An Implementation of an Industrial Internet of Things on an SMT Assembly Line

Gaurav Bhandari, Prakash Hiremath M., Ashish Joglekar, Anupama Kulkarni, Devadatta Kulkarni, C. Mahadeva, Sampad B. Mohanty, D. Raghunath, M. B. Raju, Rajeev Shorey, Rajesh Sundaresan

***Abstract*—Industry 4.0 is the latest trend in the manufacturing sector that focuses on intelligent manufacturing and smart factories. This leads to opportunities in automation, optimization, asset management and predictive maintenance, which helps reduce downtime and increase revenue. In this paper, we propose the solution that was created for a particular SMT PCB manufacturing facility in Mysuru (Vinyas IT), the features it has to offer and the methodologies that were implemented in order to achieve our goals. We will also highlight the important aspects of the solution that will be showcased during the demonstration and the impact of our solution.**

***Index Terms*—PCB, SMT, IIoT, BLE, MQTT, MODBUS, Middleware, Vinyas IT, Elasticsearch, Kibana**

I. Introduction

The Industrial Internet of Things (IIoT) employs numerous sensors in a manufacturing assembly line, for monitoring component machines and the surrounding ecosystem. The data that is retrieved from the sensors is used to provide valuable insights to factory managers for better tracking of their assets, better inventory management, increased situational awareness, more opportunities for efficiency and for cutting of costs and new insights for predictive maintenance, leading to decreased down time.

In order to demonstrate the aforementioned potential, we built a complete prototype IIoT system for use at a surface mount technology (SMT) assembly line – Vinyas Innovative Technologies Pvt. Ltd. (Vinyas IT). We chose the SMT assembly line and particularly the one at Vinyas on account of the sufficient yet manageable complexity that the line offered to showcase the power of IIoT.

In this demonstration (demo), we will highlight the following capabilities of our solution.

1) Tracking assets and providing situational awareness: We do this via algorithms that infer various machine states, detect anomalous events, correlate data across the line, associate them with boards, thereby providing end-to-end trace-ability of the assembled product. The identification of machine states enables us to detect anomalous line stoppages and provide alerts. We also monitor vibrations and current usages to detect impending failures that could trigger planned or preventive maintenance and ultimately reduce downtime.
2) Energy profiling: We map the energy usage landscape by the use of various energy sensing devices. This provides insights for possible optimisations.
3) Energy usage optimisation: We simulate the factory line via a digital twin and explore the possible optimisations through 'what-if' analysis.

Additional capabilities of our solution include our ability to keep track of throughput (number of boards manufactured), machine parameters under normal operation (idle times, processing times), machine usage parameters (time of use, maintenance, etc.).

The continuous monitoring of vibrations and energy consumption provides insights on machines with the largest energy consumption for optimisation or machines with abnormal energy consumption or vibration patterns for preventive maintenance.

We also have a 2D dashboard for better visibility of factory state. In addition, a 3D rendering provides a replica of the line for a more realistic visualisation.

Finally, we have built a digital twin that enables exploration of new configurations for optimisation of the factory line. It also enables a comparison of the ideal outcome with the actual outcome in the factory.

II. Methodology

We partnered with Vinyas Innovative Technologies Pvt. Ltd., an electronics manufacturing unit, for experimentation. The line that was instrumented by the team was a Surface Mount Technology (SMT) assembly line that takes in printed circuit boards (PCB) and produces PCBs mounted with components after traversing the line. The line includes a loader, a screen printer, a pair of pick and place machines and a reflow oven as shown in figure 1.

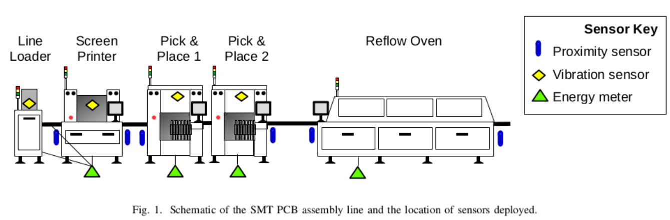

Fig. 1. Schematic of the SMT PCB assembly line and sensor locations

1) Sensing Modalities and Data Communication Architecture: Various sensors such as vibration, proximity and temperature sensors and current meters were instrumented across the line (fig 1). An IoT network for the data-flow was also setup. The data-flow architecture shows the flow of data from the sensors on the IoT End-Devices (in the factory, Vinyas, Mysuru) to the database on the IoT data store server (at the Robert Bosch Centre for Cyber-Physical Systems, IISc, Bengaluru). This is done via the IoT Gateways and the Middleware server depicted in figure 2. The IoT end-devices (IEDs) contain the sensors and gather data periodically by reading the sensors attached to them and relay them to the IoT Gateway over a suitable protocol like MODBUS, BLE, or directly over the GPIO pins. The end devices can talk to the IoT Gateway (IGW) using respective drivers which then forwards the data to the MQTT Broker (using open source Mosquitto) and stores them in a message queue. This data is then forwarded to the Middleware server over HTTP REST API. The IoT middleware is the local data store where all the sensor data is accumulated in the Elasticsearch database.

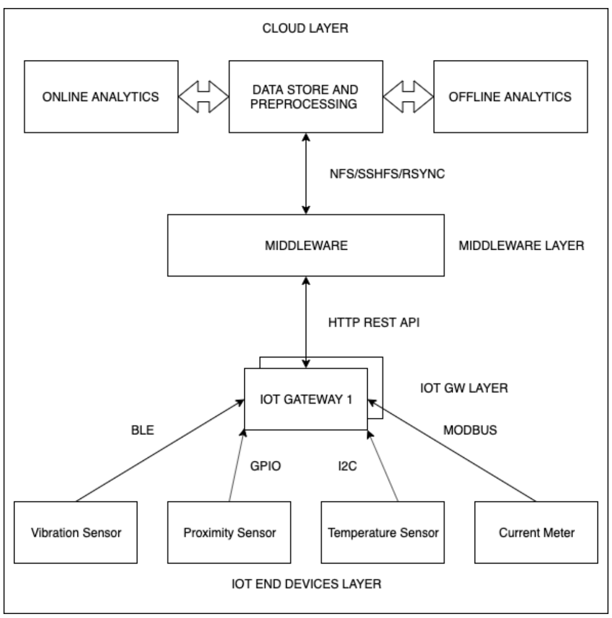

Fig. 2. Data Flow Architecture

2) Data Analysis and Line Inferences: The sensor data stored in the Middleware is then used for analysis. With the current architecture, various data analysis tools and statistical techniques have been used to infer the different events that take place during the production of PCBs across each machine on the assembly line and associate the events across the line to provide end-to-end trace-ability. From the current analysis algorithms, the following parameters for each machine are inferred:

• Machine States: Idle,Printing, Processing, Maintenance, • Number of PCBs processed during the given period

• Arrival and departure instants of PCBs
• PCB processing time
• Delays in machines due to maintenance
• Machine Utilization Factor
• Energy consumed by each machine per board

Similarly, there are other anomalous events that take place on the line such as component refills and line restarts. These events have signatures that are different from board processing. In order to corroborate these signatures with the ground truth, operators on the line were given tablets with an interface with push buttons to indicate events that took place. Every time an event took place on the factory floor, the operator clicked on the corresponding button to store the event in the database with a timestamp. This can be compared with the raw data to observe any deviations in the data pattern. Using the annotated data on the factory floor, we are able to map these unusual signatures to actual events.

3) 2D Dashboard: The algorithms run in the back-end, estimate the parameters and store the results in the database. An interface that enables the factory manager to view useful statistics and parameters derived from the algorithms from a remote location, has been developed as dashboards on Kibana, a front-end to the database used. Through this, the factory manager can infer the various events occurring on the assembly line and ensure that the processes are running smoothly. The dashboards that have been created include a parameter dashboard (fig 3) and a graphical dashboard (fig 4). The parameter dashboard shows the number of boards processed by each machine, average processing times per board and average energy consumed per board. The graphical dashboard, provides absolute and relative time spent by the boards in each machine and energy distribution in various machine states. A time window selection tool enables the user to view historical data and retrieve aggregates of data over the time period selected. These dashboards will provide the factory manager with valuable insights about the assembly line and production.

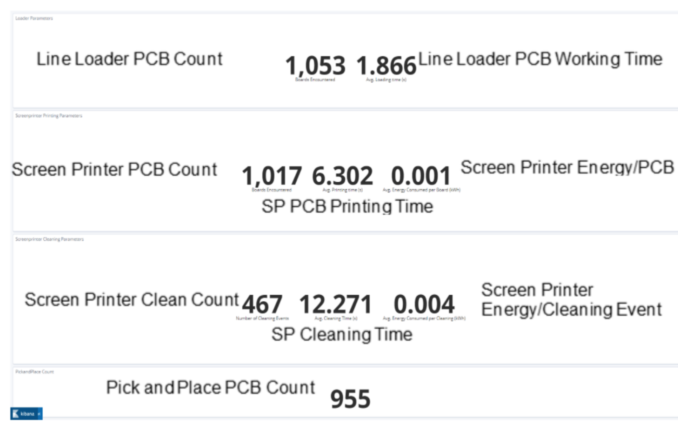

Fig. 3. Parameter Dashboard

4) Digital Twin: A parameterised simulation model, that is representative of the PCB manufacturing line in Mysuru has been built using SimPy, a discrete-event simulation library in Python. Machines and human operators are modeled on SimPy as processes and PCBs are modeled as objects. Every machine has certain jobs to complete and their behaviour has been modeled based on the various operational states they could be in such as idle, processing, waiting to output, and the corresponding actions they perform in those states. Some of

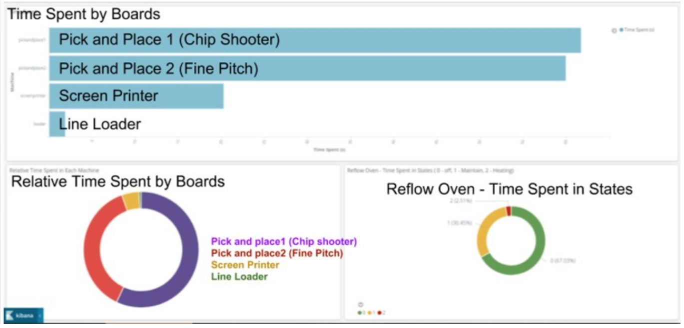

Fig. 4. Graphical Dashboard

the machines also have consumables like solder paste (Screen printer) and component reels (Pick and Place Machines). All such parameters are initialized with a certain value and a human operator is initiated by an interrupt whenever the level of a consumable falls below a set threshold, which halts the machine's processing. After the refill, the machine resumes its operation. Every process incurs a certain delay and the simulation takes place in its own environment that has its own simulation clock. Incorporating all the elements of the real assembly floor into the digital twin, we are able to simulate the actual processes and also perform what-if analysis to propose suitable optimisations for the process. The model has a simple text-based parameter setup. Support for a graphical user interface is added later as shown in the next subsection.

5) 3D Model: A 3D model of the assembly line was developed to visualise the output of the Digital-twin. The 3D model has been created in an OpenSimulator environment. OpenSimulator allows creating animations and events in this 3D environment using the OpenSimulator script language. The CAD models of the machines have been imported from the machine manufacturer's website into the OpenSim environment and real PCBs can be seen being processed within the line. Depending on the status of the machine, various colors are shown on top of each machine such as idle/waiting for board/processing. The parameter values from the real line, estimated by performing aggregate queries on the database, are input to the digital-twin in the form of a file. The file gets updated and input to the digital twin every hour to reflect the most recent trends on the line. An IRC server is used for the communication between the OpenSim 3D Model and the digital-twin. The scripted objects in the 3D model, listen to their IRC channels. The objects in the digital-twin communicate the events on their respective channels.

III. Demonstration Items

1. 3D Digital Twin Visualization with 3 parallel lines: A 3D simulation of 3 lines for a specified time period will be displayed, to compare parameters such as throughput, line efficiency, machine utilization and energy consumption. The 3 lines being showcased will be:

Line 1: Ideal line with the regular flow of PCBs from the Line Loader, through the Screen Printer, Pick and Place Machines 1 and 2 and then finally through the Reflow Oven with no line stoppages or breakdowns, except for fixed delays due to operator refills. The PCB types are fixed.

Line 2: Simulation of factory floor line movement with real line data, with all constraints modeled to represent the line on the Vinyas factory floor as closely as possible.

Line 3: Line with Buffer. Simulation of the ideal line with the buffering mechanism to compare the real line output with the outputs of optimisation explorations.

Board and Machine Statistics:

Board: While boards are navigating through the assembly line, any time a board is clicked on, board statistics such as which machine the board is in, time spent in each machine, arrival and departure instants at each machine will be provided.

Machine: Clicking on a machine, statistics such as number of boards processed by that machine, energy consumed by that machine per board, processing time per board, number of refills and number of stoppages will be displayed.

2. MQTT Diagnostic Tool: Dashboard to view real-time data that is being outputted by the sensors. This tool can be used for debugging the sensor scripts, by testing if the data is flowing. It allows visualisation of the data without the need to populate the database. Data from active or newly added sensors are automatically discovered from their topics, enabling seamless on-boarding of sensors.

3. Gateway Heartbeat Monitor and Remote Reboot: Data that is being collected from sensors and stored on gateways must not be lost. In order to ensure continuous functioning of these gateways, a Gateway Heartbeat Monitor has been developed, which gives information on the data coming in from each gateway. Any data missing from a particular gateway indicates possible issues with it and that specific gateway is rebooted by sending a request to the smart plug provider's API through HTTP.

IV. Conclusion

With so many devices connected to each other through the internet, and so much data being generated, there are endless possibilities of what can come from it. A smart manufacturing setup has been presented in this paper. The capabilities of the solution, the methodologies implemented and the impact of the solution have been highlighted. Through the insights that have been derived from this data, issues can be isolated and detected before they happen and productivity and efficiency is enhanced. Through visualizations, factory managers have more situational awareness and have remote access so there is no need for them to be physically present to solve challenges on the shop floor.

Acknowledgment

This work was supported by MHRD, SERB-IRRD of DST, and TCS under UAY grant number IISc001.

References

[1] N. Karanjkar, A. Joglekar, S. Mohanty, V. Prabhu, D. Raghunath, and R. Sundaresan, "Digital twin for energy optimization in an smt-pcb assembly line,"