MINISTRY OF EDUCATION AND SCIENCE OF THE REPUBLIC OF KAZAKHSTAN
Non-profit join - stock corporation
ALMATY UNIVERSITY OF POWER ENGINEERING AND TELECOMMUNICATION named after G. Daukeev
Department of *Electronics and robotics*

«Allowed to defence»
The head of department of «Electronics and robotics»
Chigambayev T.O. c.t.s., associate professor
(Full name, academic degree, rank)

_____«_____» _____2020 year
(sign)

DEGREE PROJECT

On the topic: *Establishment and research of IoT Mesh network*

Done by: *Nazarbay Nursultan* *PSa-16-3*
(Surname and initials of a student) (group)

Specialty *5B071600 Instrumentation Engineering*

Research supervisor *Baykenov B.S. professor*
(Surname, academic degree, rank)

_____«_____» _____2020 year
(sign)

Consultants
on economic part: *Tuzelbayev B.I. Ph.D, associate professor*
(Surname, academic degree, rank)

_____«_____» _____2020 year
(sign)

on life and environmental safety part: *Begimbetova A.S. Ph.D, senior lecturer*
(Surname, academic degree, rank)

_____«_____» _____2020 year
(sign)

Compliance supervisor: *Fazylova A.R.senior lecturer*
(Surname, academic degree, rank)

_____«_____» _____2020 year
(sign)

Almaty, 2020

MINISTRY OF EDUCATION AND SCIENCE OF THE REPUBLIC OF KAZAKHSTAN
Non-profit join - stock corporation
ALMATY UNIVERSITY OF POWER ENGINEERING AND TELECOMMUNICATION named after G. Daukeev

Institute of *space engineering and telecommunications*
Department of *Electronics and robotics*
Specialty *5B071600 Instrumentation Engineering*

ASSIGNMENT
for execution of degree project

Student *Nazarbay Nursultan Daniyarulu*
(Full name)

Topic of the work *Establishment and research of IoT Mesh network*

Approved by the order of the rector № 155 from « 23 » october 2020 y.

Deadline of the finished work « 9 » june 2020 y.

Initial data required parameters of the results and initial data:

1. *Argon board*
2. *Xenon board*
3. *AM2302 temperature sensor*
4. *laptop*

List of issues to be developed in a degree project or a summary:

1. *Technical overview of the project and boards*
2. *The establishment of mesh network*
3. *The research of mesh network*
4. *Development of life safety measures*
5. *Economic justification of the project*

List of graphical material (with precise indication of mandatory drawings);
This degree project contains 48 figures and tables

Recommended basic literature:

1. *J. Okin, The Internet Revolution: The Not-for-Dummies Guide to the History, Technology, and Use of the Internet, 1 ed., Winter Harbor, ME: Ironbound Press, 2005.*
2. *Internet of Things (IoT) Lab, Department of Engineering and Architecture, University of Parma, Parco Area delle Scienze 181/A, 43124 Parma, Italy;*

3. *X. W. Ian Akyildiz, Wireless Mesh Networks, London: John Wiley & Sons, 2009*

Consultants for work with indication of the relevant section

Section	Consultant	Date	Sign
Life safety	*Begimbetova A.S*	*05.05.2020*	
Economic part	*Tuzelbayev B.I*	*03.05.2020*	

SCHEDULE
Of degree project preparation

№	Title of section, list of issues to be developed	Deadline for submission to instructor	Note
1	*Technical part*	*02.02*	
2	*Establishment part*	*02.04*	
3	*Research part*	*02.05*	
4	*Life safety*	*05.05*	
5	*Economic part*	*03.05*	
6	*Conclusion*	*06.06*	

Date of issue of the assignment ___*« 20 » 01 2020 year*

The head of department _____*Chigambayev T.O.*_____
(sign) (Surname and initials)

Supervisor: _____*Baykenov B.S. professor*_____
(sign) (Surname and initials)

The assignment for execution is accepted by: _____*Nazarbay N.D.*_____
(sign) (Surname and initials)

Аңдатпа

Бұл жобаның мақсаты Particle-дың IoT құрылғылары арқылы mesh желісін құру, сосын осы құрылғыларға және mesh желідесіне зерттеу жасау болып табылады. Mesh желісі Argon және Xenon құрылғыларының арасында құрылды. Зерттеулерде mesh желісінің қосылу диапазоны, оның деректерді беру жылдамдығы және құрылғылардың энергия шығыны өлшенді.

Тіршілік қауіпсіздігі бөлімінде қауіпсіз жерлеу және жасанды жарықтандыру есептелген. Дипломдық жобаның экономикалық тиімділігі де есептелді.

Аннотация

Целью данного проекта является создание mesh сети с IoT устройствами от Particle и сделать некоторые исследования на этих устройствах и на mesh сети. Mesh сети создаются между устройствами Argon и Xenon. В исследованиях измеряется дальность соединения mesh сети, скорость передачи данных и энергопотребление устройств.

В разделе безопасность жизнедеятельности, рассчитано безопасное заземление и искусственное освещение. Экономическая эффективность дипломного проекта также рассчитывается.

Annotation

The aim of this project is to establish a mesh network with IoT devices from Particle and make some research on those devices and on a mesh network. The mesh network is created between Argon and Xenon devices. And in the research, the connection range of mesh network, its speed of data transfer and the energy consumption of devices are measured.

In the section of life safety, safety grounding and artificial lighting are calculated. The economic efficiency of the diploma project is also calculated

Content

Introduction

Due to the rapid growth of the semiconductor industry, creating small devices with powerful computing and networking capabilities is no longer a dream for engineers. Today, the Internet of Things (IoT) has become one of the hottest topics in both the industry and the field of wireless communications. Today, most research on IoT-enabled devices focuses mainly on creating data acquisition and processing units, or new sensors. However, networks, where IoT devices are integrated into the Internet, are generally unaffected by simply using existing computer networking solutions such as WLAN and Bluetooth. These computer networks are not designed for low power devices such as remote sensors. These IoT devices are considered minicomputers. Due to the single point of failure nature of this network, sensors may need to be placed in hard-to-reach locations, making the entire system very vulnerable, even in a disaster or difficult environment. In addition, the capacity of the network's central hub/router can also limit the coverage of services provided by IoT devices, and the range is limited by the same factors. Most of these remote IoT devices are small, and the devices usually operate on battery power, so energy-intensive network options such as cellular networks and satellite usage are also ideal for most remote IoT network scenarios.

A mesh network (MN) is communication in which a network consists of radio nodes that are organized in a mesh topology instead of the star topology used in most networks. This is not a new concept, which appeared in the 1970s from several special Packet Radio NETwork (PRNET) networks created by the US Department of Defense Advanced Research Agency (DARPA). Later, in the 1990s, many other civilian solutions were proposed and created for various purposes, such as expanding the reach of broadband services. The distributed network nature of a mesh network with its simple configuration is ideal for implementation on IoT networks to take advantage of the extended range and minimize hardware design with small network modules. Such networks are more robust even in harsh conditions, as the network is distributed without a single central point of failure.

The goal of my project is to build a mesh network with Particle IoT devices and do some research on this mesh network. In the study, I will measure the area of the mesh connection and the speed of its data transfer. I will also measure the power consumption of IoT devices.

1 Technical part

1.1 IoT mesh network

A mesh network (mesh topology) is a way to connect infrastructure nodes. These nodes connect to a variety of other nodes and communicate with each other to transmit data. Each device has at least two ways to send and receive information. For this reason, the entire network does not depend on a single node.

The traditional star typology is designed so that all devices are directly connected to the gateway. This makes the entire structure highly dependent and prone to connection errors. When a node stops working, you cannot send or receive data.

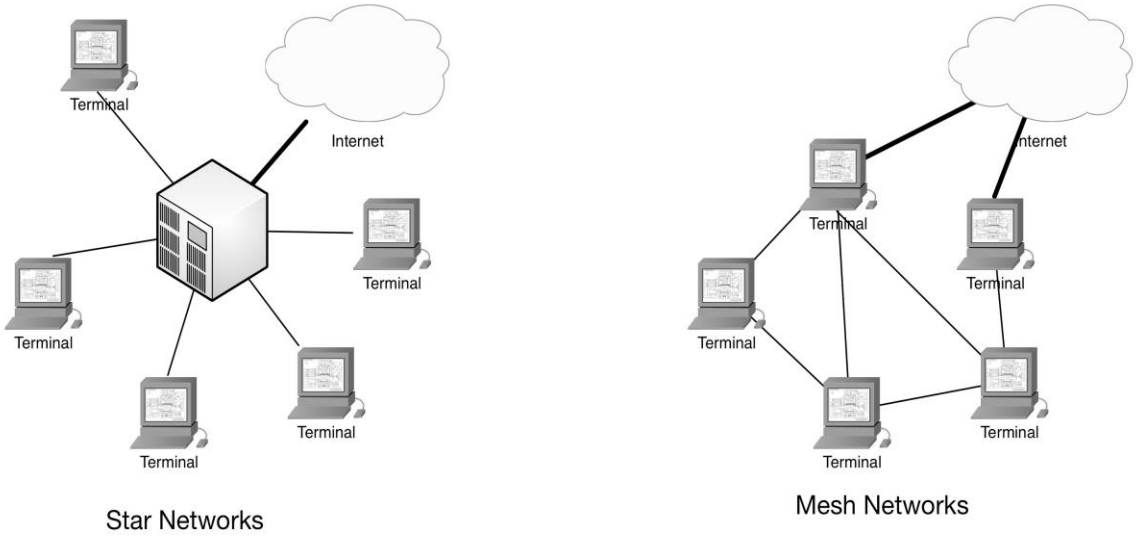

Figure 1.1 – Star topology vs mesh topology

MNs are wireless networks that can do this dynamically self-organizing and self-adjusting, and with grid connect automatically between nodes on the network, but on a regular star network, It has a star topology, which means that all the end nodes connected to one central point associated with this top-level network. Figure 1.1 shows the topology of the two networks.

MNs are currently adapting at various points, mainly in three different options as follows:

1) Infrastructure/trunk MN: As shown in Figure 1.2, this includes the MN type network routers that form the infrastructure for clients which connect devices to them on a network router supply areas still form a star network Mesh routers form a self-configuring and self-healing mesh communication with each other. With a gate, functionality can be connected to a network router of the Internet. Infrastructure/trunk MN are the most MN is often used as simple and easy to use. Integration into existing devices such as routers only It must be installed on wire mesh.

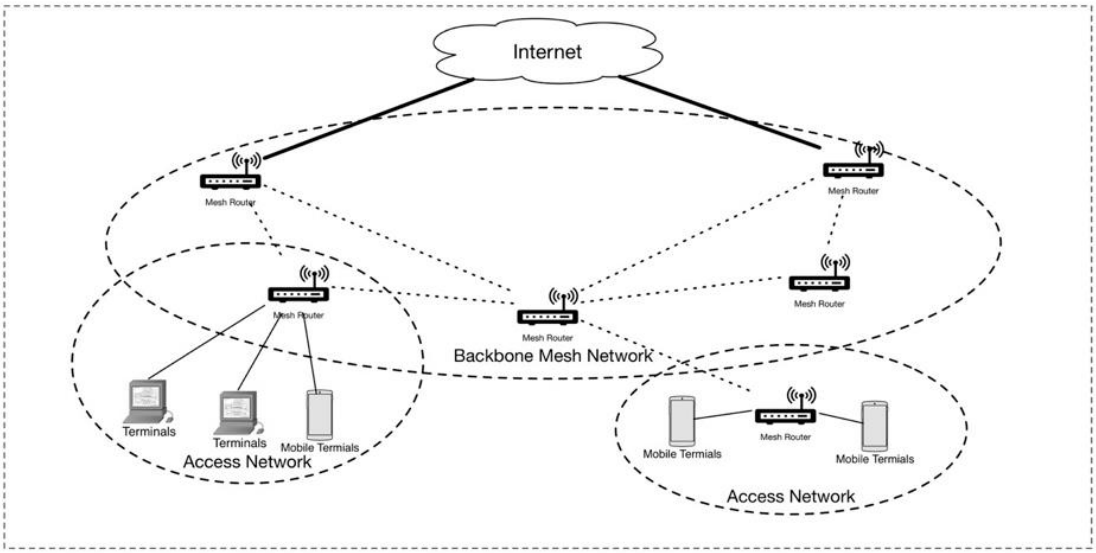

Figure 1.2 – Infrastructure/trunk MN

2) MN client: Client grid offers peer-to-peer networks between client devices in the form of a large special network. With this type of architecture, client nodes form a real network for routing and configuration functions as well as provide the end-user applications for customers. Therefore, a mesh router does not require this type of network. This kind of network is usually unavailable to the Internet.

3) Hybrid MN: As in Figure 1.3, hybrid wire mesh Architecture is a combination of infrastructure and Client network, as shown in the following figure. Mesh fabric Clients can access the network through network routers as well as directly with grids with other grid clients. While the infrastructure provides connectivity for other Networks such as Internet routing features customers offer improved connectivity and coverage within MN. Only MN infrastructure and hybrid MN suitable for the network as per requirement different scenarios so discussion in the background This document focuses mainly on these two types of MN.

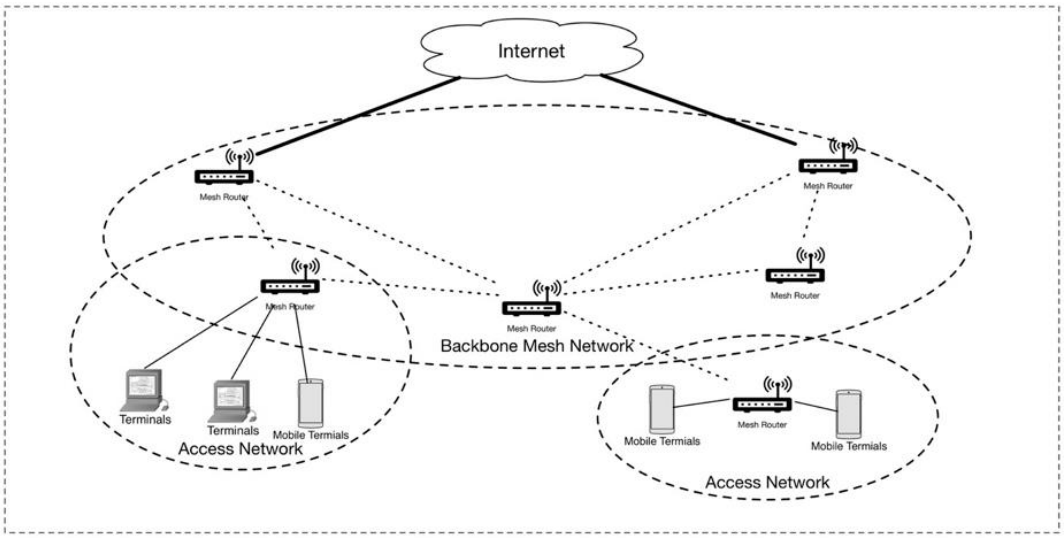

Figure 1.3 – Hybrid MN

There are two to four different components, depending on the type of standard used in a particular wire mesh:

• Node – Each mesh network has a node, which is a device that communicates with each other. This is how they send and receive information.

• Repeaters – For Wi-Fi networks, repeaters are often devices that support Internet signals.

•End point – The mesh network may work so that some devices do not send information to other devices. It just sends the information to the other nodes. That's why it's called an endpoint. However, their implementation is becoming less common because all nodes can send and receive packets without an endpoint on the network.

• Gateway – There is also a gateway that communicates with the Internet as well as the nodes. Therefore, the information package will eventually be sent to the database. Figure 1.4.

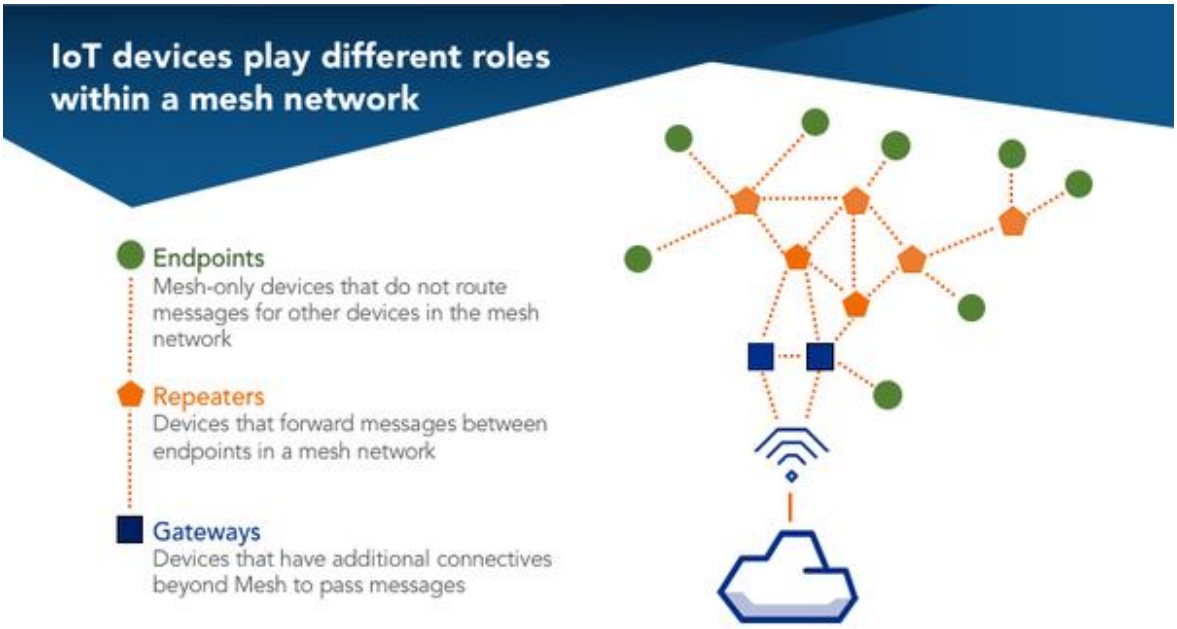

Figure 1.4 – IoT devices in mesh network

The mesh network of IoT devices can send messages in two ways: flooding or routing. Flooding is a method in which all nodes act as senders of information. As a result, the time between sending and receiving information is very short. However, this method is very energy-consuming.

On the other hand, this step selects the path from a node to node and sends the packet to the endpoint. Compared to flooding, routing sends messages to hosts. Another route is chosen only if no connection is possible. For maximum efficiency, the network uses the shortest path bridging (SPB or IEEE 802.1aq algorithms). This allows you to transfer information along the shortest available path.

There is a lot of advantages of IoT mesh network. And that makes this type of network very popular among other networks. Depending on mesh topology it has a variety of cons.

Firstly, it has functionality which heals it by himself. It is called self-healing. Its algorithm finds the shortest way to transfer data and make a connection between them. Shortest Path Bridging is the name of that algorithm. Also, this bridging chooses only working and available nodes, so, the data transfer will be reliable and safe. If one of the devices is broken or not working the network still works besides that.

Secondly, self-configuring functionality. It is a good way to establish a mesh network. When you want to create a mesh network and make devices to connect to it you don't need to configure all of the devices, because the mesh devices are automatically configured while connecting.

Scalability and reliability. The great thing about mesh networks is that they can be scaled very easily. Nodes can be added or removed without any performance issues.

The more devices you have, the more problems you encounter. However, the opposite is true for wire mesh.

More nodes, more routes. The mesh network of IoT devices sends information about the fastest route. Adding more nodes gives you more options for moving packages. This makes the network faster and less prone to errors.

Price cut. Mesh networks use almost no electricity because they do not require an internet connection. The sensor is also very cheap and can be used for many years.

Mesh networks of IoT devices are cheap in two ways. First, it uses almost no electricity because it doesn't require an internet connection. At the same time, the sensor is also very cheap and the battery can be used for many years.

On the other hand, implementing smart solutions can save you money in other ways. Better management, optimized use of resources, etc. Later in this project, you will find some use cases that show how different projects can help reduce costs.

Although mesh networks have many advantages, they also have certain disadvantages. Therefore, it is important to be informed before deciding if a mesh network is suitable.

Maintain. Because mesh networks are self-healing, it can take a while to find a broken site. The network is designed to function properly even if not all nodes are accessible. This means that you do not know when the node encountered the problem.

At the same time, mesh networks of IoT devices are designed to make systems smarter and more efficient. This usually also means getting information about the performance of each node individually.

Waiting time. Moving from one node to another can slow down the reception of data. If your system needs the package every few minutes, that's fine. However,

this may not be enough for your system. In a full mesh network, connecting each node can speed up the process.

Low capacity. Mesh networks are most effective when used to send small packets of information. Unfortunately, the size of the data video file does not work well. When you need to send large amounts of information, using a WiFi mesh network is more appropriate.

1.2 Argon board characteristics

The Argon is a powerful Wi-Fi enabled development board for Wi-Fi networks. It is based on the Nordic nRF52840 and has built-in battery charging circuitry so it's easy to connect a Li-Po and deploy your local network in minutes. The Argon is great for connecting existing projects to the Particle Device Cloud over a Wi-Fi network.

The USB port is the easiest way to power up the Argon. Please make sure that the USB port is able to provide at least 500mA. Power from the USB is regulated down to 3.3V by the on board Torex XC9258A step-down regulator.

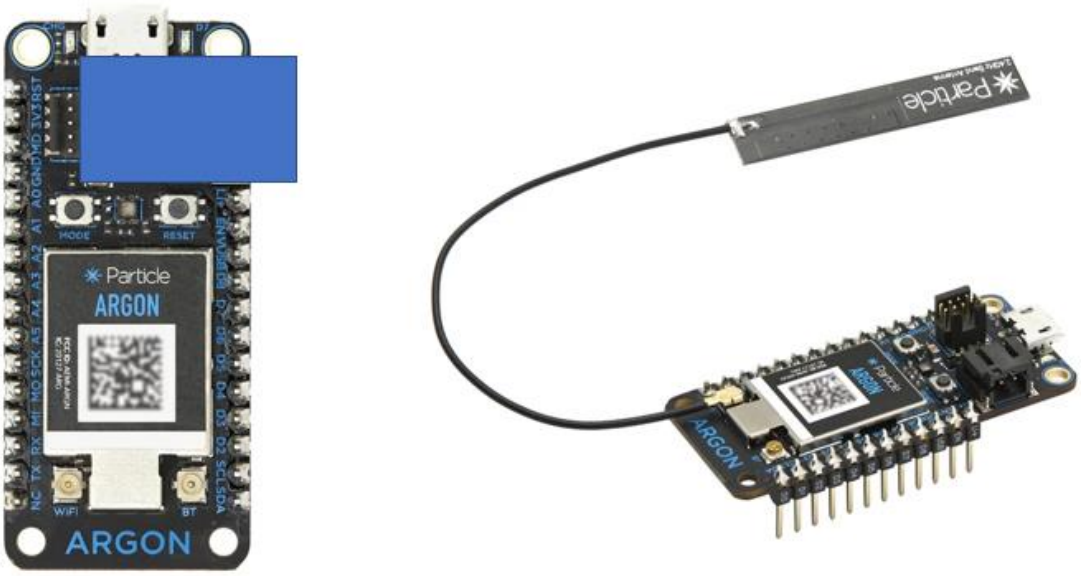

Figure 1.5 – Argon board

As you can see from Figure 1.5 Argon board is like Arduino Nano. Also, it has an antenna on the wi-fi pin on the front side. This antenna works as a wi-fi connector. When you are buying an Argon board, you can specify if you want an antenna with the board. In my case, I needed a wi-fi connection to my mesh network, as a result, I have bought an antenna with the board. And in process, it showed his advantage and was useful when connecting to wi-fi.

The pin is internally connected to the VBUS of the USB port. The nominal output should be around 4.5 to 5 VDC when the device is plugged into the USB port

and 0 when not connected to a USB source. You can use this pin to power peripherals that operate at such voltages. Do not exceed the current rating of the USB port, which is nominally rated to 500mA. This pin is also protected with an internal fuse rated at 1000mA.

Moreover, on the front side of the board, there is a small white square, and on it, there is a data matrix sticker. This sticker is unique for each board and with its help, you can access the Particle Device Cloud. To do that you can scan the code on your phone camera while you are in Particle app special for phones.

If you want to make your projects truly wireless, you can power the device with a single cell LiPo (3.7V). The Argon has an on board LiPo charger that will charge and power the device when USB source is plugged in, or power the device from the LiPo alone in the absence of the USB.

Argon board features:

- Espressif ESP32-D0WD 2.4 GHz Wi-Fi coprocessor
    - On-board 4MB flash for ESP32
    - 802.11 b/g/n support
    - 802.11 n (2.4 GHz), up to 150 Mbps
- Nordic Semiconductor nRF52840 SoC
    - ARM Cortex-M4F 32-bit processor @ 64MHz
    - 1MB flash, 256KB RAM
    - Bluetooth 5: 2 Mbps, 1 Mbps, 500 Kbps, 125 Kbps
    - Supports DSP instructions, HW accelerated Floating Point Unit (FPU) calculations
    - ARM TrustZone CryptoCell-310 Cryptographic and security module
    - Up to +8 dBm TX power (down to -20 dBm in 4 dB steps)
    - NFC-A tag
- On-board additional 4MB SPI flash
- 20 mixed signal GPIO (6 x Analog, 8 x PWM), UART, I2C, SPI
- Micro USB 2.0 full speed (12 Mbps)
- Integrated Li-Po charging and battery connector
- JTAG (SWD) Connector
- RGB status LED
- Reset and Mode buttons
- On-board PCB antenna
- U.FL connector for external antenna
- Meets the Adafruit Feather specification in dimensions and pinout
- FCC, CE and IC certified
- RoHS compliant (lead-free)

Argon has two radio modules: Nordic Semiconductor nRF52840 SoC for BLE and NFC and Espressif Systems ESP32 for WiFi

Feature	Description
Operating Frequencies	2360 to 2500 MHz
Output Power	Programmable -20dBm to +8dBm
PLL channel spacing	1 MHz
On the air data rate	125 to 2000 kbps

Figure 1.6 – Nordic Semiconductor nRF52840 SoC for BLE and NFC features

Feature	Description
WLAN Standards	IEEE 802.11b/g/n
Antenna Port	Single Antenna
Frequency Band	2412 to 2484 MHz

Figure 1.7 – Espressif Systems ESP32 for WiFi features

Argon radio modules is powerful and useful for establishing mesh network. As shown in the Figure 1.6 and 1.7 Argon radio modules have a lot of features that is unique for them.

There are two radios on the Argon. A BLE radio (nRF52840) and a WiFi radio (ESP32). For the WiFi radio, we have provide a u.FL connector to plug in the WiFi antenna. This is required if you wish to use the WiFi connectivity.

There are two options for the BLE antenna on the Argon. It comes with an on-board PCB antenna which is selected by default in the device OS and a u.FL connector if you wish to connect an external antenna. If you wish to use the external antenna, you'll need to issue an appropriate command in the firmware.

It is also possible to use most antennas designed for Wi-Fi (2.4 GHz) as a BLE antenna. In some cases, a u.FL to RP-SMA adapter will be required. If you are building a product using alternative antennas, additional certification may be required.

Stresses beyond those listed under absolute maximum ratings may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under recommended operating conditions is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

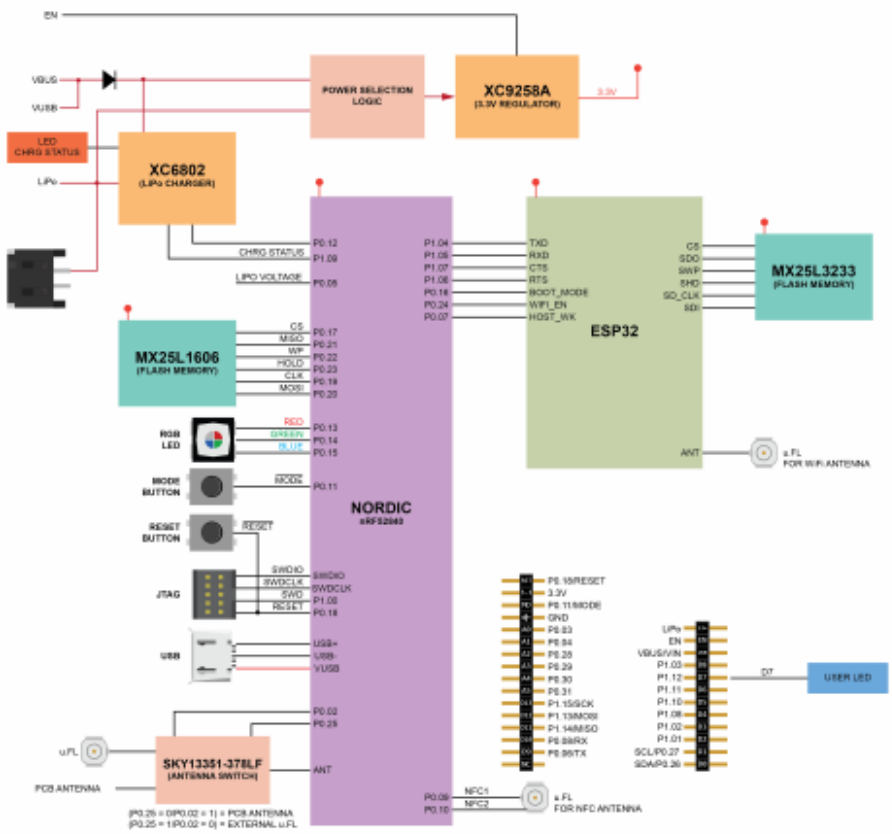

Figure 1.8 – Argon board block diagram

Argon has a lot of input and output pins. You can use all of them as you need. Each pin has its own type of function, which is located in separate part of the device. However, to work with them properly, you have to program it in the Particle Workbench. The reason why I have chosen this board is that this board is powerful when establishing mesh network and doesn't require a hard settings and setup. This device works as a gateway in a mesh network.

The Argon comes preprogrammed with a bootloader and a user application called Tinker. This application works with an iOS and Android app also named Tinker that allows you to very easily toggle digital pins, take analog and digital readings and drive variable PWM outputs.

The bootloader allows you to easily update the user application via several different methods, USB, OTA, Serial Y-Modem, and also internally via the Factory Reset procedure. All of these methods have multiple tools associated with them as well.

1.3 Xenon board characteristics

The Xenon is a low cost mesh-enabled development board that can act as either an endpoint or repeater within a Particle Mesh network.

The Xenon is mesh only and designed to function as the endpoint of your IoT network. It is based on the Nordic nRF52840 and has built-in battery charging circuitry so it's easy to connect a Li-Po and deploy your local network in minutes.

The Xenon is best for connecting sensors, motors, pumps, valves, and points of data-interest. Pair it with an Argon or Boron gateway to get all that great data into the Device Cloud.

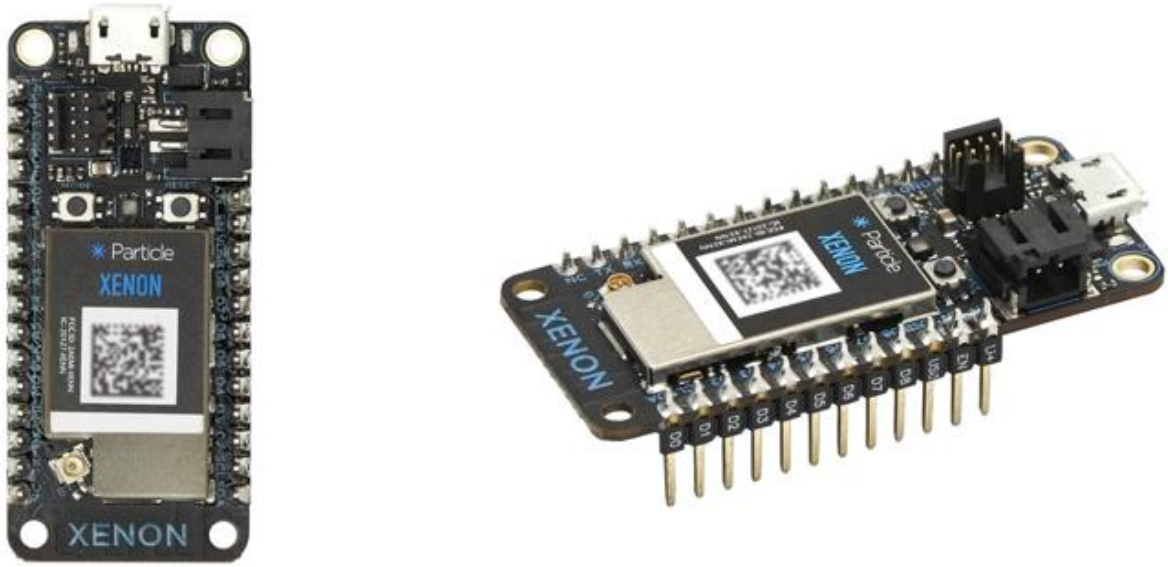

Figure 1.9 – Xenon board

The Xenon comes preprogrammed with a bootloader and a user application called Tinker. This application works with an iOS and Android app also named Tinker that allows you to very easily toggle digital pins, take analog and digital readings and drive variable PWM outputs.

The bootloader allows you to easily update the user application via several different methods, USB, OTA, Serial Y-Modem, and also internally via the Factory Reset procedure. All of these methods have multiple tools associated with them as well.

The USB port is the easiest way to power up the Xenon. Please make sure that the USB port is able to provided at least 500mA. Power from the USB is regulated down to 3.3V by the on board Torex XC9258A step-down regulator.

The pin is internally connected to the VBUS of the USB port. The typical output should be around 4.5 to 5 VDC when the device is plugged into the USB port and 0 when not connected to a USB source. You can use this pin to power peripherals that operate at such voltages. Do not exceed the current rating of the USB port, which is nominally rated to 500mA. This pin is also protected with an internal fuse rated at 1000mA.

Xenon board features:

- Nordic Semiconductor nRF52840 SoC
    - ARM Cortex-M4F 32-bit processor @ 64MHz
    - 1MB flash, 256KB RAM
    - Bluetooth 5: 2 Mbps, 1 Mbps, 500 Kbps, 125 Kbps
    - Supports DSP instructions, HW accelerated Floating Point Unit (FPU) calculations
    - ARM TrustZone CryptoCell-310 Cryptographic and security module
    - Up to +8 dBm TX power (down to -20 dBm in 4 dB steps)
    - NFC-A tag
- On-board additional 4MB SPI flash
- 20 mixed signal GPIO (6 x Analog, 8 x PWM), UART, I2C, SPI
- Micro USB 2.0 full speed (12 Mbps)
- Integrated Li-Po charging and battery connector
- JTAG (SWD) Connector
- RGB status LED
- Reset and Mode buttons
- On-board PCB antenna
- U.FL connector for external antenna
- Meets the Adafruit Feather specification in dimensions and pinout
- FCC, CE and IC certified
- RoHS compliant (lead-free)

If you want to make your projects truly wireless, you can power the device with a single cell LiPo/Lithium Ion (3.7V). The Xenon has an on board LiPo charger that will charge and power the device when USB source is plugged in, or power the device from the LiPo alone in the absence of the USB.

There are two options for the Mesh antenna on the Xenon. It comes with an on-board PCB antenna which is selected by default in the device OS and a u.FL connector if you wish to connect an external antenna. If you wish to use the external antenna, you'll need to issue an appropriate command in the firmware.

The U.FL antenna connector is not designed to be constantly plugged and unplugged. The antenna pin is static sensitive and you can destroy the radio with improper handling. A tiny dab of glue (epoxy, rubber cement, liquid tape or hot glue) on the connector can be used securely hold the plug in place.

The 10 pin SWD connector provides an easy in-system debugging access to the device. The pins on the connector can easily be damaged if the mating connector cable is inserted improperly. If you are trying to debug the device, you probably are not in a good mood to begin with. The last thing you want is to render the connector useless. Be nice, and be gentle on the connector. Good luck with the debugging!

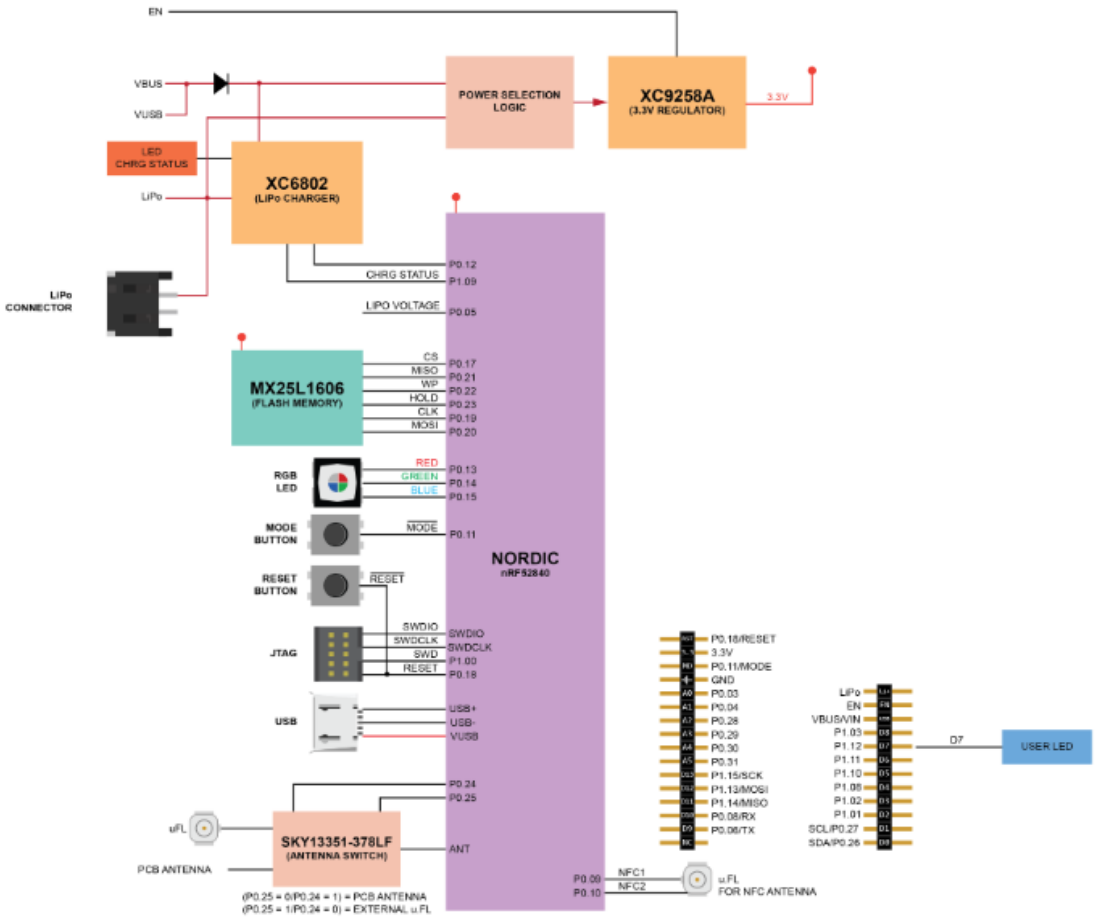

Figure 1.10 – Xenon board block diagram

The Xenon contains highly sensitive electronic circuitry and is an Electrostatic Sensitive Device (ESD). Handling Xenon without proper ESD protection may destroy or damage it permanently. Proper ESD handling and packaging procedures must be applied throughout the processing, handling and operation of any application that incorporates Xenon. ESD precautions should be implemented on the application board where the Xenon is mounted. Failure to observe these precautions can result in severe damage to the Xenon.

Xenon uses the Nordic Semiconductor nRF52840 SoC as the main controller and the mesh radio.

Feature	Description
Operating Frequencies	2360 to 2500 MHz
Output Power	Programmable -20dBm to +8dBm
PLL channel spacing	1 MHz
On the air data rate	125 to 2000 kbps

Figure 1.11 – Nordic Semiconductor nRF52840 SoC features

The U.FL antenna connector is not designed to be constantly plugged and unplugged. The antenna pin is static sensitive and you can destroy the radio with improper handling. A tiny dab of glue (epoxy, rubber cement, liquid tape or hot glue) on the connector can be used securely hold the plug in place.

The 10 pin SWD connector provides an easy in-system debugging access to the device. The pins on the connector can easily be damaged if the mating connector cable is inserted improperly. If you are trying to debug the device, you probably are not in a good mood to begin with. The last thing you want is to render the connector useless. Be nice, and be gentle on the connector. Good luck with the debugging.

The Xenon comes preprogrammed with a bootloader and a user application called Tinker. This application works with an iOS and Android app also named Tinker that allows you to very easily toggle digital pins, take analog and digital readings and drive variable PWM outputs.

The bootloader allows you to easily update the user application via several different methods, USB, OTA, Serial Y-Modem, and also internally via the Factory Reset procedure. All of these methods have multiple tools associated with them as well.

1.4 Temperature sensor AM2302(DHT22)

The AM2302 transmits a calibrated digital signal. It adopts original digital signal collection technology and humidity detection technology to guarantee reliability and stability. The sensing element is connected to an 8-bit single-chip computer. All the sensors of this model are temperature compensated, calibrated in a precise calibration chamber and the calibration factors are stored in the OTP memory in the program type. When the sensor detects it, it recovers the memory coefficient. Due to its small size, low power consumption and long-distance transmission (100 m), the AM2302 is suitable for all kinds of difficult applications. Single row package with 4 rows, very convenient to connect.

You can use the DHT22 (or AM2302) Humidity/Temperature Sensor and Arduino UNO board to read the data and print it on a serial monitor or display it on an LCD screen. For me, I chose the serial monitor version because it’s fast and cheap. If you use LCD screens, it is recommended to check other articles that contain xenon.

When the sensor detects it, it recovers the memory coefficient. Due to its small size, low power consumption and long-distance transmission (100 m), the AM2302 is suitable for all kinds of difficult applications. Single row package with 4 rows, very convenient to connect.

We chose DHT22 over DHT11 because of its wide measurement range. Humidity is 0-100%, temperature is -40°C to +125°C. It also has a digital output (single bus) that provides higher data accuracy. We also use a DC fan that starts to

spin when the humidity level reaches 60% or the temperature exceeds 40°C, but these values can be modified in the sketch.

DHT11		DHT22
0 - 50°C / ± 2°C	*Temperature Range*	-40 - 125 °C / ± 0.5 °C
20 - 80% / ± 5%	*Humidity Range*	0 - 100 % / ± 2-5%
1Hz (one reading every second)	*Sampling Rate*	0.5 Hz (one reading every two seconds)
15.5mm x 12mm x 5.5mm	*Body Size*	15.1mm x 25mm x 7.7mm
3 - 5V	*Operating Voltage*	3 - 5V
2.5mA	*Max Current During Measuring*	2.5mA

Figure 1.11 – DHT 22 vs DHT11 temperature sensors

You can use the DHT22 (or AM2302) Humidity/Temperature Sensor and Arduino UNO board to read the data and print it on a serial monitor or display it on an LCD screen. For me, I chose the serial monitor version because it's fast and cheap. If you use LCD screens, it is recommended to check other articles that contain xenon.

When the sensor detects it, it recovers the memory coefficient. Due to its small size, low power consumption and long-distance transmission (100 m), the AM2302 is suitable for all kinds of difficult applications. Single row package with 4 rows, very convenient to connect.

Table 1.1 AM2302 sensor technical specification:

Model	AM2302
Power supply	3.3-5.5V DC
Output signal	digital signal via 1-wire bus
Sensing element	Polymer humidity capacitor
Operating range	humidity 0-100%RH; temperature -40~80Celsius
Accuracy	humidity +-2%RH(Max +-5%RH); temperature +-0.5Celsius
Resolution or sensitivity	humidity 0.1%RH; temperature 0.1Celsius
Repeatability	humidity +-1%RH; temperature +-0.2Celsius
Humidity hysteresis	+-0.3%RH
Long-term Stability	+-0.5%RH/year

Interchangeability	fully interchangeable

You can use the DHT22 (or AM2302) Humidity/Temperature Sensor and Arduino UNO board to read the data and print it on a serial monitor or display it on an LCD screen. For me, I chose the serial monitor version because it's fast and cheap. If you use LCD screens, it is recommended to check other articles that contain xenon.

We chose DHT22 over DHT11 because of its wide measurement range. Humidity is 0-100%, temperature is -40°C to +125°C. It also has a digital output (single bus) that provides higher data accuracy. We also use a DC fan that starts to spin when the humidity level reaches 60% or the temperature exceeds 40°C, but these values can be modified in the sketch.

2 Establishment of IoT mesh network

2.1 Overall description and Particle website

The goal of the project is to establish a mesh network and make some research on that network. To achieve this in this part of the project it was explained how to establish a mesh network. The main thing to create a mesh network is IoT devices that support mesh technology.

In this diploma project, we have used devices from a company called Particle. This company makes a lot of IoT products and sales on the internet. And Particle offers a variety of development kits including Argon and Xenon. Also, Particle has its Device Cloud where all devices from them are connected to that Cloud storage automatically.

However, to establish a real mesh network we need a lot of devices in this network. And it costs a lot of money. So, I decided to create this network between two devices. For research purposes, it won't require a lot of devices to be connected. From the connection range of one node, we can calculate the overall occupation area of the mesh network, and the energy consumption of one device is almost enough to calculate overall consumption. You can see my functional scheme of nodes in Figure 2.1.

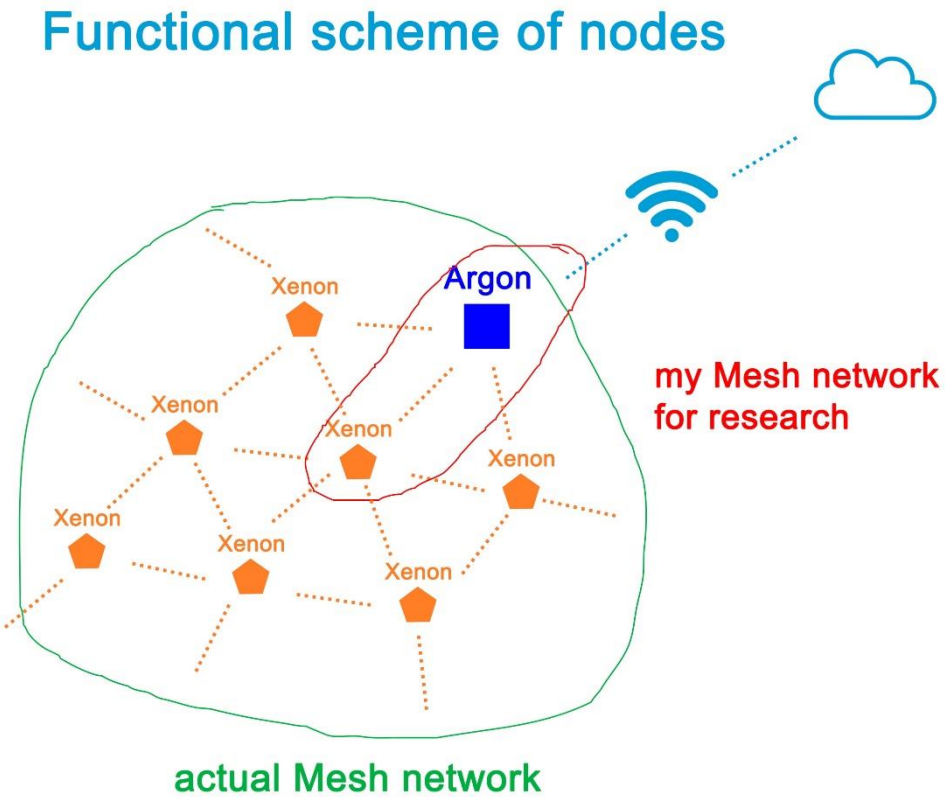

Figure 2.1 – Functional scheme of nodes

To establish mesh network Argon and Xenon were used. Firstly, Argon is been set up with a phone app from Particle and its data matrix code. After, we

connected Argon to a local wi-fi to access the internet and Particle Device Cloud. And also in the phone app, on the Argon we have created a mesh network called “Nursultan” and gave a password to this mesh network, to make it a secure network.

Then I have set up the Xenon board as an Argon. However, Xenon can’t connect to wi-fi without a special antenna. So, to access the internet on Xenon we connected it to a mesh network created by Argon on the mobile app. After this, all of two devices have been accessible in the Particle Console. In the console, you can implement Tinker and control the device via wi-fi and mesh network.

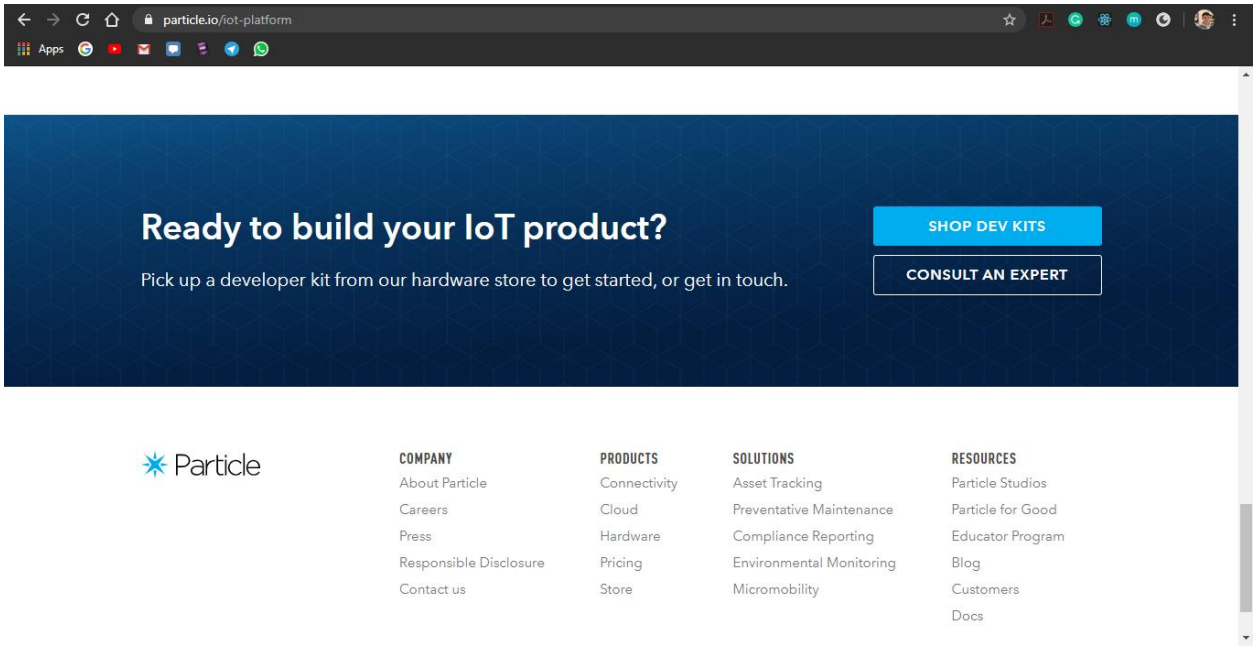

Figure 2.2 – Particle website

Whether you’re scaling a revolutionary micromobility fleet, driving efficiencies by tracking the location and status of business-critical assets, or monitoring protected lands for illegal forestry, the promise of IoT is compelling: bring connectivity, intelligence, and control to unconnected physical objects.

While the promise of IoT is alluring, the road to IoT nirvana is plagued with challenges ranging from security risks to complexities spanning hardware, connectivity, and software.

That’s where Particle comes in. Particle is the edge-to-cloud IoT platform to help you connect the unconnected. With powerful hardware solutions that thrive at the edge, reliable and efficient connectivity technologies, and software to manage your IoT deployment, Particle empowers you to unlock the promise of IoT.

In this diploma project, we have used devices from a company called Particle. This company makes a lot of IoT products and sales on the internet. And Particle offers a variety of development kits including Argon and Xenon.

And to test the network is working we used temperature sensor and connected it to Xenon. This sensor will sense the temperature and sends temperature value to Xenon device as a digital signal. After Xenon sends this value to Particle Device

Cloud via a mesh network to Argon and from Argon via wi-fi to Cloud storage. Then, in the Console of the device, the value appears.

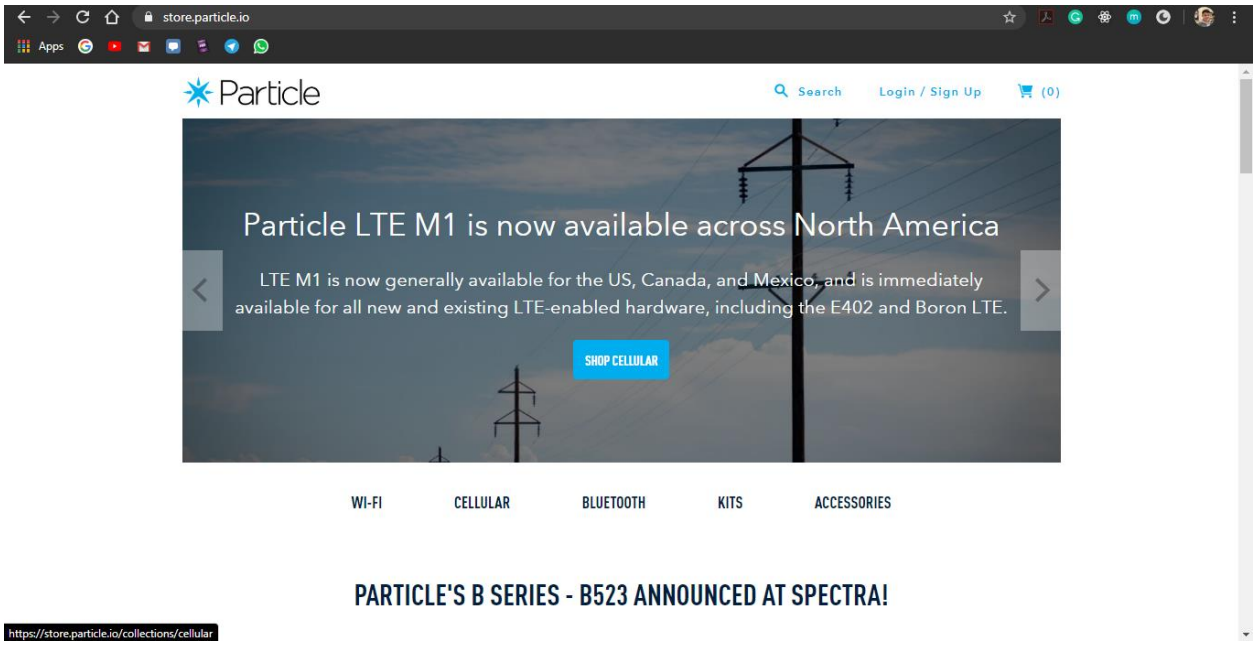

Figure 2.3 – Particle store

The Particle Console is your centralized IoT command center. It provides interfaces to make interacting with and managing Particle devices easy. This guide is divided into two main sections, tools for developers and tools to manage product fleets.

The Console does not yet work in Microsoft Internet Explorer including Edge. Please use another browser, such as Chrome or Firefox, to access the Console.

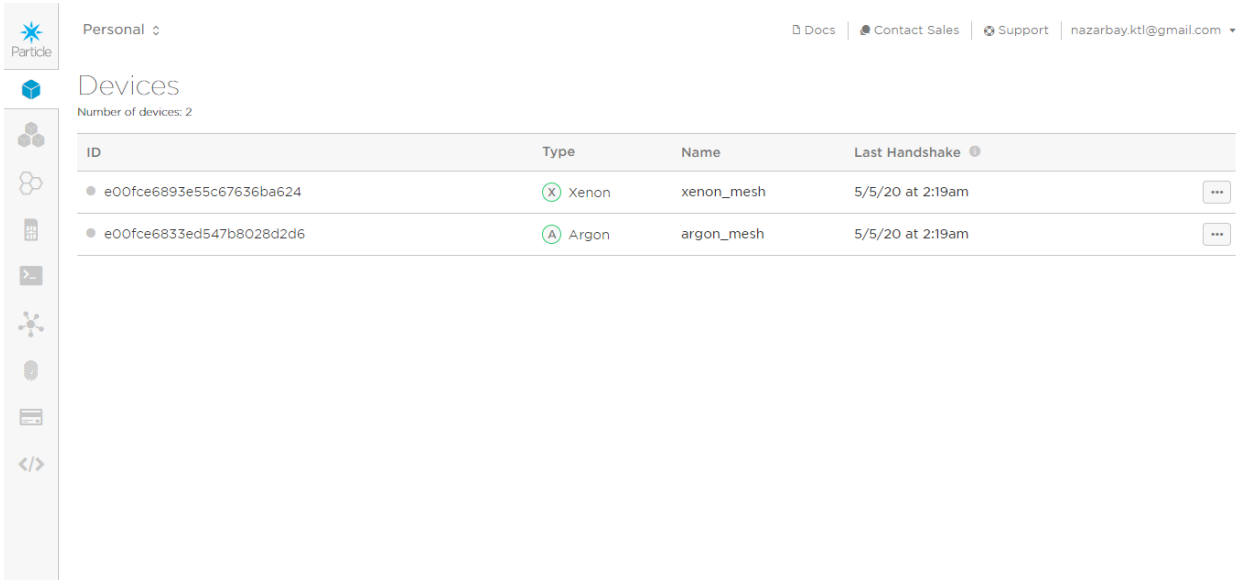

Figure 2.4 – Particle Device Console

The Devices page allows you to see a list of the devices associated with your account. Here, you can see specific information about each device, including it's

unique Device ID, it's name, the type of device (i.e. Argon or Xenon) the last time it connected to the Particle Device Cloud, and whether or not the device is currently online.

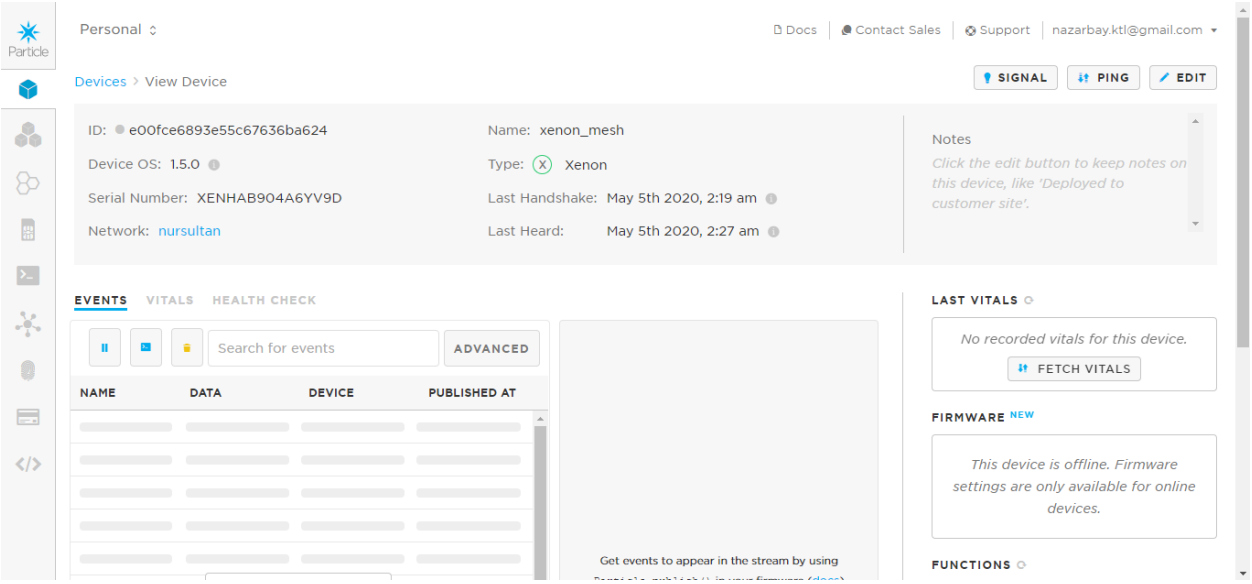

Figure 2.5 – Xenon board on Particle Console

You can also take certain actions on devices from this view, such as renaming the device and unclaiming it from your account.

Unclaiming a cellular device removes it from your account, but does not stop billing. As the claiming status and SIM are separate, you must also pause or release ownership of your SIM to stop billing.

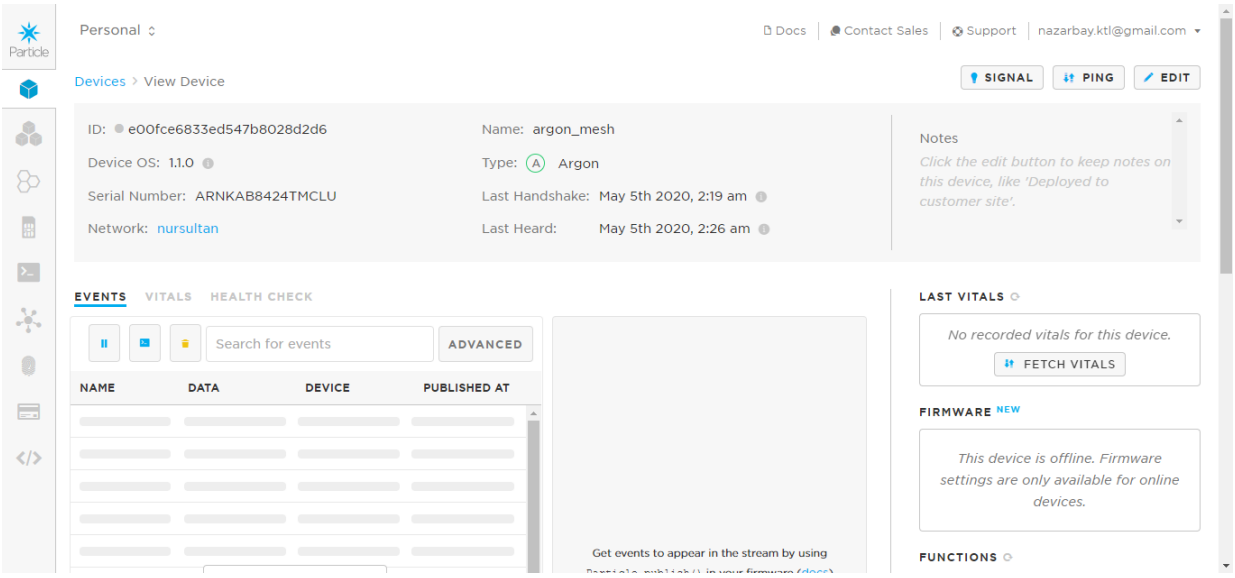

Figure 2.6 – Argon board on Particle Console

While actively developing an IoT project or product, the Console offers many helpful features to make prototyping a breeze. See the last time a device connected, debug a firmware issue by observing event logs, set up a webhook to send data to an external service, and more.

The Logs feature provides a clean interface to view event information in real-time, just from your devices. We're hoping that this is handy both while debugging code during development, and checking out recent activity on your device once you power-on your finished project. Tailored around improving the experience of browsing logs, the page provides a handful of tools, like filtering, modifiers which enable you to narrow down your search for events, making it easier to view only the data that is relevant to you. In this view, you'll only see events that come in while the browser window is open.

2.2 Setting up Argon and Xenon boards

2.2.1 Argon setup. Connecting to wi-fi. Establishing a mesh network

To setup Argon I used mobile app Particle. In the app, as you can see from figure 2.7 on the top right side there is + button to add devices to Particle Console and Cloud. If you click this button, the names of available devices appear. Argon is on the first column of this list and then comes instruction like this "Pair your device with your phone. Find the data matrix sticker on the front of your device. Use your phone camera to scan the code.". When you click "Scan data matrix" button, the camera is opened automatically and scans the data matrix of the device.

After, it asks you to attach wi-fi antenna to Argon device and then you can start to set it up. It will automatically pair with the phone by Bluetooth. Then it asks "Do you want to use this Argon in a mesh network?" and "This Argon can act as a gateway to the Internet and create a local wireless mesh network that other devices can join.". In the bottom side, there are two buttons "Use in mesh network" and "Don't use in mesh network" (Figure 2.8).

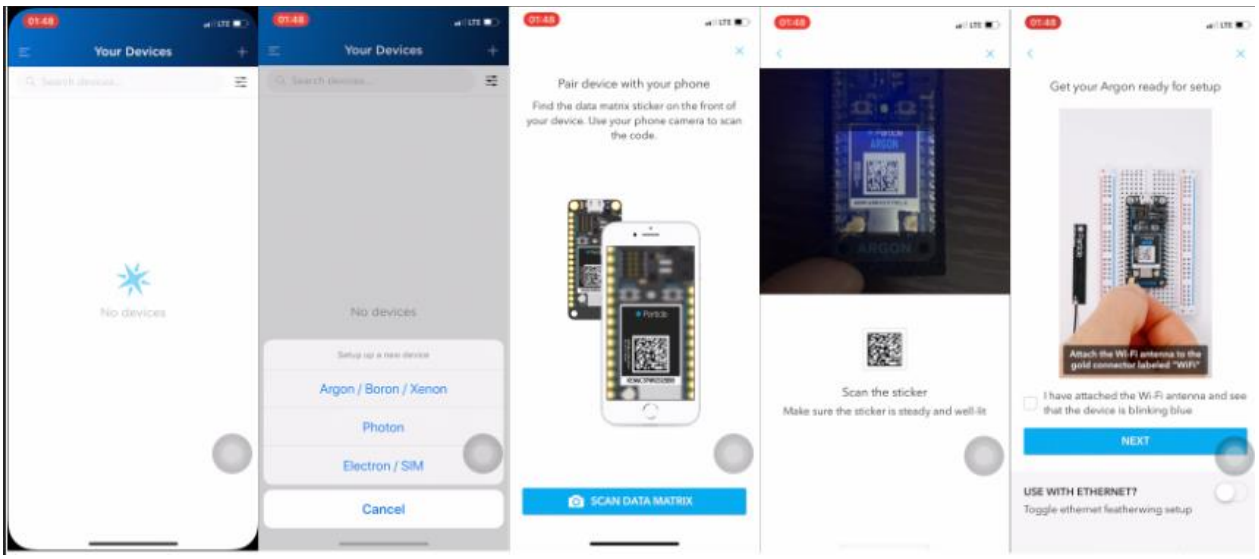

Figure 2.7 – Argon setup beginning

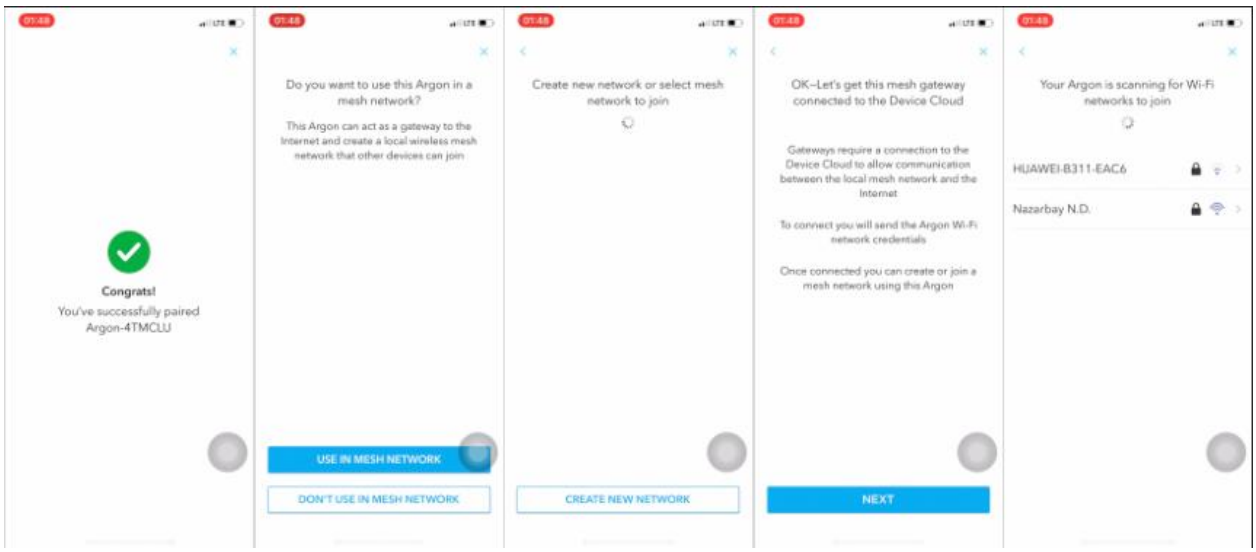

Figure 2.8 – Connecting Argon to wi-fi and establishing the mesh network

To establish a mesh network you have to click the first button. After that, you also must click the button “Create new network”. To create a new network it is required to connect Argon to wi-fi with a message “OK – Let’s get this mesh gateway connected to the Device Cloud.” And “Gateways require a connection to the Device Cloud to allow communication between the local mesh network and the Internet. To connect you will send the Argon Wi-Fi network credentials. Once connected you can create or join a mesh network using this Argon.”. After you should connect to wi-fi and wait one or two minutes before Argon is connected to Particle Device Cloud (Figure 2.9).

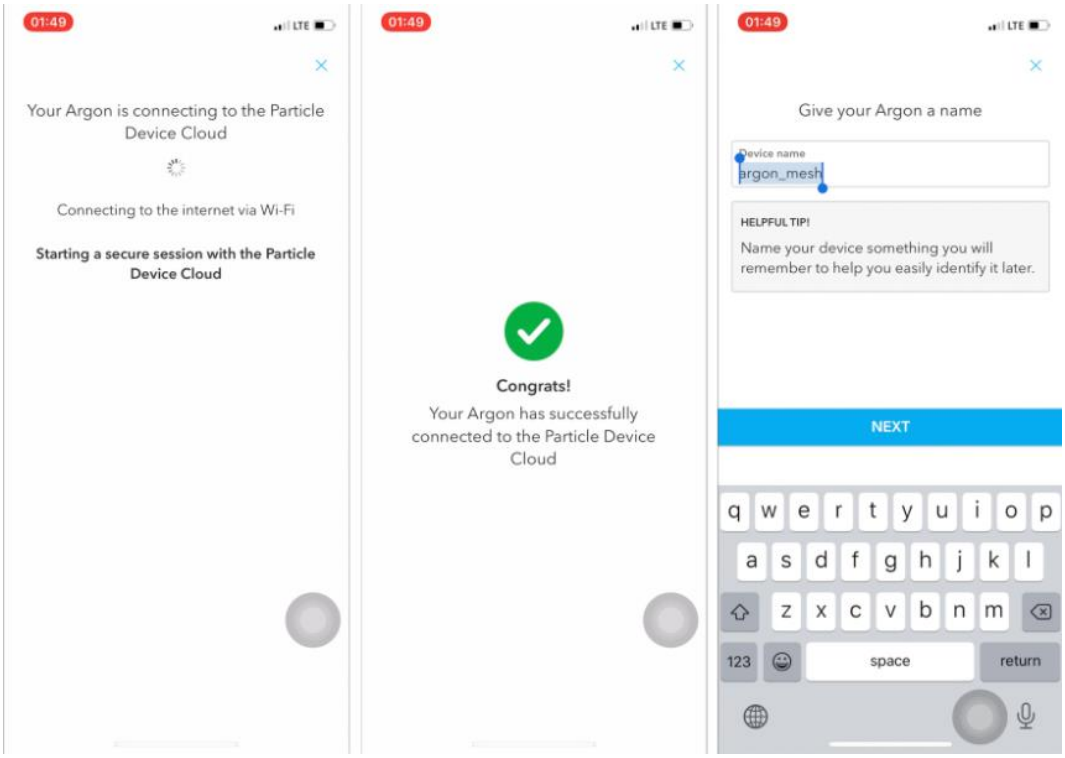

Figure 2.9 – Connection to Particle Device Cloud

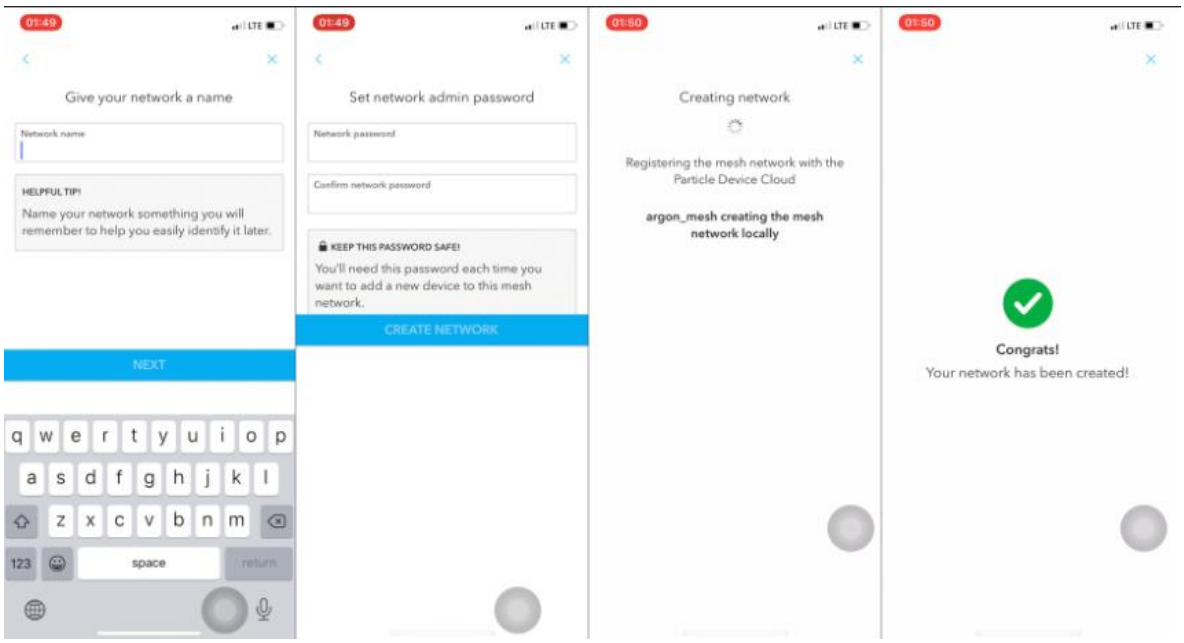

Figure 2.10 – Establishment of a local mesh network

If everything goes well the congrats message appears “Your Argon has successfully connected to the Particle Device Cloud”. After you have to set a name to the device and to the network. In my case, I called Argon as “argon_mesh”, and the network with my name “28ccurred28”. Also, there you should enter a password to the mesh network to secure it. And it automatically registers the network with the Particle Device Cloud. In the end, congrats message appears “Your network has been created” (Figure 2.10).

2.2.2 Xenon setup. Connecting to mesh network

The setup of Xenon is most likely to Argons. However, there are a few differences. First, you have to scan data matrix code in front of Xenon board. “Make sure the sticker is steady and well-lit” message appears. And you have to plug your device into a power source like USB or battery (Figure 2.11).

After you successfully paired with the device, it will automatically connect to the mesh network that is established by Argon. And makes a connection with Particle Device Cloud via the gateway calling a message “Starting a secure session with the Particle Device Cloud via gateway”. After appears congrats page “Your Xenon has been successfully added to 28ccurred28 mesh network”. In the end, on the console page, two devices appear lighting blue as a sign that the devices are online and connected to Particle Device Cloud (Figure 2.12).

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This transmitter module must not be co-located or operating in conjunction with any other antenna or transmitter. This End equipment should be installed and operated with a minimum distance of 20 centimeters between the radiator and your body.

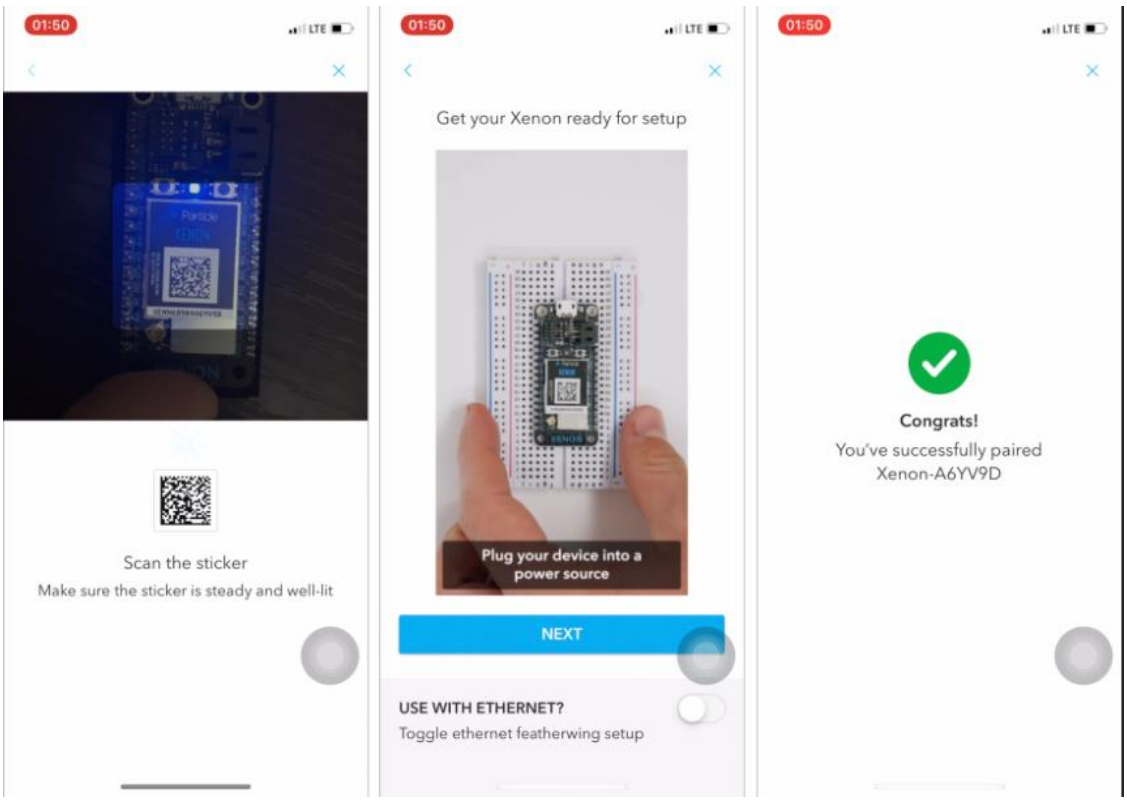

Figure 2.11 – Xenon setup

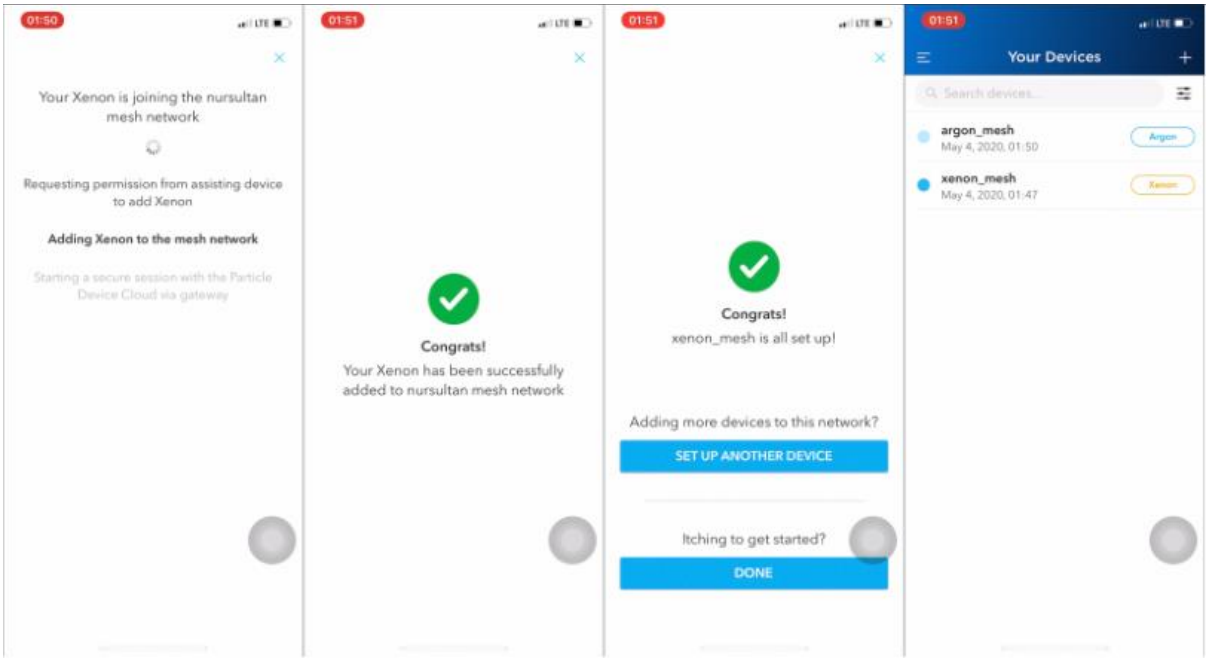

Figure 2.12 – Connection of Xenon to mesh network

2.3 Connecting temperature sensor to Xenon

To test that mesh network is working, I used a temperature sensor and connected it to the Xenon board. First, I have selected a temperature sensor and my choice was AM2302(DHT22). Because my main goal is to establish and research the mesh network I choose the optimal temperature sensor for this project. I think

that this sensor is good for this project and it is commonly used. Also, in the development area, you can easily find and install its library.

To come to the electrical scheme of this connection you can see this in Figure 2.13. The temperature sensor has three wires, first is for Power source, a second wire is for control and data sending, last is for Grounding. The Power source and Grounding wires are connected to Xenon special pins for that purpose. However, the controlling wire needs a power supply to work properly. So, I connected a 10kOhm resistor between the power source and the control wire. And the second end of control wire goes to the D1 pin on the Xenon board.

Moreover, to test the temperature sensor is working I added a red LED light to the Xenon board. It lights when the temperature sensor sends the value of temperature to Cloud. In the programming part, I have programmed them all.

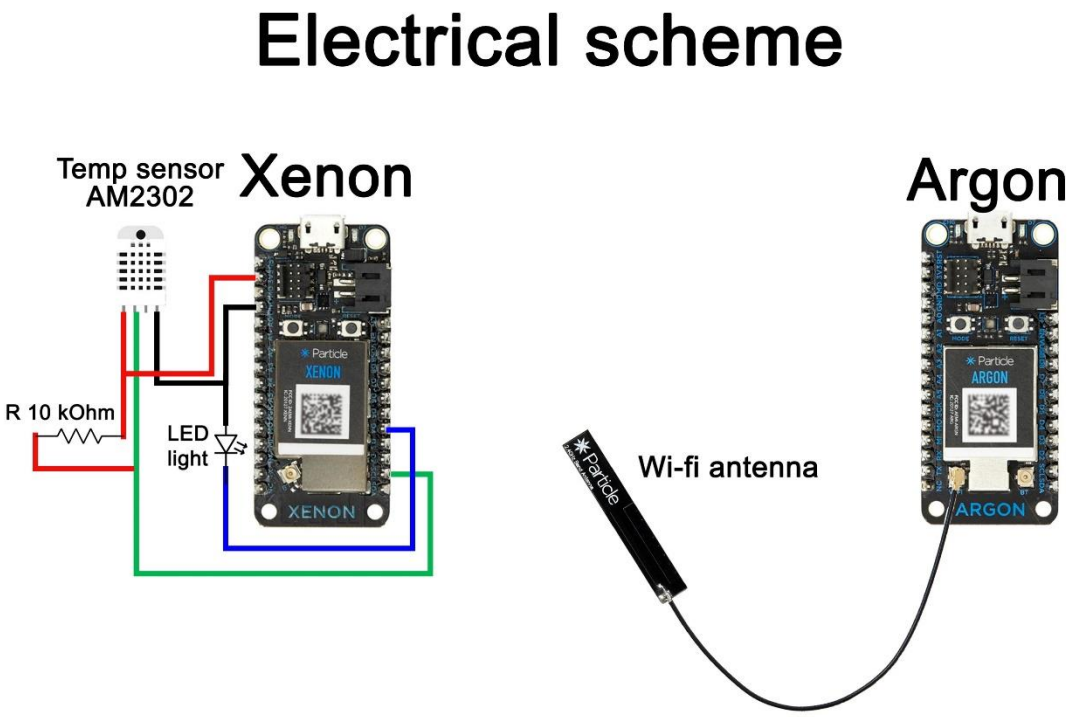

Figure 2.13 – Electrical scheme

2.4 Programming the devices

It has been programmed in the way that the Xenon board can read the temperature sensor and lights the LED. After sends the temperature value from sensor to Cloud via the Argon gateway.

Firsts of all, we need to download the Visual Studio Code developer tool from Microsoft corporation and install it to our Windows machine. Then, on the website of Particle, we downloaded Particle Workbench for VS Code. After its installation

in the VS Code appears a new extension Particle Workbench as you can see in Figure 2.15. From there you can start your new project.

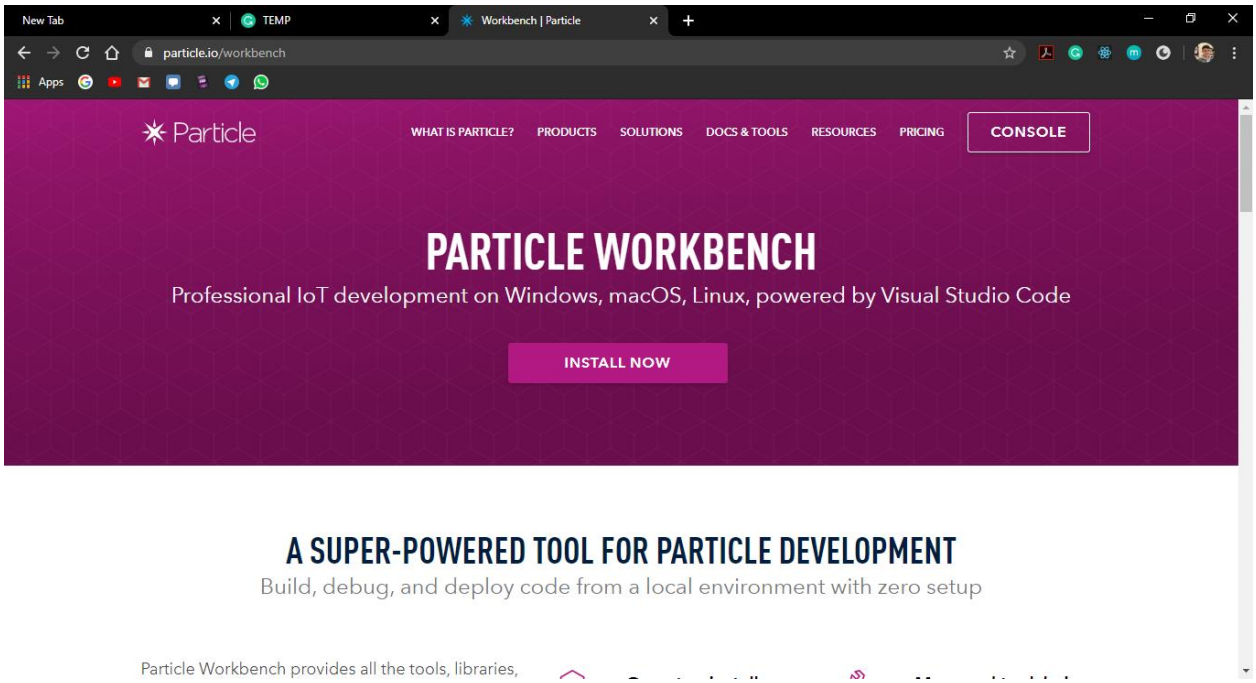

Figure 2.14 – Particle Workbench Download page

If you're new to Visual Studio Code, the Command Palette will become a familiar part of the user interface. As the name implies, the Command Palette provides access to many commands such as open files, search for symbols, and see a quick outline of a file, all using the same interactive window. It can be invoked via cmd+shift+p on macOS or ctrl+shift+p on Linux and Windows.

Workbench adds custom Particle commands to the palette. Start typing Particle to see all the currently available commands.

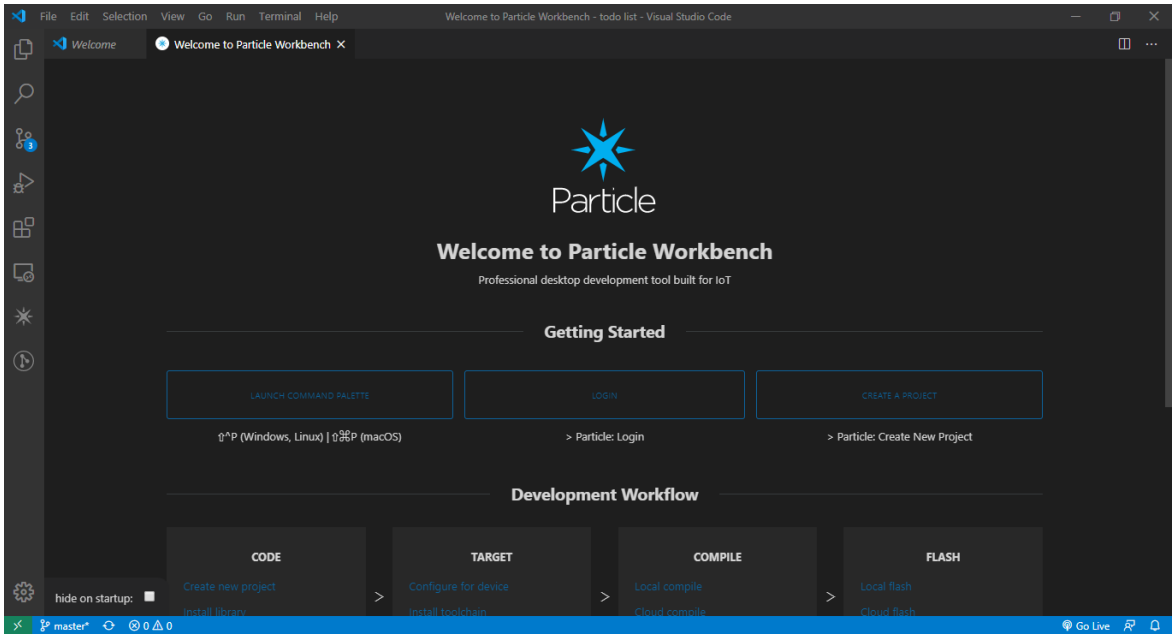

Figure 2.15 – VS Code Particle Workbench look

Downloading and maintaining a local toolchain can be a full-time job so Workbench introduces a new dependency manager. It downloads the Device OS, build system, compiler, and anything else needed to develop and debug Device OS apps, and places them in a local, private location in user space as to not mess with your current configuration.

You can install and uninstall different versions of Device OS (and dependencies) with Particle: Install Local Compiler. You can uninstall unneeded versions just as easily.

As you can see in Figure 2.16 my actual code was short however there a lot of files and dependencies under it. On the left side, there are files and libraries for the project. First, .vscode folder includes all default dependency files for the project. Second, the lib folder includes libraries that I have installed manually. And there is an “Adafruit_DHT” library for my temperature sensor. The “src” folder contains my main files where I have written all necessary code.

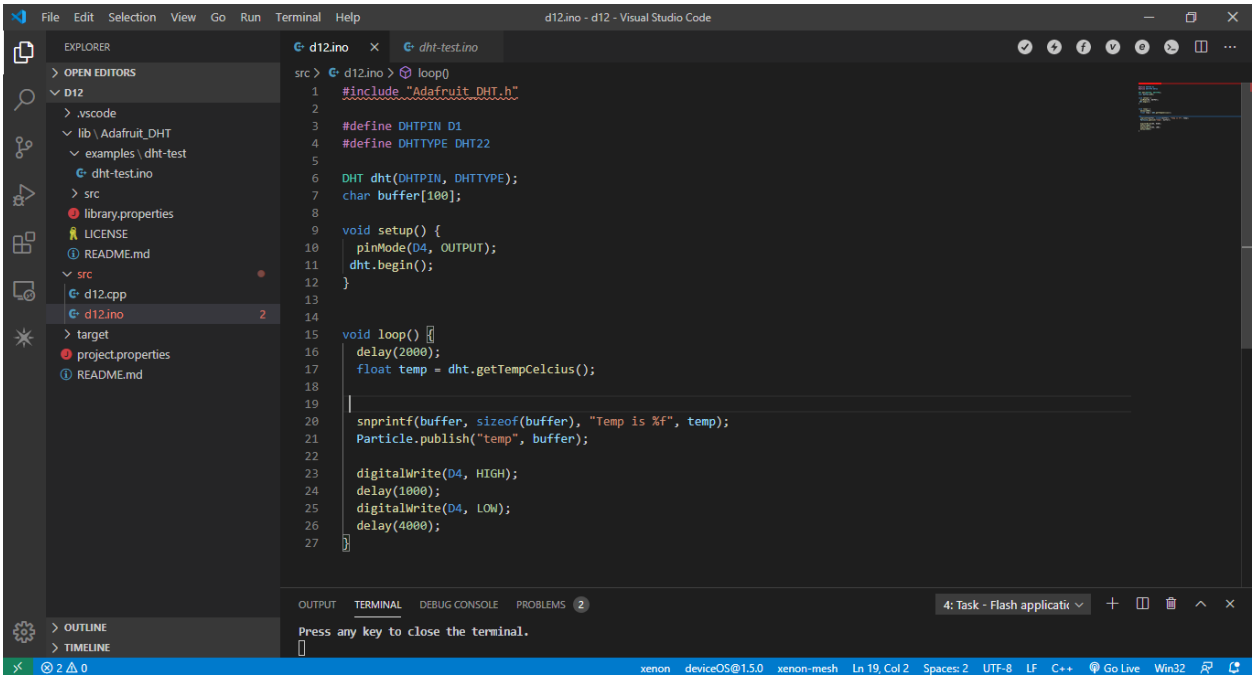

Figure 2.16 – Code of the project in VS Code

At the first line of the code, we imported the DHT library and then defined its type and control pin as D1. In the setup function, we started our temp sensor and the pin for the red LED. In the loop function, it first delays 2s because of temperature sensor sense temperature in 2s.

After, we save temperature value in the “temp” variable and send it to Particle Device Cloud by command “Particle.publish(temp)”.

At the end on the pin, D4 goes signal to LED light and lights it.

You can see the block diagram of the process in Figure 2.17. This loop is infinite and works till you switch off the device. And it is for research purpose such as measuring the working range and speed of data transmission.

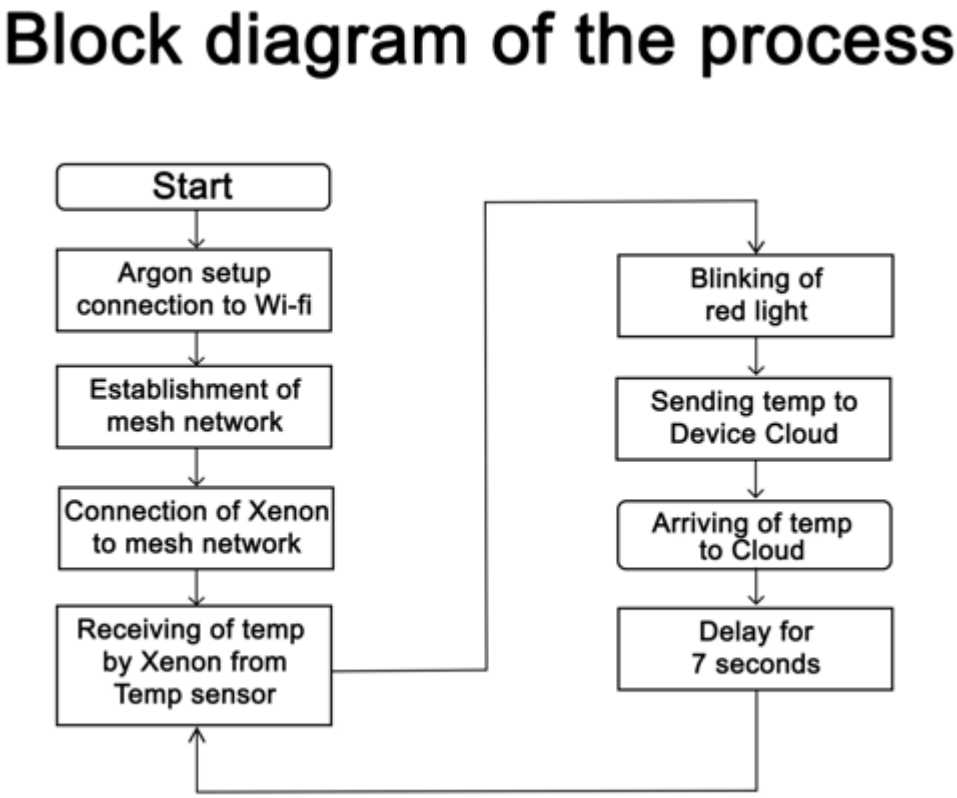

Figure 2.17 – Block diagram of the process

2.5 Testing the network

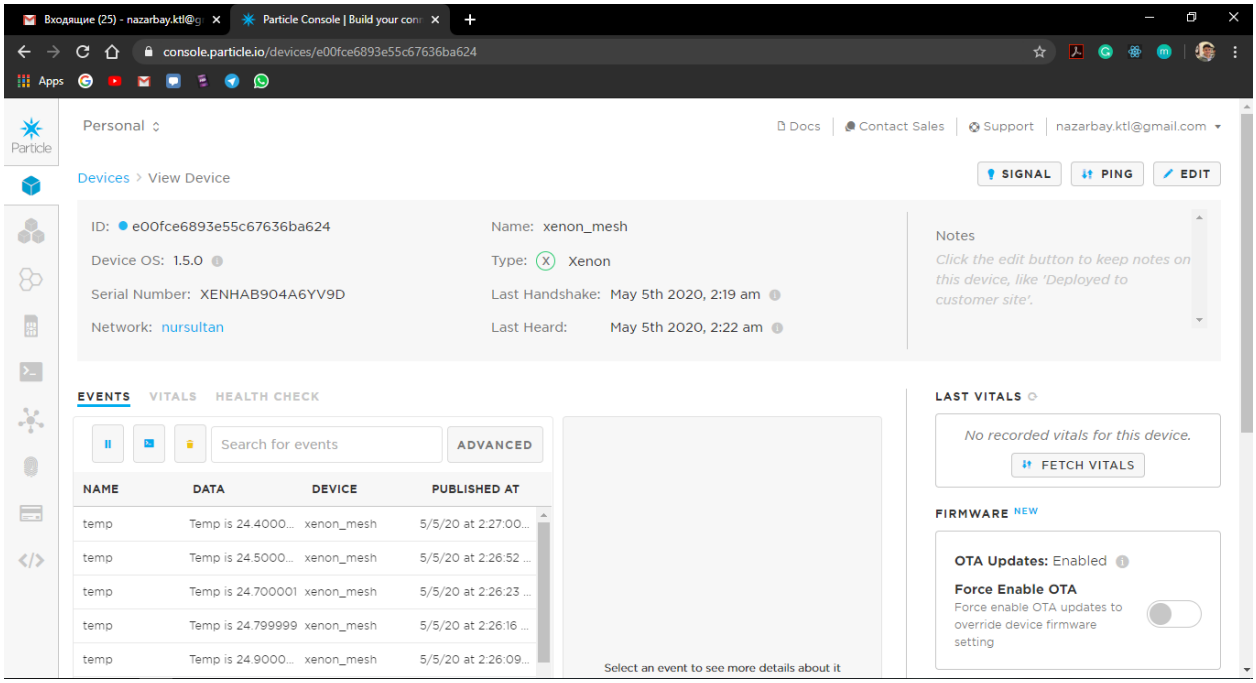

Figure 2.18 – Particle Device Cloud Xenon is sending messages

After programming the Xenon board, we need to test if our application and the mesh network is working. To do that we just have to log to Particle Device Console and watch the Xenon boards event log. In this part of Console, there should

be our temperature value with its time of sending. If it is true, our application is working. In my project, my application is working well and without any problems. You can see it in the picture above.

You are viewing the Hardware Examples for the Xenon. To view the documentation for other devices, use the blue device selector below the Particle logo on the left side of the page.

Here you will find a bunch of examples to get you started with your new Particle device! The diagrams here show the Photon, but these examples will work with either the Photon.

These examples are also listed in the online IDE in the Code menu.

3 Research of the IoT mesh network

3.1 The applications look

In the pictures below you can see my final application. Figure 3.1 shows that the mesh network is established between Devices as they are blinking green light. And the temperature sensor is sensing a temperature of the surrounding area. In Figure 3.2 the red LED is lighting showing that our Xenon successfully takes the value of “temp” and sending it to Cloud.

It’s good practice to connect the red (+) bus bar on the top to 3V3 and the blue (-) bus bar on the bottom to ground.

3V3 is second from the left on the bottom (with the USB connector on the left). It is often connected using a red wire.

Ground is the fourth from the left on the bottom. It is typically connected using a black wire.

Position the LED in the breadboard. The long lead (anode) goes to + (left) and the short lead (cathode) goes to – (right). When using an LED, you must always add a current liming resistor. Normally you’d use a 220 ohm resistor (red-red-brown-gold) for 3.3 volt circuits.

In the picture, the long lead of the LED connects to pin D6 using the blue wire. The short lead of the LED connects to a 220 ohm resistor that connects it to ground. That completes the circuit.

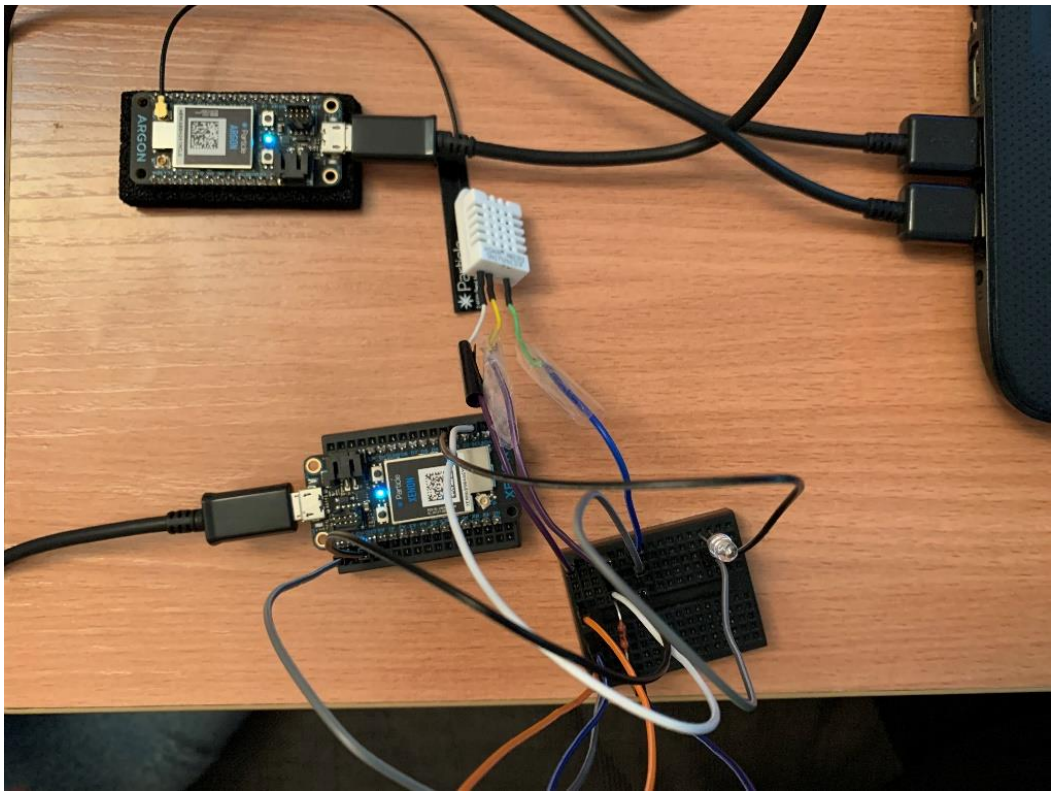

Figure 3.1 – Completed look of application

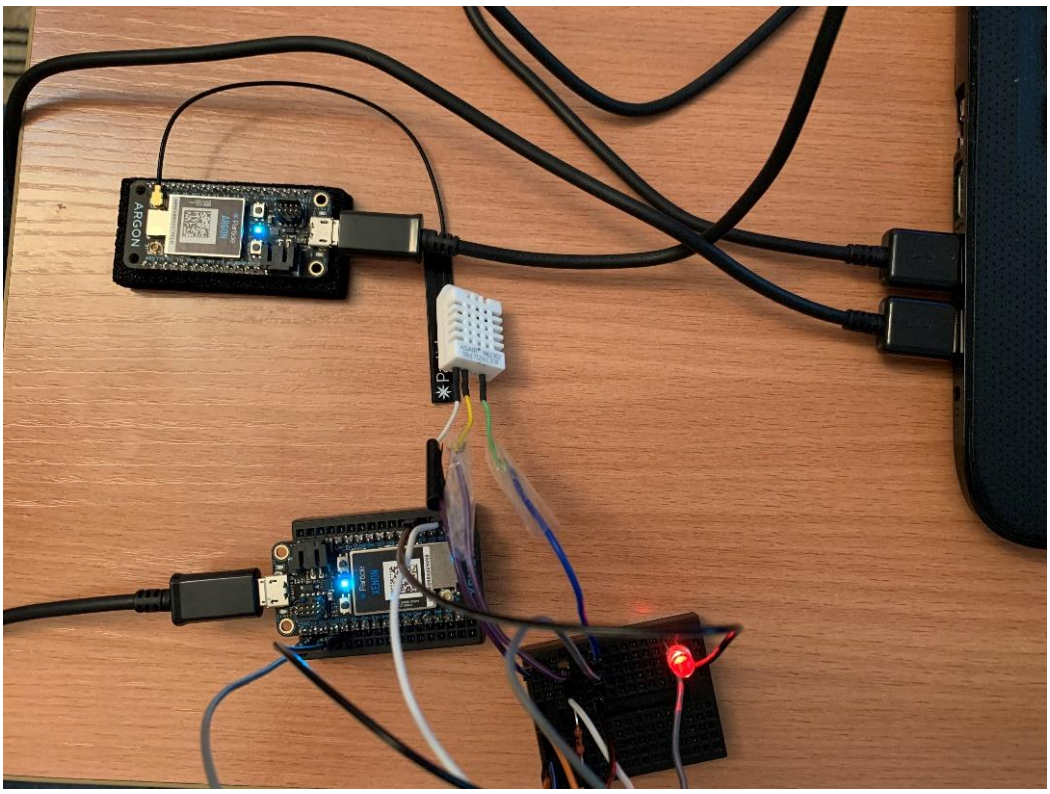

Figure 3.2 – Lighting red LED when sending data to Cloud

Remember back when we were blinking lights and reading sensors with Tinker on the mobile app?

When you tap a pin on the mobile app, it sends a message up to the cloud. Your device is always listening to the cloud and waiting for instructions–like “write D7 HIGH” or “read the voltage at A0”.

Your device already knew how to communicate with the mobile app because of the firmware loaded onto your device as a default. We call this the Tinker firmware. It’s just like the user firmware you’ve been loading onto your device in these examples. It’s just that with the Tinker firmware, we’ve specified special that the mobile app knows and understands.

If your device is new, it already has the Tinker firmware on it. It’s the default firmware stored on your device right from the factory. When you put your own user firmware on your device, you’ll rewrite the Tinker firmware. (That means that your device will no longer understand commands from the Particle mobile app.) However, you can always get the Tinker firmware back on your device by putting it in factory reset mode, or by re-flashing your device with Tinker in the Particle app.

To reflash Tinker from within the app:

iOS Users: Tap the list button at the top left. Then tap the arrow next to your desired device and tap the “Re-flash Tinker” button in the pop out menu.

Android Users: With your desired device selected, tap the options button in the upper right and tap the “Reflash Tinker” option in the drop down menu.

The Tinker app is a great example of how to build a very powerful application with not all that much code. If you're a technical person, you can have a look at the latest release.

I know what you're thinking: this is amazing, but I really want to use Tinker while my code is running so I can see what's happening! Now you can.

Combine your code with this framework, flash it to your device, and Tinker away. You can also access Tinker code by clicking on the last example in the online IDE's code menu.

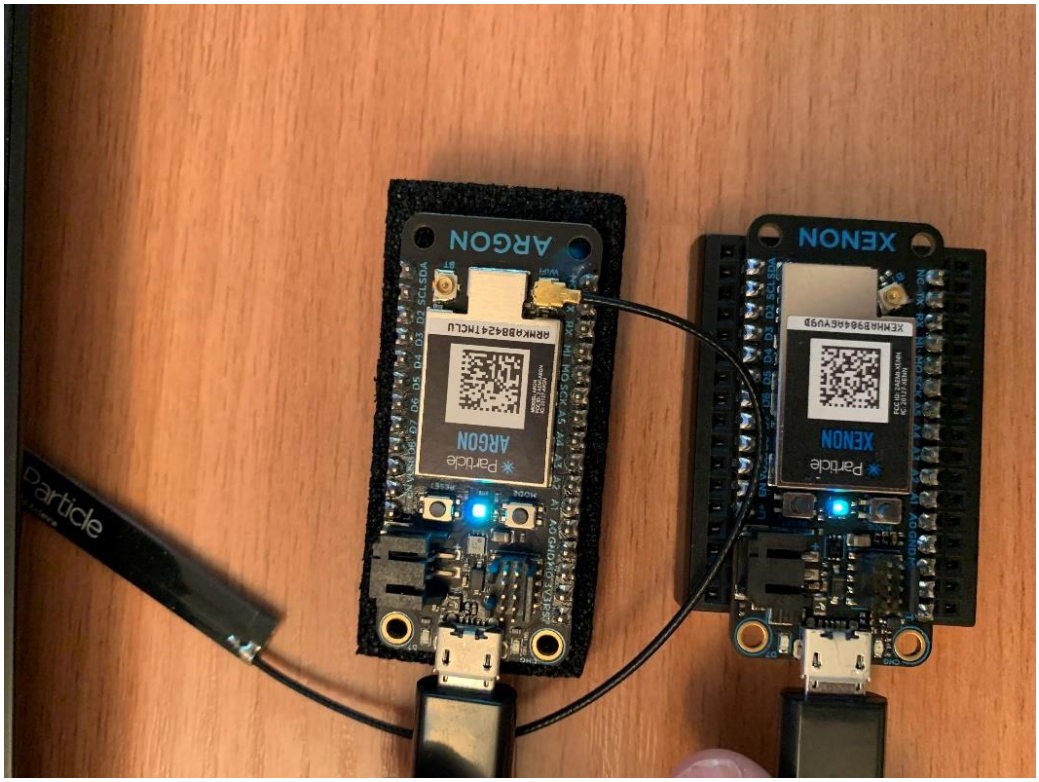

Figure 3.3 – Argon and Xenon in mesh network

3.2 Range and speed of the mesh network

First of all, I have measured the working range of the mesh network between Argon and Xenon. As it was expected the range is smaller without a special antenna for Xenon and Argon. The result was:

- 20 meter indoors, 10/10 broadcasts, 13 – 15 ms
- 40 meter indoors, 9/10 broadcasts, 15 – 17 ms
- 50 meter indoors, 7/10 broadcasts, 20 – 30 ms
- 60 meter indoors, 1/10 broadcasts, 40 – 70 ms

Next we have the loop function, the other essential part of a microcontroller program. This routine gets repeated over and over, as quickly as possible and as many times as possible, after the setup function is called. Note: Code that blocks for too long (like more than 5 seconds), can make weird things happen (like dropping the network connection). The built-in delay function shown below safely interleaves

required background activity, so arbitrarily long delays can safely be done if you need them.

When we register a function or variable, we're basically making a space for it on the internet, similar to the way there's a space for a website you'd navigate to with your browser. Thanks to the REST API, there's a specific address that identifies you and your device. You can send requests, like GET and POST requests, to this URL just like you would with any webpage in a browser.

Remember the last time you submitted a form online? You may not have known it, but the website probably sent a POST request with the info you put in the form over to another URL that would store your data. We can do the same thing to send information to your device, telling it to turn the LED on and off.

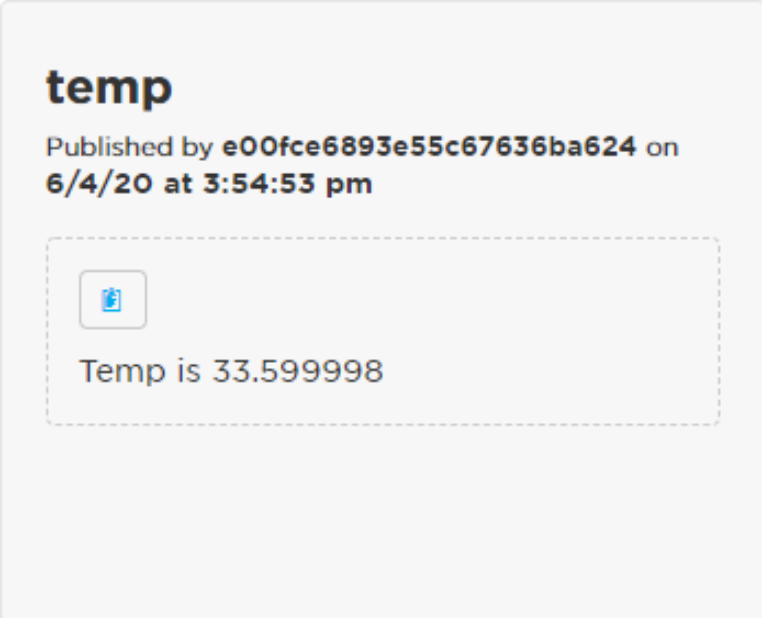

Figure 3.4 – Temperature value and its time in Cloud

As we can see from results the mesh network properly works approximately 50 meters range. As we move forward from this range it can execute errors and data can be not delivered. So, it is recommended that you use a mesh network in 40 – 50 meters range.

3.3 Power consumption of the devices

We recommend those two sources because they are known to carry good quality batteries and the JST-PH connector is wired the same as Particle batteries.

Unfortunately, there is no standard for polarity, and a large number batteries you find on Amazon, eBay, and AliExpress will be wired reverse polarity.

Particle devices must use batteries wired like this: With the key facing down and the wires toward you, the red is on the right.

We do not recommend wiring multiple batteries in parallel. The Fuel Gauge (battery monitor) is only designed for a single cell, and in the case of failure of one

battery, excess current can flow into the defective battery, causing damage or even fire.

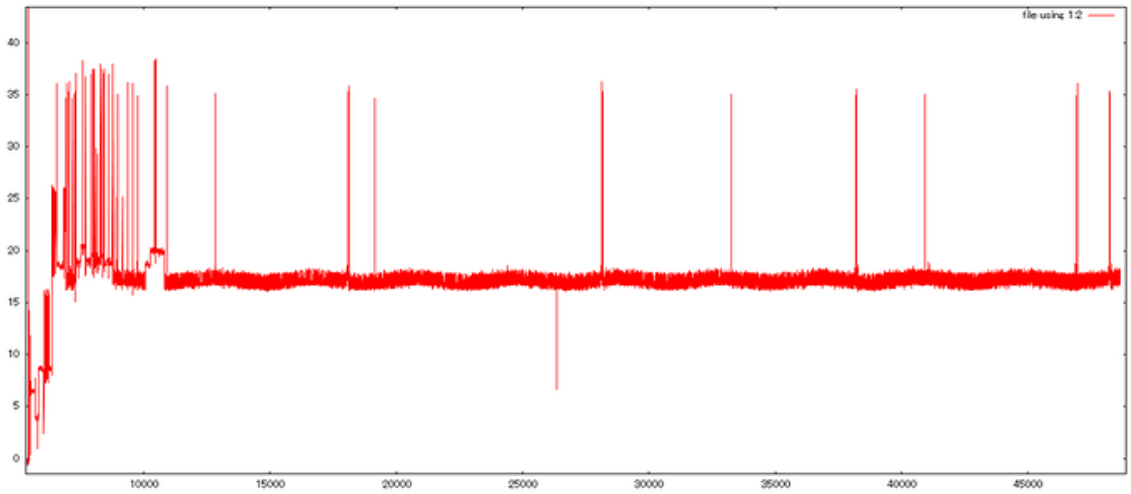

Figure 3.5 – Xenon current consumption graph

Axis: X=millisecond, Y=mA

What it does: powering on at the leftmost side, establishing the network in ~6 seconds, and then sending the voltage of an analog pin by Mesh.publish() every 10 seconds

Observation: ~17mA for normal CPU cycles (including delay()) and ~35mA for communication

I used INA219+Arduino to measure the current. The power source is USB 5V dropped down to 3.3V by my own regulator, which is eventually connected to the 3V3 input of Xenon.

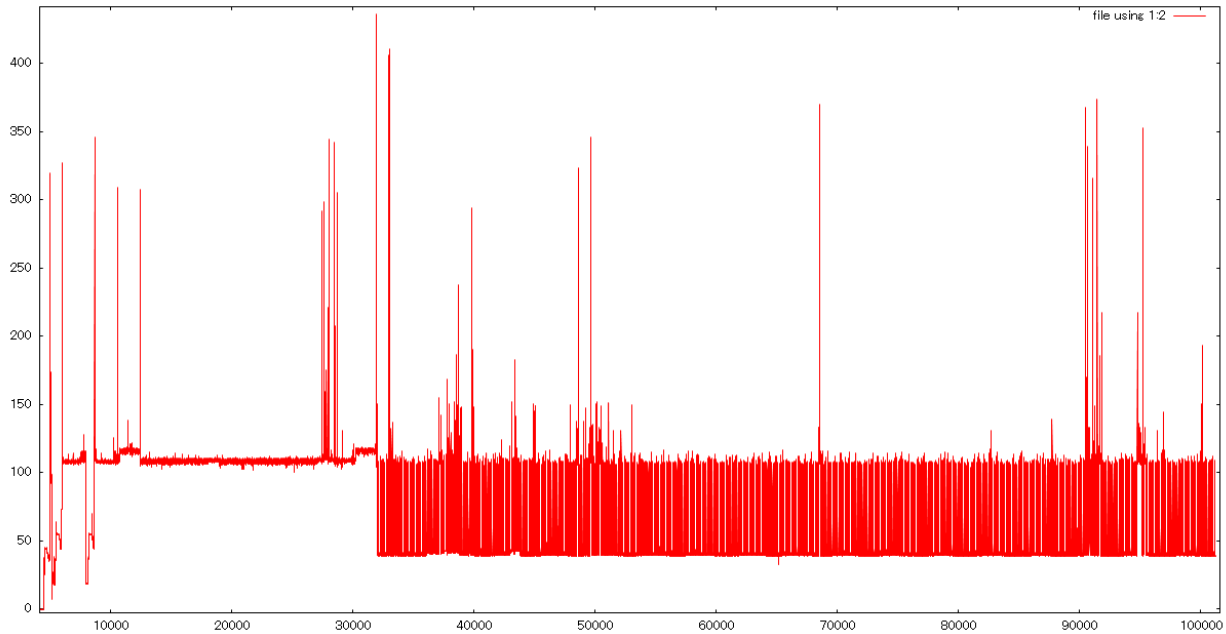

Figure 3.6 – Argon current consumption graph

Axis: X=millisecond, Y=mA

What it does: powering on at the leftmost side, establishing the network in ~30 seconds (I do not remember exactly), and then sending a cloud event by Particle.publish() when Mesh.subscribe() receives something from mesh nodes (Xenon's) and at some other occasions

Observation: ~40mA at idle, ~110mA spike every 100ms, and 200~400mA (largely variable) for WiFi communication.

Here are my thoughts trying to understand this chart. (Just my guess, it could be wrong.)

It looks like ESP32 is at the "light sleep mode" when idle. It would take ~20mA as to my previous ESP8266 experience.

It looks like ESP32 wakes up every 100ms for house keeping (probably processing event subscriptions etc.). This "working" state would take 90mA as to my previous ESP8266 experience likewise.

Since Argon is like Xenon+ESP32, if you add up Xenon's current consumption = 17mA~35mA as shown in my previous post, then the total current consumption would be pretty much like this chart.

What if we simply want to know that something has happened, without all the information of a variable or all the action of a fuction? We might have a security system that tells us, "motion was detected." Or a smart washing machine that tells us "your laundry is done." In that case, we might want to use Particle.publish.

Particle.publish sends a message to the cloud saying that some event has 40ccurred. We're allowed to name that event, set the privacy of that event, and add a little bit of info to go along with the event.

In this example, we've created a system where you turn your LED and photo sensor to face each other, making a beam of light that can be broken by the motion of your finger. Every time the beam is broken or reconnected, your device will send a Particle.publish to the cloud letting it know the state of the beam. Basically, a tripwire.

For your convenience, we've set up a little calibrate function so that your device will work no matter how bright your LED is, or how bright the ambient light may be. Put your finger in the beam when the D7 LED goes on and hold it in the beam until you see two flashes from the D7 LED. Then take your finger out of the beam. If you mess up, don't worry–you can just hit "reset" on your device and do it again.

You can check out the results on your console at console.particle.io. As you put your finger in front of the beam, you'll see an event appear that says the beam was broken. When you remove your finger, the event says that the beam is now intact.

Remember back when we were blinking lights and reading sensors with Tinker on the mobile app?

When you tap a pin on the mobile app, it sends a message up to the cloud. Your device is always listening to the cloud and waiting for instructions–like “write D7 HIGH” or “read the voltage at A0”.

Your device already knew how to communicate with the mobile app because of the firmware loaded onto your device as a default. We call this the Tinker firmware. It’s just like the user firmware you’ve been loading onto your device in these examples. It’s just that with the Tinker firmware, we’ve specified special that the mobile app knows and understands.

If your device is new, it already has the Tinker firmware on it. It’s the default firmware stored on your device right from the factory. When you put your own user firmware on your device, you’ll rewrite the Tinker firmware. (That means that your device will no longer understand commands from the Particle mobile app.) However, you can always get the Tinker firmware back on your device by putting it in factory reset mode, or by re-flashing your device with Tinker in the Particle app.

To reflash Tinker from within the app:

iOS Users: Tap the list button at the top left. Then tap the arrow next to your desired device and tap the “Re-flash Tinker” button in the pop out menu.

Android Users: With your desired device selected, tap the options button in the upper right and tap the “Reflash Tinker” option in the drop down menu.

The Tinker app is a great example of how to build a very powerful application with not all that much code. If you’re a technical person, you can have a look at the latest release.

I know what you’re thinking: this is amazing, but I really want to use Tinker while my code is running so I can see what’s happening! Now you can.

Combine your code with this framework, flash it to your device, and Tinker away. You can also access Tinker code by clicking on the last example in the online IDE’s code menu.

4 Life safety

4.1.1 Development of questions on ensuring electrical safety

The danger of electric current, unlike other dangerous and harmful production factors, is exacerbated by the fact that the human senses do not detect imminent danger from a distance. A person's reaction to electric current occurs only when it passes through the body. Electric current has a thermal, electrolytic, mechanical and biological effect on humans.

The thermal effect of the current manifests itself in burns, heating of blood vessels and other organs, as a result of which functional disorders arise in them.
The electrolytic effect of the current is characterized by the decomposition of blood and other organic liquids, which causes a violation of their physico-chemical composition.

The mechanical action of the current is manifested in damage (rupture, stratification, etc.) of various tissues of the body as a result of the electrodynamic effect.

The biological effect of current on living tissue is expressed in the dangerous excitation of cells and body tissues, accompanied by involuntary convulsive muscle contractions. As a result of such excitement, a disturbance and even complete cessation of the activity of the respiratory and circulatory organs can occur.

The irritating effect of current on body tissues can be direct when the current passes directly through these tissues, and reflex, through the central nervous system. There are two main types of electric shock: electrical injuries and electric shocks.

Electrical injuries are divided into electrical burns, electrical signs, electrometallization of the skin, mechanical damage and electrophthalmia.

Electric burns, depending on the conditions of their occurrence, are of two types: current (contact) and arc.

A current burn is a consequence of the conversion of electrical energy into heat and is caused by the passage of current directly through the human body as a result of touching live parts. Distinguish electric burns of four degrees. The main signs of burns of I degree – redness of the skin, II degree – the formation of blisters, III degree – carbonization of the skin, IV degree – carbonization of the subcutaneous tissue, muscles, bones.

Arc burn is the result of the action on the human body of an electric arc in high voltage electrical installations. Such a burn is usually severe (III or IV degree). Electrical signs (electrical marks) are spots of gray or pale yellow on the surface of the skin at the point of contact with live parts. In most cases, they are painless. Over time, the damaged skin layer disappears.

Electrometallization of the skin is the penetration into the upper layers of the skin of the smallest metal particles during its melting or evaporation under the influence of an electric arc. The damaged area of the skin becomes hard and rough,

has a specific color, which is determined by the color of the metal that has penetrated the skin. Electrometallization of the skin is not dangerous. Over time, the damaged skin layer disappears, and the affected area takes on a normal appearance.

Mechanical damage occurs due to sharp involuntary convulsive muscle contractions under the influence of current. As a result, ruptures of the skin, blood vessels, nerve tissue, as well as dislocation of joints and bone fractures are possible.

Electrophthalmia is eye damage due to exposure to ultraviolet radiation from an electric arc.

An electric shock is the excitation of living body tissues by an electric current passing through it, accompanied by involuntary convulsive muscle contraction. With an electric shock, clinical death can occur, which in the absence of qualified medical care after 7-8 minutes passes into biological death. If, in the event of clinical death, the victim is immediately released from the action of electric current and urgently begin to provide the necessary assistance (artificial respiration, heart massage), then the life of the victim can be saved.

Causes of death from exposure to electric current may be cardiac arrest or its fibrillation. Fibrillation (from lat. Fibra – fiber). The chaotic contraction of individual fibers of the heart muscle, which is unable to maintain its effective work and cannot pass on its own (without vigorous therapeutic measures), respiratory arrest and electric shock are a kind of neuro-reflex reaction of the body in response to severe irritation by an electric current, accompanied by circulatory, respiratory distress metabolism, etc. The shock state can last from several tens of minutes to a day. With prolonged shock, death may occur.

The nature of the effect of electric current on the human body and the severity of the lesion depend on the strength of the current, the duration of its impact, the nature and frequency, the path of the passage of current in the body. The individual properties of a person and some other factors are of particular importance.

The strength of the current passing through the human body is the main factor determining the outcome of the lesion. Different in magnitude currents have a different effect on the human body.

Distinguish tangible, non-releasing and fibrillation currents.

The threshold values of perceptible currents are: 0.6-1.5 mA with alternating current with a frequency of 50 Hz and 5-7 mA with direct current. Such a current causes mild itching, pinching of the skin under the electrodes, and an alternating current of 8-10 mA already causes severe pain and cramps throughout the arm, including the forearm. It is difficult to hand, but in most cases you can still tear it away from the electrode.

An electric current that causes insurmountable convulsive contractions of the muscles of the arm in which the conductor is clamped when passing through a person is called a non-releasing current.

Alternating current (50 Hz) with a power of 10-15 mA causes barely tolerable pain in the whole arm. In many cases, the hand cannot be torn from the electrode. With an alternating current of 20-25 mA, the hands become paralyzed instantly, it becomes impossible to break away from the electrodes, and a current of 25-50 mA causes very severe pain in the arms and chest. Breathing is extremely difficult.

With an AC current of 50-80 mA, breathing is paralyzed after a few seconds, the heart is disrupted. With prolonged current flow, cardiac fibrillation may occur. An electric current that causes cardiac fibrillation to pass through the body is called fibrillation current. An alternating current of 100 mA after 2-3 s causes fibrillation of the heart, and after a few seconds it causes paralysis. The upper limit of the fibrillation current is 5 A. A current of more than 5 A, both alternating and constant, causes an immediate cardiac arrest, bypassing the state of fibrillation.

The voltage largely determines the outcome of the lesion, since the resistance of the skin and the strength of the current passing through the human body depend on it.

The electrical resistance of the human body is determined by the resistance of the skin in places of inclusion in the electrical circuit and the resistance of internal organs. Moreover, skin resistance is the main share of the total resistance. The highest stratum corneum of the skin has the highest resistance (epidermis). The resistance of the human body varies in the range of 1-100 kOhm or more.

When moisturizing, contaminating and damaging the skin (perspiration, cuts, abrasions, scratches, etc.), increasing the current strength and its duration, as well as increasing the area of contact with current-carrying elements, the resistance of the human body decreases to a minimum value (Fig. 1.1)

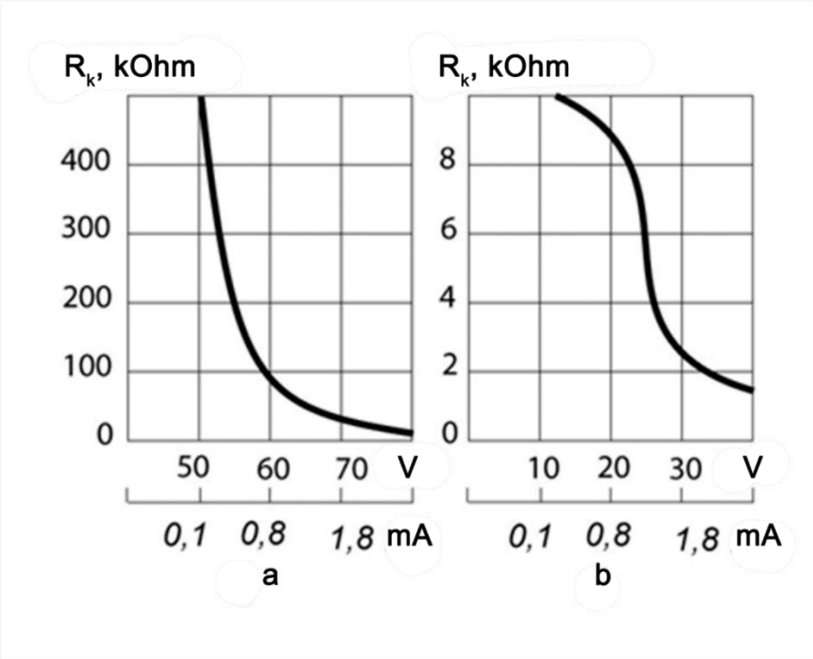

Figure 4.1 – The dependence of the resistance of the human body on voltage at a frequency of 50 Hz: a – dry skin, b – wet skin.

The resistance of the internal tissues of the human body is negligible and amounts to 300-500 Ohms. In the calculations, the electrical resistance of the human body is taken equal to 1000 Ohms.

The total resistance of the human body Z, taking into account the active resistance and electrical capacity C of the human body is determined by the formula:

$$Z = \sqrt{R_h^2 + (\frac{1}{\omega C})^2}$$

Where: ω is the angular frequency of the current.

At a current frequency of 50 Hz, the influence of capacitance is negligible and is not taken into account in the calculations. Thus, we can assume that the strength of the current passing through the human body is equal to:

$$I_h = \frac{U}{R_h}$$

Where: U is the voltage applied to two points of the human body.

The duration of the effect of current on the human body in many cases is a determining factor on which the outcome of the lesion depends: the longer the action of the current, the greater the likelihood of a severe or fatal outcome.

The nature and frequency of the current also affect the severity of the lesion. The most dangerous is an alternating current with a frequency of 20-100 Hz. At a frequency of less than 20 or more than 100 Hz, the risk of electric shock is significantly reduced.

A constant current of the same magnitude as a variable causes weaker muscle contractions and is less noticeable. Its effect is mainly thermal, but with significant values, burns can be very severe and even fatal. Currents above 500 kHz cannot stop the work of the heart or lungs. However, this current can cause burns.

The current path through the human body significantly affects the outcome of the lesion. The danger of damage is especially great if the current passing through vital organs – the heart, lungs, brain, acts directly on these organs. If the current does not pass through them, then its effect is only reflex, and the likelihood of severe damage decreases (Table 4.1).

Table 4.1 – Characterization of the most common current paths in the human body

Current path	Frequency occurrence this way %	Share of those who lost consciousness during current exposure %	Share of current passing through heart area in % of the total current

Hand-to-hand	40	83	3,3
Right arm-legs	20	87	6,7
Left leg-arm	17	80	3,7
Foot to foot	6	15	0,4
Head-legs	5	88	6,8
Head-hands	4	92	7
Other	8	65	-

Individual characteristics of a person significantly affect the outcome of an electric shock. The current causing weak sensations in one person may turn out to be non-releasing for another. The nature of the effect of the current of the same force depends on the mass of a person and his physical development. For women, the threshold current values are about one and a half times lower than for men. The degree of current exposure depends on the state of the body. So, in a state of fatigue and intoxication, people are much more sensitive to the effects of current. It was found that quite healthy and physically strong people withstand electric shocks more easily than sick and weak. Persons suffering from diseases of the skin, cardiovascular system, organs of internal secretion, lungs, nervous and other diseases have an increased susceptibility to electric current.

Of great importance is the psychological preparedness for the possible danger of electric shock. In the vast of cases, an unexpected electric shock leads to more serious consequences. When a person expects an impact, the degree of damage is significantly reduced.

4.1.2 Protective grounding

Protective grounding - a deliberate electrical connection to the ground or its equivalent metal non-current-carrying parts that may be energized as a result of insulation damage (GOST 12.1.009-76 GOST 12.1.009-76 SSBT. "Electrical safety. Terms and definitions."). Protective grounding is used in networks with voltages up to 1000 V with insulated neutral and in networks with voltages higher than 1000 V with both insulated and grounded neutrals.

According to GOST 12.1.030-81 GOST 12.1.030-81 SSBT. "Electrical safety. Protective grounding, grounding. " metal parts of electrical installations that are accessible to human touch and do not have other types of protection ensuring electrical safety are subject to protective grounding. Protective grounding should be carried out: at a rated voltage of 380 V and above alternating current and 440 V and above direct current - in all cases; at rated voltage of 42-380 V of alternating current

and 110-440 V of direct current when working in conditions with increased danger and especially dangerous.

Protective grounding is designed to eliminate the risk of electric shock in the event of voltage on metal non-conductive parts of electrical equipment (for example, due to a short to the housing if the insulation is damaged). Human protection is ensured by reducing the voltage of touch and step to safe values.

If the equipment case is not grounded and one of the phases is shorted to it, then touching a person to such a case is equivalent to touching the phase. The task is to create an electrical connection between the enclosure of the protected equipment and the ground with a sufficiently low resistance so that in the event of a short circuit to the enclosure of this equipment, touching a person could not cause a dangerous current to pass through his body. This is achieved by reducing the potential of the grounded equipment, as well as by equalizing the potentials by raising the potential of the base on which a person stands to a value close to the potential of the grounded equipment.

The resistance of the grounding device in electrical installations with voltage up to 1000 V, working with insulated neutrals, must not exceed 4 Ohms.

With the power of sources supplying the network up to 100 kVA, the grounding resistance can be within 10 Ohms.

A grounding device is a set of structurally combined grounding conductors and an earthing switch (Fig. 1.2). Earthing switches are natural and artificial.

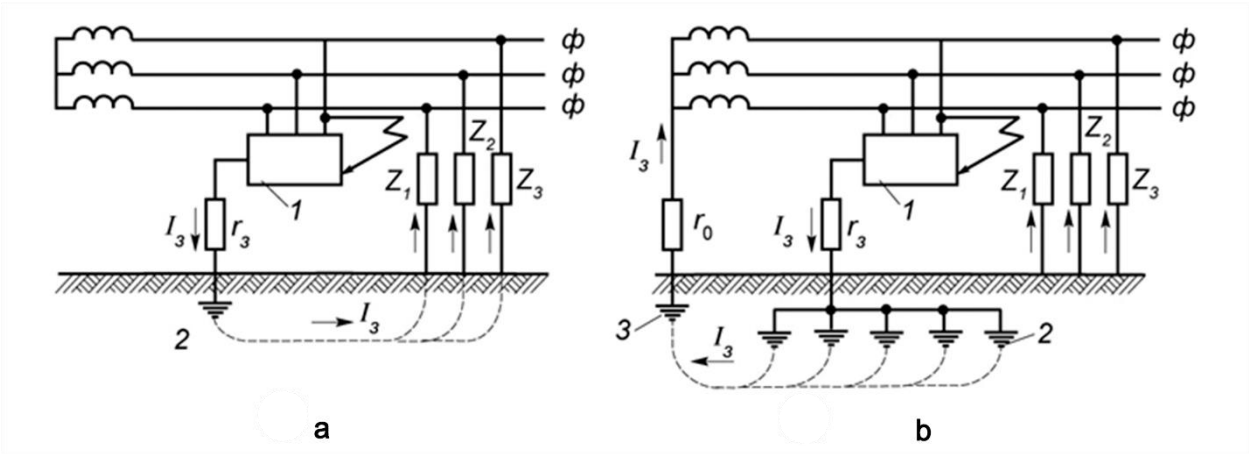

Figure 4.2 - Schematic diagrams of protective grounding

a - in a network with isolated neutral up to 1000 V and above; b - in a network with a grounded neutral; 1 - grounded equipment; 2 - grounding conductor of protective grounding; 3 - worker grounding; r_3, r_0 - resistance of the respective protective and working grounding; I_3- earth fault current.

As natural grounding conductors are used electrically conductive parts of building and industrial structures and communications.

As artificial grounding conductors, use steel pipes vertically laid in the ground (diameter 30-60 mm, length 200-300 cm, wall thickness no less than 3-5 mm); steel

corners (dimensions 60x60 mm, length 250-300 cm); steel bars (diameter 10-12 mm, length up to 10 m) or strips. The thickness of the strips must be at least 4 mm, and the cross section must be at least 48 mm^2.

As grounding conductors, steel strips and round steel are used. Grounding conductors are connected to grounding conductors and to each other by welding, and to the bodies of grounding equipment - by welding or bolts. Grounding objects are connected to the grounding line in parallel. Each electrical installation enclosure must be connected to the grounding line using a separate branch. Serial connection of several grounded equipment enclosures to the grounding line is prohibited.

The resistance of grounding conductors to current spreading is determined by their shape and size, as well as the specific resistance of the soil, depending on its type and humidity. For example, at a soil moisture content of 10-20%, the resistivity is *R* (Ohm*m) will be: for sand - 700, loamy sand 300, loam 100, clay - 40 and chernozem - 20.

In practice, for an approximate calculation of the resistance of the ground electrode (electrodes) to current spreading, you can use simplified formulas: for pipes $R_p = 0.9\ R\ /\ l$, for strip $R_p = 2.1\ R\ /\ l$, where l is the length of the electrodes (grounding), m.

For a more accurate calculation of the resistance of some single grounding conductors, use the formulas:

- for pipes (rods) of diameter *d(m)* clogged into the ground at the same level with it:

$$R_P = \frac{\rho}{2\pi l_p} ln \frac{4l_p}{d}$$

- for pipes (rods) driven into the ground to a depth h from the surface of the earth:

$$R_P = \frac{\rho}{2\pi l_p} (ln \frac{2l_p}{d} + 0.5 ln \frac{4t + l_p}{4t - l_p})$$

where *t* is the distance from the soil surface to the middle of the pipe, *m*;

- for a strip laid on the ground surface:

$$R_s = \frac{\rho}{2\pi l_s} ln \frac{4l_s}{b}$$

where *b*, *l* is the width and length of the strip, respectively, m;

- for a strip laid horizontally in the ground to a depth h from the surface of the earth:

$$R_s = \frac{\rho}{2\pi l_s} ln \frac{2l_s^2}{bh}$$

The required number of electrodes (pipes, rods) n is determined by the formula:

$$n = \frac{R_P}{R_n \eta_P}$$

where R_n - maximum permissible grounding resistance (R_n < 4 ohms); η_P - utilization of pipes, taking into account their mutual shielding. Depending on the ratio of the distance between the pipes a and their length l_m with sufficient accuracy, you can take $\eta_P = 0.45 + 0.6$ (with relation $a / l_m = 2$), and $\eta_P = 0.6 + 0.8$ (with $a / l_m = 3$).

Tie Strip Length $l_n = 1.05a(n\text{-}1)$, where a is the distance between the pipes, which is assumed to be 3-10 m (usually $a = 2l_m$).

The total resistance of the entire protective grounding system will be

$$R_{C3} = \frac{R_s R_P}{R_P \eta_n + R_n \eta_{Pn}}$$

where η_n - coefficient taking into account the mutual shielding of the electrodes with the connecting strip (you can take $\eta_n = 0.8 / 0.9$)

η_P- coefficient taking into account the mutual shielding of the electrodes (with an open circuit, you can take T = 0.75 / 0.8 and when closed – T = 0.66 / 0.75); n is the number of electrodes.

In practice, after such a calculation, a correct calculation of the required number of electrodes is carried out taking into account the spreading resistance of the connecting strip so that R_{C3} was R_3. This calculation allows to reduce the number of electrodes, i.e. save some of the metal used for protective grounding.

Suppose, Rzd = 4 Ohms is required in a house located in a warm fourth climatic zone. Soil - on top of chernozem H = 0.9 m, clay below. The length of the trench around the house (square) Lg = 40m, depth t = 0.5m. A strip of 4 * 40mm is used, round rod L = 2m, d = 0.02m. It is required to calculate the number of vertical grounding electrodes and the distance between them.

We calculate the resistance of the horizontal strip that lies in the black soil:

$$R_r = 0.366 * \left(\frac{50 * 1.75 * 1.8}{40 * 0.55}\right) * \lg\left(\frac{2 * 40^2}{0.04 * 0.5}\right) = 13.635\ Ohm$$

This value is too large and does not satisfy the required value Rzd = 4 Ohms, so you need to continue to calculate and calculate the total resistance of the vertical ground electrodes:

$$R_S = \frac{13.635 * 4}{13.635 - 4} = 5.66\ Ohm$$

Next, we calculate the equivalent soil resistivity. You need to remember about the correction factor of 1.75 - for clarity, the calculations with it are enclosed in additional brackets:

$$p_e = \frac{1.3 * (60 * 1.75) * (50 * 1.75) * 2}{(60 * 1.75) * (2 - 0.9 + 0.5) + (50 * 1.75) * (0.9 - 0.5)} = 117.672\ Ohm * m$$

Mid buried electrode:

$$T = \frac{L}{2} + t = \frac{2}{2} + 0.5 = 1.5\ m$$

We calculate a single vertical grounding switch:

$$R_p = \frac{117.672}{2\pi * 2} * \left(\ln\left(\frac{2 * 2}{0.02}\right) + 0.5 \ln\left(\frac{4 * 1.5 + 2}{4 * 1.5 - 2}\right)\right) = 52.858\ Ohm$$

We find the number of vertical grounding conductors:

$$n = \frac{52.858}{5.66 * 0.78} = 11.97$$

We round this value up to twelve electrodes, and evenly distribute them along the contour at a distance of 40/12 = 3.33 m from each other.

This example shows the calculation of grounding resistance installed in favorable soils. For other types of soils, the calculation process does not differ, but the embodiment of the calculated contour becomes more expensive at times.

4.2.1 Plan of office

Working conditions are a set of factors of the working environment that affect human health and performance in the labor process. Working conditions should be comfortable and exclude the preconditions for the occurrence of injuries and occupational diseases.

The office is located on the 2nd floor of a five-story building, with windows facing the North and the West. The dimension of the room is 6x5 m; the total area of the room is 30m^2. The size of the windows is 4*1.80*1.50 = 10.8m; all windows start a meter from the floor.

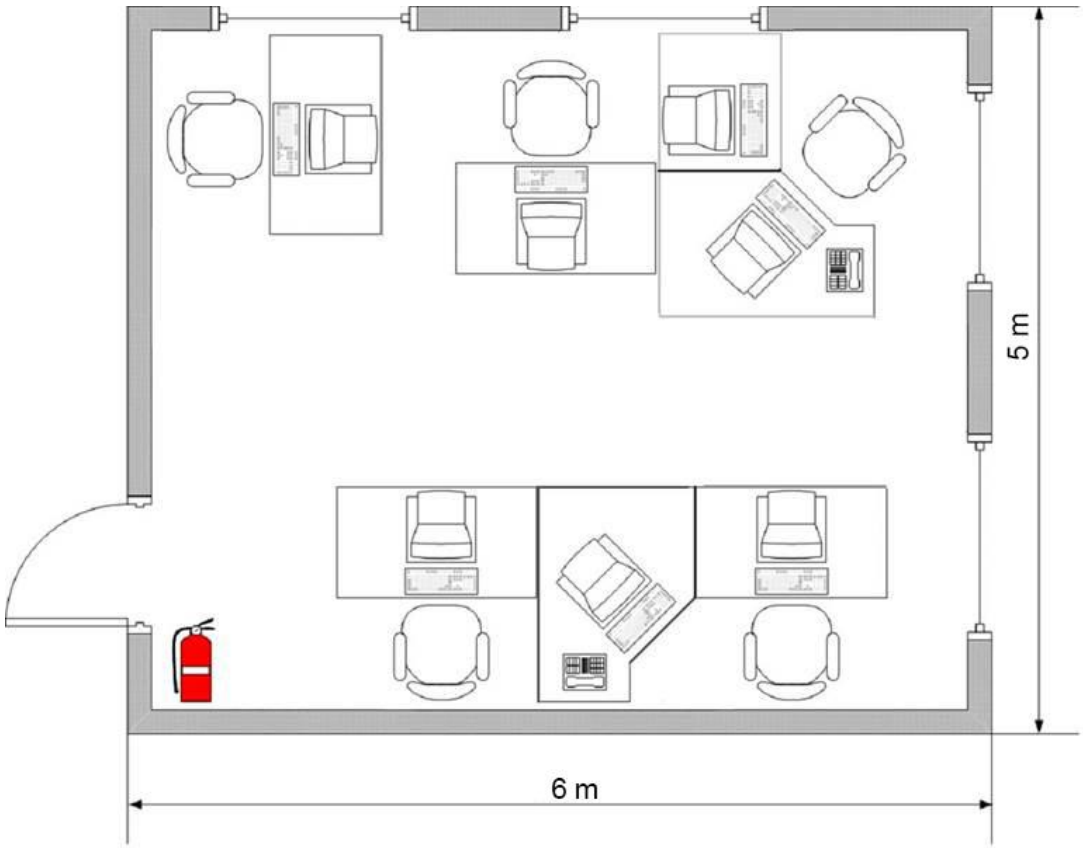

Figure 4.3 – Working room plan

In the office there are 5 workplaces of software engineers with all the adjacent equipment. Room area S = 6x5 = 30m^2, Volume - V_{contr} = 30x3 = 90m^3. Therefore, per person has an area of 30/5 = 6m^2 and a volume of 90/5 = 18m^3.

This is more than the minimum area and volume per employee set by the norms (volume - at least 15 m^3, area - at least 4.5 m^2).

Indoors are:

- 7 workstations;
- 3 laser printers;
- 2 scanners.

It is assumed that 5 employees will work in the room during the daytime: a system administrator, 3 programmers and a department head.

4.2.2 Calculation of artificial lighting using the spot method

Using this method, you need to remember that it is used for round-symmetrical lamps (for example, DRL). In this case, it is necessary to take the luminous flux of the lamp equal to 1000 lm. This is the so-called conditional illumination. This parameter depends on the following factors:

- light distribution of the lamp;
- its geometric dimensions: the installation height of the lighting installation, the distance from the lamp to the projection of the incidence of the light flux.

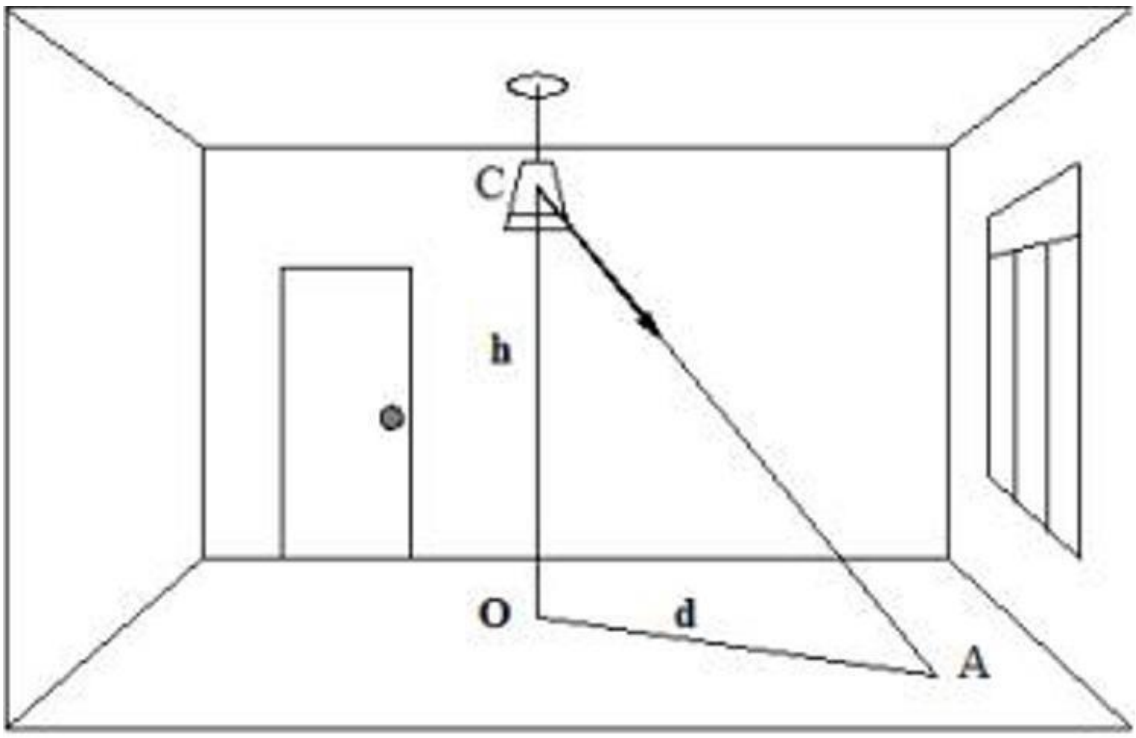

Figure 4.4 – Spot method

In this situation, the luminous flux from the lighting device is determined by the following formula: Φ \u003d 1000 · Eu · Cz / μ · ∑Eu, where:

- μ is a special coefficient by which the effect of "remote" lamps is taken into account;
- ∑Ey - conditional total illumination set for the control point;
- Ey - for a separate lamp.

According to the calculated luminous flux, one should choose bulbs, this parameter of which will be in the range of -10 ... + 20%.

For artificial lighting of the room should be used mainly fluorescent lamps. The most acceptable are white and warm – white fluorescent lamps.

According to spot method, for round-symmetric point emitters (incandescent lamps and mercury arc lamp), it is assumed that the luminous flux of the lamp (or the total luminous flux of the lamp) in each lamp is equal to 1000 lm. The illumination created by such a lamp is called conditional. The value of conditional illumination depends on the light distribution of the lamp and the geometric dimensions: the distance from the point to the projection of the illuminating lamp (α) and the height of the lamp above the level of the illuminated surface (h) [3].

Required:

Calculate the total lighting of the working room 6 m long, 5 m wide, 3.2 m high with a whitewashed ceiling, light walls with unfinished windows. Normalized illumination is given by instructor and equal to 300 Lux. The factor of safety. The height of the lamp suspension above the working surface $h = 3.2 - 0.75 - 0.25 = 2.2$ m. the illumination scheme is shown in figure 4.3.

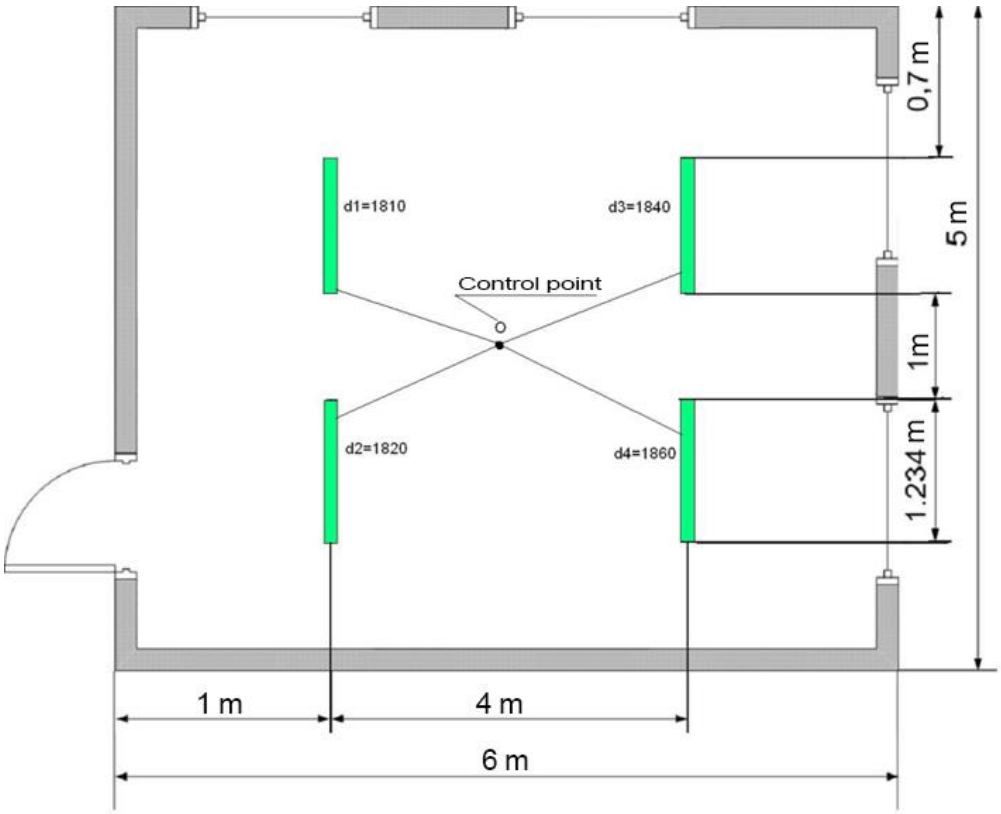

Figure 4.5 – Illumination calculation scheme

We outline the control point O. Find the distance from this point to each lamp and then, from the spatial isolux chart, we find the illumination of each lamp.

The first lamp:

d_1=1.810 m;

E_1=10 lx;

The second lamp:

d_2=1.820 m;

E_2=20 lx;

The third lamp:

d_3=1.840 m;

E_3=20 lx;

The forth lamp:

d_4=1.860 m;

E_4=15 lx;

The total illumination from all the lamps at the control point O:

$$\sum E_O = E_1 + E_2 + E_3 + E_4, \tag{4.1}$$

$$\sum E_O = 10 + 20 + 20 + 15 = 65 \text{ lx}.$$

The luminous flux is calculated by the formula:

$$\phi = \frac{1000 \cdot E \cdot k_Z}{\mu \cdot \sum E_O}, \qquad (4.2)$$

where μ is the coefficient that takes into account the action of "remote"/ enameled direct light lamps (1.1-1.2 is accepted);

E - normalized illumination of the room;

$$\phi = \frac{1000*300*1.5}{1.2*65} = \frac{450000}{78} = 5769 \text{ lm};$$

Consequently, for illumination of the workplace, general lighting is sufficient, in which the workplace is illuminated by two lamps, each lamp has two lamps with a luminous flux of radiation of 1442 lm each.

4.3 Conclusion of life safety part

The first chapter outlined the requirements for the workplace. These conditions should ensure the comfortable work of the engineer. Based on the materials studied, we have calculated of the optimal lighting of the production room and evacuation time. The point method makes it possible to determine the illumination at any point in the room. It allows you to calculate the luminous flux of a light source. You can also estimate the actual light distribution. The calculation of the evacuation time is carried out in order to determine the time values of the onset of fire hazards. This takes into account the size and layout of buildings. The timing is necessary so that people can freely leave the danger zone in the event of a fire.

To increase the working capacity and labor productivity, compliance with the conditions have great importance on the organization of the workplace. All these activities play an important role in maintaining health.

5 Economic part

5.1 Calculation of costs for the development of hardware and software

The determination of the costs for the development of software is made by drawing up the corresponding estimates, which includes the following articles:
1) Material costs.
2) Labor costs.
3) Social tax.
4) Depreciation of fixed assets.
5) Other costs.

5.2 Material cost

The article “Material costs” includes the costs of basic and auxiliary materials (paper, cartridges and others), energy necessary for the development of software. Calculation of costs for material resources is made in the form given in table 5.1.

Table 5.1 - the Cost of material resources

Name of material	Unit Measuring	Amount of material used	Price per unit, tg	Amount, tg
Argon + Wi-fi antenna	number	1	15000	15000
Xenon	number	1	5000	5000
Temp sensor AM2302	number	1	4400	4400
Breadboard	number	1	900	900
Usb cable	number	2	2000	2000
Wire	number	10	100	1000
Power bank	number	1	9000	9000
TOTAL material costs				37300

The total cost of material resources (ZM) is determined by the formula:

$$T_M = \sum_{i=1}^{n} N_i * P_i, \quad (5.1)$$

Where:
- N_i is the consumption of the i-th type of material resource, natural units;
- P_i is the price per unit of the i-th type of material resource, tg;
- i - type of material resource;
- n is the number of types of material resources.

If electrical equipment is used for the development of PP, then it is necessary to calculate the cost of electricity in the form shown in table 5.2.

Table 5.2 - the cost of electricity

Name of equipment	Nameplate power, kW	Power factor	Operating time of equipment for software development, h	The price of electro energy, tg / kW	Amount, tg
Argon	1,65	0,8	5	16	105,6
Xenon	1,65	0,8	5	16	105,6
Laptop Hp	65	0,2	5	16	1040
Power bank	38,5	0,2	1	16	123,2
TOTAL electricity costs					1374,4

The total cost of electricity (EE) is calculated by the formula:

$$T_E = \sum_{i=1}^{n} M_i * K_i * T_i * P, \quad (5.2)$$

Where:

- Mi is the rated power of the i-th electrical equipment, kW;
- Ki - power utilization factor of the i-th electrical equipment (assumed Ki = 0.7¸0.9);
- Ti - the operating time of the i-th equipment for the entire period of development of the software;
- P is the price of electricity, tg / kW × h;
- i - type of electrical equipment;
- n is the number of electrical equipment.

The article “Labor costs” includes the labor costs of all employees involved in the development of postgraduate education (graduate student, managers and consultants of diploma work, involved persons).

5.3 Labor costs

Labor costs are calculated according to the form given in table 5.3.

Table 5.3 - the Cost of labor

Employee category	Qualification	The complexity of the development of software, people × h	Hourly rate, tg / h	Amount, tg
Developer	Hardware	5	1000	5000
Developer	Software	5	2000	10000
TOTAL labor costs				15000

The total cost of labor (T_{LC}) is determined by the formula:

$$T_{LC} = \sum_{i=1}^{n} Ph_i * T_i, \quad (5.3)$$

Where:

- Ph_i - hourly rate of the i-th employee, tg;
- T_i - the complexity of the development of software, people × h;
- i - employee category;
- n is the number of workers involved in the development of software.

The hourly rate of the employee can be calculated by the formula:

$$Ph = \frac{MS}{MHR}$$

Where:

- MS - monthly salary of the i-th employee, tg;
- MHR - monthly fund of working hours of the i-th employee, hour.

The complexity of the development of software is determined according to table 5.4.

Table 5.4 - Distribution of work by stages and types and assessment of their complexity

PP development stage	Type of work at this stage	The complexity of the development of software, people × h
Hardware	Connection and setup	5
Software	Writing codes	5
TOTAL complexity of thesis		10

5.4 Social tax

The article “Social tax” includes an amount that is calculated as 9.5% of the cost of labor for all employees (T_{LC}) involved in the development of software. When

calculating, it must be taken into account that pension contributions (10% of T_{LC}) are not subject to social tax.

$$S_T = 0.11 * T_{LC} = 0.095 * 15000 = 1425$$

5.5 Depreciation of fixed assets

The article "Depreciation of fixed assets" includes the amount of depreciation deductions from the cost of equipment and software (software) used in the development of software. Depreciation charges are calculated according to the form given in table 5.5.

Table 5.5 - Depreciation of fixed assets (FA)

Name of hardware and software	The cost of hardware and software, tg	Annual depreciation rate,%	Effective fund of operating time of hardware and software, h / year	Operating time of equipment and software for software development, h	Amount, tg
Mesh network	40000	10	5	5	4000
TOTAL depreciation of fixed assets					4000

The total amount of depreciation is determined by the formula:

$$T_{DFA} = \sum_{i=1}^{n} \frac{F_i * ADR_i * OT_i}{100 * EF_i}, \quad (5.5)$$

Where:

- F_i is the cost of the i-th FA asset, tg;
- ADR_i is the annual depreciation rate of the i-th asset,%;
- OT_i - operating time of the i-th FA for the entire period of development of software, h;
- EF_i - an effective fund of the operating time of the i-th FA for the year, h / year;
- i - type of formatting object;
- n is the number of FA.

When determining the value of FA, it is also necessary to take into account the costs of delivery and installation, software installation. These costs can be taken in the amount of 10-25% of the cost of acquiring FA.

The annual amortization rates of public funds are adopted according to the tax code of the Republic of Kazakhstan or are determined based on the possible useful life of the public funds:

$$ADR_i = \frac{100}{TN_i}, \quad (5.6)$$

Where:

- TN_i is the possible period of use of the i-th FA, year;

The possible useful life of the FA can be taken from 3 to 10 years (in agreement with the consultant for the economic part).

5.6 Final calculation

Based on the data obtained for individual articles, an estimate is made of the costs of developing software in the form shown in table 5.6.

Table 5.6 - Estimated costs for the development of software

Expenditures	Amount, tg
1. Material costs, including:	
materials	37300
electricity	1374
2. The cost of labor.	15000
3. Deductions for social needs.	1425
4. Depreciation of fixed assets.	4000
5. Other costs.	0
TOTAL by estimate	59099

5.7 Conclusion of economic part

In this economic part of my diploma work I have made calculation of costs for developing and researching a IoT mesh network using Argon and Xenon. First of all, I have calculated the material cost of my project. Secondly, I estimated cost for electric power, because my hardware and software are using electric power. Also, I used Laptop Hp to program the Argon and Xenon. After, I calculated cost for programmer workers. Then, I calculated depreciation of fixed assets for my project. Finally, I have summed all these costs and get my final cost of project that was almost 60 thousand tg at all.

Conclusion

Using mesh network for IoT devices is a great way to gather important data in a very cost effective way. However, these solution works best for small packages that contain logs, diagnostic data etc.

It's also a great in between of bluetooth and Internet. It can connect to nodes placed a lot further than it would have been using bluetooth. At the same time, it uses very little energy because it doesn't need constant connection to the Internet.

'When bluetooth is not enough and Internet is too much go for the mesh network'. However, some solutions simply require constant connection to the Internet or contain more data than mesh network is able to transfer.

In this research paper, we have analyzed several approaches that can be adopted in scenarios in which mesh networking is attractive/necessary (e.g., IoT-oriented scenarios). On top of this, we have discussed on how mesh networking can be carried out through different wireless technologies, ranging from IEEE 802.11 to Bluetooth (and its lightweight Bluetooth Low Energy (BLE) version), IEEE 802.15.4-oriented protocols (such as Thread, LWMesh, and ZigBee Pro and ZigBee 3.0), till Sub-GHz-based LoRa protocol. We have outlined how some protocols were born with a native support for mesh networking (i.e., being specifically tailored for mesh scenarios), while other communication protocols have been extended to support this type of networking. An interesting conclusion from our survey is that the considered technologies, being either standard as well as proprietary, well adapt to scenarios in which the heterogeneity of devices composing the mesh network is a must, such as IoT environments. In this case, a "network of networks" is the best definition that can be used and that better represents the variety of communications, ranging from short-range to long-range, that can be found in modern scenarios—such as in smart city and smart agriculture scenarios.

As described in this project, mesh network have all these great features using for the communication in the IoT networks. It is still under-developed as the industry advancing today. With the much more powerful MCU and processors today, the dynamic network topology can be achieved even in the tiny IoT devices. The mesh network topology has its unique advantage and disadvantage in the world of the IoT networks that can leverage the scale, distributed nature and low require of data-rate of the IoT devices. The advantage certainly outweighs the disadvantages of using the WMNs in such environment. Newer hybrid WMNs can be a solid choice when it comes to design the structure of the network, especially in the remote areas with its the robustness and scalability. This paper is simply discussing the possibility and the basic way of integrating such under-utilized network topology into the current and future IoT networks in the background of the advanced technology we have today. WMNs will certainly make a difference in the industry once being deployed on large scale in the IoT world and make the IoT more accessible to a wider audience.

List of reference

1. X. W. Ian Akyildiz, Wireless Mesh Networks, London: John Wiley & Sons, 2009.

2. J. Okin, The Internet Revolution: The Not-for-Dummies Guide to the History, Technology, and Use of the Internet, 1 ed., Winter Harbor, ME: Ironbound Press, 2005.

3. P. M. M.-S., a. L. E. M. Elizabeth M. Royer, “An Analysis of the Optimum Node Density for Ad hoc Mobile Networks,” 2001.

4. Internet of Things (IoT) Lab, Department of Engineering and Architecture, University of Parma, Parco Area delle Scienze 181/A, 43124 Parma, Italy; luca.davoli@unipr.it (L.D.); laura.belli@unipr.it (L.B.);

5. Johnson, D.; Hu, Y.; Maltz, D. The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4. RFC 4728, Internet Engineering Task Force, 2007. Available online: https://tools.ietf.org/html/rfc4728 (accessed on 15 April 2019).

6. Perkins, C.; Belding-Royer, E.; Das, S. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561, Internet Engineering Task Force, 2003. Available online: https://tools.ietf.org/html/rfc3561 (accessed on 15 April 2019).

7. Clausen, T.; Jacquet, P. Optimized Link State Routing Protocol (OLSR). RFC 3626, Internet Engineering Task Force, 2003. Available online: https://tools.ietf.org/html/rfc3626 (accessed on 15 April 2019).

8. Institute of Electrical and Electronics Engineers (IEEE). IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks–Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications; IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012); IEEE: Piscataway, NJ, USA, 2016; pp. 1–3534.

9. Institute of Electrical and Electronics Engineers (IEEE). IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems–Local and Metropolitan Area Networks–Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 10: Mesh Networking; IEEE Std 802.11s-2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1–372.

10. Общие требования к построению, изложению, оформлению и содержанию учебно-методических и учебных работ. СТ НАО 56023-1910-04-2014.

11. Wireless Standards for IoT: WiFi, BLE, SigFox, NB-IoT and LoRa. Available on: http://wireless.ictp.it/school_2017/Slides/IoTWirelessStandards.pdf (accessed on 27 March 2019).

12. Diez, M. Secure Position Data Transmission for Object Tracking using https://files.ifi.uzh.ch/CSG/staff/schmitt/Extern/Theses/Matthias_Diez_MA.pdf.

13. Petrosky, E.E.; Michaels, A.J.; Ridge, D.B. Network Scalability Comparison of IEEE 802.15.4 and Receiver-Assigned CDMA. IEEE Internet Things J. 2019.

14. Particle Device Cloud: https://console.particle.io/

15. Argon datasheet: https://docs.particle.io/assets/pdfs/datasheets/argon-datasheet.pdf

16. Xenon datasheet: https://docs.particle.io/datasheets/discontinued/xenon-datasheet/

17. Application notes: https://docs.particle.io/datasheets/app-notes/application-notes/

18. Adafruit products AM2302 temperature sensor: https://www.adafruit.com/product/393

Appendix A

```
#include "Particle.h"
SYSTEM_THREAD(ENABLED);
SYSTEM_MODE(SEMI_AUTOMATIC);
void setup() {
Serial.begin(115200);
pinMode(D7, OUTPUT);
pinMode(D2, INPUT);
// Mesh ON but not connected, no cloud at this point (status LED is ON)
delay(5000);
digitalWrite(D7, HIGH);
delay(1000);
digitalWrite(D7, LOW);
// Mesh, no cloud at this point (status LED is ON)
Mesh.connect();
waitFor(Mesh.ready, 10000);
//Particle.connect();
//waitFor(Particle.connected, 10000);
delay(10000);
digitalWrite(D7, HIGH);
delay(1000);
digitalWrite(D7, LOW);
// Mesh OFF, no cloud at this point (status LED is ON)
Mesh.disconnect();
Mesh.off();
delay(10000);
digitalWrite(D7, HIGH);
delay(1000);
digitalWrite(D7, LOW);
delay(50);
// Mesh OFF, no cloud, going to STOP sleep
System.sleep(D2, RISING, 30);
delay(1000);
digitalWrite(D7, HIGH);
delay(1000);
digitalWrite(D7, LOW);
delay(50);
// Mesh OFF, no cloud, going to STANDBY sleep forever
System.sleep(SLEEP_MODE_DEEP);
}
void loop() {}
```

Acceptance letters

В ответ на: Re: Экономическая часть дипломной работы группа ПСа-16-3 Входящие ×

bakbergen tuzelbaev 3 мая 2020 г., 11:33
кому: я

О.К.

Отправлено из Yahoo Почты для Android

вс, 3 мая 2020 в 0:09 Nursultan Nazarbay <nazarbay.ktl@gmail.com> написал(-а):

сб, 2 мая 2020 г. в 23:48, bakbergen tuzelbaev <bakbergen@yahoo.com>:
Социальный налог 9,5%, пересчитай и бери более высокую цену на электроэнергию

On Saturday, May 2, 2020, 10:51:14 PM GMT+6, Nursultan Nazarbay <nazarbay.ktl@gmail.com> wrote:

Назарбай Нурсултан

Ответить Переслать

Economic part acceptance letter

Nursultan Nazarbay вс, 3 мая, 23:46
Привел вс, 3 мая 2020 г. в 12:54, Ainur <ainur.begimbetova@mail.ru>:

Ainur вт, 5 мая, 19:56
кому: я

английский > русский Перевести сообщение Отключить для языка: английский ×

work accepted

--

Ainur

Воскресенье, 3 мая 2020, 23:46 +06:00 от Nursultan Nazarbay <nazarbay.ktl@gmail.com>:
...

Ответить Переслать

Life safety part acceptance letter

Diploma project tech part Входящие ×

Nursultan Nazarbay 17:57 (5 часов назад)

Ayazbai Abu-Alim <a.ayazbay@aues.kz> 23:50 (0 минут назад)
кому: я

Назарбай Н. допущен к защите.

From: Nursultan Nazarbay
Sent: вторник, 9 июня 2020 г. 17:57
To: a.ayazbay@aues.kz
Subject: Diploma project tech part

Алматинский университет энергетики и связи имени Гумарбека Даукеева
050013, г. Алматы, Байтурсынулы 126/1
тел. +7 727 323 11 75
aues.kz

Tech part acceptance letter