IoT-based wireless induction motor monitoring

Conference Paper · September 2017

DOI: 10.1109/ET.2017.8124386

2 authors, including:

Mehmet Sen
Uludag University
8 PUBLICATIONS **17** CITATIONS

SEE PROFILE

IoT-Based Wireless Induction Motor Monitoring

Mehmet Şen and Basri Kul
Department of Electrics and Electronics, Vocational School of Technical Sciences,
Uludağ University of Bursa
Nilufer,16059, Bursa, Turkey
mehmetsen@uludag.edu.tr; basri@uludag.edu.tr

Abstract – **In this study, a factory induction motor (IM) was monitored with wireless TCP/IP protocol in order to detect and predict deviations from normal operating parameters before the occurrence of motor failure. In this way, the production process is not impeded and the required maintenance or replacement can be performed with the least possible disruption. In this study, the motor cycle, the current drawn by the motor and the motor voltage were read by the Hall-effect current sensor and the required power consumption was calculated. With this aim, the designed architecture read the accepted parameters of the motor and reported them to the central management software. The central management software operating in real time was then able to assemble these parameters and form predictive maintenance models.**

Keywords – **wireless Wi-Fi sensor monitoring, induction motor, motor parameters**

I. INTRODUCTION

In today's manufacturing industries, mechanical and electromechanical systems are driven by electric motors on the premises. The drivers of these motors are mostly on motor control and the predictive maintenance schedules of the motors are not calculated. Attempts are being made to maximize efficiency by using enterprise resource planning (ERP), especially in 7/24 production enterprises. However, unexpected failures not predicted by the ERP system can cause disruptions in the production process. In this study, the temperature, current, voltage, cycle, speed, frequency, torque and flux data of single and three phase induction motors (S / 3P-IMs) were read using TCP/IP protocol via Wi-Fi. By using the existing Internet network, these parameters were read and transferred to the central software without the need for additional wiring. The central software collected the parameters of all the motors and determined the necessary maintenance schedules. This system has been applied and used in a textile factory. Frequency-controlled motors have special 50/60 Hz filters in their structure which make it impossible to measure energy using the unified integrations developed for normal energy measurement. For this reason, the energy measurements were calculated by the processor.

A number of relevant studies carried out on the subject are briefly summarized here:

[1] Motor failure was detected by using different models to analyze the stator current. These included FFT, Hilbert-transform, continuous wavelet transform (CWT), discrete wavelet transform (DWT), the Wigner-Ville distribution (WVD) and instantaneous frequency (IF).
[2] A hybrid model was developed combining fuzzy min-max (FMM), neural networks and classification and regression tree (CART) models. The motor stator current was also applied with the model.
[3] Motor currents, acoustic vibrations and mechanical vibrations were analyzed wirelessly using the Hilbert-Huang transform.
[4] A different wavelet transform was used to remove errors caused by load variability in measurements made with motor current signature analysis (MCSA).
[5] Pattern recognition studies were carried out on motor currents and voltages and as a result, motor faults were classified using the hidden Markov model.
[6] An industrial wireless sensor network that transmits motor currents and vibrations wirelessly was applied and its behavior at different loads examined.
[7] In order to detect motor faults, an evaluation was carried out under unloaded and fully loaded conditions, and real-time motor currents and voltages were examined.
[8] Electrical devices were monitored via the wireless ZigBee network. Motor parameters included current, voltage and torque.
[9] Motor health was explored both mechanically and electrically. The study demonstrated that readings obtained by monitoring motor parameters could reveal information about the motor.

In the present study, the hardware was designed with the NXP LPC1769 cortex-m3 100 MHz ARM architecture. This processor included a hardware encoder reader and a 6-channel 12-bit ADC. The Wi-Fi communicated with the central software via UDP protocol with 802.11.bg. The central software was developed with C ++ to read motor parameters, apply mathematical models of prediction and present statistics. Details of the designed IoT induction motor monitoring hardware (IoT-IMM-H) and central monitoring system (CMS) software are described below.

978-1-5090-2883-2 /16/$31.00 ©2016 IEEE

II. MATERIALS AND METHODS

2.1. Proposed IoT-IMM-H structure

The overall structure of the system hardware is shown in Figure 1. Figure 2 shows the connections of the IoT-IMM-H. Figure 3 shows the physical form of the IoT-IMM-H equipment that was developed. The system has two components, hardware and software, which are described in the following sections.

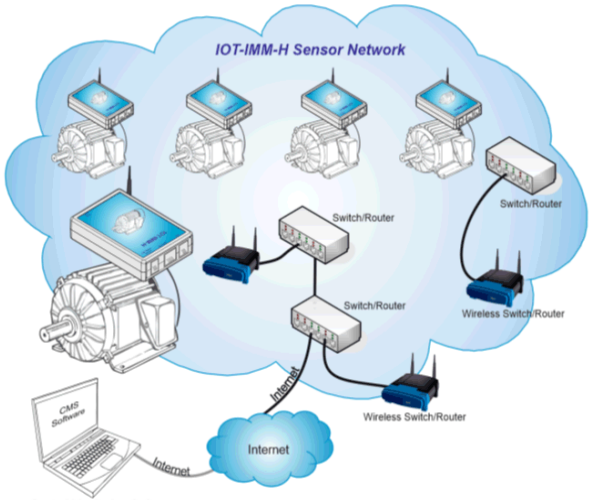

Fig. 1. IoT-based monitoring structure

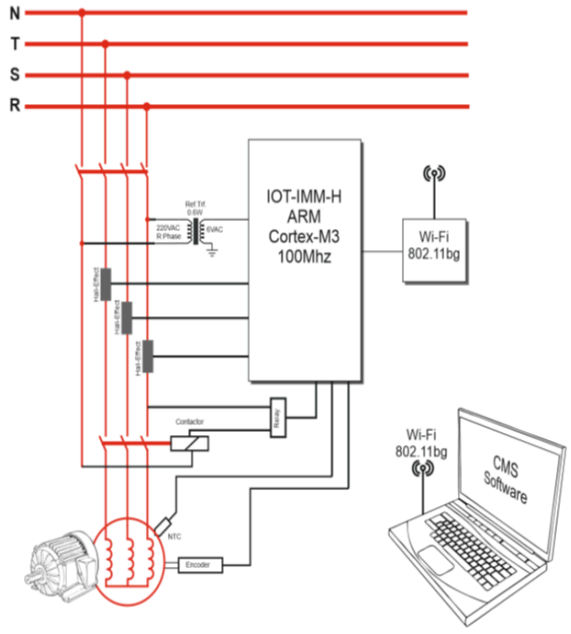

Fig. 2. Connections diagram of IoT-IMM-H

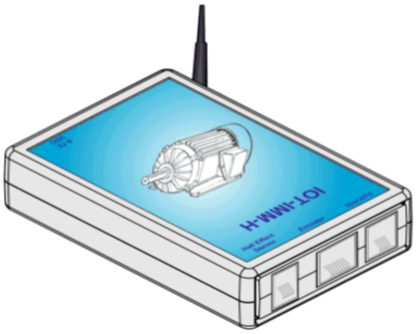

Fig. 3. IoT-IMM-H module

2.2. The Hardware

In this study, experiments were carried out on the three phase IM at ~ 0.3 kW/1400 rpm and 2.5 kW/1200 rpm. The designed hardware used an ARM NXP LPC1769 Cortex-M3 processor, a 802.11.bg Wi-Fi module, an ACS711 (-25 – +25A) Hall-effect current sensor, a high-speed Hall-effect proximity sensor for cycle measurement, NTC and a 220 VAC/6 VAC/0.6 W reference transformer. The entire system was fed externally with a 9-12 VDC power supply. The LPC1769 contained a 6-channel 12-bit ADC in its make-up. The ACS711 modules used for each phase were 2.5 V centered and produced output voltage of between 0-5 V, according to the current drawn. The ACS711 connection is shown in Figure 4. The ACS711 has a bandwidth of 100 kHz with negligible phase shift. The outputs of the ACS711 modules were connected to the analog input of the IoT-IMM-H. Moreover, the IOT-IMM-H was linked via encoder and NTC connections. The ACS711 Hall-effect current sensor, with -/+ 1% nonlinearity, met the desired accuracy. Figure 4 shows a schematic diagram of the ACS711 connection and the module.

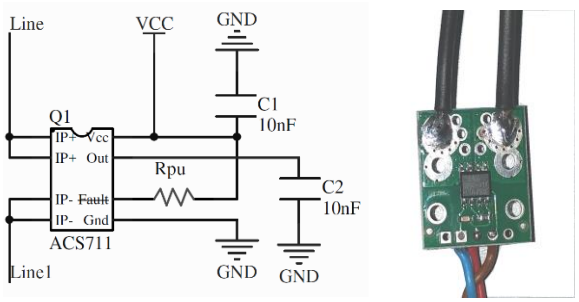

Fig. 4. ACS711 current connection diagram and module

Figure 5 shows the PCB design of the IoT-IMM-H, while the implemented design is shown in Figure 6. Table 1 gives the IoT-IMM-H connection details.

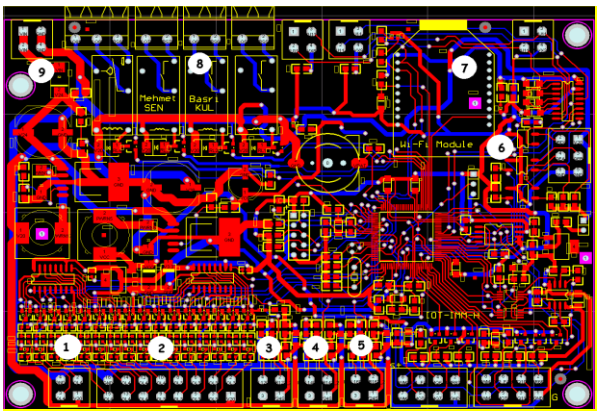

Fig. 5. IoT-IMM-H PCB artwork

Fig. 6. Implemented IoT-IMM-H without enclosure

TABLE 1. IoT-IMM-H connection details

1	Vref, NTC input	6	Encoder input, Phase A,B,Z
2	Proximity Inputs, 0-30VDC	7	Zigbee 802.11bg Wi-Fi module
3	Phase R, ACS711 Hall-effect sensor input	8	Relay outputs
4	Phase S, ACS711 Hall-effect sensor input	9	16-27VDC power input
5	Phase T, ACS711 Hall-effect sensor input		

The ACS711 modules were prefabricated and due to the long cable connection, a low pass filter was added on the IoT-IMM-H for each phase, as shown in Figure 7.

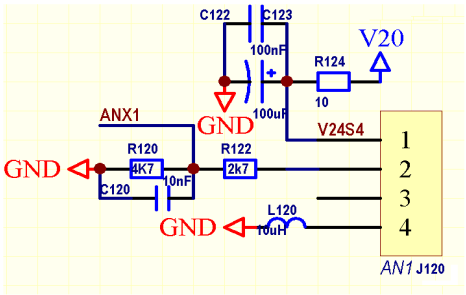

Fig. 7. Low pass filter on IoT-IMM-H for ACS711 modules

Figure 8 shows the encoder input on the IoT-IMM-H. The encoder inputs were isolated with an optocoupler. Because of the long cable length, EMI effects occurred at quadrature encoder outputs and both isolation and a low pass filter were required. Figure 9 shows the schematic part of the Wi-Fi module on the IoT-IMM-H. As the diagram of the whole circuit is quite extensive, only the necessary parts are shown.

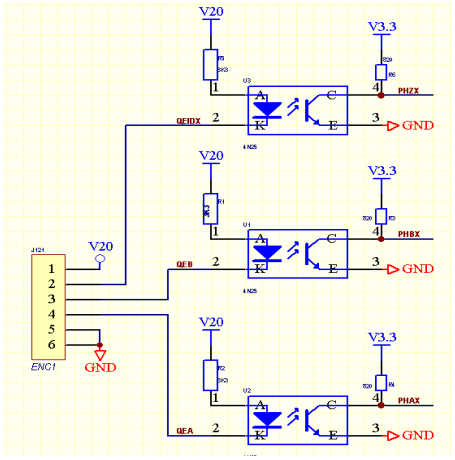

Fig. 8. IoT-IMM-H encoder input details

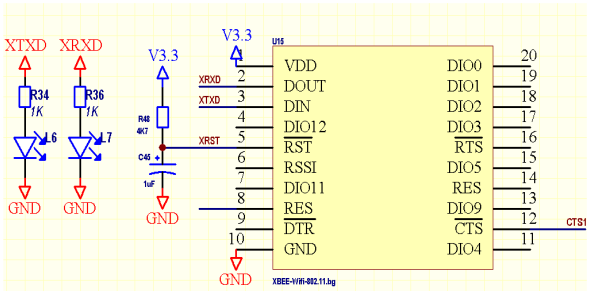

Fig. 9. IoT-IMM-H Wi-Fi-802.11.bg section

2.3. Embedded Software

The embedded software was developed with GNU-ARM C. In recent years, encoders, ADC and RTC have been read using the CMSIS-CM3 library developed as OpenSource for Cortex-M3 processors and supported by

the processor manufacturers. In addition, the energy measurements of each phase were made using the "FFT-Based Algorithm for Metering Applications" library developed by Freescale Semiconductor Inc. The reference voltage input on the IoT-IMM-H was taken on a transformer, using a single reference for the R, S, and T phases. The difference of 120 ° between the R, S, and T phases was equalized by the software. A single voltage reference was used to reduce cost. However, the currents belonging to each phase were taken separately. The sensitivity required for predictive maintenance was not needed since the difference between the phases was minimal.

In AC energy measurement, true power (P), measured power (W), apparent power (S), measured voltage (VA) and reactive power (Q) are calculated. Complex and real components are shown trigonometrically in Equations 1 and 2, and a depiction of energy calculation is given in Figure 10a. The "FFT-Based Algorithm for Metering Applications" is explained in detail in the AN4255 application note and will not be described further in this study. The embedded software performed Vrms and Irms energy calculations separately for each phase. The embedded software performed 3750 samplings for this process, at an average of 25 samplings per alternation (Fig. 10b). The root mean square (RMS) was used to determine the effective value of the AC signals. As a result, with its high processing power, the LPC1769 was easily able to perform the energy calculations for the three phases.

(1) (2)

$$I_{RMS} = \sqrt{\sum_{k=0}^{\frac{N}{2}-1} (I_{RE}^2(k) + I_{IM}^2(k))}$$

$$U_{RMS} = \sqrt{\sum_{k=1}^{\frac{N}{2}-1} (U_{RE}^2(k) + U_{IM}^2(k))}$$

where *IRE(k), URE(k)* are real parts and *IIM(k),UIM(k)* imaginary parts of *k.*

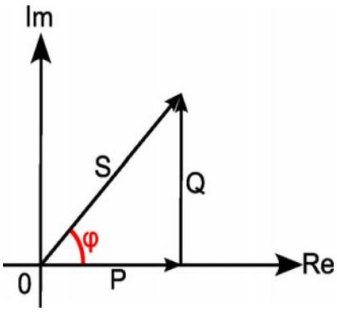

Fig. 10a. Energy calculation triangle: apparent power (S), true power (P), reactive power (Q) and CosFi

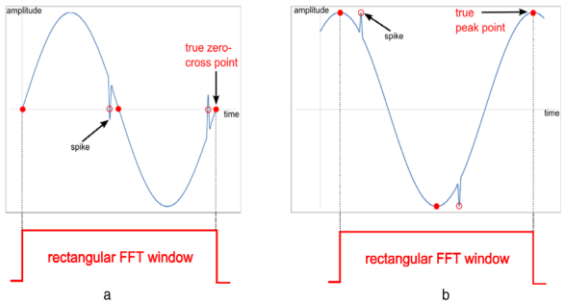

Figure-10b ADC sampling window

2.4. CMS Software

The CMS software reads by scanning the IoT-IMM-H modules and the other sensory data calculated by the IoT-IMM-H with UDP protocol. The UDP protocol is ideal for such sensor readings because handshaking is not necessary. If there are new packet losses, or if some of the IoT-IMMH modules are closed, the TCP/IP stack software does not waste extra processing power. The CMS software reads the data about the motor for about three seconds and then sends a UDP broadcast packet to all IoT-IMM-H modules. They receive this packet and send a data packet in response. Because the CMS software is written as "event driving", there is no packet loss, even if all IoT-IMM-H modules have the same data. The data packets from the IoT-IMM-H modules are immediately added to the pool of data packets. According to the data in the pool, a CMS thread updates the database with other predictive maintenance and energy data. A screenshot of the process can be seen in Figure 11.

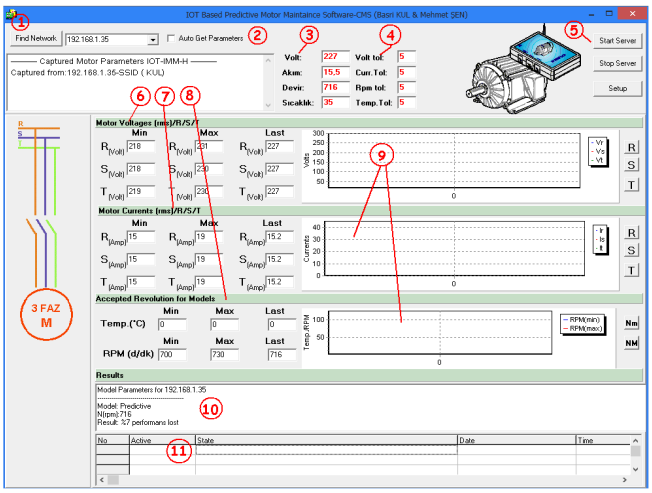

Fig. 11. CMS screenshot

TABLE 2. Description of CMS software

1	Find on Network Connectivity check	7	Min,Max, Last measured current for prediction at settled rev.
2	Update each captured data	8	Revolution setting for model
3	Last captured data	9	V,I,Rpm graphics
4	Fault tolerances	10	Model results
5	UDP server thread functions	11	Updated values and status
6	Min,Max, Last measured voltage for prediction at established rev.		

Motor current signature analysis (MCSA) was used in the study. Unlike other studies, a preloading method was chosen. In mass-production enterprises, the operation of the machines is repetitive. As a solution to the variable load problem, the method allowed the introduction of motor parameters during the processing of the relevant product. Data outside the user-specified tolerance limit was eliminated if it was not repeated. Power values according to motor parameters were updated via FIFO with 1024 samples. Then, after calculating the standard deviation over the 1024 samples in the table, data apart from the average deviation values were eliminated. The average of the remaining data was taken and compared with the learned values (%). The incoming data was repeated at 3-second intervals for all machines in all IoT-IMM-H units. As a result, the magnitude of deviation was regarded as the wear value of the motor.

III. EXPERIMENTAL STUDY

The system is being tested in a factory in Bursa, the textile center of Turkey. There are 18 looms in the factory working in the range of 2.7 kW - 4.5 kW. The maintenance schedule of the motors has been determined and the energy consumed per product calculated. The data indicate that better linear results are obtained when the stator winding temperature of the motor is included in the model.

IV. CONCLUSION

The system developed in this study has yielded successful results for production environments where there is little variation in the load on the motors. This learning method can be used for motor power ratings, especially for 7/24 machines. This study has provided statistics not only for creating mathematical models but also for enabling the CMS operator to establish a motor maintenance schedule.

REFERENCES

[1] A. Pilloni et al. *Fault detection in induction motors*. AC electric motors control: Advanced Design Techniques and Applications (2013): 275-309.

[2] M. Seera et al. *Fault detection and diagnosis of induction motors using motor current signature analysis and a hybrid FMM–CART model.* IEEE Transactions on Neural Networks and Learning Systems 23.1 (2012): 97-108.

[3] E. T. Esfahani, S. Wang, V. Sundararajan. *Multisensor wireless system for eccentricity and bearing fault detection in induction motors*. IEEE/ASME Transactions on Mechatronics 19.3 (2014): 818-826.

[4] K. M. Siddiqui, V. K. Giri. *Broken rotor bar fault detection in induction motors using wavelet transform.* International Conference on Computing, Electronics and Electrical Technologies (ICCEET), IEEE, 2012.

[5] A. Soualhi et al. *Fault detection and diagnosis of induction motors based on hidden Markov model.* Electrical Machines (ICEM), 2012 XXth International Conference on. IEEE, 2012.

[6] L. Hou, N. W. Bergmann. *Novel industrial wireless sensor networks for machine condition monitoring and fault diagnosis.* IEEE Transactions on Instrumentation and Measurement 61.10 (2012): 2787-2798.

[7] M. Irfan et al. *An on-line condition monitoring system for induction motors via instantaneous power analysis.* Journal of Mechanical Science and Technology 29.4 (2015): 1483-1492.

[8] M. R. Mikhov et al. *An application of wireless standards for remote monitoring of electric drive systems.* International Journal of Engineering Research and Development 2.12 (2012): 30-36.

[9] K. M. Siddiqui, K. Sahay, V. K. Giri. *Health monitoring and fault diagnosis in induction motor-a review.* International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering 3.1 (2014): 6549-6565.