Програмно осигуряване на микрокомпютърен уред за диагностика на автомобилни двигатели

Георги Кръстев

Programming of microcomputer device for diagnostics of autocar engines*: The programming of a microcomputer device for diagnostics of autocar engines has been described in the paper. The structure of the messages and the general outline of the transmitted data have been described in detail. The two-way connection with the board controller has been organized via interface K-Line and a protocol of information exchange Keyword 2000. The devise operates in the following three regimes: a regime for examining the inner parameters of the electronic management block (EMB); a regime for examining the codes of failures in EMB; a regime for managing the operational mechanisms of EMB.*

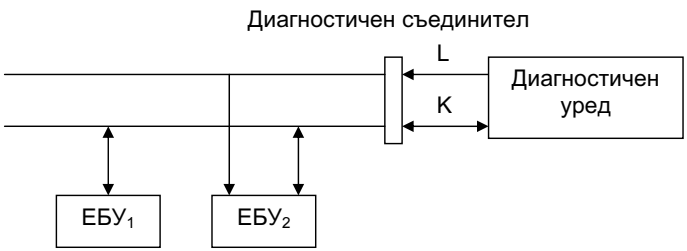
Key words*: Internal Combustion Engines, Road Vehicles, Diagnostic Systems, Single Chip Microcontrollers.*

ВЪВЕДЕНИЕ

Микропроцесорните системи за управление и диагностика стават съставна и неразривна част от съвременните автомобили. Те постепенно увеличават своята комплексност и без които е немислимо по нататъшното развитие на автомобилната промишленост. Наблюдава се тенденция с помощта на електрониката да се оптимизират експлоатационните качества на автомобила като цяло.

В настоящата работа се предлага програмното осигуряване и по-точно комуникационният протокол между микрокомпютърен уред за диагностика на автомобилни двигатели и електронни блокове за управление на автомобилни двигатели разработен в Русенския университет [2].

Обменът на съобщения между електронните блокове за управление (ЕБУ) и диагностичния уред се извършва по интерфейс ***K***-линия и по протокол Keyword 2000 [1, 3, 4]. Физическата реализация на канала за предаване на данни е показана на фиг. 1.



Фиг. 1. Физическа реализация на канала за предаване на данни.

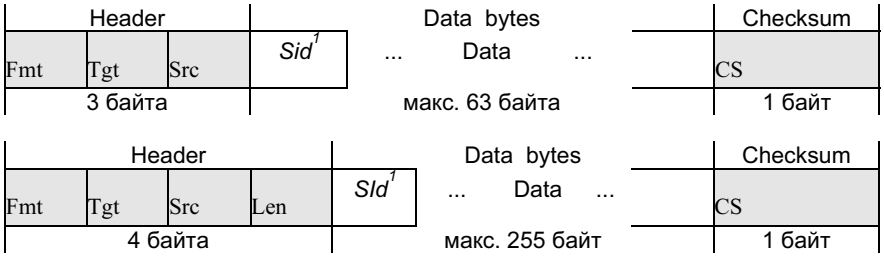
K-линията се използва за инициализация и обмен на диагностичните съобщения между различните блокове за управление и диагностичния уред. В дадената разработка ***L***-линията не се използва.

1. СТРУКТУРА НА СЪОБЩЕНИЯТА

Структурата на съобщенията се състои от три части:

- Заглавие (Header)
- Байтове с данни (Data bytes)
- Контролна сума (Checksum)

Общият вид е показан на фиг. 2.



Фиг. 2. Структура на съобщенията

Където: ***Fmt*** – байт определящ формата на съобщението; ***Tgt*** – байт определящ адреса на приемника на съобщението; ***Src*** - байт определящ адреса на източника на съобщението; ***Len*** – байт определящ дължината на съобщението при 4-и байтово заглавие; ***Sld*** – байт определящ типа на предаваните данни, който е част от байтовете с данни; ***CS*** – байт с контролната сума.

Заглавието може да съдържа три или четири байта. Байта с формата на съобщението (***Fmt***) съдържа информация за типа на съобщението и е показан на табл. 1.

Таблица1

Байт с формата на съобщенията (***Fmt***)

7	6	5	4	3	2	1	0	Бит
A1	A0	L5	L4	L3	L2	L1	L0	Условно означение

Битове A0, A1 определят типа на заглавието, което се използва в съобщенията. За реализиране на комуникационен канал между ЕБУ и тестера, трябва в тези битове да имаме комбинация “01”, при което ще имаме режим на физическа адресация. Битове L5 до L0 определят дължината на съобщението от началото на полето с данни до байта с контролната сума (не се включва), като се включи байта с типа на предаваните данни (***Sld***).

При данновата реализация на този протокол се използва физическа адресация. Така че под адрес на приемника на съобщението се подразбира физическия адрес на блока за управление, за който е предназначено конкретното съобщение. Байта с адреса на приемника винаги се използва заедно с байта с адреса на източника. Физическият адрес на ЕБУ е приет да бъде 10h, а адреса на тестера - F1h.

Полето с данни може да съдържа различен брой байтове с информация. Първият байт от полето с данни е винаги байта с типа предавани данни, който характеризира предназначението на предаваната информация.

Байта с контролната сума се намира в края на пакета на съобщението и представлява проста 8-м битова сума от всички байтове в съобщението без контролната сума.

2. ОБЩ ВИД НА ПОТОКА НА ПРЕДАВАНИТЕ ДАННИ

Съобщения с положителен/ отрицателен отговор са описани в табл. 2.

Където: ***Идентификатор*** – първият байт от полето с данни, който определя типа на предаваните данни; ***КО*** – код на отговора, определящ следващото действие в случай на отрицателен отговор

Съобщения с отрицателен отговор и код на отговора “зает-повтори запитването” са описани в табл. 3.

Таблица 2

Съобщения с положителен/отрицателен отговор

Диагностичен уред	ЕБУ
<Идентификатор> Запитване [...]	<Идентификатор> Положителен отговор [...]
<Идентификатор> Запитване [...] <Идентификатор> Запитване [...]	<Идентификатор> Отрицателен отговор [КО] <Идентификатор> Положителен отговор [...]

Таблица 3

Съобщения с отрицателен отговор и код на отговора “зает-повтори запитването”

Диагностичен уред	ЕБУ
<Идентификатор> Запитване [...]	<Идентификатор> Отрицателен отговор [зает-повтори запитването]
<Идентификатор> Запитване [...] <Идентификатор> Запитване [...]	<Идентификатор> Отрицателен отговор [зает-повтори запитването] <Идентификатор> Положителен отговор [...]

Съобщения с отрицателен отговор и код на отговора “запитването получено-задръжка на отговора” са показани в табл. 4.

Таблица 4

Съобщения с отрицателен отговор и код на отговора “запитването получено-задръжка на отговора”

Диагностичен уред	ЕБУ
<Идентификатор>Запитване [...]	<Идентификатор> Отр. отг.[зап. получено-задръжка на отговора] <Идентификатор> Положителен отговор [...]

В табл. 5 са показани имената на различните идентификатори при обмен на съобщения между ЕБУ и диагностичния тестер. Кодът на положителния отговор се формира от съответстващия код на запитването с установяване на 6-я бит в “1” Идентификатора на отрицателния отговор винаги е 7FH. В табл. 6 са описани стойностите на кодовете на отговорите.

За реализиране на програмното осигуряване е използван пакета ICC 8051. Той представлява съвкупност от крос-средства, работещи под управлението на MS-DOS на персонални компютри и включва асемблер A8051, C-компилатора ICC 8051, свързващия редактор XLINK, библиотеката XLIB и симулатора SIM51.

3. ПРИМЕР

С дадената процедура на C се четат от електронния блок за управление кодовете на неизправности и техния статус.

```
/*-----------------------------------------------------------------------------------------------------------*/
/* Процедура за четене от ЕБУ на кодовете на неизправности и техния статус.      */
/* Кодовете на неизправностите се разполагат в приемния буфер на UART,           */
/* с изместване 1 (Message[1] и т.н.).                                           */
/* Връща: Резултата от изпълнената функция.                                     */
/*-----------------------------------------------------------------------------------------------------------*/
enum KEYWORD_RESULT Read_Diagnostic_Trouble_Codes(void)
{
  /* Формиране и изпращане на съобщение readDiagnosticTroubleCodesByStatus    */
  Message[0]=0x18; Message[1]=0x00;
```

```
  Message[2]=0x00; Message[3]=0x00;
  Start_UART_Tx(4); while(is_UART_Tx_Busy()) IDLE();
  /* Очакване изтичането на 100 мсек или приемане на положителния отговор        */
  Start_Time_Out(100/TIMER_TICK);     // Пускане на тайм-аут очакване на отговора
  for(;;)
    {
     Start_UART_Rx();        // Разрешаване на приемането на съобщения от ЕБУ
     while((is_Timer_Count()) && (is_UART_Rx_Busy())) IDLE();
     if(is_Timer_Stopt())
       {
        // Таймаут очакването на положителния отговор е изтекло
        // Излизане от подпрограмата с установяване флаг за грешка
        return ERROR;
       }
     if((is_UART_Rx_Ready()) && (Message[0] == 0x58))
       {
        /* Получен е положителен отговор на readDiagnosticTroubleCodes                */
         Stop_Time_Out;      // Спиране брояча на тайм-аут.
        return OK;
         }
    }
}
```

Таблица 5

Идентификатори

Наименование на идентификатора	Съкращение	Значение на кода(Hex)	
		Запитване	Отговор
StartCommunication	STC	81	C1
StopCommunication	SPC	82	C2
StartDiagnosticSession	STDS	10	50
StopDiagnosticSession	SPDS	20	60
EcuReset	ER	11	51
ClearDiagnosticInformation	CDI	14	54
ReadDiagnosticTroubleCodesByStatus	RDTCBS	18	58
ReadEcuIdentification	REI	1A	5A
ReadDataByLocalIdentifier	RDBLI	21	61
ReadMemoryByAddress	RMBA	23	63
InputOutputControlByLocalIdentifier	IOCBLI	30	70
WriteDataByLocalIdentifier	WDBLI	3B	7B
TesterPresent	TP	3E	7E

ЗАКЛЮЧЕНИЕ

В доклада е описано разработеното програмно осигуряване на микрокомпютърен уред за диагностика на автомобилни двигатели. Дадена е структурата на съобщенията, общият вид на предаваните данни, както и различните типове идентификатори и кодовете на отговори според съществуващите стандарти. Предложена е

и примерна процедура на С за четене от електронния блок за управление на кодовете на неизправности и техния статус.

Таблица 6

Описание на кодовете на отговорите

Hex	Код на отговора	Съкращение
10	**GeneralReject** Запитването отхвърлено, но без указване на причина.	**GR**
11	**ServiceNotSupported** Запитването не може да бъде изпълнено, защото приемника не поддържа такъв вид данни .	**SNS**
12	**subFunctionNotSupported-invalidFormat** Изискваното действие не може да бъде изпълнено, защото приемника не поддържа аргумента на съобщението или формата на байта с аргумента не съответства.	**SFNS-IF**
21	**busy-RepeatRequest** Приемника е временно зает. В тази ситуация повторното запитване ще се изпълни със запълване на цялото поля с данни. Когато приемника може да завърши изпълнението на изискваното действие, той изпраща положителен отговор.	**B-RR**
31	**RequestOutOfRange** Изискваното действие не може да се изпълни, защото приемника определя байт с данни, излизащ от допустимия диапазон.	**ROOR**
72	**TransferAborted** Предаването на данните е прекъснато по неизвестна причина и не може да бъде завършено по-късно.	**TA**
77	**BlockTransferDataChecksumError** Контролната сума на данните на приетото съобщение не съответства на изпратеното.	**BTDCE**

ЛИТЕРАТУРА

[1] Белев Ст., Йоана Р., Модул за логическо управление, реализиран на база SPLD, Девета национална научно-приложна конференция „Електронна техника ЕТ `2000, сборник доклади том IV, Електронни системи за измерване и управление, стр. 157 – 162.

[2] Кръстев, Г., Б. Христов. Микрокомпютърен уред за диагностика на автомобилни двигатели. Известия на съюза на учените - Русе, Серия "Технически науки" №1, с. 43-46, 2009.

[3] ISO 14230-1 Road Vehicles - Diagnostic Systems. Keyword Protocol 2000 Part 1: Physical Layer

[4] ISO 14230-2 Road Vehicles - Diagnostic Systems. Keyword Protocol 2000 Part 2: Data Link Layer

За контакти:

Доц д-р Георги Кръстев, Катедра “Компютърни системи и технологии”, Русенски университет “Ангел Кънчев”, Тел.: 082 888 672, E-mail: gkrastev@ecs.ru.acad.bg.

Докладът е рецензиран.