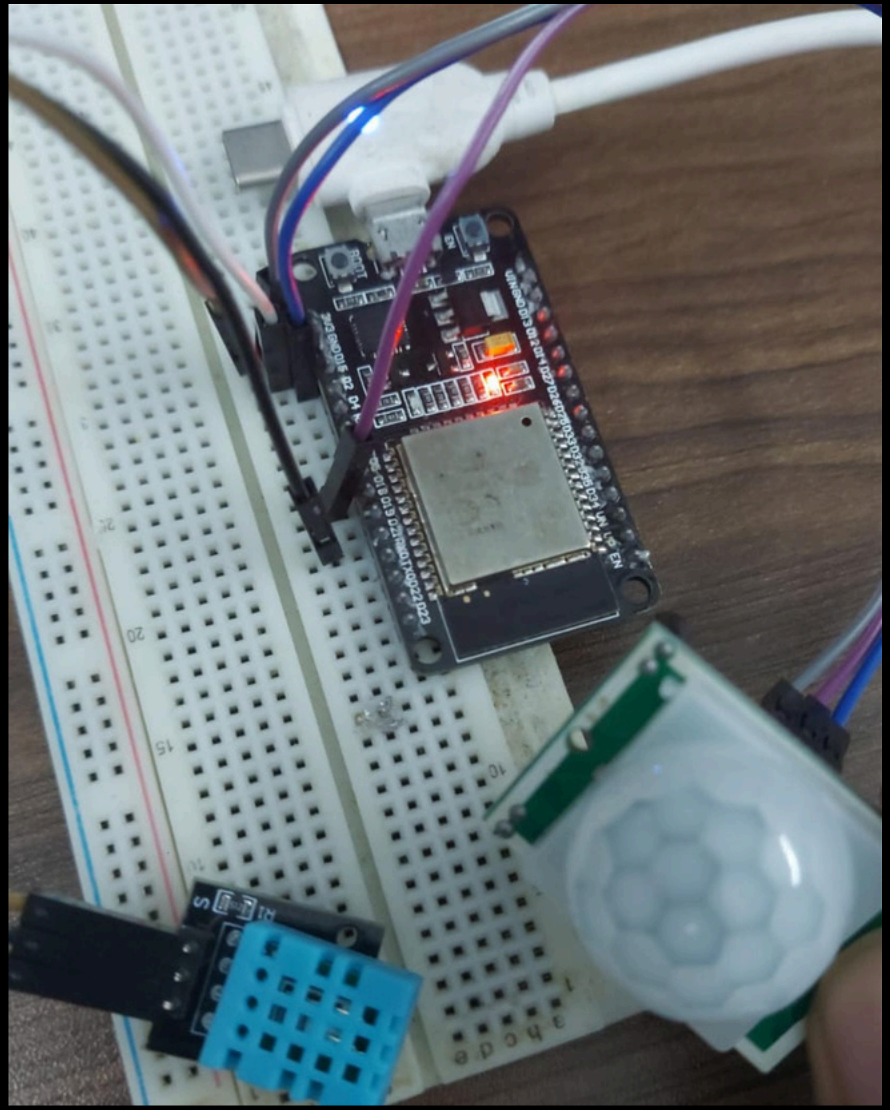
IOT BASED INDUSTRIAL EQUIPMENT MONITORING

By Krish Soliya, Varun Kashyap, Aditya Gupta

INTRODUCTION



IoT-based industrial equipment monitoring revolutionizes the way we oversee machinery and processes, bringing real-time insights to the factory floor. By seamlessly integrating smart sensors, data acquisition devices, and cloud platforms, this cutting-edge solution tracks key parameters like temperature, pressure, vibration, and energy consumption.

HISTORICAL DEVELOPMENT

- **Manual Monitoring (Pre-20th Century):** Manual inspections using basic tools; reactive and error-prone.
- **Basic Instrumentation (Early 20th Century):** Mechanical sensors and analog meters enabled real-time data and basic alarms.
- **Electromechanical Systems (Mid-20th Century):** Integration of electrical components and centralized control rooms for proactive monitoring.
- **Digital Revolution (1970s-1990s):** Microprocessors, PLCs, and SCADA systems improved automation and remote monitoring.
- **Networked Systems & IoT (1990s-2010s):** IoT and wireless technologies enabled cloud-based real-time monitoring and analytics.

GOALS

REAL TIME MONITORING

OPERATIONAL EFFICIENCY

PREDICTIVE MAINTENANCE

COST REDUCTION

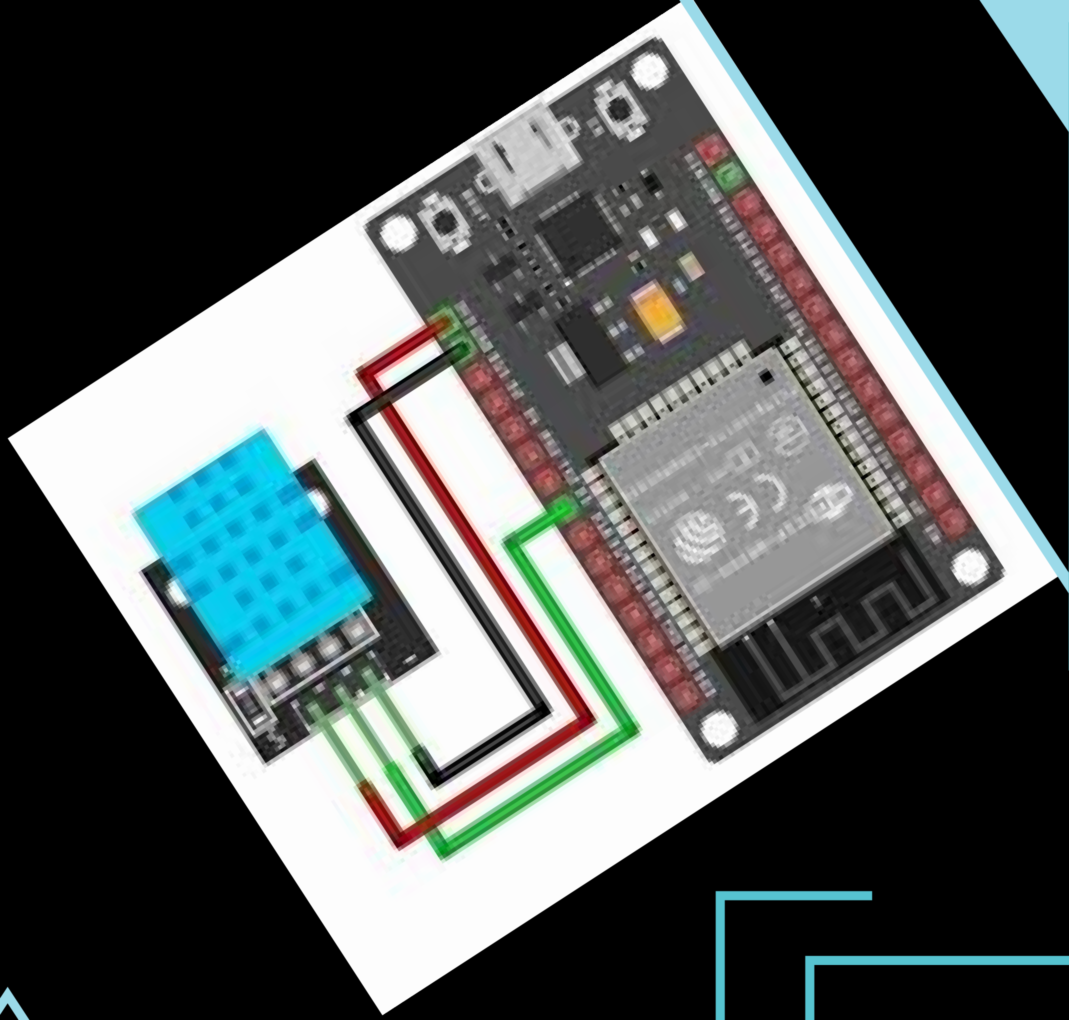
IMPROVED SAFETY

PROBLEMS

- Delayed Insights: Lack of real-time data hinders quick responses.
- High Costs: Reactive maintenance leads to frequent downtimes.
- Limited Predictions: Conventional systems can't predict failures.
- Scalability Issues: Expansion requires costly upgrades.
- Data Silos: Poor integration limits operational visibility.
- Human Dependency: Manual inspections are error-prone and inefficient.
- Energy Wastage: Inefficient resource usage increases costs.

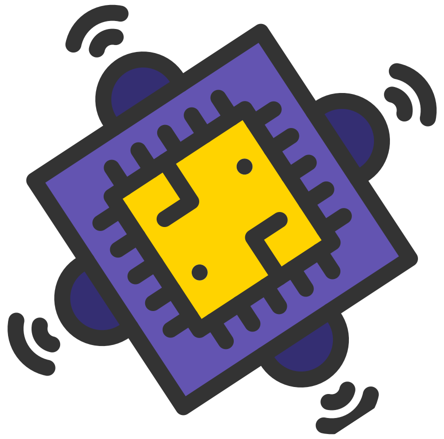
KEY COMPONENTS

- **Temperature Sensors (DHT Sensor):** Monitor machinery temperature.
- **Proximity Sensors (PIR Sensor):** Track positions of moving parts.
- **ESP32:** Microcontroller for sensor data collection and wireless transmission.
- **ML Models:** Analyze and predict equipment failures using historical and real-time data.
- **Cloud Platform (Influx DB):** Store, process, and analyze sensor data with integrated ML tools.



influxdb

PHYSICAL LAYER

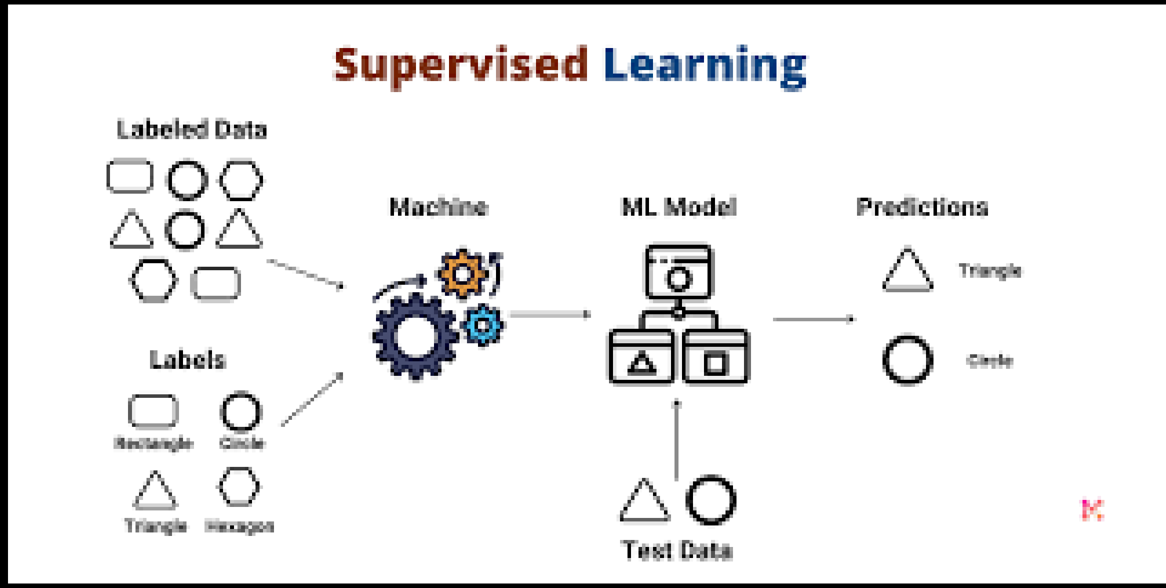


The Physical Layer forms the foundation of the IoT model, encompassing sensors and microcontrollers. In this project:

- Sensors like Proximity Sensors and DHT Sensors collect real-world data such as position, temperature, and humidity.
- ESP32 acts as the microcontroller, processing sensor data and transmitting it to higher layers for analysis and decision-making.

ML MODEL

Handling Missing Data



The model was designed to automatically handle missing values by applying techniques such as imputation, where missing data points were replaced with the mean, median, or using predictions based on other variables. This ensured data integrity and consistency during training.

ML MODEL

Data Collection and Preparation

- Sample data was collected over a period, including temperature, humidity, and proximity sensor readings.
- This data was preprocessed to clean any noise, normalize values, and extract meaningful features for model training.

Data Splitting

- The dataset was divided into training and testing subsets to validate the model's performance.
- The training set was used to teach the model patterns, while the testing set evaluated its ability to make accurate predictions.

ML MODEL

Observations and Predictions

- From the split data, critical observations were identified, such as trends in temperature or changes in proximity that could indicate equipment issues.
- Using these insights, the model generated predictions about potential equipment faults and their timing.

Validation with Real-Life Scenarios

- The predictions were compared with real-life equipment performance to verify the accuracy of the model.
- Feedback from these validations was used to refine the model further, ensuring reliable predictions.

ML MODEL

Regression Analysis for Model Optimization

- Various regression techniques were tested (e.g., Linear Regression, Polynomial Regression, Logistic Regression) to identify the most suitable approach for analyzing sensor data.
- The method that provided the best accuracy and reliability was chosen for the final model.

Model Deployment and Sensor Integration

- Once the model was trained and optimized, it was linked with real-time sensor data from the IoT system.
- The model continuously processes incoming data, predicts equipment conditions, and provides actionable insights.

CLOUD LAYER

- InfluxDB serves as the cloud storage platform for sensor data.
- Data generated by Proximity Sensors and DHT Sensors is stored in organized buckets.
- Enables centralized data management, real-time access, and long-term storage.
- Supports advanced analytics, visualization, and integration with machine learning models.
- Provides scalability and reliability for handling large volumes of IoT data.

WORKING

Data Collection

- Sensors, including Proximity Sensors and DHT Sensors, collect critical data such as temperature, humidity, and equipment positioning.
- These sensors are connected to the ESP32 microcontroller, which processes the raw data locally.

Physical Layer

- The ESP32 is connected to a Wi-Fi network, forming the Physical Layer of the IoT model.
- This layer facilitates the initial collection and transmission of data from the sensors to the centralized network.

WORKING

Centralized Network

- All devices, including sensors and ESP32, are connected via a centralized Wi-Fi network to ensure seamless communication.
- This network enables the smooth flow of data to the cloud storage layer.

WORKING

Cloud Storage with InfluxDB

- Sensor data is sent from the ESP32 to InfluxDB, a high-performance cloud database designed for real-time data storage and management.
- Data is stored in buckets, allowing efficient organization and retrieval for further analysis.

Data Transfer to Google Colab

- Real-time data stored in InfluxDB is transferred to Google Colab for advanced analytics and machine learning.
- This ensures seamless integration between cloud storage and the machine learning model.

WORKING

Machine Learning for Predictive Analysis

- A machine learning model is trained and deployed on Google Colab to analyze the sensor data.
- The model predicts the condition of industrial equipment based on patterns in temperature and proximity sensor readings.
- Predictive insights help anticipate potential failures and optimize maintenance schedules.
- The results of the machine learning predictions are displayed on a dashboard for easy visualization and decision-making.

CONCLUSION

IoT-based industrial equipment monitoring revolutionizes traditional practices by enabling real-time data collection, predictive maintenance, and operational efficiency. Integrating advanced sensors, microcontrollers like ESP32, and cloud platforms, this system reduces downtime, lowers costs, and enhances safety. Machine learning further empowers the solution by predicting equipment failures, ensuring timely interventions. This approach not only optimizes resource utilization but also lays the foundation for smarter, more sustainable industrial operations in the era of Industry 4.0.

THANK YOU