**Принцип измерения скорости с использованием датчика Холла и платы Arduino**

**Аппаратное обеспечение**

1. Плата Arduino Pro Mini (5V 16MHz).
2. FTDI плата (также для программирования Arduino Pro Mini можно использовать плату Arduino UNO).
3. Повышающий конвертер с 3V до 5V (DC-DC) с выходом для USB зарядки.
4. Модуль заряда литиевой батареи (аккумулятора) TP4056.
5. Bluetooth модуль (HC-05/HC-06).
6. Датчик Холла (Hall effect sensor) US1881/04E.
7. Литиевая батарея (аккумулятор) 18650.
8. Небольшой кусок магнита.
9. Перфорированная плата (Perf Board).
10. Соединительные колодки (папа и мама).
11. Набор для пайки.
12. Небольшая закрытая коробка чтобы разместить в ней наше устройство

**Программное обеспечение**

Arduino IDE

Windows/Mac PC

Датчик Холла – это устройство, способное определять полярность магнита. Если один из концов магнита поместить рядом с датчиком Холла, то датчик изменит свое состояние. Существует много различных исполнений этого датчика, но при его покупке помните о том, что для нашего проекта нужен цифровой датчик Холла.

Для работы нашего устройства необходимо прикрепить небольшой кусок магнита на колесо нашего транспортного средства. При этом всегда когда магнит будет пересекать (находиться рядом) датчик Холла, датчик будет обнаруживать это и передавать соответствующую информацию на плату Arduino.

Каждый раз, когда рядом с датчиком Холла будет обнаруживаться магнит будет генерироваться прерывание в плате Arduino. В проекте мы будем использовать непрерывный таймер на основе функции millis() и вычислять время необходимое для совершения колесом двух полных оборотов (для минимизации ошибки) с помощью следующей формулы:

Timetaken = millis() – pevtime;

Поскольку мы теперь знаем это время мы можем рассчитать число оборотов в минуту (rpm, rotations/revolutions per minute) по следующей формуле:

rpm = (1000/timetaken) \* 60;

где 1000/timetaken – это число оборотов в секунду (rps, Revolutions per second), мы его умножаем на 60 чтобы конвертировать число оборотов в секунду в число оборотов в минуту (rpm).

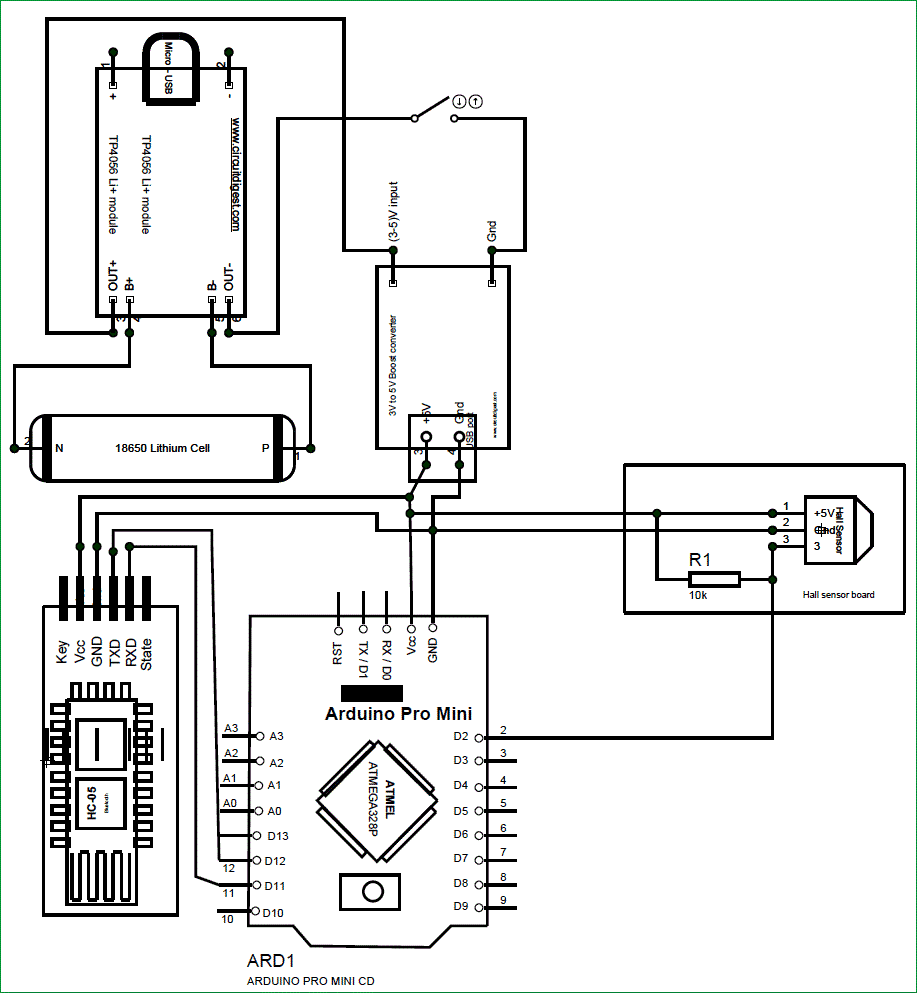
После определения числа оборотов в минуту и зная радиус колеса мы можем рассчитать скорость транспортного средства по следующей формуле:

v= radius\_of\_wheel \* rpm \* 0.37699;

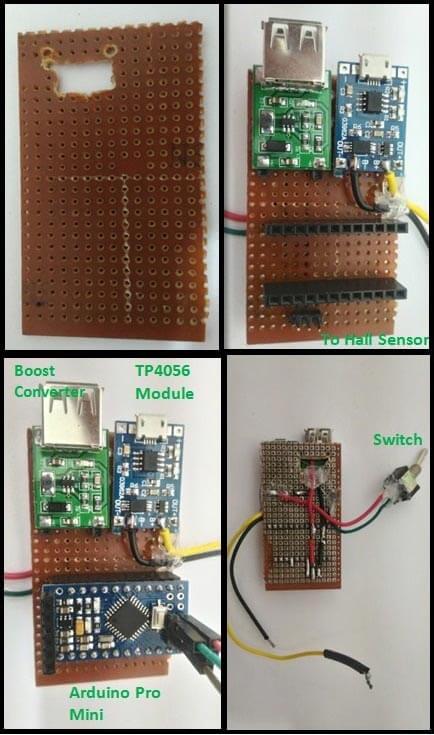
После расчета скорости Arduino передает ее значение к нам на компьютер при помощи Bluetooth модуля.

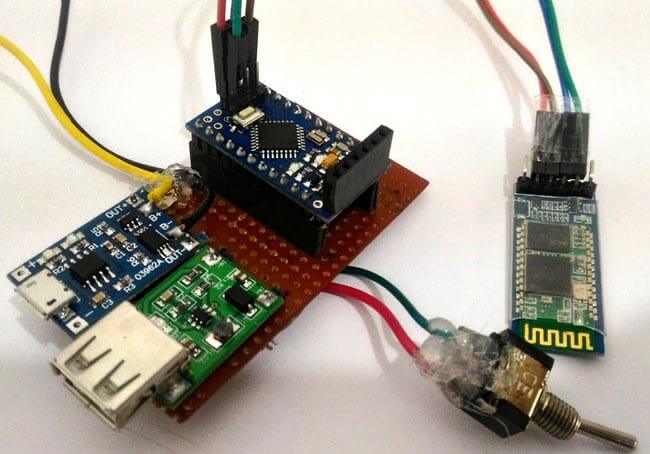
**Работа схемы**

Схема устройства представлена на следующем рисунке.

[](https://microkontroller.ru/wp-content/uploads/2020/04/speedometer-using-arduino-and-processing-android-app-circuit-diagram.gif)

Как можно видеть, схема состоит из двух частей – основная плата, которая содержит все основные компоненты, и дополнительная плата, которая содержит датчик Холла и резистор и монтируется рядом с колесом. Сначала давайте изучим основную плату.

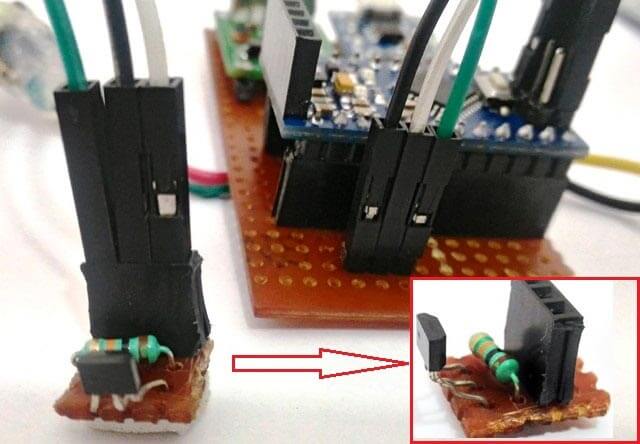




Когда все соединения на ней сделаны ее можно протестировать с использованием литиевой батареи (аккумулятора) 18650. По своей сущности литиевые батареи в некоторой степени взрывоопасны, поэтому их следует устанавливать, соблюдая меры предосторожности. По этой причине мы и используем в нашем проекте модуль заряда литиевых аккумуляторов TP4056, который обеспечивает защиту при заряде и разряде аккумулятора, а также защиту от смены полярности. Поэтому теперь наш аккумулятор можно будет без проблем заряжать с использованием обычной micro USB зарядки и безопасно разряжать. Некоторые важные характеристики модуля TP4056 приведены в следующей таблице.

|  |  |
| --- | --- |
| **Параметр** | **Значение (на одну ячейку)** |
| Under Voltage cut-off | 2.4V |
| Over voltage Cut-off | 4.2V |
| Ток заряда | 1A |
| Защита | от превышения напряжения и смены полярности |
| Микросхемы | TP4056 (charger IC) и DW01 Protection IC |
| Светодиодные индикаторы | красный - идет заряд, зеленый - зарядка окончена |

Теперь займемся платой с датчиком Холла, которая будет содержать всего два компонента – сам датчик Холла и резистор на 10 кОм. Соединения на ней показаны на схеме устройства, приведенной выше. После подключения к ней соединительных проводов мы должны получить примерный вид устройства, показанный на следующем рисунке:



Следующий ключевой шаг в сборке нашего проекта – это соединение аккумулятора 18650 к контактам B+ и B- модуля TP4056 с использованием провода. Поскольку литиевые аккумуляторы взрывоопасны, то крайне не рекомендуется использовать здесь паяное соединение. Хотя ряд радиолюбителей все же делают это, помните о том, что в этом случае вы подвергаете ваше устройство риску повреждения. Простой способ преодолеть это – использовать магниты как показано на следующем рисунке.



Просто припаяйте провод к небольшому куску магнита и затем прикрепите эти магниты к выводам аккумулятора как показано на рисунке – они будут держаться очень хорошо. Но также можно дополнительно использовать какую-нибудь ленту для укрепления этих соединений, то есть фиксации позиции магнитов.

**Программирование Arduino**

Программа для этого проекта крайне проста. Нам всего лишь будет нужно рассчитывать скорость вращения колеса и передавать ее на компьютер с помощью технологии Bluetooth. Полный текст программы с необходимыми комментариями приведен в конце статьи.

Каждый раз, когда датчик Холла обнаруживает вблизи себя магнит, он генерирует прерывание. Функция magnet\_detect() будет вызываться для обработки этого прерывания. В этой функции производится расчет числа оборотов колеса в минуту.

Когда число оборотов колеса в минуту известно в функции loop () можно вычислить скорость движения транспортного средства.

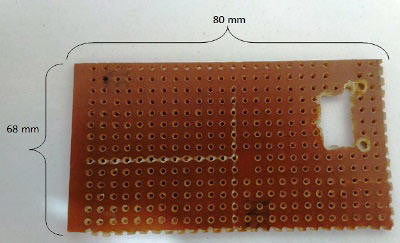
**Установка спидометра на транспортное средство**

В нашем проекте мы установили этот спидометр на велосипед, на наш взгляд получилось весьма неплохо. Далее представлены решения каким образом мы все это смонтировали и разместили на велосипеде, но вы можете сделать этот шаг на ваше усмотрение, с использованием имеющихся у вас средств и материалов. Единственное, о чем нужно побеспокоиться – чтобы магнит был надежно прикреплен к ободу колеса, а датчик Холла был размещен как можно ближе к магниту чтобы он срабатывал всегда, когда магнит будет пересекать его.

|  |  |
| --- | --- |
|  | Установка платы с датчиком Холла на колесе |

Мы использовали 3D принтер для изготовления всех необходимых коробочек и креплений, поэтому мы не были ограничены в дизайне этих вещей. Если у вас нет доступа к 3D принтеру пропустите эту часть статьи и используйте свою собственную фантазию для закрепления спидометра на вашем транспортном средстве.

Если у вас есть доступ к 3D принтеру и вы хотите использовать наши файлы для работы 3D принтера, то убедитесь что размеры вашей платы примерно такие же, как и у нас на приведенном рисунке.

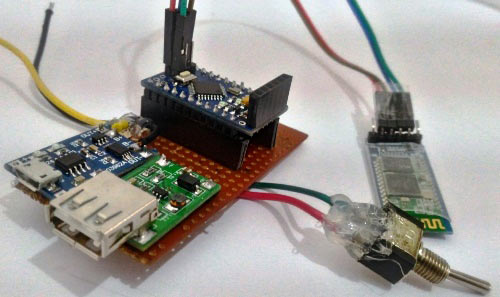


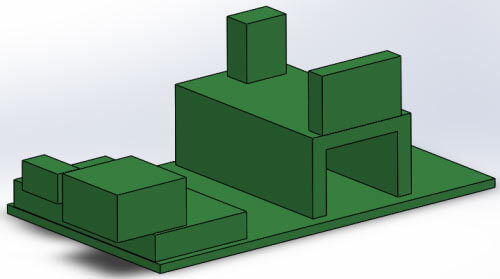
Полный комплект файлов дизайна и STL файлов для 3D печати можно скачать по [следующей ссылке](https://circuitdigest.com/sites/default/files/Speedometer-STL-and-Design-Files-for-3D-printing.zip). Если размеры вашей платы совпадают с нашими то вы можете использовать скачанные STL файлы для печати корпусов устройства. Если же размеры не совпадают, то вы можете самостоятельно их подкорректировать, используя скачанные файлы.

Сначала напечатайте на 3D принтере корпус для нашей вспомогательной платы, содержащей датчик Холла и резистор, и разместите их на вашем транспортном средстве как показано на следующих рисунках.

|  |  |
| --- | --- |
| Корпус для платы с датчиком Холла | Установка платы с датчиком Холла на колесе |

Перед печатью корпуса для основной платы желательно смоделировать как все это будет выглядеть чтобы тщательно подогнать все размеры. Вид этой модели показан на следующем рисунке.





Теперь можно приступать к дизайну корпуса для нашей основной платы. Мы разбили дизайн этого корпуса на два файла, на одной части будет смонтирована вся электроника, а вторая будет неподвижно закреплена на велосипеде с помощью гаек и болтов. Эти две части в любой момент можно будет легко соединять и разъединять. После размещения в корпусе электроники мы получим следующий вид нашего устройства:

|  |  |
| --- | --- |
|  | Размещенная основная плата спидометра в корпусе |

Как вы можете видеть, в передней части корпуса имеются два отверстия. Одно будет использоваться для вывода USB, через который мы будем заряжать наш мобильный телефон. А второе будет использоваться для micro USB, через которое мы сможем заряжать наш литиевый аккумулятор.

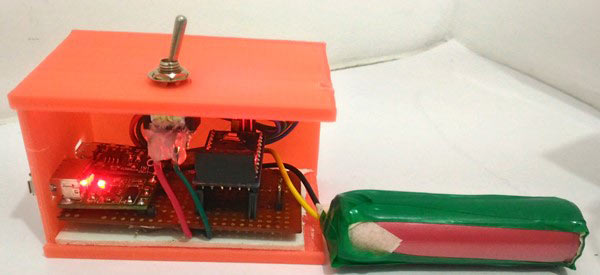
После этого печатаем вторую часть корпуса для главной платы и проверяем насколько хорошо они стыкуются друг с другом.

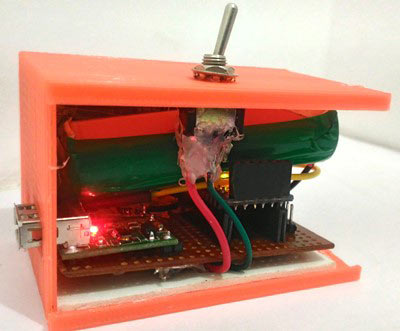
|  |  |
| --- | --- |
|  |  |

Если вы удовлетворены качеством стыковки этих двух частей, то вы можете установить неподвижную часть корпуса на велосипед.

|  |  |
| --- | --- |
| Приспособления для крепления основной платы | Крепление неподвижной части корпуса основной платы спидометра |

Теперь подсоединяем аккумулятор к нашему устройству. Желательно замотать его в герметичную ленту чтобы обеспечить целостность соединений.





Теперь наше устройство готово к окончательному монтажу. Просто соедините модуль датчика Холла с основной платой и устройство будет готово к работе.

|  |  |
| --- | --- |
| Внешний вид установленного спидометра на велосипеде | Соединение основной платы с дополнительной |

**Исходный код программы**

Если ваша литиевая батарея заряжена, то можете включить устройство с помощью переключателя, показанного на рисунках, и запустить Android приложение. Если все нормально, то вы должны увидеть на экране смартфона картинку, показанную на следующем рисунке. Перед тем как запускать приложение удостоверьтесь в наличии связи между Bluetooth модулем и вашим смартфоном.

|  |  |
| --- | --- |
|  |  |

Теперь немного проведите ваш велосипед и вы увидите как спидометр показывает вашу текущую скорость. Вы также можете заряжать свой мобильный телефон во время движения используя обычный кабель для зарядки телефона. После того как вы закончили поездку вы можете снять коробку с устройством с велосипеда и зарядить находящийся в ней литиевый аккумулятор используя зарядное устройство от мобильного телефона.

Оригинальная инструкция: <https://microkontroller.ru/arduino-projects/spidometr-s-ispolzovaniem-arduino-i-prilozheniya-na-android/>

Подключение гироскопа MPU6050 к Arduino