

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	6
1 ОБЗОР ЛИТЕРАТУРЫ .....	8
1.1 Общие сведения.....	8
1.2 Способы измерения скорости .....	8
1.3 Обоснование выбора метода измерения .....	9
1.4 Навигационной системы .....	10
1.5 Принцип работы системы GPS .....	10
1.6 Описание протокола NMEA0183.....	12
1.6.1 GPGGA .....	12
1.6.2 GPGLL .....	13
1.6.3 GPVTG .....	14
1.7 Обзор архитектуры AVR.....	14
1.8 Интерфейс I <sub>2</sub> C.....	16
1.9 Обзор аналогов .....	18
2 РАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ .....	20
2.1 Микроконтроллер.....	20
2.2 Модуль GPS .....	21
2.3 Источник питания .....	21
2.4 Каскад согласования измеряемого напряжения .....	21
2.5 Датчик освещенности.....	21
2.6 Модуль индикации .....	21
2.7 Преобразователь напряжения.....	22
2.8 Часы реального времени .....	22
3 РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ СХЕМЫ .....	23
3.1 Микроконтроллер.....	23
3.1.1 Подсистема ввода/вывода GPIO.....	24
3.1.2 Вычислительное ядро .....	26
3.1.3 Аналогово-цифровой преобразователь.....	26
3.1.4 UART .....	27
3.1.10 Выбор архитектуры .....	28
3.2 Модуль GPS .....	28
3.3 Источник питания .....	30
3.4 Каскад согласования измеряемого напряжения .....	30
3.5 Датчик освещенности.....	31
3.6 Преобразователь напряжения.....	32
3.7 Модуль индикации .....	32
3.7.1 Звуковая индикация .....	33
3.7.2 Знакосинтезирующая индикация .....	33
3.8 Часы реального времени .....	37
3.9. Алгоритм функционирования комплекса .....	38

4 РАЗРАБОТКА ПРИНЦИПИАЛЬНОЙ СХЕМЫ .....	42
4.1 Обоснование выбора схемы и расчет дополнительных элементов .....	42
4.1.1 Преобразование питания .....	43
5 ТЕХНИКО - ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ РАЗРАБОТКИ И ПРОИЗВОДСТВА КОМПЛЕКСА ИЗМЕРЕНИЯ СКОРОСТИ ОБЪЕКТА.....	47
5.1 Характеристика аппаратно-программного комплекса .....	47
5.2 Расчет экономического эффекта от производства аппаратно- программного комплекса .....	47
5.3 Расчет инвестиций в проектирование и производство аппаратно- программного комплекса .....	51
5.3.1 Расчет инвестиций на разработку аппаратно-программного комплекса .....	51
5.3.2 Расчет инвестиций в прирост оборотного капитала .....	52
ЗАКЛЮЧЕНИЕ .....	54
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	55
ПРИЛОЖЕНИЕ А .....	56
ПРИЛОЖЕНИЕ Б.....	58

## ВВЕДЕНИЕ

Развитие прогресса во второй половине XX века сопутствовал вместе с противостоянием сверхдержав в холодной войне. В XXI веке эти достижения продолжились развиваться, но в мирное русло, применяя достижения прошлого и нового веков в мирных целях. Многие вещи, которые используются в повседневном обиходе у людей, произошли от достижений в военно-промышленном комплексе: пластик, липучки, обезболивающие средства, лейкопластырь, микроволновая печь и т.д.

Сферу высоких технологий электронную отрасль эти процессы также не обошли стороной. Достижения прошлого века, придуманные в первую очередь для эффективного ведения боевых действий, в нашем веке стали все больше применяться в мирных сферах жизни людей. Все более и стремительнее. Современные достижения в области высоких технологий основаны именно на этом прогрессе прошлого. Конечно, они морально устарели, зато создали отличную основу для развития новых технологий.

А в тоже время, в современном обществе произошло развитие и улучшение социального положения человека. Все больше и больше людей способны приобрести то, что ранее считалось невозможным среднему классу и было доступно только обеспеченному слою населения. Одним из предметов является автомобиль. Все больше и больше на сегодняшний день появляется автомобилистов, в крупных городах все чаще и чаще в часы пик стали появляются автомобильные пробки, а также дорожно-транспортных происшествий.

Современные автомобили оснащаются всеми современными устройствами индикации, которые сигнализируют об окружающей обстановке. Но до сих пор у конструкторов автомобилей не сформировался ответ на главный вопрос: так какая же информация необходима и важна для глаз водителя? Приборный интерфейс постоянно модернизируется и совершенствуется. Необходимость информировать водителя о любых изменениях показателей движения автомобиля и окружающей обстановки с целью улучшения безопасности движения и комфортной поездки. И в тоже время не перегружать водителя лишней информацией и временем за просмотром всех этих показателей. Ведь пользоваться водителем ею должен во время движения транспортного средства.

Еще одной проблемой, для многих пользователей транспортным средством, является контроль скорости. В современных автомобилях продолжают использоваться стрелочные индикаторы скорости, что не совсем удобно при быстрой езде. Так как внимание уделяется на то чтобы посмотреть текущую скорость, при условии, что не на всей шкале нанесены цифры. Водитель должен с помощью одного взгляда узнать свою текущую скорость. Также, все производители автомобилей намеренно изменяют показания приборов по контролю скорости. Спидометры показывают завышенные показатели скорости. Все это связано с «психологическим эффектом» -

уберечь водителя от превышения скорости. Но если смотреть объективно, то больше 60% нарушений водителей в Республике Беларусь является превышение скорости. Искоренить это невозможно. Научить культуре вождения - тоже. Но можно их предостеречь.

Целью данного дипломного проекта является разработка комплекса по измерению скорости объекта. Данный комплекс будет работать на основе системы измерения геолокации. Также в нем будет присутствовать минималистичный оригинальный интерфейс, с возможностью автоматической корректировки яркости, чтобы в темноте данный интерфейс не ослеплял водителя. Также, есть и дополнительные функции: измерение напряжения на аккумуляторной батарее автомобиля, часы реального времени с корректировкой, а также количество видимых устройств спутников.

Данное устройство позволит получать необходимую в движении автомобиля информацию о нем. Предназначено для любого наземного транспортного средства.

Исходя из поставленной цели, были сформулированы следующие задачи:

- 1) Проанализировать литературу.
- 2) Составить структурную и функциональную схему прибора.
- 3) Анализ и выбор необходимых компонентов.
- 4) Разработка структуры программы управления контроллером.
- 5) Составить электрическую принципиальную схему.
- 6) Написание программного кода управления контроллером.

# **1 ОБЗОР ЛИТЕРАТУРЫ**

## **1.1 Общие сведения**

С развитием в конце XIX – начале XX вв. механической техники дало толчок к появлению и развитию приборов, которые позволяли мерить необходимые показатели. Одним из таких приборов является спидометр. Он входит в число обязательных приборов, которые необходимы для автомобилиста. И не только: контроль скорости используется практически во всех самоходных средствах (на земле, на воде, в воздухе). Спидометр – измерительный прибор, способный определять модуль мгновенного движения тела относительно земли.

В первой половине XX века спидометр считался очень необычной вещью и экстраординарным прибором. Однако с развитием автомобилестроения, транспортные средства того времени стали способны все большие и большие скорости перемещения. Под влиянием этих факторов, а также увеличением дорожно-транспортных происшествий с участием автомобилей, было принято решение устанавливать спидометр как обязательный измерительный прибор автомобиля. Первым изобретателем спидометра считается сербский ученый Никола Тесла.

## **1.2 Способы измерения скорости**

На сегодняшний день, с развитием науки и технологий, методы и способы измерения скорости постоянно меняются и совершенствуются. Стандартный автомобильный спидометр основан на счете количества оборота вторичного вала за определенное время.

Классификация устройств по измерению скорости [1]:

1) Аэрометрический. Основан на измерении динамического напора, функционально связанного со скоростью тела, движущегося в воздушной среде. Средства измерения, построенные на аэрометрическом методе, позволяют измерять скорость с погрешностью, не превышающей 2 – 3%

2) Акустический. Основан этот способ измерения скорости на приеме электроакустическим преобразователем звука от движущихся пар. При этом способе, выделяют так же тональную составляющую из спектра шумоизлучения автомобиля. Этот способ может применяться в условиях плохой видимости, но не является в достаточной степени надежным и не позволяет произвести идентификацию транспортного средства, поэтому такой способ не получил распространения.

3) Корреляционный. Предназначен в основном для контроля скорости движущихся транспортных средств. За перемещаемым объектом следят два фотоприемника, размещенные на расстоянии друг от друга. Они воспринимают отраженный от движущегося объекта свет. Сигналы фотоприемников усиливаются, фильтруются и преобразуются в цифровой вид

после чего поступают в процессор, который выполняет функции коррелятора. При появлении нерегулярностей соответствующие пики выходного сигнала второго фотоприемника оказываются сдвинутыми, что сигнализирует об изменении и на сколько именно.

4) Использование спутниковой системы навигации. Он основан на измерении задержки распространения сигнала от спутника до приемника. Из полученного сигнала приемник получает данные о местонахождении спутника. Подробнее рассмотрено в подразделе 1.4.

5) Оптический. Основан на использовании лазера (или лидара). Лазер излучает на уровне инфракрасного диапазона несколько коротких импульсов в сторону измеряемого объекта. Далее это излучение дойдя до объекта отражается от него и попадает на лидар. Устройство управления анализирует дальность до объекта в разные фиксированные моменты времени. Используя константное время излучения инфракрасных импульсов, и динамически изменяющееся расстояние из двух точек в двух моментах времени, устройство управления с легкостью вычисляет скорость объекта.

6) Радиочастотный способ, основанный на использовании эффекта Доплера, имеющий в своем составе радар, включающий в себя источник ультразвуковых колебаний с частотой  $f_0$ , и приемник ультразвуковых колебаний, отразившихся от движущегося объекта. Если объект приближается к радару, то частота колебаний  $f$ , отразившихся от него, будет больше первоначальной. Если наоборот, то частота будет меньше. Таким образом, по разности частот  $(f - f_0)$  можно узнать о скорости объекта и о направлении его движения. Есть следующие диапазоны частот: X (10,5 ГГц), K (24,150 ГГц).

### **1.3 Обоснование выбора метода измерения**

Причина, по которой была выбрана методика с использованием навигационных спутников, связана именно с проблематикой обычных автомобильных спидометров – их точность.

Производители транспортных средств намерено завышают показания спидометров в среднем на 5%. Правило ЕЭК ООН № 39 [2], регламентирующее производство измерительных приборов для автомобилей, устанавливает норму, что исправный спидометр автомобиля всегда должен или завышать измеряемую скорость, (на величину не более 10%), или показывать ее абсолютно точно.

Измерение же посредством приема спутникового сигнала всегда показывает скорость движения гораздо более близкую к реальной, чем любой штатный автомобильный спидометр. Некоторое влияние на точность показаний может оказывать положение спутников, но для средних широт эта проблема не так актуальна. Плохая точность такого способа измерения неэффективна в полярных широтах, ввиду удаленности орбит спутников от этих областей. Некоторое влияние на точность измерений скорости могут также оказать рельеф местности, промышленные низкочастотные помехи,

архитектурные сооружения – из-за этих факторов сужается видимый горизонт, но это кратковременные явления.

#### **1.4 Навигационной системы**

В современный век развития, спутниковая навигация развивается быстрыми шагами. В настоящее время в мире существуют четыре спутниковые системы навигации: GPS, Galileo, Бэйдоу и ГЛОНАСС.

Вообще, началом спутниковой навигации считается первый полет искусственного спутника Земли (ИСЗ) в 1957 году [3]. Американские ученые во главе с Ричардом Кешнером наблюдали эффект Доплера на испускаемом сигнале от спутника. Возникла идея от обратного: траектория спутника заранее известна, но неизвестен месторасположение приемника – точно зная положение спутника, можно определить собственную скорость и координаты.

Навигационная Система Глобального Позиционирования *Global Positioning System* (GPS) является частью комплекса навигационной системы определения времени и дальности (*Navigation Satellites providing Time And Range* – NAVSTAR), в США [4]. Разработка комплекса навигационной системы была начата в 1973 году. Гражданский сегмент спутниковой сети NAVSTAR принято называть аббревиатурой GPS, коммерческая эксплуатация которой началась в 1995 году. Основой системы GPS являются 24 навигационных спутника, движущиеся вокруг Земли по 6 круговым орбитальным траекториям, на высоте 20 тысяч километров. Наземная часть системы GPS состоит из десяти станций слежения.

ГЛОбальная Навигационная Спутниковая Система (ГЛОНАСС) была разработана в СССР и запущена в 1982 году. Изначально был также, как и в США, реализован под военные нужды, в первую очередь в навигации на море и позиционирование в пространстве в воздухе. Позже был предоставлен и в гражданский сегмент. Основой системы являются также, как и в GPS 24 спутника, движущихся над поверхностью Земли, но в трех орбитальных плоскостях и на высоте около 19 тысяч километров. Основное отличие от системы GPS в том, что спутники ГЛОНАСС не имеют резонанса (синхронности) с вращением Земли – это обеспечивает большую стабильность, которая в свою очередь не требует постоянных корректировок орбит спутников во время эксплуатации.

#### **1.5 Принцип работы системы GPS**

Основу системы составляют сеть ИСЗ, расположенных на околоземной орбите. Орбита каждого спутника точно рассчитана, для того чтобы в любой момент времени можно было знать месторасположение спутника. И спутник, чтобы быстро воспринимал где он находится. Эти спутники непрерывно излучают сигнал в направлении Земли.

Координаты местоположения GPS-приемника вычисляются по

расстоянию до ИСЗ. Для вычисления расстояния используется свойство распространения радиосигнала – электромагнитные волны распространяются со скоростью света. Как упоминалось ранее в разделе 1.4, спутники излучают радиосигнал. Приемник принимает их с учетом известной времени отправки и времени приема. Это называется *беззапросная* радионавигация. Так узнается расстояние от приемника до спутника. Однако для позиционирования этого недостаточно. Необходима также информация от другого спутника. Зная орбиту и получая координаты обоих спутников, полученные путем измерения сдвига радиочастот (эффект Доплера – что дает еще и возможность получить скорость объекта), в точке пересечения окружностей с радиусом равных расстояниям до обоих спутников мы имеем место нахождения приемника. Третий спутник необходим для определения на какой высоте находится приемник относительно уровня моря.

Однако, при данном методе радионавигации точное определение времени распространения сигнала возможно лишь при наличии синхронизированном времени приемника и ИСЗ. Размещение точных часов (атомных) в приемнике заведомо дорогой вариант. Поэтому в состав аппаратуры ИСЗ входят эталонные часы, причем точность спутникового эталона времени исключительно высокая. [6]

На измерениях всегда присутствует ошибка, обусловленная несовпадением времени у спутника и приемника. По этой причине в приемнике вычисляется искаженное значение дальности до спутника (псевдодальность). Поэтому, необходимо вычислять еще и необходимую поправку к часам приемника. Для их определения необходимо выполнить измерения еще на четвертом спутнике. В результате обработки этих измерений в приемнике вычисляются расстояния до трех спутников, что соответствует позиционированию по трем плоскостям, и точное время.

Спутник излучает синусоидальные сигналы на двух несущих частотах:  $L1 = 1575,42$  МГц и  $L2 = 1227,60$  МГц. Прежде, чем передать их на Землю, сигнал кодируется псевдослучайной цифровой последовательностью – происходит фазовая модуляция. Это необходимо чтобы ограничить доступ к GPS, и одновременно повысить надежность сообщений от шумов, так как псевдослучайные коды самые стойкие к таким помехам.

Сигнал  $L1$  имеет два дальномерных кода с псевдослучайным шумом (PRN), P-код и C/A код (Рисунок 1.1). P-код зашифрован для военных целей. C/A код не зашифрован. Сигнал  $L2$  модулируется только с P-кодом. Большинство гражданских пользователей используют C/A код при работе с GPS системами. Обе несущие частоты дополнительно кодируются навигационным сообщением, в котором содержатся данные об орбитах спутника, информация о параметрах атмосферы, поправки системного времени [6]. Приемник проверяет входящий сигнал со спутника и определяет, когда он генерировал такой же код. Далее происходит долгий, для аппаратной части анализ.



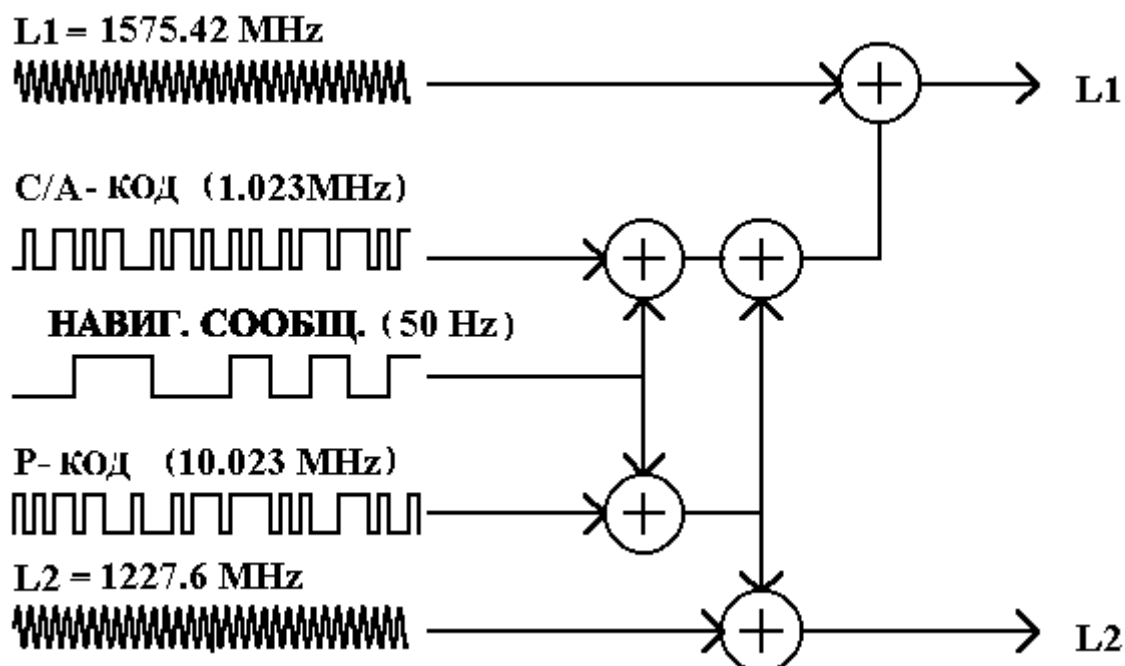


Рисунок 1.1 – Формирование спутникового сигнала [6]

После получения всех необходимых данных формируется пакет. Полученная разница, вместе с синхронизацией по четвертому спутнику и корректировке времени, дает и синхронизацией по четвертому спутнику и корректировке времени, дает искомое расстояние.

## 1.6 Описание протокола NMEA0183

В разрабатываемом многофункциональном GPS-спидометре GPS-приемник выдает сигнал на микроконтроллер в формате NMEA0183 [7]. NMEA – это формат передачи сообщений между корабельными приборами. Он включает в себя систему сообщений для обмена информацией между навигационными GPS-приемниками и потребителями навигационной информации. Все команды и сообщения передаются в ASCII коде. Первый байт сообщения, для идентификации типа навигационной системы, указывается \$GP, что означает что приемник провзаимодействовал с GPS и передает пакет. В последнем поле сообщения может быть указана контрольная сумма текущего сообщения в CRC-коде, разделенная от основного сообщения символом «\*». Контрольная сумма всех символов сообщения, включая пробелы, расположенных между разделителями «\$» и «\*», не включая последних.

### 1.6.1 GPGGA

Сообщение содержит данные о местоположении, времени определения местоположения, качестве данных, количестве использованных спутников,

фактор ухудшения точности координат, информацию о дифференциальных поправках и их возраст. Пример сообщения: \$GPGGA, 004241.47, 5532.8492, N, 03729.0987, E, 1, 04, 2.0, -0015,M,,,\*31, расшифровка представлена в таблице 1.1

Таблица 1.1 – Содержание сообщения \$GPGGA

Тип сообщения	1	2	3	4	5	6	7
\$GPGGA	hhmmss s.ss	1111.1 1	a	yyyyy.y y	a	x	xx
8	9	10	11	12	13	14	15
x.x	xxx	M	x.x	M	x.x	xxxx	*hh

где 1 – гринвичское время на момент определения местоположения;  
 2 – географическая широта местоположения;  
 3 – север/юг (N/S);  
 4 – географическая долгота местоположения запад/восток (E/W);  
 5 – индикатор качества GPS-сигнала: 0 – Определение местоположения невозможно или не верно; 1 – GPS-режим обычной точности, возможно определение местоположения; 2 – Дифференциальный GPS-режим, точность обычная, возможно определение местоположения;  
 6 – количество используемых спутников  
 7 – фактор ухудшения точности плановых координат (HDOP);  
 8 – высота антенны приемника над уровнем моря;  
 9 – единица измерения высоты расположения антенны, метры;  
 10 – геоидальное различие – различие между земным эллипсоидом WGS 84 и уровнем моря – уровень моря ниже эллипсоида;  
 11 – единица измерения различия, метры; 12 – возраст дифференциальных данных GPS – время в секундах с момента последнего обновления SC104 типа 1 или 9.

## 1.6.2 GPGLL

Сообщение содержит данные о географической широте, долготе и времени определения координат. Пример сообщения: \$GPGLL, 5532.8492, N, 03729.0987, E, 004241.469, A\*33. Расшифровка содержания приведена в таблице 1.2.

Таблица 1.2 – Содержания сообщения \$GPGLL

Тип сообщения	1	2	3	4	5	6	7
\$GPGLL	1111.11	a	yyyyy.yy	a	hhmmss.ss	A	*hh

- где 1 – географическая широта местоположения;  
 2 – север/юг (*N/S*);  
 3 – географическая долгота местоположения;  
 4 – запад/Восток (*E/W*);  
 5 – гринвичское время на момент определения местоположения;  
 6 – статус: *A* – данные верны, *V* – данные не верны;  
 7 – контрольная сумма строки.

### 1.6.3 GPVTG

Сообщение содержит информацию о направлении и скорости движения.  
 Пример сообщения: *\$GPVTG, 360.0, T, 348.7, M, 000.0, N, 000.0, K\*43*  
 Расшифровка сообщения приведена в таблице 1.7

Таблица 1.3 – Содержание сообщения *\$GPVTG*.

Тип сообщения	1	2	3	4	5	6	7	8	9
<i>\$GPVTG</i>	<i>x.x</i>	<i>T</i>	<i>x.x</i>	<i>M</i>	<i>s.ss</i>	<i>N</i>	<i>s.ss</i>	<i>K</i>	<i>*hh</i>

- где 1 – направление движения в градусах;  
 2 – относительно Северного полюса;  
 3 – направление движения в градусах (может не использоваться);  
 4 – относительно северного магнитного полюса (может не использоваться);  
 5 – скорость;  
 6 – единица измерения скорости, узлы;  
 7 – скорость;  
 8 – единица измерения скорости, км/ч;  
 9 – контрольная сумма строки.

Проведя анализ раздела обзора литературы, были получены выводы свидетельствующие, о полном изучении преимуществ, свойств и специфики измерения скорости посредством технологии GPS. Было приведено сравнение достоинств и недостатков иных способов измерений.

## 1.7 Обзор архитектуры AVR

Контроллер является ядром системы, именно он управляет драйверами устройств индикации, получает сообщения от GPS модуля и обрабатывает их, а также исполняет основной алгоритм программы, записанный в его флеш-память.

Ввиду того, что необходимо использовать дешевую, малогабаритную, и в тоже время крайне распространенную архитектуру, выбор был сделан в пользу семейства AVR.

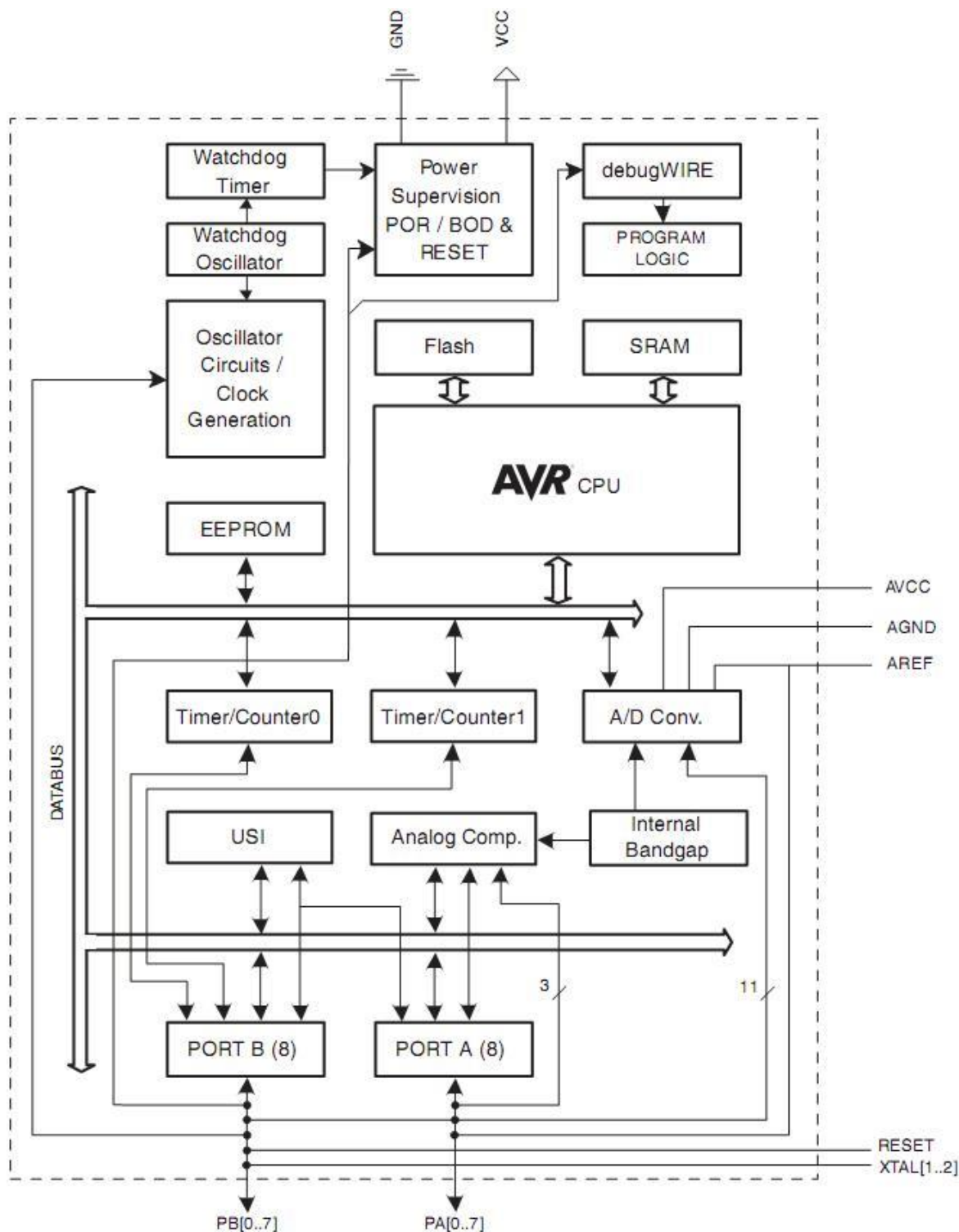


Рисунок 1.2 – Архитектура ATtiny865 [7]

Наибольшим преимуществом семейства AVR – это наличие 32 оперативных регистров. Несмотря на их разные в правах доступа, это позволяет не обращаться в оперативную память и стек. На этом фоне семейство PIC крайне малоприспособлены для конструирования и проектирования – они скорее больше «заточены» под несложные устройства с высоким

тиражированием. Контролеры от производителей ST и Texas Instruments будут слишком дорогими и избыточными [6]. Также имеет место простота использования в программировании данной системы, хотя больших отличий с более развитых систем компаний названных выше мало.

Рассмотрим архитектуру и принцип работы на примере микроконтроллера ATtiny865. Это 8-разрядный микроконтроллер на базе RISC архитектуры из семейства AVR с низким энергопотреблением, объединяющий 16 КБ программируемой флэш-памяти, 1 КБ SRAM, 512B EEPROM, 8-канальный 10-разрядный аналого-цифровой преобразователь и интерфейс JTAG для включения отладки. Устройство поддерживает пропускную способность 16 MIPS на 16 МГц и работает от 4,5 до 5,5 В. Выполняя инструкции за один такт, устройство достигает пропускной способности, приближающейся к 1 MIPS на МГц, балансируя энергопотребление и скорость обработки. Ядро AVR объединяет богатый набор команд с 32 рабочими регистрами общего назначения. Все 32 регистра напрямую подключены к арифметическому логическому блоку (АЛУ), что позволяет получить доступ к двум независимым регистрам в одной инструкции, выполняемой за один такт (Рисунок 1.2). Получающаяся архитектура более эффективна в отношении кода, обеспечивая при этом пропускную способность до десяти раз быстрее, чем у обычных CISC-микроконтроллеров. Обладает следующими характеристиками:

- 1) Усовершенствованная архитектура RISC - 2-тактный множитель на кристалле.
- 2) Высокопрочные, энергонезависимые сегменты памяти.
- 3) Интерфейс JTAG (совместимый с IEEE 1149.1).
- 4) Два 8-битных таймера / счетчика с отдельными делителями и режимами сравнения.
- 5) Один 16-битный таймер / счетчик с отдельным делителем, режимом сравнения и режимом захвата.
- 6) Сброс при включении питания (POR) и программируемое обнаружение отключения.
- 7) Внутренний калиброванный RC генератор.
- 8) Внешние и внутренние источники прерываний.

## 1.8 Интерфейс I<sup>2</sup>C

Интерфейс I<sup>2</sup>C был придуман и стандартизирован фирмой Philips. В терминологии Atmel, из-за патентной чистоты, этот интерфейс называется, как *Двухпроводной последовательный интерфейс* (Two-Wire Serial Interface, TWI) [8]. Это двухпроводная шина, состоящая из двух проводов для последовательной передачи данных *serial data* (SDA), и синхросигнала *serial clock* (SCL). Распределение ролей в данной системе классическое: *Ведущий* (Master) и *Ведомый* (Slave).

Master инициирует передачу данных по шине, Slave только отвечает на

запросы мастера и только по его вызову. Шина может работать с несколькими главными устройствами (Multi-Master), когда в каждый любой момент времени на шине активно только одно Master устройство. Однако чаще всего используют простую конфигурацию шины только с одним Master устройством, все остальные устройства работают как Slave (подчиненных устройств может быть одно или несколько).

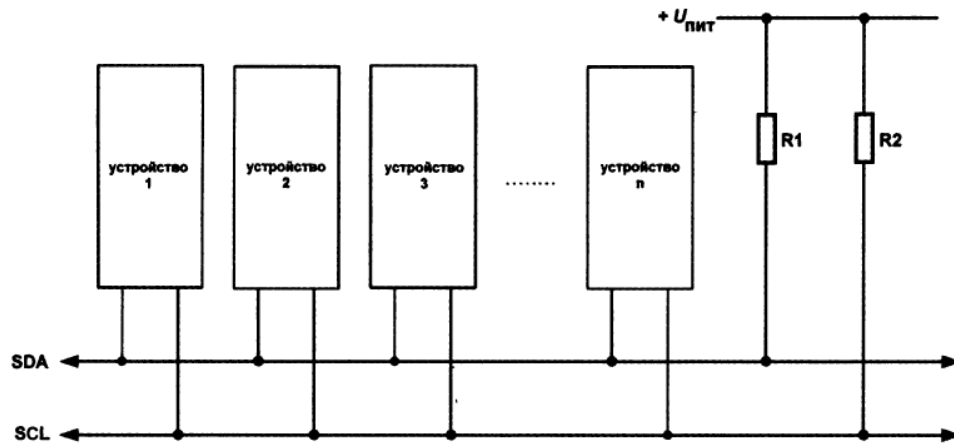


Рисунок 1.3 – Соединение устройств по интерфейсу I<sub>2</sub>C [9]

Все устройства на шине соединены друг с другом параллельно (Рисунок 1.3), используя драйверы с открытым стоком для формирования на шине цифровых сигналов. По этой причине на шине необходимо присутствие два верхних подтягивающих резистора. Они должны быть достаточно большие, чтобы не перегружать выходные стоки драйверов устройств, которые подключены к шине. Чем выше частота передачи, тем меньше номинал нагрузочных резисторов.

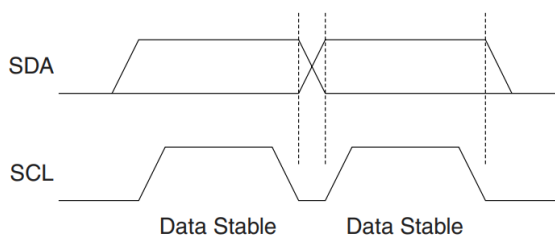


Рисунок 1.4 – Валидность данных [9]

Данные передаются по принципу MSB (Старшим разряд первый поступает на линию SDA). Каждая транзакция байта сопровождается START и STOP сигналами на линиях. Данные на линии SDA валидны в момент, когда на линии SCL высокий логический уровень (Рисунок 1.4).

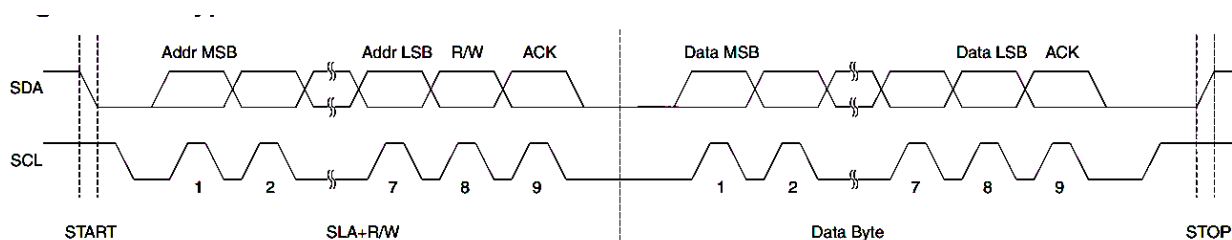


Рисунок 1.5 – Диаграмма передачи данных на ведомое устройство [9]

Для того, чтобы отличать подчиненные устройства на шине, используется 7-битный адрес (так изначально определила компания Philips), передаваемый как первый байт после сигнала START (Рисунок 1.5). Биты адреса передаются в старших семи битах байта. Младший бит LSB байта адреса имеет бит чтения/записи. В конце передачи адреса Мастер ждет подтверждение от ведомого. После его получения мастер начинает передавать байт данных, и последним битом, после квитирования отправленных данных, передается сигнал STOP.

## 1.9 Обзор аналогов

GPS-спидометр RL-SM001. Его внешний вид представлен на рисунке 1.6. Этот GPS-спидометр является компактным прибором, с минимальным количеством функций. Он способен отображать скорость, время, температуру. Его исполнение (цилиндр круглого сечения) подразумевает монтаж в



Рисунок 1.6 – GPS-спидометр RL-SM001

Источник изображения: <https://driveboat.by/katalog/rl-sm001.html>

приборную панель. Он имеет стандарт защиты IP65 и подсветку дисплея зеленого цвета. Но, однако это не позволяет использовать его портативно и

быстро на разных транспортных средствах и объектах, что дает ему большой минус.



Рисунок 1.7 – GPS-спидометр C60 Hud

Источник изображения: <https://www.pinterest.es/pin/776589529476809577/>

Для пользователя доступно меню прибора, где он может задать единицы измерения. Преимущества такого спидометра в том, что он компактен и имеет пылевлагозащитную конструкцию корпуса, что позволяет его устанавливать на катамараны, лодки и т.п. Из недостатков – малый набор измеряемых величин, принимающая сигнал антенна спрятана в корпусе, что снижает чувствительность.

Еще одним рассматриваемым аналогом является GPS-спидометр C60 Hud. Его внешний вид представлен на рисунке 1.7. Данный прибор имеет предназначение для измерения текущей скорости, высоту относительно уровня мирового океана, скорость в двух мерах (км/ч и м/ч), а также одометр. Большим минусом является отсутствие выносной антенны – она встроена в корпус как у и предыдущего аналога. А также время на поиск и анализ спутников.

Сравнив представленные аналоги, можно выделить, что в разрабатываемый в дипломном проектировании прибор, должен обладать большим количеством преимуществ, которыми являются:

- наглядное представление информации;
- наличие звукового оповещения;
- наличие выносной антенны, что позволяет обеспечить лучший прием для антенны GPS-приемника.



## 2 РАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ

После изучения теоретических основ и формирования полного представления о том, что должна делать система, можно перейти к проектированию на уровне блоков и соответственно, к построению структурной схемы. Схема структурная электрическая раскрывает структуру системы управления комплексом по измерению скорости объекта с точки зрения крупноблочного проектирования и приведена на чертеже ГУИР.400201.034 Э1.

Комплекс измерения скорости объекта включает в себя следующие модули:

- 1) Микроконтроллер.
- 2) Модуль GPS.
- 3) Источник питания.
- 4) Каскад согласования измеряемого напряжения.
- 5) Датчик освещенности.
- 6) Преобразователь напряжения.
- 7) Модуль индикации.
- 8) Часы реального времени.

Рассмотрим каждый из перечисленных модулей подробнее.

### 2.1 Микроконтроллер

*Микроконтроллер* является ядром системы, именно он управляет всеми остальными модулями и осуществляет управление системой в целом, путем исполнения прописанной в него программы. Он связан практически со всеми системами односторонней связью (кроме часов реального времени). Любые изменения на одном из блоков немедленно поступают на микроконтроллер, который непрерывно все это обрабатывает. И вносит изменения на блок индикации.

Модуль GPS соединен через последовательный USART интерфейс. После получения пакетов с информацией о местоположении, контролер, приняв эту информацию начинает обрабатывать и вносить изменения на блоке индикации. Также, необходимо предусмотреть возможность немедленно перепрограммировать контролер, без извлечения его из устройства.

Часы реального времени необходимы для реализации часов в данном устройстве. Для большей точности и внесения корректировок в их ход этот модуль соединен с микроконтроллером двусторонней связью. Для внесения изменений от полученных данных времени по спутнику, а затем отображение их на блок индикации.

Также контролер получает через определенный промежуток времени данные с каскада согласования преобразователя напряжения.

## **2.2 Модуль GPS**

*Модуль GPS*, используемый в составе прибора, служит приема сигнала от спутника GPS. Он принимает пакеты данных от спутников, и после анализа формирует пакеты, которые далее будут отправлены микроконтроллеру, посредством интерфейса UART. В модуле необходимо использовать активную антенну, для того чтобы позволить повысить чувствительность приемника к приему сигнала. Модуль выдает результаты в формате NMEA.

## **2.3 Источник питания**

*Источником питания* служит аккумуляторная батарея автомобиля на 11В – 16В. Она подключена напрямую к преобразователю напряжения и каскаду согласования измеряемого напряжения.

## **2.4 Каскад согласования измеряемого напряжения**

*Каскад согласования измеряемого напряжения* состоит из стабилизирующей и понижающей аппаратуры. Необходим для измерения текущего напряжения на источнике питания автомобиля через аналого-цифровой преобразователь микроконтроллера.

## **2.5 Датчик освещенности**

*Датчиком освещенности* необходим для корректировки яркости на блоке индикации. Этот модуль состоит из фоторезистора, подключенного к системе компараторов, сравнивающих текущий свет с эталонным значением, которое подтянуто от преобразователя напряжения. Полученный сигнал постоянно получает микроконтроллер и периодически опрашивая свои порты ввода/вывода, реагирует на изменения состояния этого модуля.

## **2.6 Модуль индикации**

*Модуль индикации* состоит из звукового устройства, матричного дисплея и трех жидкокристаллических OLED дисплеев. Электромагнитный излучатель звука используется для звуковой сигнализации каких-либо запланированных событий. OLED дисплей предназначен в устройстве для вывода текстовой информации. Подключено через последовательный интерфейс. Также есть отдельный матричный дисплей. На котором будет отображаться текущая скорость. Звуковое устройство будет использоваться как звуковой сигнализатор, дабы уменьшить воздействие глаз водителя и постоянно не отвлекать его.

## 2.7 Преобразователь напряжения

*Преобразователь напряжения* необходим для создания 5 вольт, которые используются для питания микроконтроллера, часов реального времени, модуля индикации и GPS-модуля. Совокупность этих частей является значительным потребителем энергии. Поэтому, применение линейных стабилизаторов не желательно, так как на них будет выделяться большое количество тепла, что внесет большую проблему при эксплуатации в автомобиле. Исходя из этого, необходимо установить импульсный понижающий преобразователь с большим значением КПД. Внутри он состоит из делителя напряжения с возможностью точной коррекции необходим для того, чтобы согласовать уровень значения напряжения питания с уровнем, обрабатываемым АЦП микроконтроллера. Это необходимо для корректного измерения входного уровня напряжения питания. Делитель задает соотношение 1:10. В нем предусмотрен подстроечный многооборотный резистор, сопротивлением которого выстраивается точность измерений, которая нарушается из-за разброса номиналов деталей.

## 2.8 Часы реального времени

*Часы реального времени* с последовательным интерфейсом – это малопотребляющие полные двоично-десятичные часы-календарь, включающие 56 байтов энергонезависимой статической ОЗУ. Адреса и данные передаются последовательно по двухпроводной двунаправленной шине. Часы-календарь отсчитывают секунды, минуты, часы, день, дату, месяц и год. Последняя дата месяца автоматически корректируется для месяцев с количеством дней меньше 31, включая коррекцию високосного года. Часы работают как в 24-часовом, так и в 12-часовом режимах с индикатором АМ/РМ. Часы реального времени имеют встроенную схему наблюдения за питанием, которая обнаруживает перебои питания и автоматически переключается на питание от батареи. Именно низкое потребление (порядка 500 нА) от батареи, при отключении внешнего питания наиболее лучше характеризует эти часы и их применение в схеме. Так как преимущественно большее время они будут отключены от питания (когда автомобиль не эксплуатируется). Для автономной работы этих часов, на плате предусмотрена посадочная колодка под элемент питания.

### 3 РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ СХЕМЫ

В данном разделе будет описана более подробно, с функциональной точки зрения, ранее спроектированная в предыдущем разделе структурная схема аппаратно-программного комплекса измерения скорости.

Схема функциональная электрическая раскрывает функционал системы управления комплексом по измерению скорости объекта с точки зрения алгоритмов и систем взаимодействия и приведена на чертеже ГУИР.400201.034 Э2.

Комплекс измерения скорости объекта включает в себя следующие модули:

- 1) Микроконтроллер.
- 2) Модуль GPS.
- 3) Источник питания.
- 4) Каскад согласования измеряемого напряжения.
- 5) Датчик освещенности.
- 6) Преобразователь напряжения.
- 7) Модуль индикации.
- 8) Часы реального времени.

#### 3.1 Микроконтроллер

Микроконтроллер является устройством, которое принимает, обрабатывает и отправляет сигналы от и на внутренние блоки и модули. В результате осуществляется управление и функционирование всей системы.

Микроконтроллер должен содержать в себе следующие функции:

- сбор цифровых данных с периферийных модулей;
- сбор аналоговых данных с преобразователей напряжения и периферийных аналоговых датчиков;
- анализ принятых данных (аналоговых и цифровых), их вычисления и преобразования;
- формирование сигнала для отправки вычисленных сигналов на периферийные устройства в виде полезной информации;
- передача данных по интерфейсу с использованием внутренних средств обработки с данными интерфейсами;
- передача команд исполнительным устройствам;
- тактирование периферийных устройств.

Изображенная блок схема микроконтроллера на рисунке 3.1 включает в себя следующие подсистемы:

- Подсистема ввода/вывода *General Purpose Input/Output (GPIO)*;
- Вычислительное ядро (*core*);
- Аналогово-цифровой преобразователь (*Analog Digital Converter ADC*);

- Подсистему устройства последовательной асинхронной передачи и приема данных (universal asynchronous serial receiver and transmitter – далее *UART*);
- Двухпроводной последовательный интерфейс *I2C*;
- Последовательный периферийный интерфейс (serial peripheral interface – далее *SPI*);
- Запоминающие устройства энергонезависимые (*SRAM*), энергонезависимые (*EEPROM*) и перепрограммируемые (*FLASH*);
- Подсистема прерываний (Interrupt unit – далее *INT*);
- Подсистема тактирования (oscillator – далее *OSC*).

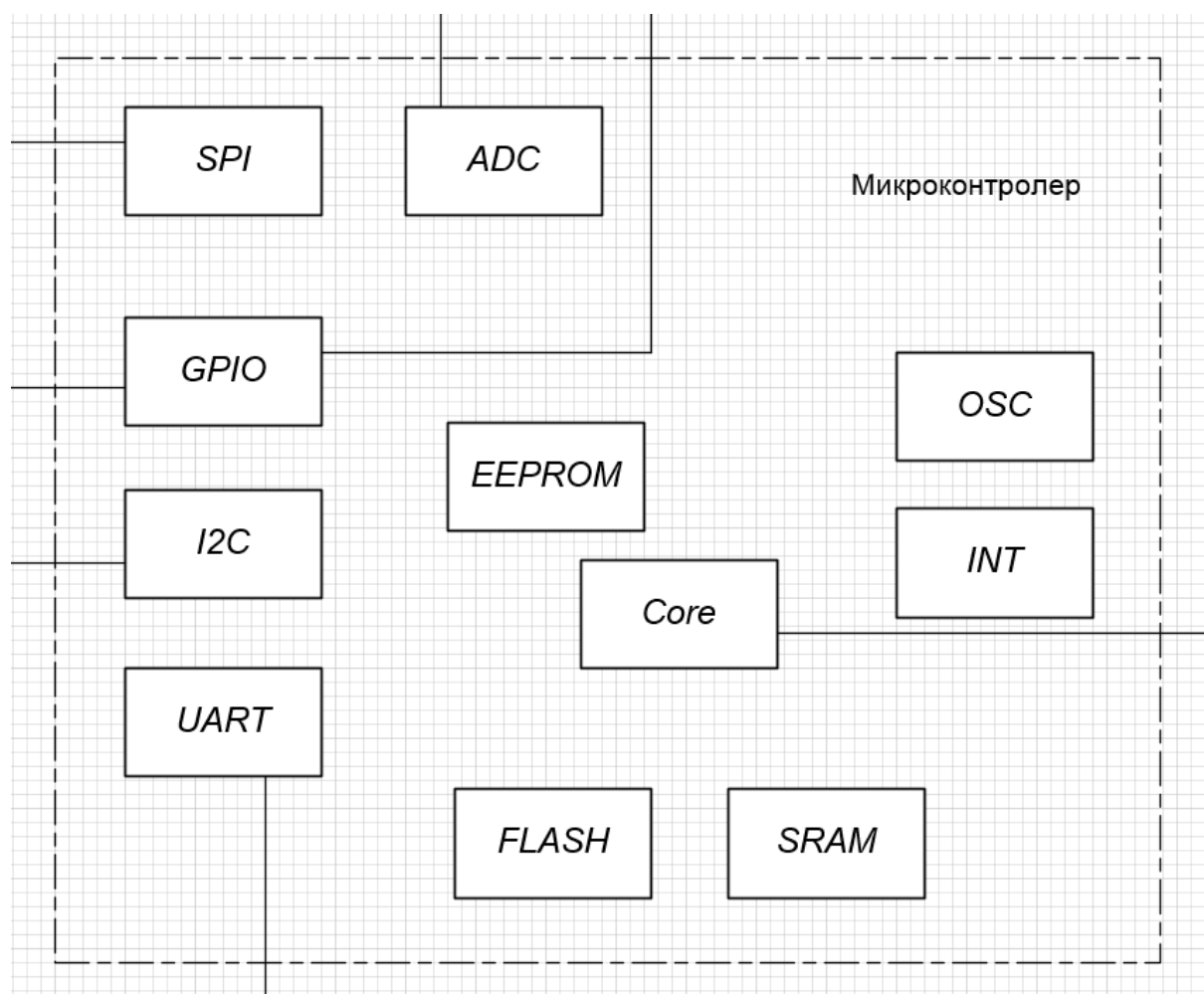


Рисунок 3.1 – Блок-схема микроконтроллера

Ниже будет более подробно рассмотрена применение и назначение каждой указанной подсистемы в данном проекте.

### 3.1.1 Подсистема ввода/вывода **GPIO**

Все порты имеют настоящую функциональность Read-Modify-Write при

использовании в качестве общего цифрового порта ввода/вывода. Это означает, что направление одного вывода порта может быть изменено без непреднамеренного изменения направления любого другого вывода с помощью инструкций SBI и CBI.

То же самое относится к включению или выключению подтягивающих резисторов (которые необходимы для работы в режиме приема). Каждый выходной буфер имеет симметричный диск характеристик как с высокой нагрузкой, так и с возможностями управлять светодиодными дисплеями напрямую. Все выводы порта должны иметь индивидуально выбираемые подтягивающие резисторы с инвариантным сопротивлением напряжения питания. Все выводы ввода/вывода должны иметь защитные диоды к обоим выводам питания Vcc и заземление, как показано на рисунке 3.2.

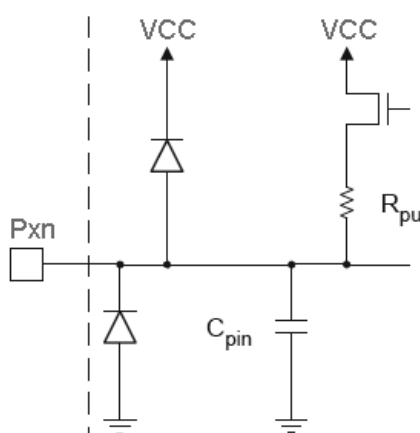


Рисунок 3.2 – Схема портов GPIO

Все регистры и битовые ссылки в этом разделе записаны в общем виде. Строчные буквы «X» обозначает букву нумерации порта, а строчная буква «n» обозначает бит число. Однако при использовании регистра или бита в программе указывается точная форма необходимо использовать. PORTB3 для бита 3 в порте B, документально, как правило, PORTxn.

Три адреса памяти ввода-вывода выделены для каждого порта:

- регистр данных – PORTx;
- регистр направления данных DDRx;
- входные контакты порта PINx.

Расположение ввода/вывода входных контактов порта доступно только для чтения, в то время как регистр данных и направление данных регистр для чтения/записи.

Принимая и передавая данные и сигналы с подсистемы ввода/вывода, мы можем получать их для анализа, и для проведения необходимых вычислений в ядре, а также для управления иными периферийными устройствами.

Также, в подсистеме GPIO существует возможность подключать

альтернативные функции. Это необходимо для использования иных внутренних периферийных блоков напрямую, в обход стандартных подсистем приема и буфера ввода/вывода.

### 3.1.2 Вычислительное ядро

Вычислительное ядро представляет из себя главный компонент микроконтроллера. Именно он занимается вычислительной мощностью всего комплекса. Преимущественно состоит из арифметико-логического устройства (АЛУ), общих регистров назначения (РОН), регистров статуса (SR), регистра инструкций, счетчика команд (program counter), стека и декодера команд.

Программный счетчик, устанавливается в ноль, тем самым указывая на начальный адрес инструкций, записанных в память FLASH, находящуюся вне ядра. Прочитав оттуда первую инструкцию, она сохраняется в регистре инструкций и декодируется в декодере команд. Когда устройство управления распознало команду, начинает ее выполнять, задействуя регистры общего назначения и АЛУ. результат вычислений и манипуляций выносится на системную шину, расположенную вне ядра. Далее эта информация попадает на внутреннюю периферию или в оперативное запоминающее устройство (ОЗУ).

### 3.1.3 Аналогово-цифровой преобразователь

Аналого-цифровое преобразование — это процесс преобразования входной физической величины в ее числовое представление. Аналого-цифровой преобразователь или АЦП (от англ. *analog to digital converter*, сокращенно *ADC*) — это устройство, выполняющее такое преобразование. Формально, входной величиной АЦП может быть любая физическая величина — напряжение, ток, сопротивление, емкость, частота следования импульсов, угол поворота вала и т.п. Однако, для определенности, в дальнейшем под АЦП мы будем понимать исключительно преобразователи напряжение-код.

АЦП имеет множество характеристик, из которых основными можно назвать частоту преобразования и разрядность. Частота преобразования обычно выражается в отсчетах в секунду (*samples per second*, SPS), разрядность — в битах. Современные АЦП могут иметь разрядность до 24 бит и скорость преобразования до единиц GSPS (конечно, не одновременно). Чем выше скорость и разрядность, тем труднее получить требуемые характеристики, тем дороже и сложнее преобразователь. Скорость преобразования и разрядность связаны друг с другом определенным образом, и мы можем повысить эффективную разрядность преобразования, пожертвовав скоростью.

На ADC микроконтроллера поступает опорный сигнал, равный  $V_{cc}$ . С каскада согласования измеряемого напряжения поступает сигнал равный некой неизвестной величине, меньшей чем  $V_{cc}$ . Далее они сравниваются

аналоговым компаратором и компаратором и получившиеся кривая раскладывается на бинарные значения специальным кодером. Величина полученного значения будет зависеть от разрядности этого конвертера.

Далее полученное значение будет программно изменено в реальное путем пропорционального соотношения и выведено на знаковосинтезирующий экран.

### 3.1.4 UART

Последовательным асинхронным приемопередатчиком является интерфейс, который использует одну линию. подключение с периферийными устройства по этому интерфейсу осуществляется скрещено.

Основные рабочие линии – RXD и TXD, или просто RX и TX. Передающая линия – TXD (*Transmitted Data*), а порт RXD (*Received Data*) – принимающая.

Данная подсистема обычно разделяют три основные части:

- тактовый генератор;
- передатчик;
- приемник.

Передачик состоит из одного буфера записи, регистра последовательного сдвига, генератора четности и логики управления для обработки различных форматов последовательных кадров. Буфер записи позволяет осуществлять непрерывную передачу данных без задержки между кадрами. Приемник является наиболее сложной частью модуля UART благодаря своим модулям синхронизации и восстановления данных. Блоки восстановления используются для асинхронного приема данных. В дополнение к блокам восстановления, приемник включает проверку четности, логику управления, регистр сдвига и двухуровневый буфер приема.

В данном проекте будет использоваться 1 стоп бит в кадре, а размер информации 8 бит.

В данном комплексе будет использоваться только прием информации, поэтому будет задействован только входной буфер приема данных. Данные последовательно поступают на сдвиговый регистр с TXD линии, после которого при появлении стоп бита защелкивается и загружается в регистр приема. После этого в регистре статуса UART устанавливается флаг, который говорит о том что данные получены. После прочтения появится флаг что буфер пуст, и следующее принятое сообщение будет загружено в приемный буфер. Но так как скорость этого интерфейса в разы меньше скорости обработки внутри контролера, большую часть времени контролер будет простаивать на приеме. Будет выгоднее по затратам времени пользоваться прерываниями этого подсистемы, которые срабатывают по флагу заполняемости буфера.

Также необходимо определить скорость работы данной подсистемы.



Вычисление коэффициента, который позже будет использован при настройке контролера, находится по формуле (3.1):

$$UBBB = \frac{f_{osc}}{(16 \cdot BAUD)} - 1 \quad (3.1)$$

где UBBB – коэффициент, используемый для конфигурации подсистемы UART;

$f_{osc}$  – частота тактирования контролера, которая равна, МГц;

BAUD – скорость с которой модуль GPS передает контролеру, бод/с.

Воспользовавшись формулой (3.1), и зная, что частота тактирования контролера  $f_{osc} = 8$  МГц, а скорость приема данных должна быть  $BAUD = 9600$  бод/с, следовательно, коэффициент  $UBBB = 51$ .

### 3.1.10 Выбор архитектуры

После рассмотренных всех необходимых обязательных внутренних подсистем и внутренних модулей необходимо выбрать какое семейство использовать микроконтроллеров из микроархитектур гарвардского типа RISC. Кроме всех вышеуказанных требований, также необходимое требование — это размер памяти. Ввиду того, что в таких микроархитектурах адресное пространство разделено (код и данные в разных областях памяти) необходимо выбрать именно так, чтобы памяти хватало на выполнение поставленных целей, и в тоже время, контролер не должен быть перенасыщен внутренними периферийными устройствами. По распространенности и простоте программирования лучше всех подойдет семейство AVR, которое был описано в разделе 1.7.

## 3.2 Модуль GPS

Модуль GPS представляет из себя устройство, состоящее из антенны приемника, устройства управления, обработчика сигналов и контролера передачи данных. Оно осуществляет прием и анализ данных, полученных от приемников. Также, именно этот модуль осуществляет подключение и производит аналитическо-вычислительные процессы, которые в дальнейшем формируются в пакет данных, поступающих на микроконтроллер.

Навигационные ИСЗ постоянно знают свои координаты, а также вместе с полезным сигналом излучают его на закодированной частоте  $L1$  и  $L2$ . При перехвате модулем этого сигнала, он дешифруется (только сигнал с частотой  $L1$  – дешифровка  $L2$  невозможна на гражданских модулях, так как он закодирован Р-кодом и имеет военное предназначение), и вносится в выделенную устройством управления область памяти модуля.

Принимающая антенна модуля принимает поступающие от ИСЗ

закодированные сигналы C/A кодом на частоте  $L1$ . Приняв и обработав сигнал от первого найденного спутника, осуществилось позиционирование приемника относительно спутника. Принятый сигнал высчитывает время приема, и сравнивая с временем передачи сообщения узнает время прохождения сигнала от спутника до приемника. Однако полученное расстояние, равное произведению скорости света (299 792 км/с) на полученное время дает расстояния от спутника до приемника, притом с большой погрешностью! Но неизвестно остается направление этого расстояния. Найден лишь радиус круга, на концах которого находится приемник (ось  $X$ ) относительно спутника. Модуль продолжает поиск второго спутника из другой орбиты, для более точного позиционирования на плоскости  $X$ - $Y$ .

При перехвате сигнала от второго спутника с другой орбиты, приемник фиксирует номер спутника, его орбиту, и далее производит вычисления точно такие же которые были произведены с первым найденным спутником. И также с большой погрешностью. Однако, направление уже известно. Найден еще один радиус круга, на концах которого находится приемник (ось  $Y$ ). Пересечение радиусов осуществляется в одной точке. Именно в этой точке и находится приемник.

Приемник продолжает поиск третьего ИСЗ. Все это необходимо для того, чтобы можно было позиционировать приемный модуль относительно координат этих спутников также и по высоте (ось  $Z$ ). Произведя такие же процедуры приема и расчета расстояния, как было сделано с двумя предыдущими спутниками, будет найдена высота относительно уровня моря. Итак, будет позиционировать в трех плоскостях положение приемного модуля относительно трех спутников. Однако, остается нерешенным проблема правильности данных измерений. Поэтому, необходим четвертый спутник, который поможет приемнику скорректировать текущее время по Гринвичскому часовому поясу, по причине наличия на каждом ИСЗ атомных часов.

После установления связи с четвертым спутником происходит прием данных о текущем времени от четвертого спутника. Но так как на прием и расшифровку сообщения уходило время, принятое значение также окажется некорректным. Поэтому в течении некоторого времени производится прием всех сообщений и производится расчет по специальной формуле. Найденные коэффициенты отслеживаются с течением времени и если в течении этого времени результат не меняется – то приемник находит специальный коэффициент, который и влияет на погрешность. И далее он вносит его в принятое время, и далее истинное значение текущего времени найдено.

Далее, после нахождения текущего времени, оно используется для вычисления расстояний до трех первых найденных спутников. И вот теперь они являются истинными.

После получения всех данных от спутника начинается формирование пакета, который будет отправлен микроконтроллеру по последовательному USART порту. Модуль передает пакеты постоянно, однако в процессе поиска

и вычислений всех истинных параметров он непрерывно отправляет пакеты, содержащие в себе пустые поля (или поля заполнены специальными знаками, как это реализовано в основном GPS протоколе NMEA, предназначенном для передачи данных от приемника к вычислительному устройству). Далее сформированные пакеты, передаются по последовательному порту к контролеру (на порт приема RX). Данные, передаваемые модулем GPS, и принимаемые микроконтроллером представляют из себя двоичные сигналы, равные одному байту, и передающие определенное количество байт за единицу времени. Контролер воспринимает эти байты информации как текст, зашифрованный в ASCII код. Далее микроконтроллер сохраняет принятые байты информации, переданные модулем, идентифицирует, и начинает читать и обрабатывать полученную информацию. Структура принимаемых пакетов описана в подразделе 1.6.

### **3.3 Источник питания**

Источником питания является автомобильный бортовой аккумулятор, или иной источник питания, с диапазоном напряжения от 4 до 16 вольт. Несмотря на то, что преобразователь напряжения будет рассчитана на напряжение входное от 4 до 40 вольт, это создаст нагрузку и некоторые проблемы на модуле аналогово-цифрового преобразователя микроконтроллера.

### **3.4 Каскад согласования измеряемого напряжения**

Каскадом согласования измеряемого напряжения необходим для уменьшения напряжения, необходимого для измерения на аналогово-цифровом преобразователе. Это связано с тем, что опорное напряжение будет равно напряжению питания микроконтроллера. Ведь аналого-цифровой преобразователь не сможет принять настолько большое напряжение и разложить его.

Каскадом согласования измеряемого напряжения служит входной каскад, состоящий из элемента сглаживая уровня напряжения и обратной защиты, и регулируемый делитель напряжения, который уменьшает напряжение, подаваемое с бортового источника питания до величины опорного напряжения. Получение напряжение будет подаваться на аналогово-цифровой преобразователь (АЦП) контролера, которое будет раскладываться в сравнении с опорным напряжением.

На входном каскаде необходимо использовать емкость, которая будет сглаживать колебания, возникающие на источнике питания, ввиду того что он может, например, при работающем двигателе автомобиля заряжаться, тем самым напряжение на его концах повысится.

### 3.5 Датчик освещенности

Датчиком освещенности служит для анализа окружающего света и обработка полученных сигнала от светочувствительного прибора, относительно некоторого опорного сигнала.

Датчик освещенности должен будет иметь возможность определять уровень освещенности извне корпуса комплекса и передавать сигнал на порт общего назначения (а именно ввода/вывода) микроконтроллера и затем обрабатываться им. Допустимое питание датчика освещенности должно входить в диапазон от 1.8 до 5 вольт.

Датчик освещенности обычно состоит из светорезистивного элемента с нагрузочным элементом, а также устройства сравнения – компаратора. Также необходим развязка для работы с компаратором. В качестве нагрузки компаратора можно использовать любую нагрузку с током потребления не более 50 мА. Это могут быть непосредственно обмотки реле, резисторы, светодиоды индикации и оптронов исполнительных устройств, с ограничивающими ток резисторами. Индуктивные нагрузки желательно шунтировать диодами от обратного выброса напряжения. Напряжение питания компаратора может быть от 5 до 36 вольт однополярного (или сумма двухполярного) напряжения. Ввиду того, что сопротивление светорезистивного элемента не фиксировано, то необходимо использовать аналоговый компаратор.

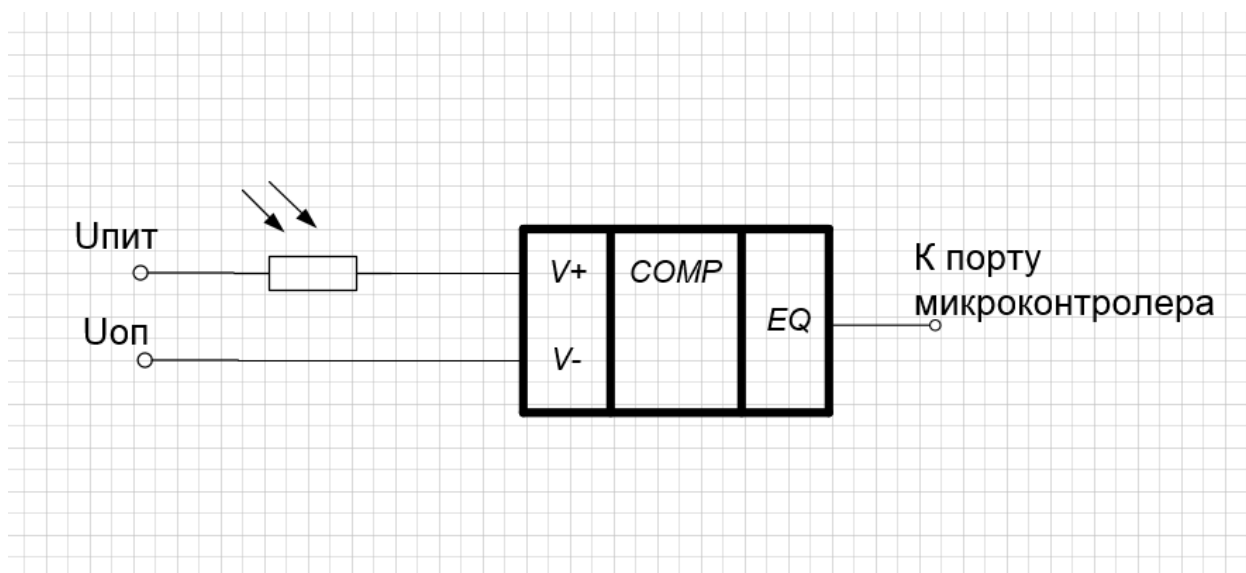


Рисунок 3.3 – Подключение аналогового компаратора.

Классическая схема подключения аналогового компаратора с коллектором представлена на рисунке 3.3.

Чтобы иметь устойчивый сигнал, сигнализирующий об освещенности бинарной величиной (логическим нулем или единицей), в зависимости от

степени освещенности фоторезистора, нам необходимо установить порог переключения. Для этого служит не инвертирующий вход ( $V+$ ) на который необходимо подать опорное (неизменяемое) напряжение. Это опорное напряжение будет взято от преобразователя напряжения и равно напряжению питания всей схемы.

Компаратор будет сравнивать два уровня напряжения (на выводах  $V+$  и  $V-$ ). Если напряжение на входе  $V-$  будет больше чем на входе  $V+$ , то на выходе данного модуля будет установлена логический ноль, который в свою очередь будет сигнализировать микроконтроллеру об изменении интенсивности света. Как только напряжение на входе  $V-$  опустится (при освещении фоторезистора) ниже уровня напряжения на входе  $V+$ , на выходе появится положительный потенциал и установится логическая единица, которая также будет сигнализировать о том, что необходимо произвести некие действия. В данном проекте, это будет использоваться для настройки яркости на модуле индикации (а конкретнее – в матрице светодиодной).

### **3.6 Преобразователь напряжения**

Преобразователь напряжения служит для трансформации входного напряжения в необходимые величины.

Данный функциональный блок должен обеспечивать питанием все датчики и устройства, подключенные к микроконтроллеру напрямую. При рассмотрении документации было установлено, что все датчики и модули, необходимые для работы исполнительных устройств могут работать от 5 вольт. Следовательно, питание, которое будет приходить извне должно быть преобразовано до 5 вольт.

Преобразователь напряжения внутри состоит из входного каскада, преобразователя напряжения и выходного каскада.

Наиболее подходящим вариантом является схема преобразователя, реализованная на импульсном преобразователе. Выбор данной системы связан именно с тем, что линейные преобразователи имеют малую КПД, а также требуют дополнительную систему охлаждения, что будет влиять на габариты.

### **3.7 Модуль индикации**

Модуль индикации — это комплекс устройств, необходимых для сигнализации пользователя о производимых событиях и отображении результата вычислительного процесса аппаратно-программного комплекса.

В состав модуля входит звуковая и знакосинтезирующая индикация. Знакосинтезирующая индикация состоит из трех органических светодиодных дисплеев и одной светодиодной матрице.

### 3.7.1 Звуковая индикация

Звуковая индикация – это генерация звука, сигнализирующая о каком-то происходящем событии. В данном проекте, этими событиями являются:

- подача питания на прибор;
- нахождение необходимого для навигации числа спутников;
- потеря связи со спутника;
- отсутствие спутников.

Отличительной особенностью каждого события будет разная длительность и генерация периодических звуковых сигналов. Генерация разной длины и частоты осуществляется с помощью подачи на определенный порт логической единицы командой `PORTx |= (1<<Pxy)` или отключение, путем подачи логического нуля командой `PORTx &= ~(1<<Pxy)`, и задержка, вызываемая функцией `_delay_ms`, определенной в библиотеке `delay.h`. Эта функция ничего не возвращает, но принимает следующий параметр:

1. `double __ms.`

Аргумент `__ms` имеет тип `double` и является временем задержки выполнения текущего кода. Значение `__ms` ограничена в размерности типа данных с плавающей запятой.

### 3.7.2 Знакосинтезирующая индикация

Знакосинтезирующая индикация – это генерация изображений на средствах, предназначенных для знакоотображения. Такой информацией, которую необходимо отображать, является:

- текущее время (часы и минуты с разделительным символом двоеточия, который мигает с частотой 0,5 Гц);
- текущая скорость в километрах за час (км/ч);
- количество подключенных спутников;
- напряжение аккумуляторной батареи автомобиля в вольтах.

Для большей эффективности, эргономичности и стоимости проекта, был сделан выбор в пользу комбинированной индикации. Отображение скорости является приоритетной, поэтому ее отображать будет знакосинтезирующая светодиодная матрица. А прочая, то есть второстепенная информация, будет отображаться на трех органических светодиодных экранах и большим разрешением. Рассмотрим каждый поподробнее.

#### 3.7.2.1 Знакосинтезирующая светодиодная матрица

Знакосинтезирующая светодиодная матрица представляет собой набор светодиодов, включенных параллельно с общим катодом или анодом. Стандартная матрица имеет 64 светодиода на одном сегменте, квадратной формы с восьмью светодиодами на каждой стороне. Более оптимальный для

человеческого глаза цвет, который бы не раздражал и не слепил водителя в темноте, был выбран именно красный. Также, яркость будет обеспечиваться коррекцией интенсивности путем отправки команд с микроконтроллера, который в свою очередь получает и обрабатывает информацию с датчика освещенности. При темноте интенсивность минимальная, однако при дневном свете полная яркость. Регистр управления, отвечающий за яркость определен как `INTENSITY` и равен шестнадцати битовому адресу `0x0A00`. Младшая половина адреса указывает на принадлежность к командам и необходимости записи в регистр управления, старшая половина адреса указывает собственно адрес регистра, который управляет интенсивностью матричного сегмента. Для организации управлением интенсивности матричным модулем используется функция `SetIntensity`, возвращающее пустое значение, и принимающая в качестве параметра аргумент типа `uint8_t` который определен в стандартной библиотеке `stdint.h` языка C. Ниже приведен текст этой функции:

```
void SetIntensity(uint8_t a)    // 0 down to 15
{
    SendLed((INTENSITY >> 8), (SHUTDOWN | a));
}
```

Аргумент `a` это интенсивность сегмента, способное принимать значение от минимального `0x00` до максимального `0x0F`.

Добавление данных будет осуществляться путем записи в драйвер сегментов матрицы данных, которые будут записаны в регистр данных. Под данными, в свою очередь, подразумевается запись в ряд необходимое состояние всех светодиодов в этом ряду. Представление осуществляется в бинарном (8 битном) виде. Единица соответствует состоянию, когда светодиод загорелся, ноль – выключился светодиод. Для организации управлением интенсивности матричным модулем используется функция `WriteNum`, возвращающее пустое `void` значение, и принимающая в качестве параметра следующие аргументы:

- `char* z`;
- `char* y`;
- `char* x`.

Аргумент `z` является старшей частью трехзначного числа, имеет тип `char*` и является указателем, который возвращает функция `pgm_read_byte`, который принимает в виде параметра указатель типа `static const char c` атрибутом `PROGMEM`, из файла `pgmspace.h` на массив символа, сохраненного в энергонезависимой памяти `FLASH`. Может принимать значения от `EMPTY` и `0` и до `9`. Аргумент `y` является средней частью трехзначного числа, имеет тип `char*` и является указателем, который возвращает функция `pgm_read_byte`, который принимает в виде параметра указатель типа `static const char c` атрибутом `PROGMEM`, из файла `pgmspace.h` на массив символа, сохраненного в

энергонезависимой памяти FLASH. Может принимать значения от EMPTY и 0 и до 9. Аргумент `x` является младшей частью трехзначного числа, имеет тип `char*` и является указателем, который возвращает функция `pgm_read_byte`, который принимает в виде параметра указатель типа `static const char c` атрибутом `PROGMEM`, из файла `pgmspace.h` на массив символа, сохраненного в энергонезависимой памяти FLASH. Может принимать значения от EMPTY и 0 и до 9. Ниже приведен пример кода на языке C:

```
void WriteNum(char *z, char *y, char *x)
{
    for(int i = 0; i < 8; i++)
    {
        PORTB &= ~(CS);

        SPI_WriteByte(i + 1);
        SPI_WriteByte(pgm_read_byte(&(z[i])));

        SPI_WriteByte(i + 1);
        SPI_WriteByte(pgm_read_byte(&(y[i])));

        SPI_WriteByte(i + 1);
        SPI_WriteByte(pgm_read_byte(&(x[i])));

        PORTB |= CS;
    }
}
```

Наличие цикла `for` соответствует записи в восемь информационных регистров состояния светодиодной линии в сегменте. Ввиду того, что большинство драйверов светодиодных матриц внешний интерфейс реализован в виде интерфейса *Serial Peripheral Interface* (SPI), то при его использовании возникнет необходимость выделения для каждого сегмента отдельного вывода с микроконтроллера, который бы управлял бы по отдельности записью (по концепции SPI это вывод *Chip Select* (CS) на принимаемом устройстве). Наличие одновременной записи сразу трех знаковимволов соответствует решению, принятому с целью экономии выводов контроллера и количества дорожек. В итоге, на светодиодную матрицу было задействовано три вывода вместо пяти (без учета двух выводов для питания).

Запись осуществляется через систему интерфейса SPI в режиме *Master* (ведущий) и *Slave* (ведомый). Ведущим в данной ситуации является микроконтроллер, который управляет всем аппаратно-программным комплексом, а ведомым является драйвер светодиодной матрицы, состоящий из параллельно подключенных драйверов, каждый из которых представляющий из себя шестнадцатибитный сдвиговый регистр, соединенные в один единый 48-битный сдвиговый регистр, каждый отдельный драйвер из



которых управляет отдельным сегментом светодиодной сборки, состоящей как, было сказано выше, из 64 светодиодов в одной сегменте.

При выборе элементной базы (а именно микроконтролера) стоит учесть то, что знакосинтезирующая матрица должна будет генерировать цифры в одном сегменте (от 0 до 9 включая пустое поле), символы (G, P, S), при условии, что каждый символ для знакосинтезирующей матрицы имеет размер 8 байт. Итого, для матричного дисплея необходимо зарезервировать место в FLASH памяти 14 байт под цифры и символы.

### **3.7.2.2 Знакосинтезирующие органические светодиодные экраны**

Знакосинтезирующими органическими светодиодными экранами (*organic light-emitting diode*, далее OLED) называются полупроводниковые приборы, изготовленные из органических соединений, эффективно излучающих свет при прохождении через них электрического тока. Управление дисплея осуществляется с помощью специального драйвера, представляющего из себя устройство управления, устройство ввода/вывода и дампа памяти, каждая ячейка которой соединена с элементами OLED дисплея.

Таким параметром, как разрешения дисплея, считается в данном случае количество OLED элементов на дисплее, а также размер дампа памяти, способного работать с данным дисплеем. Размеры дисплеев сильно варьируются, и потому чтобы не терять разрешение или наоборот не увеличивать его тем самым удорожая его, принято делать драйверы под определенный дисплей как абсолютно нелогичное и затратное занятие. Чтобы найти компромисс в данной ситуации, было найдено простое решение: делать дампы памяти дисплея кратным размерам стандартных дисплеев, а использовать его с дисплеями меньшей размерности. Конечно, это повлечет за собой большие неудобства при разработке и работе с такими дисплеями, однако на сегодняшний день это единственное решение. Но разработчики добавили гибкую управляемость адресным пространством, что дает возможность назначать программному указателю драйвера начальное смещение относительно начала памяти, тем самым приняв его за начальное значение дампа. Также есть драйверы с расширенным функционалом, например, управление интенсивностью свечения, инверсия изображения путем инверсии данных во всех ячейках памяти, а также включение и отключение OLED дисплея. В данном проекте самым подходящим будет разрешение 128\*32, соответственно дамп памяти будет не меньше 8192 ячеек.

Управлением дампом памяти осуществляется устройством управления, в лице которого является драйвер дисплея. Также, оно конфигурирует и управляет настройками отображения данных, переданного в дамп. Данные, получаемые драйвером, делятся на команды и данные. Данные имеют размер двух байт, первый байт является идентификатором данных, второй байт собственно полезной информацией. Такая раздельная передача вызвана тем, что большинство регистров передачи данных с микроконтроллера из

семейства AVR, имеют разрядность восемь бит. Основные команды определены в файле `OLED.h` в виде макросов.

В связи с тем, что дамп памяти имеет размер 8192, а необходимый размер ячеек 4096, значит имеет значение использование половины дампа. Важно это помнить при написании программного кода, так как можно ошибиться и записать область памяти, которая не будет отражаться на дисплее (невидимая область памяти).

Также, важной проблемой является использование трех дисплеев на интерфейсе I2C. Ввиду того, что они имеют одинаковые адреса, а зачастую на микроконтроллерах имеется только одно устройство I2C, возникает проблема индивидуального использования. Так как по правилам подключения нескольких устройств на шину I2C, подключение производится параллельно. Необходимо добавить устройство демультимплексирования или проще говоря – устройство управлением потоком данных, которое будет переключать информационный и синхронизирующий потоки к необходимому дисплею.

При выборе элементной базы (а именно микроконтроллера) стоит учесть то, что OLED экран должен будет генерировать цифры (от 0 до 9 включая символ двоеточия), значки и эмблемы (в количестве четырех), при условии, что каждый символ для имеет размер 100 байт. Итого, для матричного дисплея необходимо зарезервировать место в FLASH памяти 1 килобайт под цифры и около 2 килобайт под символы, при условии, что каждый занимает 100% дисплея при отрисовке.

### 3.8 Часы реального времени

Часами реального времени называют модуль, который отсчитывает текущее время, а также имеет возможность настройки и считывания данных.

Модуль часов реального времени состоит из основного устройства реального времени (*real time clock*, далее *RTC*), тактового генератора, долговременного памяти и резервного питания.

На рисунке 3.4 предоставлена блок схема RTC, которая представляет из себя устройство, состоящее из блоков тактирования (*oscillation block*), управления памятью (*power control*), шины интерфейса (*serial bus interface*), устройством управления (*control logic*), внешнего тактирования (*mux/buffer*), память *RAM*, часы/календарь и регистры управления (*control register and calendar*). Шина интерфейса является основным связующим с микроконтроллером. Команды идут в обход основного устройства управления (это чем-то походе на работу как в DMA). Память используется для хранения записываемой информации, и имеет зависимость от энергопитания. Часы/календарь и регистры управления служат для занесения команд и данных, а также в их формировании. В формировании участвует блок тактирования, являющийся по сути счетчиком отсчитывающий каждую секунду, и способный формировать сигнал на выходе с частотой 1 Гц. С учетом распространенных номиналов такой счетчик должен быть 15-ти

разрядным, чтобы за одну секунду он точно мог посчитать от 0 до 32767.

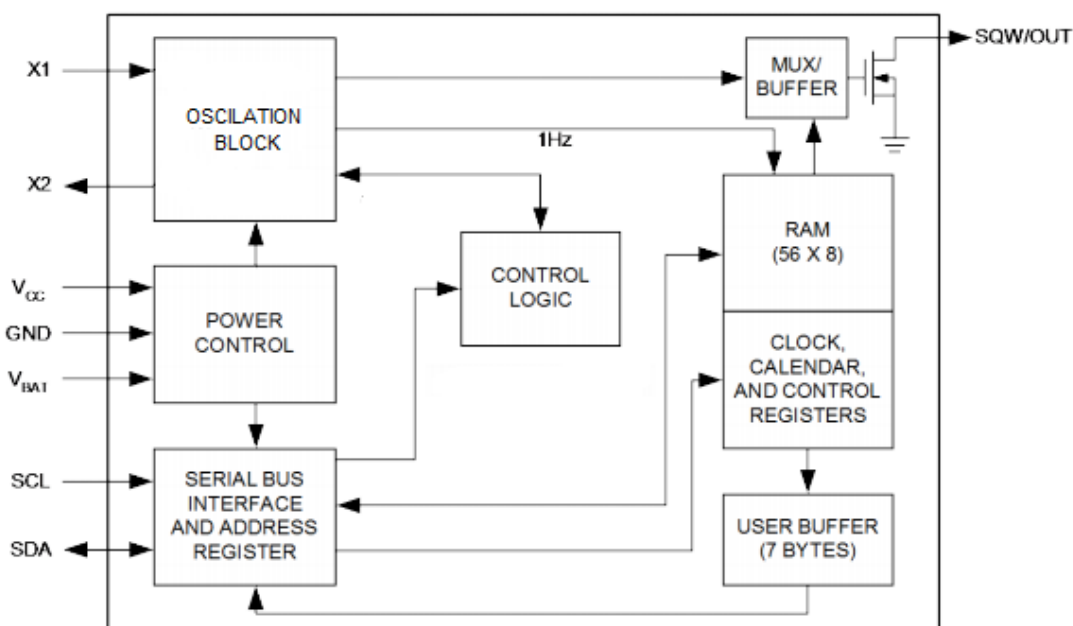


Рисунок 3.4 – Real Time clock

Долговременная память используется для записи и хранения данных, хранение которых необходимо при отсутствии основного питания. Однако, напрямую без источника питания с ней RTC взаимодействовать не сможет. Так как система однопоточная, необходимо участие одноканального устройства управления типа контроллера.

Резервное питание необходимо для обеспечения работы основной схемы при отсутствии основного питания и поддержания хода часов.

Работу часового механизма обеспечивает тактовый генератор, генерирующий тактированные сигналы с определенной частотой. Частотой, при которой счетчик в блоке тактирования сможет посчитать свое максимальное значение за 1 секунду является 32768 Гц. Именно такое количество колебаний тактового сигнала, пришедших на блок тактирования, ровно за 1 секунду переполняют счетчик от минимального до максимального значения.

### 3.9. Алгоритм функционирования комплекса

Функционирование всего комплекса выглядит следующим образом.

При подаче питания с источника питания автомобиля оно поступает на преобразователь напряжения и каскад согласования измеряемого напряжения. Так как Множество систем начинают пользоваться этим напряжением, появляется ток. Далее происходит небольшое проседание напряжения (приблизительно на 0,5 В – 1,5 В). Это связано с законом Ома: увеличивается

ток, сопротивление неизменно (сопротивление в данном случае являются все наши потребители (нагрузка) так называемые). Но это не мешает функционированию системы. Далее на выходе каскада преобразования измеряемого напряжения формируется пониженное напряжение относительно напряжения питания в +5 вольт. А на преобразователе напряжения происходит формирование пониженного напряжения равное напряжению питания микроконтроллера равное +5 В. Процесс формирования пониженного напряжения описан в подразделе 3.6.

После преобразователя питание поступает на все системы и подсистемы комплекса. Подается питание на модуль GPS, на котором в свою очередь стабилизирующий преобразователь понижает его до + 3.3 В, которыми происходит запитывание приемника и всех подсистем модуля. И сразу же на линии передачи UART (а именно TX) происходит передача сформированных пакетов на модуле GPS. Принцип формирования описан в подразделе 3.2. Но ввиду того что на раннем этапе формируются невалидные пакеты, микроконтроллер игнорирует эти сигналы. Также, питание поступает на систему индикации и часы реального времени. Но не один из этих модулей не начнет работу, так как необходимо настроить их драйвера. Также, на микроконтроллер поступает трансформированное напряжение с каскада согласования измеряемого напряжения.

При подаче питания микроконтроллер начинает выполнять код программы, расположенной в сегменте кода памяти FLASH. На начальном этапе производится инициализация и настройка всех систем и внутренней периферии контроллера. В первую очередь происходит поиск адреса с главной исполнительной функцией. Далее идет вход в адрес начала этой функции и выполнение ее инструкций.

Для начала идет настройка подсистемы GPIO, далее идет сигнализирование о том, что комплекс включен и начал работу, посредством подачи логической единицы на порт, к которому подключен динамик. Далее следует двухсекундная задержка.

Далее происходит настройка подсистемы интерфейсов SPI, UART и I2C. При настройке последовательного порта UART настраивается обработчик прерывания от данной подсистемы и настраивается обработка байтов, приходящих с этого модуля на ядро микроконтроллера. Однако, сразу включается запрет прерываний, чтобы не мешать настраивать остальные подсистемы, тем более что первые пакеты, приходящие от модуля GPS невалидны, и расходования системного времени на их обработку считаю нецелесообразной и излишней.

После настройки интерфейсов осуществляется настройка системы индикации посредством передачи команд:

- по SPI записывающей матрице команд на настройку внутренних систем и драйвера, а также очистка внутреннего дампа памяти и включение отображение этой памяти;

– по I2C интерфейсу трем знаковинтезирующим OLED дисплеям, предварительно передав необходимый адрес на устройство разадресации системы индикации.

Далее, по интерфейсу I2C происходит инициализация часов реального времени. Устанавливается начальное время 00:00:00 и запускается тактирование.

Далее свою работу начинает подсистема АЦП. Осуществляется ее инициализация и анализ принятого от каскада согласования измеряемого напряжения от бортового источника питания относительно источника питания. После задержки на вычисления полученной величины появляется величина имеющая размер от 0 до 1024. Оно сохраняется в оперативной памяти SRAM, и далее заносится в OLED дисплей и отображается. Также настраивается прерывание от АЦП. При изменении измеряемой величины происходит вызов обработчика прерывания, в котором осуществляется изменение переменной, которая хранит данные об напряжении источника питания, и устанавливается флаг о том, что данные изменены. Прерывание оканчивается. Именно такое краткое событие необходимо потому, что нахождение программного счетчика (или как его еще называют «*Instruction Pointer*», «*Program Counter*») в теле прерывания плохо скажется на работе всей системы в целом. Это считается «дурным тоном» не только потому, что контролер большую часть действий делает в прерывании, но и еще то что происходит резкая смена контекста, а старый контекст сохранен в стеке, пока прерывание не будет окончено. Но стек не бесконечен, как и время его задержки. В процессе вычислительных операций стек также заполняется, что может повлечь нарушение контекста и потерю данных. Поэтому самым безопасным и рекомендованным способом от ведущих производителей микропроцессорных систем является установка особого, предварительно созданного разработчиком флага, который будет опрашиваться в процессе нахождения в бесконечном цикле главной программы. Это более безопасно, и проще для вычислительных возможностей микроконтроллера, да и инструкций данный подход занимает наименьшее количество.

Далее осуществляется отображения логотипов и изображений спутника и знака разделения часов и минут. Осуществляется разрешение внешних прерываний от подсистем контролера. Далее следует бесконечный цикл

Вызывается прерывание от подсистемы UART о приеме байта данных. После получения кучи байт и сложении их в один пакет выставляется флаг о приходе пакета. Далее в цикле, при опросе всех существующих пользовательских флагов вызывается функция обработки пакета. Внутри нее анализируется тип пакета, и из тела этой функции вызывается следующая, в зависимости от того какой тип пакета необходимо обработать. Такое разделение вызвано тем, что размеры и значения пакетов сильно разнятся и каждый раз динамически вычислять одной функцией очень избыточно по растратам процессорного времени. При нахождении валидных пакетов

начинается их парсинг полученного пакета, и обработка данных. Далее результат обработки посылается на устройства индикации. Скорость – на знаковосинтезирующую матрицу, а время и количество спутников – на знаковосинтезирующие OLED дисплеи.

Также, осуществляет свою работу датчик освещенности, построенный на аналоговом компараторе и фоторезисторе. При наличии света на нем формируется логический ноль. При таком наборе данных микроконтроллер посылает команду на увеличение яркости на светодиодные матрицы и увеличение контраста на знаковосинтезирующих дисплеях. Как только количество света уменьшается настолько, что провоцирует изменение сигнала на аналоговом компараторе, то на вход микроконтроллера меняются данные на логическую единицу. Это сигнал к тому, что необходимо отправить команду на знаковосинтезирующую матрицу и дисплеи о понижении яркости и контраста. Если этого не выполнить, то свечение этих знаковосинтезирующих элементов будет слепить пользователя и мешать ему осуществлять движение на транспортном средстве в темное или сумрачное время суток.

## 4 РАЗРАБОТКА ПРИНЦИПИАЛЬНОЙ СХЕМЫ

На данном этапе будут выбраны конкретные модели интегральных микросхем, выбраны схемы их включения, рассчитаны вторичные источники питания, выбраны интерфейсы для сопряжения с MCU. На электрической принципиальной схеме (см. приложение Г) реализованы блоки коммутации пассивных элементов к микроконтроллеру, подключение средств индикации к микроконтроллеру, а также коммутацию внешних модулей.

### 4.1 Обоснование выбора схемы и расчет дополнительных элементов

Основным элементом электрической принципиальной схемы является микроконтроллер.

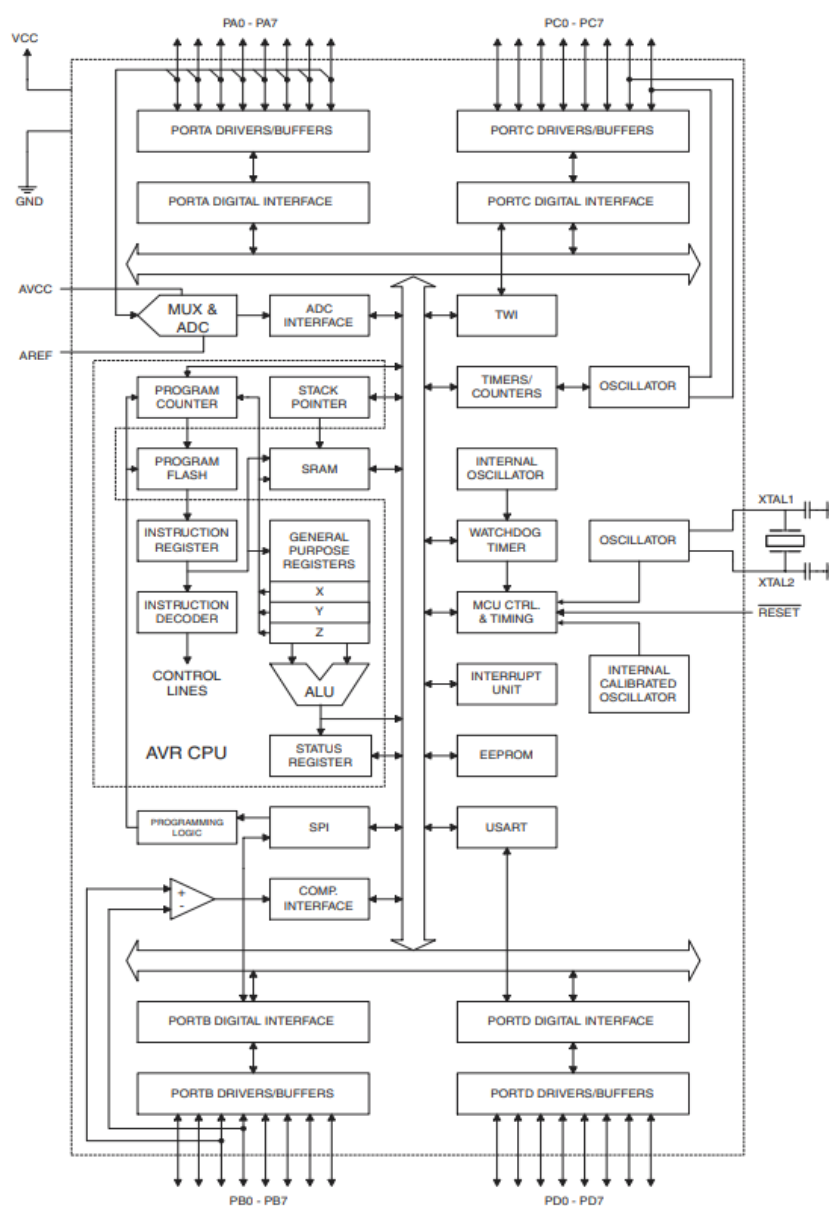


Рисунок 4.1 – Блок-схема микроконтроллера ATmega16

После того как были определены основные задачи, возложенные на данный микроконтроллер и архитектура которая будет использована, было принято решение использовать микроконтроллер архитектуры AVR из семейства Mega, который имеет все необходимые интерфейсы для взаимодействия с периферийными устройствами и звукооповещения, прием и передача данных на высокоскоростных и последовательных интерфейсах SPI и I2C. Исходя из требований подпункта 3.1.10, необходимо выбрать чип с высокой (до 16 Кб памяти FLASH и от 1 Кб ОЗУ). Наиболее подходящей является микроконтроллер ATmega16A, имеющий 40 пинов, 1 Кб ОЗУ и 16 Кб памяти FLASH, имеет все необходимые внутренние периферии. Также имеет возможность использования встраиваемых операционных систем, что даёт возможность на модернизацию и продолжение усовершенствования комплекса, как минимум, повысив быстродействие всей системы, путём использования так называемого «параллелизма» и многопоточности.

#### **4.1.1 Преобразование питания**

Данный функциональный блок должен обеспечивать питанием все датчики и устройства, подключенные к микроконтроллеру напрямую. При рассмотрении документации было установлено, что все датчики и модули, необходимые для работы исполнительных устройств могут работать от 5 вольт. Следовательно, питание, которое будет приходить извне должно быть преобразовано до 5 вольт. Полученное напряжение от источника питания будет варьироваться от 11 до 16 вольт.

Наиболее подходящим вариантом является схема преобразователя, реализованная на импульсном преобразователе. Выбор данной системы связан именно с тем, что линейные преобразователи имеют малую КПД, а также требуют дополнительную систему охлаждения, что будет влиять на габариты.

Для этих целей будет использоваться DC-DC импульсный преобразователь LM2596. Выбор данного преобразователя обусловлен тем, что диапазон его входных напряжений от 4 до 40 В, выходное напряжение, регулируемое от 1,25 до 35 В.

Преобразователь напряжения внутри состоит из входного каскада, преобразователя напряжения и выходного каскада.

Входной каскад служит для удаления пульсации с источника бортового питания, а также к медленной динамической работе, в случае резкого отключения питания. Удалять такие биение проще всего простыми фильтрами, которыми могут выступить конденсаторы.

Далее следует собственно сам преобразователь, который реализуется в стабилизационной микросхеме. Её роль является понижение входного напряжения с бортового питания путём кратковременных импульсов, которые после сглаживания попадают на нагрузку, то есть на всю нашу систему. Через резистор R1 регулируется количество и частота этих импульсов. Импульсный



преобразователь имеет большой плюс в отсутствии использования высокоомощных элементов обвязки, а также экономит место в корпусе и на плате. Однако есть минут – низкая помехоустойчивость. Но в нашем проекте нету такого требования как высокоточная напряжения питания, так что этим можно пренебречь.

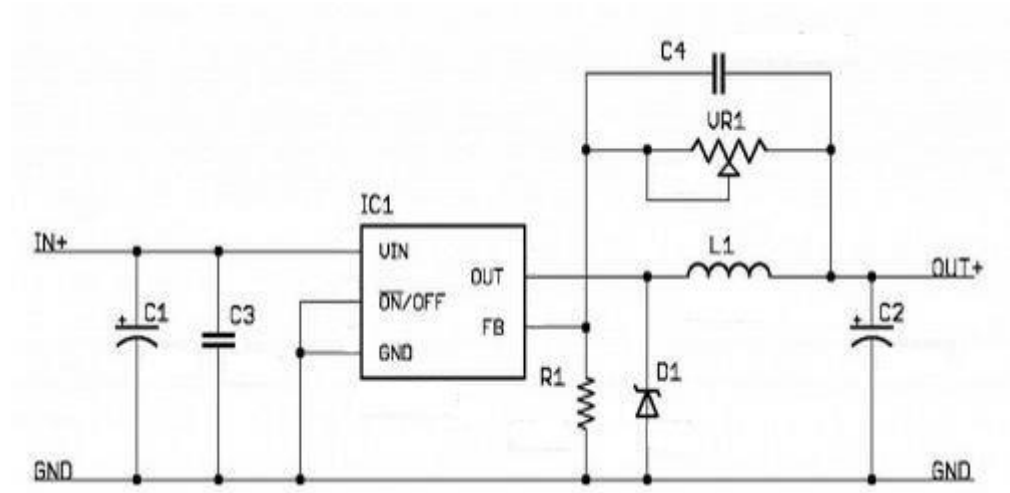


Рисунок 4.2 – Подключение преобразователя напряжения LM2596

Далее следует выходной каскад, состоящий внутри из подстроенного резистора, защитного диода, фильтрующего конденсатора и катушки индуктивности. При прохождении пульсирующего напряжения, полученного с импульсного стабилизатора, через катушку возникает ЭДС самоиндукции. Далее оно стабилизируется конденсатором, который сглаживает пульсацию. Далее напряжение попадает на делитель напряжения, состоящий из R1 и VR1. С помощью подстроенного резистора можно на месте, во время сборки сделать коррекцию получаемого напряжения. Так как далее оно попадает на вход стабилизатора, который отвечает за внесение изменений в выходной сигнал (изменение заполняемости, длительности, ведь работа импульсного преобразователя схожа с частотной модуляцией). Это называется обратная связь. Наличие диода необходимо для отсечения отрицательной полуволны пульсирующего напряжения.

#### 4.1.2 Расчёт делителя для каскада согласования измеряемого напряжения

Для классической схемы делителя, изображенной на рисунке 4.3, необходимо рассчитать сопротивления R1 и R2, из формулы (4.1):

$$U_{\text{ВЫХ}} = U_{\text{ВХ}} \cdot \frac{R2}{(R1 + R2)} \quad (4.1)$$

где  $U_{\text{вых}}$  – выходное напряжение, В;  
 $U_{\text{вх}}$  – входное напряжение, В;  
 $R1$  – сопротивление на уменьшение напряжения, Ом;  
 $R2$  – сопротивление на увеличение напряжения, Ом

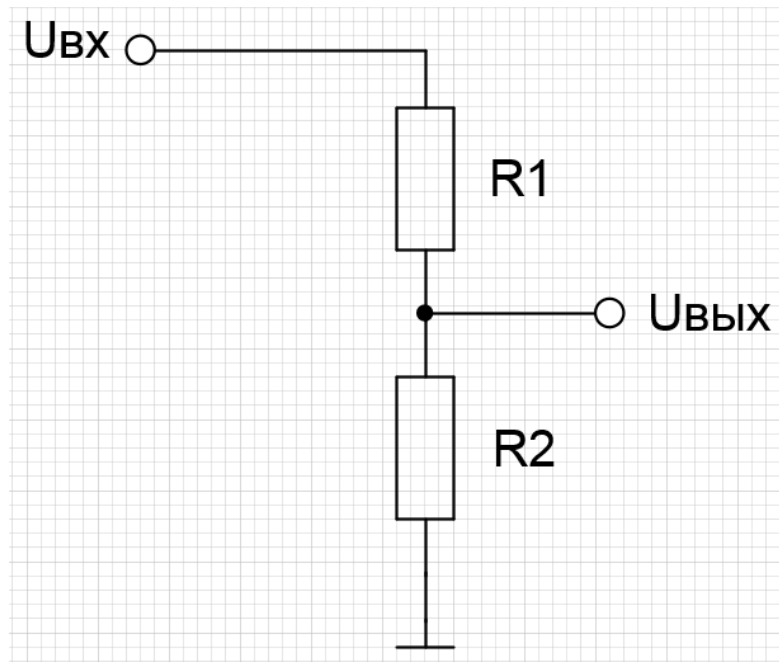


Рисунок 4.3 – Делитель напряжения. Схема.

Исходя из формулы (4.1) и исходя из максимального входного напряжения в 15 В и минимального 11 В, наиболее подходящим вариантом будет размещение  $R2$  построечное сопротивление около 1 кОм для более точной настройки каскада, а сопротивление  $R1 = 2.2$  кОм. Такая компоновка более эффективна, ввиду того что дискретные элементы имеют допуск погрешности, однако изменение значений может привести к неточным данным на АЦП, и соответственно ошибке и неверному замеру напряжения.

#### 4.1.3 Расчёт номиналов для датчика освещённости

Датчик освещенности обычно состоит из светорезистивного элемента с нагрузочным элементом, а также устройства сравнения – компаратора. Также необходим развязка для работы с компаратором. В качестве нагрузки компаратора можно использовать любую нагрузку с током потребления не более 50 мА. Это могут быть непосредственно обмотки реле, резисторы, светодиоды индикации и оптронов исполнительных устройств, с ограничивающими ток резисторами. Индуктивные нагрузки желательно шунтировать диодами от обратного выброса напряжения. Напряжение питания компаратора может быть от 5 до 36 вольт однополярного (или сумма двухполярного) напряжения. Ввиду того, что сопротивление

светорезистивного элемента не фиксировано, то необходимо использовать аналоговый компаратор. Самый распространённый и подходящий под наши требования это LM393

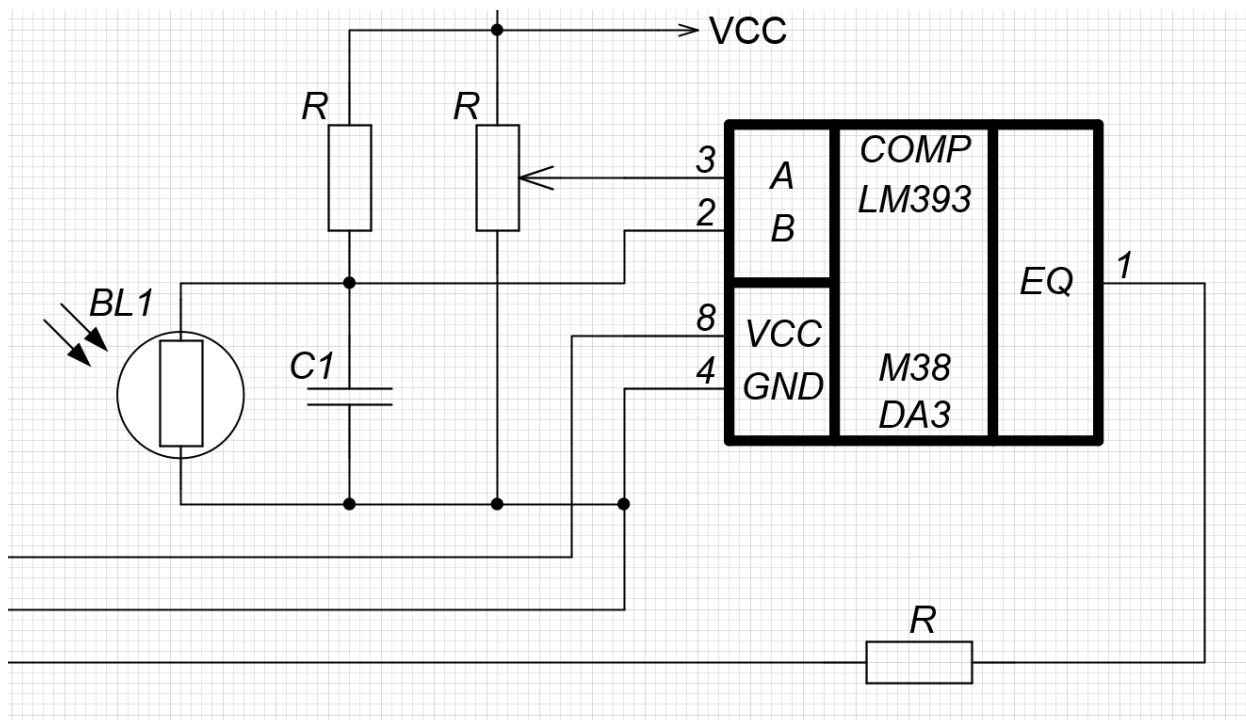


Рисунок 4.4 – Схема датчика освещённости

Глядя на схему (рисунок 4.4), мы видим, что оба входа компаратора подключены к делителям напряжения. Первый делитель напряжения, подключенный к инвертирующему входу (В), состоит из постоянного резистора и светорезистивного элемента. Как известно сопротивление неосвещенного светорезистивного элемента (фоторезистора) имеет очень большое сопротивление (более 1МОм), и малое при освещении. Поэтому в ночное время суток, согласно логике работы делителя напряжения, напряжение на инвертируемом входе В компаратора будет выше, чем в дневное время суток.

## **5 ТЕХНИКО - ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ РАЗРАБОТКИ И ПРОИЗВОДСТВА КОМПЛЕКСА ИЗМЕРЕНИЯ СКОРОСТИ ОБЪЕКТА**

### **5.1 Характеристика аппаратно-программного комплекса**

Разрабатываемый в данном дипломном проекте «Комплекс измерения скорости объекта» является функциональным прибором, который позволяет получать данные о скорости объекта относительно земли, к которому прикреплен комплекс. Данный прибор может быть использован в любой сфере, связанной с использованием транспортных средств (кроме подземных). Его пользователями могут быть как обычные водители, имеющие свой собственный автомобиль или иное транспортное средство, так и предприятия, эксплуатирующие транспортные средства.

Его преимущество — это отсутствие большой погрешности относительно стандартных приборов измерения, которыми снабжают в базовой комплектации. Основные средства по контролю скорости имеют погрешность, а также намерено заниженные показатели. И не отражают реальную обстановку объекта по отношению к окружающему миру. Данный комплекс обеспечивает недорого и точно получать данные по скорости.

### **5.2 Расчет экономического эффекта от производства аппаратно-программного комплекса**

Для производства данного комплекса были использованы следующие материалы (см. таблицу 5.1) и комплектующие (см. таблицу 5.2):

Таблица 5.1 – Расчет затрат на основные и вспомогательные материалы

Наименование материала	Ед. Изм.	Норма расхода материала	Цена, р.	Сумма, р.
1. Припой ПОС 61, 100г	г	0,28	21,00	5,88
2. Флюс ЛТИ-120, 20 мл	мл	0,4	4,40	1,76
3. МГТФ 0.12 кв.мм, Провод монтажный, 1м	м	2,8	0,76	2,13
3. Текстолит двухстороний 1,5мм, 100х100мм	мм	0,9	5,00	4,50
5. Кембрик, 1м	м	1,2	0,59	0,71
6. Стойка дистанцирующая	шт	4	0,34	1,36
Итого				16,34
Всего с учетом транспортных расходов, $P_m$				17,97

Таблица 5.2 – Расчет затрат на комплектующие изделия.

Наименование материала	Количество на изд, шт.	Норма расхода материала	Цена, р.	Сумма, р.
1. Микроконтролер Atmega16	1	1	17,00	17,00
2. GPS модуль	1	1	22,24	22,24
3. LCD модули 8x8	1	1	19,00	19,00
4. OLED дисплей 128*32	3	3	10,00	30,00
5. Резистор (1 кОм)	3	3	0,09	0,27
6. Кварц 8МГц	1	1	0,35	0,35
7. Разъем ВН10	1	1	2,00	2,00
8. Фоторезистор	1	1	3,80	3,80
9. Посадочное место для МК	1	1	1,00	1,00
10. Компаратор	1	1	8,00	8,00
11. Корпус пластиковый	1	1	9,96	9,96
Итого:				113,62
Всего с учетом транспортных расходов, Рк				124,98

Все материалы и комплектующие взяты по ценам, сложившимся на рынке на текущую дату. Коэффициент транспортных расходов взят 1,1.

Расчет общей суммы прямых затрат на производство аппаратной части изделия предоставлен в таблице 5.3.

Таблица 5.3 – Расчет общих прямых затрат.

Показатель	Сумма, р.
1. Сырье и материалы	17,97
2. Покупные комплектующие изделия	118,60
Всего прямые затраты на производство аппаратной части ( $Z_p^{ач}$ )	136,57

Расчет затрат на основную заработную плату разработчиков программной части комплекса представлена в таблице 53.4.

При Расчете зарплаты использовались данные среднемесячной зарплаты в Республике Беларусь для сотрудников IT отрасли. Премия не начисляется.

Основная зарплата определяется по формуле (5.1):

$$З_o = K_{пр} \sum_{i=1}^n З_{чи} \cdot t_i, \quad (5.1)$$

где  $K_{пр}$  – коэффициент премий ( в нашем Расчете)

$n$  – категории исполнителей, занятых разработкой;

$З_{чи}$  – часовая заработная плата исполнителя  $i$ -й категории, р.;

$t_i$  – трудоемкость работ, выполняемых исполнителем  $i$ -й категории, определяется исходя из сложности разработки программного обеспечения и объема выполняемых им функций, ч.

Часовая заработная плата каждого исполнителя определяется путем деления его месячной заработной платы (оклад плюс надбавки) на количество рабочих часов в месяце (Расчетная норма рабочего времени на 2020г. для 5-дневной недели составляет 168 ч по данным Министерства труда и социальной защиты населения на момент проведения Расчетов).

Таблица 5.4. – Расчет основной зарплаты разработчиков программной части

Категория разработчика	Месячная заработная плата, р.	Часовая заработная плата, р.	Трудоемкость работ, ч	Итого, р.
Инженер программист	2150,00	12,80	168	2150,00
Техник программист	1500,00	8,93	168	1500,00
Итого				3650,00
Премия (50–100 %)				1,00

Дополнительная зарплата разработчиков определяется по формуле (5.2)

$$З_d = \frac{З_o \cdot Н_d}{100}, \quad (5.2)$$

где  $Н_d$  – норматив дополнительной зарплаты, 15%

Отчисления в фонд социальной защиты населения и обязательное страхование БелГосстрах (Зсз) определяется в соответствии с действующим законодательством по формуле (5.3)

$$P_{соц} = \frac{(З_o + З_d) \cdot Н_{соц}}{100} \quad (5.3)$$

где  $Н_{соц}$  – ставка отчислений в фонд социальный защиты населения (ФСЗН)

и БелГосстрах (в соответствии с действующим законодательством на 01.01.2020 г. составляет 34,6%)

Расчет общей суммы затрат на разработку программной части программно-управляемого комплекса в таблице 5.5.

Таблица 5.5 – Расчет затрат на разработку программного средства

Наименование статьи затрат	Расчет по формуле	Сумма, р.
1. Основная заработная плата разработчиков	Табл.5.4	3650,00
2. Дополнительная заработная плата разработчиков	Формула (5.1)	547,50
3. Отчисления на социальные нужды	Формула (5.1)	1452,33
Итого		5649,84

Формирование отпускной цены программно-аппаратного комплекса представлена в таблице 5.6.

Таблица 5.6 – Расчет отпускной цены

Показатель	Расчет по формуле (в таблице)	Сумма, р.
1. Затраты на производство аппаратной части ( $Z_p^{ач}$ )	Табл.3.3	136,57
2. Затраты на разработку программной части ( $Z_p^{пч}$ )	Табл.3.5	5649,84
3. Сумма затрат на производство программно-аппаратного комплекса	$Z_{пр} = 136,57 + 5649,84$	5786,41
4. Накладные расходы	$P_{накл} = 5786,41 \cdot 0,56$	3240,39
5. Расходы на реализацию	$P_{рел} = 5786,41 \cdot 0,02$	115,73
6. Полная себестоимость	$C_{п} = 5786,41 + 3240,39 + 115,73$	9142,52
7. Плановая прибыль, включаемая в цену	$П_{ед} = 9142,52 \cdot 25/100$	2285,63
8. Отпускная цена	$Ц_{отп} = 9142,52 + 2285,63$	11428,15

Результатом производства аппаратно-программного комплекса является прирост чистой прибыли, полученный от их реализации.

$$\Delta\Pi_q = \Pi_{ед} \cdot N_{п} \left(1 - \frac{H_{п}}{100}\right), \quad (5.4)$$

где  $N_{п}$  – прогнозируемый годовой объем производства и реализации, шт.;

$\Pi_{ед}$  – прибыль, включаемая в цену, р.;

$H_{п}$  – ставка налога на прибыль согласно действующему законодательству, % (по состоянию на 01.01.2020 г. – 18 %).

$$\Delta\Pi_q = 2285,63 \cdot 50 (1 - 0,18) = 93710,85 \text{ руб}$$

### 5.3 Расчет инвестиций в проектирование и производство аппаратно-программного комплекса

Инвестиции в производство программно-аппаратного комплекса включают в общем случае:

- инвестиции на его разработку;
- инвестиции в прирост основного капитала (затраты на приобретение необходимого для производства нового изделия оборудования, станков и т.п.);
- инвестиции в прирост собственного оборотного капитала (затраты на приобретение необходимых для производства нового изделия материалов, комплектующих, начатой, но незавершенной продукции и т.п.).

#### 5.3.1 Расчет инвестиций на разработку аппаратно-программного комплекса

Инвестиции рассчитываем ( $I_p$ ) по затратам на разработку нового изделия. Для начала необходимо рассчитать заработную плату технологом и инженерам предприятия-производителя, ввиду того что именно они являются первой статьей первичных затрат.

Расчет заработной платы разработчиков нового изделия в таблице 5.7.

Таблица 5.7 – Расчет заработной платы разработчиков нового изделия

Категория исполнителя	Количество, чел.	Месячная зарплата, р.	Дневная зарплата, р.	Время, д.	Заработная плата, р.
1	2	3	4	5	6
1.Руководитель проекта	1	2350	111,90	84	9400,00
2. Инженер по электронной технике	1	2200	104,76	84	8800,00
3. Техник - технолог	2	1580	75,24	22	3310,48



Продолжение таблицы 5.7

1	2	3	4	5	6
4. Инженер - системотехник	1	2300	109,52	80	8761,90
Итого					30272,38
Премия, 30%					1,3
Всего основная заработная плата (З <sub>о</sub> )					39354,10

Расчета затрат на разработку нового изделия в таблице 5.8.

Таблица 3.8 – Расчет инвестиций на разработку нового изделия

Наименование статьи затрат	Расчет по формуле (в таблице)	Сумма, р.
1. Основная заработная плата разработчиков З <sub>о</sub>	табл. 5.7	39354,10
2. Дополнительная заработная плата разработчиков	$З_д = З_о \cdot 0,15$	5903,11
3. Отчисления на социальные нужды	$P_{соц} = (З_о + З_д) \cdot 0,346$	15658,99
4. Инвестиции на разработку нового изделия	$I_p = З_о + З_д + P_{соц}$	60916,20

Инвестиции в прирост основного капитала не требуются, т. к. производство нового изделия планируется осуществлять на действующем оборудовании в связи с наличием на предприятии-производителе свободных производственных мощностей.

### 5.3.2 Расчет инвестиций в прирост оборотного капитала

1) Определяется годовая потребность в материалах по формуле:

$$\Pi_m = P_m \cdot N_{п}, \quad (5.5)$$

где  $P_m$  – затраты на материалы на единицу изделия, р. (см. таблицу 3.1),  
 $N_{п}$  – прогнозируемый годовой объем.

$$\Pi_m = 17,97 \cdot 50 = 898,5 \text{ р.}$$

2) Определяется годовая потребность в комплектующих изделиях по

формуле:

$$\Pi_k = P_k \cdot N_{\Pi}, \quad (5.6)$$

где  $P_k$  – затраты на комплектующие изделия на единицу продукции, р.

$$\Pi_k = 118,60 \cdot 50 = 5930 \text{ руб}$$

3) Определяются инвестиции в прирост собственного оборотного капитала в процентах от годовой потребности в материалах и комплектующих изделиях (исходя из среднего уровня по экономике: 20–30 %) по формуле:

$$\Delta I_{\text{сок}} = 0,20(\Pi_m + \Pi_k). \quad (5.7)$$

$$\Delta I_{\text{сок}} = 0,20(898,5 + 5930) = 1365,7 \text{ руб}$$

Оценка экономической эффективности разработки изделия зависит от результата сравнения инвестиции в разработку и прирост собственных оборотных средств и полученного годового прироста чистой прибыли.

Рассчитаем рентабельность инвестиций по формуле

$$P_{\text{и}} = \frac{\Delta \Pi_{\text{ч}}}{I_{\text{пр}}} \cdot 100 \%, \quad (5.8)$$

где  $\Delta \Pi_{\text{ч}}$  – прирост чистой прибыли, руб.;

$I_{\text{пр}}$  – инвестиции в производство ( $I_{\text{сок}} + I_{\text{р}}$ ), руб.

$$P_{\text{и}} = (93710,83 / 62281,92) \cdot 100\% = 150,46\%$$

Сравнивая инвестиции в разработку изделия и прирост собственных оборотных средств, с приростом годовой чистой прибыли можно сделать вывод, что инвестиции окупаются в течении года.

Рентабельность инвестиций в производство превысила 100%, следовательно, разработка данного продукта является целесообразным.

## **ЗАКЛЮЧЕНИЕ**

Во время преддипломной практики была изучена предметная область, детально рассмотрен драйвер светодиодной матрицы. Также были изучен новый интерфейс и его протокол.

Также, сформированы первые функции управления матрицей, выведение знаков на нее, а также опробована передача сообщений через последовательный порт.

В ходе преддипломной практики была разработана структурная схема, которая в дальнейшем поможет при реализации функциональной и электрической принципиальной, отображает структуру проектируемого изделия.

Технико-экономическое обоснование выполнено.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Дивин, А.Г. Методы и средства измерений, испытаний и контроля: учебное пособие. В 5 ч. / А.Г. Дивин, С.В. Пономарев, Г.В. Мозгова. – Тамбов: Изд-во ФГБОУ ВПО «ТГТУ», 2012. – Ч. 2. – 108 с.
- [2] ЕЭК ООН №34 – [Электронный ресурс]. – Электронный данные. – Режим доступа: <http://www.unece.org/fileadmin/DAM/t/R039r2r.pdf/> – Дата доступа: 12.04.2020
- [3] GPS – [Электронный ресурс]. – Электронный данные. – Режим доступа: <https://ru.wikipedia.org/wiki/GPS> – Дата доступа: 12.04.2020
- [4] Яценков В.С. Основы спутниковой навигации / В.С. Яценков – М.: Атомиздат, 2008 – 124с.
- [5] Принцип работы системы GPS [Электронный ресурс]. – Режим доступа: <http://kunegin.com/ref6/gps/pringps1.htm> Дата доступа: 12.04.2020
- [6] Описание протокола *NMEA0813* [Электронный ресурс]. – Режим доступа: [https://www.irz.ru/uploads/files/226\\_1.pdf](https://www.irz.ru/uploads/files/226_1.pdf) – Дата доступа: 12.04.2020
- [7] Описание микроконтроллера ATtiny861 – [Электронный ресурс]. – Электронный данные. – Режим доступа : <http://ww1.microchip.com/downloads/en/devicedoc/doc2466.pdf> – Дата доступа: 12.04.2020
- [8] Описание протокола I2C – [Электронный ресурс]. – Электронный данные. – Режим доступа: <http://microsin.net/programming/avr/example-using-the-twi-i2c.html> – Дата доступа: 12.04.2020
- [9] Ревич, Ю.В. Практическое программирование микроконтроллеров Atmel AVR / Ю.В. Ревич. – СПб.: БХВ-Петербург, 2014 – 368 с.

**ПРИЛОЖЕНИЕ А**  
***(обязательное)***  
**Вводный плакат**

**Сам плакат на А1  
(рамка на обратной стороне)**

**ПРИЛОЖЕНИЕ Б**  
***(обязательное)***  
**Схема структурная**

**Сама схема на А1 с рамкой**

**на А1 с рамкой**