Ukrainian Catholic University

Faculty of Applied Sciences

Computer Science Programmes

A central module of communication for a network of sensors (HUB)

Architecture of the Computer System project report

Authors:
Maksym Mykhasiuta
Bohdan Pelekh
Fedir Zhurba

Supervisor:
Markian Matsiuk

01 June 2023

1 Introduction

The project's main idea is to create a distributed network of sensors that will collect and transmit information to a single hub, which in turn will transmit the data to a server where it will be available to users.

We decided to develop a whole hierarchy of modules, each class of which is designed for its own purpose.

- First level module. A module that, with the help of sensors, collects the necessary information and passes it on.
- Second level module. A module that receives information from the first-level modules. At this hierarchy level, a mesh system is also implemented, allowing you to efficiently distribute information between modules and deliver data from a greater distance than the basic transmission protocol is available.
- Hub. The central module receives information from level 2 modules and sends it to the server via the Internet.

2 Transmitting Protocols

In the context of our project, we have carefully chosen two distinct modes of interaction between modules: a radio protocol for facilitating communication between level 1 modules and level 2 modules, and a WiFi protocol for enabling communication between level 2 modules, the hub, and the server.

2.1 Radio Protocol

The selection of the radio protocol was based on several advantageous features it possesses when compared to alternative protocols such as WiFi and Bluetooth.

- The radio modules are cheaper than their counterparts, making them a more cost-effective choice for our system.
- The radio protocol demonstrates superior energy efficiency, consuming less power than competing protocols. This results in improved overall energy efficiency for our system.
- Incorporating radio modules provided an intriguing challenge during the development process, adding an exciting aspect to the project.

2.2 WiFi Protocol

The WiFi protocol has been chosen as the primary foundation for establishing a mesh system for second-level sensors in our project. Additionally, this protocol is crucial in facilitating communication between the hub and the server over the Internet.

3 Hardware equipment

3.1 Radio modules

- NRF24L01

  Those modules are used at first and second-level modules for radio transmitting and receiving.

- Breakout board for NRF24.

  NRF24 modules are extremely sensitive to voltage drops and require a stable power supply. According to this reason using a breakout board is crucial.

3.2 Micro-controllers

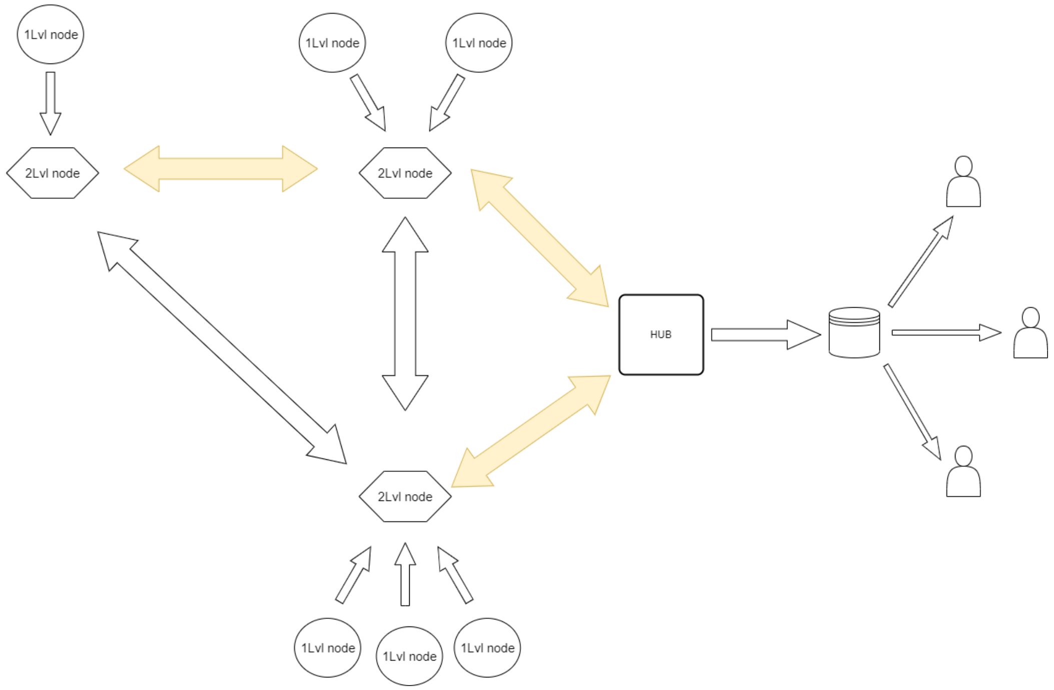
- ESP8266 - used for second-level modules and HUB.
- Arduino Nano - used for first-level modules.

3.3 Miscellaneous

- DHT11 - temperature and humidity sensor
- 5-12v to 3.3-5v DC-DC step-down power adapter
- Power sources - sets of AA batteries, Li-Ion batteries

4 Data streams

Here is the data streams scheme:



Each second-level module is paired with multiple first-level nodes that transmit sensors' data to them. The number of first-level nodes can vary for each second-level node and is determined solely through prior planning. This communication mechanism is established through the utilization of radio communication technology.

Moving forward, let's consider the implementation of a mesh system within the second level of the module hierarchy. Each node within this mesh system is interconnected, enabling direct communication among all nodes. The system automatically selects the most efficient path for data transmission. In case of a node failure, the data transmission path dynamically changes to an alternative route. This seamless communication is achieved using WiFi technology.

During the processing we need to develop an efficient way to distinguish received data and determine the source of each observation (sensor of the first level). The signal may achieve the final destination (hub) by distinct paths, due to a mesh system, where sensors of the second level form their own optimal network. We decided to manage it using a unique identifier for each sensor and assigning a unique sensor of the second level to the group of first-level sensors.

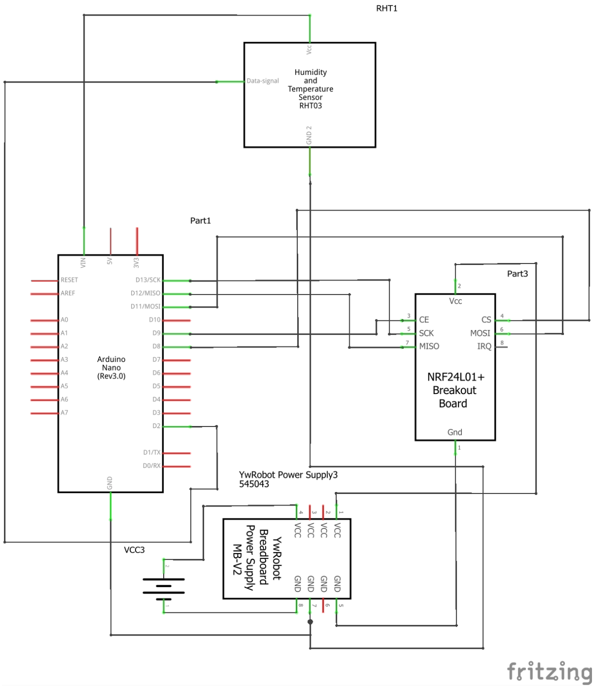
Ultimately, the data reaches the hub, which is a central point for data aggregation. The hub then transmits the collected data to an Internet server, where it can be made accessible and presented to users.

5 Guideline

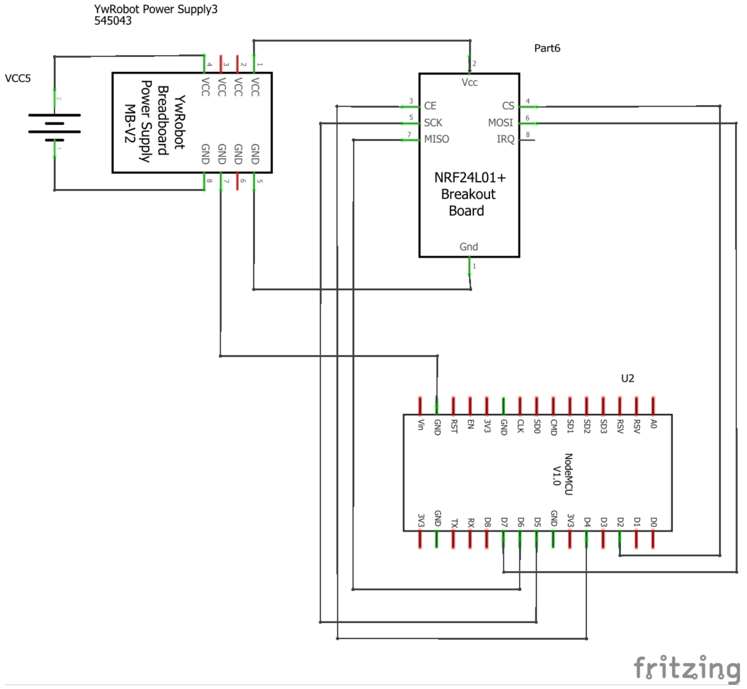
5.1 Connection scheme

Hardware requirements were specified in the Hardware equipment section.

First-level node scheme:



Second-level node scheme:



As we can see, the general scheme of connection to the power supply is similar for both modules. We connect the battery to the breakout board. This initial step is required due to the reasons described upper. Then we supplied the ESP module with stable power.

In modules of the first level, we connect the ESP module to Arduino. Arduino is connected to a temperature and humidity sensor. Processed data is passed to ESP, which transfers data to other ESP with radio protocol.

ESP in the second-level node receives a radio signal with the required values and transfers it to NodeMCU, which is connected to other nodes with WiFi mesh. This way, data reaches our hub (bridge) and is transferred to the server.

5.2 Libraries

5.2.1 painlessMesh

Mesh is implemented for second-level modules using the painlessMesh library[2]. Our mesh network is self-healing, which means that removing a node would not cause a failure of the whole system. Maps of data flow are dynamically reordered, allowing us to get a new road for the sensor's data to reach the bridge. painlessMesh is a true ad-hoc network, meaning that no-planning, central controller, or router is required. Any system of one or more nodes will self-organize into a fully functional mesh.

This library does not use TCP/IP protocol for nodes identification. Instead, each node receives a 32-bit unique identifier. This way, nodes communicate using this id. The node can send messages to all available nodes or to the set of nodes, determined by their identifiers.

Information is transferred using JSON format, which makes messages human-readable, allows to transfer messages to the server directly, and makes data processing easier.

Important limitation: do not use delays in your code, they can stop the routine of the library and may break the network or lead to using longer paths.

5.2.2 RF24

The RF24 library simplifies the process of setting up and managing communication between modules. It provides a high-level interface with intuitive functions and abstractions, making it easier for developers to work with the radio modules and build wireless applications.

First-level modules communicate with second-level modules using RF24 and God's bless. RF24 library provides a convenient way to work with radio communication protocol. What is more, it allows to create a multi-node system, enabling support of several modules interface.

6 Conclusion

In conclusion, this project successfully developed a central module of communication for a sensor network using mesh technology. The implementation of mesh networking principles resulted in a robust, scalable, and fault-tolerant communication system. The project opens up great opportunities for further improvement and modernization. The example is the implementation of our own mesh library based on the network used in work. One more step to proceed with project development is further scaling of the system, which involves adding more types of sensors and more advanced system of module layers.

References

[1] nRF24. RF24 Library Documentation. Retrieved from https://nrf24.github.io/RF24

[2] painlessMesh. painlessMesh Documentation. Retrieved from https://gitlab.com/painlessMesh/painlessMesh

[3] esp8266. esp8266 Arduino code Documentation. Retrieved from https://arduino-esp8266.readthedocs.io/en/latest

[4] esp8266. Getting started with ESP-MESH and ESP-8266 Retrieved from https://www.instructables.com/Getting-Started-With-ESP-MESH-and-ESP-8266/