	00
odpowiedź	0x552	80	90	00	35	00	00	00	00
żadanie	0x252	03	AA	01	0A	00	00	00	00
odpowiedź	0x552	01	26	01	0C	00	00	00	00
żadanie	0x252	02	1A	44	00	00	00	00	00
odpowiedź	0x652	03	7F	1A	31	00	00	00	00
żadanie	0x252	03	AA	01	11	00	00	00	00
odpowiedź	0x552	11	00	3F	F0	32	E0	00	5B

Autorzy projektu:

- Andrzej Brodzicki – interfejs graficzny, testy
- Aleksander Pasiut – kod aplikacji, symulator obiektu testowanego, testy, dokumentacja
- Michał Trojnarski – interfejs graficzny, testy
- Mateusz Wąsala – kod aplikacji, testy