

Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Политехнический институт

Кафедра «Конструкторско-технологического обеспечения
машиностроительных производств»

УТВЕРЖДАЮ

Заведующий кафедрой

_____ М. П. Головин

подпись

«_____» _____ 2018 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.01 – Информатика и вычислительная техника

Разработка программного обеспечения для устройства, предупреждающего о
столкновении

Научный руководитель _____

подпись, дата

ст.преподаватель Р.С. Лукин

Выпускник _____

подпись, дата

Э. Г. Агамамедов

Нормоконтролер _____

подпись, дата

ст.преподаватель Р.С. Лукин

Красноярск 2018

СОДЕРЖАНИЕ

| | |
|--|----|
| ВВЕДЕНИЕ..... | 4 |
| 1. Обзор..... | 5 |
| 1.1. Заинтересованные стороны..... | 5 |
| 1.2. Требования..... | 6 |
| 1.3. Аналогии..... | 9 |
| 1.3.1. City Safety™ от Volvo..... | 9 |
| 1.3.2. Превентивная система безопасности PRE-SAFE® от компании Mercedes-Benz10 | |
| 1.3.3. Активный Круиз-Контроль с функцией STOP&GO от BMW. 11 | |
| 1.3.4. Система автоматического торможения Active City Stop (ACS) 13 | |
| 2. Аналитическая модель..... | 15 |
| 2.1. Модель определения безопасного расстояния..... | 15 |
| 2.1.1. Обзор алгоритма расчета безопасного расстояния..... | 15 |
| 2.1.2. Расчет относительной скорости по разностной схеме..... | 15 |
| 2.1.3. Расчет абсолютной скорости..... | 16 |
| 2.2. Коэффициенты трения..... | 16 |
| 2.2.1. Влияние качества дорожного покрытия на коэффициент сцепления 16 | |
| 2.2.2. Влияние температуры шины на коэффициент сцепления..... | 17 |
| 2.2.3. Влияние скорости движения автомобиля на коэффициент сцепления 18 | |
| 3. Алгоритм..... | 20 |
| 3.1. Исходный алгоритм..... | 20 |
| 3.2. Алгоритм с учетом относительной скорости..... | 21 |
| 3.3. Алгоритм с учетом поворота головы..... | 22 |
| 3.4. Определения вхождения препятствия в диапазон автомобиля 23 | |
| 3.5. Количество шагов сервопривода..... | 24 |
| 3.6. Результаты..... | 24 |
| 4. Тестирование..... | 25 |
| 4.1. Создание тестовых задач..... | 25 |
| 4.2. Автотестирование алгоритмов..... | 25 |
| 4.3. Тестирование собранного устройства..... | 26 |

| | | |
|------|---|----|
| 4.4. | Обработка результатов тестирования..... | 26 |
| | ВЫВОД | 28 |
| | СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ | 29 |
| | Приложение А (обязательное) Алгоритм основной части... .. | 31 |
| | Приложение Б (обязательное) Алгоритм проверки опасности... .. | 32 |
| | Приложение В (обязательное) Исходный код программы... .. | 33 |

ВВЕДЕНИЕ

В современном мире реальной проблемой является повышенное число аварий. Помимо аварий с летальным исходом, которые являются наиболее резонансными, существуют множество аварий с небольшими повреждениями, которые приводят к образованию заторов на дорогах, и как следствие к стрессовым ситуациям у водителей. Для предотвращения таких ситуаций необходимо устройство, которое будет предупреждать водителя о возможной аварии.

Цели

Реализация программной части устройства, которое будет предупреждать водителя о возможном столкновении.

Задачи

1. Изучить требования к сценарию работы устройства.
 2. Разработать математическую модель.
 3. Провести тесты погрешности разработанных программных модулей.
- Задание на преддипломную практику:
1. Разработать математическую модель
 2. Реализовать код для платформы Arduino на основе математической модели
 3. Провести тестирования.
 4. На основании теста внести изменения в код.

1. Обзор

1.1. Заинтересованные стороны

У данного проекта будут следующие стейкхолдеры со своими целями:

1. Программист проекта: в качестве программиста проекта будет выступать бакалавр. Программисты могут работать над отдельными компонентами системы в рамках своих бакалаврских работ.

a. Потребности: написание библиотек и программ, используемых системой, согласно техническому заданию или спецификации; публикация научных статей по теме проекта

b. Точка зрения: разработчик

c. Описание: программный код (язык Arduino); спецификация с требованиями к системе и компонентам

2. Аппаратный разработчик проекта: в качестве аппаратного разработчика проекта будет выступать бакалавр. Аппаратные разработчики могут работать над отдельными компонентами системы в рамках своих бакалаврских работ.

a. Потребности: создание аппаратного продукта, удовлетворяющего требованиям технического задания

b. Точка зрения: разработчик

c. Описание: аппаратная платформа из разных модулей; спецификация с требованиями к системе и компонентам

3. Тестировщик проекта: в качестве тестировщиков проекта будут выступать бакалавры как в рамках своих магистерских диссертаций и выпускных квалификационных работ, так и по личной инициативе.

a. Потребности: протестировать систему; обеспечить стабильность работы системы

b. Точка зрения: пользователь

c. Описание: проверка работы устройства; необходимость докладывать разработчикам проекта об ошибках

4. Научный руководитель: может помочь с проектом, принять участие в проекте, а также выделить материальные ресурсы для реализации проекта.

a. Потребности: повышение рейтинга университета за счёт опубликованных в рамках проекта научных статей, полученных грантов

b. Точка зрения: администратор

c. Описание: научные публикации о системе

5. Водитель: в качестве водителей в данном проекте будут выступать начинающие водители по личной инициативе.

a. Потребности: получить систему предупреждения столкновений

b. Точка зрения: пользователь

c. Описание: использование готового устройства

6. Конкуренент: в качестве конкурентов в данном проекте будут выступать производители аналогов.

a. Потребности: сделать систему, которая будет лучше вашей по большинству параметров

b. Точка зрения: разработчик

c. Описание: создание здоровой конкуренции

7. Регулирующие органы: в качестве регулирующих органов в данном проекте будут выступать сотрудники ГИБДД.

a. Потребности: получить систему, которая не будет являться модификацией транспортного средства

b. Точка зрения: регулирующий орган

c. Описание: проверка устройства на введение им модификаций в транспортное средство

1.2. Требования

Зная потребности стейкхолдеров можно составить список требований к устройству, чтобы эти потребности оказались удовлетворены, т.е. устройство могло бы считаться успешным. Для описания требований воспользуемся подходом известным как «пользовательские истории», которые сведём в единую таблицу 1.

Таблица 1 – Список требований

| Стейкхолдер (Я как..) | Действие (хочу, чтобы устройство..) | Цель (Для того, чтобы..) |
|--|---|---|
| Водитель | Работало в диапазоне температур от -40 до +50 °С | Пользоваться устройством по всей России |
| Водитель | Имело жесткое крепление к стеклу или приборной панели | Устройство не могло отсоединиться от присоединяемой поверхности и быть в зоне видимости |
| Программист проекта, Тестировщик проекта | На стадии разработки имело возможность записывать ошибки при надобности | Оперативно исправлять ошибки |
| Аппаратный разработчик проекта | Обеспечивало определенную дальность работы | Обеспечивать своевременное реагирование человека на сигналы о столкновении |
| Аппаратный разработчик проекта | Максимальная погрешность измерений не должна превышать 10см | Устройство правильно реагировало на изменение расстояния между препятствиями |

Продолжение таблицы 1

| | | |
|---|--|--|
| Программист проекта, Тестировщик проекта, Водитель | Не имело лишних срабатываний критически влияющих на ситуацию, а именно устройство не должно предоставлять недостоверные сведения о расстоянии до объекта | Не повлечь за собой ущерб |
| Водитель | Не отвлекало от дороги яркими вспышками или сильно громкими сигналами | Не отвлекаться от дороги |
| Тестировщик проекта, Водитель | Имело возможность калибровки | Для точной настройки датчика расстояния |
| Аппаратный разработчик проекта, Водитель | Могло быть установлено в любом транспортном средстве | Использовать устройство в любом транспортном средстве |
| Научный руководитель, Программист проекта, Аппаратный разработчик проекта | Было конкурентоспособным в своей сфере применения, имело цену не больше 10000р | Использовать его в дальнейшем для продажи или научных статей |
| Конкурент | Было неконкурентоспособным в своей сфере применения | Устройство конкурента было хуже по всем показателям |
| Регулирующие органы, Аппаратный разработчик проекта | Не вносило модификации в транспортное средство | Не пришлось регистрировать устройство |

1.3. Аналоги

1.3.1. City Safety™ от Volvo.

Эта функция помогает водителю избежать столкновения, например, при движении в пробках, когда неравномерное движение транспорта впереди и снижение внимания могут приводить к авариям. Оно активируется на скорости ниже 50 км/ч и помогает водителю путем автоматического торможения автомобиля при возникновении опасности столкновения с едущим впереди автомобилем, если водитель вовремя не реагирует и не приступил к торможению и/или выруливанию. Система подключается в ситуациях, когда водитель должен был бы начать торможение намного раньше, и именно поэтому не может помочь водителю во всех ситуациях. Функция разработана так, чтобы включаться на самом последнем этапе, избегая ненужного вмешательства. Пример устройства приведен на рисунке 1.

Система считывает дорожную ситуацию перед автомобилем с помощью лазерного датчика, установленного у верхнего края ветрового стекла. При угрозе столкновения функция City Safety™ включает автоматическое торможение автомобиля, которое может восприниматься, как резкое торможение.



Рисунок 1 - Приемно-выходное окно лазерного датчика

Если различие в скорости по отношению к автомобилю впереди составляет 4-15 км/ч, City Safety™ позволяет вообще не допустить столкновения.

City Safety™ включает краткое резкое торможение и обычно останавливает автомобиль точно позади автомобиля перед вами. Для большинства водителей это далеко не обычный стиль вождения, что может восприниматься, как не очень приятная ситуация.

Если различие в скорости между автомобилями составляет более 15 км/ч, City Safety™ не может сама предотвратить столкновение – для приложения полного тормозного усилия водитель должен выжать педаль тормоза, и тогда столкновение можно предотвратить даже при разнице скоростей более 15 км/ч..

Когда функция активируется и проводит торможение, в комбинированном приборе появляется сообщение о том, что функция активирована или была активирована.

1.3.2. Превентивная система безопасности PRE-SAFE® от компании Mercedes-Benz

Интеллектуальное раннее распознавание аварии осуществляется системой превентивной безопасности PRE-SAFE® посредством имеющихся датчиков - к примеру, датчиков электронной системы стабилизации ESP® либо системы превентивного экстренного торможения BAS PRO. Защитные механизмы системы превентивной безопасности PRE-SAFE® срабатывают в случаях резкого торможения, когда водитель стремится сильнее замедлить автомобиль, чем это возможно при данных условиях на дорожной полосе, а также при критических поворотах управляемых колес на высокой скорости.(рисунок 2)



Рисунок 2 – Работа PRE-SAFE

Датчики обнаруживают на пути автомобиля препятствие, будь то стена, столб другой автомобиль, либо просто посторонний объект соответствующего размера. Сигнал об опасном сближении передается на устройство управления, которое мгновенно переводит в безопасное положение Автомобильные сидения, увеличивает натяжение ремней безопасности. Срабатывание подушек безопасности в случае столкновения зависит от веса и комплектации пассажиров. Система также закрывает боковые стекла и люк при опасности переворота и увеличивает тормозное усилие, а также помогает в рулении. На сегодняшний момент системой PRE-SAFE комплектуются купе CL-класса, седаны, кабриолеты и универсалы E-класса, внедорожники M, R и GL классов, седаны S- класса, а также как опция, может быть установлена на C-класс, как седаны, так и универсалы. Стоимость этой опции – 24 300 руб.

1.3.3. Активный Круиз-Контроль с функцией STOP&GO от BMW.

Это радиолокационные датчики непрерывно сканирующие дорожное полотно(рисунок 3), которые адаптируют скорость и поддерживают безопасную дистанцию до движущегося впереди автомобиля. Когда полоса движения освобождается, система автоматически увеличивает скорость автомобиля до заданной.

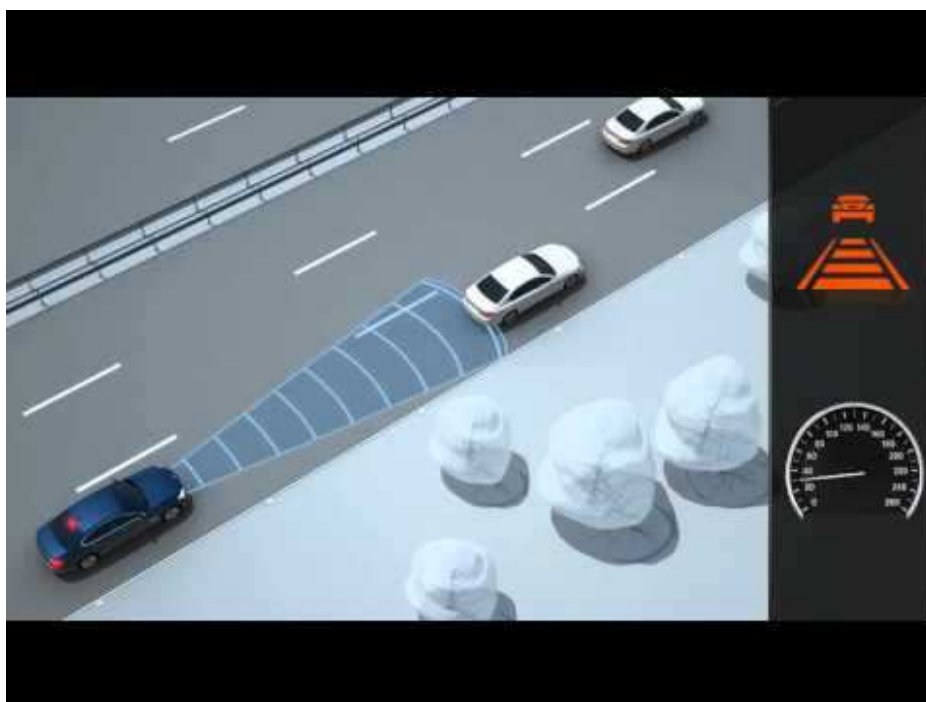


Рисунок 3 – Работа STOP&GO

Включает в себя радарную систему предупреждения о столкновении с функцией торможения до полной остановки обеспечивает максимальный комфорт и безопасность при движении в пробках. Система автоматически поддерживает заданную водителем скорость и дистанцию до впереди идущего автомобиля. Автомобиль ускоряется и замедляется автоматически вплоть до полной остановки, контролируя дистанцию даже в плотном трафике и в пробках. Если автомобиль начинает приближаться к впереди идущим участникам дорожного движения, или если между ним и впереди идущим автомобилем перестраивается другой автомобиль, система точно корректирует скорость, подстраиваясь под изменения дорожной ситуации. Система может быть активирована на скоростях до 210 км/ч.

Функция Stop&Go не только обеспечивает автоматическое замедление вплоть до полной остановки, но и самостоятельно разгоняет автомобиль до заданной скорости, если в пределах определенного интервала времени дорожная ситуация снова это позволяет. В случае если впереди идущий автомобиль резко тормозит, автомобиль BMW мгновенно реагирует на это активацией системы предупреждения о столкновении. Она действует в три

этапа: сначала загорается предупреждающий индикатор, затем индикатор начинает мигать и система подает звуковой сигнал, и наконец, система интенсивно задействует тормоза автомобиля. Стоимость данной опции у автодилеров BMW – 26 400 руб.

1.3.4. Система автоматического торможения Active City Stop (ACS)

Эта система использует сложный световой датчик (рисунок 4), измеряющий расстояние и способный сканировать дорожную ситуацию 50 раз в секунду, что более чем в два раза превышает стандартную частоту кадров, принятую в кинематографе. Это позволяет предотвратить столкновение при скорости автомобиля до 15 км/ч и способствует минимизации последствий аварии при скорости до 30 км/ч (Fiesta, Kuga), до 40 км/ч (Mondeo), до 50 км/ч (Focus).

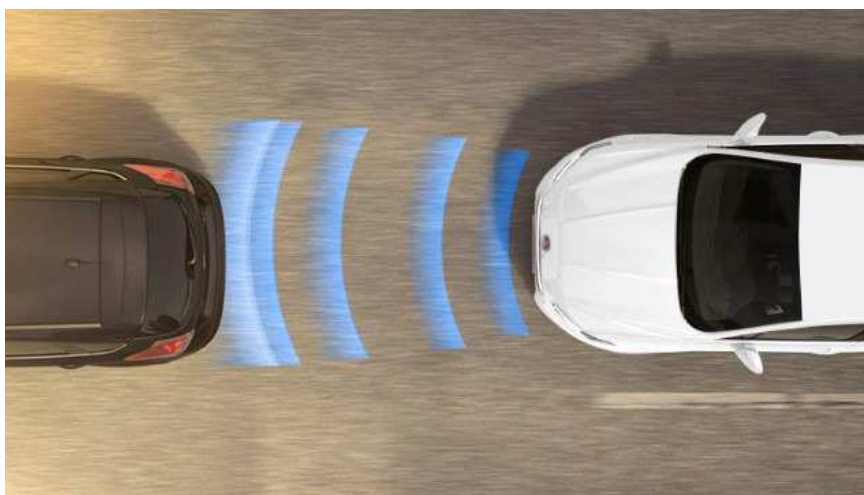


Рисунок 4 – Световой датчик ACS

Условия движения на городских улицах с плотным транспортным потоком и множеством отвлекающих факторов часто провоцируют ситуации, на которые водители не всегда успевают среагировать вовремя, что приводит к мелким авариям на дороге и потере времени всеми участниками движения. В таких условиях система автоматического торможения становится незаменимым помощником – буквально «дополнительным глазом», позволяющим предотвращать аварийные ситуации на дороге, а также

минимизировать или вовсе избежать последствий столкновения, в результате которых люди могут получить травмы.

Данная система автоматического торможения осуществляет постоянный мониторинг расстояния до впереди идущего автомобиля, рассчитывает риск столкновения с неподвижным или движущимся с малой скоростью объектом и запускает процесс предварительной активации тормозной системы с целью обеспечения максимально эффективного торможения. Если водитель не предпринимает каких-либо активных действий (например, не поворачивает рулевое колесо или не нажимает на педаль тормоза), система мгновенно приводит в действие тормозную систему, снижает крутящий момент двигателя и активирует аварийную световую сигнализацию. Система автоматического торможения Active City Stop впервые была представлена в России на новом смарт-кроссовере Ford Kuga и теперь также доступна покупателям Ford Fiesta, Ford Focus и Ford Mondeo.

2. Аналитическая модель

2.1. Модель определения безопасного расстояния

2.1.1. Обзор алгоритма расчета безопасного расстояния

При торможении на автомобиль действует сила трения скольжения, поэтому по 2 закону Ньютона:

$$\mu mg = ma \quad (1),$$

Отсюда

$$a = \mu g \quad (2).$$

Путь до полной остановки рассчитывается по формуле:

$$S = v^2 / 2a \quad (3).$$

Тогда тормозной путь равен:

$$S = v^2 / 2\mu g \quad (4).$$

Остановочный путь равен сумме пути автомобиля во время реакции водителя и тормозного пути:

$$S = vt + (v^2 / 2\mu g) \quad (5), \text{ где}$$

μ - Коэффициент трения шин об дорогу,

t – Время реакции водителя и тормозной системы

v – Скорость автомобиля

Из (5) следует, что для вычисления безопасного расстояния нужно учитывать два параметра – скорость и коэффициент сцепления. Для того чтобы обеспечить отсутствие лишних срабатываний, следует учитывать еще относительную скорость от впередиидущего автомобиля.

2.1.2. Расчет относительной скорости по разностной схеме

Разностная схема — это конечная система алгебраических уравнений, поставленная в соответствие какой-либо дифференциальной задаче, содержащей дифференциальное уравнение и дополнительные условия. Таким образом, разностные схемы применяются для сведения дифференциальной задачи, имеющей континуальный характер, к конечной системе уравнений, численное решение которых принципиально возможно на вычислительных машинах.

Для вычисления относительной скорости между автомобилями по разностной схеме требуется два раза через определенный промежуток времени считать расстояние до впередиидущего препятствия.

$$v = \frac{l_1 + l_2}{t} \quad (6), \text{ где}$$

l_1, l_2 – измеренные расстояния,

t – время задержки между измерениями

2.1.3. Расчет абсолютной скорости

Водители редко учитывают, что впередиидущий автомобиль резко может остановиться и приравнять свою скорость нулю. Это пренебрежение часто является причиной аварий, при несоблюдении безопасного расстояния. Для этого следует считать безопасное расстояние по абсолютной скорости.

Все-же такое предупреждение будет являться не критичным, чтоб не отвлекать сильно водителя, было принято решение сделать предупреждение по абсолютной скорости с помощью желтого светодиода и без звукового сопровождения.

Скорость автомобиля считывается с помощью GPS. Модуль GPS сам рассчитывает скорость и возвращает все данные. Для работы с этим модулем используется библиотека TinyGPS.

2.2. Коэффициенты трения

2.2.1. Влияние качества дорожного покрытия на коэффициент сцепления

Это самый важный параметр, влияющий на сцепление шины с дорогой. Ниже приведена таблица зависимостей коэффициента сцепления от типа дорожного покрытия.

Таблица 1 - Зависимость коэффициента сцепления с дорогой от типа дорожного покрытия

| Тип дорожного покрытия | Коэффициент сцепления |
|------------------------|-----------------------|
| Сухой асфальт | 0,8 |
| Влажный асфальт | 0,6 |
| Мокрый асфальт | 0,4 |
| Рыхлый снег | 0,3 |
| Укатанный снег | 0,2 |
| Лед | 0,1 |
| Мокрый лед | 0,05 |

2.2.2. Влияние температуры шины на коэффициент сцепления

Деформация любого тела приводит к его нагреву. Шина в процессе езды деформируется, особенно при разгонах, торможениях и поворотах, и, как следствие, нагревается. При прямолинейном и равномерном движении она тоже деформируется, но в большей степени из-за вертикальных колебаний вследствие дорожных неровностей.

Коэффициент сцепления шины с дорогой зависит от температуры шины, причем его максимуму соответствует некая оптимальная температура. То есть на холодной шине коэффициент имеет какое-то значение, при нагреве увеличивается, а при перегреве шины снова уменьшается. Оптимальная температура для разных шин разная, для летних дорожных шин она находится в интервале 60-90 градусов, для гоночных шин – выше и может превышать 100 градусов. На рисунке 5 приведена зависимость коэффициента сцепления от температуры шин.

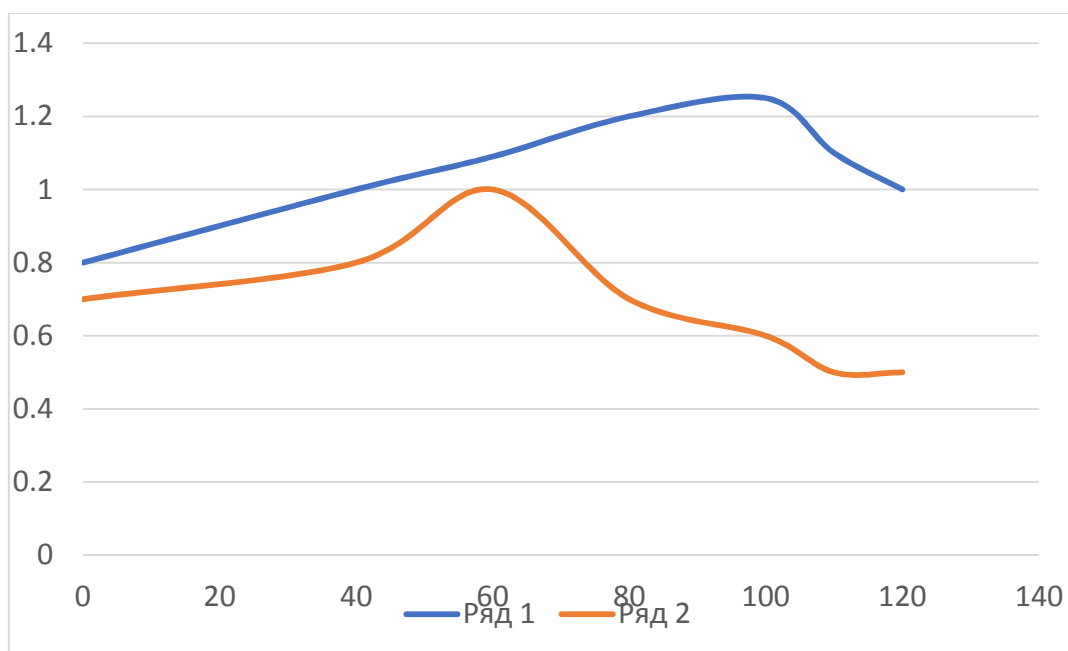


Рисунок 5. Зависимость коэффициента сцепления от температуры шин

2.2.3. Влияние скорости движения автомобиля на коэффициент сцепления

В случае мокрой дороги дождевые канавки в протекторе шины при увеличении скорости успевают отводить все меньше воды из пятна контакта. Поэтому чем больше скорость, тем больше воды скапливается в пятне контакта и тем меньше коэффициент сцепления.

В конце концов может наступить момент, когда шина полностью потеряет контакт с дорогой и всплывает, а автомобиль потеряет управляемость. Это явление называется аквапланированием. Критическая скорость, при которой обычно возникает аквапланирование – около 90 км/ч, а толщина водяной пленки – несколько сантиметров. Как правило, аквапланирование возникает во время дождя при движении в асфальтовой колее. При большей толщине водяной пленки аквапланирование не возникает, потому что создается сильное сопротивление качению шин, которое быстро гасит скорость, и до критической скорости машина не может разогнаться. Если же толщина водяной пленки небольшая, скажем, пара миллиметров, аквапланирования вообще не будет. Просто коэффициент сцепления с дорогой немного

уменьшится, к примеру, с 0,8 на сухой дороге до 0,6 на влажной. При этом чем шире профиль шины, тем при меньшей скорости наступает аквапланирование.

При движении по сухой дороге коэффициент сцепления тоже незначительно уменьшается. Это объясняется тем, что при увеличении скорости возрастает частота вертикальных колебаний шины, и в результате шина контактирует с поверхностью дороги меньшее время. То есть она не успевает охватывать микронеровности дороги, как при более низкой скорости. Этот факт оказывает негативное влияние на тормозные свойства автомобиля при больших скоростях. Мало того, что тормозной путь сам по себе пропорционален квадрату скорости – то есть увеличивается в 4 раза при увеличении скорости в 2 раза.

В таблице 2 приведены зависимости коэффициента сцепления от скорости движения автомобиля на разных дорожных покрытиях. На рисунке 6 график изменения коэффициента сцепления с изменением скорости.

Таблица 2. Зависимость коэффициента сцепления от скорости движения автомобиля на разных дорожных покрытиях.

| Скорость движения автомобиля, км/ч | Коэффициент сцепления для различных типов дорожного покрытия | | |
|------------------------------------|--|-------|------|
| | Асфальт | Дождь | Лужа |
| 50 | 0,85 | 0,55 | 0,5 |
| 90 | 0,8 | 0,3 | 0,05 |
| 130 | 0,75 | 0,2 | 0 |

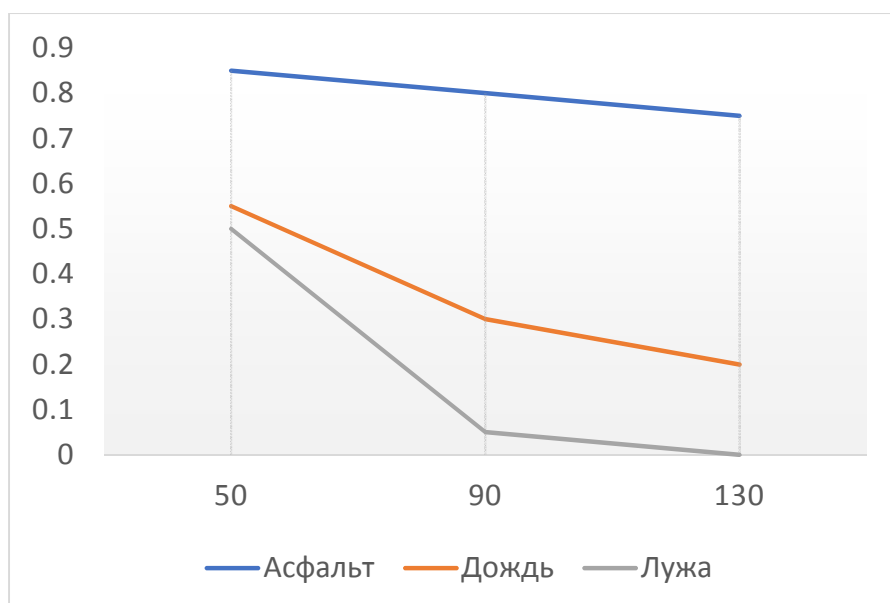


Рисунок 6 – Зависимость коэффициента сцепления от скорости движения

Из таблицы видно, что коэффициент сцепления при увеличении скорости незначительно уменьшается на сухой дороге, более явно уменьшается на мокрой дороге в дождь и при попадании в длинную лужу (колею) исчезает при скорости выше 90 км/ч, то есть при возникновении аквапланирования.

3. Алгоритм

3.1. Исходный алгоритм

Исходный алгоритм представляет собой алгоритм предупреждения по абсолютной скорости в котором считывается расстояние и скорость автомобиля. Далее рассчитывается безопасное расстояние по формуле 5 на основе текущей скорости автомобиля. На рисунке 7 изображена ситуация при котором автомобиль 1 движется, а автомобиль 2 принял скорость равную 0 и остановился.



Рисунок 7 – Остановочный путь по относительной скорости

Если рассчитанное безопасное расстояние меньше считанного, то выводится предупреждение в виде желтого светодиода. При отсутствии опасности на следующих итерациях, индикатор отключается.

3.2. Алгоритм с учетом относительной скорости

Алгоритм считывает два раза через определенный промежуток времени расстояние до препятствия. Далее по разностной схеме рассчитывается относительная скорость автомобиля.

$$\Delta v = \frac{l_1 + l_2}{t}, \text{ где} \quad (6)$$

l_1, l_2 – расстояния считанные через t .

t – время между считаванием расстояний.

После высчитывается скорость 2-го автомобиля.

$$\Delta v = v_1 + v_2 \quad (7)$$

$$v_2 = \Delta v - v_1 \quad (8)$$

Узнав скорости автомобилей, рассчитывается остановочный путь для обеих автомобилей по формуле 5. Если остановочный путь первого автомобиля меньше остановочного пути второго (рисунок 8), то выводится предупреждение в виде красного светодиода и прерывистого звукового сигнала.

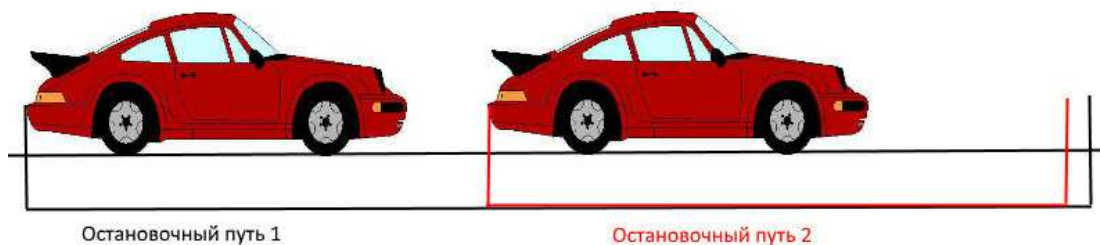


Рисунок 8– Остановочный путь двух автомобилей

На рисунке 8 показана ситуация при котором тормозной путь первого автомобиля меньше тормозного пути второго. При такой ситуации предупреждение будет отключено.

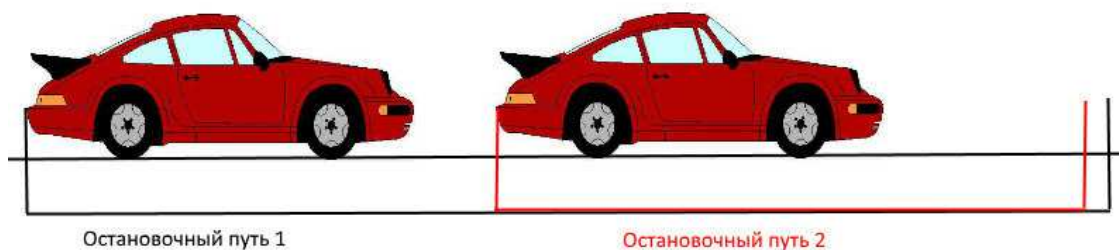


Рисунок 8 – Тормозной путь автомобилей

3.3. Алгоритм с учетом поворота головы

Лазерный дальномер в отличие от ультразвукового измеряет расстояние до точки в пространстве, а для понимания полной картины происходящего перед автомобилем требуется обзор препятствий по всей ширине автомобиля. Для решения данной проблемы было предложено поставить дальномер на вращающийся сервопривод.

Сервопривод с закреплённым на нём датчиком расстояния совершает движение с шагом 5 градусов в две стороны. На каждом шагу движения сервопривода совершается проверка на безопасность движения.

3.4. Определения вхождения препятствия в диапазон автомобиля

Датчик на каждые 10 градусов измеряет расстояние до препятствия в связи с этим могут возникать случаи, когда препятствие находится за пределами диапазона автомобиля, следует их учесть. Данная ситуация изображена на рисунке 10.

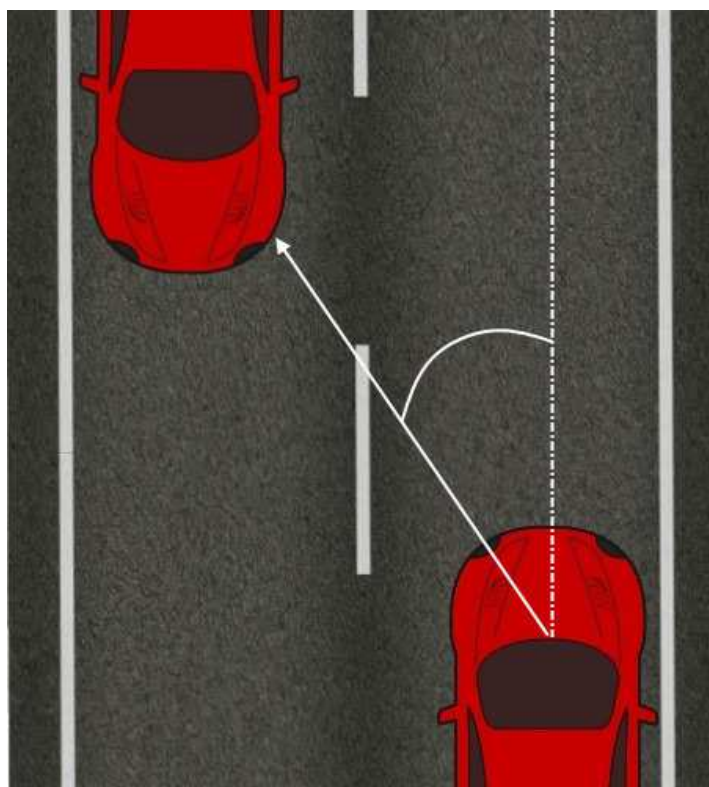


Рисунок 10 – Вхождение в диапазон

Для определения вхождения автомобиля в диапазон автомобиля предложена следующая формула.

$$w = \cos(\alpha)l \quad (7), \text{ где}$$

w – расстояние от датчика до препятствия по x

α – текущий угол поворота сервопривода

l – расстояние до препятствия

После нахождения w её следует сравнить с половиной ширины автомобиля, если w превышает её, то препятствие не входит в диапазон автомобиля и это препятствие не обрабатывается.

3.5. Количество шагов сервопривода

Был проведен анализ для выявления оптимального количества шагов и угла шага для сервопривода. Изначально было предусмотрено 5 шагов с углом в 10 градусов. После анализа данных выяснилось, что такое количество шагов не подходит для дальних расстояний, а именно для максимального расстояния в 50 метров.

Для достижения оптимальных результатов требуется, чтоб при анализе на максимальном расстоянии диапазон шага сервопривода не составлял больше 0.5 метра. То есть:

$$\sin\gamma = \frac{0.5}{l} \quad (8), \text{ где}$$

l – измеренное расстояние

γ – угол шага сервопривода

То есть угол шага для максимального расстояния будет составлять 2 градуса.

3.6. Результаты

После проектирования алгоритма, был написан программный код для платформы Arduino в среде программирования Arduino 1.8.5. Блок-схемы основного цикла программы изложено в приложении А. Блок-схема основной функции проверки опасности изложен в приложении Б. Полностью код программы приведен в приложении В.

Программный код разделен на подпрограммы, которые отвечают за определенную задачу. Такой подход позволяет быстро вносить изменения и тестировать весь функционал. Так же код является масштабируемым, что позволит расширять функционал программы.

4. Тестирование

4.1. Создание тестовых задач

Для обеспечения устойчивости алгоритмов к изменениям следует написать автотестовые программы, которые автоматически будут проводить тестирование всех функций. Это также позволит сократить время разработки и увеличит стабильность работы алгоритма.

Следует разработать методы тестирования уже собранного устройства и анализа полученных данных, для подбора устойчивого алгоритма и учета ошибок.

В качестве данных для проверки правильности работы алгоритма была использована база безопасных расстояний в зависимости от скорости автомобиля.

4.2. Автотестирование алгоритмов

Предполагается, что алгоритм будет подвергаться изменениям и сами коэффициенты тоже, чтоб увеличить скорость тестирования алгоритма было предложено написать автотестирующую программу, которая на основе тестовых данных и данных о правильных результатах будет проверять алгоритм на процент расхождения рассчитанного безопасного расстояния. Если процент расхождения превышает 20%, то алгоритм считается не эффективным и продолжаются его дальнейшие доработки.

В качестве языка проведения автотестов был выбран Python в связи с его быстрой скоростью разработки и легким переносом на иную платформу.

Было написано несколько отдельных функций с формулами расчетов безопасного расстояния и относительной скорости.

Создан набор искусственных данных скорости автомобиля, расстояний до препятствия, которые используются для проведения тестирования.

После внесения правок в алгоритм все функции подвергались автотестированию и анализу результатов.

4.3. Тестирование собранного устройства

Для проведения тестов для устройства одним из разработчиков проекта был предоставлен автомобиль. Устройство, во время детектирования опасности записывало отладочную информацию в файл. В файле содержалась информация по скорости, расстоянию, считанному безопасному расстоянию и предупреждению.

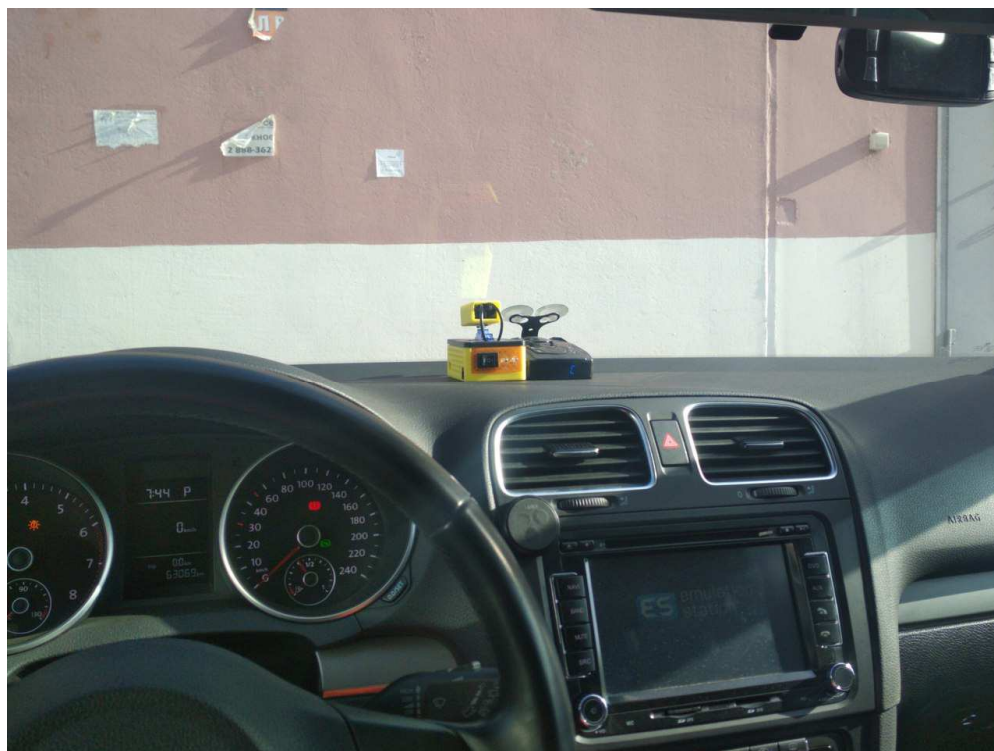


Рисунок 12 – Устройство в автомобиле

4.4. Обработка результатов тестирования

Машина ездилa по плотному потоку города час. Устройство выдало 10 предупреждений, 5 срабатываний были верными. В трех случаях устройство выдало предупреждение, когда автомобиль не был подвержен опасности. В других двух предупреждение было выдано из-за шумовых значений.

После просмотра логов выяснилось, что устройство выдало предупреждение, когда он не был подвержен опасности из-за неправильно подобранного коэффициента сцепления. Коэффициенты сцепления требовали уменьшения на 0,1.

В 2-ух случаях датчик считывания расстояния выдал шумовое значение. После этого было принято решение написать обработчик обработки шумовых значений.

ВЫВОД

Вместе с научным руководителем и партнером изучены аналоги и требования к устройству. Проведен обзор параметров влияющие на стабильность и работоспособность алгоритма. Был разработан алгоритм для предупреждения водителя при возникновении опасности столкновения с впередиидущим автомобилем. Продуман процесс тестирования устройства, который помогает быстро определить эффективность алгоритма, что позволило быстро адаптировать изменения и учесть множество ситуаций, которые могут возникнуть на дороге в реальных условиях.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

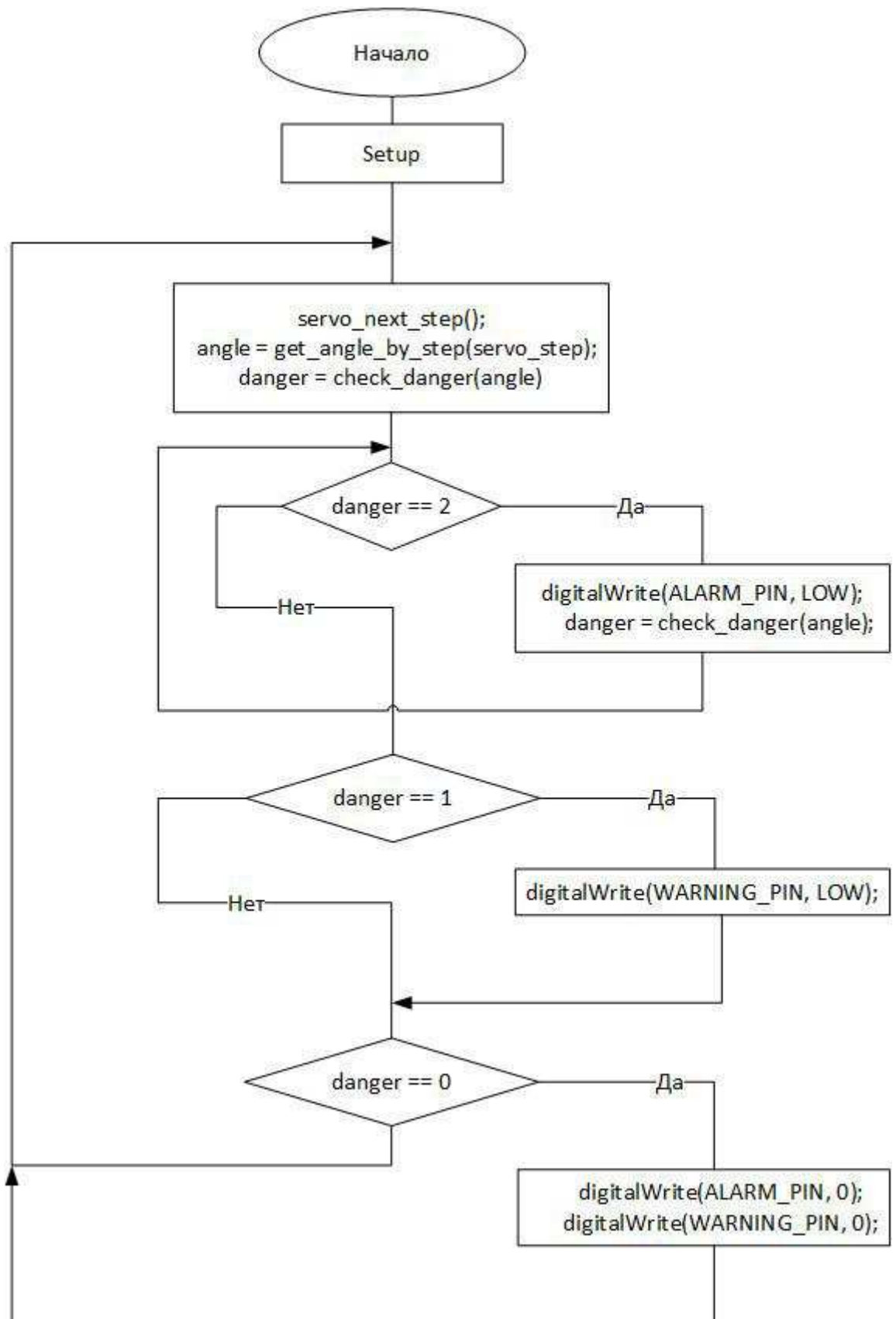
1. Как рассчитать тормозной путь [Электронный ресурс] Режим доступа: <https://sites.google.com/site/opatpofizike/teoria/-pirtormozi/statistika-dtp-po-aroslavskoj-oblasti-za-2013-god>
2. Разностная схема [Электронный ресурс] Режим доступа: https://ru.wikipedia.org/wiki/Разностная_схема
3. От чего зависит сцепление шин с дорогой(Часть 1) [Электронный ресурс] Режим доступа: <https://kaminsky.su/blog/ot-chego-zavisit-sceplenie-shin-s-dorogoj-chast-1>
4. От чего зависит сцепление шин с дорогой(Часть 2) [Электронный ресурс] Режим доступа: <https://kaminsky.su/blog/ot-chego-zavisit-sceplenie-shin-s-dorogoj-chast-2>
5. От чего зависит сцепление шин с дорогой(Часть 3) [Электронный ресурс] Режим доступа: <https://kaminsky.su/blog/ot-chego-zavisit-sceplenie-shin-s-dorogoj-chast-3>
6. City Safety™ [Электронный ресурс] Режим доступа: <https://support.volvocars.com/ru/cars/Pages/owners-manual.aspx?mc=312H&my=2016&sw=15w17&article=806aaf1ff78fb6b5c0a801e800ca3e51>
7. Система превентивной безопасности PRE-SAFE® [Электронный ресурс] Режим доступа : https://www.mercedes-benz.ru/content/russia/mpc/mpc_russia_website/ru/home_mpc/van/home/new_vans/models/vito_447/mixto/advantages/safety/pre-safe.html
8. Активный круиз-контроль с функцией Stop & Go [Электронный ресурс] Режим доступа: http://www.ezauto.lv/ru/technologies/active_cruise_control_with_stop_g_o_function
9. СИСТЕМА АВТОМАТИЧЕСКОГО ТОРМОЖЕНИЯ ACTIVE CITY STOP (ACS) [Электронный ресурс] Режим доступа: <http://www.ford.ru/Technology/Driver-Assist/Active-City-Stop>

10. Автоматизированное тестирование [Электронный ресурс] Режим
доступа:

https://ru.wikipedia.org/wiki/Автоматизированное_тестирование

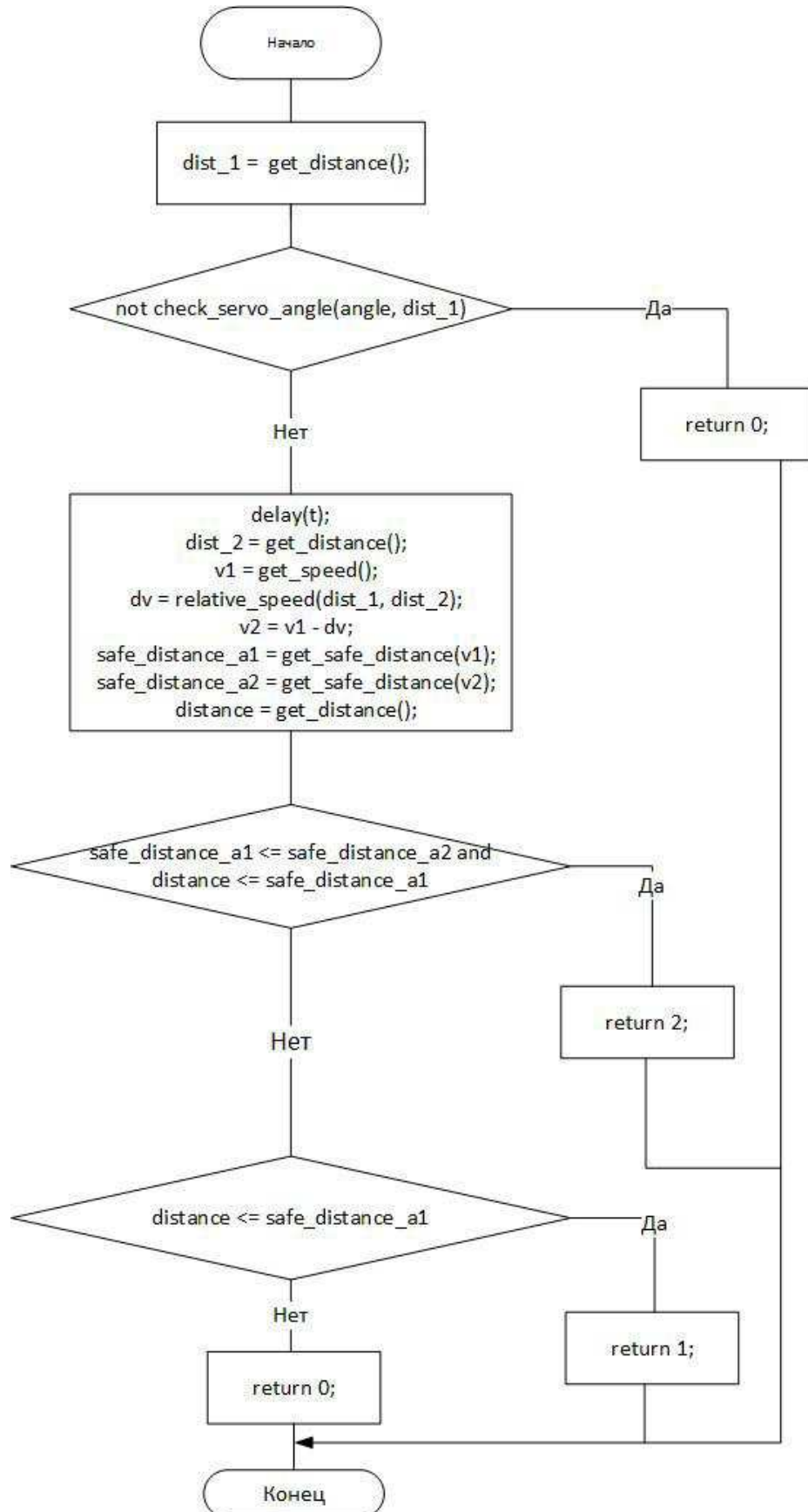
Приложение А
(обязательное)

Алгоритм основной части программы



Приложение Б (обязательное)

Алгоритм проверки опасности



Приложение В

(обязательное)

Исходный код программы

```
#include <Servo.h>
#include <AltSoftSerial.h>
#include <SoftwareSerial.h>
#include <TinyGPS.h>

SoftwareSerial RangefinderUART(10, 11); // RX, TX
AltSoftSerial GPSUART(8, 9); // RX(8), TX(9)
TinyGPS gps;
Servo servo;

const int W = 2000; // Ширина капота(см)
const int ALARM_PIN = 9; // Красный светодиод
const int WARNING_PIN = 8; // Желтый светодиод
const int SERVO_MAX = 80;
const int SERVO_MIN = 20;

// Вводные параметры
int t = 100; // Задержка между расчетами (м/с)
float k = 1.7; // Коэффициент трения TODO: зависимость от
выбранного режима

int angle = 0; // Угол поворота сервопривода
int servo_step = SERVO_MIN; // Шаг сервопривода
float step_max = 10;
float step_count = 5;
int danger = 0;
```

```

boolean trig = true; // Сторона движения сервопривода
float v1, v2, dv;
float safe_distance_a1, safe_distance_a2, distance;
float dist_1, dist_2;
char result[15];
char d[5];

void setup() {
  Serial.begin(19200);
  while (!Serial) ;
  GPSUART.begin(19200);
  GPSUART.read();
  RangefinderUART.begin(19200);
  servo.attach(2);
  servo.write(0);

  pinMode(ALARM_PIN, OUTPUT);
  pinMode(WARNING_PIN, OUTPUT);
}

void loop() {
  servo_next_step();

  angle = get_angle_by_step();

  danger = check_danger();

  while (danger == 2) {
    digitalWrite(ALARM_PIN, LOW);

```

```

    danger = check_danger();
}

if (danger == 1) {
    digitalWrite(WARNING_PIN, LOW);
}

if (danger == 0) {
    digitalWrite(ALARM_PIN, 0);
    digitalWrite(WARNING_PIN, 0);
}

delay(100);
}

int check_danger() {
    // Проверка на опасность
    // return 2 - Включить красный светодиод, 1 - Включить желтый
    // светодиод, 0 - Ничего не включать

    dist_1 = get_distance();
    // Проверка вхождения текущего препятствия в диапазон
    // автомобиля
    if (not check_servo_angle()) {
        return 0;
    }

    delay(t);
    dist_2 = get_distance();

```

```

v1 = get_speed(); // Скорость первого автомобиля
dv = relative_speed(); // Относительная скорость
v2 = v1 - dv; // Скорость второго автомобиля

safe_distance_a1 = get_safe_distance();
safe_distance_a2 = get_safe_distance();

distance = get_distance();

// Проверка по относительной скорости
if ((safe_distance_a1 <= safe_distance_a2) and (distance <=
safe_distance_a1)) {
    return 2;
}

// Проверка по абсолютной скорости
if (distance <= safe_distance_a1) {
    return 1;
}

return 0;
}

void servo_next_step() {
    step_max = sin(0.5/dist_1);
    step_count = (SERVO_MAX - SERVO_MIN) / step_max;

    if(!trig){servo_step += step_count;}
}

```

```

if(trig){servo_step -= step_count;}
if(servo_step == SERVO_MAX){trig = true;}
if(servo_step == SERVO_MIN) {trig = false;}
delay(5);// Необходима для исключения влияния помех сервы на
датчик
servo.write(servo_step);
}

bool check_servo_angle() {
// Проверка вхождения препятствия в диапазон автомобиля
return cos(angle)*dist_1 >= W/2;
}

float relative_speed() {
// Расчет относительной скорости по разностной схеме
// dist_1 - Первая дистанция
// dist_2 - Вторая дистанция
return (dist_1 - dist_2) / (t/1000);
}

float get_safe_distance() {
// Расчет безопасного расстояния
// v - скорость в см/с
return (v*1.2) + ((v*v) / (2*k*9.81));
}

float get_speed() {
// Считывание скорости с устройства
return gps.speed.kmph();
}

```

```
float get_distance() {  
    // Считывание расстояния с устройства  
    RangefinderUART.send("F");  
    result = RangefinderUART.read();  
    d = "";  
    for (int i = 3; i < 8) {  
        d = d + result[i]  
    }  
    return int(d);  
}
```

```
int get_angle_by_step() {  
    switch(servo_step) {  
        case 20..30:  
            return 30;  
        case 30..40:  
            return 60;  
        case 40..50:  
            return 90;  
        case 50..60:  
            return 120;  
        case 70..80:  
            return 150;  
    }  
}
```

Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Политехнический институт

Кафедра «Конструкторско-технологического обеспечения
машиностроительных производств»

УТВЕРЖДАЮ

Заведующий кафедрой

М. П. Головин

подпись

«_____» _____ 2018 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.01 – Информатика и вычислительная техника

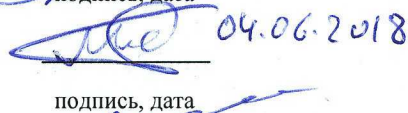
Разработка программного обеспечения для устройства, предупреждающего о
столкновении

Научный руководитель


04.06.18
подпись, дата

ст.преподаватель Р.С. Лукин

Выпускник


04.06.2018
подпись, дата

Э. Г. Агамамедов

Нормоконтролер


04.06.18
подпись, дата

ст.преподаватель Р.С. Лукин

Красноярск 2018