

Industry Production Manager using Raspberry pi

Harsh Vardhan¹, Sahil Wadekar², Harsh Kalway³, Mrs. Sandhya Shinde⁴

^{1,2,3}BE Student, Department of E&tc DYPIEMR, Akurdi, Pune, Maharashtra, India.

⁴Assistant Professor, Department of E&tc DYPIEMR, Akurdi, Pune, Maharashtra, India.

Abstract - In today's era while manufacturing, the production database is needed to be maintained. This paper describes the method of production of products in given specified time and managing the database on system and cloud server. The job timing is considered with quantity. The complete system is built using Raspberry Pi a single board computer with algorithm written in python and with the use of Django framework for web interface and using graphical user interface for easy user interaction. Input form machine is processed and with the help of website it is connected to the owner or the maintenance team. The system also keeps the record of errors in machine to maintain it easily.

Key Words: Raspberry Pi, python, django, html, css, javascript, mysql, AWS, optocoupler, 11" touchscreen display, indicator lamps.

I. INTRODUCTION

- In every industry during manufacturing they make, build and assemble hundreds of units of products each day and during production if we can manage the number of units produced and the number of units needed to be produced more to complete the total quota for that day then it will be easy for the workers to do their job, and this is where our project comes in picture.
- There are many ways we can keep track of the current, previous and future state of our projects, we can use a computer, an Arduino, Raspberry pi and many more. We are using Raspberry pi due to two reasons: it has many features similar to computer and it's affordable.
- Our project will keep track of slots of shifts, total target of the day, time shift and many more parameters.
- To save the total progress of the project we are using internal of our project is to connect it to the cloud server and save the progress in the database so that data is safe in case of any errors.

The paper is followed with more details in following sections which is continued with conclusion and followed by references used.

Components:

Raspberry pi 4 model B:

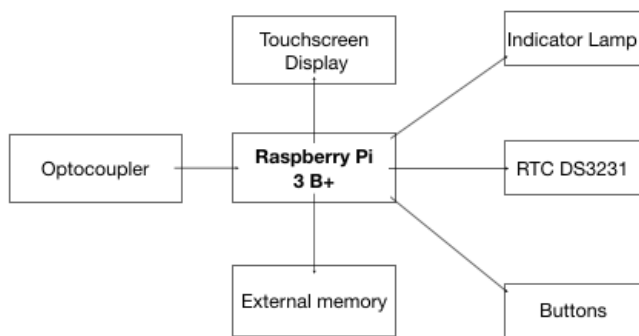
- The proposed system uses raspberry pi model 4 B which is a single board computer and offers the following features:
 - Broadcom BCM2711, Quadcore cortex-A72.
 - 1 GB RAM.
 - Gigabit Ethernet.
 - 2 USB 3.0 port, 2 USB 2.0 ports.
 - 40 GPIO pins.

The Raspberry pi runs Raspbian OS a Linux based OS and is programmed using GUI and python, which is an open source. The python contains wide range of libraries. The system uses mainly GPIO pins to interface with other devices. Touch screen display is attached to Raspberry pi using HDMI port.

II. Implementation:

- Every product produced by the manufacturing unit will give input to the system.
- System will keep proper count of the manufactured product.
- If any error occurred during the process, system will inform the maintenance department and they will check the database(previous errors in machine) for identifying the type of error generated.
- If the type of error is identified, proper service will be done according to the problem.
- If an unknown error is occurred, the employee working on that system will update the database for that error for future benefits.
- Employee will keep the time counts for every manufactured product using the input provided by the manufacturing unit.
- Employees working on that system will update their working shifts and regularly update the database.

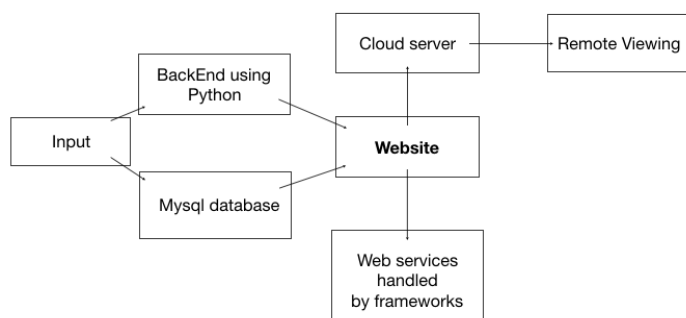
Below is the block diagram of hardware:



Block diagram of interface

- Website is designed using html, css and javascript, input to the website is given from the raspberry pi.
- Backend of the website is managed using python, mysql.
- Regular interface is done with the server for the remote viewing of the manufacturing status.

Below is the block diagram of software side implementation:



Block diagram of software

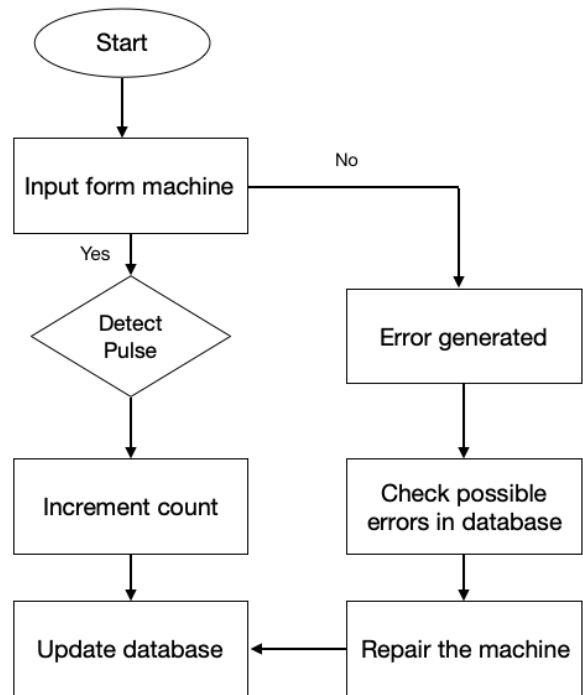
III. Flowchart:

- Input is given from machine to the raspberry pi and raspberry pi will detect whether it has received the pulse.
- If the pulse is detected then the counter of product is increased and the database is updated accordingly.
- If no pulse is generated up until the average time then downtime will start and it will show the user that some error is generated in the machine.
- Maintenance team will check the database for possible errors and repair the machine, if no records of current error is found maintenance team will enter the new record in the database for future convenience.

- All this is shown on the website which can be accessed from anywhere via the server.

Flowchart of the unit

IV. Applications:



- Can be used in any product manufacturing industry.
- Simple user webapp interface.
- Remote viewing of product manufacturing status.
- Both offline and cloud based database.
- Data protection from memory corruption with the help of CloudBase.
- Indicator lamps used to show errors generated.
- Buttons used to interface with system.
- Touchscreen display interfaced with raspberry pi as I/O device.

V. Advantages:

- Remote viewing of production using cloud server.
- Organised data management.

- Greater efficiency while manufacturing.
- Affordable than desktop computers.
- Better employee management.
- Time management.
- Easy to handle.

VI. Disadvantages:

- Increased complexity.
- Heating issue.
- Possibility of memory corruption.
- Only displays possible errors which need to be resolved manually.

VII. CONCLUSION

This module is an approach towards improving the database managing and remote control of production with cloud interface in manufacturing industry. It also improves the initial economy of industry. The timing constraint will give precise output of completion or error occurred in the machine to the proposed module.

REFERENCES

1. Tanmaya A. Onkar and Prof. P. T. Karule, "Web based maintenance for Industrial Application using Rasperry-Pi", IEEE 2016 published.
2. Suryadiputra Liawatimena, Harco Leslie Hendric Spits Warnars, "Django web framework software metrics measurement using radon and pylint", IEEE 2018 published.
3. A. Scarpellini, L. Fasanotti, A. Piccinini, S. Ierace, F. Floreani, "A web-based monitoring application for textile machinery industry", IEEE 2016 published.
4. Tussanai Parthornratt, Natchaphon Burapanonte, Wisarute Gunjarueg, "People identification and counting system using Rasperry pi", IEEE 2016 published.