Energy Conserving Mechanics for Remote Monitoring of Unit Operation in IOT System

Asha A. Gaikwad*

People's Education Society' College of Engineering, Aurangabad, Maharashtra, India.

Abstract:

Recent technological advances in the electronics and communication technologies have significantly attracted researchers towards the Internet of things (IoT). IoT has become an efficient tool to monitor remote systems deployed to study the various physical parameters of the environment. IoT brings a lot of technical and automated solutions to assist the maintenance of unbalanced electrical systems. This piece of research work unfurl the implementation of NodeMcuESP8266 as an efficient IoT hardware for remote monitoring of electrical equipment. It has become popular and it can be utilized for various special purpose controlled systems since it is cost effective and developed open source support of the ESP8266 Wi-Fi Chip. $Ni_{0.35}Mn_{0.15}Cu_{0.2}Zn_{0.3}Fe_2O_4$ ferrite core was used in the energy harvesting device. This research work reveals the concept of remote monitoring of a unit operating through energy harvester and its performance via real-time online parameters display on cloud. The average current consumption of NodeMcu is 9.8 mA and operated satisfactorily while using the supply from the energy harvesting system. The data is presented graphically which can also be stored on cloud and can be retrieved by authorized users from anywhere.

Keywords: Internet of things, Remote monitoring, Current, Voltage, Energy harvesting.

I. Introduction

Recently, the energy management is an imperative factor for any institution, thus, the energy usage need always monitored for its controlled use. In this context, the application of Internet of Things (IoT) towards an effective monitoring for the electric power management and related parameters can be utilized. In general, at the low voltage application; electrical measures such as voltage, current, power, power factor, energy, and frequency in electrical network structure fluctuates due to the change in the load, instabilities and other anomalous states. The variation in electrical measures must be timely noticed otherwise it may cause a stern problem for the entire electrical network system. Henceforth, it is essential to control the variation of electrical parameters timely and properly in command to make operative results[1]. Thus, a remote monitoring system is important to update the variation in the rate of electrical parameters in order to avoid the damages to the electrical network system. There are numerous ways to manage the use of electricity in the low and high voltage systems by employing wireless sensor technology (WSN). Nowadays, IoT is the best coding and prominent option for remote monitoring [2 -6]. IoT is a set of networks which comprehend physical appliance and additional electronics embedded system, software, sensors, actuators and network connectivity, and enable the environment to gather and exchange or store information on the cloud. The IoT allows any physical information to be sensed and monitored remotely at the existing infrastructure to improve the effectiveness, correctness and budgetary benefit without any harm to the environment. The IoT is an organized structure associated with the physical devices and advanced sensors to transmit the data over the remote intermediate without any interference.[3]

D. Despa el al. [1] have presented the measurement of numerous data of a three phase system where the data was processed by the Arduino to accumulate the data on the database server, and subsequently transferred remotely by implementing IoT application. Kalaivanan et al. [4] have described IoT applications with the Zig-Bee technology which is implemented for smart home systems. Wesley et al. [5] have accepted the raspberry pi as a miniature programmable computing device to monitor and control the air conditioning appliances and lighting units. The theft of water was supervised by monitoring the flow rate by utilizing an embedded system in conjunction with web based mobile application.[7] A model was developed and contrived to monitor the various gasses concentration in the air with the help of a set consisting ATMEGA328 and GSM based module to transmit the data to the online server.[8] S Ananda Kumar et al. [9] have reported the application of IoT in healthcare monitoring system for making a proper decision in critical condition. In this monitoring system, all the sensors have been mounted on wearable devices to gather the data and subsequently transmitted to the cloud that include Bluetooth Wi-Fi, ZigBee and RFID.[9] Likewise, Jie Wan el al. [10] have reported a novel system that include a Wearable IoT-cloud-based health monitoring system for real-time self-health monitoring. It adopts the sensing signals from the body network to assist the real-time health monitoring. M. A. Patil et al. [11] have developed a farm monitoring system which consists of data monitoring of soil pH level, moisture temperature and humidity to take proper action for maintaining the plant condition using smart phone based IoT application.

Herein, this work presents the implementation of NodeMcu [12-15] as an IoT device with the aid of energy harvesting application. In addition, this invention was concentrated on reducing the energy consumption required for the operation of NodeMcuESP8266 without compromising the facility of data transmission to the cloud. It can avail the convenience of power supplied from the low power energy harvesting system since the NodeMcu operation is energy conservative.

The main concept of the present work is to use internet connectivity to upload the data of interest to the cloud which can be accessed by authorized users from anywhere. In this method, the definite current drawn from the mains will continuously sense using the current sensor (ACS712), whereas at the same a microcontroller-based circuit monitors the operation parameters and keeps on averaging. At a pre-set interval the NodeMcu will transmit the average values of the crucial parameters of interest and this data will systematically be stored on cloud which will be displayed in graphical format on Think speak web server. The microcontroller and sensor will be powered by the energy harvester device where the monitoring intended to implement. The schematic presentation of the arrangement of the components are shown in figure 1.

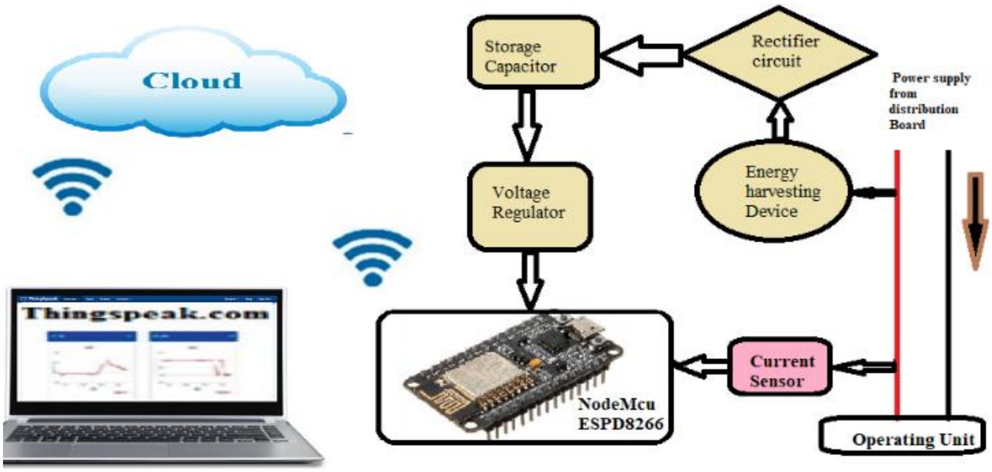

Figure 1. Arrangement of component required for monitoring based on IoT in connection with Energy harvesting systems.

There are numerous sources for energy harvesting including; thermal[16], mechanical vibration[17-18], air flow [19,20], ambient light [21], radio frequency [22] etc. The induction-based energy can be harvested in alternating magnetic flux around the phase or neutral cable [23], therefore it is a suitable energy source for the unit operation to sense electronic circuits. In current transformer based energy harvesting output depends on its size and physical dimension [24].

The NodeMcu will be fed by the power received from the energy harvester which is placed on the supplying cable of the same unit where the monitoring is implemented. The power obtained from the energy harvester will be converted into a suitable form of energy. Such energy will then be stored in a storage capacitor to meet the power requirement crests and can supply large power abundances for the short period of durations to the NodeMcu during the data transmission (wake-up) period. NodeMCU ESP8266 is a low-cost hardware structure similar to Arduino consisting of the main component as ESP8266 Wi-Fi chip which has programmable pins. It can be programmed through multiple programming environments. [7]. The value of current supplied to the load will be received from the current sensor connected in series with load which will be sent to the cloud and ThingSpeak.

The amount of power is limited in the energy harvesting applications and thus it is important to thoughtfully utilize the power extracted from the harvesting technique. In general practice, typical power levels in the order of 80 – 180 mA at 5 volts are relatively high to transmit the sensed data to the internet through devices such as NodeMcu ESP8266. Such a power level is quite high for typical energy harvesting systems. Hence, the purpose of the proposed method is to reduce the power consumed by the Node MCU, so as to be powered and operated using the harvested energy.

II. Methods and materials for the synthesis of $Ni_{0.35}Mn_{0.15}Cu_{0.2}Zn_{0.3}Fe_2O_4$ ferrite core

$Ni_{0.35}Mn_{0.15}Cu_{0.2}Zn_{0.3}Fe_2O_4$ ferrite core was used in the energy harvesting device. Thus, $Ni_{0.35}Mn_{0.15}Cu_{0.2}Zn_{0.3}Fe_2O_4$ was synthesized by sol–gel auto-combustion method. Analytical grade nickel nitrate ($Ni(NO_3)_2{\cdot}6H_2O$), copper nitrate ($Cu(NO_3)_2.6H_2O$), zinc nitrate ($Zn(NO_3)_2{\cdot}6H_2O$), manganese nitrate

($Mn(NO_3)_2 \cdot 6H_2O$) and iron nitrate ($Fe(NO_3)_3 \cdot 9H_2O$) were dissolved in distilled water to obtain a mixed solution. Reaction procedure was carried out in the air atmosphere without protection of inert gases. The molar ratio of metal nitrates to citric acid was taken as 1:3. The metal nitrates were dissolved together in a minimum amount of double distilled water to get a clear solution. An aqueous solution of citric acid was mixed with metal nitrates solution, then ammonia solution was slowly added to adjust the pH at 7. The mixed solution was kept on a hot plate with continuous stirring at 90 °C. During evaporation, the solution became viscous and finally formed a very viscous brown gel. When finally all water molecules were removed from the mixture, the viscous gel began frothing. After a few minutes, the gel automatically ignited and burnt with glowing flints. The decomposition reaction would not stop before the whole citrate complex was consumed. The auto-combustion was completed within a minute, yielding the brown-colored ashes termed as a precursor. The as-prepared powders of all the samples were sintered at 600 °C for 4 h to get the final product. The powder was granulated using 2 % PVA as a binder and was pressed into discs and toroids (internal and external diameters of 4 and 10 mm, respectively, and a thickness of 2 mm), with an applied pressure of 5000 kg/cm^2 to obtain the toroid-shaped samples for permeability measurements. The prepared pellets were sintered at 1250 °C for 12 h.[25]

III. Operation of NodeMcu

As discussed above, the NodeMcu requires higher power as compared to the power obtained from the energy harvesting system. To overcome such an issue, an experiment was carried out to scrutinize the power requirement by the NodeMcu ESP8266 in deep-sleep mode of operation as asserted below.

Initially for testing purposes, the data was collected per second and was transmitted serially to the serial monitor for verification. Subsequently, the time delay was increased to 2 minutes for the purpose of transmitting the real time data to the cloud and to display on a web page. The controlling program for the remote monitoring of process parameters like current was developed in embedded C++ for NodeMcu and was uploaded to NodeMcu using Arduino IDE. The Arduino IDE needs prior configuration with certain parameters to make it compatible with NodeMcu. The NodeMcu was made to read the data of input parameters every two seconds and transmit the data to the cloud for display and storage. The NodeMcu was put into sleep mode between two successive data acquisitions and transmissions. This was necessary to conserve the energy since the NodeMcu was energized from an energy harvesting power supply with a limited current capabilities. The temporal current consumption was monitored using a small resistance (2.7 Ω) connected in series with the power supply in the ground lead. The voltage across this series resistance was monitored using a digital storage oscilloscope interfaced to a computer in order to monitor the current variations during different activities and operation. The operation was performed through following stages.

Stage1: Active mode (Wake-up): In this mode ESP8266 can receive and transmit the data and consumes power. Typical values of current consumption are in the range of 60 to 65 mA.

Stage 2: Deep sleep mode: In this mode, most of the functions of the NodeMcu are disabled to reduce the current consumption that is useful since the current sourcing capabilities of the power supply are limited and power saving is intended. It is useful between two consecutive data transmissions when there is no effective activity [26].

In present experiments, the voltage across the current sensing resistor was recorded and the waveforms were stored on a computer with the help of an oscilloscope (Hantek 6022BE 20MHz 48MSa/s). The actual data was also saved in CSV files for further analysis when needed along with the wave form. Figure 2 showed the flow chart for the NodeMcu operation.

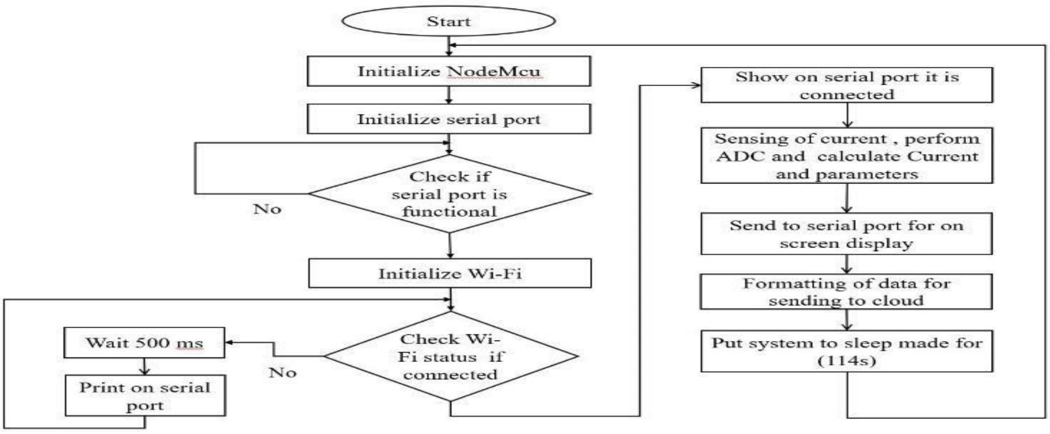

Figure 2. Process flow chart of NodeMcu

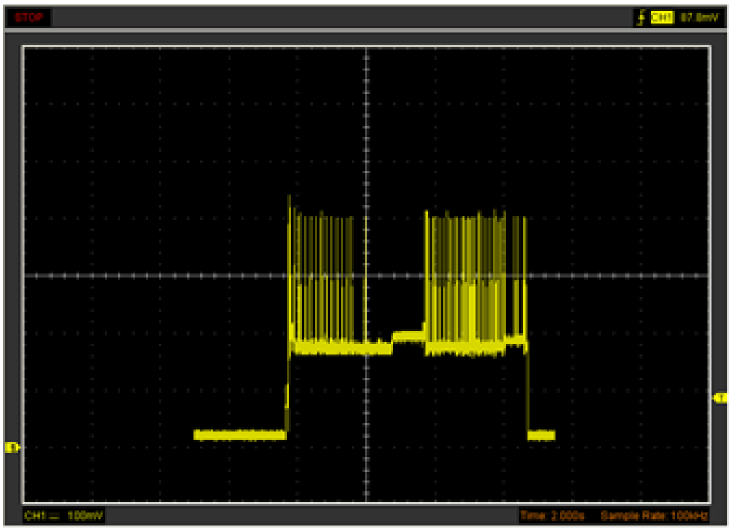

Figure 3. Oscillogram of voltage across current sensing resistor of 2.7 Ω in Deep sleep mode operation of NodeMcu

IV. Results and discussion

Results obtained in the investigation of current required by the said IoT device (NodeMcu8266) is elucidated as below as per the mode of operation.Figure 3 showed the voltage across the current sensing resistance of 2.7 Ω. The sweep rate was 2 s per division (x-axis value) which showed that the NodeMcu is in the wake-up condition for a period of ~7 seconds, whereas it remains in sleep mode for 113 seconds. The y – axis scale bar is 100 mV per division which means that in the deep sleep period the voltage across the sensing resistor of 2.7 Ω is 20 mV. On the other hand the voltage across the current sensing resistor of 2.7 Ω is 175 mV for the wake-up period. Accordingly, the calculated current was ~7.4 mA and 65 mA for the sleep and wake-up periods, respectively. System transmitted the data to the cloud every two minutes and therefore out of the 2-minute period it was awake for 7 s and was in sleep mode for 113 s. Therefore, the average current consumed by the system was 10.76 mA. The current was calculated utilizing the following equations:

$$I_T = \frac{V_d}{R}$$

$$I_{Avg} = (T_s \times I_s) + (T_w \times I_w)$$

where I_T= Current consumed by the NodeMcu, V_d= The voltage drops across the sensing resistor,
R= Value of current sensing resistance,I_{AVG}= Average current consumed,T_S = Sleep time,I_S =Current in sleep time,
T_W= Wake-Up time,
I_W =Current in wake-up time.

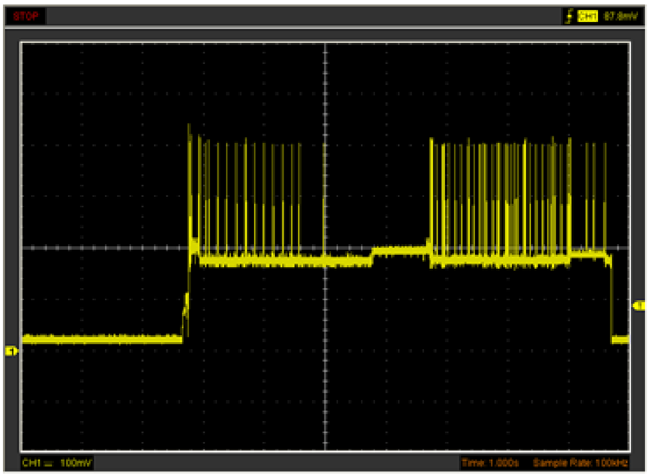

Figure 4. Oscillogram of voltage across current sensing resistor of 2.7 Ω set at 1s per Division

Both the waveforms shown in Figs. 3 and 4 are very much identical apart from the change in time scale. A small kink was observed at the beginning of the rising part corresponding to the initial housekeeping tasks performed during wake up mode. Thereafter, it performed several tasks including initializing the ports for serial communication and Wi-Fi connectivity. During this period, the current was at a higher level. After completing these tasks the system sends the data to a serial terminal to the USB port, which is helpful in monitoring the performance if needed (Fig. 4). There are several spikes observed in the initial bunch representing the current consumption during the transmission through the serial port. After the serial port transmission, it starts attempting to connect to the Wi-Fi network for sending the data to the cloud. The gap between the two bunches of spikes corresponds to the period where Wi-Fi is connected which is clearly seen by a slight increase in current as indicated by a small increase in the voltage across the sensing resistor. As can be seen by the second bunch of spikes, it started sending the data to the internet cloud after the internet connectivity was established and necessary authentications were done. After sending the data to the cloud through the Wi-Fi was completed the entire circuit goes into the deep sleep mode again that substantially reduced the consumption of the current.

A closer look at Figs. 3 and 4 suggested that the first bunch representing the serial communication via USB port is not essential during the actual operation of the present IoT based real time operation monitoring process and hence can be completely eliminated. The major advantage of eliminating such serial communication via USB is to substantially reduce the system' awake time period which in turn helped to reduce the average current consumption.

The voltage drop across the current sensing resistor of 2.7 Ω and the voltage at TX pin of the NodeMcu is presented in Fig. 5 for the comparison between the activity on TX pin and the actual current consumption of the NodeMcu. It is interesting to note that very few spikes were observed in the TX pin signal (most of the serial text transmissions were stopped), whereas the number of counts for spikes in the voltage signal are significantly high. The major reason for such difference is related to the observed voltage spikes during the transmission and while the NodeMcu is performing a number of tasks like sensing, ADC conversion and other numerical calculation procedures. Time consumed for the wake mode can be reduced if the number of tasks such as configuration and sending data on serial port exceeds the capacity of operations. It can be observed from the green trace in Fig. 5 that the wake mode is considerably reduced after eliminating the configurations which were not needed for transmitting the data (think spike) on the cloud. Herein, time of ~5 s was consumed to bring the operation in wake mode. This reduction in time spent in wake mode contributes towards the current saving requirements that resulted in reduction of the average current requirement to 9.8 mA. The voltage across the current sensing resistor is shown in green with a voltage per division of 200 mV per division, whereas yellow trace representing the voltage at the TX pin of NodeMcu using 1 V per division on the y-axis scale.

Figure 6 showed the operation of serial data transformation to the serial monitor for verification of monitoring parameters. In this work, current supplied to the electrical unit was sensed by the current sensor ACS117, and transmitted to the serial monitor and cloud by using the NodeMcuESP8266. The current consumption was updated for every 2 min on Thing Speak server as is shown in figure 7.

Figure 8 showed the value of monitoring current with time and without any unnecessary serial data on the serial monitor. A closer look at Figs. 6 and 8 indicated that the serial communication via the USB port is not essential during the actual operation of the present IoT based real time operation monitoring process. Henceforth, by reducing the extra burden on the NodeMcu it is possible to upload the monitored data after every 2 minutes on the Thing Speak server .

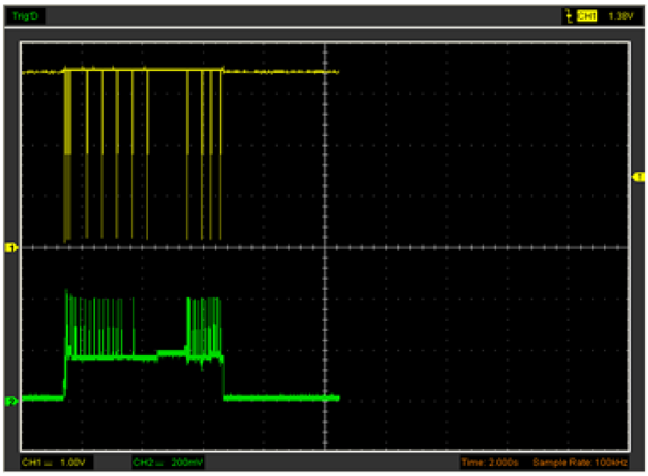

Figure 5. Waveforms from sensing resistor and TX pin of NodeMcu. Time base used is 2 s per division on the x-axis.

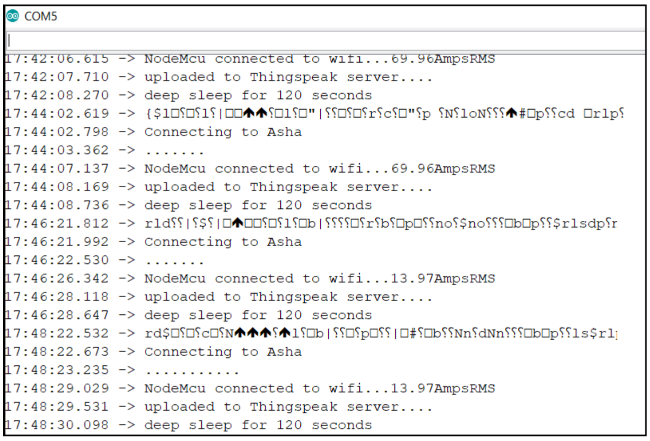

Figure 6. The serially data transferred to the serial monitor

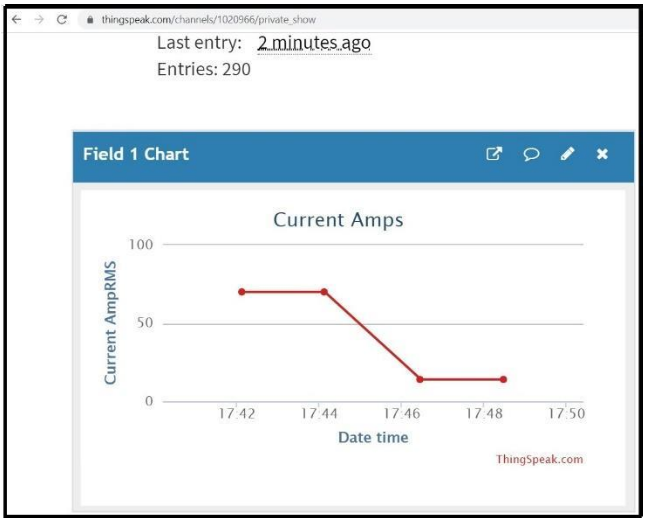

Figure 7. Monitoring Data uploaded on Think Speaks after every 2 minutes.

Figure 8. Serial Data transmitted on serial monitor after reducing the Burden on NodeMcu

The performance of the present method is compared with the method reported by Kalaivanan et al. [4], where the ZigBee router, ZigBee coordinator and ESP8266 Wi-Fi module was interfaced with Aurdino to established a set up for transmitting the data on the cloud continuously after every 30 sec. The current requirement for ZigBee is ~54 mA in the transmission mode [24] which results in higher power consumption after every 30 sec. Similarly, when compared with M. Bharathi [8], where the authors therein have preferred the source for Wi-Fi provided by the GSM module to launch the communication between the sensors and the user by means of internet as interface. It was seen that, typically, GSM modules consume 1.5 mA and 2 A in sleep mode and transmission mode, respectively, which required bulk power supply for the data processing on the cloud. Also, in a web-based monitoring system [14] the statistics of air quality, temperature and humidity in three food stores at three remote locations had been shared on the cloud with a time interval of 60 s and the ESP module had been supplied by the 9V battery. In this monitoring system, though the interim between the two transmissions is 60 s, there is an obligation to check the continuity of supply to the ESP module and the replacement of battery. Henceforth, in the proposed method using the NodeMcu in deep sleep mode the problem of high current consumption as well as battery replacement for remote monitoring is reduced.

V. Conclusions

In this research work, remote monitoring based on IoT by utilizing the NodeMcu was tested in active/wake and deep sleep modes. It has been revealed that the best results were found when the NodeMcu ESPD8266 was operated in deep-sleep mode for a period of 115 s out of 120 second and 5 seconds in wake mode to transmit the information on the cloud. With the elimination of excess burden on the microcontroller and good internet connectivity to the NodeMcu ESPD8266, it is possible to reduce the time spent in data transmission to 5 s that also fulfills the requirement of lower current consumption. It was also observed that it reduced the average current consumption in the range of 9.8 mA and it will be further applicable towards the use of power for energy harvesting application. It is also concluded that NodeMcuESP8266 has influential processing and storage abilities. This work can be extended for monitoring and controlling the different parameters like temperature, humidity and gas leakage in the smart home application.

Future scope

The operation of NodeMcu on energy harvesting is also very helpful to monitor the different parameters at remote places where humans could not maintain the supply for the operation of NodeMcu. This will be very significant to reduce frequent replacement of batteries which supplies the low power to the IoT device at remote places. Also, it will provide an almost green energy supply at lower cost with minimum or no harmful environmental impact.

References

[1] Despa, D., Nama, G. F., Muhammad, M. A., & Anwar, K. (2018, April). The implementation Internet of Things (IoT) technology in real time monitoring of electrical quantities. In *IOP Conference Series: Materials Science and Engineering* (Vol. 335, No. 1, p. 012063). IOP Publishing.

[2] Khiat, A., Bahnasse, A., Bakkoury, J., El Khaili, M., & Louhab, F. E. (2019). New approach based internet of things for a clean atmosphere. *International Journal of Information Technology*, *11*(1), 89-95.

[3] Udgata, S. K., & Suryadevara, N. K.(2020). IoT and Sensor Network. In *Internet of Things and Sensor Network for COVID-19* (pp. 19-37). Springer, Singapore.

[4] Kalaivanan, S., & Manoharan, S. (2016). Monitoring and Controlling of Smart Homes using IoT and Low Power Wireless Technology. *Indian Journal of Science and Technology*, *9*(31), 1-9.

[5] Chen, J., Yu, S., Jiang, Y., & Wang, G. (2020, August). Research on Fault Transmission Method of Switchgear Based on Internet of Things. In *Journal of Physics: Conference Series* (Vol. 1617, No. 1, p. 012022). IOP Publishing.

[6] Hartman W. T., Hansen, A., Vasquez, E., El-Tawab, S., & Altaii, K. (2018, April). Energy monitoring and control using Internet of Things (IoT) system. In *2018 Systems and Information Engineering Design Symposium (SIEDS)* (pp. 13-18). IEEE.

[7] Mirji S. C., Sanjay N. C., Sowmya C. H., Makandar S. A., and Rakshitkumar G. S.(2020).IoT Based Digital Water Supply System Using Raspberry Pi. REVA,29(10)0 pp. 8607–8612, 2020.

[8] Bharathi M.,Padmaja N., and Leela Rani D.,(2020).IoT based smart air pollution monitoring system. Int. J. Adv. Sci. Technol., 29(4), pp. 687–693.

[9] Kumar, S. A., & Mahesh, G. (2020, June). IoT in Smart Healthcare System. In *Internet of Things for Healthcare Technologies* (pp. 1-19). Springer, Singapore.

[10] Wan, J., Al-awlaqi, M. A., Li, M., O’Grady, M., Gu, X., Wang, J., & Cao, N. (2018). Wearable IoT enabled real-time health monitoring system. *EURASIP Journal on Wireless Communications and Networking*, *2018*(1), 298.

[11] Patil, M. A., Adamuthe, A. C., & Umbarkar, A. J. (2020). Smartphone and IoT Based System for Integrated Farm Monitoring. In *Techno-Societal 2018* (pp. 471-478). Springer, Cham.

[12] Al Dahoud, A., & Fezari, M. (2018). NodeMCU V3 For Fast IoT Application Development.

[13] Parihar Y.S.(2019).Internet of Things and Nodemcu: A review of use of Nodemcu ESP8266 in IoT products. J. Emerg. Technol. Innov. Res., 6(6), pp. 1085–1086.

[14] Aziz, D. A. (2018). Webserver based smart monitoring system using ESP8266 node MCU module. *International Journal of Scientific & Engineering Research*, *9*(6), 801-808.

[15] Shelke P., Kulkarni S., Yelpale S., Pawar O., and Singh R.(2018) .A NodeMCU Based Home Automation System.IJRET 5(06), (pp. 127–129).

[16] Giwa, S. O., Nwaokocha, C. N., Layeni, A. T., & Olaluwoye, O. O. (2019). Energy harvesting from household heat sources using a thermoelectric generator module. *Nigerian Journal of Technological Development*, *16*(3), 127-134.

[17] Wei, S., & Hu, H. (2017). Experimental analysis of optimized impact interval/vibration period ratio on piezoelectric energy harvesting. *Ferroelectrics*, *506*(1), 136-143.
[18] Boisseau, S., Despesse, G., & Seddik, B. A. (2012). Electrostatic conversion for vibration energy harvesting. *Small-Scale Energy Harvesting*, 1-39.
[19] Nurmanova, V., Bagheri, M., Phung, T., & Panda, S. K. (2018). Feasibility study on wind energy harvesting system implementation in moving trains. *Electrical Engineering*, *100*(3), 1837-1845.
[20] Aljadiri, R. T., Taha, L. Y., & Ivey, P. (2017, April). Wind energy harvesting systems: A better understanding of their sustainability. In *2017 3rd International Conference on Control, Automation and Robotics (ICCAR)* (pp. 582-587). IEEE.
[21] Pathak, M. J. M., Sanders, P. G., & Pearce, J. M. (2014). Optimizing limited solar roof access by exergy analysis of solar thermal, photovoltaic, and hybrid photovoltaic thermal systems. *Applied Energy*, *120*, 115-124.
[22] Sharma T. and Saini G.(2016).A Survey on RF Energy Harvesting from Mobile Towers. vol. 3, no. 1, pp. 3–5, 2016.
[23] Gaikwad, A. A., & Kulkarni, S. B. (2019). Energy Harvesting Based on Magnetic Induction. In *Information and Communication Technology for Intelligent Systems* (pp. 363-370). Springer, Singapore.
[24] Gaikwad AA, Kulkarni SB. Evaluation of dimensional effect on electromagnetic energy harvesting. Procedia computer science. 2018 Jan 1;143:58-65.
[25] Kazeem, O. O., Akintade, O. O., & Kehinde, L. O. (2017). Comparative Study of Communication Interfaces for Sensors and Actuators in the Cloud of Internet of Things. *Int. J. Internet Things*, *6*(1), 9-13.

[26] Gaikwad, A. A., & Kulkarni, S. B. (2021). Enhanced magnetic and permeability properties of Mn-substituted NiCuZn nanoparticles for ferrite core application. *Journal of Superconductivity and Novel Magnetism*, *34*(9), 2405-2414.