Vehicle Health Monitoring System

Vijay KumarG[1], MeghanaS[2], JayaChandra G[3], Girish Kumar D[4], Vidyasagar M[5], Supraja K[6], Pavitra J[7]
[1,2,3,4,5,6,7] *Department of Computer Science and Engineering, Aditya Institute of Technology and Management, Tekkali*

***Abstract*— We face problems which can be predictable and unpredictable during our journey. At times during our journey, we face many issues such as leakage of engine oil, puncture of the tyre, release of unsaturated carbons, vehicle vibrations, engine overheating which are considered as the Vehicle Health Parameters (VHP's). Every reading such as the fuel and speed are being predicted and displayed in the vehicle dashboard itself. The other needful information is not being calculated and there is no application which can constantly monitor the health of the vehicle. Our solution is to develop a web application which constantly monitors factors such as leakage of engine oil, puncture of the tyre, release of unsaturated carbons, vehicle vibrations, engine overheating.**

As an approach, we are developing an IoT based sensor network system that can send the real time data to the web application and displays the factors on the web dashboard for visualization with the aid of the integrated vehicle sensor network.

Index Terms: **Thingspeak, Javascript, Bootstrap, HTML,ESP8266 Microcontroller, MQ-02 Gas sensor, DHT11 Temperature Sensor, HCSR-04 Ultrasonic Sensor.**

I.INTRODUCTION

The Vehicle Health Monitoring System (VHMS) is a vehicle health management system that detects abnormalities and malfunctions. Sensors, processing, communications, and power supply are some of the tasks that are considered for a selected study. Traveler information is essential for maintaining safety, security, mobility, and enhancing travel reliability. This traveller data might be ongoing data based on the vehicle's operation and the condition of its interior components. An in-vehicle embedded system is being built in this study to create a vehicle health dashboard (VHD). This dashboard helps the user, to monitor the vehicle and its performance. It anticipates potential errors, allowing the driver to travel without interruption and avoid accidents. As a result, it informs the driver of potential errors and aids him in driving safely.

From the last few years, a lot of cloud computing providers are also offering IoT services. A recent communication infrastructure of IoT has envisaged the future where everyday life will be based on intra-communication of transceivers, microcontrollers, digitally communicating via stacks of suitable protocols [1].

A computer hardware system with software embedded in it is known as an embedded system. An embedded system can be a standalone unit or a component of a larger system. A microcontroller or microprocessor-based embedded system is one that is built to execute a certain purpose. In this, we are using this system to identify vehicle health by monitoring internal parameters.

II. HARDWARE

A. ESP8266 Microcontroller:

NodeMCU is open source platform, it's hardware design is open for edit/modify/build. NodeMCU Dev Kit/board consist of ESP8266 WiFi enabled chip. The ESP8266 is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol [5]. This controller is used in our project to take multiple inputs and to forward the sensed data to the cloud servers.

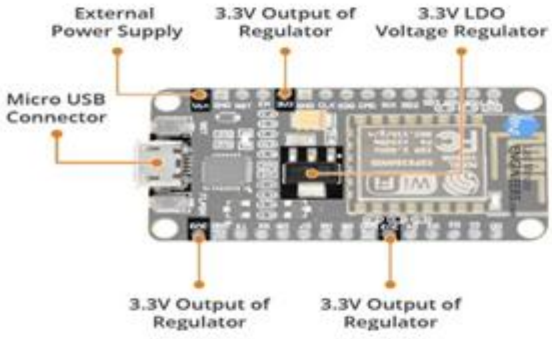


Fig 1. ESP8266 Microcontroller

Power Supply:

The ESP8266's operational voltage range is 3V to 3.6V, and the board has an LDO voltage regulator to keep the voltage stable at 3.3V. When the ESP8266 draws up to 80mA during RF broadcasts, it can dependably supply up to 600mA, which should be more than enough. The regulator's output is likewise separated off to one of the board's sides and designated as 3V3. Power can be supplied to external components through this pin.

Peripherals and I/O:

On both sides of the development board, the ESP8266 NodeMCU has 17 GPIO pins split off to pin headers. These pins may be used for a variety of different tasks, including:

- ADC Channel - A 10-bit ADC channel
- UART interface – UART interface is used to load code serially.
- PWM outputs – PWM pins for dimming LEDs or controlling motors.
- SPI, I2C & I2S interface – SPI and I2C interface to hook up all sorts of sensors and peripherals.
- I2S interface – I2S interface if you want to add sound to your project.

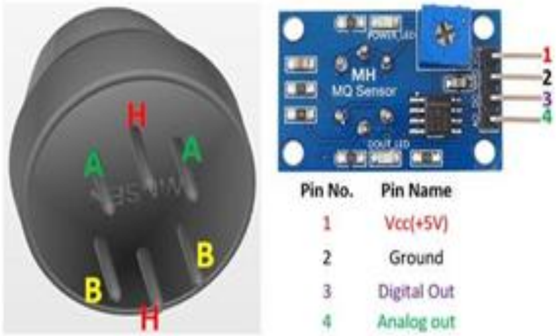
On-Board Switches & LED Indicator:

There are two buttons on the ESP8266 NodeMCU. The Reset button, labelled RST and placed in the upper left corner, is used to reset the ESP8266 chip. The download button is located in the bottom left corner and is used to upgrade firmware.

A user-programmable LED indication is also included on the board, which is coupled to the D0 pin.

B. MQ-02 Gas Sensor:

MQ-02 Gas Sensor is an electronic sensor that detects the presence of gases such as LPG, propane, methane, hydrogen, alcohol, smoke, and carbon monoxide in the air. It can detect gases in the concentration of range 200 to 10000ppm. MQ-02 is a gas sensor from the MQ sensor series that is widely used. It is a Metal Oxide Semiconductor (MOS) type Gas Sensor, also known as Chemiresistors, since the detection is dependent on a change in the sensing material's resistance when the Gas comes into contact with it. Gas concentrations may be sensed using a simple voltage divider network.



Pin No.	Pin Name
1	Vcc(+5V)
2	Ground
3	Digital Out
4	Analog out

Fig 2. MQ-02 Gas Sensor Pin Diagram

C. DHT11 Temperature Sensor:

The DHT11 is a widely used temperature and humidity sensor that includes a specialized NTC for temperature measurement and an 8-bit microprocessor for serial data transmission. The DHT11 is a basic digital temperature and humidity sensor that is extremely inexpensive. It measures the ambient air with a capacitive humidity sensor and a thermistor and outputs a digital signal on the data pin (no analogue input pins needed). It's simple to use, but data collection takes precise scheduling.

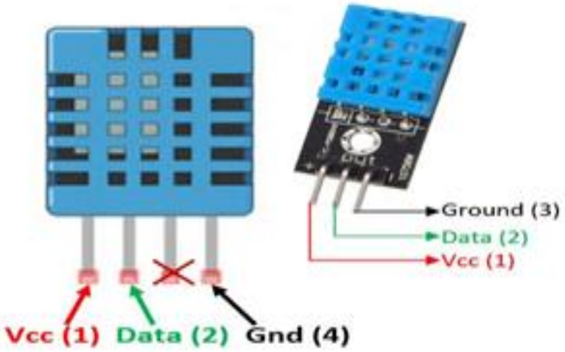


Fig 3. DHT11 Temperature Sensor Pin Diagram

D. HCSR-04 Ultrasonic Sensor:

With a range accuracy of up to 3mm, this sensor enables non-contact measuring capabilities from 2cm to 400cm. An ultrasonic transmitter, a receiver, and a control circuit are all included in each HC-SR04 module. The HC-SR04 Ultrasonic Distance Sensor uses sonar to determine the distance between two objects. It's perfect for any robotics projects that require you to avoid obstacles by sensing their proximity and steering away from them.

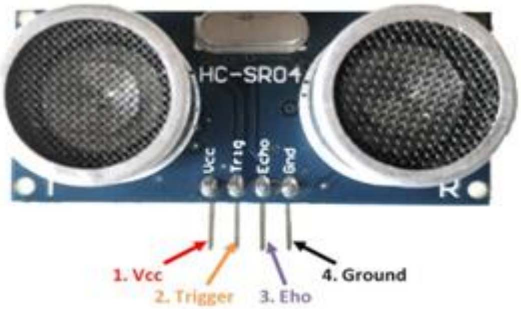


Fig 4. HCSR-04 Ultrasonic Sensor PinDiagram

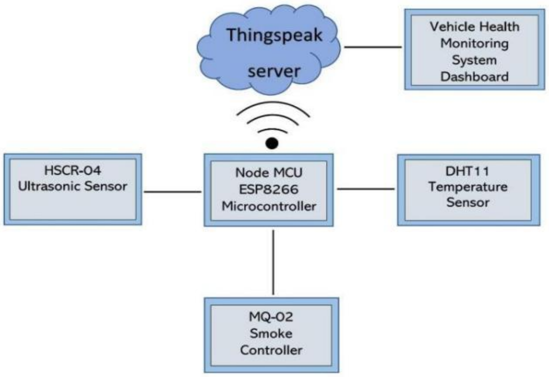

Thingspeak server
Vehicle Health Monitoring System Dashboard
HSCR-04 Ultrasonic Sensor
Node MCU ESP8266 Microcontroller
DHT11 Temperature Sensor
MQ-02 Smoke Controller

Fig 5. Block diagram for VHMS

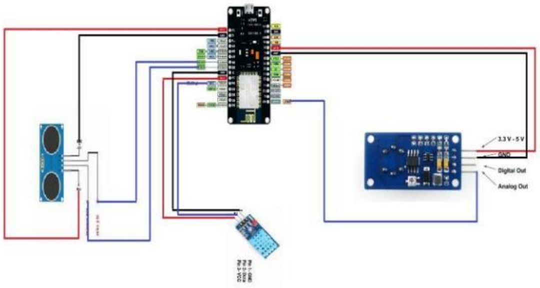


Fig 6. Circuit Diagram of VHMS

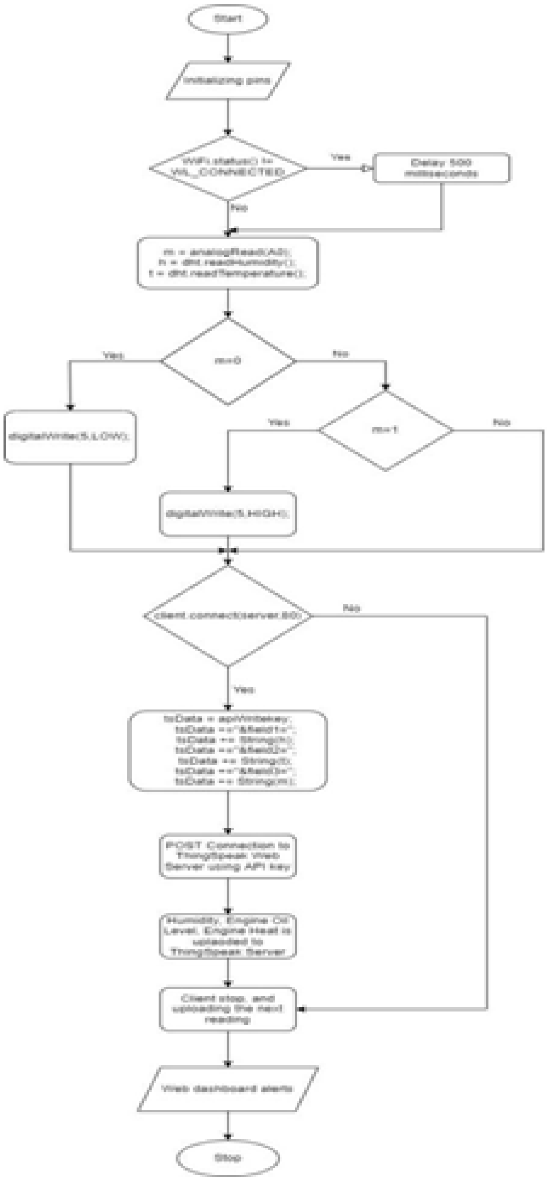
III. SOFTWARE

A. Vehicle Health Monitoring System Dashboard:

Because of the expense and lack of space, automobiles in today's world are not equipped with many sensors. The issue is predicted only when it happens but not priori. So, once if the issue occurs the issue resolving has also became a serious cause. People at times where in crisis to find the repair shop too, for resolving the trouble. To all these causes and troubles with the vehicle system working today, we came up with a proposal.

We proposed a system which calculates the health of the vehicle using different sensors. These sensors detect the problems that arise from time to time and notify the car owner of the findings. We've included a gas sensor to catch the vehicle's dangerous carbon emissions. We have a temperature sensor to measure the engine's heat. We have an ultrasonic sensor that detects the presence of engine oil. All of these sensors are connected to the microcontrollers, which are the brains of the system. The microcontroller has been programmed, and it sends data via Thingspeak server to the VHMS dashboard. This dashboard combines the data collected by the sensors with the car's restricted value needs to provide a vehicle health output for the viewer.

Flow Chart of VHMS:



B. Thingspeak Web Server :

ThingSpeak is an open source IoT platform consisting of a central server that collects, processes, and analyses data, as well as libraries and APIs for IoT devices. With both free and premium price options, the open-source central server may be deployed locally or in the cloud. ThingSpeak is distinguished by its exceptionally user-friendly interface for data processing and display, due to Matlab's support in cloud deployment, which includes graphic output and various Matlab toolboxes. In addition to basic, free Matlab support given by ThingSpeak, access to such toolboxes is granted with a premium MathWorks subscription. The platform supports a wide range of devices, including smartphones and all major IoT hardware platforms like Arduino and Raspberry Pi.

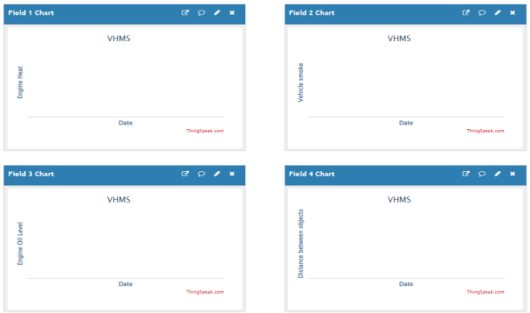


Fig 7. ThingSpeak Web Server Interface

IV. RESULTS

The ESP8266 microcontroller is connected with the 3 sensors (Gas, Temperature, Ultrasonic) which is again connected to the device or battery for the current to pass and also connected to WI FI for noting the data to the ThingSpeak Web Server (Fig 9). From this Web Server the data is collected and passed to the dashboard.

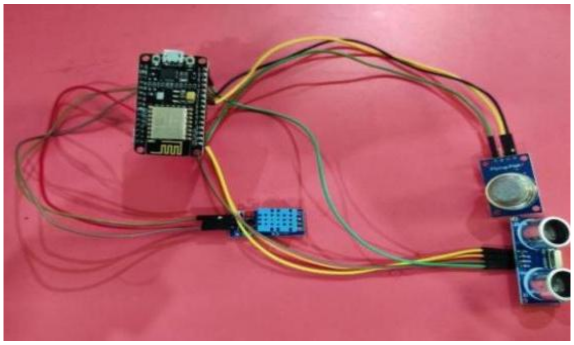


Fig 8. Final Product of VHMS

The sensors collect the data and channels it to show it on the ThingSpeak Web Server in the form of graphs (Fig 9.)

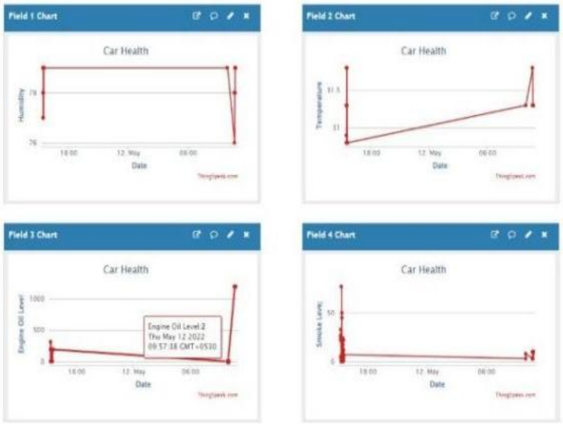


Fig 9. ThingSpeak Web Server after sensing the objects

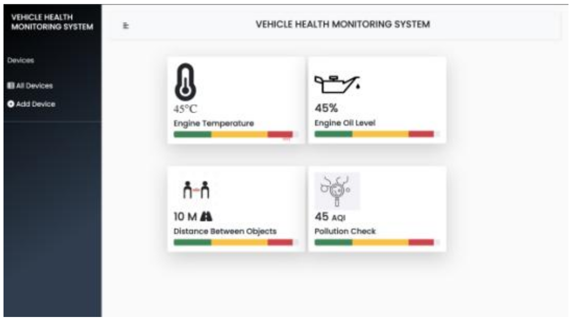


Fig 10. Web Dashboard after collecting data from ThingSpeak Web Server

V. CONCLUSION

Vehicle Health Monitoring System (VHMS) is implemented using Gas Sensor, Temperature Sensor, Node MCU to monitor the health of the vehicle, to alert user with the future cause of some problem prior its happening and to help them to avoid accidents too. By using this system, the driver is constantly observed by these senors and also the condition of the vehicle. This system is effective and can be used by each individual in their daily life.The platform supports a wide range of devices, including smartphones and all major IoT hardware platforms, such as Arduino, Raspberry Pi, Electric Imp, Particle Photon, and ESP8266, through web APIs that use both HTTP and MQTT communication protocols. Data may be exported in JSON, Comma Separated Values (CSV), and Extensible Markup Language

(XML) formats, or the ThingHTTP protocol can be used to integrate with external services.

REFERENCES

[1] Attendance Monitoring System using Amazon Web Services (A.W.S): https://ijirt.org/ Article? manuscript=149403

[2] Vehicle Monitoring System: https://www. irjet.net/archives/V6/i3/IRJET-V6I3547.pdf

[3] IoT based Advanced Accident Monitoring System using ThingSpeak Web Server: https://www.ijert.org/iot-based-advanced-accident-monitoring-system-using-thingspeak-web-server

[4] Vehicle Health Monitoring System: https:// www.ijera.com/papers/Vol2_issue5/GJ2511621167.pdf

[5] Internet of Things and NodeMCU: https:// www.researchgate.net/publication/337656615_Internet_of_Things_and_Nodemcu_A_review_of_use_of_Nodemcu_ESP8266_in_IoT_products