# Department of Electronic and Telecommunication

## University of Moratuwa

# EN2160- Electronic Design Realization

Bicycle speedo meter

**K.A.G.D.Deshan – 200117T**

# Abstraction

The bicycle speedometer presented in this design report is a versatile and essential tool for cyclists, accurately measuring and displaying vital cycling metrics in real-time. The primary focus of this product is to provide cyclists with critical information, including speed, time, temperature, and humidity, to enhance their cycling experience and promote environmental-friendly transportation.

By integrating various sensors such as the Reed switch sensor for speed measurement, DHT11 sensor for temperature and humidity readings, and an OLED display for real-time data presentation, this bicycle speedometer empowers cyclists to monitor their cycling performance effectively.

The speedometer's functionality extends beyond speed measurement, as it also includes features like time display and temperature and humidity readings, allowing cyclists to plan and optimize their rides under different weather conditions.

Moreover, the addition of a proper horn system enhances safety during cycling, alerting others and preventing potential accidents. The utilization of the Atmega328P microcontroller ensures accurate data processing and smooth operation of the speedometer.

# Table of content

## 1.Problem Statement:

*The absence of readily available speedometer and horn systems specifically designed for bicycles poses a significant challenge for riders in accurately measuring their speed, ensuring safety, and promoting environmental consciousness. Current bicycle manufacturing practices overlook the importance of integrating these essential features into bicycles, hindering the efforts to improve the cycling experience and reduce environmental pollution.*

*The lack of a reliable and easily accessible speedometer prevents bicycle riders from accurately monitoring their speed, inhibiting their ability to measure their performance, track progress, and adjust their cycling intensity accordingly. This limitation not only affects recreational riders but also poses a significant hurdle for fitness enthusiasts and professional cyclists who rely on precise speed data for training purposes.*

*Moreover, the absence of a proper horn system on bicycles leaves riders vulnerable to accidents, as they are unable to effectively signal their presence and intentions to other road users. This deficiency compromises the safety of cyclists, particularly in congested urban areas, where visibility is limited, and interaction with motor vehicles and pedestrians is frequent.*

*To build a greener world and promote environmentally friendly transportation alternatives, it is crucial to encourage and enhance bicycle ridership. By equipping bicycles with integrated speedometer and horn systems, we can empower cyclists to accurately measure their speed and communicate their presence on the road, thereby significantly reducing the likelihood of accidents and promoting a safer cycling environment.*

*Therefore, the aim of this project is to develop an innovative, user-friendly, and affordable bicycle speedometer and horn system that addresses the current limitations in bicycle design. By doing so, we strive to improve rider safety, promote environmental sustainability, and contribute to the overall advancement of cycling as an eco-friendly mode of transportation.*

# 2. Product Goals

**Functionality**

**The bicycle speedometer aims to provide cyclists with accurate and real-time speed measurement, ensuring a seamless user experience. The key functionalities of the speedometer include:**

- **Speed Measurement:** The speedometer will use a Reed switch sensor to precisely measure the speed of the bicycle based on wheel rotations.
- **OLED Display:** The device will feature an OLED display that shows real-time speed, time (using an RTC module), temperature, and humidity (measured by a DHT11 sensor).
- **Horn System:** The speedometer will include a speaker capable of producing three different tones to serve as a horn system, alerting other road users of the cyclist's presence and intentions.
- **Microcontroller:** The speedometer will be powered by an Atmega 328P microcontroller, efficiently managing speed calculations, data processing, and interface control.
- **Replaceable 9V Battery**: The device will utilize a replaceable 9V battery for easy and convenient power supply. An auto-charging system will be implemented to charge the batteries when connected to a power source, eliminating the need for manual switching.

# Market Goals

**The initial plan is to produce 100 units to gauge demand and user feedback before proceeding with mass manufacturing. The primary market goals are as follows:**

- **User-Centric Design:** The speedometer will prioritize user needs and preferences, ensuring it fulfills the requirements of cyclists effectively and efficiently.
- **Profit Maximization:** While addressing user needs, the product aims to maximize profits through effective cost management and competitive pricing strategies.
- **Market Price:** The approximate cost per unit for manufacturing is Rs. 9463.05, and the intended market price is Rs. 10,599.00, providing a balance between affordability for users and profitability for the manufacturer.

# 3.Specifications for Integrated Bicycle Speedometer and Horn System

## Technical Specifications:

**1. Speed Measurement:**

- The speedometer should utilize a Reed switch sensor to accurately measure the speed of the bicycle.
- The Reed switch sensor should be positioned to detect each rotation of the bicycle wheel and calculate the speed based on the rotation frequency.
- The speed data should be displayed in real-time on an OLED display.

**2. Display Functionality:**

- The OLED display should provide clear and easily readable information, including real-time speed, time, temperature, and humidity.
- Real-time display should be achieved using an RTC (Real-Time Clock) module.
- Temperature and humidity data should be obtained from a DHT11 sensor and displayed on the OLED screen.

**3. Horn System:**

- The system should include a speaker capable of producing three different tones to serve as the horn system.
- The horn tones should be programmable and easily distinguishable to effectively communicate the rider's presence and intentions.

**4. Microcontroller:**

- The system should be powered by an Atmega 328P microcontroller to manage the speed calculations, data processing, and interface control.
- The microcontroller should have sufficient processing power and memory to handle all required tasks efficiently.

**5. Power Source:**

- The system should be powered by a replaceable 9V battery for convenient and portable operation.
- The battery compartment should be designed for easy battery replacement.

## Operational Specifications:

**1. User-Friendly Design:**

- The speedometer and horn system should have a user-friendly interface, with clear and intuitive controls.
- The OLED display should have a suitable size and brightness to ensure easy visibility in different lighting conditions.

**2. Enclosure:**

- The speedometer and horn system should have an enclosure made of durable plastic.
- The enclosure should be weather-resistant and able to withstand exposure to various outdoor conditions, such as rain and dust.

**3. Power Efficiency:**

- The system should be designed to optimize power consumption for extended battery life.
- Low-power modes and efficient sleep/wake-up functions should be implemented to conserve energy when the device is not in use.

**4. Accurate Speed Measurement:**

- The speedometer should provide highly accurate speed measurements, with minimal deviation or lag.
- The Reed switch sensor and associated circuitry should be properly calibrated and optimized to ensure precise speed calculations.

## Disposal Specifications:

**1. Environmental Considerations:**

- The speedometer and horn system should be designed with materials that are safe for the environment and promote sustainability.

- The use of eco-friendly plastics and other recyclable materials should be prioritized in the enclosure and components.

**2. Responsible Disposal:**

- The design should allow for easy disassembly, ensuring that electronic components can be removed and recycled properly.
- Consideration should be given to reducing electronic waste and minimizing the environmental impact throughout the lifecycle of the device.

# 4.Features of the Bicycle Speedometer

**1. Accurate Speed Measurement:**

- The bicycle speedometer utilizes a high-precision Reed switch sensor to provide cyclists with accurate real-time speed measurements. By detecting each rotation of the bicycle wheel, it calculates the speed with minimal deviation or lag, enabling riders to track their performance effectively.

**2. Real-Time Data Display:**

- The speedometer features a clear and easily readable OLED display that shows essential real-time data. Cyclists can instantly view their current speed, ensuring they stay informed and maintain an optimal pace during their ride.

**3. Time Display with RTC Module:**

- The integrated Real-Time Clock (RTC) module ensures accurate timekeeping, allowing cyclists to keep track of their riding duration and manage their schedule efficiently.

**4. Temperature and Humidity Monitoring:**

- Equipped with a DHT11 sensor, the speedometer provides valuable temperature and humidity data, enabling riders to be aware of changing weather conditions and adjust their ride accordingly.

**5. User-Friendly Interface:**

- The speedometer is designed with a user-friendly interface, making it easy for cyclists of all levels to navigate and access essential features effortlessly.

**6. Horn System for Safety:**

- The speedometer includes a built-in speaker capable of producing three different horn tones. Cyclists can use this horn system to signal their presence to pedestrians and other road users, enhancing safety during their ride.

**7. Long Battery Life:**

- Powered by a replaceable 9V battery, the speedometer is designed for long-lasting performance. Its power-efficient design ensures extended battery life, allowing cyclists to enjoy uninterrupted rides without worrying about frequent battery replacements.

**8. Durable Enclosure:**

- The speedometer is housed in a robust and weather-resistant plastic enclosure, ensuring it withstands the rigors of outdoor use, including exposure to rain and dust.

**9. Easy Installation:**

- The speedometer can be easily installed on various bicycle types, requiring minimal tools and technical knowledge. Its hassle-free set-up process allows cyclists to get up and run quickly.

**10. Enhancing Environmental Consciousness:**

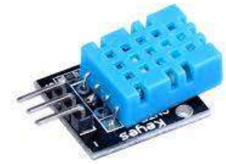
- The speedometer's design prioritizes environmental considerations, incorporating eco-friendly materials and promoting sustainability throughout its lifecycle.

# 5.Implementation of Bicycle Speedometer:

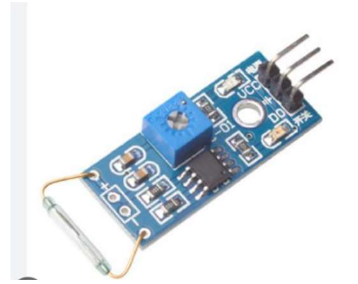
## Component Selection:

**1. DHT11 Sensor:**

- The DHT11 sensor is chosen for temperature and humidity measurement due to its affordability and reliable performance. It provides accurate readings of the ambient temperature and humidity during the ride.



**2. Reed Switch Sensor:**

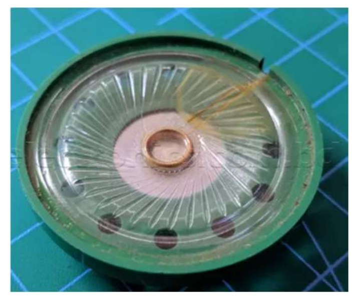


- The Reed switch sensor is selected for speed measurement as it can detect each rotation of the bicycle wheel. Its simplicity and effectiveness make it a suitable choice for calculating the speed based on wheel rotations.

**3. 9V Battery:**


- The 9V battery is chosen as the power source for the speedometer due to its portability and availability. It provides sufficient power to run the microcontroller and other components efficiently.

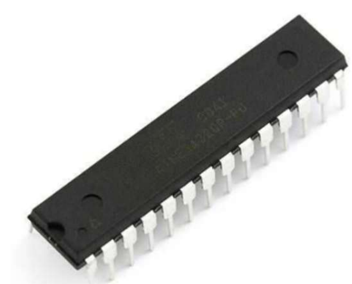
**4. Speaker (8 ohm, 0.25W):**



- The 8 ohms, 0.25W speaker is chosen for the horn system as it can produce audible and distinct tones. Its compact size and power efficiency make it suitable for integration into the speedometer.

**5. 6-Pin OLED Display (SPI):**


- The 6-pin OLED display with SPI communication is selected for visual output. Its small form factor and low power consumption make it ideal for displaying real-time speed, time, temperature, and humidity data.



**6. Atmega 328P Microcontroller:**

- The Atmega 328P microcontroller is the heart of the speedometer, responsible for data processing, control, and interface management. Its 8-bit architecture and sufficient I/O pins make it well-suited for the application.

**7. RTC 1307 module:**

# Implementation:

**1. Sensor Integration:**

- The DHT11 sensor and Reed switch sensor are connected to the Atmega 328P microcontroller to measure temperature, humidity, and detect wheel rotations. The sensor data is read and processed by the microcontroller.

**2. Power Supply:**

- The 9V battery is connected to the LM7805 voltage regulator, which converts the 9V input to a stable 5V output. The 5V output is then supplied to the Atmega 328P microcontroller, the OLED display, and other components that require 5V power.

**3. Speaker Integration:**

- The speaker is connected to the microcontroller's output pins to function as the horn system. The microcontroller controls the speaker to produce three different tones, providing audible signals to others on the road.

**4. OLED Display Operation:**

- The 6-pin OLED display is connected to the microcontroller via SPI communication. The microcontroller sends real-time speed, temperature, humidity, and time data to the display for visualization.

**5. RTC Module Interfacing:**

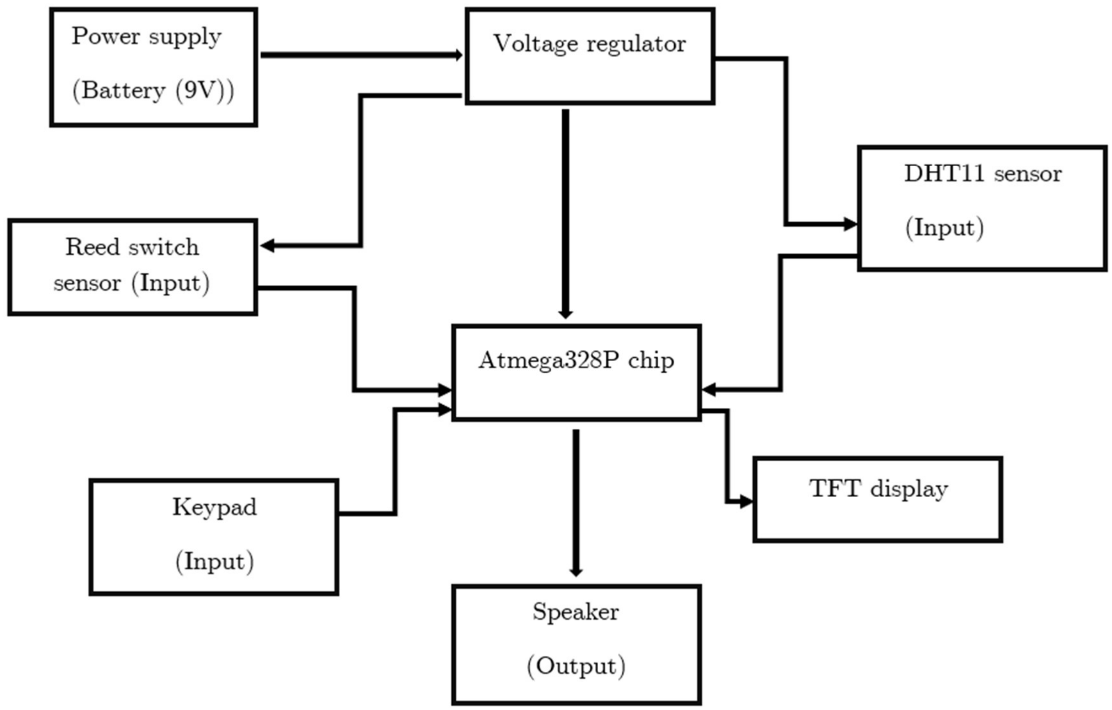
- The RTC 1307 module is interfaced with the Atmega 328P microcontroller to keep track of real-time clock data. The module ensures accurate timekeeping, allowing the microcontroller to display the current time on the OLED display.

**6. User Interface:**

- The microcontroller manages the user interface, allowing the cyclist to access different features and functionalities with intuitive controls. The user can switch between various display modes, control the horn system, and activate/deactivate specific features as needed.

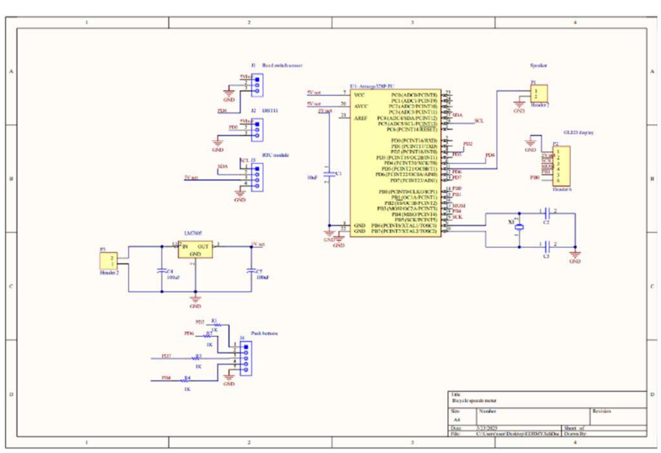
## 6.Circuit Design and Initial Testing

## Block diagram of the described circuit

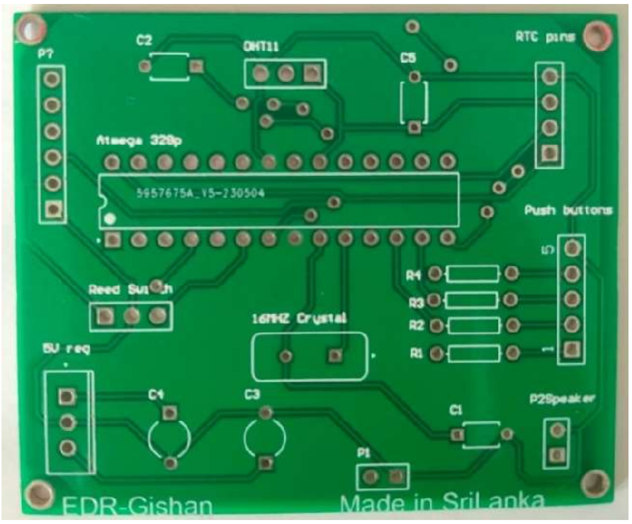


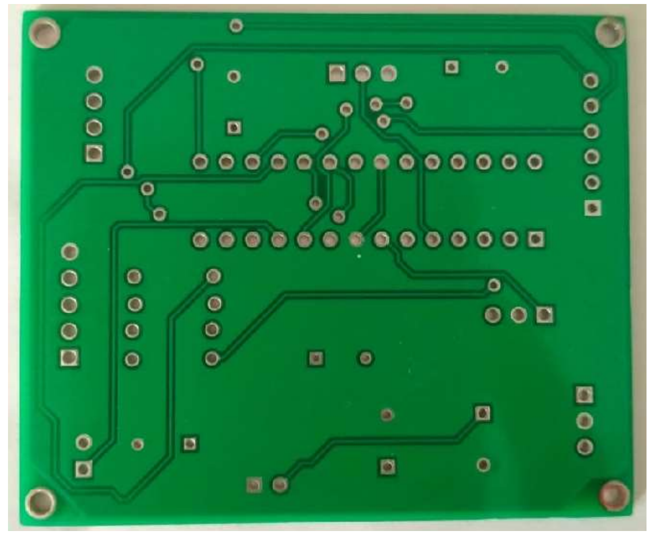

## Schematic Design

Altium software was used to create a schematic design.



## PCB Design



Top layer

Bottom layer

## Enclosure Design

# 7.Assembly Steps for Bicycle Speedometer (for manufacturing process)

**Before starting the assembly, make sure you have all the required components and tools listed below:**

## Components Required:

- Atmega328P microcontroller
- Reed switch sensor
- DHT11 sensor
- RTC 1307 module
- Speaker (0.25W, 8 ohms)
- Resistors (1K)
- Capacitors (100nF, 22pF, 100uF)
- LM7805 voltage regulator
- 9V battery holder (external, for easy replacement)
- 9V rechargeable battery
- Wires, Switches
- Soldering iron
- Solder
- Basic tools (e.g., screwdriver, pliers)

## Assembly Steps:

**1. Prepare the Circuit Board:**

- Begin by setting up the circuit board where you will assemble the bicycle speedometer. Refer to the diagrams and pictures in Appendix C for proper placement and connections.

**2. Solder Components to the Circuit Board:**

- Start by soldering the Atmega328P microcontroller, suitable JST connectors, RTC 1307 module, and speaker onto the circuit board. Pay attention to the correct orientation and pin connections.

**3. Connect Capacitors and Resistors:**

- Attach the 100uF capacitor across the power supply and ground pins of the Atmega328P to provide stable power. Connect the 22pF capacitors between the crystal pins of the microcontroller. Add the 1K resistors as needed for proper voltage regulation.

**4. Integrate the LM7805 Voltage Regulator:**

- Connect the LM7805 voltage regulator to the circuit board. It will regulate the 9V battery voltage to a stable 5V supply required for the microcontroller and other components.

**5. Wire the Circuit Components:**

- Use wires with JST connectors to establish connections between the OLED display and the circuit board, Reed switch sensor and the circuit board, DHT11 sensor and the circuit board according to the circuit diagrams in Appendix D. Double-check the connections to ensure they are correct and secure.

**6. Connect the 9V Rechargeable Battery:**

- Insert the 9V rechargeable battery into the battery holder, paying attention to the correct polarity.

**7. Secure and Insulate:**

- After connecting all the components and wires, secure them on the circuit board and insulate exposed connections using heat shrink tubing or electrical tape. This will prevent short circuits and ensure safety during use.

**8. Test the Bicycle Speedometer:**

- Before using the speedometer while riding, test it with a multimeter to verify that the output voltage is stable at 5V, and all components are functioning correctly.

(NOTE: Connections must be made in accordance with Appendix C's diagrams and images.)

## Important Safety Tips:

- Exercise caution when soldering and handling electronic components to avoid burns and injuries.
- Always check the polarity and connections to prevent any potential short circuits or accidents.
- Ensure that the battery and components used can handle the load requirements of the speedometer.
- Keep the battery and speedometer away from heat sources and flammable materials during use.
- If you are uncertain about any step or component functionality, refer to the datasheets or seek assistance from an experienced individual.

# 8.Testing the Bicycle Speedometer Functionality:

Testing the functionality of the Bicycle Speedometer is crucial to ensure its proper operation and accuracy during cycling sessions. Follow the step-by-step guide below to test the Bicycle Speedometer:

**Initial Inspection:**

- Examine the speedometer and its components for any visible damage or loose connections. If you notice any issues, address them before proceeding with testing.

**Speed Measurement Testing:**

- Place the bicycle on a stand or elevate the rear wheel to allow it to rotate freely.
- Ensure the Reed switch sensor is positioned near the bicycle wheel and the magnet is attached to the wheel's spoke.
- Rotate the bicycle wheel manually, and check whether the speedometer accurately measures the speed displayed on the OLED screen. Compare the speed displayed with a known reference, if available.
- Verify that the speed reading changes in real-time as the wheel rotates.

**Temperature and Humidity Measurement Testing:**

- Ensure the DHT11 sensor is properly placed to measure the ambient temperature and humidity during the ride.
- Expose the DHT11 sensor to different temperature and humidity conditions (e.g., warm, and cool areas) to validate its accuracy.
- Check if the displayed temperature and humidity values correspond with the actual environmental conditions.

**Real-Time Clock (RTC) Testing:**

- Check if the RTC 1307 module accurately keeps track of real-time clock data.
- Set the current time manually on the RTC module and verify if the displayed time on the OLED screen matches the set time.

**Horn System Testing:**

- Test the speaker to ensure it produces the three different tones as expected.
- Activate the horn system through the user interface and confirm that the speaker emits the selected tone.

**Power Supply Testing:**

- Connect the 9V rechargeable battery to the speedometer.

- Verify that the LM7805 voltage regulator is providing a stable 5V output to power the microcontroller and other components.
- Test the speedometer's operation under battery power and confirm that it functions correctly.

**User Interface Testing:**

- Ensure all controls on the user interface (e.g., buttons, switches) are responsive and function as intended.
- Test switching between different display modes and features to validate user interaction.

**Environmental Testing:**

- Test the speedometer under various environmental conditions, such as different temperatures and humidity levels, to ensure it operates reliably during outdoor cycling.

**Accuracy and Stability Testing:**

- Measure the accuracy of speed, temperature, and humidity readings compared to external reference measurements (e.g., GPS for speed, a separate thermometer/hygrometer for temperature and humidity).
- Observe the stability of the displayed data during various cycling scenarios, including different speeds and terrains.

**Battery Replacement (if necessary):**

- Monitor the battery's performance during testing. If the speedometer fails to operate efficiently or shows signs of battery degradation, consider replacing the battery.

**Safety Precautions:**

- Always exercise caution during testing to avoid accidents or injuries.
- Keep the speedometer and battery away from heat sources and flammable materials.
- If unsure about any step or the functionality of the components, refer to the datasheets or seek assistance from an experienced individual.

# 9.Bill of materials

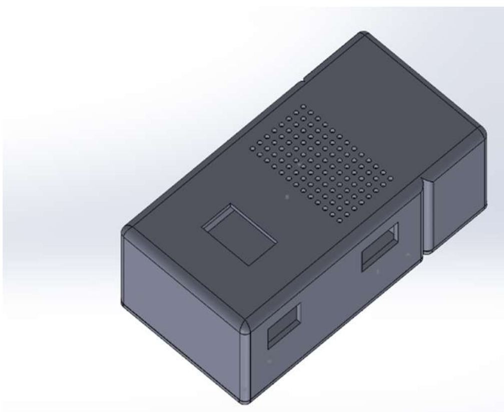
| Comment | Quantity | Manufacturer | Supplier | Price per unit | |
|---|---|---|---|---|---|
| Speaker (0.25W, 8 ohms) | 1 | Vishay | Mouser | Rs: 100 | |
| 100nF | 1 | Samsung | Mouser | Rs: 40 | |
| 100uF | 2 | Samsung | Mouser | Rs: 32 | |
| 22pF | 2 | Vishay Bicomponent | Mouser | Rs: 120 | |
| 1k | 4 | Vishay | Mouser | Rs: 50 | |
| Atmega328-PU | 1 | Microchip | LCSC | Rs:1700 | https://www.lcsc.com/product-detail/Microcontroller-Units-MCUs-MPUs-SOCs_Microchip-Tech-ATMEGA328P-PU_C33901.html |
| LM7805 | 1 | STMicroelectronics | Mouser | Rs:70 | |
| HC49US16MHz | 1 | Citizen | Farnell | Rs:120 | |
| Reed switch sensor | 1 | Goertek electronics chaina | Tronic.lk | Rs.340 | |
| DHT11 sensor | 1 | Microchip | Mouser | Rs.400 | |
| Switches | 5 | Goertek electronics chaina | Tronic.lk | Rs.25 | |
| Wires | 4m | KIND motor China | Tronic.lk | Rs.480 | |
| RTC 1307 module | 1 | Vishay Bicomponent | Mouser | Rs:780 | |
| OLED SPI display | 1 | Goertek electronics chaina | Tronic.lk | Rs.900 | https://tronic.lk/product/0-96-inch-128x64-oled-display-module-spi-serial-blue-ye |
| JST connectors | 12 | Goertek electronics chaina | Tronic.lk | Rs.900 | |
| Enclosure | | | | Rs: 2700 | |
| PCB | 1 | JLC PCB | | Rs: 2300 | https://jlcpcb.com/ |

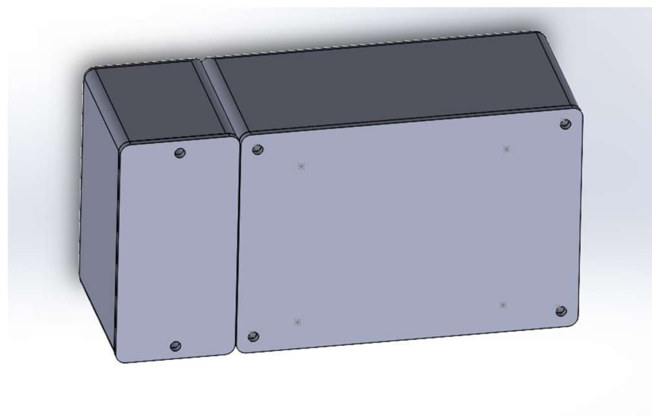
# 10.References

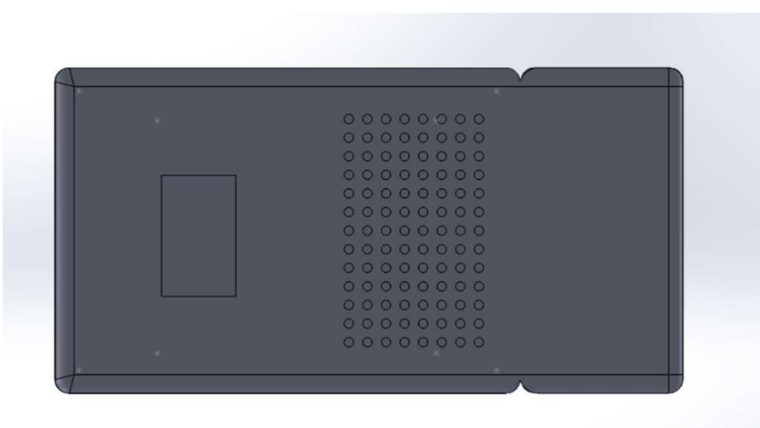

https://www.analog.com/media/en/technical-documentation/data-sheets/DS1307.pdf
https://components101.com/modules/reed-switch-sensor-module
https://cdn-shop.adafruit.com/datasheets/SSD1306.pdf
https://pdf1.alldatasheet.com/datasheet-pdf/view/241077/ATMEL/ATMEGA328P.html
https://datasheet.lcsc.com/lcsc/2304140030_TAITIEN-Elec-XIHCELNANF-16MHZ_C295075.pdf

# Appendix A

**Implemented enclosure design.**



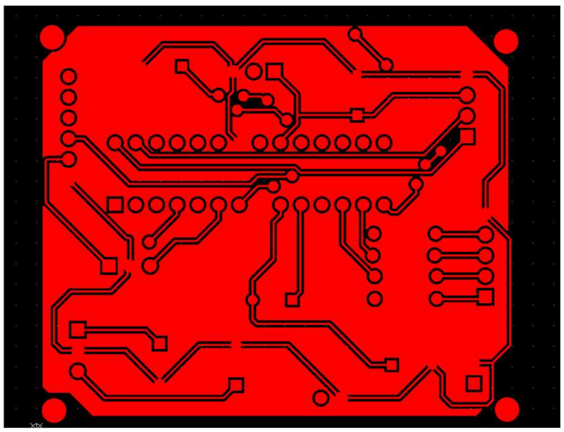




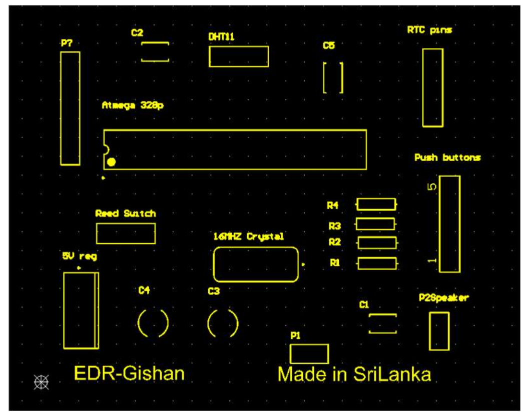
# Printed Circuit Board

## Gerber Files Details

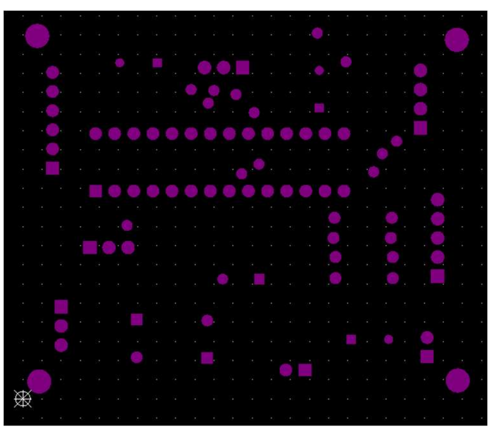
**Top layer gerber data**



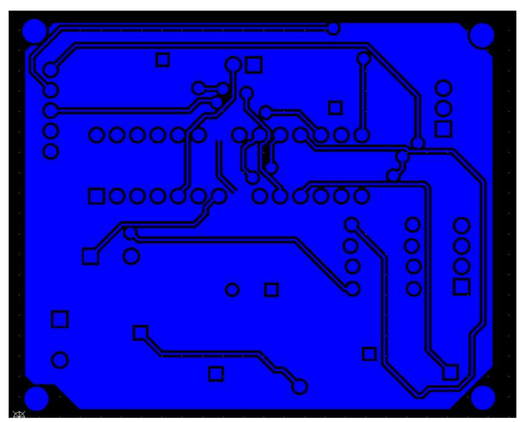
**Top overlay gerber data**




**Top solder mask gerber data**

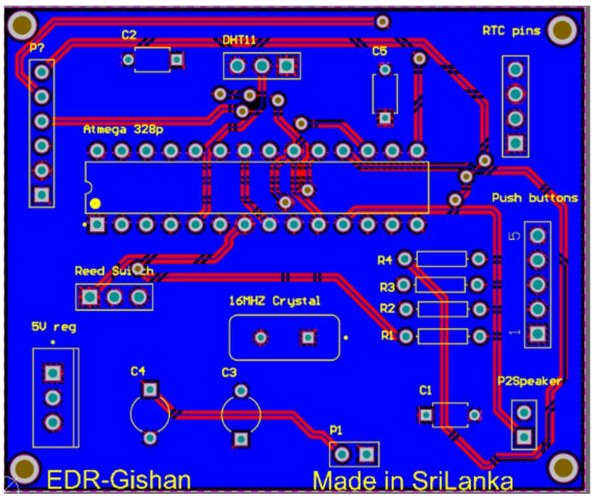


**Bottom layer gerber data**

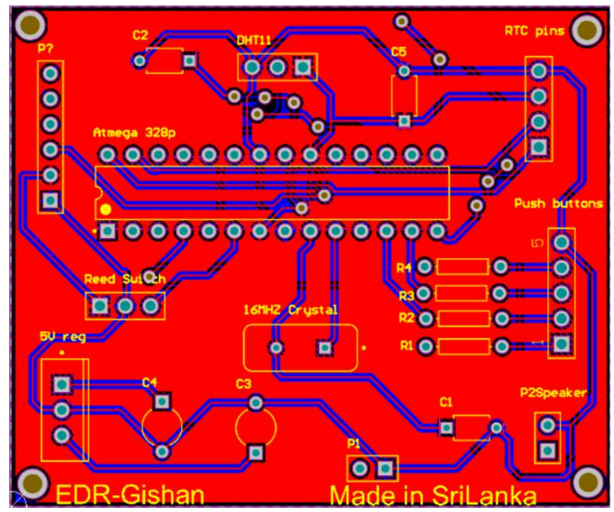


**2D view**

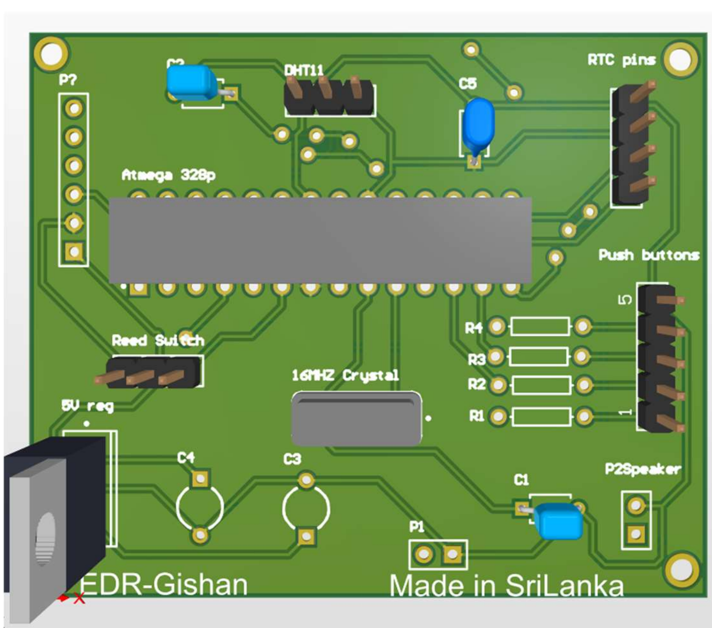
**Bottom layer**




**Top layer**




**3D view**

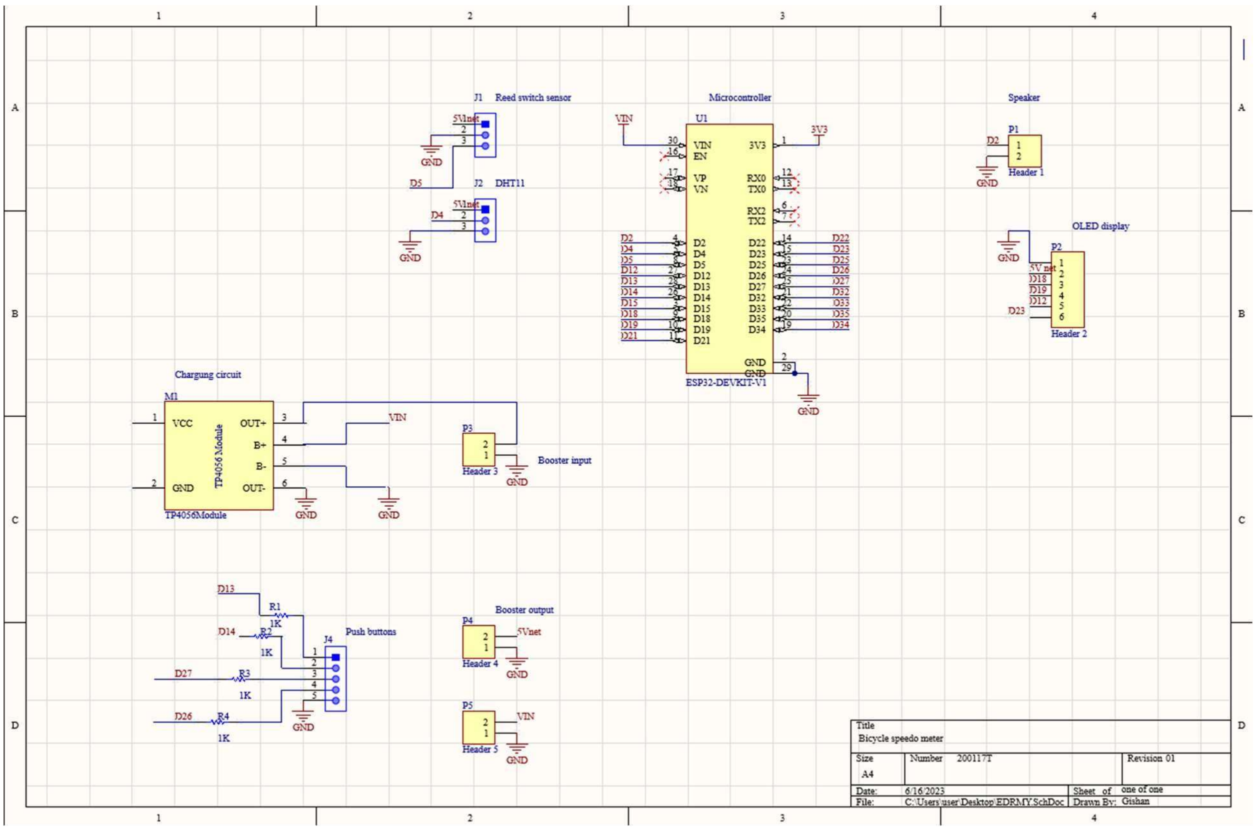


# Appendix B

## Using conceptual design, improved product design

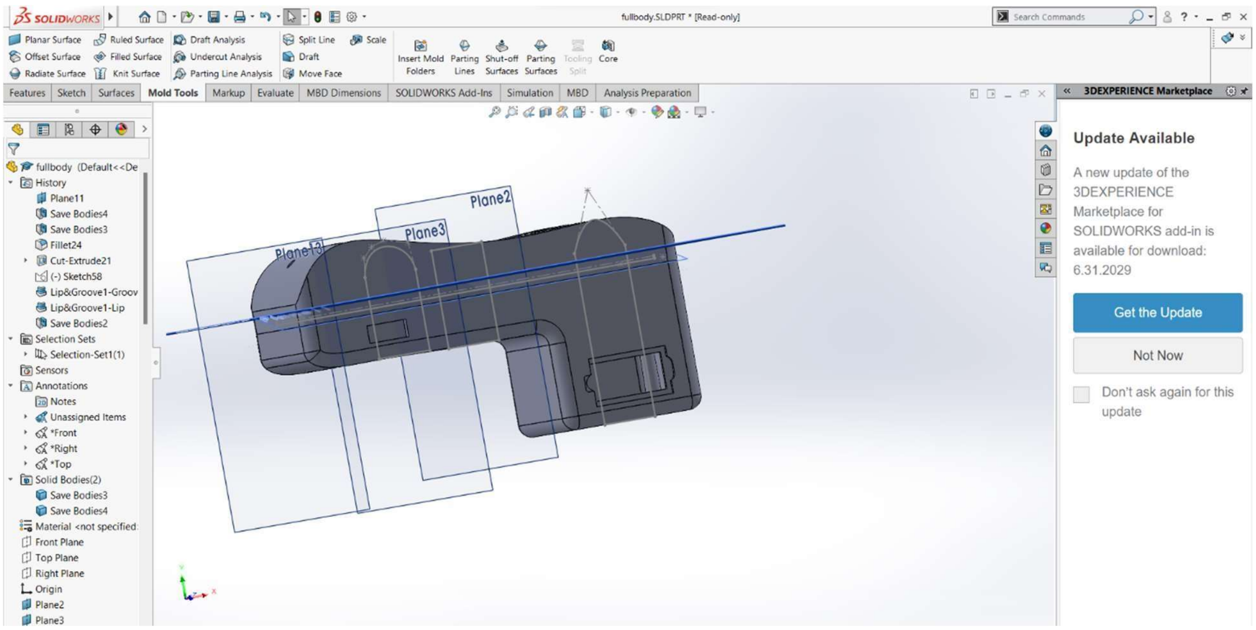
### Improved design

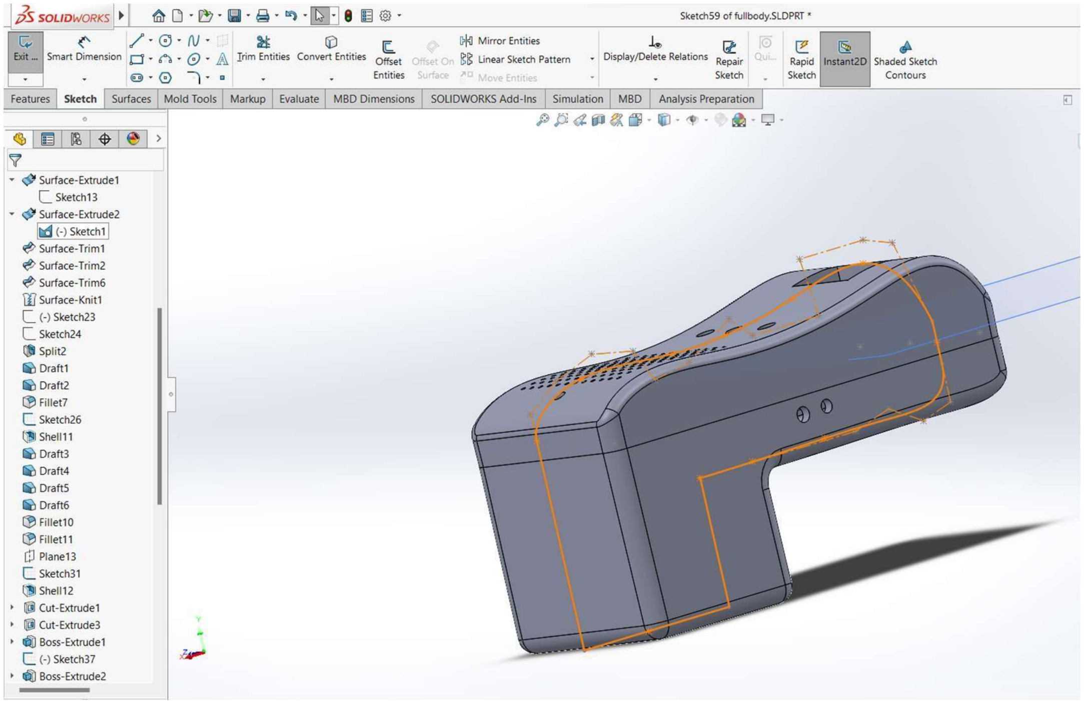
### Schematic Design



**Solid work Design**

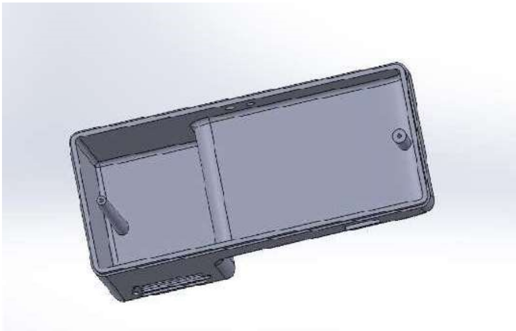
**Assembly one:**

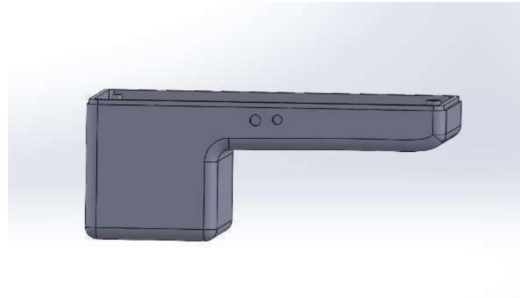



SOLIDWORKS
Sketch59 of fullbody.SLDPRT *
Exit ...
Smart Dimension
Trim Entities
Convert Entities
Offset Entities
Offset On Surface
Mirror Entities
Linear Sketch Pattern
Move Entities
Display/Delete Relations
Repair Sketch
Rapid Sketch
Instant2D
Shaded Sketch Contours
Features
Sketch
Surfaces
Mold Tools
Markup
Evaluate
MBD Dimensions
SOLIDWORKS Add-Ins
Simulation
MBD
Analysis Preparation
Surface-Extrude1
Sketch13
Surface-Extrude2
(-) Sketch1
Surface-Trim1
Surface-Trim2
Surface-Trim6
Surface-Knit1
(-) Sketch23
Sketch24
Split2
Draft1
Draft2
Fillet7
Sketch26
Shell11
Draft3
Draft4
Draft5
Draft6
Fillet10
Fillet11
Plane13
Sketch31
Shell12
Cut-Extrude1
Cut-Extrude3
Boss-Extrude1
(-) Sketch37
Boss-Extrude2

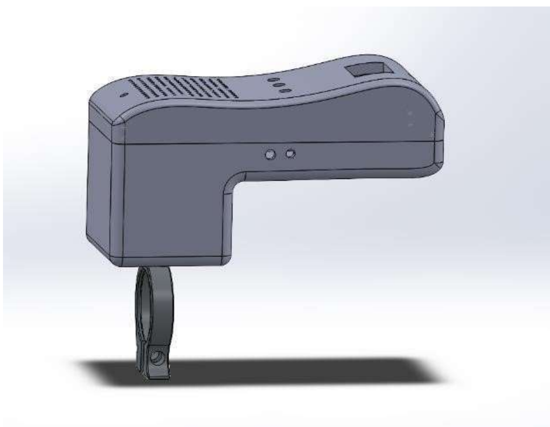
**Part 1:**

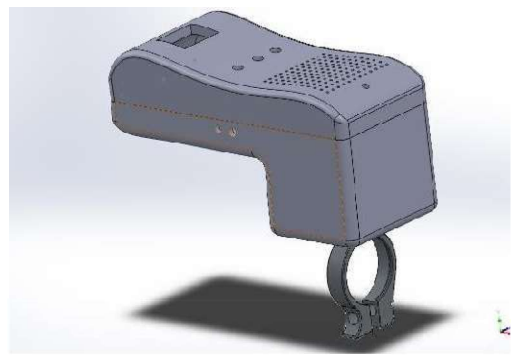
**Part 2:**

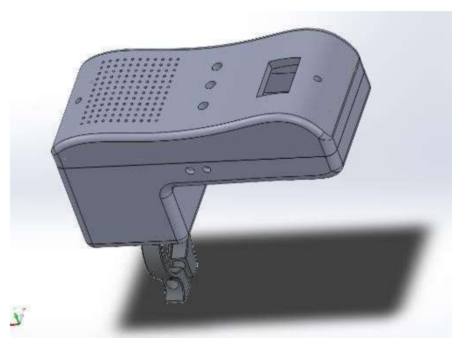




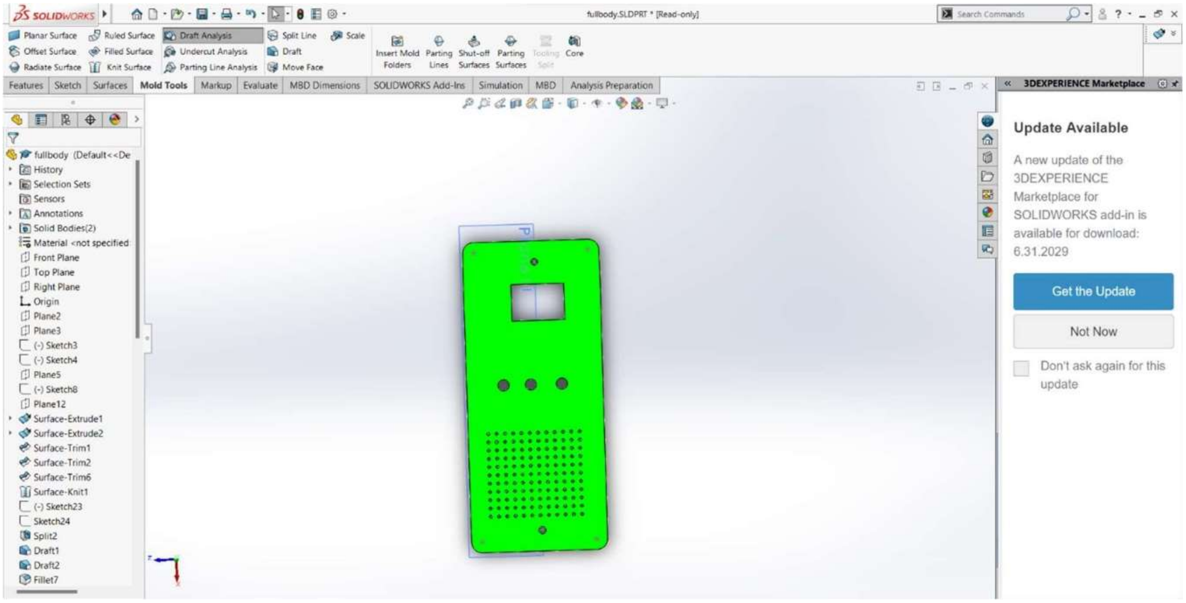
**Full view:**

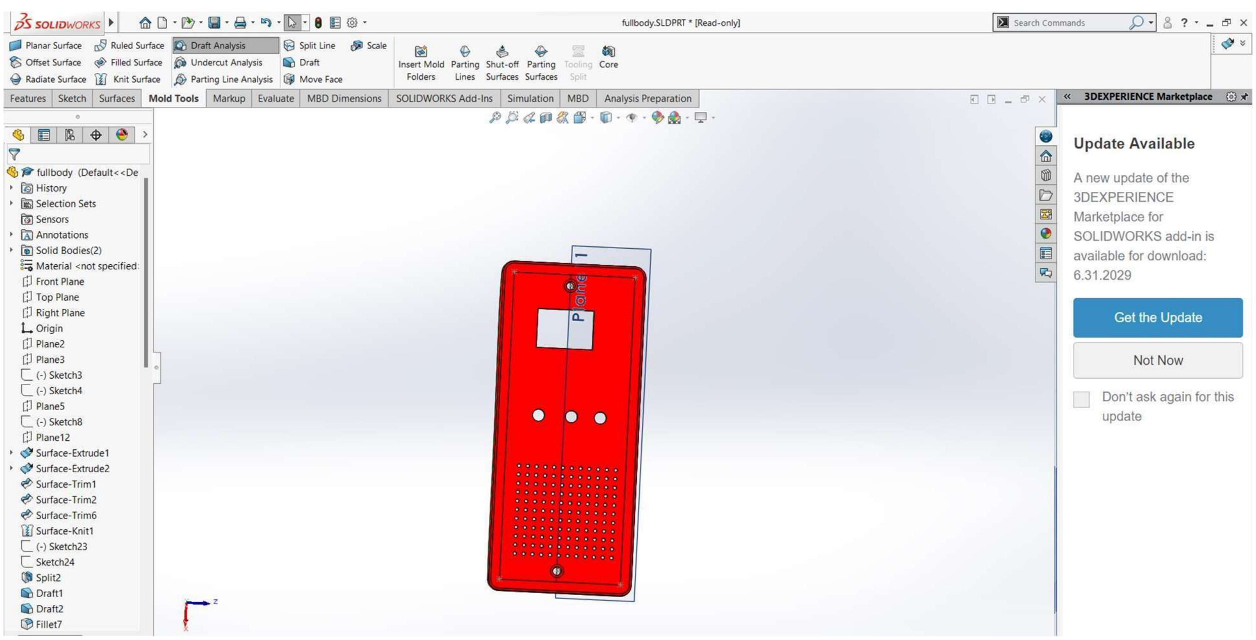


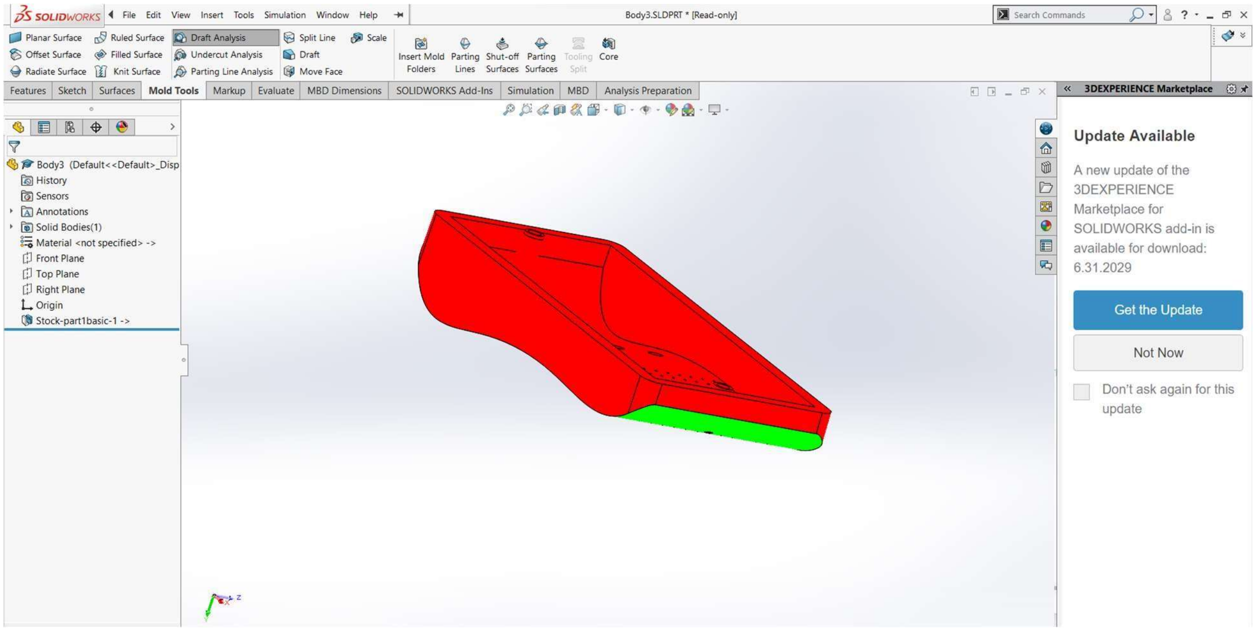




**Draft analysis of part 1:**

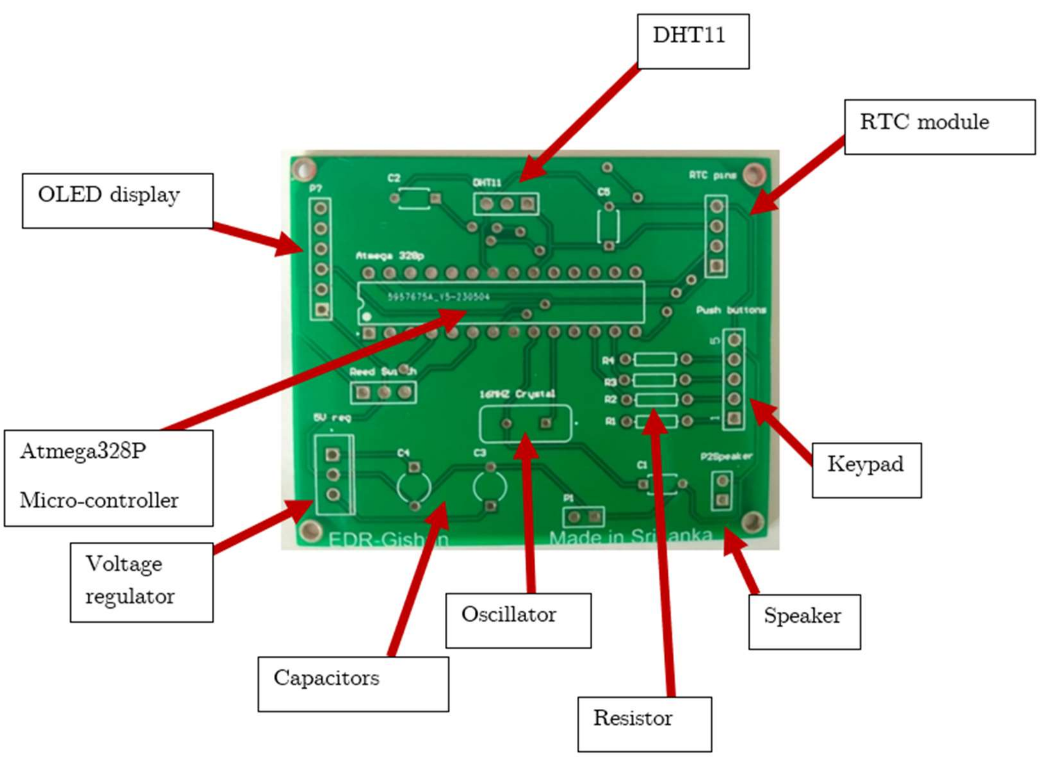

fullbody.SLDPRT * [Read-only]
Planar Surface
Ruled Surface
Draft Analysis
Split Line
Scale
Offset Surface
Filled Surface
Undercut Analysis
Draft
Radiate Surface
Knit Surface
Parting Line Analysis
Move Face
Insert Mold Folders
Parting Lines
Shut-off Surfaces
Parting Surfaces
Tooling Split
Core
Features
Sketch
Surfaces
Mold Tools
Markup
Evaluate
MBD Dimensions
SOLIDWORKS Add-Ins
Simulation
MBD
Analysis Preparation
3DEXPERIENCE Marketplace
fullbody (Default<<De
History
Selection Sets
Sensors
Annotations
Solid Bodies(2)
Material <not specified
Front Plane
Top Plane
Right Plane
Origin
Plane2
Plane3
(-) Sketch3
(-) Sketch4
Plane5
(-) Sketch8
Plane12
Surface-Extrude1
Surface-Extrude2
Surface-Trim1
Surface-Trim2
Surface-Trim6
Surface-Knit1
(-) Sketch23
Sketch24
Split2
Draft1
Draft2
Fillet7
Update Available
A new update of the 3DEXPERIENCE Marketplace for SOLIDWORKS add-in is available for download: 6.31.2029
Get the Update
Not Now
Don't ask again for this update


fullbody.SLDPRT * [Read-only]
Search Commands
Planar Surface
Ruled Surface
Draft Analysis
Split Line
Scale
Offset Surface
Filled Surface
Undercut Analysis
Draft
Radiate Surface
Knit Surface
Parting Line Analysis
Move Face
Insert Mold Folders
Parting Lines
Shut-off Surfaces
Parting Surfaces
Tooling Split
Core
Features
Sketch
Surfaces
Mold Tools
Markup
Evaluate
MBD Dimensions
SOLIDWORKS Add-Ins
Simulation
MBD
Analysis Preparation
3DEXPERIENCE Marketplace
fullbody (Default<<De
History
Selection Sets
Sensors
Annotations
Solid Bodies(2)
Material <not specified
Front Plane
Top Plane
Right Plane
Origin
Plane2
Plane3
(-) Sketch3
(-) Sketch4
Plane5
(-) Sketch8
Plane12
Surface-Extrude1
Surface-Extrude2
Surface-Trim1
Surface-Trim2
Surface-Trim6
Surface-Knit1
(-) Sketch23
Sketch24
Split2
Draft1
Draft2
Fillet7
Plane1
Update Available
A new update of the 3DEXPERIENCE Marketplace for SOLIDWORKS add-in is available for download: 6.31.2029
Get the Update
Not Now
Don't ask again for this update



SOLIDWORKS
File Edit View Insert Tools Simulation Window Help
Body3.SLDPRT * [Read-only]
Search Commands
Planar Surface
Ruled Surface
Draft Analysis
Split Line
Scale
Offset Surface
Filled Surface
Undercut Analysis
Draft
Radiate Surface
Knit Surface
Parting Line Analysis
Move Face
Insert Mold Folders
Parting Lines
Shut-off Surfaces
Parting Surfaces
Tooling Split
Core
Features
Sketch
Surfaces
Mold Tools
Markup
Evaluate
MBD Dimensions
SOLIDWORKS Add-Ins
Simulation
MBD
Analysis Preparation
3DEXPERIENCE Marketplace
Body3 (Default<<Default>_Disp
History
Sensors
Annotations
Solid Bodies(1)
Material <not specified> ->
Front Plane
Top Plane
Right Plane
Origin
Stock-part1basic-1 ->
Update Available
A new update of the 3DEXPERIENCE Marketplace for SOLIDWORKS add-in is available for download: 6.31.2029
Get the Update
Not Now
Don't ask again for this update


# Appendix C



DHT11
RTC module
OLED display
Atmega328P
Micro-controller
Keypad
Voltage regulator
Oscillator
Speaker
Capacitors
Resistor

# Appendix D

## User Manual - Bicycle Speedometer

**Introduction:**

Welcome to the Bicycle Speedometer User Manual. This manual provides essential information on efficiently setting up and using the Bicycle Speedometer to accurately measure and display your cycling speed, time, temperature, and humidity. Please read this guide carefully before using the product.

**Product Overview:**

The Bicycle Speedometer accurately measures and displays real-time data during your cycling sessions. It includes features such as speed measurement, time display, temperature, humidity readings, and a horn system for added safety while riding.

**Key Features:**

- Real-Time Speed Measurement: Accurately measures and displays your cycling speed in kilometers per hour (km/h) or miles per hour (mph).
- Time Display: Shows the current time during your ride, allowing you to track your cycling duration.
- Temperature and Humidity Readings: Provides real-time temperature and humidity measurements to help you plan your rides better.
- Horn System: Equipped with a horn system that allows you to emit three different tones for alerting others and enhancing safety on the road.

**Safety Precautions:**

- Use the speedometer only while riding a bicycle. Do not use it for any other purposes.
- Keep the speedometer and its components away from liquids and moisture to avoid damage.
- Securely mount the speedometer on your bicycle handlebar or stem to prevent it from falling during the ride.

**Getting Started:**

1. Unboxing and Inspection:

    - Carefully unbox the Bicycle Speedometer and inspect all components for any visible damage.

2. Mounting the Speedometer:

    - Use the provided mounting bracket and securely attach the speedometer to your bicycle's handlebar or stem. Ensure a stable and firm mount to prevent shaking during rides.

3. Battery Installation:

    - Open the battery compartment on the back of the speedometer and insert a 9V rechargeable battery. Ensure proper polarity alignment.

4. Initial Setup:

    - Press the power button to turn on the speedometer. Follow the on-screen instructions to set the current time and configure preferred units (km/h or mph).

5. Speed Sensor Setup:

    - Attach the Reed switch sensor to your bicycle's wheel spoke and align the magnet on the wheel rim. Adjust the sensor's position to ensure proper detection of wheel rotations.

**Operation:**

1. Speed Measurement:

    - Start cycling, and the speedometer will display your current speed in real-time.

2. Time Display:

    - During your ride, press the dedicated button to switch between speed and time displays. The time will be shown in hours and minutes.

3. Temperature and Humidity Readings:

    - Press the temperature/humidity button to view the real-time temperature and humidity readings.

4. Horn System:

    - Use the horn button to emit three different tones for signaling others or enhancing safety on the road.

**Maintenance and Troubleshooting:**

**Cleaning and Care:**

• Regularly clean the speedometer's display and buttons with a soft, dry cloth.

• Keep the speed sensor and magnet free from dirt and debris to ensure accurate speed measurements.

**Troubleshooting Guide:**

If you encounter any issues with the speedometer's functionality or display, please refer to the troubleshooting section in the product manual or contact our support team for assistance.

**Safety Information and Warnings:**

• Use the Bicycle Speedometer responsibly and adhere to local traffic laws and regulations while riding.

• Always pay attention to the road and traffic conditions while using the speedometer.

• Keep the product out of reach of children when not in use.

Don't hesitate to contact our support team for any questions or assistance with your Bicycle Speedometer. Enjoy safe and informed cycling with our product!

## Software implementation

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_Sensor.h>
#include <DHT.h>
#include "RTClib.h"
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels
#define OLED_DC     8
#define OLED_CS     10
#define OLED_RESET  9
#define DHTPIN 3
#define DHTTYPE DHT11
#define key1 12 //connect wire 1 to pin 2
#define key2 7 //connect wire 2 to pin 3
#define key3 6 //connect wire 3 to pin 4
#define key4 2 //connect wire 4 to pin 5
//int LED = ;
int reed_switch =4;
int reed_status;
unsigned long last_blink_time = 0;// Change to the appropriate pin number
//unsigned long current_time = millis(); // get current time
int i;
int minimum;


DHT dht(DHTPIN, DHTTYPE);
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT,
  &SPI, OLED_DC, OLED_RESET, OLED_CS);
RTC_DS1307 RTC;
char days[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday"};

#define NUMFLAKES     10
int f,d;
int speed=0;

void setup() {
  Serial.begin(115200);
  dht.begin();
  pinMode(key1, INPUT_PULLUP);// set pin as input
  pinMode(key2, INPUT_PULLUP);// set pin as input
```

```
  pinMode(key3, INPUT_PULLUP);// set pin as input
  pinMode(key4, INPUT_PULLUP);// set pin as input
  //pinMode(LED, OUTPUT);
  pinMode(reed_switch, INPUT);
  //Serial.begin(9600)
  if (!RTC.begin()) {
    Serial.println("Couldn't find RTC");
    while (1);
  }
  RTC.adjust(DateTime(2023, 5, 19, 8, 0, 0)); // Set the date and time to
2023/4/16 01:03:00
// Initialize DHT sensor

  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) { // Address 0x3C for 128x64
    Serial.println(F("SSD1306 allocation failed"));
    for (;;);
  }


  // Attach the interrupt for the reed switch

  delay(1000);
  display.clearDisplay();

  display.setTextSize(3);
  display.setTextColor(WHITE);
  display.setCursor(0,4);
  // Display static text
  display.println("Speedo");
  display.setTextSize(2);
  display.setCursor(40,40);
  // Display static text
  display.println("CalTone");
  delay(3000);
  display.clearDisplay();
  display.display();


}

void TimeSet(){
  DateTime now = RTC.now();
  display.clearDisplay();
  display.setTextSize(1);
  display.setCursor(0, 0);
```

```
  display.print(days[now.dayOfTheWeek()]);
  display.println(' ');
  display.setCursor(50,0);
  display.print(now.year());
  display.print('/');
  display.print(now.month());
  display.print('/');
  display.print(now.day());
  display.print(' ');
  display.setCursor(0,8);
  display.setTextSize(1);
  display.print("Time : ");
  display.println(' ');
  display.setCursor(30,8);
  display.setTextSize(1);
  if (now.hour() < 10)
    display.print('0');
  display.print(now.hour());
  display.print(':');
  if (now.minute() < 10)
    display.print('0');
  display.print(now.minute());
  display.print(':');
  if (now.second() < 10)
    display.print('0');
  display.print(now.second());
  display.display();
}
void readTem_Hum(){

  float t = dht.readTemperature();  /*read temp*/
  float h = dht.readHumidity();    /*read humidity*/
  // if (isnan(h) || isnan(t)) {
  //   Serial.println("Failed to read from DHT sensor!");
  // }

  //display.print(now.second());
  display.setCursor(0, 30);
  display.setTextSize(1);
  display.print("Humidity: ");
  display.print(h);      /*prints humidity percentage*/
  display.print(" %");
  display.setCursor(0, 40);
  display.print("Temperature: ");
  display.print(t);       /*print temp in Celsius*/
```

```
  //display.print(" ");
  display.print("°C");
  display.display();
  delay(500);
}
void speedcal(){
   if (reed_status == HIGH) { // check if reed switch is closed
      //digitalWrite(LED, LOW);
      //Serial.println(reed_status);
    }
    else {
      while(digitalRead(4)==LOW){
          //digitalWrite(LED, HIGH);
      }
      //delay(100);
      Serial.println("ON");
      Serial.println(current_time);
      //digitalWrite(LED, LOW);


      //Serial.println("ON");

      //Serial.println("led on");
      // if (current_time - last_blink_time > 1000) { // check if time since last
blink is greater than 1 second
      float time_interval = (current_time - last_blink_time) / 1000.0; //
calculate time interval in seconds

      //Serial.println(time_interval);
      float speed = (2*3.14*36*30/(time_interval*1000)); // calculate speed in
km/h
      Serial.println(time_interval);
      // Serial.print("Speed: ");
      // Serial.print(speed);
      // Serial.print(" km/h");
      // Serial.println(" ");
      display.clearDisplay();
      display.setTextSize(1);
      display.setTextColor(SSD1306_WHITE);
      display.setCursor(0,25);
      display.print("Speed");
      display.setCursor(40, 25);
      display.setTextSize(2);
      display.print(speed);
      display.setCursor(100, 25);
```

```
      display.setTextSize(1);
      display.print("km/h");
      display.display();
      last_blink_time=current_time;

    }


}



void loop() {
    unsigned long current_time = millis();
    int key1S = digitalRead(key1);
    int key2S = digitalRead(key2);
    int key3S = digitalRead(key3);
    int key4S = digitalRead(key4);
    reed_status = digitalRead(reed_switch); // read current state of reed switch
  //delay(1000);
    speedcal();






      //Serial.println(current_time); // update last blink time




    // if (!key1S) {
    //   Serial.println("key 1 is pressed");
    //   for(f=635;f<=912;f++)
    //   {
    //     tone(10, f);
    //     delay(d);

    //     //if (digitalRead(2)==HIGH)
    //     f=912;

    //     // if (digitalRead(3)==HIGH)
    //     //   d=0;
    //     // else
    //     //   d=7;
    //   }
```

```
    //   for(f=911;f>=634;f--)
    //   {
    //     tone(10, f);
    //     delay(d);

    //     //if (digitalRead(2)==HIGH)
    //     f=634;

    //     //if (digitalRead(3)==HIGH)
    //     //   d=0;
    //     // else
    //     //   d=10;
    //   }

    //   // while (digitalRead(2)==HIGH)
    //   // {
    //   //   noTone(10);
    //   //   tone(10, 250);
    //   //   delay(5);
    //   //   noTone(10);
    //   //   delay(5);
    //   // }
    // }
    if (!key1S) {
      //Serial.println("key 1 is pressed");
      TimeSet();
      readTem_Hum();

    }
    speedcal();


}
```