



Olli Nurmiainen

# Smart Manufacturing Utilizing Mesh Network Technology

Metropolia University of Applied Sciences

Bachelor of Engineering

Electronics

Bachelor's Thesis

23 October 2024

## Abstract

Author: Olli Nurmiainen  
Title: Smart Manufacturing Utilizing Mesh Network Technology  
Number of Pages: 28 pages + 2 appendices  
Date: 28 November 2024

Degree: Bachelor of Engineering  
Degree Programme: Electronics  
Professional Major: Electronics  
Supervisors: Christian Laurén, Senior Equipment Control Manager  
Janne Mäntykoski, Senior Lecturer

---

Murata Electronics Oy, while being a leading manufacturer for automotive and medical gyro- and accelerator- sensors, has had limited Equipment monitoring systems. This thesis is a concept work for utilizing mesh network technology to monitor equipment parameters and the data integration for it. This thesis study aimed to give information about a possible solution for the issue, by utilizing mesh network technology.

The thesis contains information about different networking protocols and why utilizing mesh network technology for a wireless sensing network would be the best solution. The thesis also contains the European directives and regulations, a description of a premade prototype, mesh networking guidelines for Murata, and the data upload and implementation. There are multiple different coding languages needed for the whole project. The thesis explains some C++ code, basic SQL commands and python libraries that can be used for database uploads and data handling.

The result of the thesis study is a proposition for a self-made network that the Equipment control team can utilize for predictive maintenance. Having this type of monitoring system is beneficial for minimizing unplanned equipment breakdowns and increasing the lifetime of the equipment.

Keywords: Mesh Network, Smart Manufacturing, Data Integration

---

The originality of this thesis has been checked using Turnitin Originality Check service.

## Tiivistelmä

Tekijä:	Olli Nurmiainen
Otsikko:	Älykäs Tuotanto Hyödyntäen Hajautettua Verkkoteknologiaa
Sivumäärä:	28 sivua + 2 liitettä
Aika:	28.11.2024
Tutkinto:	Insinööri (AMK)
Tutkinto-ohjelma:	Electronics
Ammatillinen pääaine:	Electronics
Ohjaajat:	Christian Laurén, Senior Manager Janne Mäntykoski, lehtori

---

Murata Electronics Oy on käyttänyt rajoitetusti ulkopuolisia laitteiden seurantajärjestelmiä, vaikka onkin johtava valmistaja kiihtyvyy- ja gyroanturialalla. Opinnäytetyö on laadittu konseptityöksi tämän ongelman ratkaisemiseksi. Ideana on, että Murata Electronics Oy:lle tulisi hajautettu verkko, jonka avulla pystyisi lisäämään mittapäitä laitteille seurantaan varten.

Opinnäytetyö sisältää tietoa eri verkkoprotokollista ja miksi hajautettu verkko on paras ratkaisu tähän ongelmaan. Lisäksi opinnäytetyö käsittelee Euroopan direktiivejä ja regulaatioita, joita tarvitsee noudattaa, kun suunnittelee ja valmistaa järjestelmiä, esittelee prototyyppiä, sääntöjä hajautetulle verkolle Muratalla sekä datan lähetystä tietokantaan ja datan käsittelyä tietokannassa. Työssä esitetään tärkeimmät C++-koodit hajautetun verkon nimeämistä varten, tavallisia SQL-komentoja ja Python-kirjastoja, joita voi käyttää tietokantaan lähetystä varten.

Opinnäytetyön lopputuloksena oli konsepti, jonka avulla pystytään monitoroimaan laitteiden haluttuja parametrejä. Laitteiden monitorointijärjestelmä auttaa ennakoivassa huollossa, laiterikkojen välttämässä ja laitteen eliniän pidentämisessä.

Avainsanat: hajautettu verkko, älykäs tuotanto, dataintegraatio

## Contents

### List of Abbreviations

1	Introduction	1
2	Network Technologies	2
2.1	Network Protocols	2
2.2	Selecting Network Protocol	3
2.3	EU Directives to Be Considered	4
3	Mesh Network	6
3.1.1	Mesh Network Topologies	7
3.1.2	Nodes In a Mesh Network	7
3.1.3	Different Devices That Can Be Used to Create a Mesh	8
3.2	How to Create an Efficient Mesh Network	9
3.3	Benefits of Mesh Technology	12
3.4	Restrictions Using Mesh Network Technology	12
3.5	Smart Manufacturing Utilizing Mesh Networks	13
4	Showcase of Working Prototype	14
4.1	CE Mark Case Study	17
4.2	Future Project Guidelines	19
4.2.1	Troubleshooting Guideline	19
4.2.2	Naming the Mesh Correctly	20
4.2.3	Sent Data Structure	20
4.2.4	Connections Tests	21
5	Mesh Network for Murata	23
5.1	Layout For Efficient Mesh Network	23
5.2	Database Upload	25
5.3	Data Integration	25
5.4	Connection Timeout Monitoring	26
6	Conclusion	27

Appendices

Appendix 1: Schematic of prototype

Appendix 2: NORVI Datasheets

## List of Abbreviations

AI: Artificial intelligence.

CE mark: Conformité Européenne. Certification mark that ensures environmental and safety compliance with the European Union.

EPROM: Electronically Erasable Programmable Read-Only Memory. Non-volatile memory, used to store small amount of erasable and reprogrammable data.

EU: European Union

MEMS: Micro Electromechanical Systems.

MQTT Broker:

Message Queuing Telemetry Transport broker, A backend system that manages messages between different clients.

SQL: Structured Query Language. Language that is used to communicate with databases.

WLAN: Wireless Local Area Network.

WMN: Wireless Mesh Network. A WLAN topology.

WSN: Wireless Sensor Networks. A form of WMN that gather data and communicate wirelessly.

## 1 Introduction

At Murata Electronics Oy, there has been limited smart manufacturing used for equipment monitoring. The equipment has integrated sensors, but all the desired metrics are not available. This thesis work was created to provide a concept work for Murata regarding the issue by possibly adding a full-scale wireless sensor network using mesh network technologies. Murata Electronics Oy manufactures and researches electronic 3D mechanical MEMS components to the automotive and health industries. Murata Electronics Oy is a world leading MEMS sensor manufacturer, with factories around the world. The focus of the thesis work was to study and test the possibility to utilize a WMN system for smart manufacturing and to gather equipment data like temperature, current consumption, and vibration of equipment for Murata Finland. The end goal was to have a studied sensing solution for predictive maintenance and limit the amount of unplanned downtime. The benefits of having this type of a system are cost savings, improved equipment runtime and improved equipment lifetime.

The thesis will handle different types of wireless networks, what is a mesh network, how mesh networks work, how to design a mesh network, EU regulations and directives, showcasing a prototype, testing connections, database upload, data utilization, efficiency, and error prevention.

This thesis is a concept work created around a prototype sensor node, which collects temperature data from a machines exhaust line, the temperature in this case correlates directly to the output quality to the wafer. There are no other nodes created yet, but there are many possibilities to add data collection for many other equipment.

## 2 Network Technologies

There are multiple different technologies used to carry data around wirelessly. This chapter compares different wireless topologies, EU directives and what networking topology should be used for smart manufacturing at Murata Electronics oy.

### 2.1 Network Protocols

The three most used wireless local area network protocols are mesh routing-, tree link- and star link-protocol (Figure 1).

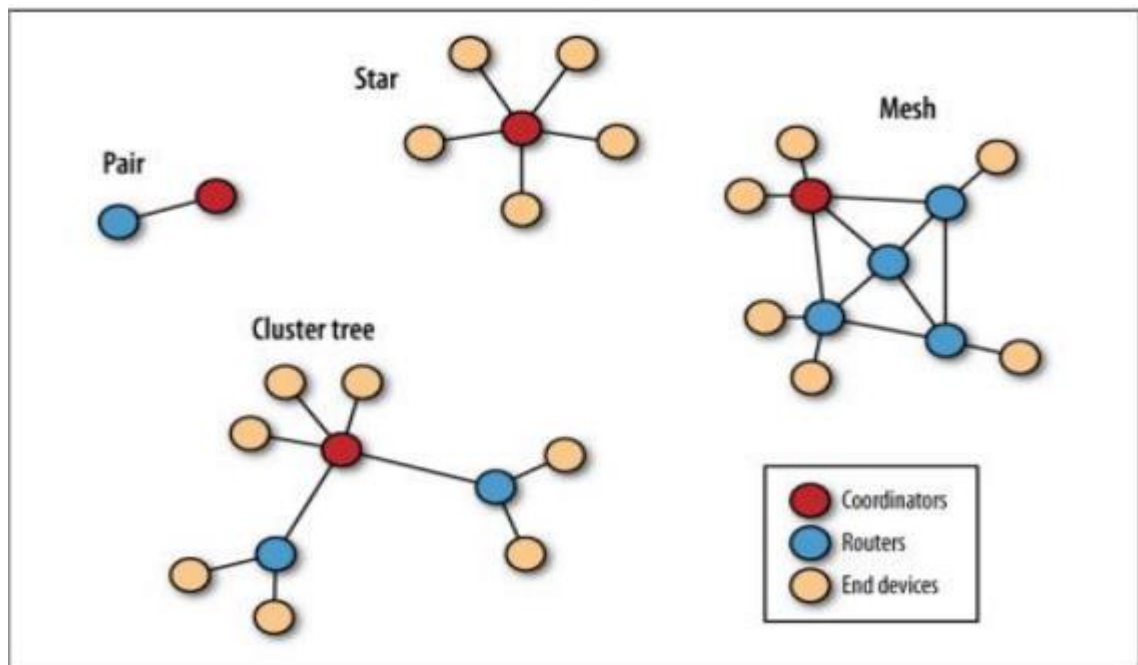


Figure 1: Wireless Topologies [1]

All the protocols listed use the same type of technology, but the mesh has the least structured out of the three and it chooses the path for the data by the shortest possible route.

The tree and the mesh protocols are similar besides the fact that the tree link branches out to multiple relay and end nodes, and the mesh does not have structure but is just one cluster of nodes that receives and transmits data. The star link protocol has one central node, and it has only end nodes connected to

it. The star link is used when the central node is positioned at the centre of the end nodes, while tree link is utilized when the pathway is more complex [1].

The above mentioned topologies can be used together as well creating a hybrid topology. The hybrid topology provides a versatile and efficient network for complex requirements. The positives for hybrid networks are efficiency, and tailored solutions. The drawbacks of this are complex management and maintenance.

## 2.2 Selecting Network Protocol

At Murata there is a need for an easily expandable and low maintenance wireless network, which can conduct measurements and send the data to a database. When deciding what protocol is used, the following was considered: price, scalability, workload, maintenance, upkeep, range, and reliability.

The star protocol was not suitable for Murata, because the end nodes cannot be far away from the central node. If a long-range star link would have been the solution, the price would be higher to get a stronger signal system that can carry messages for greater distances.

The tree link system would have been suitable, but it is more difficult to create and manage than a mesh network. If a new node was added, the routing path needs to be changed which require more work than necessary. Also, reliability and stability of the tree link is weaker due to the limited paths the data can go through.

Mesh network protocol was chosen due to the self-organizing and self-healing factors. The self-organizing makes it easy to expand. The mesh has many different paths the data can travel, making it more reliable.

### 2.3 EU Directives to Be Considered

One significant factor that needs to be considered when designing wireless networks is the EU regulations and directives it needs to follow. This is done to assure, safety, optimal performance, and compliances. The project released to the European market or to direct use by creator of the system in Europe most likely needs to be CE marked. The CE mark assures that the system follows the safety, health, and environmental EU standards.

The directives need to be read and assured that the product follows the set guidelines. This should be considered at the design part of the project. The directives needed to take into consideration are case sensitive to the project. For a mesh network the following directives need to be assessed and followed. The quotes are gathered from an official website of the European Union.

The low voltage directive has a voltage rating that the directive applies to, as stated in the Low Voltage - Directive 2014/35/EU The Directive [2, Article 1]:

This Directive shall apply to electrical equipment designed for use with a voltage rating of between 50 and 1 000 V for alternating current and between 75 and 1 500 V for direct current, other than the equipment and phenomena listed in Annex II Directive [2, Article 1].

The voltage rating of a wireless network node is usually 5–24V. It needs to be mentioned in the declaration of conformity that the directive does not meet the criteria of chapter 1, article 1.

The next directive to be taken into consideration is the Electromagnetic Compatibility Directive - Directive 2014/30/EU [3, Article 1] which states:

This Directive regulates the electromagnetic compatibility of equipment. It aims to ensure the functioning of the internal market

by requiring equipment to comply with an adequate level of electromagnetic compatibility [3, Article 1].

This directive is meant to assure that the system does not disturb other equipment and has a level of immunity for other electromagnetic disturbances.

The next directive to consider is the Radio Equipment – Directive 2014/53/EU [4]. This directive's purpose is to make reliable and well-designed radio equipment that supports the radio spectrum and does not cause harmful interference [4]. When creating a wireless network, the components used are usually very standard components which have a CE mark already, hence following this directive should not be an issue. The above-mentioned directive still needs to be mentioned in the declaration of conformity, that it has been taken into consideration when creating the system.

Components like capacitors and inductors may contain hazardous substances. There is a Restriction of Hazardous Substances in Electrical and Electronic Equipment – Directive 2011/65/EU. The Directive [5, article 1] states that:

This Directive lays down rules on the restriction of the use of hazardous substances in electrical and electronic equipment (EEE) with a view to contributing to the protection of human health and the environment, including the environmentally sound recovery and disposal of waste EEE [5, article 1].

In the system designing phase there needs to be knowledge on the components in the system and the correct repair and disposal of said components.

The last regulation to be taken into consideration is the Eco-design requirements for sustainable products - Regulation (EU) 2024/1781 [6, article 1], states that:

This Regulation establishes a framework for the setting of ecodesign requirements that products have to comply with to be placed on the market or put into service, with the aim of improving the environmental sustainability of products in order to make

sustainable products the norm and to reduce the overall carbon footprint and environmental footprint of products over their life cycle, and of ensuring the free movement of sustainable products within the internal market [6, article 1].

The purpose of this regulation, as stated before, is to make sure the system is sustainable and environmentally safe throughout the products lifetime.

The CE mark ensures that the directives and regulations have been followed. The CE mark requires a risk analysis to be done as well, and that the system or product has all the needed technical documentation and the required operational manuals in the required language. After all the documentations and analyses have been completed, the declaration of conformity can be done, and the CE mark process can be forwarded to the authorized person who can sign the declaration of conformity and the system can be CE certified.

### **3 Mesh Network**

Mesh networks are a WLAN (wireless local area network) topology type, without a specific hierarchy structure. They are mainly used for sending and receiving data or messages. The messages travel through nodes until it reaches the central node that is usually connected to a pc or server. Well-made mesh networks are self-healing, meaning that they can recover if one node goes offline, without the need for human action. The self-healing ability comes from the algorithm written for it. There are over 70 mesh network protocols and IEEE has set a standard under the title 802.11s. The following chapter will go into detail on how to design a mesh network, topology-, positives-, negatives- of mesh networks, and why mesh network protocol would be suitable for Murata Electronics Oy smart manufacturing case.

### 3.1.1 Mesh Network Topologies

For mesh network topologies there can be partial and full WMNs. In a full mesh network, all the nodes are connected, and the data can travel through each node. In a partial mesh all the nodes are not interlinked.

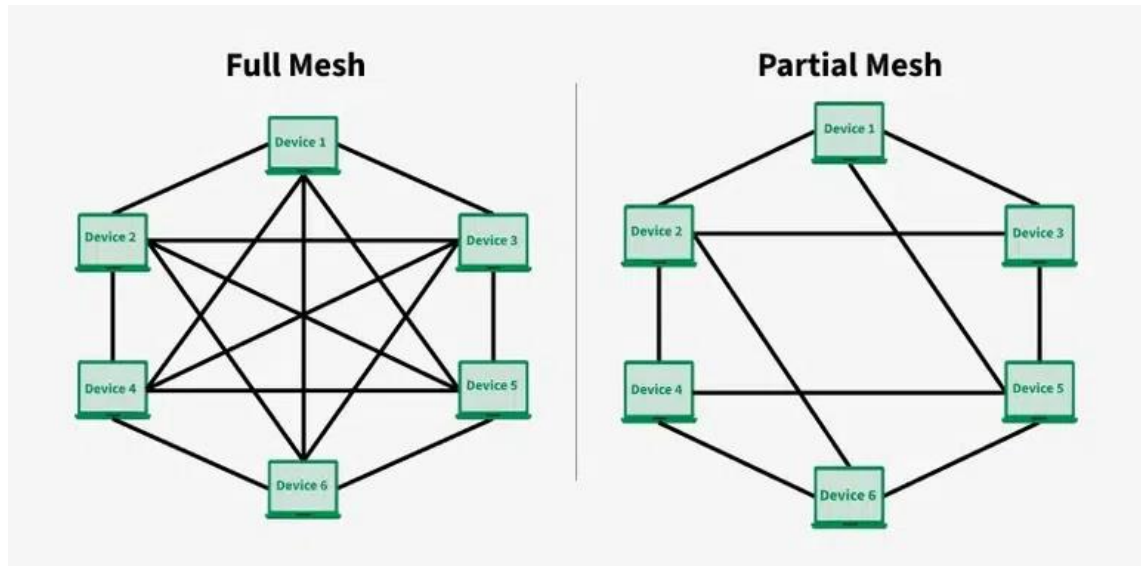


Figure 2. Mesh Network Topology [7]

In a full mesh the self-healing is done painlessly because the data can travel through all the nodes to the central node, as seen in figure 2. In a partial mesh network, if a node goes offline, depending on the location and function of the node, some of the end nodes might get disconnected from the mesh as well. From this information the designer of the mesh should always aim to create a full WMN, but this is not always possible due to out of sight factor and the full range of the mesh. The creator of the mesh can add relay nodes to at least get two pathways for the data to travel for error prevention. The full mesh network might not always be practical because completing a full mesh is expensive [7].

### 3.1.2 Nodes In a Mesh Network

There are three main node types in a simple mesh system: central node which receives the data and connects to an external network, relay nodes which boost the signal range and pass data through from the end node to the central node.

Relay nodes can conduct and send measurements as well, if the correct components are added to it, for example a thermocouple and amplifier for temperature measurement. The third node type, the end nodes, collect data and sends it to the central node. The relay node, also known as an intermediate node, can perform as an end node, also known as a leaf node, and vice versa. This is what usually happens in the self-healing of a WMN when one node goes offline. If the central node goes offline, no data can be received to the exterior network.

### 3.1.3 Different Devices That Can Be Used to Create a Mesh

There are multiple different types of mesh networks. There are Bluetooth mesh networks, wired mesh networks and WI-FI mesh networks. The most common of these are Wi-Fi mesh networks. There is a variety of different devices that can be used for Wi-Fi mesh networks. One of the main devices used for this is a microcontroller that has a Wi-Fi module integrated in it or a separate Wi-Fi module connected. This is used in most projects, because of easy modification and coding to create a specific system required. One example of this type of microcontroller is the ESP32 and ESP8266. These microcontrollers have a Wi-Fi module integrated into the board. Some home Wi-Fi utilizes mesh networks by connecting relays to the gateway to carry the signal for greater distances.

Wired Mesh Networks use the same type of ideology as wireless mesh network, but instead of using Wi-Fi or Bluetooth the mesh communicates through ethernet cables. This is a pretty rare and expensive type of way to create a mesh network and is used when an extremely high level of security and reliability are required.

Smart sensors utilize mesh network technology by collecting the measured data and communicate between different modules or a central node. These systems are called WSNs. This report goes into detail about WI-FI mesh networks and their use in WSN.

### 3.2 How to Create an Efficient Mesh Network

When designing an efficient mesh network, the best solution is aiming to create a full mesh, if the budget and spacing of the nodes allow it. Using reliable hardware, good routing protocols and positioning will support with load balancing to avoid bottleneck nodes. Another key aspect of creating an efficient mesh network is scalability and monitoring tools. The mesh network should have a possibility to painlessly add new nodes to the system, and for the routing protocol to self-organize the new pathways for the data. Having a good monitoring system will help for trouble shooting and handling the data for further visualisation. Depending on the type of data being sent, for example sensitive user data, security measures like WPA2 or WPA3 should be used. WPA is used in Wi-Fi- networks for encrypting the message. Painless mesh has the possibility to implement WPA2 for the encryption protocol. WPA2 is an older encryption protocol than WPA3.

Efficient mesh networks are reliable and utilize error prevention: the data received is accurate and reaches the end user without error. Error prevention can be done for the actual measurement and for the mesh network itself. For the measurement segment the measurements can be taken using rolling average, which basically is the average of the measurements over a certain period. This makes the measurement less susceptible to measurement spikes due to one false reading. The following shows example formulas for rolling average [8].

$$\frac{m_1+m_2+\dots+m_{10}}{10} \quad (1)$$

Equation 1 looks like a normal average formula but with the rolling average, the following equation 2 would be the next subset for rolling average.

$$\frac{m_2+m_3+\dots+m_{11}}{10} \quad (2)$$

The sample size for this project is 10, the average calculation moves along with each new measurement made.

For coding the rolling average there needs to be an array created that holds 10 measurements, and has the oldest measurement removed when a new measurement is added. Then the code just needs to recalculate the average the same way as shown in the equations above.

For the mesh communication side creating a more efficient system, there are a few tools and functions that can be utilized. One option for monitoring the state of the nodes is a MQTT broker. This helps visualise the data paths and helps detect bottleneck nodes and nodes that suffer from interference and messages get lost because of it. An example of a MQTT broker software is HiveMQ. The HiveMQ can create a network map as seen in figure 3.

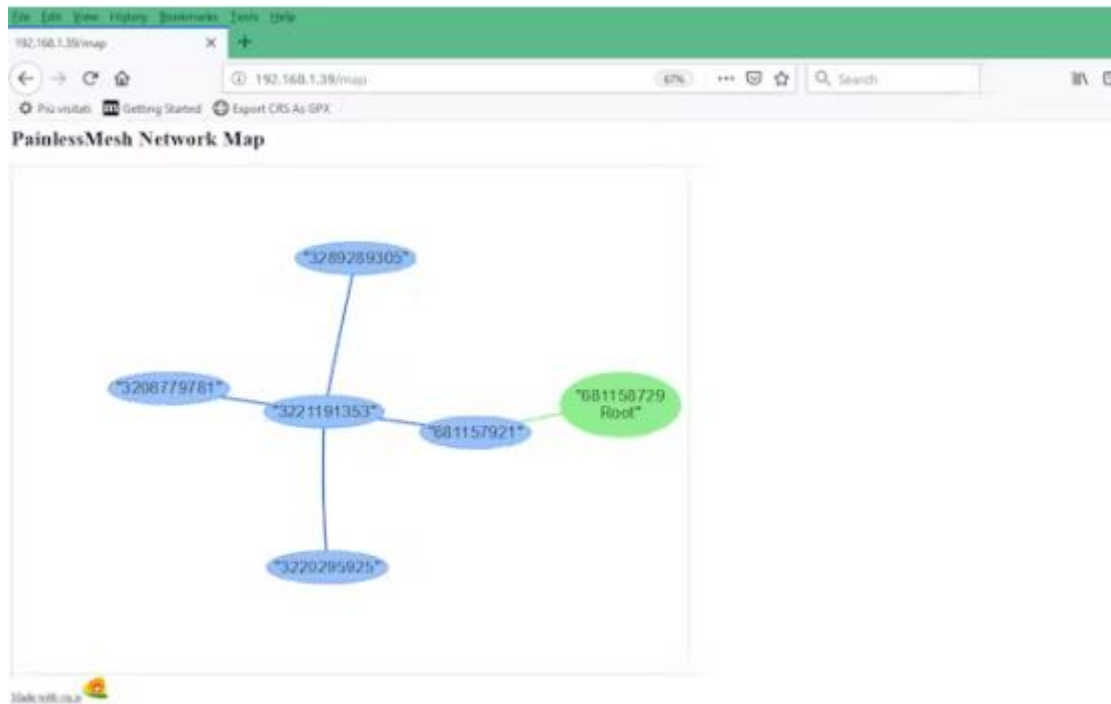


Figure 3. Example Network Map [9]

Network maps help visualize the data paths and see what nodes are connected to the mesh network [9].

Another way is to implement to the code a callback function so the central node can detect if a node has gone offline. This could be done by the central node sending a message to the nodes, which the nodes need to reply to. If a node does not reply, the central node can conclude that the specific node has gone offline and forward that message to the serial monitor. Lastly, an often overlooked part is having good documentation of the system. This is needed for troubleshooting in the future and saves a lot of time when someone else, other than, the creator is trying to fix an issue.

### 3.3 Benefits of Mesh Technology

Basic mesh networks are cheap to create, depending on the components used. A simple Esp32 microcontroller costs from 8–25€. Understanding the mesh libraries is straightforward and implementing them with other prewritten code is simple, if the earlier code is written with scheduler for the timing instead of delays. Mesh networks are good for many projects because there are many different mesh libraries to choose from. The one used and mentioned in this project is `painlessMesh`. This library has plenty of documentation and example code to reference from. The self-healing and self-organization factors are huge benefits for mesh networks, because they require less human action to upkeep the mesh network. Another benefit is easy expansion of the mesh network due to the self-organization all the user must do is write the correct ID, password and port.

### 3.4 Restrictions Using Mesh Network Technology

When creating a more complex mesh system the initial cost might be a bit higher than traditional networks. Trouble shooting without the correct knowledge or tools is very difficult in large scale mesh networks. This is why it is important to document and try to make the mesh as efficient and reliable as possible. Node placement is also crucial for creating a mesh network and in some applications mesh simply is not the best solution, for example if the distances between all nodes are far, and multiple relay nodes are required, utilizing mesh networks technology would not be the best solution for this. The main drawback of mesh networks is security issues. Because the network is broad and has many points of entry, even with improved security features there is a risk if the network is not managed properly. This needs to be taken into consideration when thinking about what type of message is being sent. In Murata's case it is very plain machine data that cannot be utilized by external identities, so the mesh networks security drawback is not as significant. Lastly, one main restriction with using mesh network technology is physical obstructions, like

walls, and electronic interference from other devices. This might lead to decreased performance.

### 3.5 Smart Manufacturing Utilizing Mesh Networks

The way of utilizing WMN in smart manufacturing is by adding to the nodes the ability to measure different metrics. This is done by creating a mesh network with microcontrollers and then adding different types of sensor modules to measure the desired input. The following figure 4 shows an example of a WSN for smart manufacturing in a cement factory.

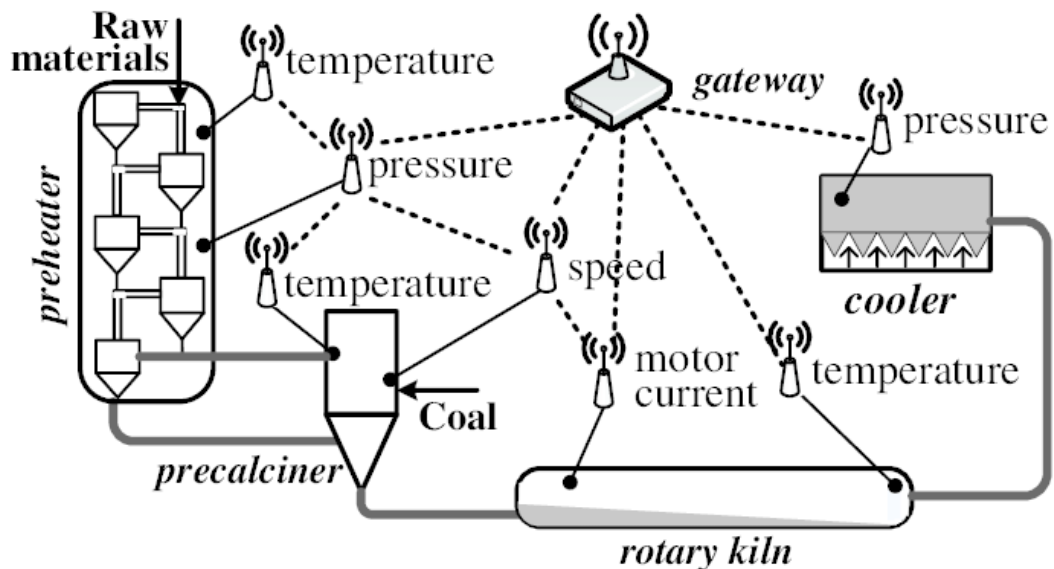


Figure 4. Smart manufacturing in cement factory [10]

WSNs allows for predictive maintenance when the measured value starts to drift. This prevents unplanned downtime and improves the lifetime of the machine. The mesh in this scenario provides the pathway for data to travel from the machines to the central node, and from the central node to the pc or another network, from where the data can be remotely analyzed and the possible actions to be taken. This tool simplifies the process of manually having to measure each of the values and allows a more time efficient way of monitoring industrial machines. The WSN allows for more measurements to be conducted in certain time period as well, because the user can set the measurement

interval to match the priority of the measurement in hand. The measurements can be sent even in a 5 second interval, but over a long period of time this might add too much traffic to the mesh and load the database with non-critical measurement data.

#### **4 Showcase of Working Prototype**

At Murata Electronics Oy there has been one prototype made for temperature measurement WSN using ESP32 microchip which has a Wi-Fi module. This prototype was created by me, Olli Nurmiainen, in my Metropolia Innovation project study module. The purpose of this system was to measure and send temperature data of the exhaust temperature of a machine. The hardware used was a NORVI-IIOT-AE02-I (appendix 2) PLC style board and a NORVI-EX-TEX01 (appendix 2) thermocouple expansion module. The system sends and routes data. Figure 5 shows the final version of the prototype.



Figure 5. Working WSN node

There are some other components added as well. There is a switch that is used for power cycle and in the boot up sequence. There are also cable feedthroughs for the power in and the thermocouple wire. The system has an offset function which can be changed in the boot up sequence. When the power is turned on from the switch the user has 10 seconds to add the offset to the temperature measurement. If no offset is added in the 10 seconds, the system will use the latest offset set by user and automatically continue to the mesh initialization phase. This was added to help the equipment control team, so the system will start automatically for example in a power outage situation. If offset changes are required, after the offset has been set, the system will store the offset data to the EEPROM. The following figure 6 shows the flow chart of the prototype node.

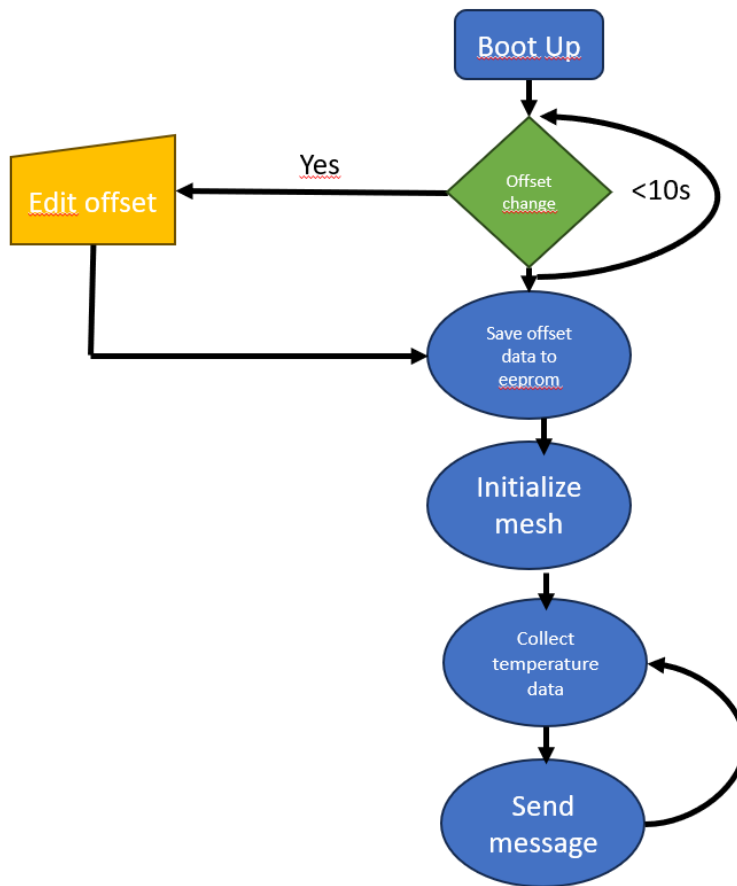


Figure 6. Flow chart of a measurement node

The system sends data to the central node. The central node is connected to an external PC and writes the received messages to the serial prompt. The following listing shows the example data measured by the WSN node in room temperature.

```

08:49:04.160 > Machine1 Temperature 19.25
08:49:06.393 > Machine1 Temperature 19.50
08:49:09.674 > Machine1 Temperature 19.25
  
```

Listing. 1 Example output to the central node

The actual equipment name is redacted and changed to display the equipment as Machine1.

#### 4.1 CE Mark Case Study

The first step for getting the exhaust measurement system CE certified was to create a risk analysis of the system. The risk analysis is used to make the system as safe as possible. There is a point system for the risk analysis, and the system needs to score low enough to be able to say risk analysis is completed. If the system is scoring very high, then the system needs to be analysed to see if it can be made safer. If the system cannot be made safer then, the system needs to have proper warnings and the system user must be properly trained to handle the system and the proper personal safety equipment needs to be specified. This prototype scored very low on the risk analysis scale, declaring the system to be safe, and does not need extra personal safety equipment or warning signs. The risk analysis scoring was done by using the following chart seen in figure 7.

	Little to no effect	Non Critical	Course of action	Fatal
Unlikely	1	4	6	10
Probable under specific circumstances	2	5	8	11
Risk is likely to happen	3	7	9	12

Figure 7. Risk Analysis Chart

The risk analysis was done by defining the use of the system in normal operation. The following table 1 displays different risks that could happen. Each of the risks is scored using the chart seen in Figure 7. After the scores for each of the rows have been assessed, the final score composed is the addition of the all the separate scores.

Table 1. Risk Analysis Scoring Table

Damage	Score	Reasoning for score
Cut or Bruise	1	There are no sharp edges, and the system has a case around it. Unlikely risk for user.
Radiation	1	The system uses standard CE marked components. No risk for user
Electrocution	2	Microcontroller uses standard 24V power supply. Voltage rating not dangerous. Unlikely risk for user.
Crushing risk	1	No possible way of being crushed or pinched, due to no automatically moving parts. No risk.
Burn	1	The system does not contain components that heat up and cause a burn. Unlikely risk of use.
Total score	6	The system scored a total score of 6. No further action is needed.

The system scored an overall six, this is the second safest possible score achievable. If a system scores somewhere around 5–15 the system is good. Anywhere between 15 to the maximum achievable value, which is 72, the system needs to be made safer to get the score under 15. The case study for the exhaust measurement system concludes that the risk analysis has been completed with a score of six. The reason for not getting the minimum score, is because there is not a fuse to protect from power surges. This will be added to the future nodes.

The next step was creating the needed documentations of the system. The system needs technical documents, that include, material list, user manual, service manual, schematics (appendix 1), and risk analysis completion

document. All the required documentation and structures of them can be found in the European Union's EUR-Lex website.

After many meetings and discussions, the conclusion for the CE mark for this prototype was that it is not required. The system has all the required documents made if there is a need to get a CE mark afterwards. The system uses all parts that have a CE mark. The system does not use any moving parts like actuators. This is one key aspect when deciding what machine or system needs a CE mark. Any correctly designed and completed project should have all the above-mentioned documents and all directives considered, so it should not be difficult for a system to get a CE mark if it is required.

## 4.2 Future Project Guidelines

The purpose of setting guidelines for future projects is to make it easy to add nodes to the existing network. There are key aspects to be considered when creating a node to the WSN. These guidelines help to create a more efficient mesh network and to make the WSN more stable and less susceptible to errors.

### 4.2.1 Troubleshooting Guideline

At Murata Electronics Oy there needs to be some trouble shooting options for easily finding issues in the node. This is done by having the nodes write to the serial monitor all the data that has travelled through the node. This is used for two reasons. First reason is that all the nodes can be a central node if a PC with the correct script is connected to it. This makes it so if for some reason the central node gets corrupted or the node is not needed anymore it is easy to change the access point for the PC. All that must be done is to configure from the source code that the relay node will become a central node and upload the code to the new central node. Both configurations exist in the source code and are marked as comments with two forward slashes. The user will make the central node code active and upload it to the microcontroller. The second reason is for trouble shooting an issue in the network. The users can connect a laptop to the node and

open the serial monitor to see what state the node is in and if it is receiving or sending messages.

#### 4.2.2 Naming the Mesh Correctly

Naming the mesh parameters correctly is a key factor for adding the node to the existing mesh network. If the mesh SSID, port, or password are named incorrectly, the node will not connect to the WSN. Without the correct naming of these parameters, the node will try to connect to another network that does not exist, or it will attempt to create its own single node WLAN. The structure for the code is made simpler by having the parameters set as local variables so they are easy to change by the user.

```
#define MESH_SSID "MurataMESH"  
#define MESH_PASSWORD "MESH1234545"  
#define MESH_PORT 4444
```

#### Listing 2. Naming the mesh code

The code above (Listing 2) names the ID, password, and port parameters. The actual parameters are redacted due to confidentiality agreements and named randomly for these lines of code. All the nodes in this network need the same lines of code as in Listing 2 for the ability to communicate with each other. If the password needs to be changed, all the nodes need to be updated by uploading the new code to the microcontroller from a PC. It is not difficult to change but requires time if the mesh network is large.

#### 4.2.3 Sent Data Structure

The message structure of the nodes needs to be organized correctly with the purpose of not overloading the system with non-critical data, and for the data to be analysed correctly by the SQL script. In the listing 1, in which the example data is displayed, the data is split by first having the time, the machine ID, what measurement is done, in this scenario it is temperature, and lastly the actual

measured value. This makes it simpler when running the SQL script to read the data, because it has a clear structure to split the message into the correct variables. When writing the code for the mesh network this was taken into consideration and written the machine ID and measurement, and measurement type to their own local variables (Listing 3). The actual machine ID is redacted and replaced with machine1. The code displayed sends the message that can be seen in Listing 1.

```
void sendMessage(){
    String MachineID = "Machine1 ";
    String MesType = "Temperature ";
    String Tempereaturestring = String(temperature, 2);
    String msg = MachineID + MesType + Tempereaturestring;
```

Listing 3. Message Structure Code

The code is written so that it is easy for future projects to name the correct measurement type and machine, just by changing the values for the variable strings.

#### 4.2.4 Connections Tests

The system must be reliable and work consistently without causing any disturbance and without losing connection randomly. In the source document [11] for `painlessMesh` found in GitHub, the creator stated that the number of nodes is limited, but the amount of memory that can be allocated to the sub-connections is high, so the `painless mesh` library can support a very large mesh network [11]. The reason for using `painless mesh` library was because the code is quite simple when you understand the basics of it, as well as the `painless mesh` sent data structure is JSON based so it human readable and easier to integrate with data visualization.

There needed to be some internal tests at Murata Electronics oy to assure that the mesh network runs without issues and does not interfere with the local WI-FI network at Murata. The test was done with one sensor node, one central

node and two message nodes, because there aren't any other sensor nodes created as of now. The two message nodes were to mimic two sensor nodes. The connection did not disturb or get disturbed by the local Wi-Fi. The connection test ran for 48 hours without any issues, hence that stated that the network was reliable. The following listing shows example data of the mesh connection test that was printed to the serial monitor of the central node.

```
10:57:01.899 > Hello from node 2!  
10:57:02.032 > Machine1 Temperature 19.75  
10:57:04.757 > Hello from node 1!  
10:57:05.732 > Machine1 Temperature 19.75
```

#### Listing 4. Connection Test

The Listing 4. shows that all the data is collected by the central node simultaneously. The only unknown from the connection test is that if the data from two nodes is received at the exact same time that can the central node collect both or does one of the messages get lost. Testing this would be impossible, because sending time varies in microseconds with the ESP32 microcontroller. This is not significant because it is very unlikely that the central node receives two messages at the exact same time even when the mesh network expands, and in the bigger picture one lost message does not matter.

While the connection test was running, the self-healing function was tested as well. This was done by taking one message node and taking it outside of the mesh range. Obviously, this node went offline for the duration it was outside the range. After returning the node to its original position the node connected to the system automatically and the message was seen in the central nodes serial monitor in about 2 minutes after returning the node. When the node is disconnected from the mesh network the node will run mesh initialization until it is connected to the WMN again.

## 5 Mesh Network for Murata

The following chapter is a case study of a possible mesh network for Murata Electronics Oy. It proposes a possible layout for the nodes, data paths, connection monitoring, and the database upload steps. The chapter will also go into detail on data integration software, and data utilization.

### 5.1 Layout For Efficient Mesh Network

One key aspect of creating an efficient mesh network, is to aim to create full mesh instead of a partial mesh. While thinking about creating a mesh network, there should be enough possibilities for measurements nodes. The minimum of measurements nodes recommended is 2, otherwise it would only be a pair and not a mesh network.

The maximum number of nodes using Painless mesh libraries is unknown by the creator of the Painless mesh, but the number of nodes is very high [11]. This needs to be considered. There are different work areas in Murata Electronics clean room, like photolithography and plasma, as well as the production happens in different buildings. The following image (figure 8) is an example of a well created efficient layout for a production area.

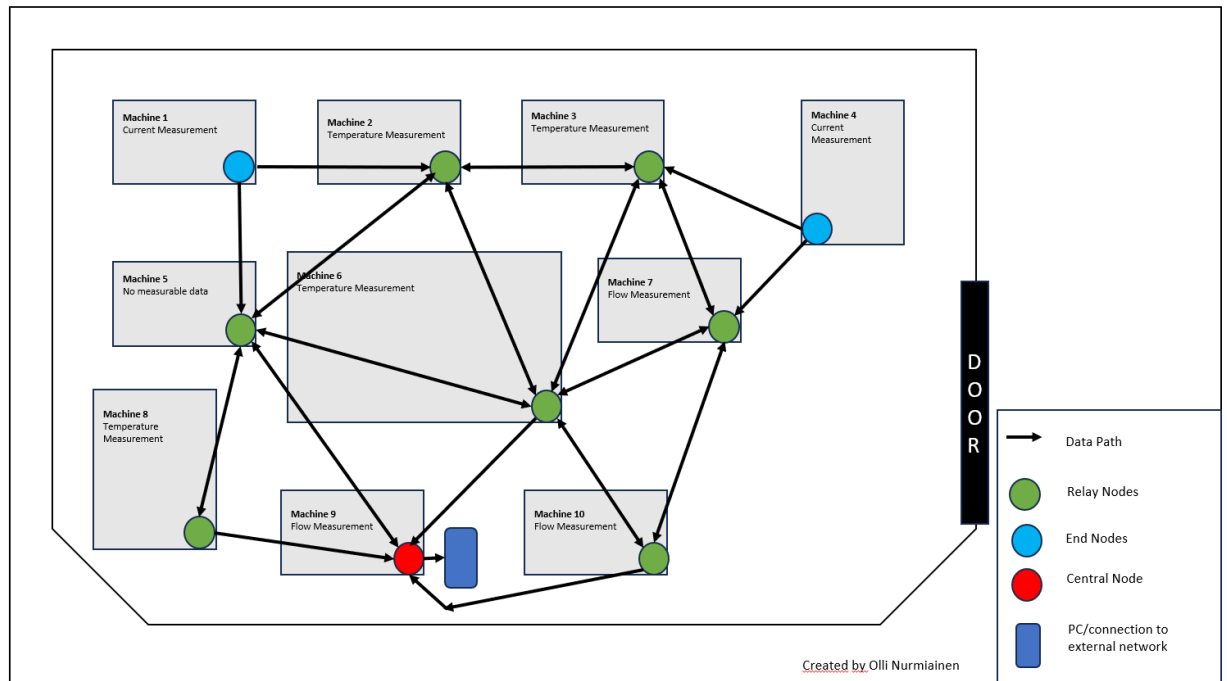


Figure 8. Example Mesh Layout

The image does not correlate to the real lithography area at Murata Electronics Oy and does not show any real machine IDs. The example is created to visualise the end goal of the WSN for Murata Electronics Oy lithography work area. This is almost a full mesh, but all the connections have multiple routes to send the data and no nodes are reliable for another node to be connected. The Central node is located at the bottom of the image because it has the best location for the PC connection.

Creating a full mesh for the whole production between all the buildings would be very difficult and would most likely consist of bottle neck nodes for data transmissions at the bridges between the buildings. One way to avoid this would have separate mesh networks for each building, with its own mesh ID, password, and port. Another issue when creating the mesh between production areas is the walls between the areas. But if the signals are strong enough, the best way is to create the mesh for the whole production area. The central node needs to be located more to the centre of all the sensor nodes, to avoid bottle neck nodes.

## 5.2 Database Upload

The WSN systems central node collects all the measured data from the other sensor nodes and prints to the serial monitor. The central node is connected to a PC via USB and the PC uses a separate software to read the serial monitor. This can be done by using python with pyserial library. The pyserial library can read and format the data to SQL. In the python script there needs to be the database upload code, with the correct database named. The script will automatically send the received serial monitor data to the database in the correct format so the data can be analysed when needed. The python script with for example SQLight library included splits the data into columns and uses simple statements like update for selecting the table that needs to be updated, set which sends the value to the right column. These commands organise the data to be utilized. In the database, there will be one data table that holds all the measurements conducted by the WSN and for each measurement there will be a row consisting of a timestamp, machine ID, measurement type and the measured value.

## 5.3 Data Integration

Data integration is mainly done utilizing JMP which is a visualization tool for SQL databases. In the JMP software the user writes scripts to pull specific data from a machine. The data is split in a specific way during the database upload, so it is simpler for the JMP to examine the data. The data is split into 4 columns, the first column is a time stamp of the measurement, the second column is the machine ID, the third column is the measurement data, and the fourth column is the actual measurement value. The JMP software collects all the data that it can find under these parameters and can create different types of graphs depending on user input. The following chart in figure 9 is created by the same measurement log as in the example data seen in Listing 1. The chart (figure 9) is done in Excel due to licencing issues for the JMP software, but the charts in both tools look very similar.

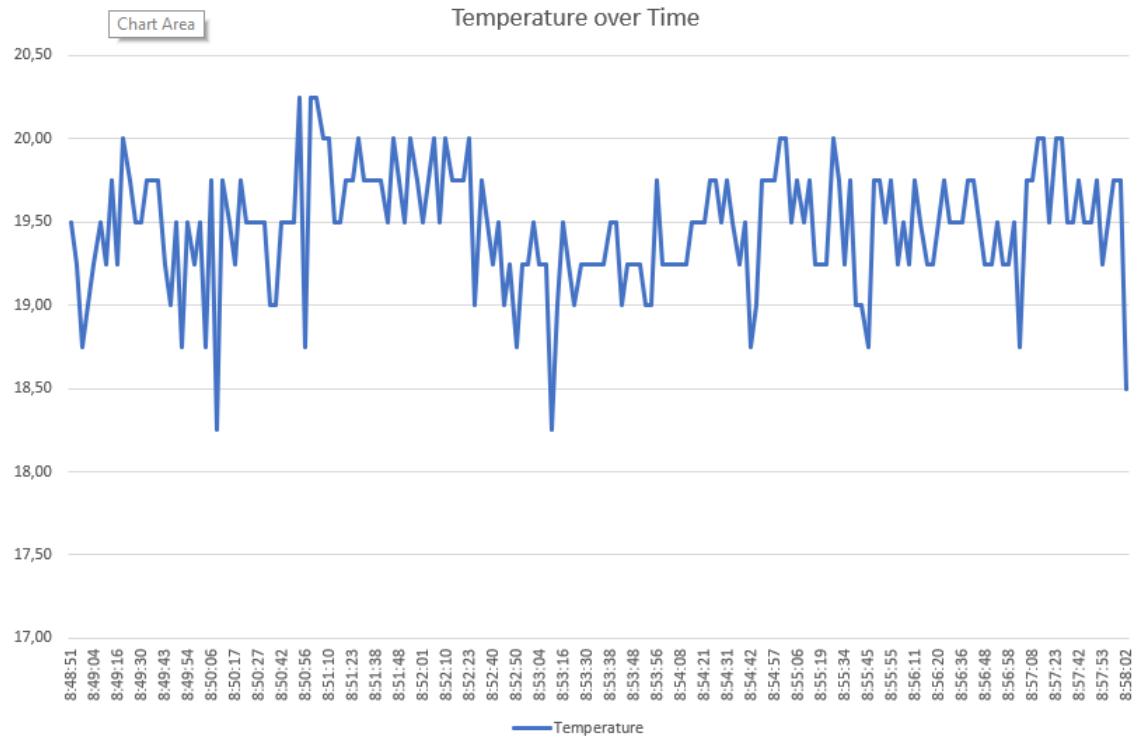


Figure 9. Example Data Chart

The reason for using JMP instead of Excel, for the data visualization is because the JMP software's ability to handle search from the SQL databases better, and JMP has a higher limit for efficient data handling and visualisation.

AI could also be implemented for following trends and informing the users about possible errors. This would result in less unplanned equipment breakdowns, and possibly enhance the equipment lifetime. This would create multiple cost saving possibilities. AI implementation is not yet done but possibly in the near future, because of the rapid innovative improvement in this field of technology. There already is an automatic data criticism software running in the database, but this could be improved with AI technologies.

#### 5.4 Connection Timeout Monitoring

A key aspect of making sure that the connection of the nodes is stable, there needs to be a connection timeout monitor in place for the system. The first step of this is adding a call back function from the central node. If the central node

does not receive a message from a node, a message is printed to the serial monitor. The message will contain the machine ID and the issue. There is a database table that contains machine IDs and corresponding equipment control engineer's email. From this table the python script compares the table to the machine ID and sends the email to the correct person.

Another way of being able to do the comparator is at the central node. This might be simpler to code but will add a substantial amount of stress to one node, because the central node already gathers all the data sent and prints it to the serial prompt. When the mesh network gets bigger, the messages will not all be collected by the central node if it is too strained. If the mesh network will consist only of a few nodes then the better solution is to have the comparator at the central node, but in Murata's case the better solution is to have it at the database integration side, for easier expansion of the network.

## **6 Conclusion**

The purpose of the thesis work was to provide a concept of smart manufacturing utilizing mesh network technologies, and to provide a work guideline for implementing a WSN for any machine. The work presented key points to focus on when deciding if a mesh network is the correct solution and things to consider when designing the mesh network.

There were connection tests ran to assure that the connection was stable, and no interference had been detected, as well the self-healing factor was determined to be functional. There is only one functional temperature measurement node as of now, and new code needs to be written for different types of measurements. The mesh part of the code can be utilized for all the measurement types so the coding part of new nodes is easy and when all the desired measurement types have been completed the real benefit of having the mesh can be used.

The next step is to have working measurement nodes for all desired measurements and when that is done, the nodes can be duplicated, and the mesh will start to scale up for the whole factory. The data integration and utilization are efficient in the future, because of the set guidelines of the sent message structure. Since the mesh network has been planned before implementing random nodes around the factory, the efficiency of the mesh will stay high, and this helps avoid bottleneck nodes and lost messages.

## References

- 1 Robert Faludi, 2009, Building Wireless Mesh Networks, Accessed on 30.08.2024 <https://ab-log.ru/files/File/books/WirelessSensorNetwork.pdf>
- 2 Commission Regulation. Directive - 2014/35 - EN - Low Voltage Directive - EUR-Lex. Accessed September 29,2024. <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32014L0035>
- 3 Commission Regulation. Directive - 2014/30 - EN - EUR-Lex. Accessed September 29,2024. <https://eurlex.europa.eu/legal-content/EN/TXT/?uri=CELEX :32014L0030>
- 4 Commission Regulation. Directive - 2014/53 - EN - EUR-Lex. Accessed September 29, 2024. <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX :32014L0053>
- 5 Commission Regulation. Directive - 2011/65 - EN - RoHS 2 - EUR-Lex. Accessed September 29, 2024. <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32011L0065>
- 6 Commission Regulation. Regulation - EU - 2024/1781 - EN - EUR-Lex Accessed September 29, 2024. <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32024R1781&qid=1719580391746>
- 7 GeeksForGeeks, Advantage and Disadvantage of Mesh Topology, GeeksforGeeks. 2020, Accessed on October 20,2024, <https://www.geeksforgeeks.org/advantage-and-disadvantage-of-mesh-topology/>
- 8 Team W. Moving Average. WallStreetMojo; 2024 Aug 21. Accessed October 21, 2024. <https://www.wallstreetmojo.com/moving-average-formula/>
- 9 DavIdefa, ESP32 Wirelesss Mesh (Made Easy with PainlessMesh), May 16, 2020, Accessed on October 30, 2024, <https://www.hackster.io/davidefa/esp32-wireless-mesh-made-easy-with-painlessmesh-part-3-982af1>
- 10 Jin X, Xia C, Xu H, Zeng P. Mixed Criticality Scheduling for Industrial Wireless Sensor Networks. MDPI; 2016. Accessed October 22, 2024. [https://www.researchgate.net/publication/307442446\\_Mixed\\_Criticality\\_Scheduling\\_for\\_Industrial\\_Wireless\\_Sensor\\_Networks](https://www.researchgate.net/publication/307442446_Mixed_Criticality_Scheduling_for_Industrial_Wireless_Sensor_Networks)

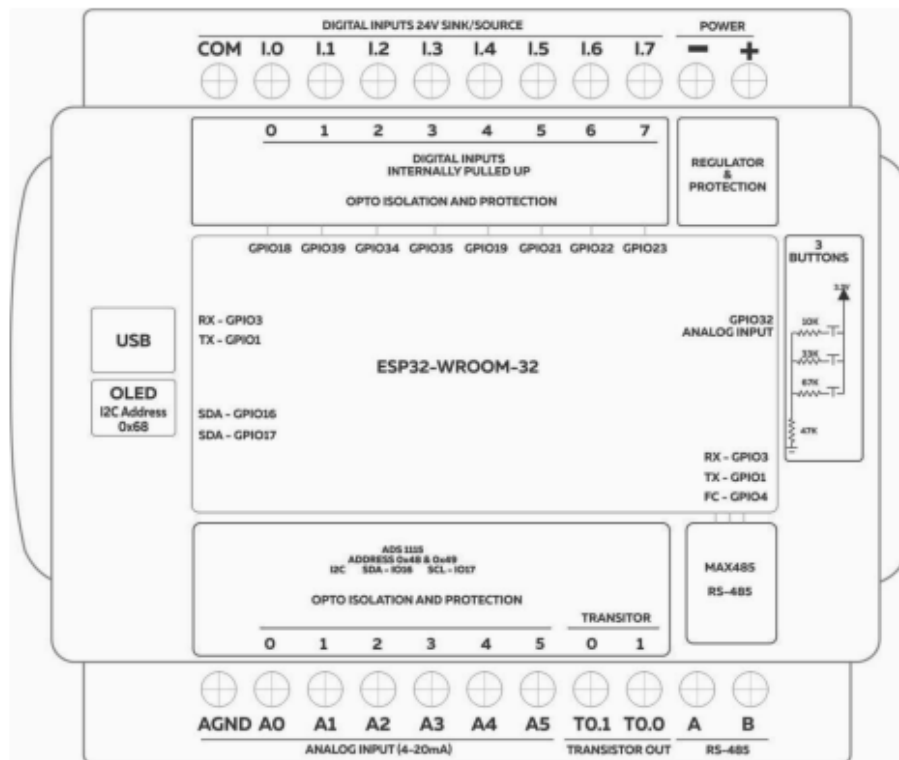
- 11 painlessMesh. GitLab; 2024. Accessed October 23, 2024: <https://gitlab.com/painlessMesh/painlessMesh>
- 12 NORVI Industrial Arduino. NORVI IIOT-AE02-I – Datasheet; 2023. Accessed October 24, 2024. <https://norvi.lk/docs/norvi-iiot-ae02-i-datasheet/>
- 13 NORVI Industrial Arduino. NORVI EX-TEX01 – DATASHEET; 2023. Accessed October 24, 2024. <https://norvi.lk/docs/norvi-ex-tex01-datasheet/>



## NORVI Datasheets [12;13]

# NORVI IIOT-AE02-I – Datasheet

## Product Features #



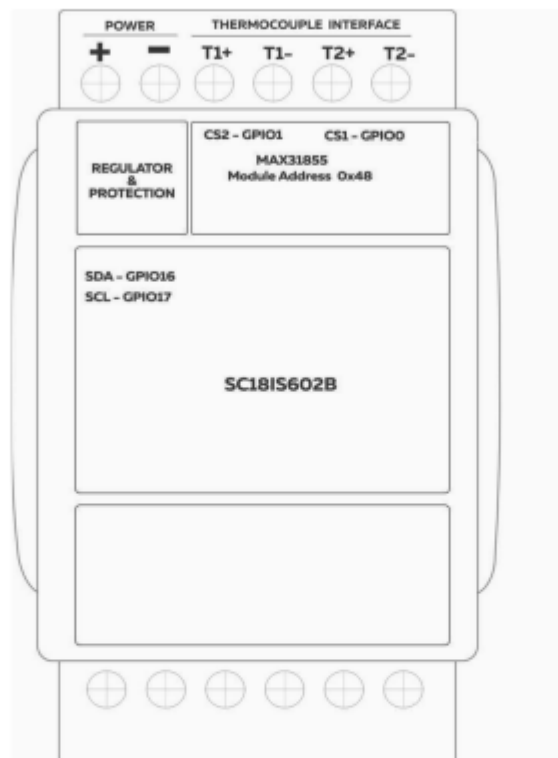
- ESP32-WROOM32 Module
- Built-in 0.96 OLED Display
- Built-in Button on the front panel
- Digital Inputs
- Analog Input
- Transistor Outputs
- DIN-Rail mounts
- Supports Expansion ports

### Expansions Supported

- Digital Input
- Analog Input
- Transistor Output

# NORVI EX-TEX01 – DATASHEET

## Product Features #



- Versions Available for Most Common Thermocouple Types: K-, J-, N-, T-, S-, R-, and E-Type
- Detects thermocouple shorts to GND or VCC.
- Detects an open thermocouple.
- Straight temperature readouts with a 0.25°C resolution
- I2C Interface with controller

### Expansions Supported

- Analog Input
- Analog Output