

**Министерство науки и высшего образования Российской  
Федерации Федеральное государственное автономное образовательное учреждение высшего образования  
«Южно-Уральский государственный университет (национальный исследовательский университет)»  
Высшая школа электроники и компьютерных наук  
Кафедра «Электронные вычислительные машины»**

**ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС СБОРА ДАННЫХ  
ТЕХНИЧЕСКОГО СОСТОЯНИЯ ТРАНСПОРТНОГО СРЕДСТВА**

Попов В.А. КЭ-406

# Актуальность

Современный автомобиль является сложным высокотехнологическим устройством, внутри которого между собой взаимодействуют высокоточные механические и электронные системы. Как и в любой системе, в автомобиле могут возникнуть неисправности.

Компьютерная диагностика помогает ускорить время на определение неисправности и также предотвратить серьезные последствия при незначительных поломках при помощи предиктивной диагностики.



# Актуальность



Вовремя незамеченная или проигнорированная поломка может вызвать другие, более серьезные неисправности в системе.

Также поломка может быть незаметной для водителя, но так или иначе представлять угрозу безопасности. Так 14% от всех ДТП происходят по техническим причинам.

# Цели

Цель разрабатываемого комплекса заключается в создании универсального программного обеспечения для сбора данных технического состояния автомобиля при помощи специального диагностического устройства и отправки полученных данных на сервер, интегрируемый в систему предприятия

# Задачи

- рассмотреть существующие на рынке аналоги;
- провести детальный анализ найденных решений, их особенности, минусы и плюсы;
- определить основной функционал программно-аппаратного комплекса;
- выполнить программную реализацию комплекса;
- определить методы тестирования разработанного программно-аппаратного комплекса.

# Обзор аналогов

На отечественном рынке получили распространение данные системы:

- «Geostron»
- «Cartrek»
- «TrackGPS»
- «Монтранс»

# Обзор аналогов

«Geostron»



Teltonika FMB 003



Teltonika FMB 010

# Обзор аналогов

«TrackGPS»



**ГЛОНАСС/GPS Track OBD**

**Снят с производства**



**Контроллер CAN-шины**

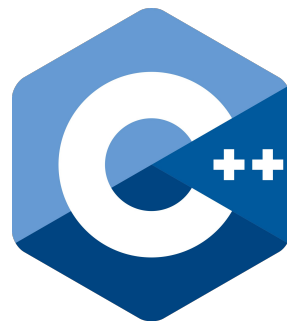


# Обзор технологических решений

Сервер:

Язык программирования C++

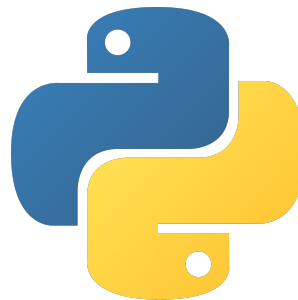
Библиотеки Winsock2, Thread



Терминал:

Язык программирования Python

Библиотеки Socket, Threading, Serial, Obd



# Обзор технологических решений

## Диагностические адаптеры ELM-327

<b>Название</b>	<b>Цена</b>
PIC18F25K80 PL2303	от 1700 р
PIC18F2480 FT232RI	от 1200 р
CH340	от 600 р

# Обзор технологических решений

## Одноплаточные решения

Название	Характеристики	Цена
Orange Pi Lite	Cortex-A7, DDR3 1GB	от 1900 р
Banana Pi M2	Cortex-A7 V40, DDR3 1GB	от 3200 р
PINE H64	Cortex A53, DDR3 2GB	от 3300 р
NanoPi NEO2	Cortex A53, DDR3 1GB	от 3800 р
Khadas VIM2	Cortex-A7 V40, DDR4 3GB	от 6200 р

# Требования к системе

## Сервер:

- операционная система: Windows 10 (1809 и выше);
- процессор: Intel Core i5 4470, LGA1150;
- место на диске: от 5Gb.

## Терминал:

- операционная система: Linux (Kernel 4 и выше);
- место на диске: от 1Gb;
- диагностический адаптер: ELM327-USB PL2303;

# Требования к системе

## Автомобиль:

- наличие и исправность OBD-II разъема;
- год выпуска для Американских автомобилей: от 1998.
- год выпуска для Европейских автомобилей: от 2000.
- год выпуска для Азиатских автомобилей: от 2002.
- год выпуска для Российских автомобилей: от 2004.

# Функциональные требования

## Сервер:

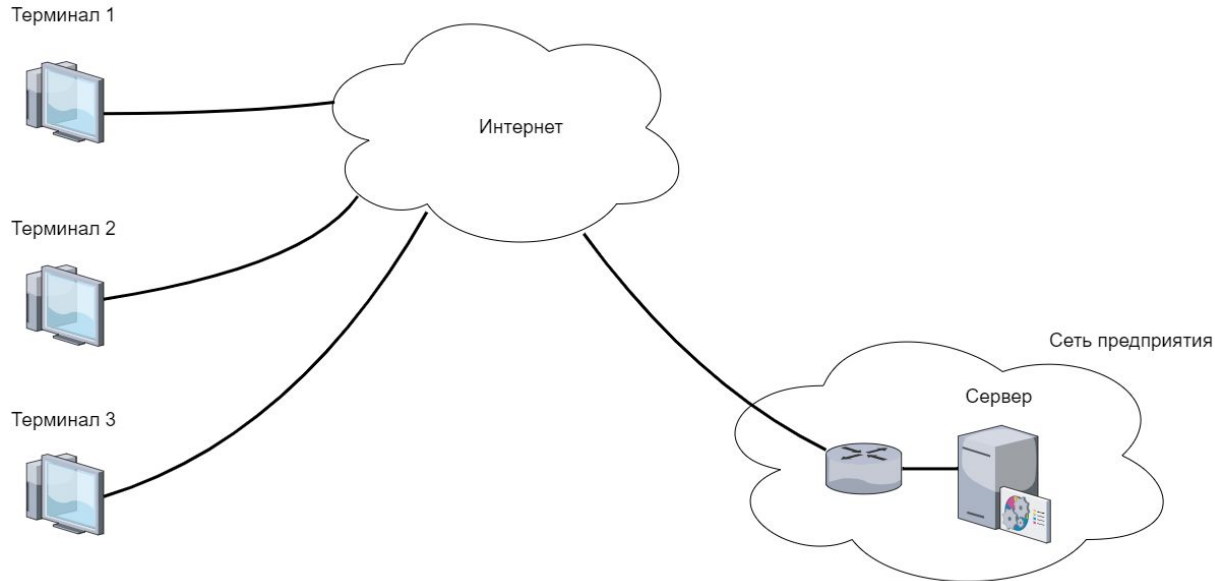
- получение технических данных с терминала;
- обработка полученных данных;
- составление отчета;
- отправка данных на локальный сокет для работы со сторонними приложениями и системами предприятия.

## Терминал:

- чтение кодов неисправности при помощи контроллера ELM327;
- расшифровка полученных технических данных состояния автомобиля;
- отправка полученных технических данных состояния автомобиля на сервер;

# Проектирование

Терминал, сервер и ПО предприятия должны находиться в одной частной виртуальной локальной сети (VPN).



## Описание данных

<b>Название</b>	<b>Описание</b>	<b>Группа команды</b>	<b>Код команды</b>
CURRENT_DTC	Get DTCs from the current/last driving cycle	PIDs [01-20] (1)	07
DTC	Get DTCs	PIDs [01-20] (1)	03
COOLANT_TEMP	Engine Coolant Temperature	PIDs [01-20] (1)	0105
FUEL_PRESSURE	Fuel Pressure	PIDs [01-20] (1)	010A
FUEL_LEVEL	Fuel Level Input	PIDs [21-40] (2)	012F
RPM	Revolutions per minute	PIDs [01-20] (1)	010C
SPEED	Speed	PIDs [01-20] (1)	010B
INTAKE_PRESSURE	Intake Manifold Pressure	PIDs [01-20] (1)	010B
INTAKE_TEMP	Intake Air Temp	PIDs [01-20] (1)	010F
RUN_TIME	Engine Run Time	PIDs [01-20] (1)	011F



# Реализация



Диагностический адаптер ELM-327  
PIC18F25K80 PL2303



Микрокомпьютер Raspberry Pi 3 MB+

# Реализация

Для отправки данных от терминала используется шаблон команд:  
`$TYPE#VALUE.`

Для того чтобы сторонней программе получить данные, необходимо отправить команду на терминал следующего шаблона:  
`&CODE#TYPE`, где CODE - индивидуальный номер ТС.

## Переменные, которые терминал передает на сервер

Название	Описание
DTC	Коды ошибок
COOLANT_TEMP	Температура охлаждающей жидкости
FUEL_PRESSURE	Давление в топливной рампе
FUEL_LEVEL	Уровень топлива
RPM	Обороты в минуту
SPEED	Скорость
INTAKE_PRES	Давление на впуске
INTAKE_TEMP	Температура на впуске
RUN_TIME	Время работы двигателя

# Реализация

Команда	Описание
/size	Получить количество подключений
/id	Получить информацию по внутреннему индексу
/value	Получить значение
/all	Получить все подключения
/vall	Получить все значения
/send	Отправка

Команды, используемые на сервере

# Тестирование

Подключение терминала к ТС.

Используемое ТС:

Chery Bonus A13 2011г.



# Тестирование



OBD-II разъем на тестируемом автомобиле.

# Тестирование

## Сервер

```
C:\> Выбрать C:\Users\sevaw\source\repos\grid_srv\Debug\grid_srv.exe
Server started.
Type client-connection port. If you want to use stock port (900) enter 0
0

/all
Terminal ID - 1. Array position - 0 0
/vall
Terminal ID - 1 1001 Coolant temp 89 DTC - [] RPM - 8.2 Speed - 0 Fuel level - None Fuel pressure - None
Intake pressure - 0.1 Intake temp - 29 Runtime - None Car code - 1001
/vall
Terminal ID - 1 1001 Coolant temp 86 DTC - [] RPM - 7.9 Speed - 0 Fuel level - None Fuel pressure - None
Intake pressure - 0.1 Intake temp - 30 Runtime - None Car code - 1001
```

# Тестирование

## Тестирование обработчика ошибок

```
/all
Terminal ID - 1. Array position - 0 0
/vall
Terminal ID - 1 1001 Coolant temp 89 DTC - [] RPM - 8.2 Speed - 0 Fuel level - None Fuel pressure - None
Intake pressure - 0.1 Intake temp - 29 Runtime - None Car code - 1001
/vall
Terminal ID - 1 1001 Coolant temp 86 DTC - [] RPM - 7.9 Speed - 0 Fuel level - None Fuel pressure - None
Intake pressure - 0.1 Intake temp - 30 Runtime - None Car code - 1001
/vall
Terminal ID - 1 1001 Coolant temp 87 DTC - P0100 Mass or Volume Air Flow Circuit RPM - 8.3 Speed - 0
Fuel level - None Fuel pressure - None Intake pressure - 0.1 Intake temp - 30 Runtime - None Car code -
1001
```

# Тестирование

## Получение данных с сервера клиентом

```
Выбрать C:\Users\sewaw\source\repos\WS_CLN\Debug\WS_CLN.exe
900
Connected.
Wellcome to the GRID; Type - Client; Connected
&DTC#1001
Error. No command 1001
&1001#DTC
P0100 Mass or Volume Air Flow Circuit
&1001#RPM
8.1
```



# Заключение

В ходе выполнения выпускной квалификационной работы был разработан программно-аппаратный комплекс сбора технических данных транспортного средства, сделан обзор аналогов и технологических решений.

**Спасибо за внимание!**