**Deployment of Low-Cost Vibration Monitoring Assessment On Rotating Machinery**

**Ahmad Fuad Ab Ghani1,[[1]](#footnote-1)\*, Abdul Rafeq Saleman1, Shukri Yob1, Norazazi Ngatiman1, Irman Ramli1, Raja Nor Firdaus Raja Othman2, Fikri Alami3**

1 Faculty of Mechanical Technology and Engineering, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia [ahmadfuad@utem.edu.my](mailto:ahmadfuad@utem.edu.my)

2Faculty of Electrical Technology and Engineering, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

3Faculty of Civil Engineering, Universiti Lampung, 35145, Bandar Lampung, Indonesia

\*E-mail: ahmadfuad@utem.edu.my

**Abstract.** This project is related to mechanical vibration. This project creates a low-cost vibration monitoring system to determine the vibration of the target machine and display the vibration trend in the interface software Blynk software. The important hardware of this low-cost vibration monitoring system are Hibiscus Sense, accelerometer GY-521 MPU6050 and 5000mAh power bank. Hibiscus Sense is an ESP32 development board that includes a microcontroller, accelerometer MPU6050 and temperature sensor BME280. It is an IoT device that can transmit data over wireless connections such as 2.4GHz Wi-Fi and BLE (Bluetooth Low Energy). The microcontroller from Hibiscus Sense is used to manage the built-in accelerometer MPU6050 and accelerometer GY-521 MPU6050, and transmit data to the Blynk software via a Wi-Fi connection. Two MPU6050s are used to detect the vibration generated by the motor. Besides, the interface software of this low-cost vibration monitoring system are Arduino IDE software and Blynk software. The Arduino IDE is the programming software used to generate the code to run the system, while Blynk is used to display the received data in gauge form and trend form for vibration analysis.

**Keywords:** Hibiscus Sense, Accelerometer MPU6050, Wireless Integrated Low Cost Vibration Monitoring System, Arduino IDE software, Blynk Software.

|  |  |
| --- | --- |
|  |  |
|  | |

**1. Introduction**

Vibration refers to the oscillation, rotation, or any repetitive movement of a rigid or flexible object or substance that is displaced from a state of balance. [1]. Machine vibration is a prevalent and frequently inevitable outcome of the movement and rotation of various parts. It is commonly caused by the tolerances in manufacturing and assembly, leading to gaps between mating parts or imbalances in spinning components. Although regular wear and tear can gradually increase vibration, a significant or sudden rise or alteration in vibration may signal a problem. This could indicate that the machine or its components are experiencing heightened stress, loss of stiffness, and premature wear[2].

Vibration analysis involves identifying irregularities and tracking alterations in the vibration pattern of a system. Changes in the magnitude, strength, and frequency characterize the vibration of any movable object. This method can reveal an emerging issue that can be rectified to prolong the machine's lifespan, monitor an ongoing problem that is irreversible and worsening, and establish criteria for acceptance testing to ensure proper installation/repairs [3].

Vibration issues are not as commonly addressed in India as they are in Western countries. However, there is a growing recognition of their significance in the industry. Due to the higher cost of importing vibration instruments, local enterprises require a cost-effective and energy-efficient vibration monitoring system that can be used in hazardous locations like Class-B and Class-C. This proposed design could be an excellent option for small or medium-sized electrical and mechanical businesses, offering both usefulness and cost-effectiveness. The suggested design is a flexible, customizable instrument that can be wired or wireless, allowing for field or remote use. It senses vibration as acceleration and velocity and displays the data on an LCD (Liquid Crystal Display) [4]. Consequently, the emergence of IoT for Structural Health Monitoring (SHM) in aircraft structures holds great promise. However, there are currently obstacles in developing rapid, precise, and cost-effective solutions for SHM systems in aircraft. In the prior investigation, the affordable MEMS accelerometer vibration sensors ADXL335 and MPU6050 were widely utilized in various fields for measurement and analysis purposes. An example of this is the application of the ADXL335 sensor in a vibration monitoring system on the PT-1000/60 conveyor belt. As indicated in the aforementioned study [5], the ADXL335 was employed to capture vibration data by positioning it on the conveyor belt and connecting it with other Arduino components like Arduino UNO and Shield Data Logger to generate time-domain graphs, FFT graphs, and to archive historical data. Additionally, the ADXL335 sensor was utilized to conduct wireless vibration analysis by detecting the unbalanced faults produced by the experimental test rig system [6].

**1.1 Problem Statement**

Until now, there has been limited research into low-cost devices that could help to track machine efficiency or to enhance the reliability of machine components. The majority of research on vibration focusses on industrial level and high end vibration monitoring system. It is worth to look at the possibility and practicality of developing in-house vibration monitoring system so that, industrial practitioners could deploy and utilize the method for beneficial of their organization engineering system. The aim of this project includes the creation and validation of an affordable vibration monitoring system, as well as the implementation of software development for said system. Data collected from the integrated MPU6050 and GY-521 MPU6050 will be displayed in the Blynk software for the purpose of data comparison and vibration analysis.

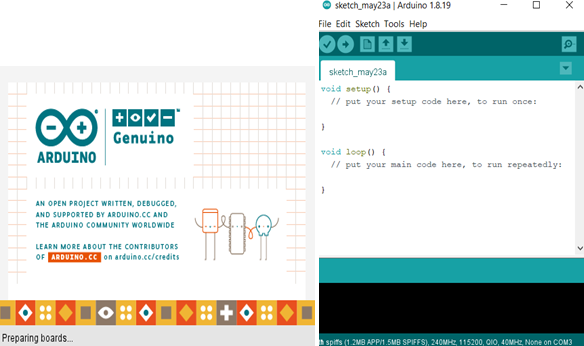
**2. Methodology**

*2.1 Vibration Monitoring System*

There are 2 significant parts of the vibration monitoring system to consider, software and hardware development. This project mainly explains the software development and hardware part which focuses on the selection of main and the actual appearance of components.

*2.2 Software Implementation*

This section provide information about the software interfaces utilized to create real-time time domain graphs for vibration analysis using received data. The project selected version 1.8.19 of the Arduino IDE software due to its free availability, ease of access, and compatibility with the ESP32 microcontroller development board, as depicted in Figure 1. The Arduino IDE software can be used to program the vibration monitoring system. Additionally, the Arduino programming code is written in Java and based on C++ programming.



**Figure 1** Arduino IDE software

The Blynk software was selected as the second interfacing software due to its user-friendly interface and cost-free nature for data visualization needs. With Blynk software, users can conveniently monitor vibration data on both the Blynk IoT website and mobile application as depicted in **Figure 2**. The data transmitted from the Hibiscus Sense's microcontroller through Wi-Fi can be presented in gauge chart and graphical formats. Figure 3 depicted the Hibiscus Sense feature used in the research project.

|  |  |
| --- | --- |
|  |  |

**Figure 2** Gauge Chart and Line Graph on Blynk software

2.3 *Required Hardware*



**Figure 3** Overview of Hibiscus Sense

**Table 1** Hibiscus Sense required specification

|  |  |
| --- | --- |
| Working Voltage | 3.3V |
| Working Frequency | 240MHz |
| MPU6050’s Sampling Range | 1kHz to 8kHz |
| MPU6050’s Temperature Sampling Range | -40 to 87 |
| Dimension | 58.7  (Length Height) |

The Hibiscus Sense is a Internet of Things (IoT) development board that has allows wireless connectivity with computer and smartphone through both Wi-Fi and Bluetooth. The microcontroller of Hibiscus Sense is used to receive the commands given in the Arduino IDE software to manage the MPU6050 accelerometer (built-in sensor of Hibiscus Sense) as per **Figure 4** and the GY-521 MPU6050 accelerometer and transmit the data of both sensors to the Blynk software. Meanwhile, Table 2 states the GY-521 MPU6050 Accelerometer specification for the use in this project.



**Figure 4** GY-521 MPU6050 Accelerometer

**Table 2** GY-521 MPU6050 Accelerometer specification

|  |  |
| --- | --- |
| **Operating Voltage & Current** | 3.3V to 5V, 500A |
| **Operating Frequency** | 20MHz |
| **Sampling Range** | 1kHz to 8kHz |
| **Temperature Sampling Range** | -40 to 85 |
| **Dimension (Excluding Pins)** | 21.2 16.4 3.3 mm |

The GY-521 MPU6050 is also used to detect the machinery vibration during machine operation for vibration analysis and data comparison. A 5000 mAh power bank is required to act as power supply for the Hibiscus Sense. This powerbank is used as the power supply to provide the current and voltage to the Hibiscus Sense and GY-521 MPU6050, just connect to the USB port on the hibiscus sense via a Type-C cable.

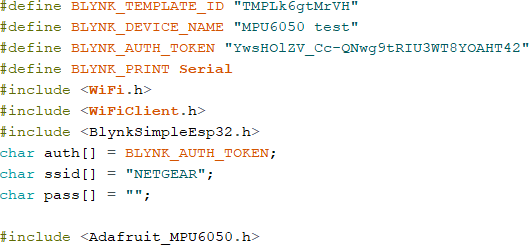
**Hardware and Software Set Up**

As a summary of this section, the Arduino IDE software setup is shown in **Figure 5**, **Figure 6**, **Figure 7** and **Figure 8** whereas the setup of Blynk software is shown in **Figure 9**. The overview of hardware setup is shown in **Figure 10.**

A white rectangular frame with black and blue stripes

Description automatically generated

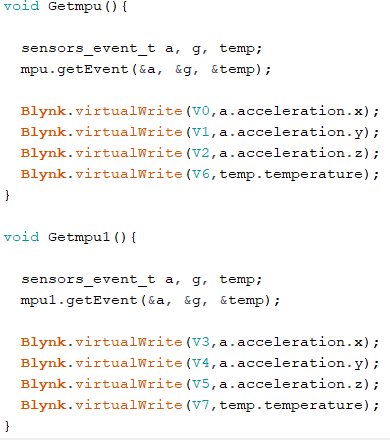
**Figure 5** Programming Code Setup (Arduino IDE software)



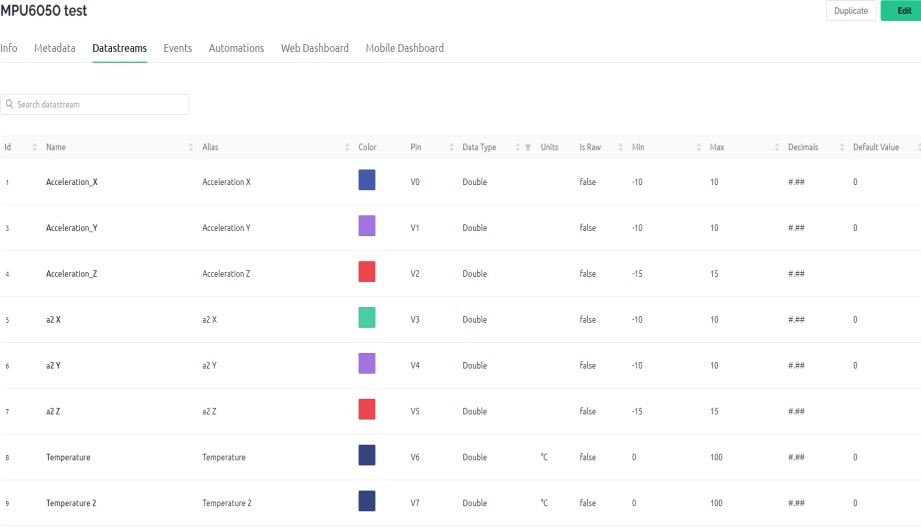
**Figure 6** Software Integration Code and Library Required Code



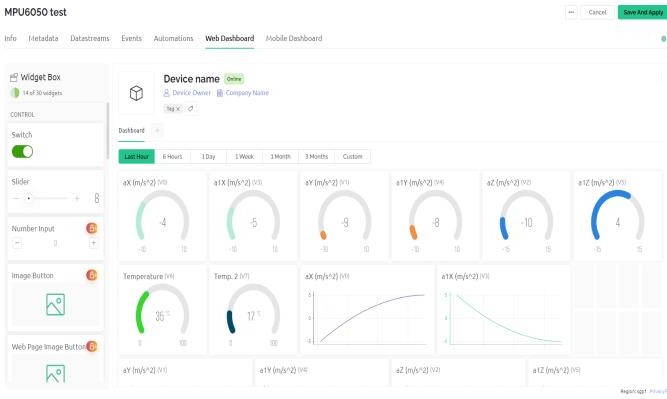
**Figure 7** Main Operation Function Code



**Figure 8** Sub-operation Function Code



**Figure 9** Setting of DataStream of Blynk Software



**Figure 10** Setting of Dashboard of Blynk Software

The coding in the Arduino IDE is uploaded to the Hibiscus Sense via the Type-C cable. The code is used to command the Hibiscus Sense’s microcontroller to manage the built-in MPU6050 and GY-521 MPU6050, and transmit the data to the Blynk software. Next the code needed to integrate with the Blynk. After that, the following lines in **Figure 6** are the library code and code needed for Hibiscus Sense to connect wifi and Blynk. There is code to define the two MPU6050. The setup function code in **Figure 8** is used to establish the connection between Hibiscus Sense and Blynk, and activate the both sensors. Furthermore, there is code to send the data to the Blynk, code to activate the sub-operation function and time interval to transmit data. In addition, there is the code to collect the required data from both sensors in the two sub-operation function and the setting of datastreams in Blynk is used to define the received data for data display. The setting of Blynk dashboard as shown in **Figure 10** is used to display the data in gauge chart form and graphically form.

**3. Result**

*3.1 Data Collection and Data Comparison*

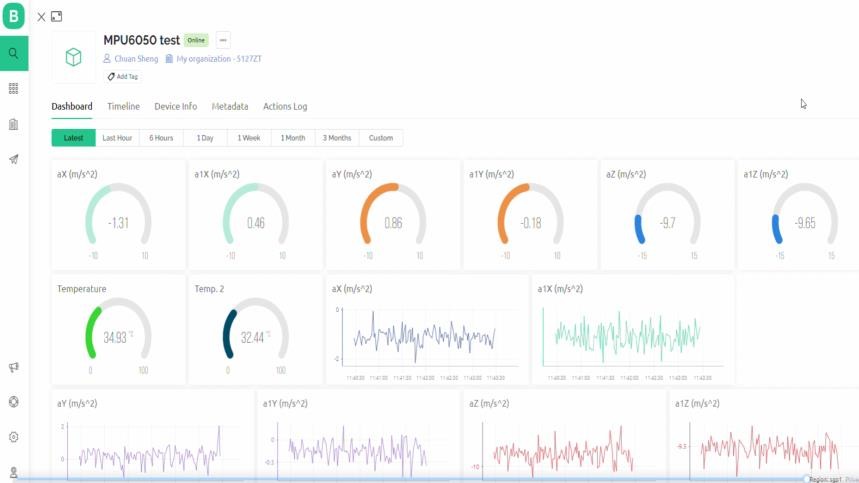
This section presents the data recorded on the machinery fault simulator in tabular and graphical form for data comparison and vibration analysis. The collected data of two MPU6050s are all in unit acceleration, m/s2. The temperature data will not be used for data acquisition and data comparison because this data is used to detect the ambient temperature of the vibration monitoring system to avoid overheating of Hibiscus Sense and GY-521 MPU6050 during data recording.

*3.1.1 Under Balanced Condition*

The machinery fault simulator creates balanced conditions when there are no bolts and nuts on the rotor as shown in **Figure 11**. The data was collected when the motor speed was set at 10RPM, 20RPM and 30RPM respectively. Data from the Blynk dashboard as shown in **Figure 12** is recorded by the computer's screen recorder for data comparison between built-in MPU6050 and GY-521 MPU6050.



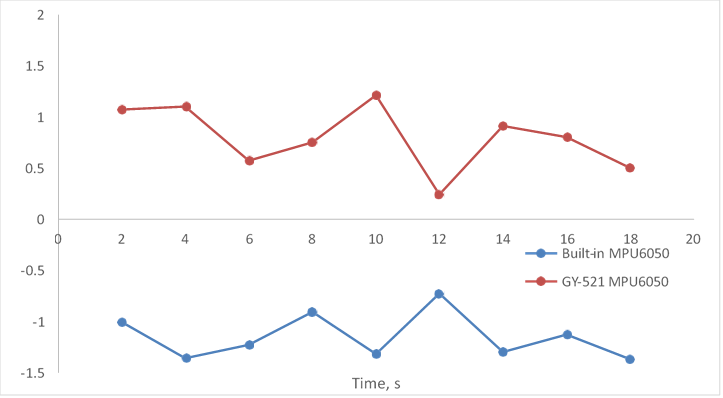
**Figure 11** Rotor with No Bolts and Nuts



**Figure 12** Displayed data in Dashboard of Blynk

**Table 3** Acceleration of Two MPU6050s when Motor Speed is 10RPM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time (s) | Acceleration of Built-In MPU6050  (m/s2) | | | Acceleration of GY-521 MPU6050  (m/s2) | | |
| X-axis | Y-axis | Z-axis | X-axis | Y-axis | Z-axis |
| 2 | -1.01 | 0.15 | -9.69 | 1.07 | -0.12 | -9.23 |
| 4 | -1.36 | 0.14 | -9.38 | 1.10 | -0.17 | -9.46 |
| 6 | -1.23 | 0.04 | -9.61 | 0.57 | -0.34 | -9.62 |
| 8 | -0.91 | 0.30 | -9.57 | 0.75 | -0.32 | -9.40 |
| 10 | -1.32 | 0.39 | -9.19 | 1.21 | -0.24 | -9.47 |
| 12 | -0.73 | 0.41 | -9.59 | 0.24 | -0.24 | -9.79 |
| 14 | -1.30 | 0.46 | -9.66 | 0.91 | -0.26 | -9.48 |
| 16 | -1.13 | 0.37 | -9.66 | 0.80 | -0.33 | -9.57 |
| 18 | -1.37 | 0.33 | -9.68 | 0.50 | -0.04 | -9.47 |



**Figure 13** Acceleration in X-axis versus Time when 10RPM motor speed

A graph with lines and dots

Description automatically generated

**Figure 14** Acceleration in Y-axis versus Time when 10RPM motor speed

A graph with lines and dots

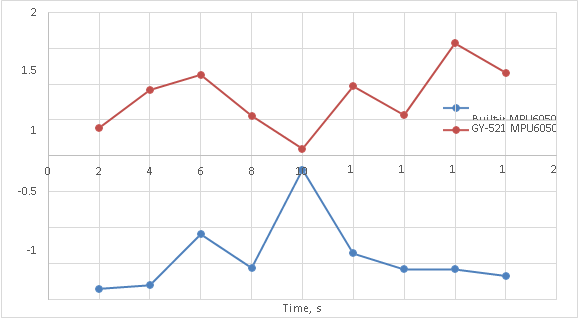
Description automatically generated

**Figure 15** Acceleration in Z-axis versus Time when 10RPM motor speed

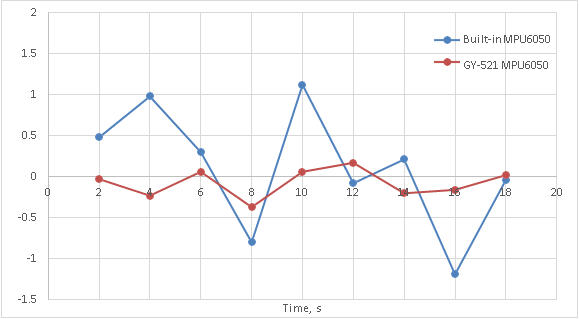
The graphs of acceleration vs. time for each axis are shown in **Figure 13, Figure 14** and **Figure 15** respectively when motor speed is set at 10RPM. The blue trend represents the data generated by the built-in MPU6050, while the red trend represents the data generated by the GY-521 MPU6050. Based on the graphs, the data between the two MPU6050s on the X-axis and Y-axis are different, but the same on the Z-axis under the balance condition measurement, the graphs of acceleration versus time on the x-axis and y-axis for both MPU6050s are in opposite areas of the graph, while the graphs of z-axis acceleration versus time for both sensors are in the same area. It can be seen from these three graphs that the built-in MPU6050 and the MPU6050 of GY-521 are operating normally due to no extreme data occurred during the balanced conditions.

**Table 4** Acceleration of Two MPU6050s when Motor Speed is 20RPM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time (s) | Acceleration of Built-In MPU6050  (m/s2) | | | Acceleration of GY-521 MPU6050  (m/s2) | | |
| X-axis | Y-axis | Z-axis | X-axis | Y-axis | Z-axis |
| 2 | -1.85 | 0.48 | -8.98 | 0.39 | -0.03 | -9.67 |
| 4 | -1.80 | 0.98 | -9.79 | 0.92 | -0.23 | -9.37 |
| 6 | -1.09 | 0.30 | -10.18 | 1.13 | 0.06 | -9.66 |
| 8 | -1.56 | -0.80 | -9.77 | 0.56 | -0.37 | -9.72 |
| 10 | -0.19 | 1.12 | -11.39 | 0.10 | 0.06 | -9.51 |
| 12 | -1.36 | -0.08 | -8.98 | 0.97 | 0.17 | -9.62 |
| 14 | -1.58 | 0.21 | -9.27 | 0.57 | -0.20 | -9.63 |
| 16 | -1.58 | -1.19 | -10.31 | 1.57 | -0.16 | -9.24 |
| 18 | -1.67 | -0.04 | -9.65 | 1.16 | 0.02 | -9.30 |



**Figure 16** Acceleration in X-axis versus Time when 20RPM motor speed



**Figure 17** Acceleration in Y-axis versus Time when 20RPM motor speed

A graph with lines and dots

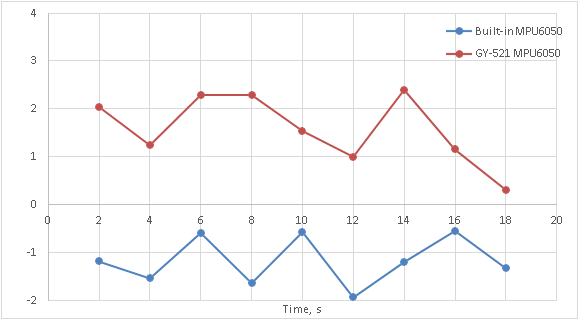
Description automatically generated

**Figure 18** Acceleration in Z-axis versus Time when 20RPM motor speed

It can be clearly seen from the graphs of acceleration versus time for each axis **in Figure 16**, **Figure 17** and **Figure 18** that when the motor speed is set to 20RPM, the built- in MPU6050 is more sensitive to machinery vibration than the GY-521 MPU6050.

**Table 5** Acceleration of Two MPU6050s when Motor Speed is 30RPM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time (s) | Acceleration of Built-In MPU6050  (m/s2) | | | Acceleration of GY-521 MPU6050  (m/s2) | | |
| X-axis | Y-axis | Z-axis | X-axis | Y-axis | Z-axis |
| 2 | -1.19 | 0.17 | -8.82 | 2.04 | -0.63 | -9.86 |
| 4 | -1.54 | -0.45 | -9.25 | 1.24 | -0.49 | -9.23 |
| 6 | -0.59 | -2.75 | -8.39 | 2.29 | 0.19 | -9.79 |
| 8 | -1.64 | 1.57 | -11.70 | 2.29 | 0.19 | -9.79 |
| 10 | -0.57 | -0.43 | -10.44 | 1.54 | -0.36 | -9.91 |
| 12 | -1.94 | -1.48 | -9.29 | 1.00 | -0.72 | -9.01 |
| 14 | -1.20 | 3.75 | -9.92 | 2.40 | 0.19 | -9.17 |
| 16 | -0.55 | 2.67 | -8.44 | 1.15 | 0.17 | -9.01 |
| 18 | -1.33 | -0.09 | -11.47 | 0.31 | 0.41 | -9.65 |

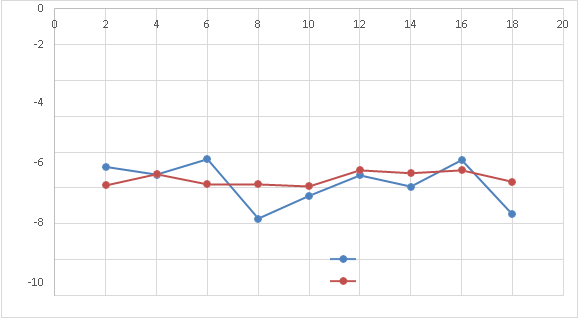


**Figure 19** Acceleration in X-axis versus Time when 30RPM motor speed

A graph with lines and dots

Description automatically generated

**Figure 20** Acceleration in Y-axis versus Time when 30RPM motor speed



**Figure 21** Acceleration in Z-axis versus Time when 30RPM motor speed

From the graphs of acceleration versus time for each axis in **Figure 19**, **Figure 20** and **Figure 21**, it also can be clearly seen that when the motor speed is set to 30RPM, the built-in MPU6050 is more sensitive to machinery vibration than the GY-521 MPU6050.

In summary, the graphs in this section prove that the built-in MPU6050 and the GY- 521 MPU6050 can be used for vibration analysis due to the absence of extreme data under the balanced condition simulated by the machinery fault simulator. This part also proves that the built-in MPU6050 is more sensitive to vibration as compared to the GY-521 MPU6050 when increasing the motor speed to 20RPM and 30RPM.

*3.1.2 Under Unbalanced Condition*

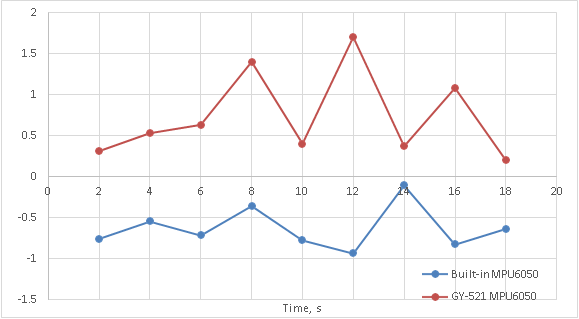
As shown in **Figure 22**, the machinery fault simulator creates unbalanced conditions when there are three bolts and nuts on the rotor. The data was collected when the motor speed was set at 10RPM, 20RPM and 30RPM respectively. Data from the Blynk dashboard is also recorded by the computer's screen recorder. The results in this section are form in tabular form and graphical form for data analysis.



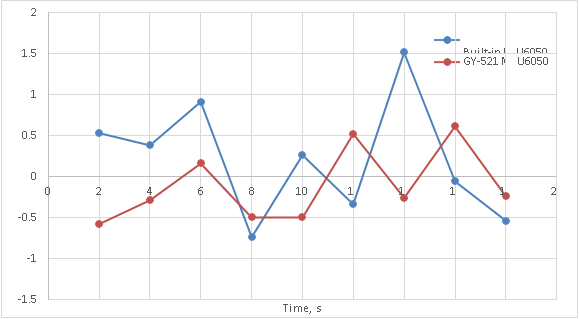
**Figure 22** Rotor with three bolts and nuts

**Table 6** Acceleration of Two MPU6050s when Motor Speed is 10RPM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time (s) | Acceleration of Built-In MPU6050  (m/s2) | | | Acceleration of GY-521 MPU6050  (m/s2) | | |
| X-axis | Y-axis | Z-axis | X-axis | Y-axis | Z-axis |
| 2 | -0.76 | 0.53 | -9.30 | 0.31 | -0.58 | -9.58 |
| 4 | -0.55 | 0.38 | -9.40 | 0.53 | -0.29 | -9.69 |
| 6 | -0.72 | 0.91 | -9.41 | 0.63 | 0.16 | -9.36 |
| 8 | -0.36 | -0.74 | -9.08 | 1.40 | -0.50 | -9.55 |
| 10 | -0.78 | 0.26 | -9.21 | 0.40 | -0.50 | -9.55 |
| 12 | -0.94 | -0.34 | -9.57 | 1.70 | 0.52 | -9.61 |
| 14 | -0.11 | 1.51 | -8.76 | 0.37 | -0.26 | -9.60 |
| 16 | -0.83 | -0.06 | -9.22 | 1.08 | 0.61 | -9.33 |
| 18 | -0.64 | -0.54 | -9.21 | 0.20 | -0.24 | -9.44 |



**Figure 23** Acceleration in X-axis versus Time when 10RPM motor speed



**Figure 24** Acceleration in Y-axis versus Time when 10RPM motor speed

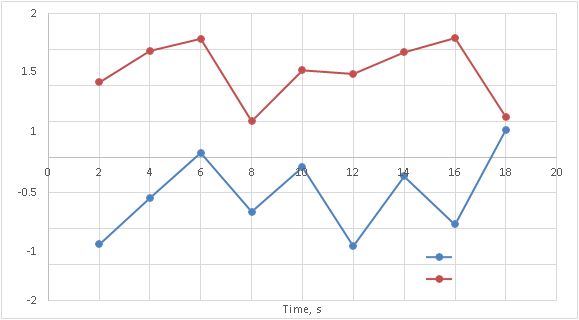
A graph with lines and dots

Description automatically generated

**Figure 25** Acceleration in Z-axis versus Time when 10RPM motor speed

**Table 7** Acceleration of Two MPU6050s when Motor Speed is 20RPM

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Time (s) | Acceleration of Built-In MPU6050  (m/s2) | | | Acceleration of GY-521 MPU6050  (m/s2) | | | |
|  | X-axis | Y-axis | Z-axis | X-axis | Y-axis | Z-axis |
| 2 | -1.22 | -0.16 | -10.61 | 1.04 | -0.03 | -10.03 |
| 4 | -0.57 | 0.06 | -9.95 | 1.48 | -0.49 | -9.05 |
| 6 | 0.06 | -0.57 | -9.36 | 1.65 | 0.02 | -9.86 |
| 8 | -0.76 | 0.52 | -8.53 | 0.50 | 0.02 | -9.24 |
| 10 | -0.14 | -0.84 | -9.83 | 1.21 | 0.56 | -9.90 |
| 12 | -1.24 | 1.54 | -9.17 | 1.16 | 0.34 | -9.83 |
| 14 | -0.27 | -0.24 | -9.32 | 1.46 | -0.56 | -9.10 |
| 16 | -0.94 | 0.45 | -10.09 | 1.66 | -0.24 | -9.01 |
| 18 | 0.38 | -0.30 | -9.89 | 0.56 | -0.13 | -10.12 |



**Figure 26** Acceleration in X-axis versus Time when 20RPM motor speed



**Figure 27** Acceleration in Y-axis versus Time when 20RPM motor speed

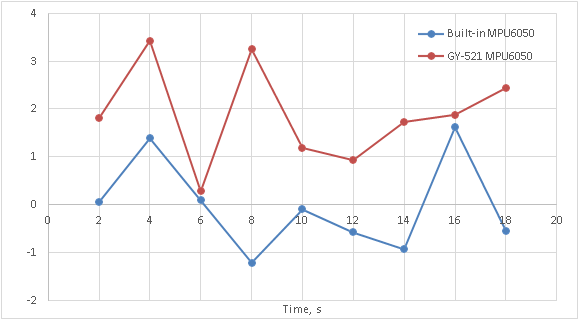
A graph with lines and dots

Description automatically generated

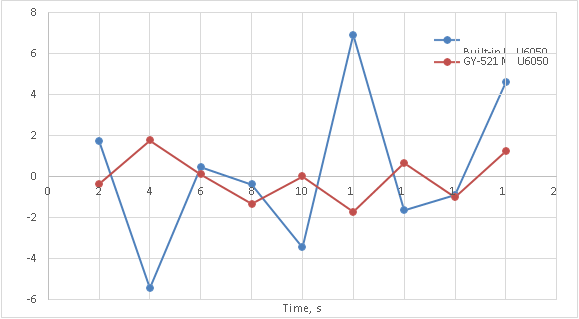
**Figure 28** Acceleration in Z-axis versus Time when 20RPM motor speed

**Table 8** Acceleration of Two MPU6050s when Motor Speed is 30RPM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time (s) | Acceleration of Built-In MPU6050  (m/s2) | | | Acceleration of GY-521 MPU6050  (m/s2) | | |
| X-axis | Y-axis | Z-axis | X-axis | Y-axis | Z-axis |
| 2 | 0.06 | 1.73 | -9.15 | 1.81 | -0.36 | -9.99 |
| 4 | 1.39 | -5.44 | -9.74 | 3.43 | 1.75 | -9.94 |
| 6 | 0.09 | 0.46 | -5.99 | 0.28 | 0.10 | -9.74 |
| 8 | -1.21 | -0.39 | -9.74 | 3.25 | -1.33 | -8.75 |
| 10 | -0.10 | -3.46 | -9.81 | 1.19 | 0.00 | -10.12 |
| 12 | -0.58 | 6.91 | -14.81 | 0.93 | -1.72 | -9.20 |
| 14 | -0.93 | -1.65 | -9.13 | 1.73 | 0.66 | -7.47 |
| 16 | 1.62 | -0.90 | -7.70 | 1.88 | -1.01 | -9.92 |
| 18 | -0.54 | 4.60 | -9.46 | 2.44 | 1.23 | -9.93 |



**Figure 29** Acceleration in X-axis versus Time when 30RPM motor speed



**Figure 30** Acceleration in Y-axis versus Time when 30RPM motor speed

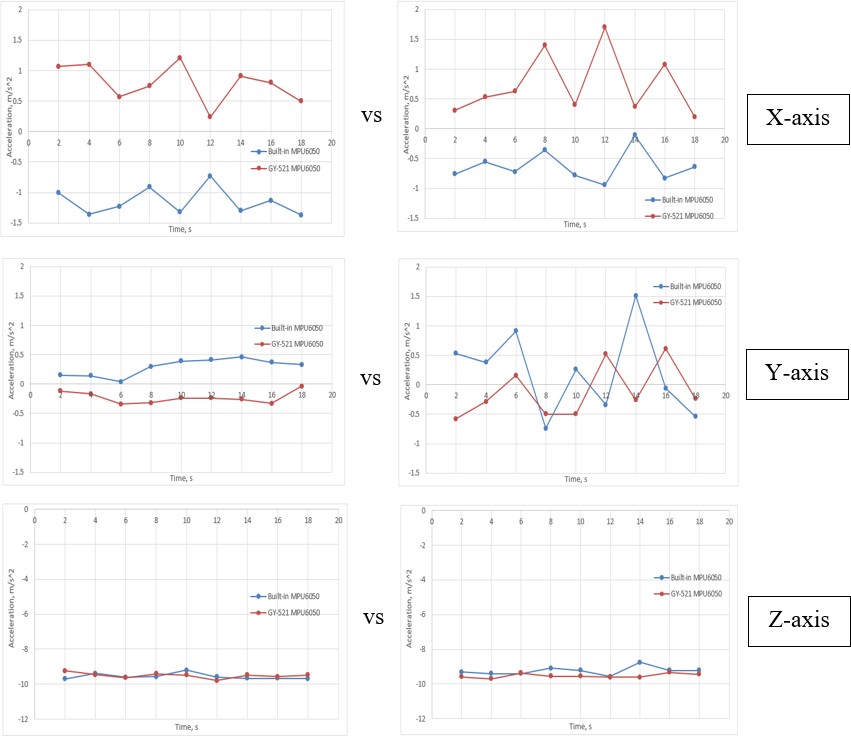
A graph with lines and dots

Description automatically generated

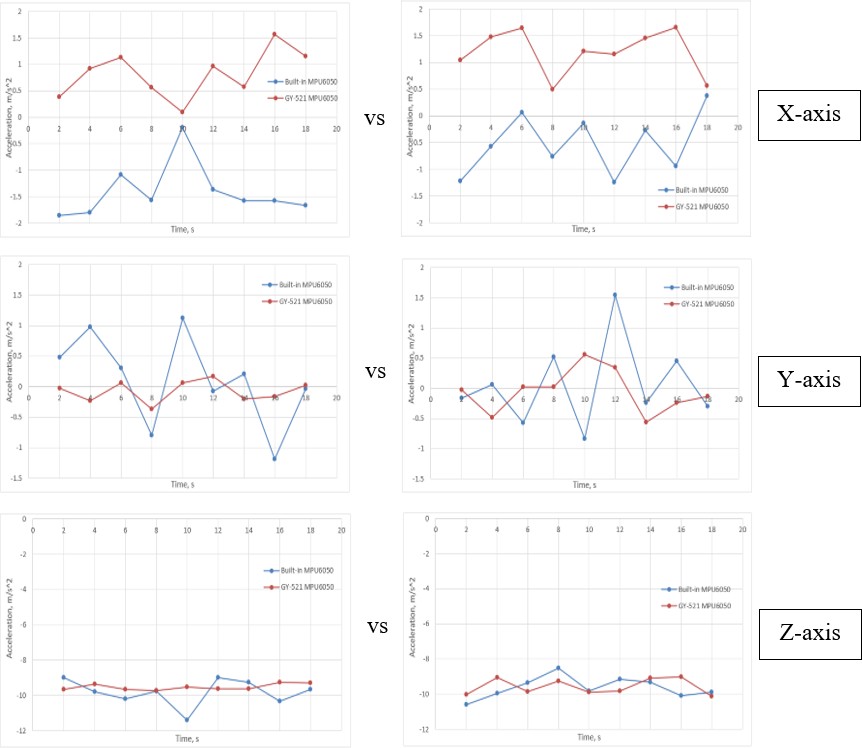
**Figure 31** Acceleration in Z-axis versus Time when 30RPM motor speed

**4. Discussion**

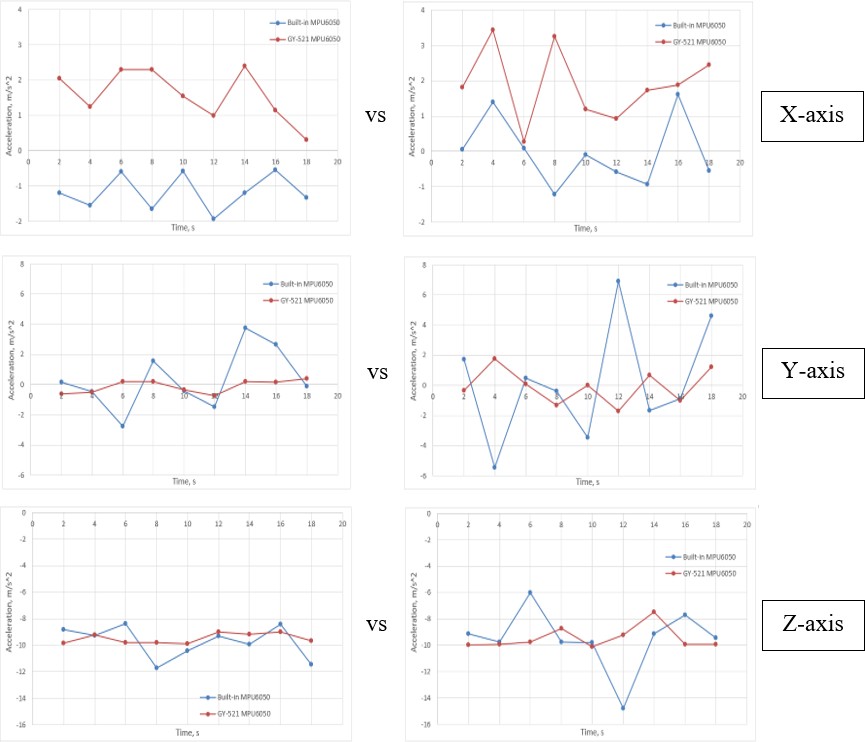
In this section, data comparison under balanced and unbalanced conditions will be performed. The purpose of this section is to prove that the integrated low-cost vibration monitoring system of this project can be used for industrial vibration analysis. In the figures below, the graph on the left is the data under balanced conditions and the graph on the right is the data under unbalanced conditions. Both horizontally and vertically phase shifted induced by vibration components are also likely in an unbalanced system[8].



**Figure 32** Acceleration versus Time when motor speed is 10RPM



**Figure 33** Acceleration versus Time when motor speed is 20RPM



**Figure 34** Acceleration versus Time when motor speed is 30RPM

In the **Figure 32**, **Figure 33** and **Figure 34**, there are the graphs of acceleration versus time. It can be clearly seen that the difference between the trend points in the unbalanced condition is higher than the balanced condition in the x-axis, y-axis and z-axis. Accordingly, it also shows that the higher the motor speed, the more pronounced the difference between the trend points under balanced and unbalanced conditions[7]. In a machine where there is an imbalance problem we will find an increase in the vibration amplitude in radial measurements, while axial vibration measurements may remain low[9]. Therefore, the data comparison in this section proves that the integrated low-cost vibration monitoring system of this project can be used for industrial vibration analysis.

**5. Conclusion**

As a conclusion, through the verification of the results, a wireless integrated low cost vibration monitoring system was successfully constructed, which means that the system has achieved the objectives. This vibration monitoring system is able to send data to Blynk software through wifi connection, and the data in Blynk is stored in Blynk IoT cloud, the study demonstrates the feasibility of implementing IoT architecture for monitoring machine conditions.

**Acknowledgement**

Authors would like to express gratitude and appreciation to Faculty of Mechanical Engineering and Technology (FTKM) and Centre of Research and Innovation, Universiti Teknikal Malaysia Melaka (UTeM) for the financial assistance on conference presentation and participation.

**References**

[1] T. Wu, Y. Chung, and C. Liu, ““Looseness diagnosis of rotating machinery via vibration analysis through hilbert– huang transform approach,” Journal of Vibration and Acoustics, vol. 132, no. 3, pp. 1–9, 2010.

[2] B. Alili, A. Hafaifa, and A. Iratni, “Faults detection based on fuzzy concepts for vibrations monitoring in gas turbine,”Diagnostyka, vol. 21, no. 4, pp. 67–77, 2020.

[3] J. Wang, L. Ye, R. X. Gao, C. Li, and L. Zhang, “Digital twin for rotating machinery fault diagnosis in smart manufacturing,” International Journal of Production Research, vol. 57, no. 12, pp. 3920–3934, 2019.

[4] H. S. Shah, P. N. Patel, S. P. Shah, and M. T. Thakker, “Cost Effective Low Power Vibration Monitoring System,” in *IEEE Xplore*, Apr. 2017, pp. 62–68. doi: 10.1109/TIIEC.2014.019.

[5] R. V. Mukane, N. M. Gurav, S. Y. Sondkar, and N. C. Fernandes, “Labview based implementation of fuzzy logic for vibration analysis to identify machinery faults,” in Proceedings of the International Conference on Computing, Communication, Control and Automation, pp. 1–5, Pune, India, August 2017.

[6] T. Han, B. S. Yang, and Z. J. Yin, “Feature-based fault diagnosis system of induction motors using vibration signal,” Journal of Quality in Maintenance Engineering, vol. 13, no. 2, pp. 163–175, 2007.

[7] J. Runesson, “Vibration analysis for condition monitoring of mechanical presses,” Master’s +esis, Lund University, Lund, Sweden, 2019.

[8]. M. Heydarzadeh, N. Madani, and M. Nourani, “Gearbox fault diagnosis using power spectral analysis,” in Proceedings of the IEEE International Workshop on Signal Processing Systems, pp. 242–247, Dallas, TX, USA, October 2016.

[9] T. Wu, Y. Chung, and C. Liu, ““Looseness diagnosis of rotating machinery via vibration analysis through hilbert–huang transform approach,” Journal of Vibration and Acoustics, vol. 132, no. 3, pp. 1–9, 2010.

1. [↑](#footnote-ref-1)