JOURNAL OF ENGINEERING RESEARCH AND TECHNOLOGY, VOLUME 10, ISSUE 1, MARCH 2023

Received on (12-06-2022) Accepted on (11-11-2022)

Luenberger Observer-Based Speed Sensor Fault Detection: real time implementation to DC Motors

Moayed Almobaied[1], Yousef Al-Mutayeb[2]

1 Department of Electrical Engineering, Islamic University of Gaza, Gaza, Palestine.

2 Department of Engineering Sciences, University College of Science and Technology, KhanYounis, Palestine.

https://doi.org/10.33976/JERT.10.1/2023/3

Abstract— Fault Tolerant Control Systems (FTCS) have emerged as a critical area of study for enhancing the safety, reliability, and efficiency of modern control systems. The FTCS technique might be active or passive control in general. In this paper, the active control branch's Fault Detection and Diagnosis (FDD) is used to detect faults in DC motor speed sensors. FDD methodologies can be divided into two types based on the process and the type of data available: model-based methods and data-based methods. The proposed method here investigates the use of the Luenberger observer technique, which is part of the model-based approach. The selected method was implemented and experimentally evaluated. This observer is dependent on the residual signal, which serves as a fault indicator in the overall system and represents the difference between the measured and estimated speed signals from the plant. Due to the increasing demand for these motors, particularly in electro-mechanical applications such as robotics, elevators, and electric-driven railways, a DC motor was chosen as a benchmark to test the proposed method. The output speed of the motor was subjected to four sensor faults: sensor fault, abrupt fault, intermittent fault, and incipient fault. The effectiveness of the suggested approach is demonstrated using MATLAB simulations, and the results show that faults are detected as anticipated with a high-performing response. Therefore, the proposed method was also implemented experimentally in real time and the obtained results showed a close match with those from simulation, thus proving the accuracy and reliability of the proposed methodology for fault detection in the DC motor speed sensor.

Index Terms— Luenberger Observer Controller; Faults; DC motor.

I Introduction

High levels of system safety are provided by fault-tolerant control systems (FTCS), a class of extremely sophisticated control functions designed in a unified framework. There are two kinds of FTCS: passive and active [1]. This study looks into fault detection and diagnosis (FDD), a kind of fault-tolerant active control. Fault Tolerance Control (FTC) has received a lot of attention in sensitive industrial applications over the last two decades. There is a growing need to improve the dependability and safety of electrical systems in many industrial applications. Electric motors, aircraft, and robotic systems are examples of industrial applications. These applications have a great need for speed control tasks. One of the most well-known electric motors, the DC motor is commonly used in industrial appliances like electric vehicles, robot arms, cranes, and elevators [2]. DC motors are suitable for applications that require frequent and adjustable starting, good speed regulation, braking and reversing [3]. DC motors have several advantages over other types of motors such as: higher starting torque, speed variation, adjustable speed, applicability for low-speed torque and the required maintenance is easy in general [4]. FDD is a technique used by researchers interested in fault tolerance control to improve the efficiency and safety of electric motors [5]. A sudden failure of a motor while it is in the operation mode can result in costly downtime, environmental damages, equipment faults or even human danger [1]. Due to the slow rate of deterioration of some faults, they can be identified early on, enhancing safety, preventing system failures and product damage, and extending the useful life of the equipment [6]. Electrical motor faults are classified into two types: electrical and mechanical. A short circuit in the stator winding, rotor failure, an inverter fault, and a damaged end ring are all examples of electrical faults. Gearbox failure, bearing damage and shaft bending are examples of mechanical faults [7], [8]. Sensors play a crucial role in the operation of motors. When it comes to motor fault detection and diagnosis, sensor faults provide a significant issue. The two most common methods for monitoring and detecting faults in electric motors are model-based and data-based approaches. The model-based approach is based on comparing the behaviors of the real plant with those of the system's mathematical model. It employs the residual signal, which is the difference between the real (measured) and model processes (estimated signal). There are several techniques to generate residuals : parity equations, observer based generation and parameters estimation [9]. The most popular fault detection techniques are parameter estimation and observer-based techniques [10].

Data-based methods, on the other hand, are defined as " a dynamic model of an real process is not required ". To identify and diagnose faults, the entire fault detection method is based on physical quantities that are measured and examined [11]. There have been numerous contributions to the field of DC motor fault detection. Using a Luenberger observer to detect a speed sensor fault on a BLDC motor was simulated by a software technique using MATLAB in [12]. In [13] a robust method for fault-tolerant control (FTC) of a permanent magnet DC motor has been created and tested in the presence of various actuator fault types. A Luenberger observer based on the Bond Graph model was suggested in [14] as a method of fault detection. The authors in [15] presented an online sensor FDD via the model-based method using a Luenberger observer applied on DC motor. Robust design of unknown inputs (UI) propositional–integral (PI) observer to estimate the state of the system with UI and detect the existing fault has been presented in [16]. The authors in [17] presented an improvement for Luenberger observer using fuzzy logic for detecting sensor faults. In this paper, the main contribution is that the model-based sensor fault detection methods is applied to DC Motor (YA-070) in real time scenario where a Luenberger observer is proposed to detect a speed sensor fault on the motor. Fig. 1 illustrates the schematic diagram of the proposed method. In [12] we applied the proposed method to BLDC motor in MATLAB simulation environment.

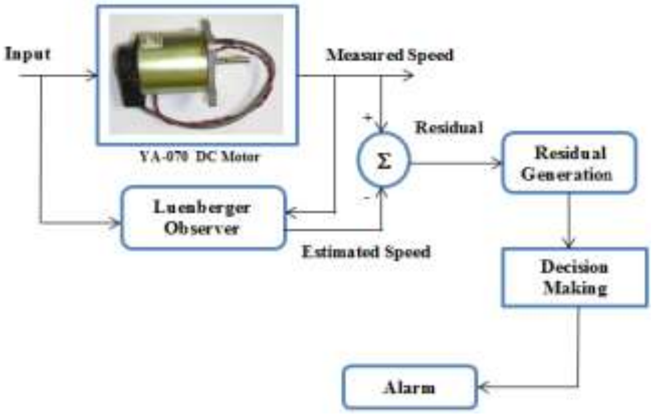

Figure 1. Speed Luenberger observer for YA-070 DC motor.

This paper is organized as following: In Section II, the mathematical model of the classical DC motor is presented. Section III provides descriptions of the Luenberger observer background. Section IV presents the design and implementation of the hardware and the algorithm used. Section V provides simulation and practical results of the proposed Luenberger observer method. Finally, some conclusion is discussed in Section VI

II DC Motor Mathematical Model

A DC Motor

The DC motor was introduced as a speed control target. The parameters of the DC motor model type (YA-070) are calculated practically [18]. Equivalent circuit of DC motor is shown in Fig. 2, that uses the armature voltage control method.

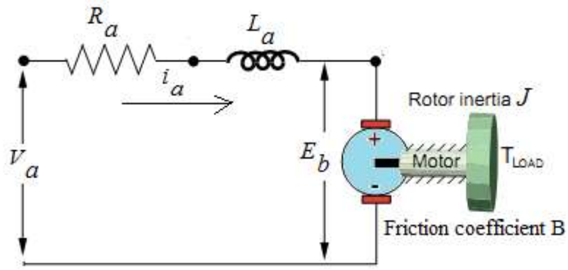

Figure 2. DC motor equivalent circuit

The specifications for the parameters are listed in Table 1 and were derived from the motor's datasheet [19].

TABLE 1
YA-070 DC MOTOR SPECIFICATIONS.

Motor Specifications	Value
Nominal drive voltage	24 V
Nominal rotational speed	3000 RPM
Nominal current	200 mA
Resistance R_a	7 Ω
Inductance L_a	0.008436 H
Rotor of inertia J	2.2097e-04 Kg.m^2
Friction coefficient B	1.65e-04 N .m/rad/sec
Back EMF constant K_b	0.094 V/rad/sec
Torque constant K_t	0.094 N .m/A

B Transfer Function of DC Motor

The final speed transfer functions for DC motor relative to the input voltage can be written as follows [18]:

$$\frac{\omega(S)}{V_a(S)} = \frac{K_t}{L_a J S^2 + (R_a J + L_a B) S + (R_a B + K_t K_b)} \quad (1)$$

The transfer functions of the DC motor in (1) are obtained in this study using the parameters in Table 1. As a result, the resulting transfer function is:

$$\mathrm{T.F} = \frac{0.094}{1.864e^{-6} S^2 + 0.001548\, S + 0.00991} \quad (2)$$

C State Space Model of the DC Motor

State space modeling was used in the design of the observer of fault detection for DC motors. The following is the general form of a state-space model [20]:

$$\dot{x}(t) = Ax(t) + Bu(\mathrm{t}) \quad (3)$$

$$y(t) = C\, x(t) \quad (4)$$

where $u(t)$ is the input, $x(t)$ is the state, A is the state matrix, B is the input matrix, C is the output matrix and $y(t)$ is the output.

State vector $x(t)$, has been chosen so that [21]:

$$x_1(t) = \omega(t) = \frac{d\theta(t)}{dt} \quad (5)$$

$$x_2(t) = i(t) \quad (6)$$

$$\dot{x}(t) = \binom{\dot{\omega}}{i} = \begin{pmatrix} -\frac{B}{J} & \frac{K_t}{J} \\ -\frac{K_b}{L_a} & -\frac{R_a}{L_a} \end{pmatrix} x + \begin{pmatrix} 0 \\ \frac{1}{L_a} \end{pmatrix} u \quad (7)$$

$$y(t) = \omega(t) = (1 \quad 0)x \quad (8)$$

where: $\omega(t)$ is the angular velocity and $i(t)$ is the current of the motor.

As a consequence, the A, B, and C matrices are as follows:

$$A = \begin{pmatrix} -\frac{B}{J} & \frac{K_t}{J} \\ -\frac{K_b}{L_a} & -\frac{R_a}{L_a} \end{pmatrix}, B = \begin{pmatrix} 0 \\ \frac{1}{L_a} \end{pmatrix}, C = (1 \quad 0) \quad (9)$$

The A, B, and C matrices are given by substituting the parameter values in (9):

$$A = \begin{pmatrix} -0.7467 & 425.4 \\ -11.14 & -829.8 \end{pmatrix}, B = \begin{pmatrix} 0 \\ 118.5 \end{pmatrix}, C = (1 \quad 0)$$

III Luenberger Observer Method

The primary principle behind the Luenberger observer method is to estimate the states of the real system using measured data [22]. The purpose of generating the residual signal is to compare the observed states to the measured states of the real system. After that, the residue is calculated by subtracting the outputs of the observed and measured signals [23]. Fig. 3 depicts the functioning diagram of the actual system as well as the Luenberger observer model that was utilized for fault detection methods. The Luenberger observer design for fault detection was really developed based on the state space theory. The following equations can be used to create the observer [22]:

$$\hat{\dot{X}} = A\hat{x}(t) + Bu(t) + Le(t) \quad (10)$$

$$\hat{y}(t) = C\hat{x}(t) \quad (11)$$

$$e(t) = y(t) - \hat{y}(t) \quad (12)$$

Where $\hat{x}(t)$is the estimated system state, $\hat{y}(t)$ is the estimated output, L is the observer gain and e(t) is the output error (difference between the real measured output y(t) and its estimate $\hat{y}(t)$). Replacing (12) in (10):

$$\hat{\dot{x}}(t) = [A - LC]\hat{x}(t) + Bu(t) + Ly(t) \quad (13)$$

The estimated error is given by:

$$e(t) = x(t) - \hat{x}(t) \quad (14)$$

By using the first derivative of e(t) , we may obtain the following:

$$\dot{e}(t) = \frac{d}{dt}\left(x(t) - \hat{x}(t)\right) = Ax(t) + Bu(t) - A\hat{x}(t) - Bu(t) - LC\left(x(t) - \hat{x}(t)\right) \quad (15)$$

The dynamics of the error is given by:

$$\dot{e}(t) = (A - LC)(x(t) - \hat{x}(t)) \quad (16)$$

Setting the eigenvalues of matrix (A-LC) to impose faster observer dynamics in comparison to real system dynamics defines the observer error dynamics [22].

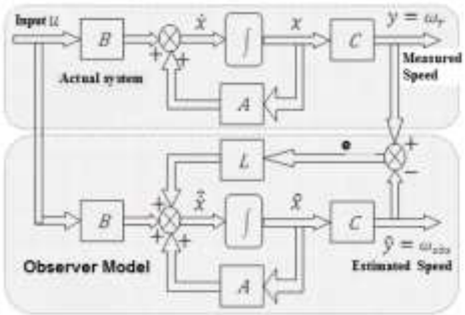

Figure 3. Observer model and real system's functional diagram.

The estimated and measured speeds are used to calculate the residual r(t). The residual r(t) is calculated as follows:

$$r(t) = y(t) - \hat{y}(t) \quad (17)$$

To detect faults, the residual value will be checked automatically. Hence, if the residual signal vanishes or approaches to zero, the system will be fault-free; otherwise, a specific fault would arise. Fig. 4 depicts the flowchart of the proposed technique for creating the Luenberger observer for fault detection [24].

IV Hardware and Software Design & Implementation

The experiments contain various hardware and software parts, which are described in this section. Drive circuits for DC motor are designed using Proteus software. Luenberger observer was also designed using Simulink-MATLAB. Arduino Mega 2560 board is used as the data acquisition card which can be programmed with the Simulink environment. Recently, the programmers can use the Simulink Support Package for Arduino Hardware to generate and run Simulink models on the Arduino Mega 2560 board [25].

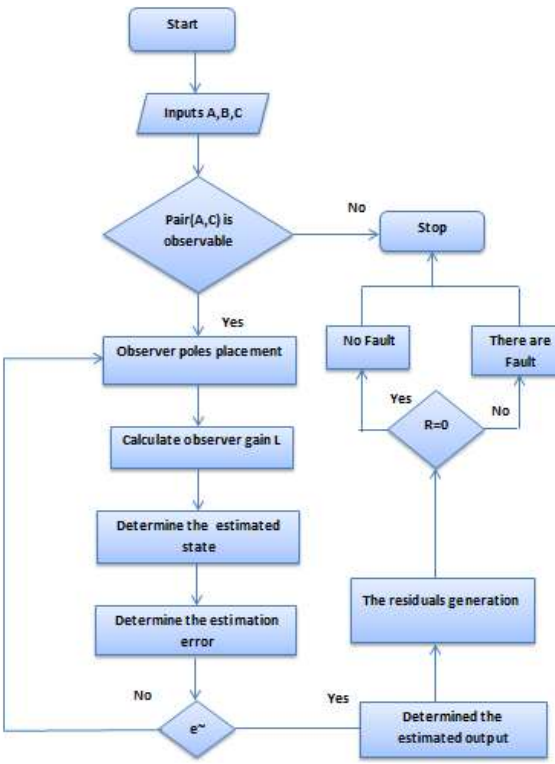

Figure 4. Luenberger observer flowchart for fault detection.

A *Design and Implementation for DC Motor Using Simulink*

Based on the step response of an armature-controlled DC motor, we constructed the model in Simulink. The speed command module generates a step signal with a range 0-255. This model explains how to use the Simulink blocks in order to generate pulse width modulation (PWM) output signals to control the DC motor. The PWM signal is sent through Pin 5 of the Arduino Mega 2560 board as shown in Fig. 5.

The DC motor speed is determined by optical encoder. The encoder type is the A-B phase incremental encoder. Each circle outputs 200 pulses. We used the pulse output of channel A to plot the speed curve and the output channel B to measure speed in RPM. The encoder pulses are counted on the Arduino Mega 2560 board via two of the board's digital inputs (pin 2, pin 3) [26]. Simulink is used to create the logic for predicting the motor's speed based on encoder counts. The Simulink supports a new package for Arduino hardware includes a tachometer. Sensors was used to measure motor speed in RPM through digital pin 2 of the Arduino Mega 2560, as shown in Fig. 5.

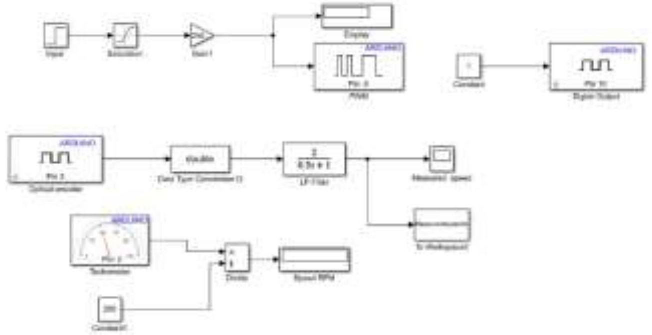

Figure 5. Overall Simulink design for DC motor.

B Experimental Setup for DC Motor

The control system consists of a DC motor with a sensor (optical encoder) , an Arduino mega 2560 controller, DC motor drive (L298N H-Bridge) and DC power supply. A schematic diagram of the equipment setup for this experiment is shown in Fig. 6 and a corresponding photo of the experimental setup is shown in Fig. 7.

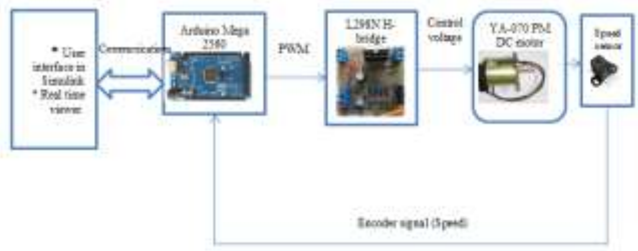

Figure 6. Schematic diagram of the equipment setup.

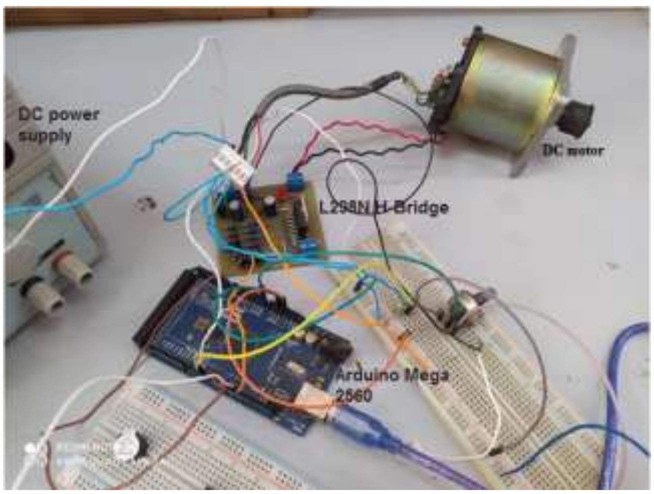

Figure 7. Photograph of equipment used.

C Details of Hardware Implementation

Motor controller (Arduino Mega 2650): The Arduino Mega 2560 is a well-known microcontroller board based on the

ATmega 2560, which is shown in Fig. 8. A 16 MHz crystal oscillator, 54 digital input/output pins (some of them can be programmed as PWM output signals), 16 analog inputs, 4 UARTs (hardware serial ports), a USB connection, a power jack, an ICSP header, and a reset button. Other specifications are given in [27].

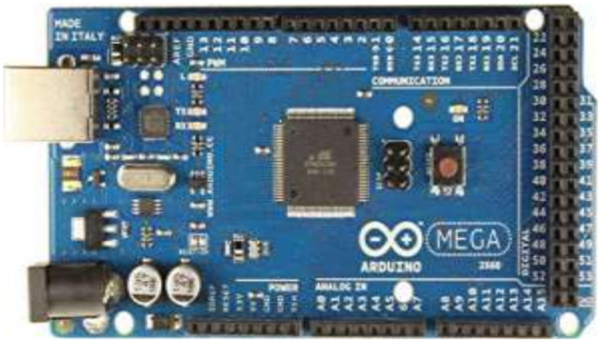

Figure 8. Arduino Mega 2650 board (motor controller).

Motor Drive: The L298N is a dual H-bridge motor driver board that is used to control the speed and direction of two DC motors with the required currents and voltages [28]. The motor drive circuit is designed by using Proteus software. The practical implementation of the PCB for the motor drive circuit shown in Fig. 9.

Figure 9. Motor driver (L298N H-bridge).

Optical encoder: The feedback used here is an optical incremental encoder, which is a linear or rotational electromechanical device with two output signals A and B that give out pulses when the device moves. More Details about optical encoder can be found in [29]. Fig. 10 shows the output signals of optical encoders where the sensor provides incremental position feedback, which can be extrapolated to accurate speed or position information.

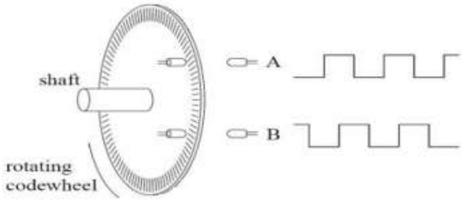

Figure 10. Optical encoder signals.

The optical incremental encoder used in the YA-070 DC motoris the H9700, as shown in Fig.11.

Figure 11. H9700 optical incremental encoder.

A Digital Storage Oscilloscope has been used to store the waveform of optical incremental encoder signal pulses. The sampled speed signal is shown in Fig.12 . In this paper, the speed sensor signal is sampled as an analog signal using a low-pass filter.

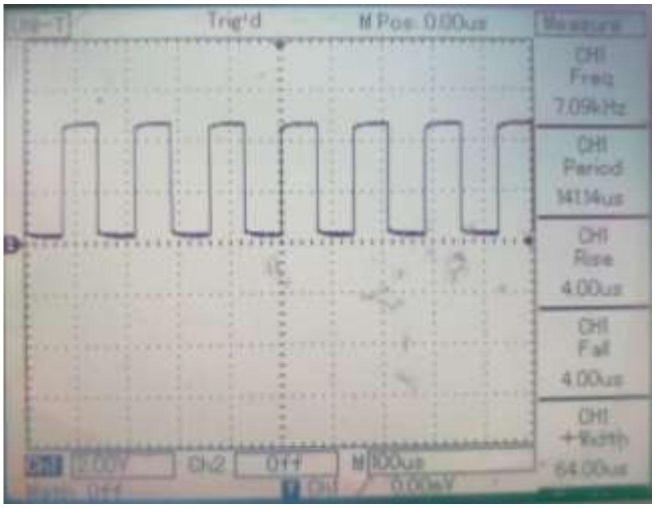

Figure 12. Optical sensor output (Scale:2 V/div).

V Simulation and Practical (Real Time) Results

Both simulation and real-time experiments were used to verify the effectiveness and accuracy of the suggested Luenberger observer for the DC motor. In the experimental system, all experiments were performed in real time, and they were carried out at a sampling time of Ts=1ms. For the DC motor (YA-070), all experiments are carried out with an input voltage of 1 V (which corresponds to 2100 RPM). DC motor models are created in the MATLAB/Simulink environment. Several types of well-known faults were tested on the motor speed sensor, which is defined in [24]:

- ***Additive fault:*** occurs as a result of a change in internal temperature or calibration issues. When there is an additional constant value in the speed sensor output, this fault occurs. There are two sorts of additive faults: abrupt (occurs instantly) and intermittent (appears and disappears repeatedly). Abrupt faults are frequently caused by hardware damage, whereas intermittent faults are caused by partial wire damage [5]. While the abrupt fault model can be created as a step function in Simulink, the intermittent fault model can be created as a series of pulses with varying amplitudes.
- ***Multiplicative fault*:** this type of fault can happen due to multiplier coefficients which are applied to the sensor. This kind of fault, known as an incipient fault, is characterized by gradual changes in its parameters. It is frequently the outcome of the system's aging [5]. The incipient defect model in Simulink can be produced as a ramp function.
- ***Sensor fault:*** this catastrophic fault happens when a sensor malfunctions at a particular moment (possibly because of a disconnect) and generates a constant zero after the fault arises [5]. Simulink allows for the simple multiplication of the speed signal by zero to simulate the sensor fault. This fault was practically implemented by separating the speed sensor from the motors for a short time period, then reconnecting them again, or by cutting off the power to the motors.

Fault detection thresholds are defined according to the maximum values achieved by residuals over several experiments. There are two types of thresholds, one of which is the fixed threshold, whereas the other is the adaptive threshold [30]. In this paper, a fixed type threshold was used, and many tests were performed on the motors in a fault-free state to determine the values of both upper and lower thresholds. As soon as the residual exceeds a specific threshold, the alarm is triggered to denote a fault. A buzzer device was used as an alarm. Moreover, the alarm is activated when this fault occurs. The Luenberger observer for DC motor simulation design is shown in Fig. 13, which includes the four types of faults mentioned above. A Simulink design of the fault detection method based on the Luenberger observer (real-time) for the DC motor as represented in Fig.14.

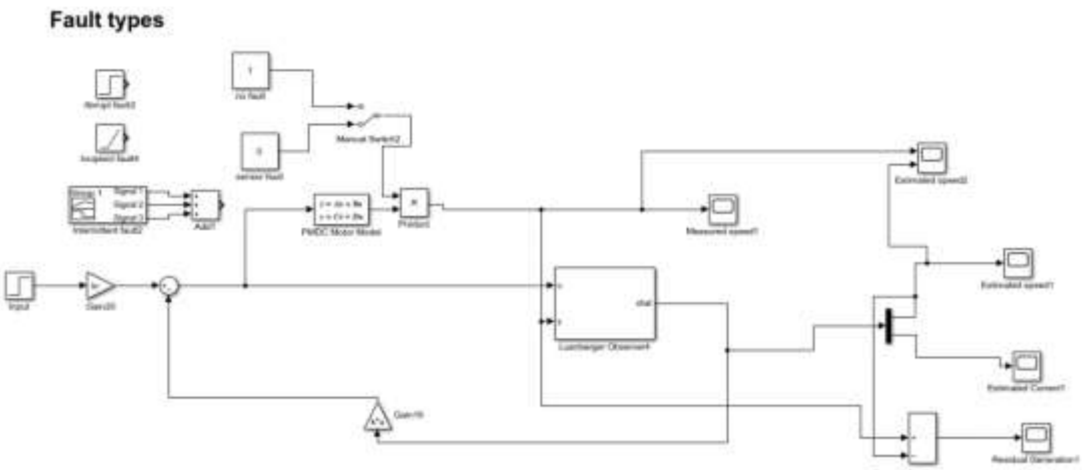

Figure 13. Luenberger observer simulation design for DC motor.

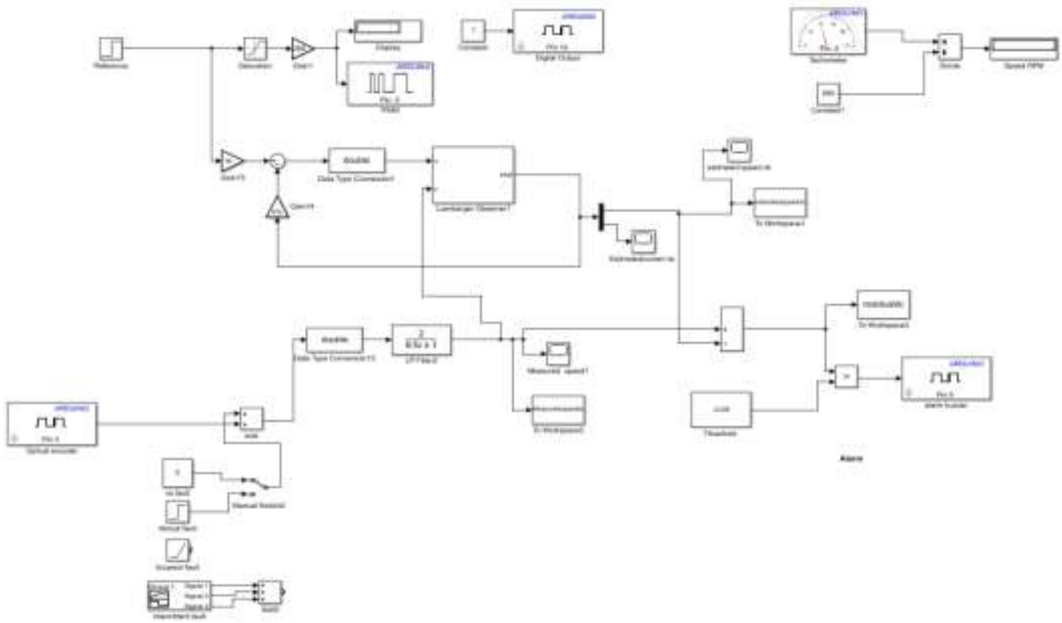

Figure 14. Luenberger observer simulation design for DC motor (real-time).

In this research, very close results have been demonstrated between simulation and practical implementation for every type of fault.

A. Simulation Results without any Faults in DC Motor

Here, no faults with the DC motor's speed are present in this simulation. The output speed (measured), estimated speed, and residual output are shown in Fig. 15, Fig. 16, and Fig. 17, respectively.

Figure 15. Simulation of faultless output speed.

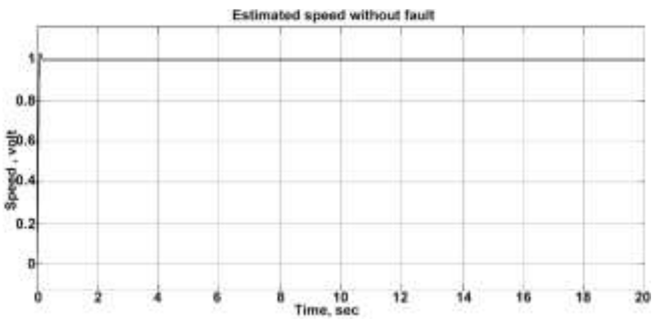

Figure 16. Estimated speed simulation without fault.

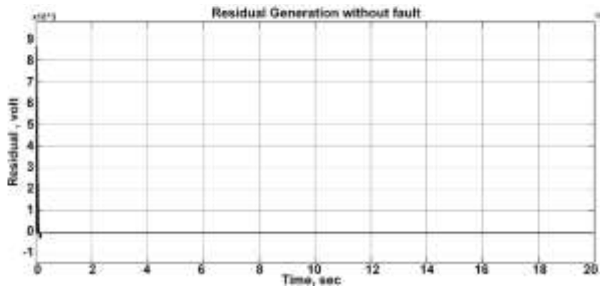

Figure 17. Simulating the residual output of a DC motor without a fault

As illustrated in Fig. 17, residual generation signal, which is nearly negligible, that the threshold value is error-free.

B. Experimental Results without any Faults in DC Motor

This experiment was performed with no fault in the speed sensor for DC motor. It was implemented in real-time. Fig. 18 and Fig. 19 show the measured and estimated output speeds.

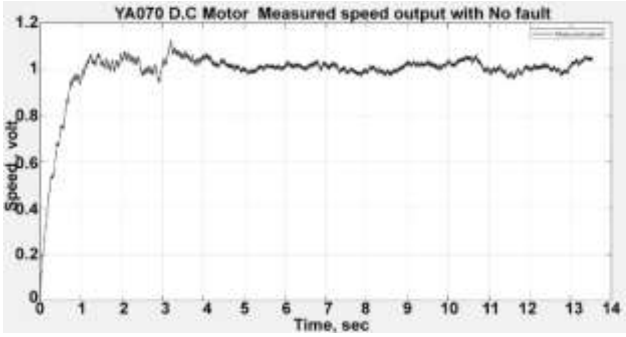

Figure 18. The measured speed of DC motor.

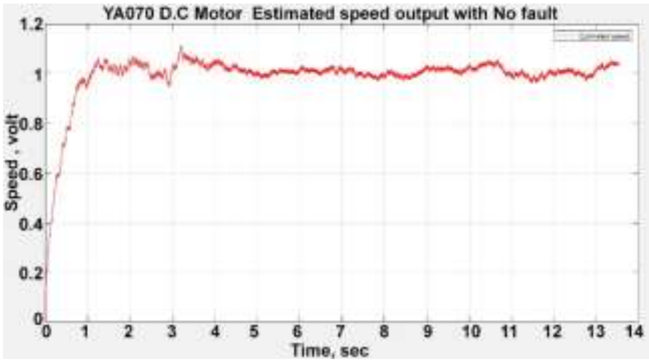

Figure 19. The estimated speed of DC motor.

Experimentally, tests were applied to the DC motor while it was fault-free in order to determine the threshold value from the residual generation. The upper threshold value was 0.0157 while the lower threshold value was -0.009 as shown in Fig. 20. It is important to find these threshold values in order to develop an alarm when faults occur in the motor speed.

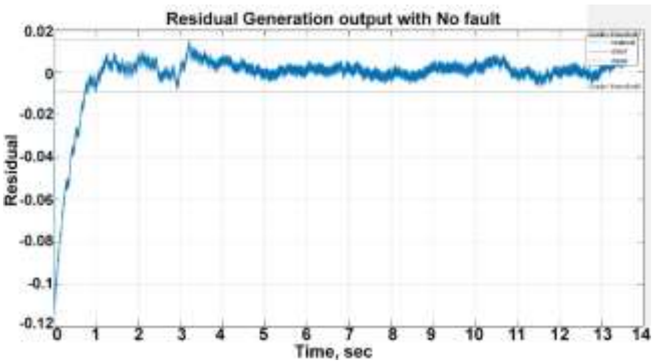

Figure 20. Residual output without any fault, upper threshold 0.0157, lower threshold value was - 0.009.

Obviously, the residual generation illustrated in Fig. 20 is not equal to zero due to the noise that is presented in the DC motor and errors in the parameters of observer design. Hence, the higher threshold value of 0.0157 and a lower value of -0.009 were chosen in order to avoid false alarms for the residual value.

C. Simulation Results of Aprupt Fault in DC Motor

By adding a constant value to the sensor reading at the fourteenth second, as seen in Fig. 13, an abrupt fault is applied to the DC motor speed sensor. Fig. 21 depicts the output simulation and estimated DC motor speeds, whereas Fig. 22 depicts the residual output. At the time of its appearance, the fault detection method was clearly carried out successfully.

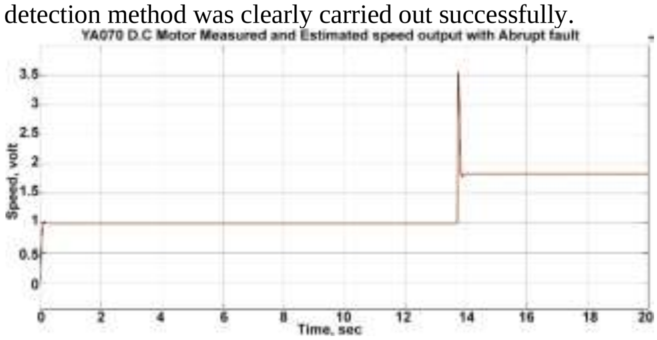

Figure 21. Output and estimated speed simulation with an abrupt fault

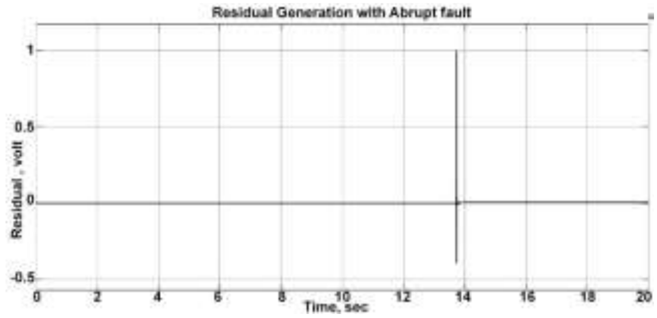

Figure 22. Simulating residual output with an abrupt

From Fig. 22, it is obvious that the residual value at the fourteenth second exceeds the threshold value, which matches the simulation result in Fig. 21.

D. Experimental Results of Aprupt Fault in DC Motor

This experiment was performed with abrupt fault in the speed sensor for DC motor. It was implemented to the output speed at the third second. The experimental of the output and estimated DC motor speeds are presented in Fig. 23 and the residual output is presented in Fig. 24. Here, the fault detection process was executed perfectly, where the fault detection was carried out successfully at the time of its appearance.

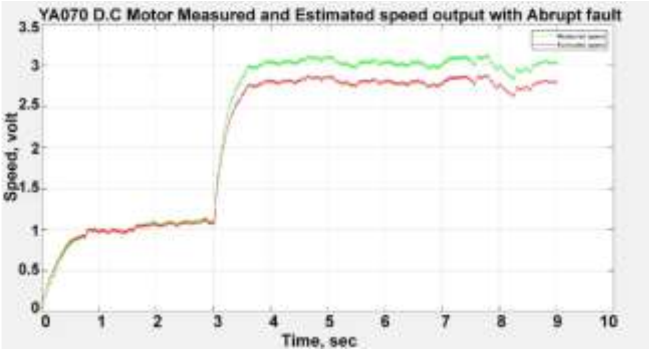

Figure 23. Measured and estimated speeds of DC motor with Abrupt fault.

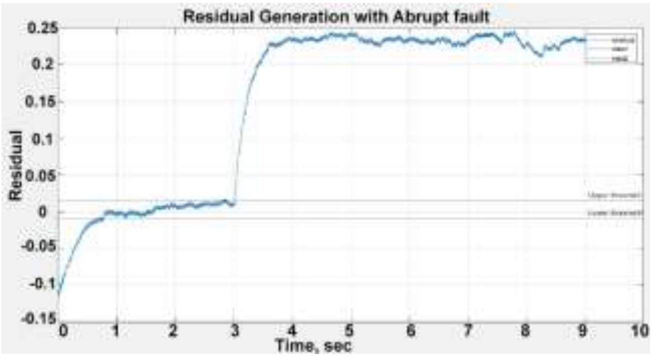

Figure 24. Residual output with an abrupt fault.

From Fig. 24, it can be noted that the residual value at third second is greater than the threshold level. As a consequence, a fault in the speed sensor occurred. At this moment, the alarm is triggered.

E. Simulation Results of Incipient Fault in DC Motor

The incipient fault that applied to the DC motor speed sensor is shown in Fig. 13. At the eleventh second, it was applied to the output speed. Fig. 25 depicts the output simulation and estimated DC motor speeds, whereas Fig. 26 depicts the residual output. These figures show that, at the time the fault first appeared, the fault detection method worked.

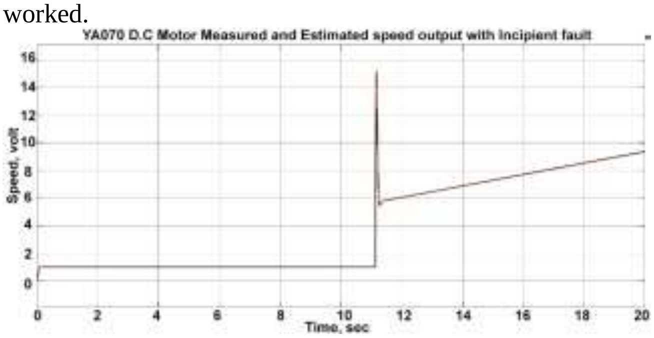

Figure 25. Simulation of output and estimated speeds with incipient fault.

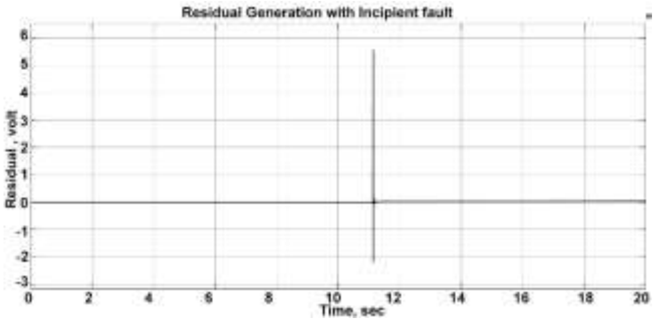

Figure 26. Simulation of residual output with incipient fault.

As shown in Fig. 26, at the start of the eleventh second, the residual value exceeds the threshold value, indicating a fault with the sensor's speed.

F. Experimental Results of Incipient Fault in DC Motor

This experiment was performed with an incipient fault in the speed sensor for the DC motor. At the 2nd second, it was implemented to the output speed. Fig. 27 shows the output experimentally and the estimated DC motor speeds, while Fig. 28 shows the residual output. It is noted that where fault detection was successful at the time it occurred, the fault detection process was carried out successfully..

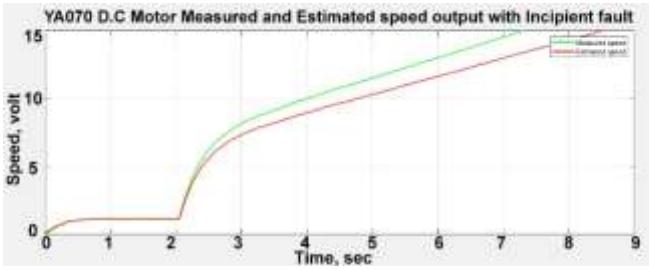

Figure 27. Measured and estimated speeds of DC motor with incipient fault.

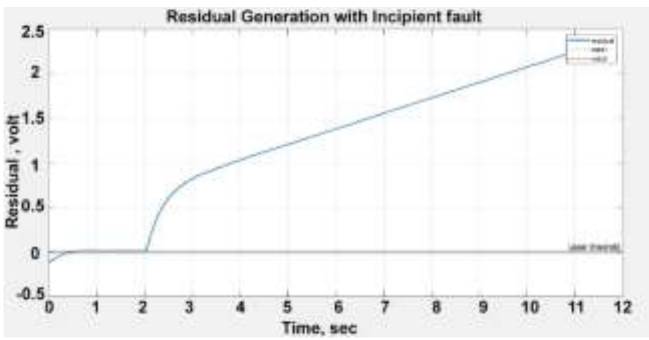

Figure 28. Residual output with incipient fault.

The residual value, in Fig. 28, is gradually increased above the threshold value in the 2nd second at the time where the fault occurred, which refers to a fault in the speed sensor.

G. Simulation Results of Intermittent Fault in DC motor

In this section, the intermittent fault shown in Fig. 13 is applied to the DC motor speed sensor. It is implemented by periodically adding a constant value to the sensor reading. An intermittent fault was applied to the output speed at the fifth, eighth, and eleventh seconds, respectively, with an amplitude of 1, 1.5, and 2. The pulse width of each of these signals is one second. Fig. 29 depicts the output simulation and estimated DC motor speeds, while Fig. 30 depicts the residual output where the residual value surpasses the threshold level.

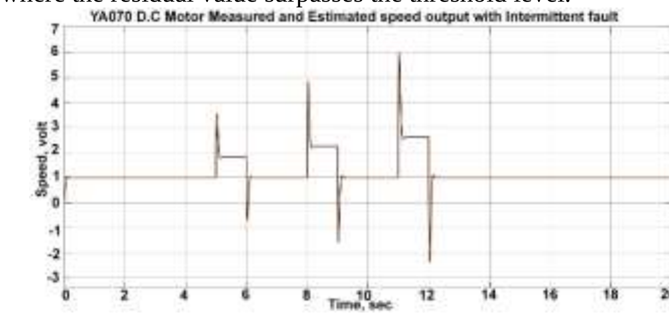

Figure 29. Output and an estimated speed simulation with an intermittent fault.

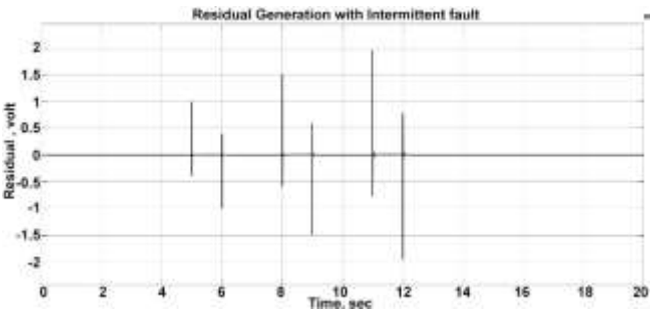

Figure 30. Simulation of residual output with intermittent fault.

H. Experimental Results of Intermittent Fault in DC motor

This experiment was performed with an intermittent fault in the speed sensor for the DC motor. It was implemented in a real-time scenario. Fig. 31 presents the measured output signal and the estimated DC motor speeds, while Fig. 32 shows the residual output. It was noted that where the residual value exceeded the threshold value, the defect detection process was successfully completed.

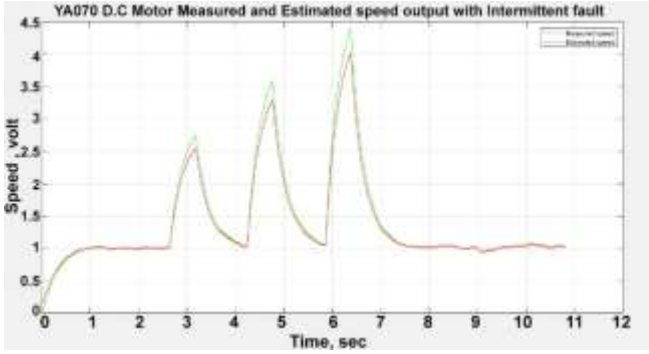

Figure 31. Measured and estimated speeds of DC motor with intermittent fault.

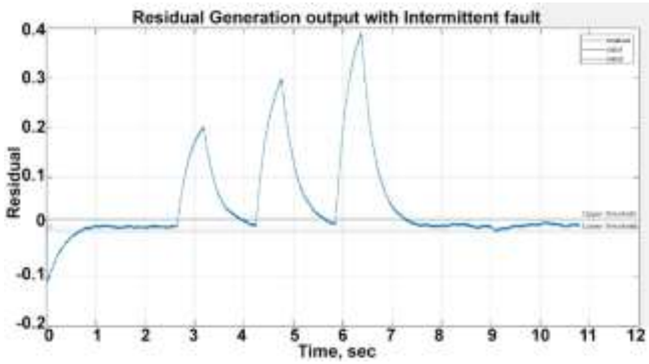

Figure 32. Residual output with intermittent fault.

I. Simulation Results of Sensor Fault in DC motor

To use these faults in the simulation, the speed signal must be multiplied by zero. Fig.33 displays the output simulation and the estimated DC motor speeds, while Fig. 34 shows the residual outpu.

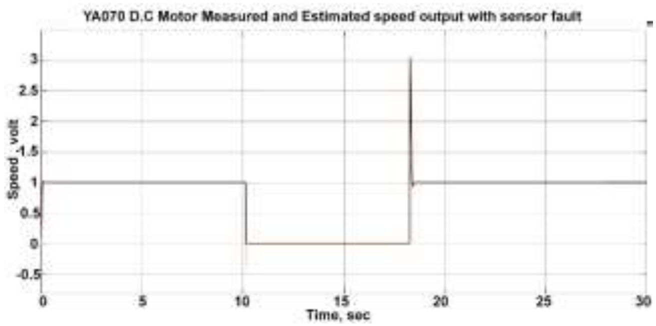

Figure 33. Simulation of DC motor output and estimated speeds with sensor failure.

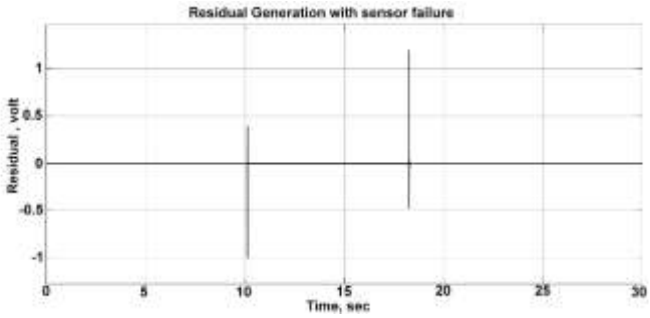

Figure 34. Residual output simulation in the presence of a sensor fault.

When the fault occurs, the residuals value drops below the threshold value, as shown in Fig.34.

J. Experimental Results of Sensor Fault in DC motor

When the speed sensor is unplugged from the DC source, a sensor fault occurs. At the third second, it was applied to the output speed. Then the sensor reconnects after one second. This fault was implemented in real-time. Fig. 35 shows the response of the measured and estimated speeds ,and Fig. 36 shows the residual generation output. Here, the fault detection process was successfully performed.

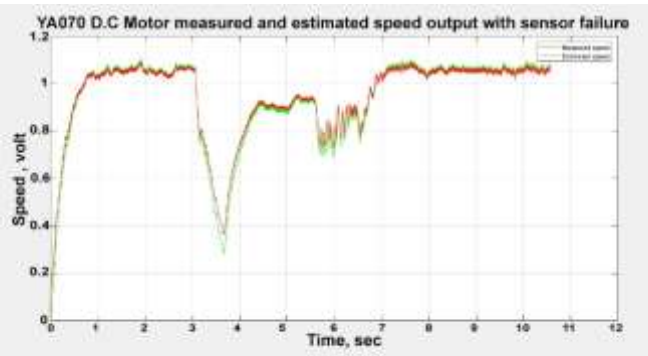

Figure 35. Measured and estimated speed due to sensor failure.

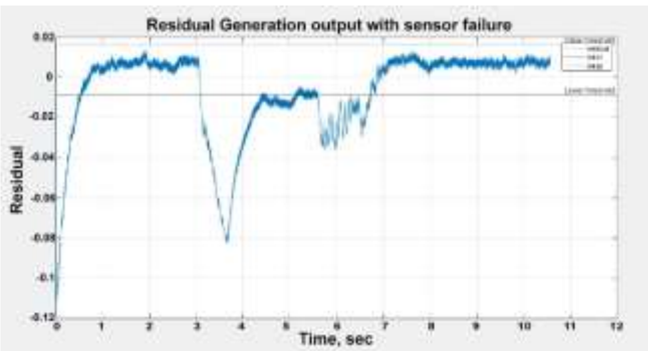

Figure 36. Residual output due to sensor failure.

From Fig.36, it can be noted that the residual value is lower than the threshold value at the time of applying the fault, which will indicate a fault in the sensor. At this time, the alarm is triggered to indicate the presence of the fault.

VI Conclusion And Futuer Work

In this research, model-based methods employing Luenberger observers were investigated in order to detect faults in a DC motor's speed sensor.

Four well-known fault types were applied to the speed sensor of the YA-070 DC motor in order to test the validity of the proposed method, namely: sensor fault, abrupt fault, intermittent fault, and incipient fault. The threshold algorithm is used in the suggested strategy to generate the residual signal that indicates a sensor failure. The proposed method's effectiveness is demonstrated using both simulation and real-time experiments. A possible future work for this research topic is to integrate the kalman filter or particle filter algorithms into the observer design process.

References

[1] K. S. Gaeid, “Fault Tolerant Control of Induction Motor,” Modern Applied Science, vol. 5, no. 4, 2011.

[2] Q. A. F. M. Sajidul, “Electro-Mechanical Modeling of Separately Excited DC Motor & Performance Improvement using different Industrial Controllers with Active Circuit Realization,” *International Conference on Mechanical, Industrial and Energy Engineering*, Dec. 2014

[3] A. A. Hassan, N. K. Al-Shamaa, and K. K. Abdalla, “Comparative Study for DC Motor Speed Control Using PID Controller,” *International Journal of Engineering and Technology*, vol. 12, no. 24, pp. 15999-16007, 2017.

[4] [Online]. Available: https://www.linquip.com/blog/dc-motor-working principles/#Advantages_and_Disadvantages_of_DC_Motors. [Accessed: 15-Jul-2021].

[5] A.Alkaya, “Novel data driven-based fault detection for electromechanical and process control systems,” PhD thesis, 2012.

[6] D. Miljković, "Fault detection methods: A literature survey," *2011 Proceedings of the 34th International Convention MIPRO*, Opatija, 2011, pp. 750-755.

[7] W. Abed, *Robust Fault Analysis for Permanent Magnet DC Motor in Safety Critical Applications*. 2015.

[8] L. Li, “Robust fault detection and diagnosis for permanent magnet synchronous motors,” 2006.

[9] K. S. Gaied and H. W. Ping, “Wavelet fault diagnosis and tolerant of induction motor: A review,” *International Journal of Physical Sciences*, vol. 6, pp. 358–376, 2011.

[10] R. Isermann and P. Ballé, “Trends in the application of model-based fault detection and diagnosis of technical processes,” *Control Engineering Practice*, vol. 5, no. 5, pp. 709–719, 1997.

[11] R. Isermann, Fault-diagnosis systems: an introduction from fault detection to fault tolerance. Berlin: Springer, 2006

[12] Y. Al-Mutayeb and M. Almobaied, “Luenberger observer-based speed sensor fault detection of BLDC Motors,” *2021 International Conference on Electric Power Engineering – Palestine (ICEPE- P)*, 2021.

[13] Q. Fazal, M. Liaquat, and N. Naz, “Robust fault tolerant control of a DC motor in the presence of actuator faults,”

2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA), 2015.

[14] G. Saoudi, R. E. Harabi, and M. N. Abdelkrim, "Graphical linear observers for fault detection: The DC motor case study," *10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13)*, 2013.

[15] A. Alkaya ve İ. Eker , "Luenberger observer-based sensor fault detection: online application to DC motor", *Turkish Journal of Electrical Engineering and Computer Science,* c. 22, sayı. 2, ss. 363-370, May. 2014, doi:10.3906/elk-1203-84.

[16] M. A. Eissa, M. S. Ahmed, R. R. Darwish, and A. M. Bassiuny, "Unknown inputs pi observer-based sensor fault detection technique for BLDC Motor," *2015 7th International Conference on Modelling, Identification and Control (ICMIC),* 2015.

[17] M. A. Eissa, M. S. Ahmed, R. R. Darwish, and A. M. Bassiuny, "Improved fuzzy luenberger observer-based fault detection for BLDC Motor," *2015 Tenth International Conference on Computer Engineering & Systems (ICCES)*, 2015.

[18] A. A. Hassan, N. K. Al-Shamaa, and K. K. Abdalla, "Comparative Study of Conventional and Optimal PID Tunned Methods for PMDCM Speed Control," *International Journal of Applied Engineering Research,* vol. 9, no. 6, pp. 4181–4192, 2017.

[19] "Imported YDK servo DC motor YA-070 YA-071 DC24V 3000 rotary motor," China online shopping - Buy asian products online from the best shoping agent yoycart.com. [Online]. Available: https://www.yoycart.com/Product/26695000175/. [Accessed: 15-Jul-2021].

[20] M. Aghaee and A. A. Jalali, "BLDC Motor Speed Control Based on MP C Sliding Mode Multi-Loop Control Strategy – Implementation on Matlab and Arduino Software," *Electrical Engineering (ICEE), Iranian Conference on,* 2018.

[21] M. Ridwan, M. N. Yuniarto, and Soedibyo, "Electrical equivalent circuit based modeling and analysis of brushless direct current (BLDC) motor," *2016 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2016.

[22] Z. Horváth and G. Molnárka, "Design Luenberger Observer for an Electromechanical Actuator," Acta Technica Jaurinensis, vol. 7, no. 4, 2014.

[23] L. Sellami, "Simulink Model of a Full State Observer for a DC Motor Position, Speed, and Current," *in World Congress in Computer Science, Computer Engineering, and Applied Computing,* 2014.

[24] M. A. Eissa, M. S. Ahmed, R. R. Darwish, and A. M. Bassiuny, "Model-based sensor fault detection to brushless DC motor using Luenberger observer," *2015 7th International Conference on Modelling, Identification and Control (ICMIC)*, 2015.

[25] A. S. Sadun, J. Jalani, and J. A. Sukor, "A Comparative Study on the Position Control Method of DC Servo Motor with Position Feedback by using Arduino ," Proceedings of Engineering Technology International Conference (ETIC 2016), Bali ,2016.

[26] W. Tang, Z. Liu and Q. Wang, "DC motor speed control based on system identification and PID auto tuning," 2017 36th Chinese Control Conference (CCC), 2017, pp. 6420-6423, doi: 10.23919/ChiCC.2017.8028376.

[27] Facebook.com/zainnasir, "Introduction to Arduino Mega 2560," *The Engineering Projects*, 21-Jun-2021. [Online]. Available: http://www.theengineeringprojects.com/2018/06/introduction-to-arduino-mega-2560.html. [Accessed: 02-Aug-2021].

[28] Dejan, "L298N motor driver - Arduino interface, how it Works, Codes, Schematics," *HowToMechatronics*, 11-May-2021. [Online]. Available: https://howtomechatronics.com/tutorials/arduino/arduino-dc-motor-control-tutorial-l298n-pwm-h-bridge/. [Accessed: 02-Aug-2021].

[29] http://www.creative-robotics.com/.

[30] P. M. Frank, S. X. Ding, and T. Marcu, "Model-based fault diagnosis in technical processes," *Transactions of the Institute of Measurement and Control*, vol. 22, no. 1, pp. 57–101, 2000.

Moayed Almobaied is an assistant professor of electrical Engineering at Islamic University of Gaza –Palestine. He received his B.Sc and M.Sc degrees in control Engineering from Islamic university of Gaza in 2001 and 2008, respectively. In 2017, he received the Ph.D. in control and automation systems from Istanbul Technical university-ITU. His current research interests include Robust control, Optimal control, Designing of modern control systems, and Robotics.

Yousef Al-Mutayeb is instructor assistant at the Engineering Science Department in University College of Science and Technology-khan Yunis. He is also working as a lab supervisor in the Department of Electronics Engineering to date. His research interests are in the area of control systems. He obtained his MSc degree in Electrical Engineering from the Islamic University of Gaza in 2020.