**Данные, передаваемые с автомобиля на базовую станцию**

1. VIN Код (он же идентификатор транспортного средства) – обязательный пункт, только так мы можем определить, с какого автомобиля данные приходят. Этот пункт может быть не доступен для чтения из ЭБУ двигателя. Тогда нужно предусмотреть зашивание VIN в EEPROM МК и возможность его через приложение зашивать. Если можно считать из автомобиля, то при первой установки VIN считывается и записывается в EEPROM и отключается возможность изменения. (сделано для того, чтобы 1 устройство было привязано к одному автомобилю, иначе это экономически не выгодно).

**Формат данных:**

* 17 Символов в заглавном формате(0-9A-Z)

Мысли об уменьшении размера для передачи (т.к чем меньше мы бит информации будет передавать, тем большее расстояние мы сможем покрыть и работать это будет быстрее)

Т.к 1 символ кодируется 1 байтом (в лучшем случае), а это 8 бит, чем мы можем просто дико загрузить весь payload. Нам требуются только числа 0-9 и буквы латинского алфавита, и то не все.

***Цитата:*** *За исключением последних четырех позиций VIN имеет алфавитно-цифровую структуру. Для его составления разрешается использовать следующие арабские цифры и латинские буквы:*

*0 1 2 3 4 5 6 7 8 9* = 10 символов  
*A B C D E F G H J K L M N P R S T U V W X Y Z* = 23 символа

То есть для кодировки VIN нам требуется 33 символа. Как можно заметить, что символов O, I, Q – нет. Они нам пока не пригодятся, но можно их внести заведомо, мало ли… В итоге получаем:

*0 1 2 3 4 5 6 7 8 9* = 10 символов  
*A B C D E F G H I J K L M N O P Q R S T U V W X Y Z* = 26 символов

В итоге наш алфавит составляет 36 символов. Один символ в итоге кодируется следующим кол-вом битов:

То есть для кодирования нашего символа можно использовать всего ***6 бит!***

Плюс необходима вспомогательная информация для определения что именно это за строка – вин код, код ошибки, либо еще какая-либо информация. У нас осталось еще 64-36=28 не закодированных символов. В принципе их будет достаточно для того, что бы иметь представление о 28 типах информации.

*В итоге всего 6 бит будет занимать 1 символ!*

**Сравнительная таблица (***VIN - 4USBT53544LT26841***)**

*UTF-8 (8 bit)*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Опр. VIN** | **4** | **U** | **S** | **B** | **T** | **5** | **3** | **5** | **4** | **4** | **L** | **T** | **2** | **6** | **8** | **4** | **1** |
| **БИТЫ** | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |

В итоге для кодировки VIN кода в UTF-8 потребуется **18\*8=144 бит** или **18 байт!**

*Наша кодировка (6 bit)*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Опр. VIN** | **4** | **U** | **S** | **B** | **T** | **5** | **3** | **5** | **4** | **4** | **L** | **T** | **2** | **6** | **8** | **4** | **1** |
| **БИТЫ** | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 |

В итоге для кодировки VIN кода в нашей кодировке потребуется **18\*6=108 бит** или **14 байт!**

Выигрыш получается в **36 бит**! Или минимум в **6 символов** в нашей кодировке!

*Таблица кодировки*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *Символ* | *Код символа* | *Символ* | *Код символа* | *Символ* | *Код символа* | *Символ* | *Код символа* |
| *0* | *000000* | *G* | *010000* | *W* | *100000* | *?* | *110000* |
| *1* | *000001* | *H* | *010001* | *X* | *100001* | *?* | *110001* |
| *2* | *000010* | *I* | *010010* | *Y* | *100010* | *?* | *110010* |
| *3* | *000011* | *J* | *010011* | *Z* | *100011* | *?* | *110011* |
| *4* | *000100* | *K* | *010100* | *vin* | *100100* | *?* | *110100* |
| *5* | *000101* | *L* | *010101* | *dtcs* | *100101* | *?* | *110101* |
| *6* | *000110* | *M* | *010110* | *lat.* | *100110* | *?* | *110110* |
| *7* | *000111* | *N* | *010111* | *long.* | *100111* | *?* | *110111* |
| *8* | *001000* | *O* | *011000* | *?* | *101000* | *?* | *111000* |
| *9* | *001001* | *P* | *011001* | *?* | *101001* | *?* | *111001* |
| *A* | *001010* | *Q* | *011010* | *?* | *101010* | *?* | *111010* |
| *B* | *001011* | *R* | *011011* | *?* | *101011* | *?* | *111011* |
| *C* | *001100* | *S* | *011100* | *?* | *101100* | *?* | *111100* |
| *D* | *001101* | *T* | *011101* | *?* | *101101* | *?* | *111101* |
| *E* | *001110* | *U* | *011110* | *?* | *101110* | *?* | *111110* |
| *F* | *001111* | *V* | *011111* | *?* | *101111* | *?* | *111111* |

*\*где стоит знак вопроса – возможность добавлять собственные функции определения.*

***Структура cобщения:***

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| vin | W | A | U | Z | ….. | 0 | 9 | 6 |

Само сообщение, где вес 1 символа 6 бит

Код определения

типа сообщения

1. Коды ошибок, получаемые из автомобиля.

Каждый из OBD-II кодов неисправностей, состоит из пяти символов. Буквы и четырёх цифр.

Нумерация ошибок OBD-II:

* P00xx — Контроль системы смесеобразования и системы доп. снижения токсичности выхлопа.
* P01xx — Контроль системы смесеобразования.
* P02xx — Контроль системы смесеобразования.
* P03xx — Система зажигания и система контроля пропусков воспламенения.
* P04xx — Вспомогательные системы контроля эмиссии.
* P05xx — Контроль скорости автомобиля, системы холостого хода и других систем.
* P06xx — Блоки управления ECM / PCM / TCM и другие системы
* P07xx — Трансмиссия.
* P08xx — Трансмиссия.
* P09xx — Трансмиссия.
* P10xx — Коды устанавливаемые производителем. Зависят от марки авто.
* P20xx — Коды устанавливаемые производителем. Зависят от марки авто.
* B00xx — Кузов (подушки безопасности, центральный замок, электростекло-подъемники).
* C00xx — Шасси (ABS противопробуксовочная система, ESP, TCS-Traction Control System Система курсовой устойчивости).
* U10xx — Межблочная шина обмена данных (CAN-bus) (CAN-II).
* U25xx — Межблочная шина обмена данных (CAN-bus) (CAN-II).

Символы «xx» ссылаются на отдельные неисправности внутри каждой подсистемы.

Как можно видеть, в данном случае используется всего 4 символа и 10 цифр, то есть 14 символов, что кодируется **4 битами**. Из этого можно сделать вывод, что 6-ти битовая кодировка избыточна для этой цели. Но можно заметить, что символы P, B, C, U, 0-9 можно в принципе закодировать *5 битами*, если убрать не значащий 0 в 6 бите. Что уже хорошо, можно программно разбирать строку, а для передачи убирать лишние не значащие 5 бит(на 1 код ошибки, так как код ошибки состоит из 5 знаков), что позволит немного сэкономить место в payload.

Второй вариант.

6-ти битная кодировка все равно остается за базу, так как кроме таких символов, которые представлены в кодировке автомобиль выдать не может. На основании того, что микроконтроллеры и в автомобиле, и на базовой станции достаточно ресурсные, мы можем со спокойной совестью делать все преобразования, даже более-менее ресурсоемкие. Идея заключается в том, что любое сообщение, принятое от одного из устройств, начинает обрабатываться в 6-битной кодировке, первый обработанный символ является типом сообщения. Далее в зависимости от типа сообщения выбирается кодировка присущая тому или иному типу для декодирования. Но так как в сообщении может быть и несколько типов данных и с разными кодировками, нужно как-то ограничить и отделить одни данные от других. Для этого можно ввести длину строки, так как фиксировать длину payload в целом или передавать только одним типом не целесообразно. Получается следующий тип передаваемого сообщения.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| type | length | message | …… | type | length | message |

Само сообщение N симв.

Длина сообщения 8 бит

Тип сообщения6 бит

Проведем сравнение, если использовать только 6-битную кодировку и, если использовать метод комбинированных кодировок.

Тестовые данные: *VIN - 4USBT53544LT26841, DTCS – P0001,P0002*

1. *6-битная кодировка*

|  |  |  |  |
| --- | --- | --- | --- |
| *VIN* | *4USBT53544LT26841* | *DTCS* | *P0001,P0002* |
| *6 бит* | *17\*6=102 бита* | *6 бит* | *10\*6=60 бит* |

*Итого:* 174 бита, примерно 22 байта для передачи.

1. *Комбинированная кодировка*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *VIN* | *17* | *4USBT53544LT26841* | *DTCS* | *10* | *P0001,P0002* |
| *6 бит* | *8 бит* | *17\*6=102 бита* | *6 бит* | *8 бит* | *10\*4=40 бит* |

*Итого: 174 бита, примерно те же самые 22 байта для передачи, но этот способ покажет себя с выгодной стороны при бОльших объёмах данных.*

*Кодировка ошибок двигателя*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *Символ* | *Код символа* | *Символ* | *Код символа* | *Символ* | *Код символа* | *Символ* | *Код символа* |
| *0* | *0000* | *4* | *0100* | *8* | *1000* | *C* | *1100* |
| *1* | *0001* | *5* | *0101* | *9* | *1001* | *U* | *1101* |
| *2* | *0010* | *6* | *0110* | *P* | *1010* | *-* | *1110* |
| *3* | *0011* | *7* | *0111* | *B* | *1011* | *-* | *1111* |
| *\*для метода комбинированной кодировки* | | | | | | | |