UNIVERSITETET I AGDER

REAl-LIFE EXPERIMENTS BASED ON IQRF IoT TESTBED: FROM SENSORS TO CLOUD

RIDWAN BIN ZAFAR

SUPERVISORS

Dr. Indika Anuradha Mendis Balapuwaduge

Prof. Dr. Frank Yong Li

University of Agder, 2018

Faculty of Engineering and Science

Department of ICT

Real-Life Experiments based on IQRF IoT Testbed: From Sensors to Cloud

By

Ridwan Bin Zafar

Supervisors: Dr. Indika Anuradha Mendis Balapuwaduge &
Prof. Dr. Frank Y. Li

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Science in Information and Communication Technology

Department of Information and Communication Technology
Faculty of Engineering and Science
University of Agder

Grimstad, Norway

June 04, 2018

Abstract

Internet of things (IoT) is the next generation internet technology which connects devices and objects intelligently to control data collected by diverse types of sensors, radio frequency identification and other physical objects. To address the challenges in IoT such as integrating artificial intelligent techniques with IoT concept, developing green IoT technologies and combining IoT and cloud computing, various platforms which support reliable and low power wireless connectivity are required.

IQRF is a recently developed platform for wireless connectivity. It provides low power, low speed, reliable and easy to use wireless connectivity in sub-GHz bands for telemetry and industrial control and building automation. The applications are used for scientific knowledge in practical purposes, especially in industrial usage. It is extensively non-identical from what we are accustomed to nowadays. It has been designed to use IQMESH technology. The purpose for IQMESH is to ensure wireless connectivity wherever it is necessary, covering tens of hundreds of meters up to several kilometers.

In this master's thesis, we implemented an IQRF platform-based testbed for IoT related application scenarios which can measure various environmental conditions and perform required communications. The research work of this thesis is carried out in three directions, i.e., single-hop and multi-hop communication for temperature measurements, light intensity measurements via integrated light dependent resistors and cloud-based connectivity for IQRF sensors.

The work performed in this thesis provides a valuable result because IQRF does not only solve problems regarding wireless technology, but it also supports industrial control, remote control and cloud control as well. Also, as the devices come with different input/output pins, they are easily maneuverable to connect different external sensors to get accurate results. In this thesis, the employed research methods, experiments, readings and related discussions have been presented accordingly.

Key words: *IQRF, IoT, Wireless Sensor Network, Light Dependent Resistor, GSM Gateway, IEEE 802.15.4, IQRF Cloud*

Preface

This Master thesis is a result of the research work carried out at the Department of Information and Communications Technology (ICT), University of Agder (UiA), in Grimstad, Norway, from 7 January 2018 to 4 June 2018. The workload is equivalent to 30 ECTS.

Dr. Indika A. M. Balapuwaduge, UiA has been the main supervisor of this MSc. thesis work and Professor Dr. Frank Y. Li, UiA, has been the co-supervisor. I would like to express my gratitude to them for their invaluable and profound guidance throughout the thesis period. Frequent meetings and regular discussions have been the foundation on which completeness of this thesis work is built. I wish to thank everyone who has directly or indirectly motivated me for the successful completion of this thesis. Their guidance, care, and most especially, ability to share knowledge with us are priceless. Finally, I thank my families, and friends for their very special place in our lives. My classmates have been so fantastic and thank you for all the nice moments together.

Production note: Microsoft word has been adopted as the tool for writing this Master thesis. For experiments, IQRF IDE version 4.45 has been used.

Ridwan Bin Zafar

Grimstad, Norway.

June 04, 2018

Table of Contents

List of Figures

List of Tables

Abbreviations

3G	Third Generation
6LoWPAN	IPv6 over Low-Power Wireless Personal Area Networks
ABB	ASEA Brown Boveri
ADC	Analog to Digital Conversion
AoA	Angle of Arrival
APN	Access Point Name
Auto-ID	Automatic Identification
CO_2	Carbon Dioxide
CPU	Central Processing Unit
CSMA-CA	Carrier-sense Multiple Access with Collision Avoidance
D2D	Device-to-Device
DNA	Deoxyribonucleic Acid
DPA	Direct Peripheral Access
ED	Energy Detection
EEPROM	Electrically Erasable Programmable Read-Only Memory
FRC	Fast Response Command
GSM	Global System for Mobile Communication
GUI	Graphical User Interface
HART	Highway Addressable Remote Transducer Protocol
HCF	HART Communications Foundation
HWPID	Hardware Profile Identification
I/O	Input/output
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoT	Internet of Things
IPv6	Internet Protocol Version 6
ISM	Industrial, Scientific and Medical

LAN	Local Area Network
LDO	Low-dropout
LDR	Light Dependent Resistor
LED	Light Emitting Diode
LP	Low Power
LPWAN	Low-power Wide Area Network
LQI	Link Quality Indicator
MIT	Massachusetts Institute of Technology
NFC	Near-field Communication
OS	Operating System
P2P	Peer-to-Peer
PAN	Personal Area Network
PC	Personal Computer
QR	Quick Response
RF	Radio Frequency
RFID	Radio Frequency Indication
RSSI	Received Signal Strength Indicator
SDK	Software Development Kit
SIM	Subscriber Identity Module
SPI	Serial Peripheral Interface
TSMP	Time Synchronized Mesh Protocol
TSP	Telecommunication and Signal Processing
USB	Universal Serial Bus
VRN	Virtual Routing Number
Wi-Fi	Wireless Fidelity
WPAN	Wireless Personal Area Network
WSN	Wireless Sensor Network
XLP	Extra Low Power

CHAPTER 1

INTRODUCTION

In this chapter an introduction of intelligent connectivity using radio frequency (IQRF) is given along with, what to expect from IQRF devices and how does it work, how the communication has been made among the nodes and the controllers, how it routes the packets, the operation of IQRF, making the connectivity among the devices, how the input/output (I/O) pins in the modules can be controlled, how to use the devices through cloud server and so on.

The results which have been obtained through the experimental tests are very much widespread. Even though IQRF is an independent device, containing its own power source and long-time battery life, it also contains a lot of limitations. The coding has been done in JAVA format, but it is not open source. The pre-made coded files are highly encrypted by company policy but they contain absolutely no glitches and bugs. General people are not acknowledged about IQRF's existence. After all, with its limitations, IQRF can be a very bright solution for low budget wireless connectivity, wherever it is needed.

1.1 Wireless Sensor Networks

When a group of wireless sensor devices form a network and cooperate to achieve a common goal, it is known as wireless sensor networks (WSNs). WSNs mainly uses ad-hoc networking because they must rely on wireless connectivity and impulsive formation of networks so that the sensed data can be transported wirelessly. These networks are also called *dust network*. The whole network consists of one or several base stations. They gain the data captured by the sensors.

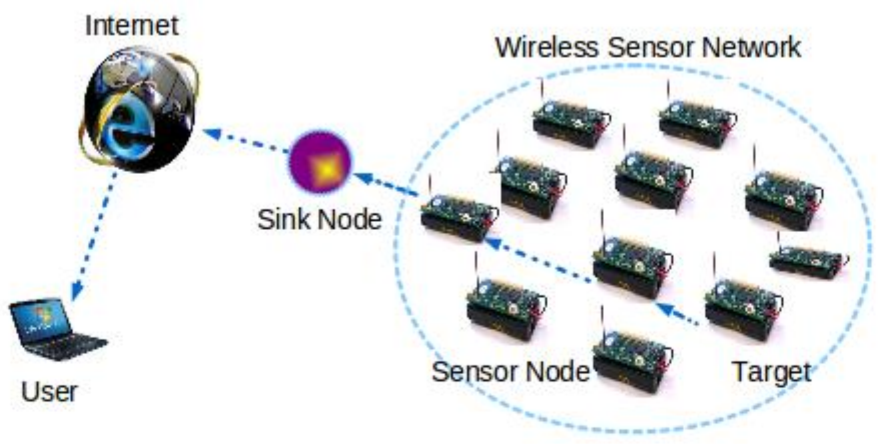


Figure 1.1: Wireless sensor network [1].

Basically, the sensors are deployed in several places, and they are used to monitor and record the physical conditions of the environment like temperature, sound, pollution levels, humidity, wind and so on. These data then organized and stored at the base station or central station.

A WSN is built of several hundreds or even thousands of nodes. Here, each node is connected to one or sometimes several other nodes. These nodes usually have several parts like a radio transceiver equipped with an internal antenna or the connection to an external antenna, a microcontroller unit, an energy source which is usually a rechargeable battery and an electronic circuit for interfacing them with the sensors and personal computer (PC). These nodes might vary in size from that of a shoebox down to the size of a grain of dust. The cost of sensor nodes is variable, ranging from a few to hundreds of dollars, depending on the complexity of the individual sensor nodes. The sizes and cost limitations on these nodes result in limitations on resources such as energy, memory, computational speed and communications bandwidth. The topology of the WSNs can vary from a simple star network to an advanced multi-hop wireless mesh network. The propagation technique between the hops of the network can be routing or flooding [2].

1.2 IEEE 802.15.4 based WSN Standards

Institute of electrical and electronics engineers (IEEE) 802.15.4 standard was initialized for wireless personal area networks (WPANs). WPANS uses the nodes inside a network to carry information over short distances. With this standard, it is possible to implement solutions for small, power efficient, inexpensive and a wide range of applications and device types. Some key features of 802.15.4 [3] are given below:

• Data rate of 250 kbps over the air in the 2.4 GHz, industrial, scientific and medical (ISM) band.

• In 2.4 GHz band, 16 independent communication channels.

• Large networks with connectivity of up to 65534 devices.

• For accessing the medium, devices use carrier sense multiple access with collision avoidance (CSMA-CA).

• Devices use energy detection (ED) [4] for channel selection.

• With link quality indication (LQI), devices contain the ability to inform the application about the quality of the wireless link.

There are two network topologies defined in the 802.15.4 standard, in which both topologies use one and only one central device which is known as the personal area network (PAN) coordinator. The PAN coordinator is the principal controller of the network. Nowadays low-power wide area network (LPWAN) is very popular. LPWAN is a wireless wide area network technology that interconnects low-bandwidth, battery-powered devices with low bit rates over long ranges [5].

• Star Network Topology — In such networks, all communication either to or from the PAN coordinator. That means communication between non-PAN coordinator devices is not possible in this topology. Fig. 1.2 shows a star network topology.

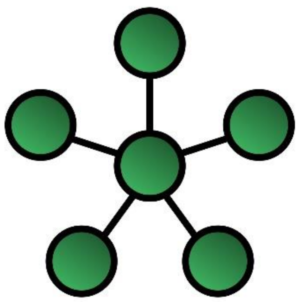


Figure 1.2: Star network topology [6].

• Peer-to-Peer (P2P) Network Topology — In a peer-to-peer network, communication is possible between any two devices in the network if they are within range of one another. Fig. 1.3 demonstrates a peer-to-peer network topology.

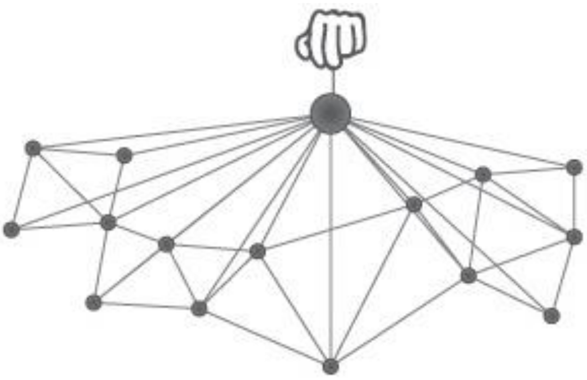


Figure 1.3: Peer-to-peer network topology [7].

For a device joining an 802.15.4 network, it must subordinate with a device that is already part of the network. In a chance, it allows other devices to associate with it. Multiple devices can be associated with the same device. The device which has other devices associated with it is considered as the coordinator. Through beacon frames, a coordinator provides synchronization services to all the devices that are associated with it. In a star network there will be only one PAN coordinator, but in a peer-to-peer network, there can be multiple coordinators plus the PAN coordinator.

A network, either star or peer-to-peer can operate in either beacon mode or non-beacon mode. In beacon mode, all coordinators within the network transmit synchronization frames to the connected devices. All the data transmissions between the coordinator and its connected devices happen in the active period following the beacon frame as shown in Fig. 1.4.

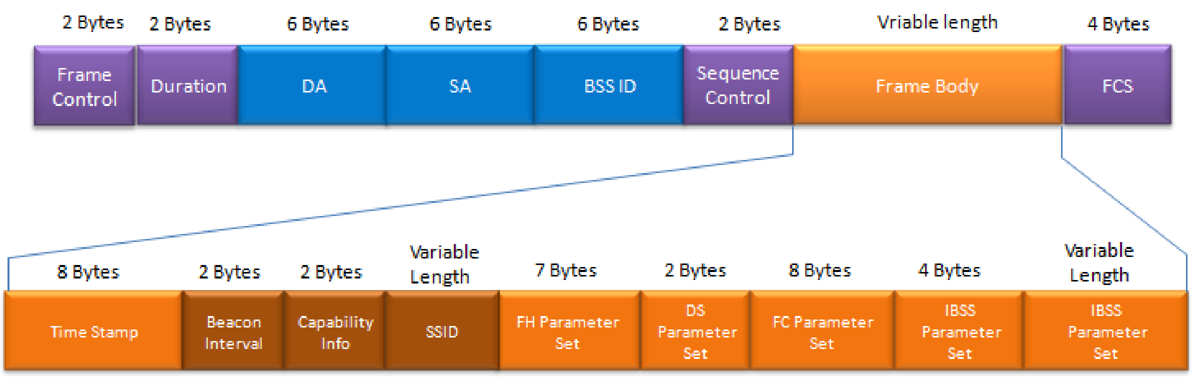


Figure 1.4: Beacon frame structure [8].

1.2.1 ZigBee

One of the communication protocol for wireless personal data networks (WPAN) is Zigbee [9] [10]. Very little consumption of power is one of the major advantages of using this protocol. For devices that are battery powered and sends data continuously, this protocol is very much suitable. This protocol is appropriate for remote controls, alarms, sensors, access controllers, automation, etc. These must be a base station or centralized system for these types of devices which they will report to. The data packets must be very small for these devices. In remote controls of Xbox and PlayStation, ZigBee is used and used widely as a communication protocol in different WSNs.

ZigBee is based on the IEEE 802.15.4 standard and has a range of up to 100m. According to the specification, the speed for ZigBee for 78m is specified to a maximum of 250 kbps data rate. Networking between ZigBee nodes can be created easily and just like IQRF, there can be as many as 65,000 nodes in the network. In Europe, ZigBee uses 16 channels in the ISM frequency band 2.4 GHz, and 1 channel at 20 kbps at 868 MHz and in the US, 915 MHz is used as well as 2.4 GHz. The communication between ZigBee nodes takes only 30ms, compared to 3s with Bluetooth. Fig. 1.5 shows a series-2 Zigbee.

Figure 1.5: Zigbee series-2 [11].

Figure 1.6: Wireless HART [12].

1.2.2 Wireless HART

Another overwhelming technology for WSNs is Wireless HART. It is based on the highway addressable remote transducer protocol (HART) [13]. The HART protocol is basically a hybrid analog + digital protocol for industrial automation. The protocol was developed as a multi-vendor, interoperable wireless standard. Wireless HART was defined for the requirements of process field device networks. The main advantage of this technology is that it can communicate over legacy 4-20 mA analog instrumentation current loops. It can also share the pair of wiring that has been used by the analog host system. Fig. 1.6 shows a Wireless HART.

The protocol utilizes a time synchronized, self-organizing, and self-healing mesh architecture. The protocol supports operation in the 2.4 GHz ISM band using IEEE 802.15.4 standard radios. The underlying wireless technology is based on the work of *dust networks'* time synchronized mesh protocol (TSMP) technology. The standard was initiated in early 2004 and developed by 37 HART communications foundation (HCF) companies including ASEA Bwown Boveri (ABB), Emerson, Endress + Hauser, Pepperl + Fuchs, Siemens which form WiTECK an open, non-profit membership organization. Their mission is to provide a reliable, cost-effective, high-quality portfolio of core enabling system software for industrial wireless sensing applications, under a company and platform-neutral umbrella [14].

1.2.3 6LowPAN

The short form of internet protocol version 6 (IPv6) over low-power wireless personal area networks is known as 6LowPAN [15] [16]. 6LoWPAN is the name of a concluded working group in the Internet area of the internet engineering task force (IETF). The 6LoWPAN concept is generated from the idea that *the Internet Protocol could and should be applied even to the smallest devices*, and the low-power devices with limited processing capabilities should be able to participate in the internet of things [17].

For allowing IPv6 packets to be sent and received over IEEE 802.15.4 based networks, 6LoWPAN group has defined encapsulation and header compression mechanisms. For data delivery for local-area networks, metropolitan area networks, and wide-area networks such as the internet, IPv4 and IPv6 are the fundamental protocols. Even though IEEE 802.15.4 devices have sensing communication-ability in the wireless domain, the inherent natures of the two networks though, are quite different. Fig. 1.7 shows the basic architecture for 6LowPAN.

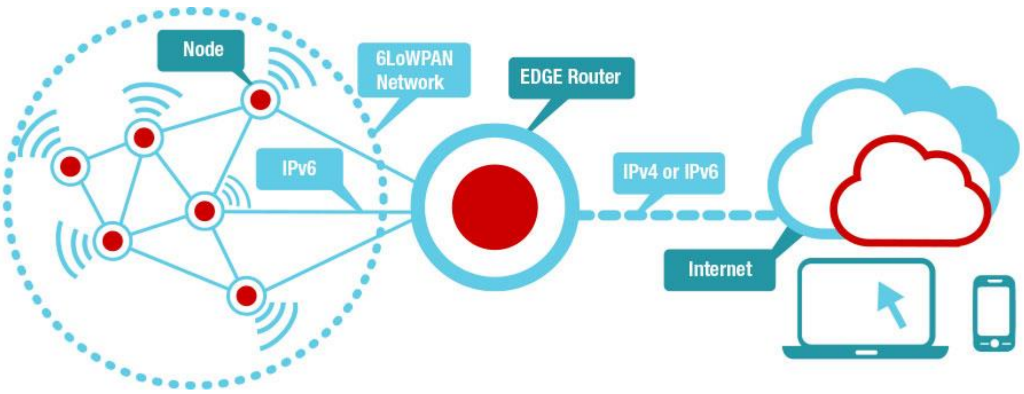


Figure 1.7: Basic architecture of 6LowPAN [18].

1.3 Problem Statement

IQRF is a hardback exemplar that is becoming much more popular with research and industries. The basic idea of IQRF is that it will connect objects around us to provide smooth communication and dependable services within low cost. Development of IQRF makes it possible to emerge these devices with different sensors, interact in-between them and co-operate with each other to make the service better and accessible anytime from anywhere.

There are so many applications which are supported to IQRF. Home automation, devices monitoring, and management of daily tasks are some of the examples for individual users. For professionals and industrial usage, telemetry, smart grids, street lighting, smart city. Operations with IQRF in industries with sensors and actuators can be efficient and more economical. Human touch on many things can be automated through connecting digital and physical objects together. Every sector for energy, computing, management, security, transportation is going to be benefitted with this new technology.

Development of several technologies made it possible to control the IQRF through internet. Through a gateway, these IQRF nodes can be controlled directly from the cloud server of IQRF. IQRF has developed three kinds of gateways for these communications. Wireless fidelity (Wi-Fi) gateway, global system for mobile communication (GSM) gateway and ethernet gateways are the three kinds. All these specially made gateway contains a programmable sensor node inside it, which acts as the coordinator. This coordinator is connected wirelessly to all the sensor nodes. Wireless sensor technology allows objects to provide real-time environmental condition and context. Such smart technologies allow objects to become more intelligent which can think and communicate.

Nanotechnologies are reducing the size of the chip, including more processing power and communication capabilities in a very small chip. Even though IQRF lacks high-speed data transmission or long-distance communication but is very low powered, packet-oriented and uses semi-static technology. In this paper, analyzing of several required characteristics of IQRF for solving problems has been discussed, and investigation of key technical components has been considered. We also illustrate the enhanced services of IQRF.

1.4 Research Methodology

In this thesis, studies have been conducted on IQRF technology as an enabling technology to achieve the *digital individual – networked* concept in IQRF alliance. In this challenge, how to perform a cooperation and communication among smart systems and devices is a key question. The first attempt was taken to configure IQRF starter kit which consists of various sensor nodes and coordinator node. The configuration needs to be done on a PC which has the software platform IQRF IDE (Integrated Development Environment).

Then, the main task will be the test for single-hop and multi-hop communication to check the maximum distance of communication corresponding to different transmission power levels by using a predefined range test. The experimental results of this test will help us to perform other communication scenarios by configuring the required power level for nodes.

Following the above range test, IQRF devices are configured to measure environmental factors such as temperature and humidity at various locations. In addition, external sensors can be integrated into the system to measure the light intensity of desired locations which can communicate the sensor data with the PC. Interestingly, a GSM gateway can be integrated with the system for uploading sensor data to an IQRF cloud. Moreover, we plan to find a way to control sensors via some commands assigned in the cloud.

1.5 Thesis Outline

The rest of this thesis is organized as follows.

Chapter 2, starts with background and related work for IoT. The basic architecture and applications of IoT are described. Research directions based on IQRF is also there.

From chapter 3, the experimental scenario begins. The first part of this chapter contains descriptions of the IQRF modules. The first experiment conducted for this thesis was the range test followed by single-hop and multi-hop communication scenarios. The chapter ended with the final experiment which is, temperature measurement for single-hop and multi-hop communication.

Light-based experiment begins in chapter 4. The whole chapter contains communication between IQRF and light dependent resistor (LDR), the configuration of IQRF module I/O pins, network setup, a brief description of LDR, the relation between LUX and Lumen, LUX calculation, the relation between LUX and resistance and all the experimental results. The major experiments for this thesis are placed in this chapter which is distance-based measurement and analysis followed by color testing and effect on the angle of arrival.

Chapter 5 is mostly the cloud-based testing of IQRF. For conducting these experiments, an additional module is required, which is the GSM gateway. All about the gateway is explained in this chapter. How the client-server communication is built, and configuration and connectivity with the gateway module to IQRF have been described briefly. The major experiments for this chapter are temperature measurement and LUX calculation which has been conducted solely from the cloud database.

Chapter 6 is the concluding part. Conclusion, Future work and contributions for this whole thesis are explained in this chapter.

The whole thesis ended with references, followed by an appendix (programming codes and datasheets) where data configuration for all the IQRF modules and the programming code is revealed.

CHAPTER 2

BACKGROUND AND ENABLING TECHNOLOGIES

The perception of the IoT has evolved due to merging of multiple technologies like real-time systematic, machine learning and embedded systems. This means that traditionally in order to authorize IoT, contribution of embedded systems, WSNs, and control system are immense. In this chapter, a brief description about concepts of IoT and related works are given.

At the beginning of the year 1982, a discussion occurred considering a network of smart devices. The first IoT device was a coke machine which was developed in Carnegie Mellon University. It was the first internet connected appliance which was able to report about weather, its own inventory, and temperature of coke. In an IEEE spectrum in 1994 a concept was described by Reza Raji. The concept was about moving small packets of data to a large set of nodes as to integrate and automate everything from home appliances to entire factories. During the year 1993 and 1996 companies like Microsoft's at Work or Novell's NEST proposed solutions. However, only in 1999, Bill Joyen presented device to device (D2D) communication as part of his *Six Webs* framework at the World Economic Forum.

In 1999, through the automatic identification (Auto-ID) center at Massachusetts Institute of Technology (MIT) and related market-analysis publications, the concept of the IoT became popular. Kevin Ashton, one of the founders of the original Auto-ID center, related radio-frequency identification (RFID) as a prerequisite for the internet of things at that point. Later he prefers the phrase *internet of things*. If all the objects and people in daily life were given identifiers, then computers could manage them and store them in its database. Besides using RFID, the tagging of things may be achieved through such technologies as near field communication, barcodes, quick response (QR) codes, etc.

A significant change is to extend *things* from the data generated from devices to objects in the physical space. In 2004, a model was proposed for future interconnection environment. The model included the concept that the universe consists of a ternary pattern which are the physical world, the virtual world, and the mental world. A multi-level reference architecture with the nature and devices at the bottom level followed by the level of the internet, sensor network, and mobile network, and intelligent human-machine communities at the top level, which supports geographically dispersed users to cooperatively accomplish tasks and solve problems by using the network to actively promote the flow of material, energy, techniques, information, knowledge, and services in this environment. This thought model envisioned the development trend of the internet of things.

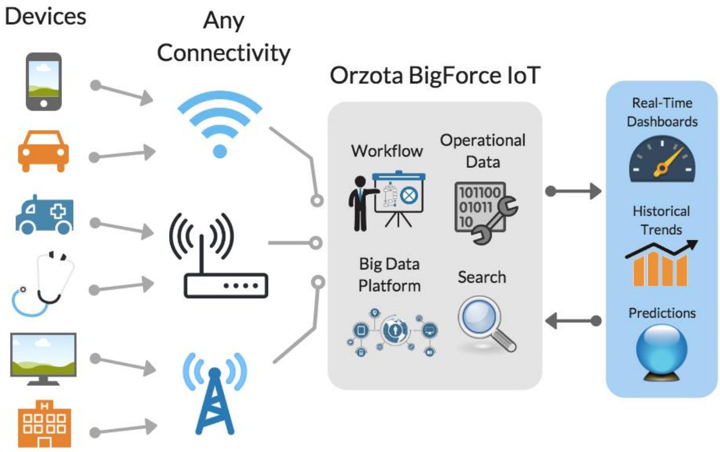


Figure 2.1: Applications of Internet of things [19].

2.1 Introduction to Internet of Things (IoT)

The internet of things (IoT) is a system which is interrelated to computing devices such as vehicles, home appliances and other embedded items with electronics, software, sensors, actuators. The network of these devices enables the ability to connect and transfer data over a network without requiring human-to-human or human-to-computer interaction. Each of these devices is uniquely identifiable by their own embedded computing system but can inter-operate within the existing internet infrastructure.

Every day the number of IoT devices increase at an exponential rate. The online-capable devices increased 31% from 2016 to 8.4 billion in 2017. Experts all over the world estimate that the IoT will consist of about 30 billion objects by 2020. It is also estimated that the global market value of IoT will reach $7.1 trillion by 2020 [20].

Each of the IoT devices contains several sensors which allow them to sense different objects which can be controlled further remotely covering existing network infrastructure creating chances for more direct integration of the physical world into a computer-based system. Thus, resulting in improved efficiency, accuracy, precision and economic benefit in addition to reduced human intervention. When IoT is engaged with sensors and actuators, the technology becomes more general to a class of cyber-physical systems, which also encloses technologies such as smart homes, intelligent transportation, smart grids, virtual power plants, and smart cities. The real-life application of IQRF as an IoT device can be found all over the world, for example,

Budapest – for more than 35k parking slots control,

Israel – for more than 50k street lights control,

Czech Republic and Slovakia – more than 30k smart street lights control,

Mexico – more than 50k shopping mall light control.

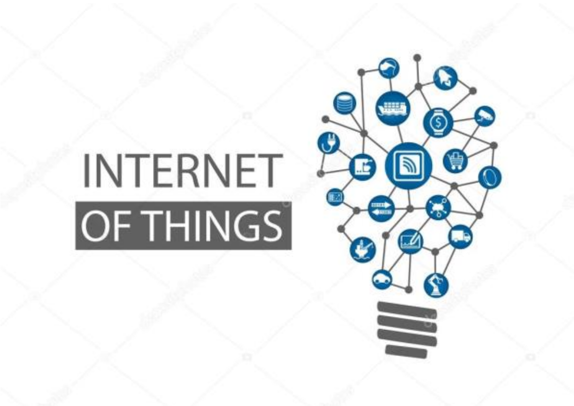


Figure 2.2: Internet of things.

Also, in different places IQRF is being used for industrial hall lighting, heating and cooling, oxygen and carbon dioxide (CO_2) monitoring, telemetry and temperature control, etc.

Things part, in the *Internet of things* refer to the sensing part, which consist wide variety of devices such as heart monitoring implants, biochip transponders on farm animals, cameras streaming live feeds of wild animals in coastal waters, automobiles with built-in sensors, deoxyribonucleic acid (DNA) analysis devices for environmental/food/pathogen monitoring or field operation devices that assist firefighters in search and rescue operations. Legal scholars suggest regarding *things* as an *inextricable mixture of hardware, software, data and service* [21]. These devices collect data with the help of various existing technologies and then autonomously flow the data between other devices.

2.2 Basic Architecture of IoT

For the architecture for IoT, there is no single general agreement. There are several architectures which have been proposed by different researchers. But if we talk about the most basic things, then these IoT devices can be categorized into two architectures – three layer and five layer [22].

Three and Five Layer Architectures:

The most basic architecture is a three-layer architecture for an IoT device which has been showed in Fig. 2.3. It was introduced in the very first stages of research in this area. From the figure, we can see that part A has three layers which are application layer, network layer and perception layer. Perception layer is the physical layer, which consist of sensors for sensing and gathering information about the environment. It senses physical parameters or identifier objects in the environment. The network layer is responsible for connecting the devices to other smart things such as network devices and servers. Some of its features are also used for transmitting and processing sensor data. The application layer is mainly responsible for delivering application services to the end user such as smart homes, smart cities, and smart health.

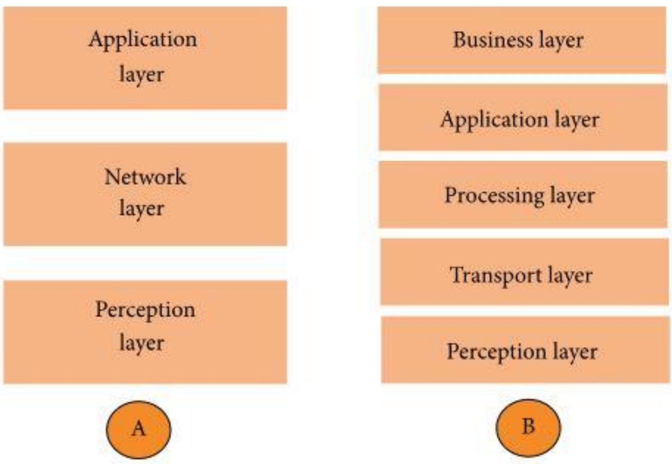


Figure 2.3: Basic IoT architectures (Part A – Three layers, Part B – Five layers) [14].

The three-layer architecture defines the main idea of the internet of things. Even though it is not sufficient as research often focuses on finer aspects of the internet of things. That is when the five-layer architecture comes in. The five layers are perception, transport, processing, application and business layers. Here the role of the perception and application layers remains the same as the architecture with three layers. The remaining 3 layers are quite different.

The transport layer transfers the sensor data from the perception layer to the processing layer and vice versa through networks such as wireless, third generation (3G), local area network (LAN), bluetooth, radio frequency indication (RFID), and near-field communication (NFC). The processing layer, also known as the middleware layer is responsible for storing, analyzing, and processing huge amounts of data that comes from the transport layer. It can manage and provide different types of services to the lower layers. It employs many technologies, for example, databases, cloud computing and big data processing. The business layer manages the whole administrative system of the IoT including applications, business models, profit calculations and users' privacy.

2.3 IoT Application Scenarios

The applications for IoT devices are massive. At the beginning of 21st century, several categorizations have been suggested. Most of them agree on a common base between consumer, enterprise, and infrastructure applications. The internet of things is the next stage of the information revolution and referenced the inter-connectivity of everything from urban transport to medical devices to household appliances is quoted by the former British chancellor of the exchequer, George Osborne [23].

The IoT devices are mostly very small in size, they are compact and can be embedded in networks with limited central processing unit (CPU) capability, memory and power resources. That means that IoT finds its way into most of the applications in almost every field. These types of systems could oversee collecting information in settings ranging from natural ecosystems to building houses and factories.

As IoT devices can connect to the internet without any hassle, recently most of the household appliances are equipped with IoT devices embedded inside it. For instance, intelligent shopping systems can monitor users' purchasing habits by tracking their specific mobile phones or computers and can store the information in the small memory they have. Later these users could then be provided with special offers on their favorite products or even location of the items that they need.

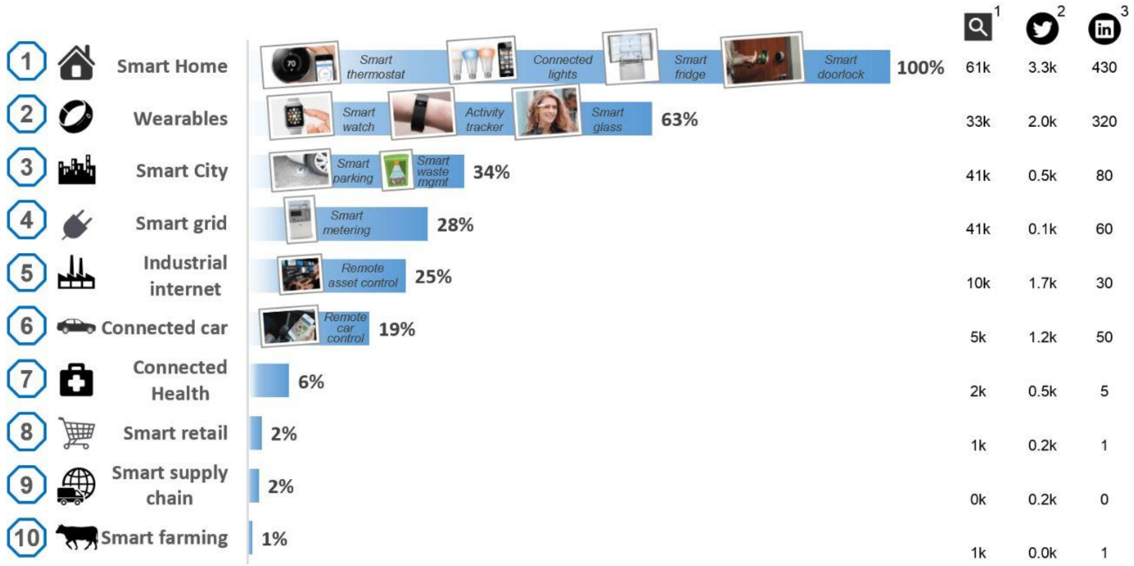


Figure 2.4: 10 best applications of IoT [24].

The embedded device in the fridge can automatically convey information to the phone to notify a user for his/her daily grocery shopping conditions. Also, supplementary examples of sensing and actuating are reflected in applications that deal with heat, water, electricity, and energy management. Cruise control and assisting transportation systems is a trend in automobiles nowadays. Other applications for IoT devices can be enabling smart home security features and home automation. There is another new concept *internet of living things* has been proposed to describe networks of biological sensors that could use cloud-based analyses to allow users to study DNA or other molecules.

2.4 IQRF Technology

The IQRF system is designed according to user’s needs for allowing RF wireless connection. For executing the defined functionalities of a user, there is a transceiver module which contains a microcontroller for controlling the transceiver operations. Originally the IQRF transceiver module architecture has two software layers:

- Built-in basic routines programmed which are provided by the manufacturer. The set of functions are called operating system (OS).
- Application layer exploits routines from the basic layer to customize the module for user-specific operation.

There is no need to compile protocol related routines, just the application is compiled. This approach significantly reduces time and development costs [25].

For this thesis work, 2 pre-built programmes have been used which are provided in the IQRF IDE software for using the modules as coordinator and nodes. They are known as *demo hardware profiles*. They can be uploaded in the modules for doing simple experiments. But for further complex experiments, the demo hardware profile is not enough. For this purpose, IQRF Alliance has provided the users with *general hardware profiles*. All the experiments in this thesis have been conducted using the general hardware profiles.

2.4.1 IQRF Basic Features

There is a built-in operating system for the transceiver architecture system using IQMESH network protocol. There are several optional direct peripheral access (DPA) layers for an application which needs no programming. It has different power modes such as configurable extra low power with the sleeping mode. Receiving and routing consumes very low power which is about 15µA standard, low power (LP) and extra low power (XLP) are some of the low power modes.

IQRF has independent receiving and transmitting capability with excellent networking, containing up to 240 devices and 240 hops in a single network. Every transceiver can work as a network coordinator or as a common node. Every node can additionally route on the background. Real-time discovery option is also placed in IQRF. This function means that the routing paths can found automatically for transparent addressing of randomly placed nodes. FRC - fast response command – the fastest network control, management and data aggregation are also some of the unique features of IQRF. Bonding (network construction), discovery and FRC options can be used with low power modes.

2.4.2 Bonding

IQRF IDE contains 2 general hardware profiles. One is for the coordinator and the other one is for the nodes. There is a set of configuring window called TR configuration. For modifying the configuration of the transceivers and the receivers, it is used. The TR configuration contains several peripherals for changing the frequency band, channel, wireless programming and so on. The TR configuration and general hardware profiles both need to be uploaded to the modules for operating the network properly. The configurations must be same for both the transceiver and the receivers. After uploading, the node modules will keep one of their red light emitting diodes (LED) blinking for 10 seconds. Which means that the nodes are ready to be bonded and the upload was successful. On the other hand, the coordinator is ready to bond the nodes. Fig. 2.5 shows that the coordinator is ready and there are no nodes bonded to it.

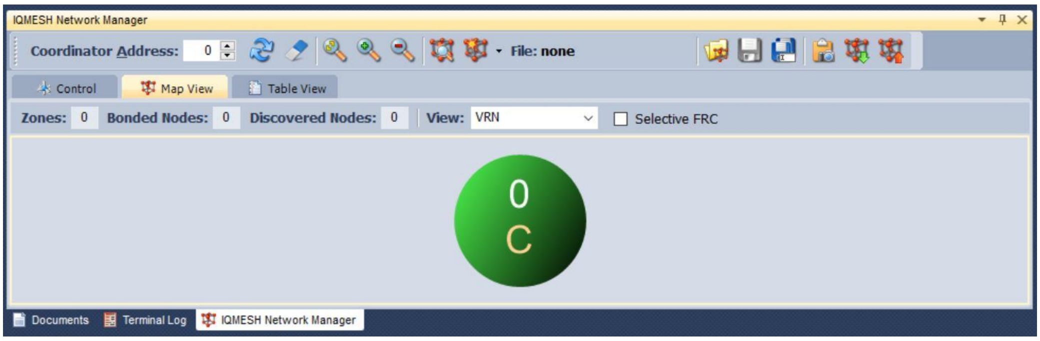


Figure 2.5: Coordinator is ready to bond nodes, and no nodes are bonded yet.

Now the nodes need to pair with the coordinator. For bonding nodes, the user needs to hit the bond node button inside the network manager, and it will start searching for nodes. From the node module, hitting the application button will respond to the search message of the coordinator, and they will be paired. Upon bonding a node, an orange bullet box up inside the matrix glows up. In this way, all the nodes must be bonded within few seconds. There are ways of breaking the bonding. Hitting the unbond node, after selecting a node, will break the bonding from coordinator side. In this way, the node is unbonded, and the node is aware of this.

Another way is that the nodes can also be deleted only from the coordinator and in this case, the node is not aware of it. The unbonding part must be done manually. The nodes can also be reset by holding the application button and pressing the reset button at the same time. This will reset the nodes, and it will start flashing red LED again which means the node is ready to bond again. The application button must be released when the green LED stops flashing while resetting. This is a crucial part as only 1 second delay can hamper the resetting process and the user must repeat the whole process again. Before rebonding a node, the unbond process needs to be done properly. Otherwise, it will create problems while rebonding again.

2.4.3 Discovery

For communicating with the network, the discovery button must be pressed. It is an automatic procedure which sets up the network topology automatically and is described in the following paragraph.

During discovery operation, the coordinator searches for nodes connected to the network to give them unique virtual routing numbers (VRNs), with respect to the distance from the coordinator using different power levels, for example, power level 0-7, where 0 is the lowest and 7 is the highest power levels respectively. The coordinator starts to search its neighbors by sending an *answer me* message. Nodes responding to the *answer me* message will receive their unique VRNs. This process can also be done automatically.

There is custom DPA handler called ‘CustomDpaHandler-Autobond’ inside the IQRF starter package. The user can upload the ‘CustomDpaHandler-Autobond.c’ file easily to all the nodes by compiling them to hex file from IQRF IDE. The nodes will seek for the coordinator's searching message and upon finding the message, the nodes will automatically answer and be bonded. It can be done manually by pressing the *answer me* button on every node when the coordinator is seeking. When the nodes are bonded and discovered, it can be visible from the matrix inside the IQMESH network manager which has been shown in Fig. 2.6.

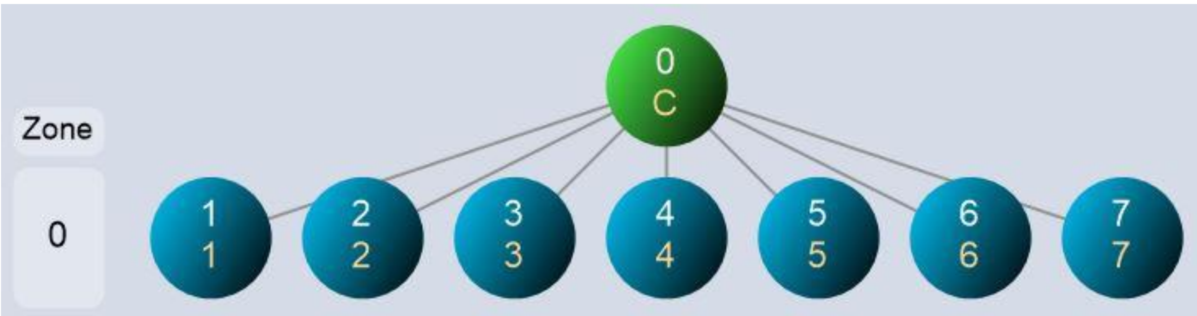


Figure 2.6: Nodes are bonded with the coordinator (single-hop).

Dynamic timing is one of the unique features. It helps IQRF with selectable number of hops and time slots. It also contains various routing algorithms to fit various topologies. Advanced network management tools, remote bonding of nodes to the network, backup (seamless replacement of network devices), automated network self-construction results in very low implementation cost.

2.4.4 IQRF Integrated Development Environment

IQRF Alliance has provided the users with a PC software for development, debugging, network management, service, and maintenance. It contains powerful software environment for Open source IQRF software development kit (SDK) for development of devices programmable in C or JAVA. There is no license or carrier fees for this software. There are two free design tools which are IDE with graphical user interface (GUI) and command line IDE [26]. Both are partly similar but different in the way of using.

IDE with GUI: This is full featured IDE with graphical user interface. It is intended for developers requiring interactive project-oriented work, complexity and sophisticated tools for design, upload, debug, testing, service, and maintenance.

Command Line IDE: This is used for transceiver programming running from the command line. It is intended for experienced developers to make design process faster and more effective or for unskilled operators to speed up and secure the manufacturing.

2.4.5 IQRF-based Research Directions

Since IQRF is a recently introduced complex communication platform, in its research field, it is not able to observe much work. However, in the following, brief discussion of some existing work on these topics are given. In [27] the IQMESH protocol is analyzed by highlighting its benefits and limitations as well. A case study is performed in [28] for a smart house which has IQRF communication platform. Simulations are carried out for a wireless sensor mesh network considering both free space and building environment. A similar case study is done in [29] for home and industrial automation. Therein, possibilities of using the IQRF communications platform in the home automation are discussed. Same authors presented IQMESH protocol, its advantages, strengths, also limitations and specific implementations in [30].

Moreover, various aspects of LPWAN technology coexistence in IoT environment is discussed in [31] with respect to IQRF and other frequency distribution LPWAN technologies. A measurement study performed in [32] studied the concentration of highly toxic carbon monoxide (CO) within the space of waste rock storage. Wireless IQRF technology is used to transmit measured concentration.

CHAPTER 3

IQRF IOT TESTBED EXPERIMENTS

In this chapter, a brief description about IQRF IoT testbed, how the single and multi-hop networks are built and configured, IQRF range test experiment as well as single-hop, multi-hop and temperature measurement experiments are explained. All the experiments in this thesis have been conducted using the general hardware profiles.

3.1 IQRF IoT Testbed

This part includes different technological features of IQRF. IQRF Alliance is the manufacturer of this testbed. They are also an international community for developers and system integrators engaging in different connectivity of IQRF. The experimental package comes with different types of devices. The list of devices is given below.

1. CK-USB-04 - 1 unit.

2. DK-EVAL-04 - 15 units.

3. TR-72DA - 14 units.

4. GW-USB-06 - 1 unit.

5. USB Charging dock - 2 units.

6. USB Charger - 1 unit.

7. USB Charging Cable - 1 unit.

8. Remote Control - 1 unit.

A brief description and working principle of these devices are given below [33].

TR-72DA: TR-72DA is the main radio frequency (RF) transceiver module. It has an operation mode in 433MHz, 868 MHz, and 916 MHz ISM band. Although the 433 MHz is reserved in

European Union (EU), the 868MHz and the 916 MHz are the main functioning bands. This device is highly compact and integrated with a ready-to-use design containing microcontroller unit, RF circuitry, low-dropout regulator (LDO), serial electrically erasable programmable read-only memory (EEPROM), optional temperature sensor and optional onboard antenna. External antenna results in higher RF range. It consumes extremely low power for battery powered applications.

Figure 3.1: IQRF TR-72DA [33].

Figure 3.2: IQRF CK-USB-04 [33].

The operating system is specially designed for the built-in microcontroller unit which is inside the TR module. It has optional DPA framework which supports programming for fewer applications. This TR module is plugged inside the CK-USB-04 device for programming. It can be programmed to either to act as a coordinator or a node. The built-in antenna inside this device is used for communication. The device is shown in Fig. 3.1.

CK-USB-04: CK-USB-04 device is shown in Fig. 3.2. IQRF Programming and Debugging are done with this device. In-circuit programming support is given to this device. It is also known as an IQRF application host. Uploading of coded programs can be done using wired or wirelessly. The CK-USB-04 device consists of 2 push buttons including 2 LED indicators attached to two buttons of the module. The main power source for this device is the Universal Serial Bus (USB) connectivity of the PC. Its firmware can also be upgraded using the IQRF software which is known as IQRF IDE. This device is white in color.

DK-EVAL-04: DK-EVAL-04 is the IQRF's universal development kit. There are 15 devices inside the packaging, and all of them comes in black color. It is used as the host for the IQRF's TR modules. This device contains a battery which is its main power source, and this device is completely portable. Like the CK-USB-04 module, this device also contains 2 push buttons, but it contains another extra LED which makes it in total of 3 LED indicators. The battery it contains can be recharged by a micro USB charging port. The built-in accumulator, guards over current flow. DK-EVAL-04 device is shown in Fig. 3.3.

Figure 3.3: IQRF DK-EVAL-04 [33].

GW-USB-06: A small USB gateway comes with the whole IQRF packaging. This gateway is specially intended as an interface between IQRF network and PC. This device comes with an internal TR module which can do specific functionalities by software. The main purpose for this device is, IQRF – PC interfacing, diagnostic tools and wireless programmer for TR modules. Unlike the previous 2 devices, it contains no push buttons but contains 3 LED indicators. It has only USB connectivity with no batteries inside it. The internal antenna of this device is used for communication. The gateway has been shown in Fig. 3.4.

Figure 3.4: IQRF GW-USB-06 [33].

Charging Docks: Charging docks can recharge the batteries of five DK-EVAL-04 units simultaneously. It provides all the unit with 1 ampere (A) current, and the batteries need around 3 to 4 hours to recharge fully. The Fig. 3.5 shows the charging dock that IQRF Alliance has provided.

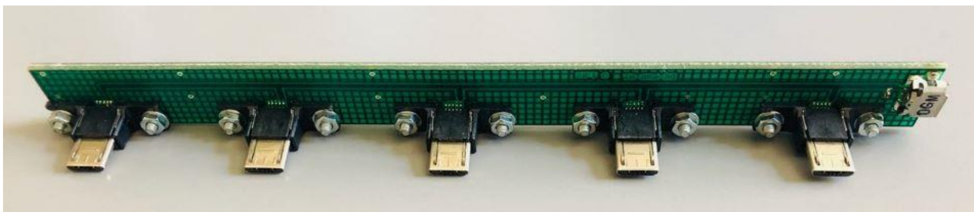


Figure 3.5: IQRF Charging Dock.

Using these IQRF modules, three major experiments are described in chapter 3. Scenario 3A is range testing. Scenario 3B and 3C are single-hop and multi-hop experiments respectively.

Scenario 3D is temperature measurement of the surrounding, based on the single-hop and multi-hop communication.

3.2 Scenario 3A - IQRF Range Test

The range of IQRF per hop depends on the transmission power for both ends. It has a per-hop range up to tens of meters inside buildings and hundreds of meters in free space. It can cover up to several kilometers with special arrangements. This experiment was performed with a direct line of sight in a windy morning. As it was mentioned before that the devices can perform in different transmission power, set by the user. At first, the devices were programmed with a simple code. It was programmed to receive and transmit in both ends with a transmission level 1 to transmission level 7 in an open space (See Fig. 3.6).

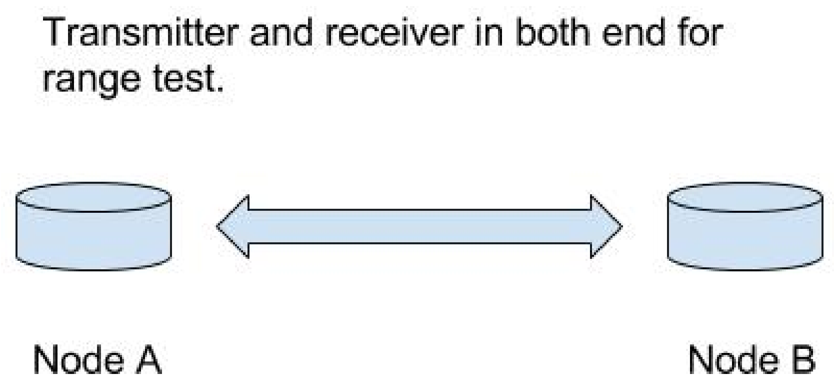


Figure 3.6: Transmitter and receiver range test experiment.

When the devices were powered, and the IQRF transceivers were inserted, they start flashing one of the LEDs immediately. Which means that they are directly in range with each other. The transceivers were programmed to flash green while sending a packet and blinking red while receiving. When they were out of range, only the green LED will flash. The red LED is much more visible than the green one. That is why, the transceivers were programmed like this, not the other way around. But, depending on the user, it can be changed the other way around from the code. So, the transceivers were only sending packets, but they were not receiving anything. The results are shown in the Table 3.1.

Transmission Power	Approximate Range (meters)
1	1.5
2	3
3	6
4	140
5	174
6	229
7	304

Table 3.1: Range test results.

3.3 Scenario 3B - Single-Hop Communication

Making applications wireless is very easy with IQRF. IQRF is mainly based on JAVA programming. A product running on JAVA programming can easily be integrated with IQRF, for example, Arduino or Raspberry Pi. But for this topic, the main concern is how the connections are made with each other. Single hop connections between IQRF nodes can be made within few minutes. From the IQRF website, the user can download the IQRF starter package.

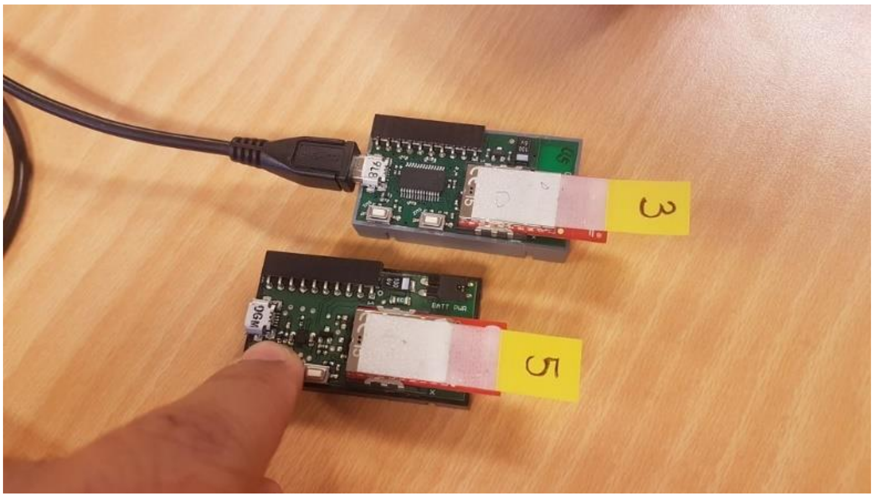


Figure 3.7: No transmission has been executed.

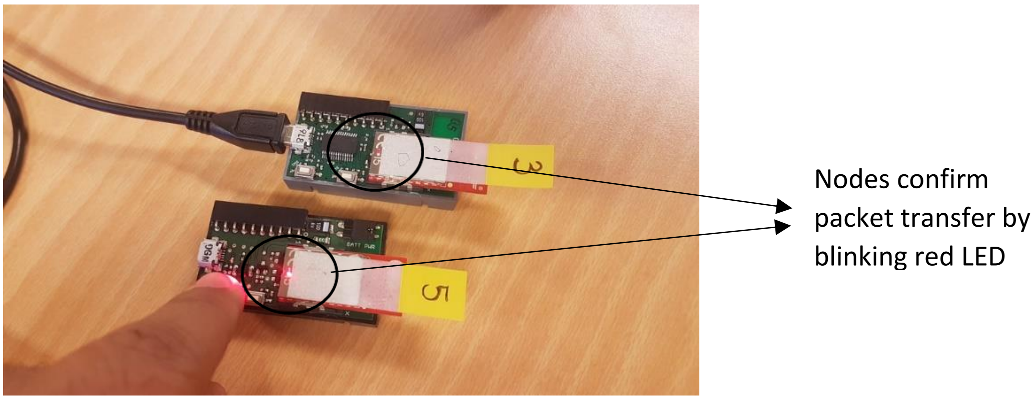
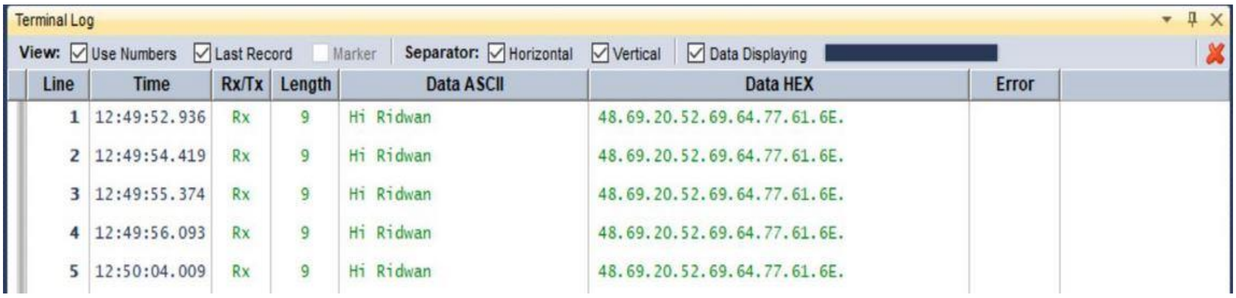


Figure 3.8: Transmission has been executed (Red LED blinking in both modules).

The starter package contains several pre-built experimental coded files. Such files are E01-TX and E02-RX which are the transceiver and receiver coding respectively. These codes are simple and short. In the appendix part of this thesis, the codes are provided.

These source codes need to be compiled to get the appropriate hex file. This compilation and uploading of the code can be done easily using the IQRF IDE software which IQRF website provides. Later the hex file needs to be uploaded to the TR module. There is also another important part of the software which needs to be configured properly. It is known as the TR configuration. The radio frequency band and channels can be changed in this section. Note that, if the TR configuration of a node and a coordinator is not same, the connections cannot be made within.

When the hex file and the TR configurations are uploaded in two nodes, one of them acts as a coordinator, and the other one acts as a receiver. It can be shown using the LED indicators in the node modules and by using the software's terminal log. While programming the transmitter code, the transmission message and the number of bytes can be changed. After all the preparations are made, when the application button on the transceiver is pressed, a package is sent to the receiver, and the receiver node receives the package, and it confirms it by blinking one of its LEDs. Also, if we see it from the terminal log, we can easily see a transmission message has been sent and is being received by the receiver. In this condition, the transmission message was 'Hi Ridwan'. The Fig. 3.7, Fig. 3.8 and Fig. 3.9 confirm the experiment.



Terminal Log

View: Use Numbers, Last Record, Marker | Separator: Horizontal, Vertical, Data Displaying

Line	Time	Rx/Tx	Length	Data ASCII	Data HEX	Error
1	12:49:52.936	Rx	9	Hi Ridwan	48.69.20.52.69.64.77.61.6E.	
2	12:49:54.419	Rx	9	Hi Ridwan	48.69.20.52.69.64.77.61.6E.	
3	12:49:55.374	Rx	9	Hi Ridwan	48.69.20.52.69.64.77.61.6E.	
4	12:49:56.093	Rx	9	Hi Ridwan	48.69.20.52.69.64.77.61.6E.	
5	12:50:04.009	Rx	9	Hi Ridwan	48.69.20.52.69.64.77.61.6E.	

Figure 3.9: Reading transmission message through terminal log.

3.4 Scenario 3C - Multi-Hop Communications

When a radio frequency communication covers a large area, usually that communication is in multi-hop. Several nodes connect to each other for making a large network. In this way, the transmission message depends on other nodes which need to work as replays to reach its destination. Transmission in communication takes major power drain for long distance coverage. Also, the chances of losing communication or packet loss are much higher in this case. For these purposes, a multi-hop communication is much effective and energy efficient.

IQRF mainly uses ad hoc networking. Multi-hop in IQRF doesn’t require any fixed wireless infrastructure. The multi-hop requires bonding the nodes with the coordinator. Here, a coordinator can either be a TR module, The USB gateway or the GSM gateway. Using the IQRF IDE software, the user needs to bond the nodes, and later the discovery mode needs to be turned on for searching and activating the bonded nodes. Once the nodes are bonded, they can’t be unbonded without the IDE software. Whichever the coordinator is, the IDE software is required to bond the nodes. The bonded nodes can be later reset for bonding with another coordinator. It means a node can’t be bonded with multiple coordinators without resetting. The bonding and discovery functions are described in chapter 2, subsection 2.4.2 and 2.4.3.

Once the nodes are bonded and discovered, the transmission power can be changed wirelessly. Right-clicking in the connected nodes which can be seen from the IDE IQMESH network manager will open a window. The user can access the TR configuration of every node from there and upon changing the transmission power will automatically be uploaded to that node by the coordinator wirelessly. Depending on the transmission power, a user can easily set up the multi-hop communication. The dropping position of the nodes creates a topology and changing the position of the nodes can change the topology. In Fig. 3.10 and Fig. 3.11, the transmission power of the nodes and the coordinator were set to 0 and placed them in such a way so that it gives a network topology with 7 hops.

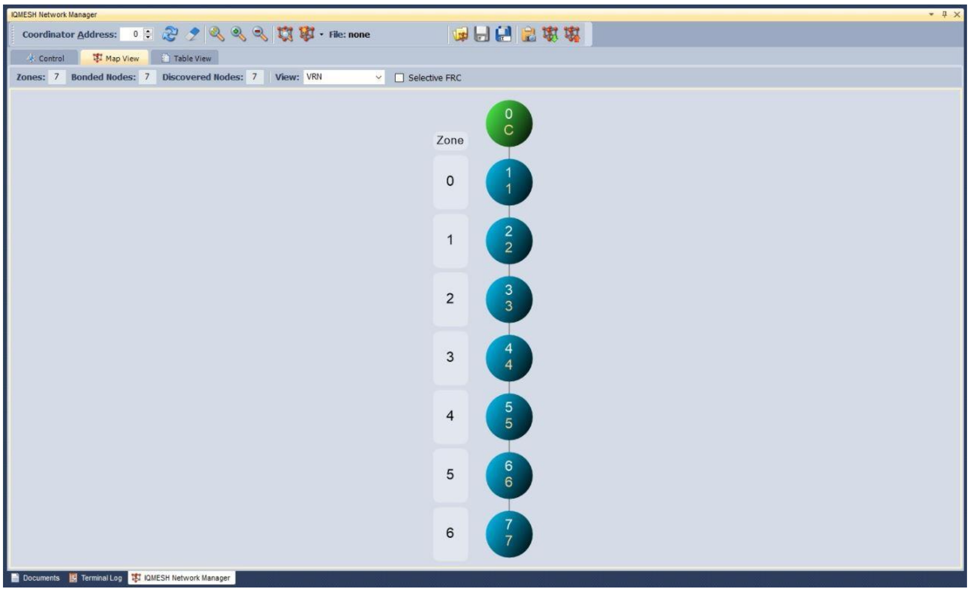


Figure 3.10: Network topology with 7 hops.

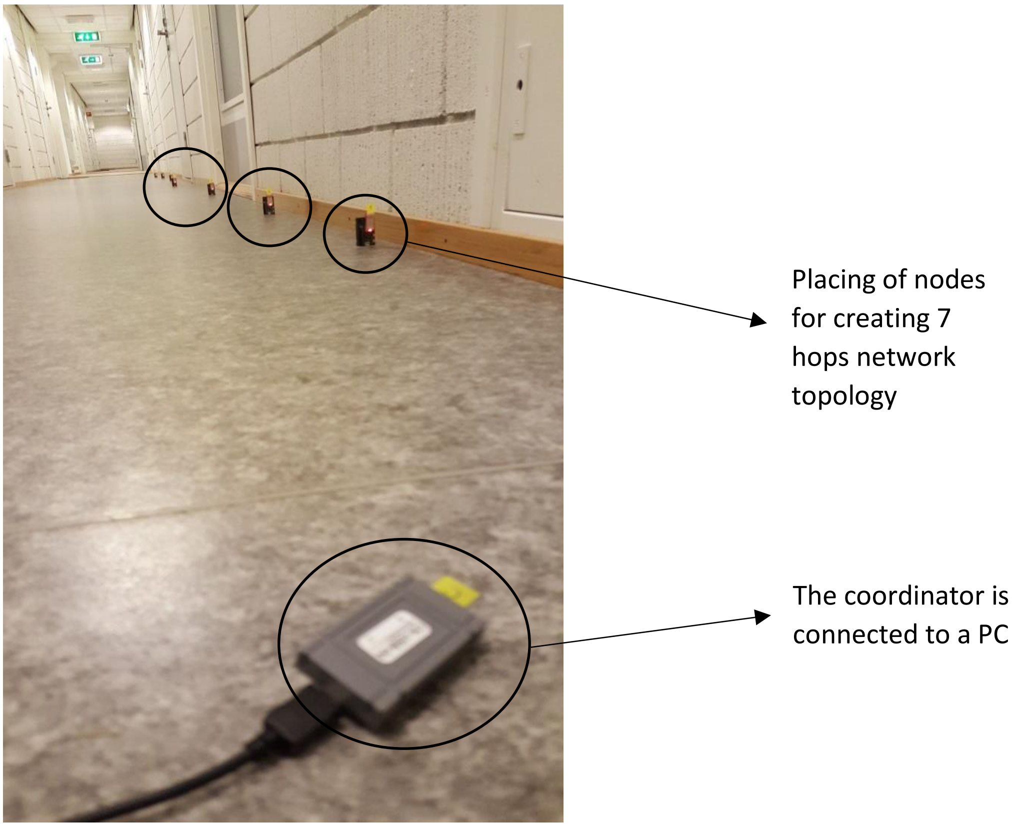


Figure 3.11: Positioning for creating a 7 hops network.

3.5 Scenario 3D – Temperature Measurements in Single-hop/Multi-hop Scenarios

The whole experiment has been divided into two sub-sections. Single-hop scenario and multi-hop scenario respectively.

Single-Hop Scenario

All the IQRF DK Evaluation Board comes with a built-in temperature sensor. We can use the IQRF IDE to measure the surrounding temperature of a DK-EVAL-04 module. First, the TR modules need to be programmed and inserted in the modules. Bonding and Discovery are necessary for this experiment.

At the DPA terminal of IQRF IDE, 5 rows needed to be filled to send the proper message which will ask the node to send its temperature to the user. The first one is *NADR* which remarks as node address. In here, different nodes can be controlled. It can be done both for single and multi-hop networking. For controlling all the nodes at once, the NADR value must be set to FFFF. For this experiment NARD value FFFF has been used for asking all the nodes to show their surrounding temperature. The second row is *PNUM*, which is defined as peripheral number. Within this row some of the predefined peripherals are placed. For this scenario, the peripheral *Thermometer* was used which remarks as 0A in the PNUM section. All the sections needed to be filled with hexadecimal numbers.

In this case, 0A has been used because it represents the number 10 in decimal and we used number 10 peripheral from PNUM part. Each peripheral has a list of commands. It can be selected from the third field which is known as *PCMD*. For this case, PCMD value 00 has been used which remarks get temperature. The peripherals can be modified or can be created with user's own command using the custom DPA handler codes. This provides flexibility in using the IQRF. The forth row is called hardware profile identification (HWPID). This is hardware profiles, which defines the group of products a user wants to communicate with. For the experiment scenario, it will be left as full broadcast. The fifth row is *DATA*, using which a user can send up to 56 bytes of data. The bytes can be entered in this field.

For the single hop temperature measurement, underneath values have been used in the DPA Terminal.

NADR - FFFF

PNUM - 0A

PCMD - 00

HWPID - FFFF

DATA - No Value

After setting this value, a user must send the message to all the connected nodes, and the value can be seen instantly in the terminal log of IQRF IDE. With the double click on the response log, a user can open the packet inspector, where the temperature can be read in Celsius. For achieving the high-resolution temperature, it can be done using the last following bytes of the packet inspector. They will be in hexadecimal values and can be decoded using the following Table 3.2.

TEMPERATURE REGISTER FORMAT

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
LS Byte	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}

	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
MS Byte	S	S	S	S	S	2^6	2^5	2^4

TEMPERATURE/DATA RELATIONSHIP

TEMPERATURE	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+125°C	0000 0111 1101 0000	07D0h
+85°C*	0000 0101 0101 0000	0550h
+25.0625°C	0000 0001 1001 0001	0191h
+10.125°C	0000 0000 1010 0010	00A2h
+0.5°C	0000 0000 0000 1000	0008h
0°C	0000 0000 0000 0000	0000h
-0.5°C	1111 1111 1111 1000	FFF8h
-10.125°C	1111 1111 0101 1110	FF5Eh
-25.0625°C	1111 1110 0110 1111	FE6Fh
-55°C	1111 1100 1001 0000	FC90h

Table 3.2: Temperature measuring table [34].

Also, there is another way to collect data from all the nodes in the network. FRC is the best way to do it. For using FRC, the user must select FRC, which is placed under macros in IQRF IDE. Simply by clicking the 1B: Temperature button, this function can be achieved. Fig. 3.12 shows part of macros FRC. This will fill all the command fields with the necessary command and just by hitting send, within less than 1 second, user can get all the data from all the nodes inside the network. From the packet inspector, the user can read all the data that has been captured by the nodes, which is shown in Fig. 3.13. For experimental purpose, node 7 and node 8 has been placed in different temperature zones, such as node number 7 has been placed inside a freezer for few minutes and node number 8 has been placed on the top of a heater for observing the efficiency of the devices and the results were extremely accurate.

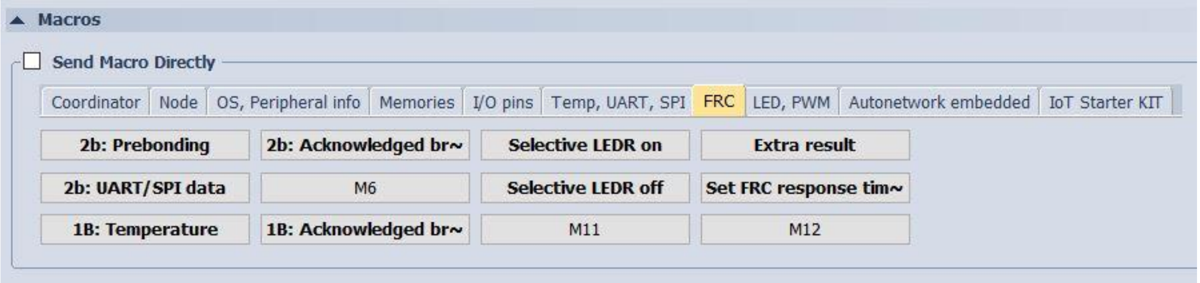


Figure 3.12: Macros FRC – 1B: Temperature.

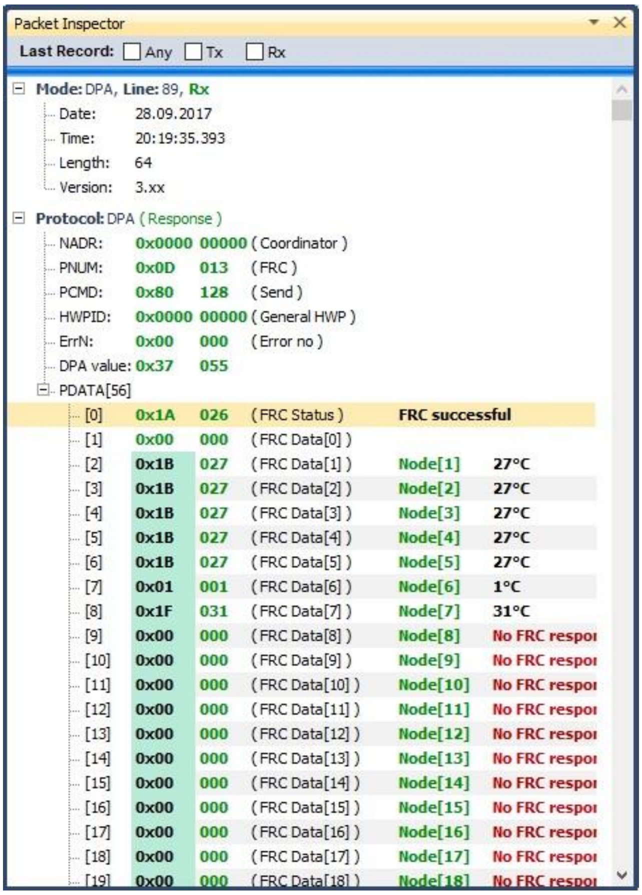


Figure 3.13: Single-hop temperature measurement.

Multi-Hop Scenario

For this test, the devices were placed in a multiple hop structure as shown in Fig. 3.14, each of them at 1.5m from each other forming a linear topology. In this case, the node 6 were inside in a refrigerator and node 7 was placed on the top of a heater for several minutes.

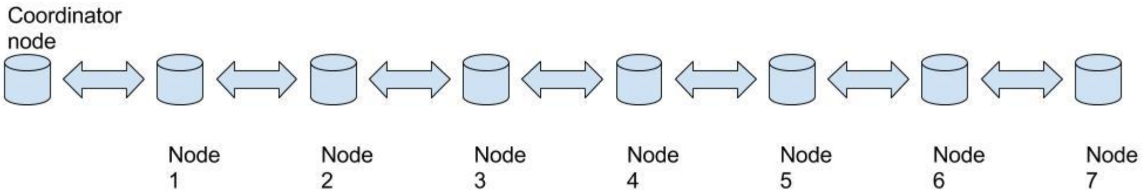


Figure 3.14: Nodes placed with linear topology.

The linear arrangement with the least transmission power (range of 1.5 m) to make sure the data must do a multi-hop transit to the coordinator. The results are shown in Fig.3.15, taken from packet inspector in the IDE. The temperature measurement procedure is same for single-hop and multi-hop scenario.

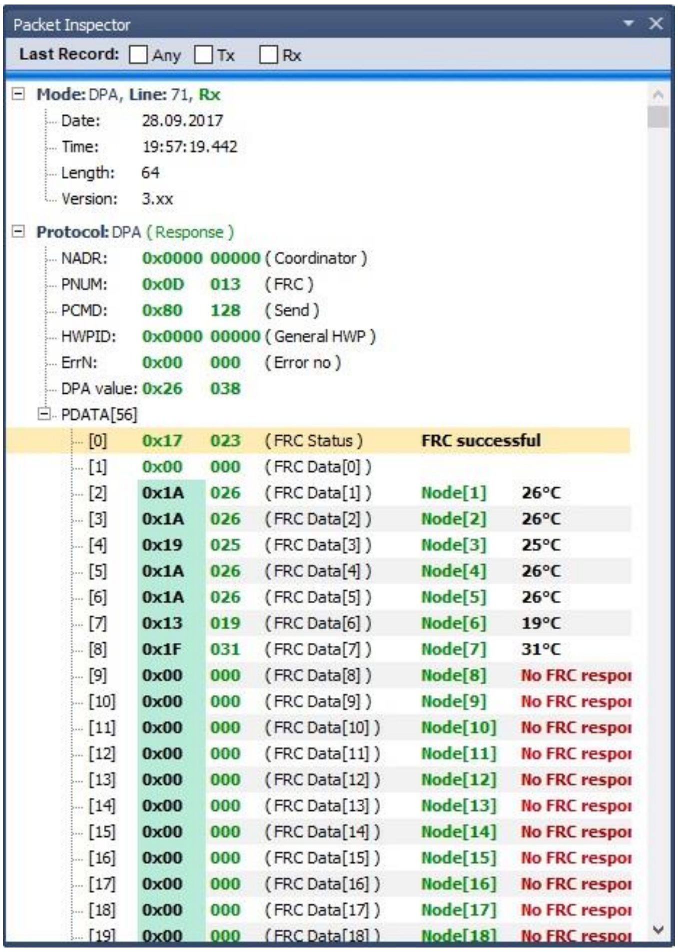


Figure 3.15: Multi-hop temperature measurement.

CHAPTER 4

LIGHT INTENSITY TEST BASED ON IQRF TESTBED

IQRF contains the option for connecting external sensors with the DK-EVAL-04 board for conducting complex types of experiments. For observing the possibilities, the LDR experiments have been conducted. In this chapter, a brief description of LDR, how the devices were interconnected, the network construction for LDR, experiments about distance-based measurements and effects on the angle of arrival has been discussed.

4.1 IQRF Configuration for Light Intensity Measurements

The DK-EVAL-04 module of IQRF comes with different I/O pins. The Fig. 4.1 explains the pin configuration of the module.

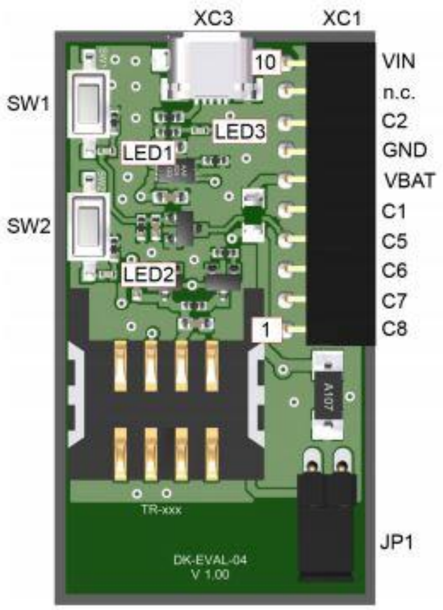


Figure 4.1: DK-EVAL-04 pin configurations [33].

This I/O pins can be configured, and different external sensors can be linked with them. In this experiment, a GL5528, LDR was connected to the DK-EVAL-04 module. At first, two codes "GeneralHWP-Coordinator-STD-SPI-7xD-V223-151023.iqrf" and "GeneralHWP-Node-

STD-SPI-7xD-V223-151023.iqrf” was uploaded to the TR modules for configuring them as coordinator and nodes respectively. These are manufacturer in-build codes and are highly encrypted. After that a custom DPA handler, “CustomDpaHandler-UserPeripheral-ADC.c” which can be found in the starter package of IQRF needs to be uploaded to the nodes. More about custom DPA handlers are explained in A.4 section in appendix part.

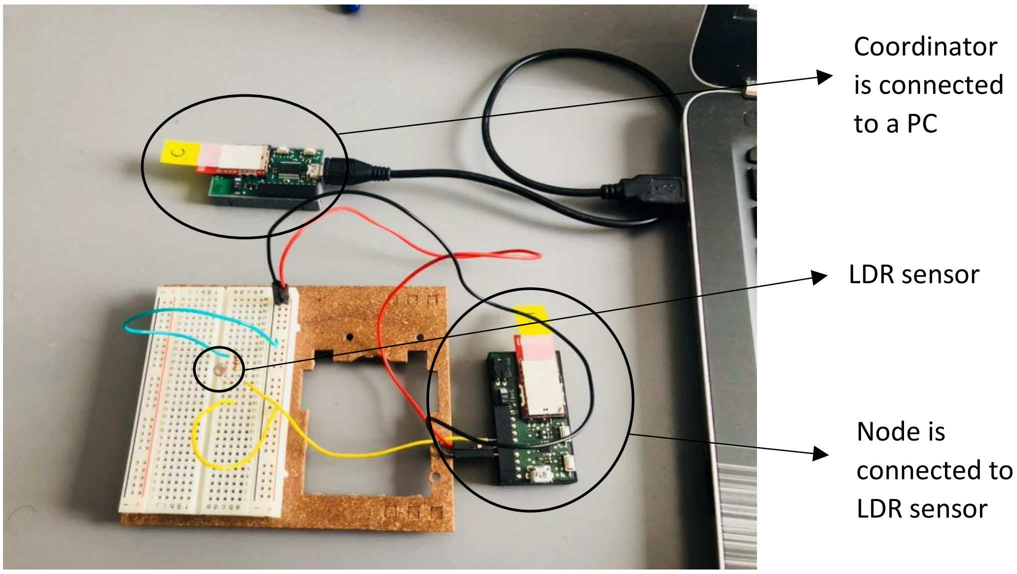


Figure 4.2: Connection of GL5528 LDR with IQRF.

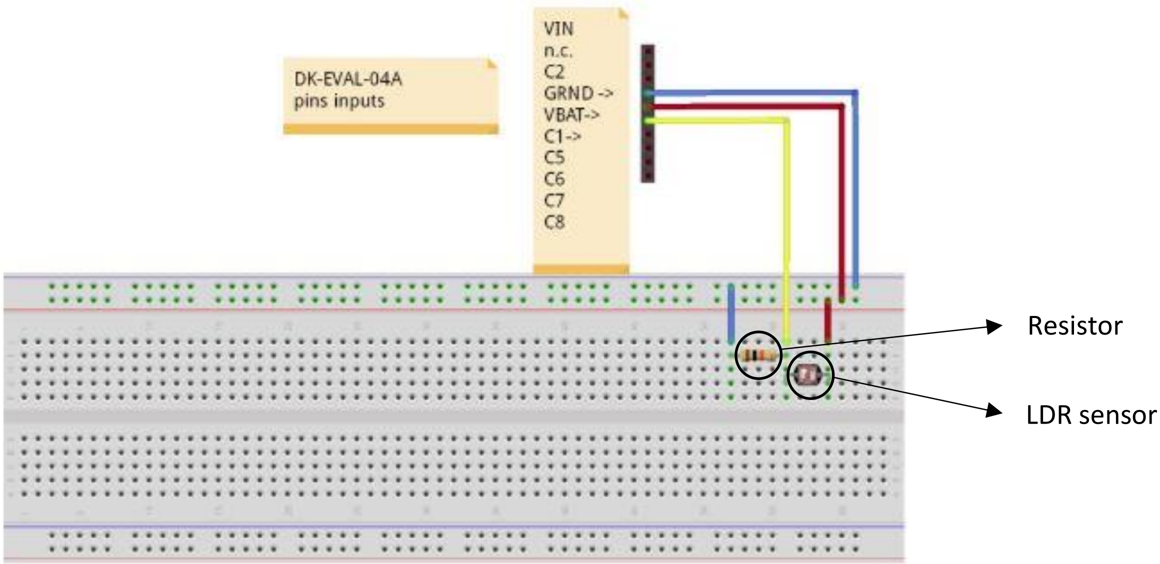


Figure 4.3: Connection of LDR with IQRF in 2D resolution.

This custom DPA handler has been uploaded for modifying the functioning behavior of the transceiver beyond that provided by the hardware profile along with the operating system. Custom DPA handlers are programmed ‘C’ files which can be later compiled to hex and uploaded to the TR modules. A resistor is needed with the LDR sensor for better range of readings. The connections are shown in Fig. 4.2 and Fig. 4.3 respectively.

4.1.1 Light Dependent Resistor (LDR)

LDR, also known as light dependent resistor is a component which has a variable resistance that changes with the intensity of light that falls upon it. Here, light intensity refers to the strength of light, produced by a specific lighting source. It is the measure of the wavelength-weighted power emitted by a light source and varies depending on the lighting source. There are specific high and low light intensity fixtures, lamps, and bulbs for experiments. For example, high-intensity discharge lamps emit a high light intensity, while fluorescent lamps are considered a cool or low-intensity light source. The unit for light intensity is called LUX. Lux is the unit of illuminance which is equal to 1 Lumen per square meter. Lumen is the international SI unit for luminous flux, a measure of the total quantity of visible light emitted by a source [35]. The relationship between Lumen and LUX has been demonstrated in Fig. 4.4

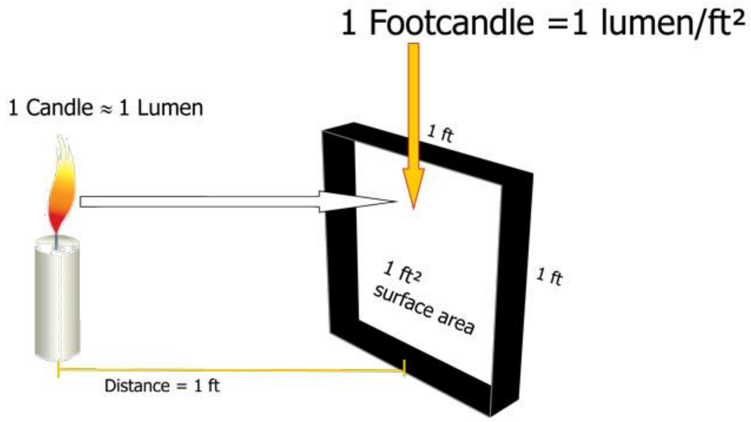


Figure 4.4: Relationship of Lumen and LUX [35].

Every light source contains a color temperature. It is one of the characteristics of visible light. This color temperature has different applications for lighting, photography, videography, publishing, manufacturing and many more. The unit for this color temperature is Kalvin (K). For experimental purpose, 2700K color temperature room lights have been used. Fig. 4.5 shows different color temperature lighting.

Figure 4.5: Different color temperature lights [36].

LDR is mostly used in the light sensing circuit. The main working principle of an LDR is very similar to a resistor, but its resistance depends on the lighting. When light falls over it, the resistance reduces gradually, but at the dark time the resistance keeps increasing up to a certain point. The resistance of an LDR may typically have the following resistances:

Bright Light = 150Ω

Darkness = 20000000Ω

Fig. 4.6 shows a curve how the resistance of an LDR when its exposed to lighting. Examining the Table 4.1, it can be observed that the LUX value decreased when the resistance value increases. So, it is proved that the LUX value shows an inverse relationship with the resistance value.

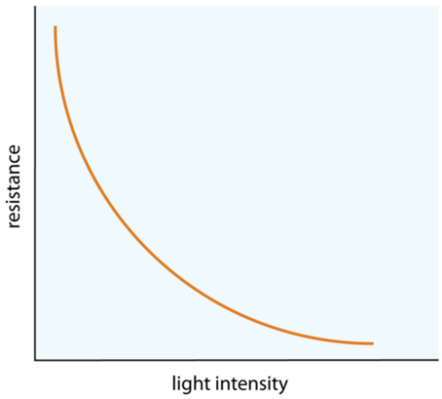


Figure 4.6: Resistance vs. Light intensity.

Lamp Distance	LUX	Ohm
9cm	3072.125	179
10cm	2640.125	193
11cm	2320.125	211
12cm	1952.125	222
13cm	1584.125	239
14cm	1360.125	249

Table 4.1: Results of resistance vs LUX value.

4.1.2 Network Setup and LUX Calculation

The C1 pin of DK-EVAL-04 module was used as input PORTA.0. This pin needs to be activated by sending a simple command from the DPA Test terminal of the IQRF IDE. The command's configuration should be as following,

NADR = 0001, PNUM = 09, PCMD = 00, HWPID = FFFF, Data = 00.01.01.

More details have been described in Scenario 3B – Single-Hop Communication. In this way, the user can activate the C1 pin to act as an input for the LDR data.

After that, the user needs to configure the C1 pin in such a way, so that it can show its status and deliver the data to the IQRF packet inspector window according to the uploaded custom DPA handler. For doing this, the user needs to send another command to the module. The corresponding configuration should be as following,

NADR=0x0001, PNUM=0x21, PCMD=0x00, HWPID=0xFFFF

Now, the device is ready to take LDR reading and deliver it to the packet inspector window of the IQRF IDE, as a hexadecimal number. The user needs to set up the combined circuit and the IQRF module to the desired lighting position, and upon hitting the send button from IQRF IDE, the user can see the light intensity value in the screen in hexadecimal value. See Fig. 4.7 for reference. From the figure, it is observed that the values are given in hexadecimal and in two different parts. Part number 0 is representing the MS Bytes, and part number 1 is representing the LS Bytes. From Table 4.2, the user can calculate the high-resolution values of light intensity.

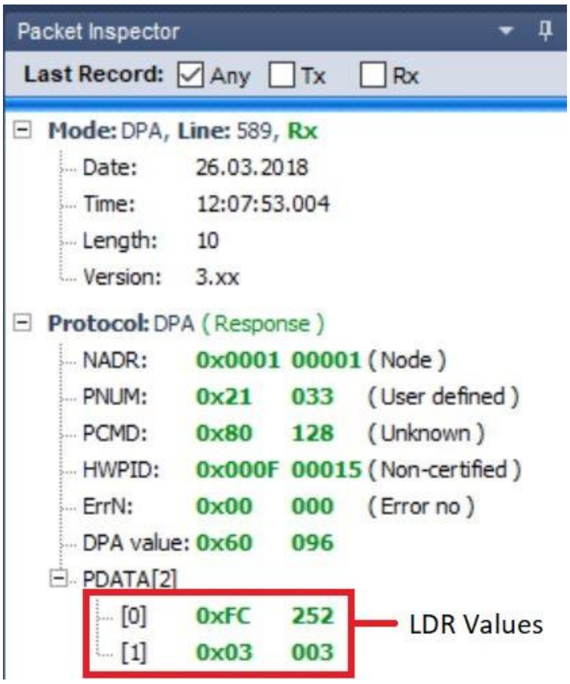


Figure 4.7: Light intensity value in hexadecimal.

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
LS Byte	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
MS Byte	S	S	S	S	S	2^6	2^5	2^4

Table 4.2: LDR calculation table.

Calculation of Light Intensity Value (LUX)

From Fig. 4.7, we can see that the MS and LS bytes are FC and 03 respectively. The values in decimal which are 252 and 03 are there only for noticing the value change for light intensity. It is hard to convert a hexadecimal number to find the decimal value every time. For this reason, the decimal conversion is there. If the intensity of the light is higher, the hexadecimal numbers will be revealed in a higher form giving the user a higher decimal value and vice versa if the intensity value is low.

To get the actual intensity value, the user must convert the hexadecimal number to binary. Otherwise, the values cannot be put in the LDR calculation table. For example, in the above case, the hexadecimal numbers are FC and 03. Converting them into binary becomes:

F – 1 1 1 1

C – 1 1 0 0

0 – 0 0 0 0

3 – 0 0 1 1

In this way, the user gets a 16-bit value. Now, these bits need to be put in the table to get the actual light intensity. After putting the values to their corresponding places, the user needs to add the whole thing. The whole calculation is shown below.

(2^11*1) + (2^10*1) + (2^9*1) + (2^8*1) + (2^7*1) + (2^6*1) + (2^5*0) + (2^4*0) + (2^3*0) + (2^2*0) + (2^1*0) + (2^0*0) + (2^-1*0) + (2^-0*0) + (2^-3*1) + (2^-4*1) =

2048 + 1024 + 512 + 256 + 128 + 64+ 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0.125 + .0625 =

4032.1875 LUX.

So, the light intensity value for FC 03 is 4032.1875 LUX.

4.2 Scenario 4A – Distance-based Measurement and Analysis

For this scenario, a table lamp has been used along with the IQRF and LDR circuitry. At first, the distance between the LDR and the light source has been moved horizontally for finding the lighting intensity, keeping the LDR position fixed. The experimental setup is shown in Fig. 4.8. The light intensity vs. light source position is given in Table 4.3.

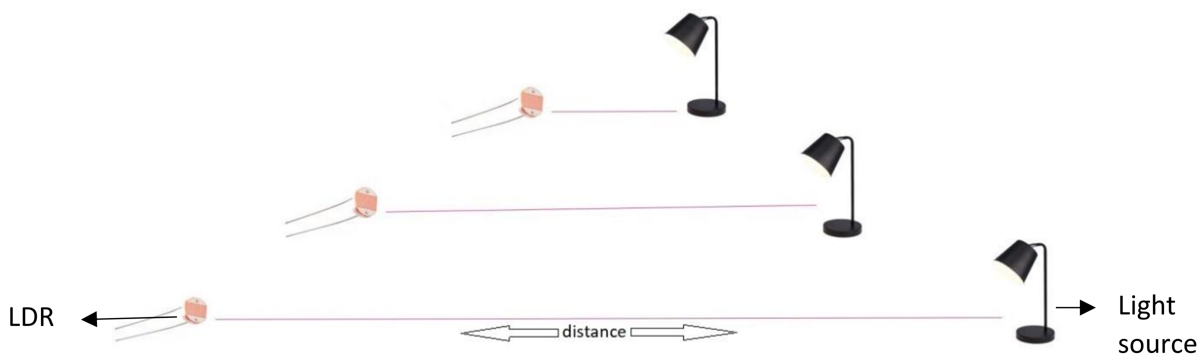


Figure 4.8: Increasing distance between light source and LDR.

Distance from Light (cm)	Lux in Hex	Lux in Binary	Lux in Decimal
12	CF 00	1100 1111 0000 0000	3312
15	A9 00	1010 1001 0000 0000	2704
16	8D 00	1000 1101 0000 0000	2256
17.5	6E 00	0110 1110 0000 0000	1760
19.5	51 00	0101 0001 0000 0000	1296
21	33 00	0011 0011 0000 0000	816
23	12 00	0001 0010 0000 0000	288
26	FA 00	1111 1010 0000 0000	4000
30	DB 00	1101 1011 0000 0000	3504
32	BF 00	1011 1111 0000 0000	3056

Table 4.3: Experiment result for light intensity vs. lighting position.

From the results presented in Table 4.3, it can be observed that the intensity of the light differs when the light source is moving. A graphical representation of the above table is shown in Fig. 4.9. The sudden increase of the intensity value from the graphical figure and the table has been explained in the comment section of Table 4.6

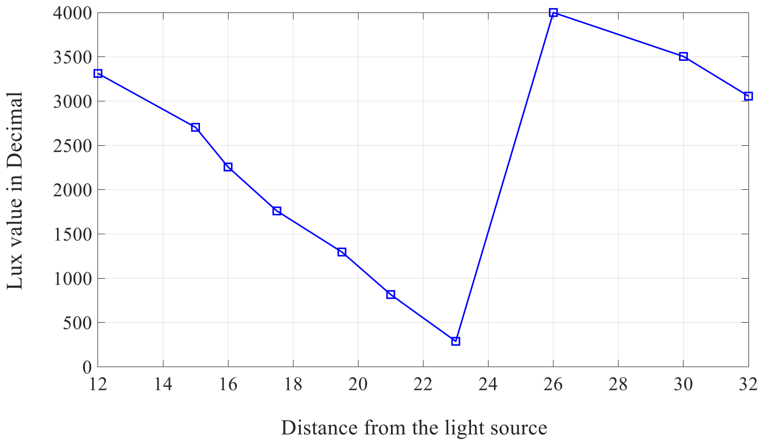


Figure 4.9: Graphical representation of Table 4.3.

Color Testing

For color testing, 5 different color LED has been used. See Fig. 4.10 for reference. For this experiment, the LDR position was fixed, and the LEDs were moved vertically. The results we obtained from this experiment has been given in Table 4.4.

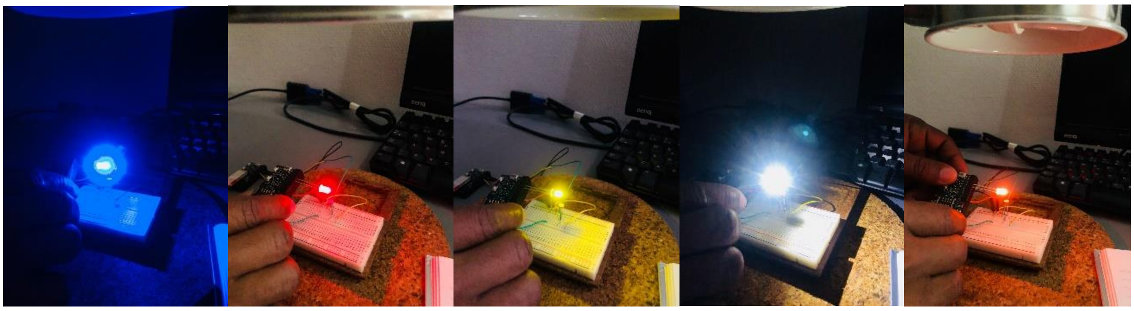


Figure 4.10: Color testing with different colored LEDs.

Distance (cm)	Red Led (LUX)	Green Led (LUX)	Yellow Led (LUX)	Blue Led (LUX)	White Led (LUX)
3.5	3152	3104	2448	3536	1472
4	2000	1664	1984	2896	1344
4.5	1712	1168	2176	2048	2912
5	1296	912	1584	1600	1712
5.5	1632	672	1440	1296	3440
6	1216	544	1200	992	2512
6.5	992	432	864	864	2416
7	784	416	816	816	2272
7.5	880	336	704	672	1616
8	736	384	672	656	1360
8.5	720	288	576	656	1296
9	624	352	448	640	1040

9.5	512	256	384	608	1120
10	416	256	464	480	896

Table 4.4: Experiment results for different color sources.

From this experiment, it can be concluded that LDR cannot recognize colors. It gives readings based on the brightness of the light source. A graphical representation of the above table is shown in Fig. 4.11. From Fig. 4.11, we can see that white color LED gives the highest readings as it is the brightest followed by blue, red, yellow and green. For this experiment, the same power source was used for all the LEDs.

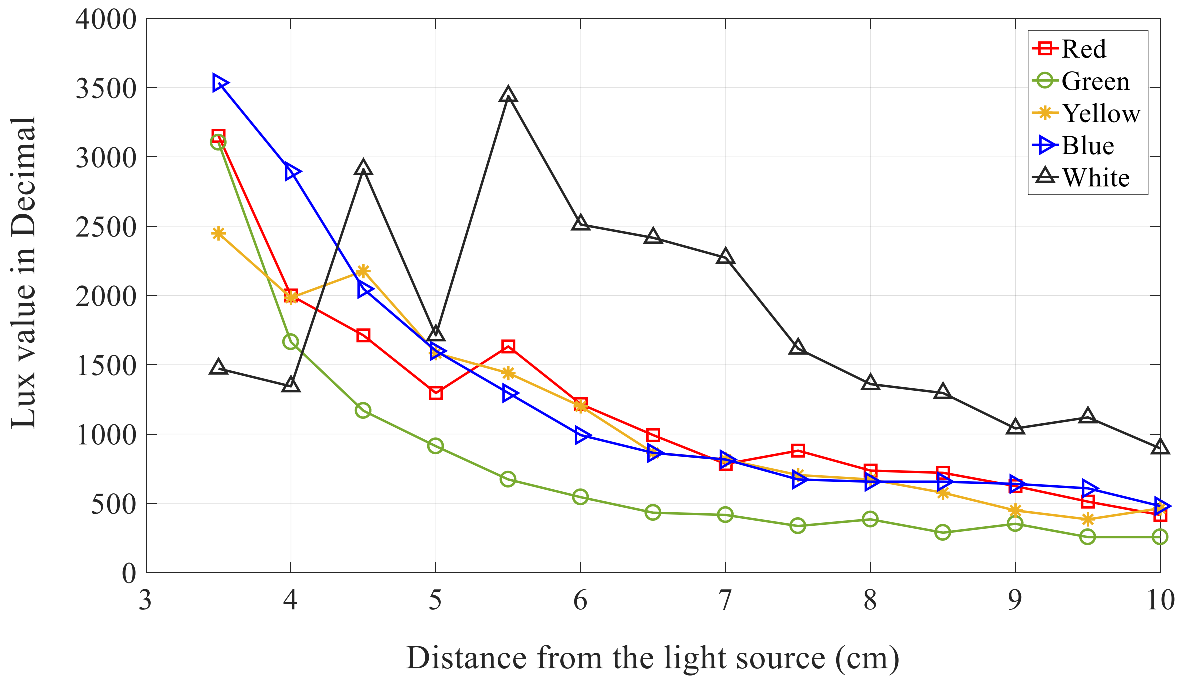


Figure 4.11: Graphical representation of Table 4.4.

4.3 Scenario 4B – Effect on Angle of Arrival

The angle of arrival (AoA) of a light measurement is a method for determining the direction of wavelength of a light source. For this experiment, LDR sensor has been used along with the IQRF to determine the angle of arrival, of a light source. We determined the angle using cosine equation. From trigonometry functions, we know that cosine of an angle is equal to $\frac{Adjacent\ distance}{Hypotenuse\ distance}$ of a right triangle. See Fig. 4.12 for reference.

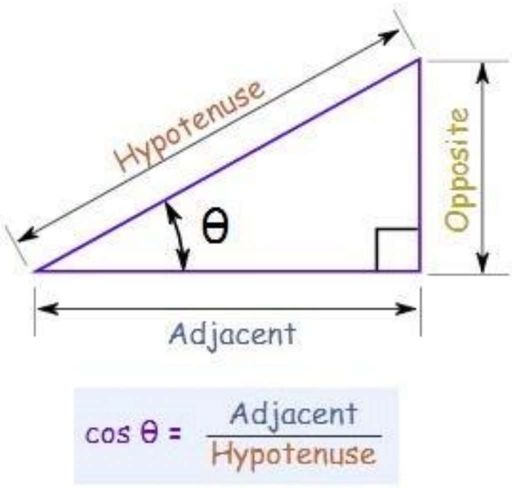


Figure 4.12: Cosine equation.

For measuring the adjacent and hypotenuse, the LDR sensor has been placed in the middle of a protractor printed in a paper. This will help to know the angle of the lighting position as well. Fig. 4.13 explains how the experiment has been conducted.

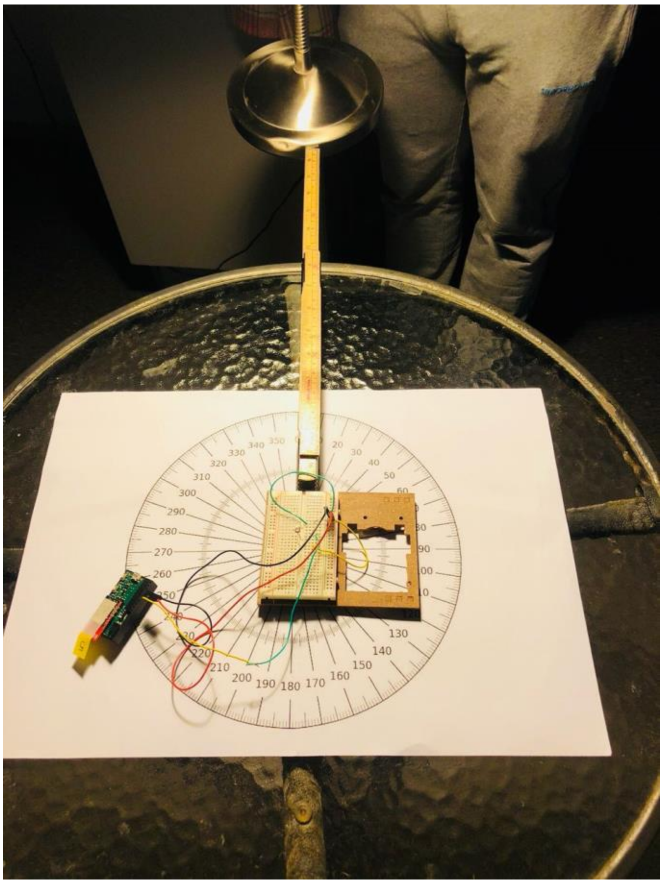


Figure 4.13: Conducting the angle of arrival experiment.

By keeping the LDR position fixed, the light source has been moved in a horizontal direction. The measurements were gathered very accurately. The whole experiment was divided into two sub-categories. First, keeping the adjacent and hypotenuse distance same but changing the angle of light source. Second, horizontally moving the light source, resulting in both adjacent and hypotenuse change. Both are explained below.

Fixed Adjacent and Hypotenuse Distance with Changing Angle

As mentioned, the LDR was in the middle of a protractor, and the angle was changed. See Fig. 4.14 for reference

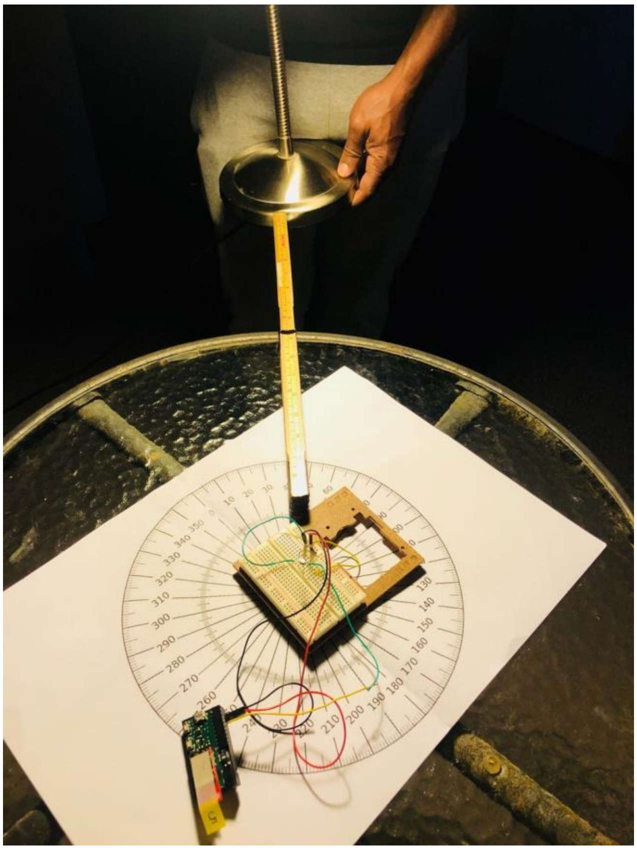


Figure 4.14: Fixed distance with changing angle.

As seen from Fig. 4.14, the position of light source was changed position keeping the adjacent and hypotenuse distance same very accurately. The results acquired from this experiment are given in Table 4.5.

Angle of Light (degree)	Adjacent Distance (cm)	Hypotenuse Distance (cm)	Angle of Arrival (degree)
0	45.72	55.88	35.09
20	45.72	55.88	35.09
40	45.72	55.88	35.09
60	45.72	55.88	35.09
80	45.72	55.88	35.09
100	45.72	55.88	35.09
120	45.72	55.88	35.09
140	45.72	55.88	35.09
160	45.72	55.88	35.09
180	45.72	55.88	35.09
200	45.72	55.88	35.09

Table 4.5: Experiment result with fixed adjacent and hypotenuse distance with changing light source angle.

From the results, it can be observed that when the adjacent and hypotenuse distance along with the LDR position is fixed, the angle of arrival, of light will be same for all angles.

Moving the Light Source in a Horizontal Direction

For this experiment, the LDR was fixed in a position, and the light source was moved horizontally. See Fig. 4.15 for reference.

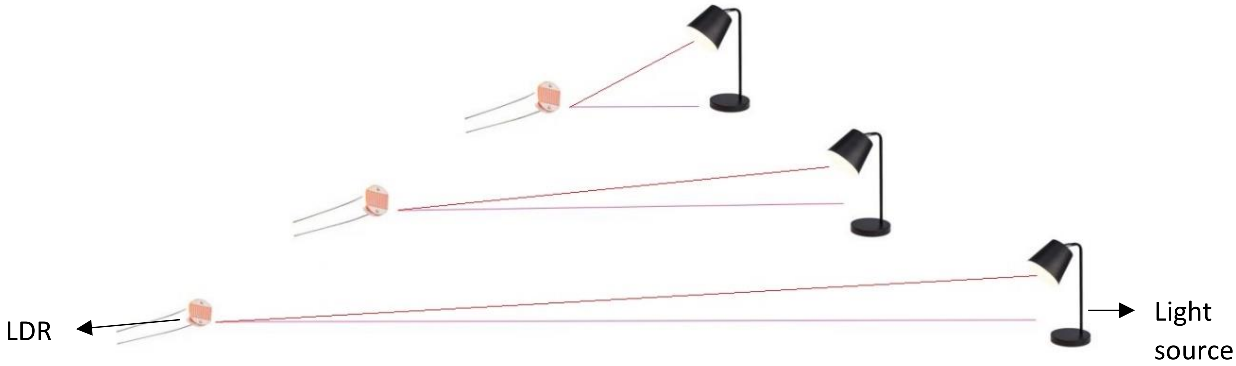


Figure 4.15: Moving the light source horizontally.

Table 4.6 shows the result for this experiment.

Hypotenuse Distance (cm)	Adjacent Distance (cm)	LUX in Decimal	Angle of Arrival (degree)
22	12	3312	56.94
24	15	2704	51.32
25	16	2256	50.21
26	17.5	1760	47.69
27	19.5	1296	43.76
28.5	21	816	42.54
30	23	288	39.94
32	26	4000	35.66
33	30	3504	24.62
34.5	32	3056	21.95

Table 4.6: Experiment result when light source is moved horizontally.

From the results, it can be observed that when the light source is changing position horizontally, the intensity and the angle of arrival changes. But at one point, the light intensity value rises again. So, in conclusion, it can be said that when the light source is at 35.66 degrees angular position with the LDR sensor, the intensity will be highest.

CHAPTER 5

CLOUD-BASED TESTBED FOR IQRF SENSORS

IQRF is also compatible with cloud servers. IQRF Alliance has made a cloud server https://cloud.iqrf.org [37]

Using the cloud server, is it possible to control the IQRF nodes. But for this, a gateway is needed. To interface between IQRF network and other networks such as cellular systems, gateway allows connection to the internet and allowing remote monitoring, data collection, and control of devices in IQRF Network. IQRF alliance has provided the users with three types of gateways. For experimental purpose, a GSM gateway has been used. This chapter contains a brief discussion about the client-server type of communication, How the network is configured, the configuration of gateway and connectivity, temperature and intensity measurements using the cloud server.

5.1 Client-server Type of Communication

Fig. 5.1 shows the working principle of IQRF cloud service. The most demanded internet connectivity by the remote gateway is possible in IQRF via the cloud. IQRF cloud is a server, providing plug and play service worldwide to access end devices by high-level methods which the users are used to such as application program interface (API) and web interfaces, database, etc.

The basic communication component for the experiments is a TR-72DA RF transceiver. The microcontroller unit inside it is equipped with IQRF operating system, implementing wireless communication, mesh networking and a lot of other functionalities. IQRF designed gateways with such kind of devices implemented inside it.

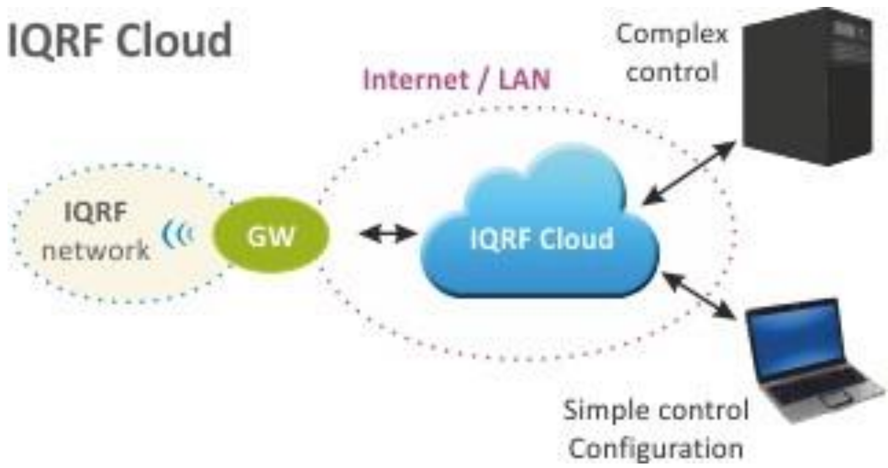


Figure 5.1: IQRF cloud service [33].

IQRF designed with 3 kinds of gateways which are Wi-Fi, GSM, and Ethernet gateways. All the gateways are equipped with a TR-72DA transceiver unit inside it. The transceiver unit maintains all the communication with the node modules while the gateway uses internet connectivity to maintain the communication with the IQRF cloud. Basically, the gateway needs to be configured through IQRF IDE first. For these experiments, a GSM gateway *GW-GSM-02A* have been used. Fig. 5.2 shows the gateway.

Figure 5.2: IQRF GSM gateway GW-GSM-02A [33].

When user sends a command from the cloud, the command first goes to the IQRF server which is located in *Czech Republic*. From the server, the command is uploaded to the gateway. Clearly, the gateway needs an active SIM (Subscriber Identity Module) card with internet connectivity to acquire the command from the server. The gateway then delivers the command and informs the TR module inside it to act accordingly. For example, there are 5 nodes connected to the TR module of the gateway. These nodes can be connected either in single-hop or in multi-hop communication mode. Node number 1 has been asked to give its surrounding temperature. So, the message will be conveyed to node number 1, and the node will comply accordingly and send the data to the TR module of the gateway. The TR module will then send the response to the gateway and the gateway uploads the response in IQRF cloud. The user can see the data from the cloud server. This whole process can be done in less than 10 seconds. User can set up the timing while configuring the gateway through IQRF IDE. The minimum timing for the whole process cannot be lesser than 10 seconds, but the user can increase the timing.

5.2 Latency Requirement of Narrowband IoT

In this section, we briefly consider requirements of the 3GPP Rel-13 narrowband system, named Narrowband Internet of Things (NB-IoT) to show that the performance of the cloud-based IQRF testbed can confirm the latency requirements of IoT applications. NB-IoT is a new radio access system built from existing LTE functionalities. Ultra-low complexity devices to support IoT applications, improved indoor coverage of 20 dB, support of a massive number of low-throughput devices, improved power efficiency and exception report latency of 10 seconds or less are main performance objectives of NB-IoT [38]. In [39] it is stated that the latency of 10 seconds or less is the target for 99% devices [39] is required. Therefore, in this study, the cloud-based control process latency is well under the stated goal of 10 seconds in IoT.

5.3 IQRF Configuration for GSM Gateway and Connectivity

Upon connection, the IQRF IDE recognizes the gateway, and it shows the connected device in the left corner side of the IQRF IDE (See Fig. 5.3 for reference). Then using the IQMESH network manager of IQRF IDE, user needs to bond the nodes with the TR module inside the gateway. (See Fig. 5.4 for reference).

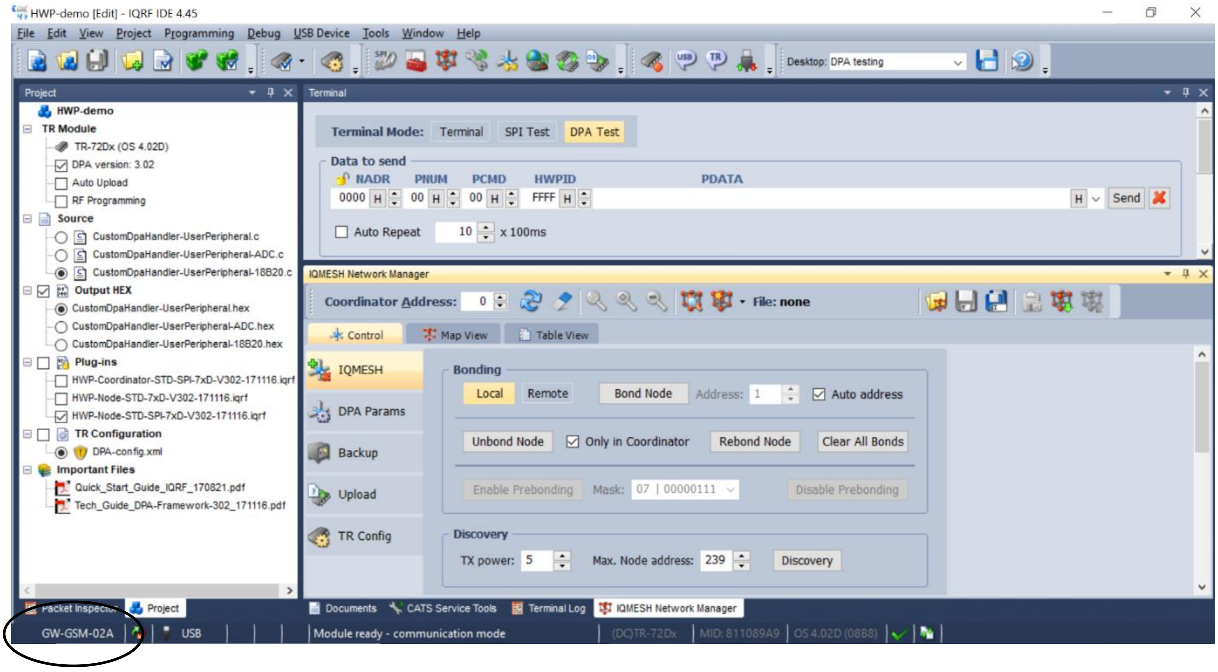


Figure 5.3: Gateway GW-GSM-02A is connected to IQRF IDE.

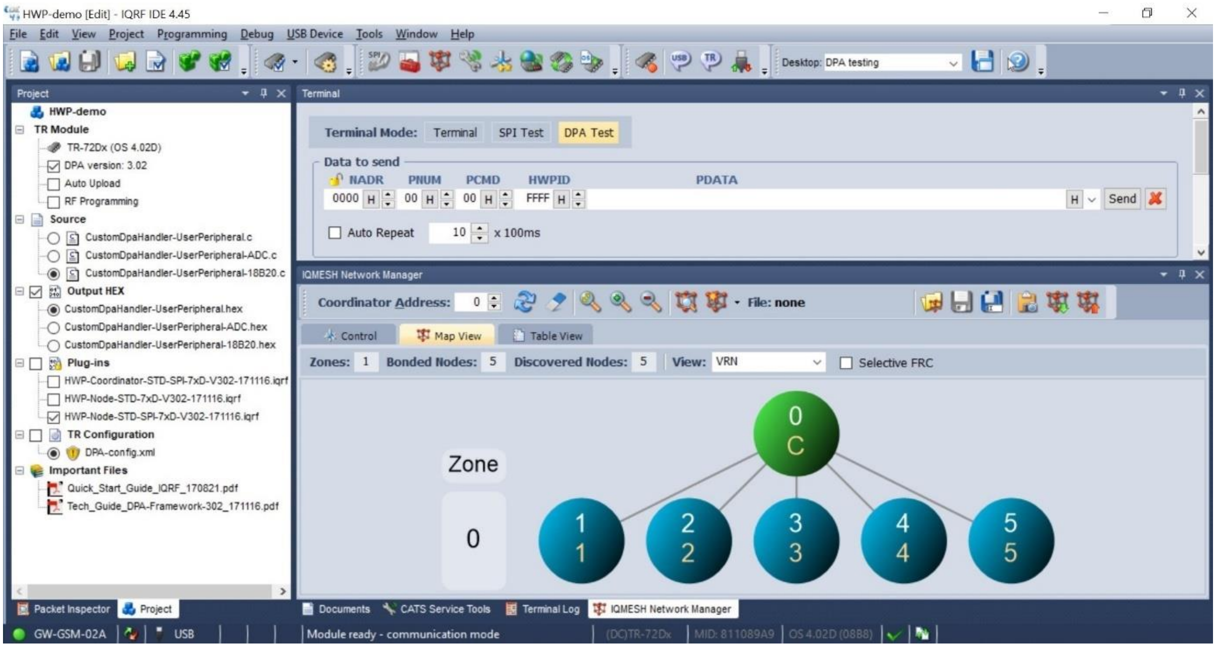


Figure 5.4: With IQMESH network manager, nodes are bonded to the built-in TR module of the gateway.

After all the nodes are bonded, it is required to configure the access point of the SIM card to the gateway. Note that the access point name (APN) of the SIM card must be confirmed, to get proper connectivity. Only after connecting to a proper gateway from IQRF, the *GW Tool* section of IQRF IDE can be accessed.

Therein, a user can check and modify basic things such as the gateway status, firmware version, received signal strength indicator (RSSI), response time, the access point, IP address, etc. In that window, the user can also find the gateway ID, which is very important because after logging in to IQRF cloud, the user needs to put the gateway ID over there for the server to recognize the connected gateway. Fig. 5.5 and Fig. 5.6 shows the *GW Tool* section of IQRF IDE. For these experiments, a SIM card from *One call*, which is a part of the Telia Norge AS telecom company in Norway, has been used with the APN '*internet*'.

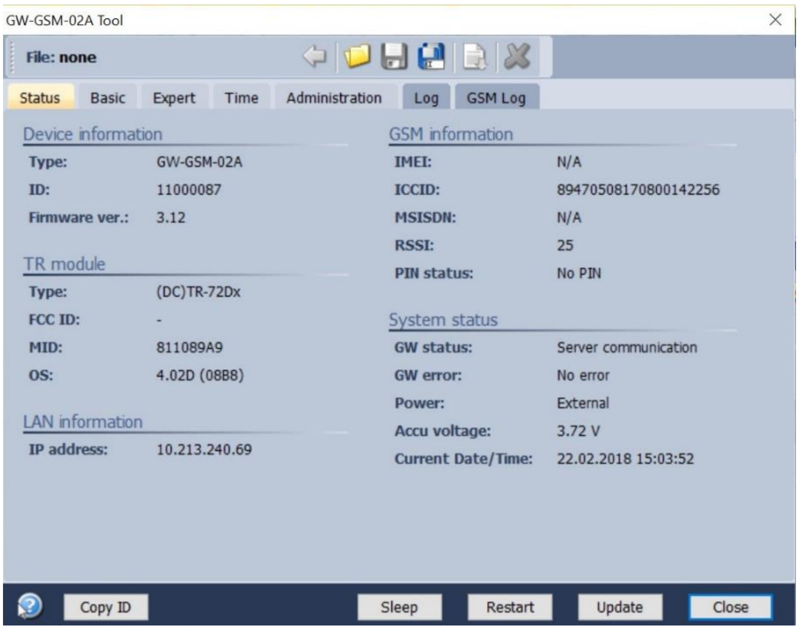


Figure 5.5: 'GW Tool' section of IQRF IDE – Status window.

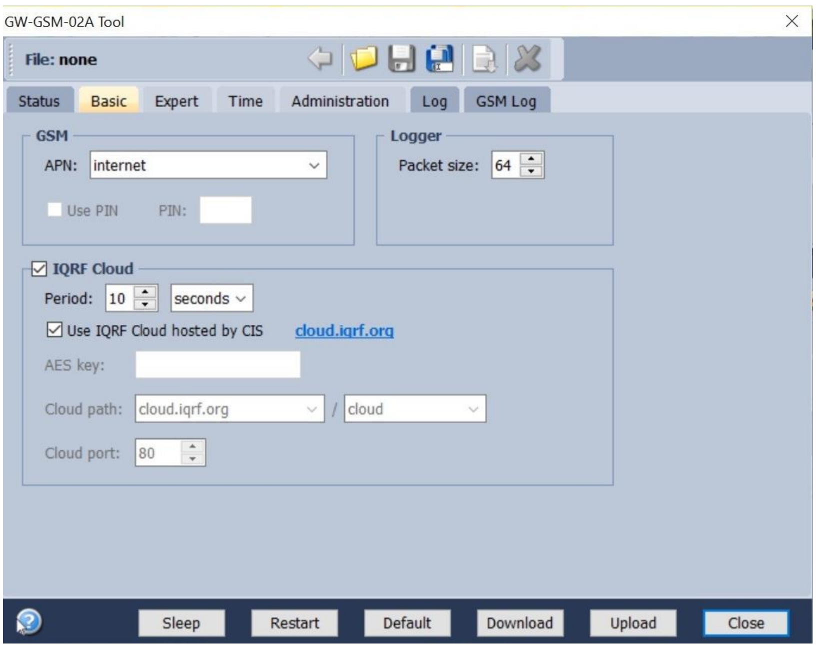


Figure 5.6: 'GW Tool' section of IQRF IDE – Basic Window.

After configuring the gateway with IQRF IDE, the user can log in to cloud service from https://cloud.iqrf.org [37]. The user will be asked to register with a valid email address. After registration, a window will appear asking for the gateway ID. How to get the gateway ID, has been described in the previous paragraph.

Now the user can send command directly from the cloud server. There is a link *View data/Send Command* under the data part of the cloud window. Clicking that a new window will appear where the user can see all the commands and responses which has been sent and received from the gateway. The process is shown in Fig. 5.7 and Fig. 5.8.

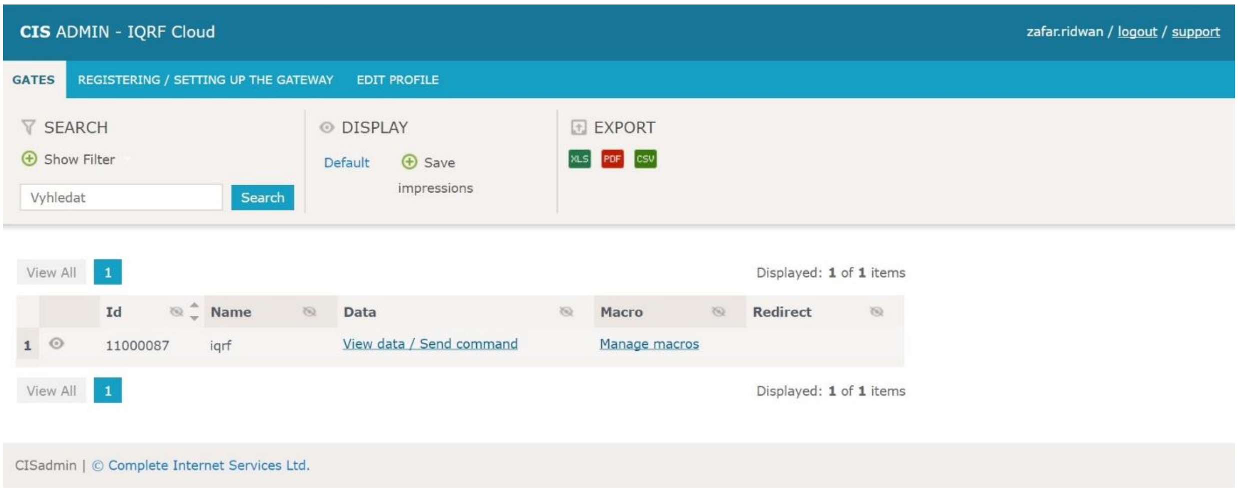


Figure 5.7: IQRF cloud window after logging in.

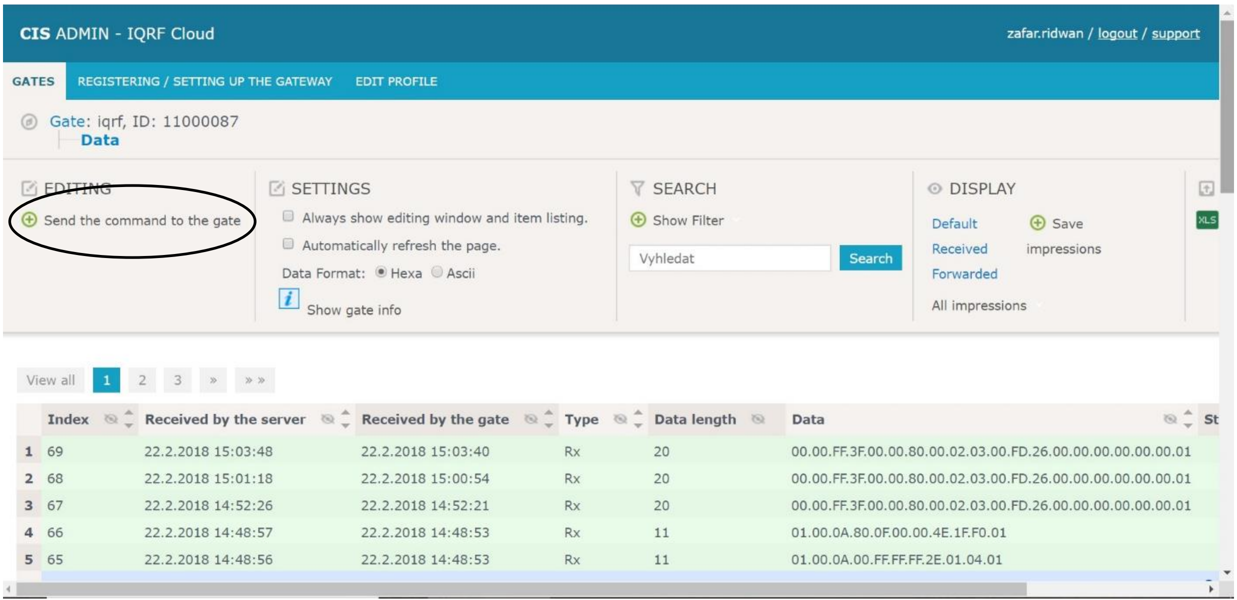


Figure 5.8: IQRF cloud command sending and response reviewing.

From Fig. 5.8, we can observe that the command has been sent and has received by the gateway and for sending new commands, the user needs to click the *Send the command to the gate* link. The new window will be shown in Fig. 5.9. The command to the gateway needs to be sent in the hexadecimal form. In the data section, the user needs to insert the correct commands. The command in Fig. 5.8, is asking node number 1 to give its surrounding temperature value of the corresponding location. The whole command is described below.

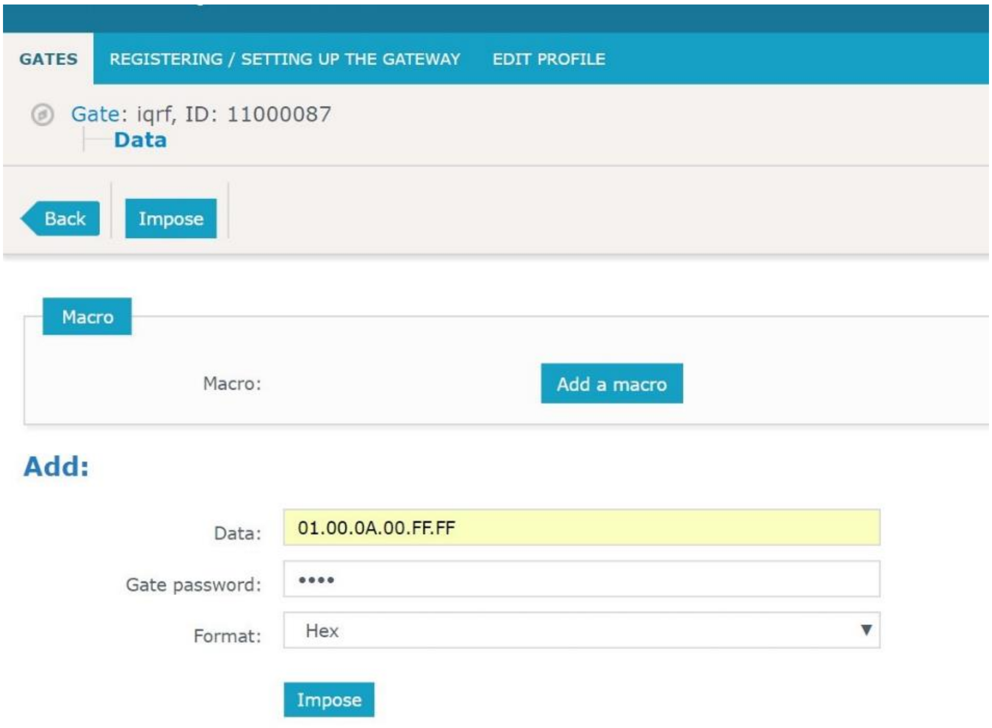


Figure 5.9: IQRF command sending terminal.

The digits in the *Data* box is the command for asking node number 1 to illustrate its surrounding temperature. The command is in hexadecimal and is described underneath.

1st and 2nd Digits (01.00) – The first 2 columns are used to call up the nodes. Basically, these are the node addressing. Setting up 01.00 will call up the node number 1 in the network.

3rd Digits (0A) – Every node comes with a list of predefined peripherals. The commands for these peripherals can be accessed from this section. Fig. 5.10 shows the list of commands. Here putting the number '00' will select the peripheral *coordinator* from the list. Likewise, putting 01, 02, 03, 04 will select *node, OS, EEPROM, EEEPROM* respectively. In this way, '0A' will set the peripheral *thermometer*.

Figure 5.10: Predefined peripherals.

4th Digit (00) – All the peripheral in the above list has some predefined commands. For example, putting the number '00' will ask the *thermometer* peripheral to get the temperature. Fig. 5.11 confirms the statement.

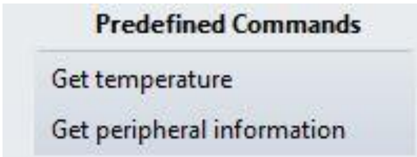


Figure 5.11: Predefined commands.

5th and 6th Digits (FF.FF): Inside and IQRF network there can be several devices connected. They are known as hardware profiles. There are some predefined hardware profiles which can be accessed from this section of the command line as shown in Fig. 5.12.

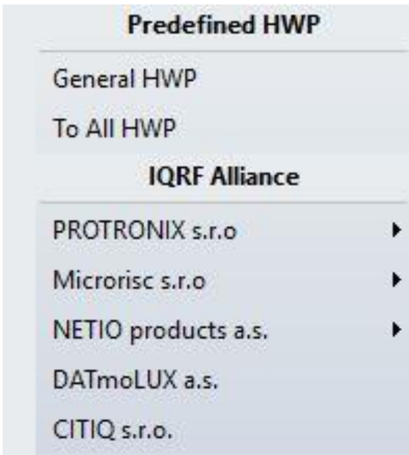


Figure 5.12: Predefined hardware profiles.

Now, for sending extensive commands, the user can put them after finishing up the main command lines just like the mentioned way. The user also needs to put the gateway password, in this case, the built-in password ‘iqrf’ has been used. The user can also change the password from the *GW Tool* section. After finishing the above instruction, the user needs to hit the *impose* button for sending the command to the gateway and the nodes will comply accordingly. See Fig. 5.13 for sent and received commands.

View all 1 2 3 » »»

	Index	Received by the server	Received by the gate	Type	Data length	Data
1	71	22.2.2018 15:07:30	22.2.2018 15:07:25	Rx	11	01.00.0A.80.0F.00.00.5C.19.98.01
2	70	22.2.2018 15:07:30	22.2.2018 15:07:25	Rx	11	01.00.0A.00.FF.FF.FF.00.01.04.01
3	38	22.2.2018 15:07:16	22.2.2018 15:07:20	Tx	6	01.00.0A.00.FF.FF

Figure 5.13: Sent and received commands from cloud server.

5.4 Scenario 5A – IoT Setup for Temperature Measurements

From Fig. 5.13, a command in hexadecimal number was sent, and we received 2 separate responses which are also in hexadecimal form under the data section of the cloud window. Decoding this response, the user can easily find out the temperature. The first response contains the temperature reading. The respond command containing the number ’19’ is the temperature header in hexadecimal. Converting ‘19’ into decimal gives us the number ‘25’. See Fig. 5.14 for reference. So, the surrounding temperature of node number 1 was 25 degrees.

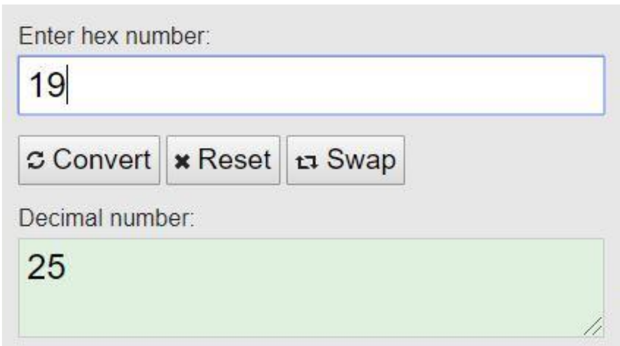


Figure 5.14: Conversion of hexadecimal to decimal (Temperature).

5.5 Scenario 5B – IoT Setup for LDR Measurements

With a GSM gateway, it is also possible to operate the IQRF and LDR circuitry from the cloud. In this section, the calculation procedure of readings will be shown.

For collecting the data from the LDR circuit, the C1 pin of the DK-EVAL-04 module needs to be activated using a command in hexadecimal number. After that, another command needs to be sent for asking the C1 pin to reveal the data to the cloud interface. The below commands are sent respectively.

01.00.09.00.FF.FF.00.01.01

01.00.21.00.FF.FF

The first command is for activating the C1 pin and the second command is sent for asking the C1 pin to reveal the command in the cloud window. For experimental purpose, node number 1 is asked to reveal the light intensity value of its surrounding. After sending the commands, the user will get a reply from the C1 pin in hexadecimal. The reply message is shown below.

01.00.21.80.0F.00.00.3D.95.02

From this message, the last two positions are representing the light intensity value in hexadecimal which is 95 and 02. These are the MS and LS bytes respectively. Later these hexadecimal values need to be converted into decimal and calculating using the intensity measurement table for getting the actual decimal values of lighting intensity. The whole procedure has been explained in subsection 4.1.2 in chapter 4. The real-life experiment results are shown in Fig. 5.15.

22	119	2.5.2018 0:07:34	2.5.2018 0:07:30	Rx	8	01.00.09.80.0F.00.00.40	
23	118	2.5.2018 0:07:34	2.5.2018 0:07:30	Rx	11	01.00.09.00.FF.FF.FF.38.03.04.03	
24	67	2.5.2018 0:07:21	2.5.2018 0:07:25	Tx	9	01.00.09.00.FF.FF.00.01.01	Confirmed
13	125	2.5.2018 0:15:26	2.5.2018 0:15:22	Rx	10	01.00.21.80.0F.00.00.3D.95.02	
14	124	2.5.2018 0:15:26	2.5.2018 0:15:21	Rx	11	01.00.21.00.FF.FF.FF.3A.03.04.03	
15	70	2.5.2018 0:15:15	2.5.2018 0:15:16	Tx	6	01.00.21.00.FF.FF	Confirmed

Figure 5.15: Light intensity values in cloud server window.

From Fig 5.15 we can also observe that the commands sent from the cloud and received by the gateway have been represented by Tx and Rx respectively. A *Confirmed* message is displayed when the command is transmitted successfully.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

This is the concluding chapter of this thesis. In this chapter conclusion, contributions and future work have been discussed. In the conclusion part, why IQRF is better is better than other devices as well as the uniqueness of IQRF have been described. In contribution part, the summary of contribution of this thesis has been defined. The future work will contain the parts which have not been covered in this thesis.

6.1 Conclusions

IQRF platform meet their objective which is an easy-to-merge radio frequency transceivers and receivers to form a strong WSN. It has custom DPA handler with many ready to use codes provided by the manufacturer, that makes the device plug and play to any kind of sensor and establish a communication to a WSN.

It has different kinds of modes where most of the functionalities can be switched off. The IQRF have different sleeping modes with low power and ultra-low power configuration for receiving and transmitting mode, reducing the use of the battery power significantly. The IQRF modules used for this thesis does not contain any external antennas. Still, in comparison with some other devices like ZigBee, IQRF range is greater. The built-in antenna gives a great coverage. It is possible to achieve even higher range, using a transceiver module with built-in printed circuit board (PCB) with an external antenna provided by the vendor.

One of the strongest features of IQRF is to build a strong network structure supporting a high number of nodes that IQRF manufacturer provide the users. Nowadays, the increasing amount of interconnection between devices is more demanded by the industry and end users. Thus, IQRF is a great solution to that. Discussing about limitations, IQRF has fixed low bit rate. But, it also makes it easier to expand the network and collect data from large number of nodes in a short time. It may not be suitable for many applications, but this is the trade-off this company has bet for. This technology can contribute around IoT due to the connection of the nodes to the internet via a node working as a gateway.

IQRF DDC-SE-01 module contains four kinds of sensors which are Dallas 1-wire temperature sensor, LDR sensor, potentiometer and LDO regulator sensors. For this thesis, that module was not used. This kind of modules needs to be connected to a DK-EVAL-04 module. All these sensors can also be merged externally and the results they will stretch will be accurate. The result from the temperature sensor was much more accurate as they were placed in places like inside a fridge or on the top of a heater. Later those temperatures were measured with external thermometers, and the results were the same. Moreover, IQRF was providing high-resolution data which is not possible for an external thermometer. The LDR and cloud-based experiments were also accurate as for the light intensity measurement experiment, both technologies provided the same kinds of results. Finally, it can be said that using IQRF, a lot of experiments are possible, and they can have very reliable data processing.

These experiments, e.g., light intensity measurement and angle of arrival determination can be used for indoor localization. Data (intensity of light, the angle of arrival, sensor temperature, etc.) collected from IoT devices using the IQRF cloud can be used for further post-processing (for data analytics).

6.2 Contributions

The main contributions of this thesis can be summarized as follows:

- Mainly, this thesis investigates the applicability of IQRF technology as a promising technique for the future IoT networks. Regarding this, we set up real-life testbeds to collect environmental factors such as temperature by using the in-built temperature sensors. Therein, we investigate on how to connect IQRF sensors in a multi-hop manner to extend the connectivity of a sensor network with low power devices.
- Despite of having built-in sensors, additional LDR sensor has been merged externally for experimenting the accuracy, effectiveness, and possibility of IQRF. With the help of LDR sensor, the light intensity of the surrounding and angle of arrival of a light source can be measured very accurately which can be later used for indoor localization program.
- In addition, GSM gateway is studied and used to build a connectivity interface between IQRF devices and the rest of the world. Therein, the most demanded gateway function

which is to collect events and data is tested, and an arrangement which can access sensor data and control them from internet via IQRF cloud is set up.

6.3 Future Work

Since the gateway was ordered additionally, it took some time to arrive from the manufacturer. Therefore, some advanced functionalities of the gateway could not be tested completely. But all the functionalities that could be done using a coordinator, connected to a PC, can be done using the gateway. Almost all the features can be covered from the cloud server.

For short-term experimental purpose, only one external LDR sensor was merged with IQRF module. But other sensors like external relay, digital input button, hall sensors, motion detectors, etc. can be easily be connected to IQRF DK-EVAL-04 module and can be operated either from coordinator or from the cloud server. Remote switching for devices is also possible.

Home automation has already been implemented using IQRF, but the applications can be further improved for more accuracy and efficiency. IQRF can also be merged with 3rd party devices like Raspberry Pi and Arduino using MQTT and JSON protocols. It can be also used from any 3rd party cloud server, not only from the one IQRF has provided.

References

[1] [Online] Available: https://www.quora.com/What-is-wireless-sensor-network-WSN-technology, Accessed on 1 May 2018.

[2] P. S. Lingam, S. Parthasarathi, and K. Hariharan, “Energy Efficient Shortest Path Routing Protocol for Wireless Sensor Networks”, International Journal of Innovative Research in Advanced Engineering (IJIRAE), vol. 04, issue 06, pp. 72-79, Jun. 2017.

[3] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "Performance Evaluation of the IEEE 802.15.4 MAC for Low-rate Low-power Wireless Networks," IEEE International Conference on Performance, Computing, and Communications, Phoenix, AZ, USA, 2004, pp. 701-706.

[4] L. Li, J. P. Seymour, L. J. Cimini and C. C. Shen, "Coexistence of Wi-Fi and LAA Networks with Adaptive Energy Detection," in IEEE Transactions on Vehicular Technology, vol. 66, no. 11, pp. 10384-10393, Nov. 2017.

[5] [Online] Available: https://internetofthingsagenda.techtarget.com/definition/LPWAN-low-power-wide-area-network, Accessed on 3 May 2018.

[6] [Online] Available: https://en.wikipedia.org/wiki/Star_network, Accessed on 5 May 2018.

[7] [Online] Available: https://blog.maidsafe.net/tag/bitcoin, Accessed on 6 May 2018.

[8] [Online] Available: http://www.sharetechnote.com/html/WLAN_FrameStructure.html Accessed on 12 May 2018.

[9] Y. Zhang and Q. Li, "Exploiting ZigBee in Reducing WiFi Power Consumption for Mobile Devices," in IEEE Transactions on Mobile Computing, vol. 13, no. 12, pp. 2806-2819, Dec. 1 2014.

[10] H. Qin and W. Zhang, "ZigBee-Assisted Power Saving for More Efficient and Sustainable Ad Hoc Networks," in IEEE Transactions on Wireless Communications, vol. 12, no. 12, pp. 6180-6193, December 2013.

[11] [Online] Available: http://kartji.com/product/xbee-or-zigbee-wireless-communication-module-wire-antenna, Accessed on 13 May 2018.

[12] [Online] Available: https://w3.siemens.com/mcms/sensor-systems/en/process-instrumentation/communication-and-software/wirelesshart/pages/wirelesshart.aspx, Accessed on 13 May 2018.

[13] V. C. Gungor and G. P. Hancke, "Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches," in IEEE Transactions on Industrial Electronics, vol. 56, no. 10, pp. 4258-4265, Oct. 2009.

[14] R. Zurawski, Industrial Communication Technology Handbook, Second Ed., CRC Press, 2015, ISBN: 978-1482207323.

[15] T. Gomes, F. Salgado, S. Pinto, J. Cabral and A. Tavares, "A 6LoWPAN Accelerator for Internet of Things Endpoint Devices," in IEEE Internet of Things Journal, vol. 5, no. 1, pp. 371-377, Feb. 2018.

[16] B. R. Al-Kaseem, H. S. Al-Raweshidy, Y. Al-Dunainawi and K. Banitsas, "A New Intelligent Approach for Optimizing 6LoWPAN MAC Layer Parameters," in IEEE Access, vol. 5, pp. 16229-16240, 2017.

[17] G. Mulligan, “The 6LoWPAN Architecture”, in Proc. 4th Workshop on Embedded Networked Sensors, Cork, Ireland, Jun. 2007, pp. 78-82.

[18] [Online] Available: http://processors.wiki.ti.com/index.php/Contiki-6LOWPAN, Accessed on 13 May 2018.

[19] [Online] Available: http://www.logistiikanmaailma.fi/en/logistics/digitalization/internet-of-things-iot, Accessed on 18 May 2018.

[20] [Online] Available: https://en.wikipedia.org/wiki/Internet_of_things, Accessed on 18 May 2018.

[21] [Online] Available: https://en.wikipedia.org/wiki/Internet_of_things, Accessed on 19 May 2018.

[22] [Online] Available: https://www.hindawi.com/journals/jece/2017/9324035, Accessed on 20 May 2018.

[23] [Online] Available: https://www.theguardian.com/politics/2015/mar/18/budget-2015-osborne-pins-digital-future-internet-of-things, Accessed on 20 May 2018.

[24] [Online] Available: https://iot-analytics.com/10-internet-of-things-applications, Accessed on 20 May 2018.

[25] IQRF Operating System, User's Guide, Version 4.02D for DC TR-7xD, [Online] Available: https://www.iqrf.org/support/download&kat=35&ids=155, Accessed on 23 May 2018.

[26] [Online] Available: https://www.iqrf.org/technology/iqrf-ide, Accessed on 24 May 2018.

[27] P. Seflova, V. Sulc, J. Pos, and R. Spinar, "IQRF Wireless Technology Utilizing IQMESH Protocol", In Proc. International Conference on Telecommunications and Signal Processing (TSP), Prague, 2012, pp. 101-104.

[28] V. Sulc, R. Kuchta, and R. Vrba, "IQRF Smart House - A Case Study", In Proc. International Conference on Advances in Mesh Networks, Venice, 2010, pp. 103-108.

[29] R. Kuchta, R. Vrba, and V. Sulc, "IQRF Smart Wireless Platform for Home Automation: A Case Study", In Proc. International Conference on Wireless and Mobile Communications, Cannes, La Bocca, 2009, pp. 168-173.

[30] V. Sulc, R. Kuchta, and R. Vrba, "IQMESH Implementation in IQRF Wireless Communication Platform", In Proc. International Conference on Advances in Mesh Networks, Athens, Glyfada, 2009, pp. 62-65.

[31] L. Krupka, L. Vojtech, and M. Neruda, "The Issue of LPWAN Technology Coexistence in IoT Environment", In Proc. International Conference on Mechatronics - Mechatronika (ME), Prague, 2016, pp. 1-8.

[32] M. Pies, R. Hajovsky and S. Ozana, "Wireless Measurement of Carbon Monoxide Concentration", In Proc. International Conference on Control, Automation and Systems (ICCAS 2014), Seoul, 2014, pp. 567-571.

[33] [Online] Available: https://www.iqrf.org/support/download [Data sheets], Accessed on 28 May 2018.

[34] [Online] Available: https://www.root.cz/clanky/bezdratovy-prenos-s-iqrf-cteni-z-externich-cidel/?ic=serial-box&icc=text-title, Accessed on 29 May 2018.

[35] [Online] Available: https://en.wikipedia.org/wiki/Lumen_(unit), Accessed on 30 May 2018.

[36] [Online] Available: https://www.pinterest.com/pin/259942209719687943, Accessed on 30 May 2018.

[37] [Online] Available: https://cloud.iqrf.org/prihlasit, Accessed on 1 June 2018.

[38] TR 45.820, "Cellular System Support for Ultra Low Complexity and Low Throughput Internet of Things," V2.1.0, August 2015.

[39] R. Ratasuk, B. Vejlgaard, N. Mangalvedhe, and A. Ghosh, "NB-IoT system for M2M communication," in Proc. IEEE Wireless Communications and Networking Conference, Doha, 2016, pp. 1-5.

Appendix A - IQRF Program Codes

A.1 Code for E01-TX:

The E01-TX is a simple transmitter code. This code makes the TR module work as a transmitter and upon pressing the application button in the DK-EVAL-04 module, a packet is transmitted. The transmission of a packet is confirmed by the TR module by blinking its red LED.

```
void APPLICATION()
{
appInfo();
   copyBufferINFO2RF();

   while (1)
   {
         sleepWOC();

      if (buttonPressed)
      {
         pulseLEDR();
         PIN = 0;
         DLEN = 10;
         RFTXpacket();
         waitDelay(25);
      }
   }
}
```

A.2 Code for E02-RX:

E02-RX is a simple receiver code. This makes the TR module to act as a receiver. It keeps searching for transmitted packets and upon capturing a packet from the transmitter, the TR module also blinks its red LED for confirming the delivery.

```
void APPLICATION()
{
        enableSPI();
   while (1)
   {
      if (RFRXpacket())
      {
         pulseLEDR();
         copyBufferRF2COM();
         startSPI(DLEN);
      }
   }
}
```

A.3 Code for CustomDpaHandler-Autobond:

CustomDpaHandler-Autobond is a code that make the nodes to bond with the coordinator automatically. CuatomDpaHandler is an optional service for the IQRF modules. If the custom dpa handler is enabled inside a node, it can do additional features. Initially bonding must be done manually but uploading this custom dpa handler makes the bonding automatic. It basically makes the node answer the ‘*answer me*' message sent by the coordinator while bonding.

```
bit CustomDpaHandler()
{
    clrwdt();
     switch ( GetDpaEvent() )
  {
DpaHandleReturnTRUE:
              return TRUE;
              case DpaEvent_Reset:
                     if ( !amIBonded() )
              {
                     pulsingLEDG();
                     waitDelay( 200 );
                     stopLEDG();
                     moduleInfo();
                     rand = bufferINFO[0];
                     if ( rand == 0 )
                       rand.0 = 1;
                     uns8 gapSize = 1;
                     for ( ;; )
                     {
                       waitMS( Rand() );
                       pulseLEDG();
                       uns8 loop = 200;
                       do
```

```
            {
                Rand();
                waitMS( 1 );
                if ( wasRFICrestarted() )
                  DpaApiSetRfDefaults();
            } while ( !checkRF( 20 ) && --loop != 0 );
            if ( loop == 0 )
            {
                pulseLEDR();
                bondRequestAdvanced();
            }
            if ( amIBonded() )
            {
                NodeWasBonded = TRUE;
                _LEDG = 1;
                waitDelay( 25 );
                stopLEDG();
                goto DpaHandleReturnTRUE;
            }
            loop = gapSize;
            uns8 rndTime = ( Rand() & 0b11111 ) + 10;
            do
            {
                waitMS( rndTime );
            } while ( --loop != 0 );
            Carry = 1;
            gapSize = rl( gapSize );
           }
       }
       break;
}
return FALSE;
```

```
}
uns8 Rand()
{
  rand = lsr( rand );
  W = 0b10111000;     // x^8 + x^6 + x^5 + x^4 + 1
  if ( Carry )
            rand ^= W;

  return rand;
}
```

A.4 Code for CustomDpaHandler-UserPheripheral-ADC:

CustomDpaHandler-UserPheripheral-ADC is the code for giving the TR modules with extra capability with analog to digital conversion. As seen from section 5.2, the predefined peripheral part contains extra peripherals for giving the IQRF modules more support. Peripheral number 20 and 21 needs to be enabled externally by uploading this code. Peripheral number 20 and 21 will assist the DK-EVAL-04 module with enabling C1 and C5 pin respectively. Any external sensor connected to these pins will be returned with 2 bytes of data. For this thesis, when the LDR sensor was merged with the IQRF module, the peripheral number 21 needed to be activated as the LDR was connected to the C1 pin of the DK-EVAL-04 module.

```
bit CustomDpaHandler()
{
 clrwdt();
 switch ( GetDpaEvent() )
 {
	case DpaEvent_Interrupt:

DpaHandleReturnTRUE:
	  return TRUE;
	case DpaEvent_Init:
	  TRISA.5 = 1;
	  TRISC.6 = 1;
	  TRISB.4 = 1;
	  break;

	case DpaEvent_DpaRequest:
	  if ( IsDpaEnumPeripheralsRequest() )
	  {
		_DpaMessage.EnumPeripheralsAnswer.UserPerNr = 2;
		FlagUserPer( _DpaMessage.EnumPeripheralsAnswer.UserPer, PNUM_USER
+ 0 );
```

```
                FlagUserPer( _DpaMessage.EnumPeripheralsAnswer.UserPer, PNUM_USER
+ 1 );
                _DpaMessage.EnumPeripheralsAnswer.HWPID = 0x000F;
                _DpaMessage.EnumPeripheralsAnswer.HWPIDver = 0xabcd;

                goto DpaHandleReturnTRUE;
        }
        else if ( IsDpaPeripheralInfoRequest() )
        {
                if ( _PNUM == PNUM_USER + 0 || _PNUM == PNUM_USER + 1 )
                {
                  _DpaMessage.PeripheralInfoAnswer.PerT = PERIPHERAL_TYPE_ADC;
                  _DpaMessage.PeripheralInfoAnswer.PerTE
= PERIPHERAL_TYPE_EXTENDED_READ;
                  goto DpaHandleReturnTRUE;
                }
                break;
        }
        else
        {
                if ( _PNUM == PNUM_USER + 0 || _PNUM == PNUM_USER + 1 )
                {
                  if ( _PCMD != 0 )
                        DpaApiReturnPeripheralError( ERROR_PCMD );
                  if ( _DpaDataLength != 0 )
                        DpaApiReturnPeripheralError( ERROR_DATA_LEN );

                  if ( _PNUM == PNUM_USER + 0 )
                  {
                        ANSELA.5 = 1;
                        ADCON0 = 0b0.00100.01;
                  }
```

```
            else
            {
                ANSELA.0 = 1;
                ADCON0 = 0b0.00000.01;
            }
            ADCON1 = 0b10010000;
            GO = 1;
            while ( GO );
            _DpaMessage.Response.PData[1] = ADRESH & 0x03;
            _DpaMessage.Response.PData[0] = ADRESL;
            _DpaDataLength = sizeof( uns16 );
            goto DpaHandleReturnTRUE;
           }
      }
 }
 return FALSE;
}
```

Appendix B – Data Sheets

Note that, all the data sheets attached in this thesis are available in www.iqrf.org/downloads.

B.1 TR-72D

TR-72D

Description

TR-72D is a family of IQRF transceiver modules operating in the 868 MHz and 916 MHz license free ISM (Industry, Scientific and Medical) frequency band. Its highly integrated ready-to-use design containing MCU, RF circuitry, integrated LDO regulator, serial EEPROM, optional temperature sensor and optional on-board antenna requires no external components. Extended RF power results in higher RF range. Ultra low power consumption fits for battery powered applications. MCU with built-in operating system significantly reduces application development time. Optional DPA framework supports applications even without programming.

There is no difference between TR and DCTR transceiver versions from IQRF OS v4.02D. All TRs support both OS as well as DPA approaches.

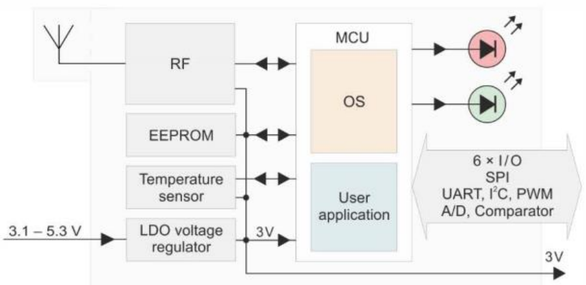
Key features

- Operating system (upgradeable at the user), easy to use
- DPA framework for Data controlled approach (formerly DCTR)
- GFSK modulation
- Selectable RF band 868 / 916 MHz, multiple channel
- RF output power 10 mW
- MCU with extended resources, user interrupt capability
- Extra low power consumption, power management modes
- SPI interface supported by OS in background
- Serial EEPROM 256 Kb
- PWM output
- Programmable HW timer
- +3 V LDO regulator output, battery monitoring
- 2 LEDs
- 8 pins, 6 I/Os
- A/D converter (2 channels), analog comparator
- Options: on-board antenna, U.FL connector, temperature sensor
- SIM card format fits KON-SIM-02 and KON-SIM-01 connectors
- Shielding can

Applications

- Bidirectional RF communication
- Point-to-point or network wireless connectivity
- Telemetry, AMR (automatic meter reading)
- WSN (wireless sensor network)
- Building automation
- Street lighting control
- Wireless monitoring, control and regulation
- Remote data acquisition
- RF connectivity in many other fields
- Also for municipal and indoor areas
- Internet of Things

Block diagram



© 2018 IQRF Tech s.r.o. www.iqrf.tech www.iqrf.org Datasheet_TR-72D_180130 Page 2

The information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets your specifications.

IQRF Tech MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND TO STATED CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE and disclaims all liability arising from this information and its use. Use of IQRF Tech devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless IQRF Tech from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any IQRF Tech intellectual property rights.

Electrical specifications

***Typical values** unless otherwise stated*

Parameters specified in this datasheet are typical values. They are at power supply V_{OUT} = 3 V only. V_{OUT} voltage different from 3 V can impact on RF range and other parameters.

Supply voltage (V_{CC})	3.1 V to 5.3 V
LDO output (V_{OUT})	+3 V ± 60 mV (V_{CC} > 3.1 V), 100 mA max.
Operating temperature [1]	-40 °C to +85 °C
Supply current	
Deep sleep mode (OS v4.00 or higher only)	1.7 µA (all peripherals disabled [3], RF IC in Standby mode)
Sleep mode	2.3 µA (all peripherals disabled [3], RF IC in Sleep mode)
Run mode	
RF sleep	1.4 mA
RF ready	2.8 mA
RX mode	
STD	12.1 mA
LP [4]	260 µA
XLP [4]	18.5 µA
TX mode	8.3 mA – 21.5 mA (according to RF output power)
Additional LED supply current	About 2 mA per LED. Rough value for brief guidance only.
RF band	868 MHz or 916 MHz (software configurable)
RF channels	See IQRF OS User's guide, Appendix *Channel maps*
RF data modulation	GFSK (Gaussian Frequency Shift Keying)
RF data transmission bit rate [5]	19.8 kb/s
RF receiver category	1.5 (according to ETSI EN 300 220-1 V3.1.1)
RF sensitivity [6]	-101 dBm, (STD RX mode, `checkRF(0)`). See *Diagram 4*.
RF output power [6, 7A]	Up to 10 dBm (for 50 Ω load), programmable in 8 levels (0 – 7).
Effective radiated power [7B]	Up to 6.5 dBm [2A], 11 dBm [2B] (868 MHz band), 2.0 to 6.5 dBm [2A] (916 MHz band). See *Diagrams 2A, 2B*.
RF interface [7A]	Single-ended, output impedance 50 Ω
Antenna [7B]	PCB meander line, linear polarization, omnidirectional. See *Diagram 1*.
RF range [2, 7B]	500 m [2A], 1100 m [2B]
Input voltage on C1, C2, C5 to C8 pins	0 V to V_{OUT}
A/D converter	10 bit, 2 inputs. Refer to MCU datasheet.
Temperature sensor	MCP9808E/MC (for TR types with 'T' postfix only, e.g. TR-72DT)
Size (L x W x H)	25.1 mm x 14.9 mm x 3.3 mm [7A] 31.8 mm x 14.9 mm x 3.3 mm [7B]

Note 1: RF range may change with lower temperature. Frost, condensation or humidity over 85% may disable module functionality. Transceiver suitability should be tested in the final application at real conditions before volume use.
Note 2: Arrangement: Two TR-72DA transceivers in DK-EVAL-04A kits, vertically, 1.6 m above the ground, in free space, bidirectional communication.
2A: TR-72DA transceivers plugged directly in DK-EVAL-04A kits.
2B: TR-72DA transceivers plugged in DK-EVAL-04A kits through the RNG-EXT-01 adapters.
Test software: E09-LINK example (STD mode, `setRFpower(7)`, `checkRF(0)`), bit rate 19.8 kb/s.
Note 3: Additional current is consumed when a peripheral (e.g. watchdog, Brown-out detection etc.) is enabled.
Note 4: Depends on interferences.
Note 5: Several RF bit rates different from 19.8 kb/s will be available in future IQRF OS versions.
Note 6: RF circuitry and RF balun included, built-in PCB antenna not included.
Note 7: 7A: For TR types without built-in antenna.
7B: For TR types with built-in antenna.

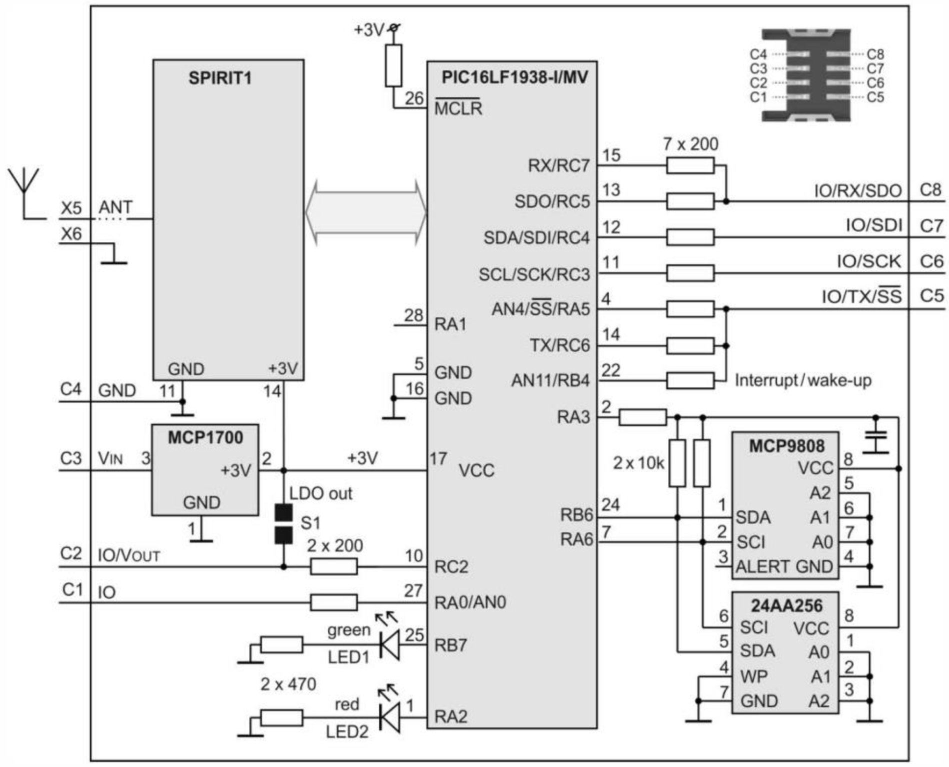
Absolute maximum ratings

Stresses above listed maximum values may cause permanent damage to the device and affect device reliability. Functional operation under these or any other conditions beyond those specified is not supported.

Supply voltage (V_{CC})	5.5 V
Voltage on C1, C2, C5 to C8 pins (configured as inputs) vs. GND	-0.3 V to (V_{OUT} + 0.3 V)
Storage temperature	-40 °C to +85 °C
Ambient temperature under bias	-40 °C to +85 °C

Caution: *Electrostatic sensitive device. Observe appropriate precautions for handling.*

Simplified schematic

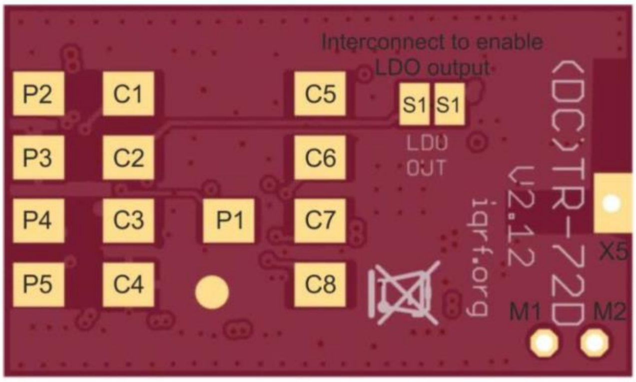


Basic components

IC	Type	Manufacturer	Note
MCU	PIC16LF1938–I/MV	Microchip	
RF IC	SPIRIT1	STMicroelectronics	
RF balun	BALF-SPI-01D3	STMicroelectronics	
LDO voltage regulator	MCP1700T-3002E/TT	Microchip	
Temperature sensor	MCP9808E/MC	Microchip	For types with 'T' postfix only, e.g. TR-72DT
EEPROM	24AA256-I/CS16K	Microchip	256 Kb

For more information refer to datasheets of ICs used.

Pin	Name		Description
C1	**IO / ADC / C-IN**		
		RA0	General I/O pin
		AN0	Analog A/D input
		C12IN0	Comparator –input
C2	**IO / VOUT**		
		RC2	General I/O pin (when S1 disconnected)
		VOUT	On-board +3 V LDO output (when S1 connected)
C3	**VIN**		Power supply voltage
C4	**GND**		Ground
C5	**IO / ADC / TX / -SS / PWM / COUT**		
		RA5	General I/O pin,
		-SS	SPI Slave select
		AN4	Analog A/D input
		C2OUT	Comparator output
		RC6	General I/O pin
		TX	UART TX
		CCP3	PWM output
		RB4	General I/O pin, with programmable pull-up and interrupt/wake-up on change (IOC), RFPGM termination
		AN11	Analog A/D input
C6	**IO / SCK / SCL**		
		RC3	General I/O pin
		SCK	SPI clock input
		SCL	I²C clock
C7	**IO / SDI / SDA**		
		RC4	General I/O pin. Used as input during initial about 200 ms boot-up (after power supply rising-up) to recognize programming mode.
		SDI	SPI data
		SDA	I²C data
C8	**IO / RX / SDO**		
		RC5	General I/O pin. Used as output during initial about 200 ms boot-up (after power supply rising-up) to recognize programming mode. That is why it should not be interconnected with the C7 pin.
		SDO	SPI data out
		RC7	General I/O pin
		RX	UART RX
X5	**ANT**		Antenna input
P1–P5			For manufacturer only
S1			LDO output enable. Interconnect both S1 pads to enable. Default (from the factory) disabled.
M1, M2			Holes for possible mechanical fixation



Bottom view

RF range

RF range strongly depends on the following design aspects:

- Hardware:
  - Construction of the devices (especially TR location within the device, PCB layout, ground planes, conductive areas and bulk objects such as metallic parts and batteries in the nearest surroundings, with respect to possible reflections and counterpoise effect). To achieve an efficient range and reliable connectivity, no parts impacting the range must be placed close to the built-in meander antenna. Even non-conductive parts including a mainboard PCB under the antenna can significantly impact the range.
  - Physical arrangement of devices (especially mutual orientations of antennas with respect to polarizations and radiation patterns)
- Application software:
  - RF output power is selectable from 8 levels
  - To increase immunity to RF noise, incoming RF signal can be filtered according to signal strength.

Refer to IQRF OS Reference guide, function `checkRF` and Application note AN014 *RF range optimizing at TR-7xDx transceivers.*

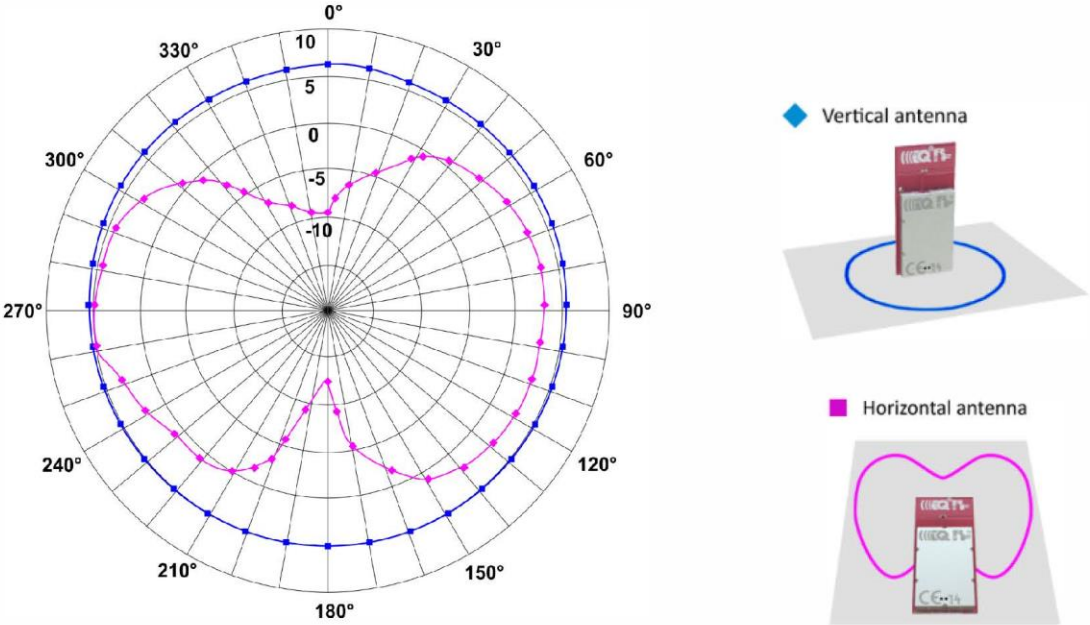
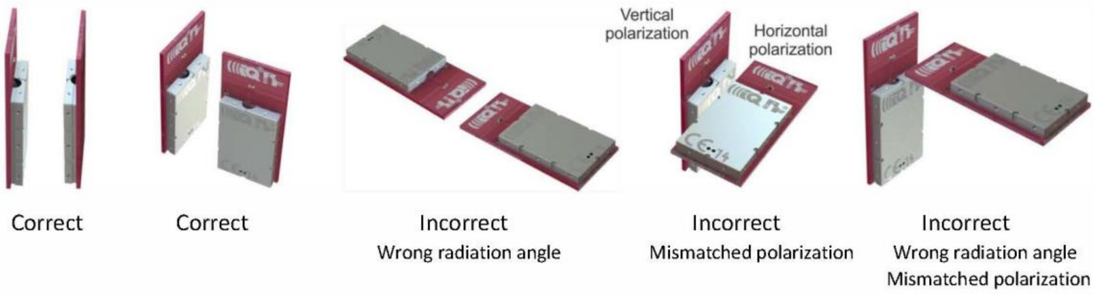


Diagram 1: *TR-7xDA RF output power [in dBm] vs. antenna orientation (radiation patterns).*

Examples of the correct and incorrect arrangement of TR-72DA pairs:



B.2 CK-USB-04A

CK-USB-04A

Description

CK-USB-04A is a development kit intended for programming and debugging of user applications with IQRF transceivers.

It can serve also as an IQRF USB gateway (with USB ↔ SPI and USB ↔ UART converter) or just a simple host for TR transceiver application.

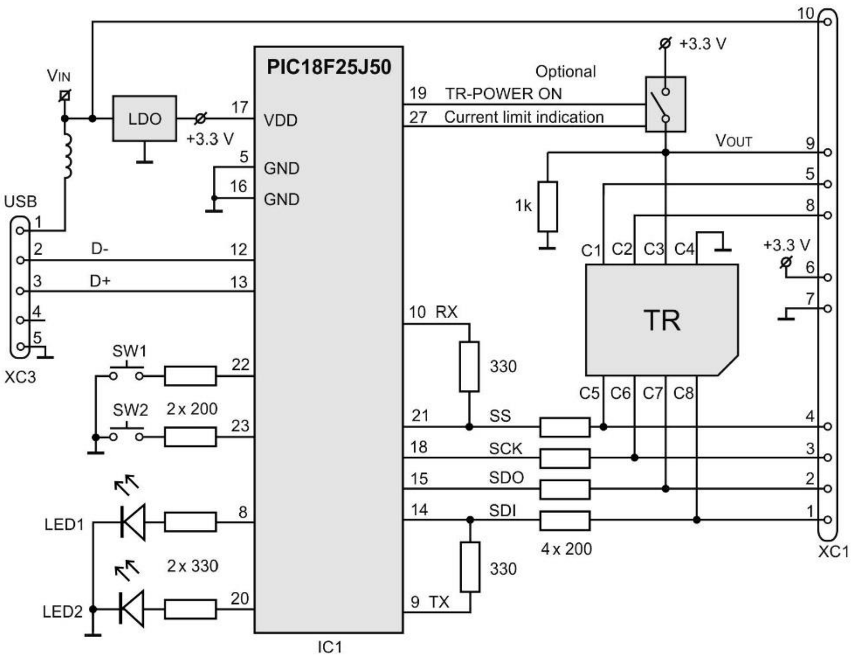
Applications

- Programmer for IQRF transceiver modules
- In-circuit programming supported
- IQRF debugging kit
- End IQRF application host
- USB gateway
- USB to SPI and USB to UART converter
- PC connectivity

Key features

- Supported by the IQRF IDE development environment
- Wired or wireless (RFPGM) upload possible
- SIM connector with plastic holder for TR module
- 2 pushbuttons, 2 LEDs
- USB interface (CDC and Custom device, MICRORISC VID & PID)
- Custom Device, CDC IQRF, CDC SPI and CDC UART modes
- Faster SPI for TR-7xD supported
- I/O, power supply and upload interface via single row connector
- Powered via USB connector
- V_{OUT} controlled by IQRF IDE, optionally current-limited
- Multiple devices controlled by IQRF IDE on single PC
- Firmware upgradeable from IQRF IDE

Simplified circuit diagram



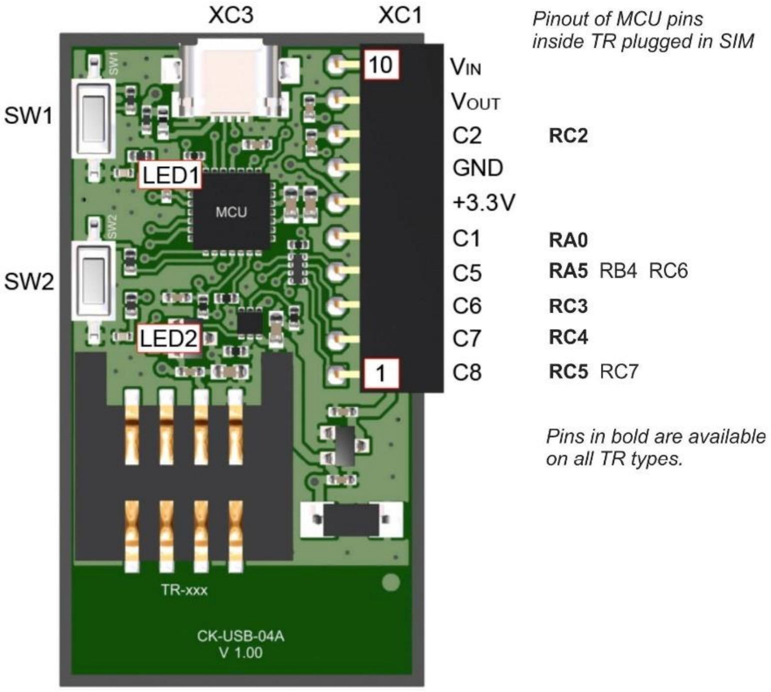
Electrical specifications	***Typical** values (until otherwise specified)*
Power supply input (V_{IN})	
Supplied with USB	5.0 V
Supplied via external power connector XC1	3.6 V – 5 V USB must be disconnected in this case
Power supply output (V_{OUT})	3.3 V, with optional current limitation
I/O and SPI voltage levels	3.3 V
Operating temperature	0 °C to +70 °C -40 °C to +85 °C (Industrial) available on request
Size	48 mm x 27 mm x 11 mm
Weight	10 g

Absolute maximum ratings

Stresses above those values may cause permanent damage to the device. Exposure to maximum rating conditions for extended periods may affect device reliability.

Supply voltage:	5.5 V
Storage temperature:	-40 °C to +85 °C

Control and indication



Pushbuttons

USB mode selection

- Selection of USB modes can be enabled by pressing the SW1 pushbutton for 2 s. Then LEDs indicate the current mode by a series of flashing (see chapters *USB* and *LEDs* below).
- Every short press of SW1 switches the mode in cyclic order. LEDs always indicate the current mode by a series of flashing.
- SW1 pressing for 2 s stops the switching, stores the current mode and LEDs indicate this newly selected USB mode by a series of flashing.

USB mode selection is canceled (without storing the changes):
- Anytime by short pressing of SW2.
- Automatically, if SW1 is not pressed for 5 s.

SPI speed selection

Switching between *TR5x legacy* and *TR7x faster* modes (see chapter *SPI speed* below) is invoked by pressing both pushbuttons when power is switched on (while USB cable is connecting). The pushbuttons can then be released just after initial LED indication starts (see chapter *LEDs* below).

C5 pin control

- A short press of SW1 sets the C5 pin to log. 0 for 800 ms. During SPI communication this pushbutton is ignored. C5 pin control is applicable in Custom Device and IQRF CDC modes only.

TR module power off

- During SW2 pressed, V_{OUT} (TR module power) is off. It is useful for TR reset and restart the application program. This is applicable in all USB modes.

LEDs

After CK-USB-04A reset (by reconnecting the USB cable), current USB mode is indicated by a series of flashing [1] of both LEDs as follows:

- Custom Device 1x
- CDC IQRF 2x
- CDC SPI 3x
- CDC UART 4x

In the idle state, both LEDs are off. If there is a need to indicate the current USB mode, it is possible either to perform the device reset or initiate the USB mode selection and abandon it then without storing the changes. See chapter *Pushbuttons*.

LED1 flashes 3x after the *Indicate USB Device* command (key *F11*) in respective IQRF IDE (to identify the kit in case of multiple IQRF USB devices or IDE instances).

LED1 flashes [1] once if SW1 is pressed (in Custom Device and CDC IQRF modes only).

LED2 flashes [1] once after TR reset induced by IQRF IDE and shines permanently when SW2 is pressed.

Complementary LED1 and LED2 flashing in ~1 s period means missing firmware or firmware upgrade in progress (see *Firmware upgrade* below).

Note 1: There are two types of flashing possible in these conditions to indicate current SPI speed (see below):

- Continual LED shining during given indication: TR5x legacy mode
- Fast LED blinking during given indication: TR7x faster mode

Connectors

- USB: Micro USB
- SIM: Supported TR types see table *Compatibility* below.
- XC1: I/O/SPI/programming/power interface. Female connector for square 0.635 mm, 2.54 mm pitch pins.

***Caution:** The TR module can be plugged / unplugged into / from the SIM connector while powered off only.*

***Tip:** Use the SW2 pushbutton for this. The TR module is not powered while the SW2 pushbutton is held.*

SPI speed

Depending on TR type, two SPI communication speeds are available.

- TR5x legacy mode – necessary for TR-5xD and lower transceiver types, possible also for all other TR types.
- TR7x faster mode – allowed (and recommended) for TR-7xD and higher transceiver types.

As a matter of interest, details regarding SPI timing are described in documents *SPI Implementation in IQRF for (DC)TR-7xD* and *SPI Implementation in IQRF for TR-5xB and (DC)TR-5xD*, chapter *Timing*.

SPI speed selection:

- Using IQRF IDE	Menu *Tools – IQRF Device Manager*.
- Using hardware pushbuttons	See chapter *Pushbuttons* above.

Refer to chapter *LEDs* for indication of the current mode. The last selected mode is restored after start-up or reset.

B.3 DK-EVAL-04

DK-EVAL-04

Description

DK-EVAL-04 is a universal development kit for wireless applications with IQRF transceiver modules. Very small size, LiPol accumulator and low cost make this kit ideal for use in networks.

A user-specific functionality can be implemented by software in internal TR module.

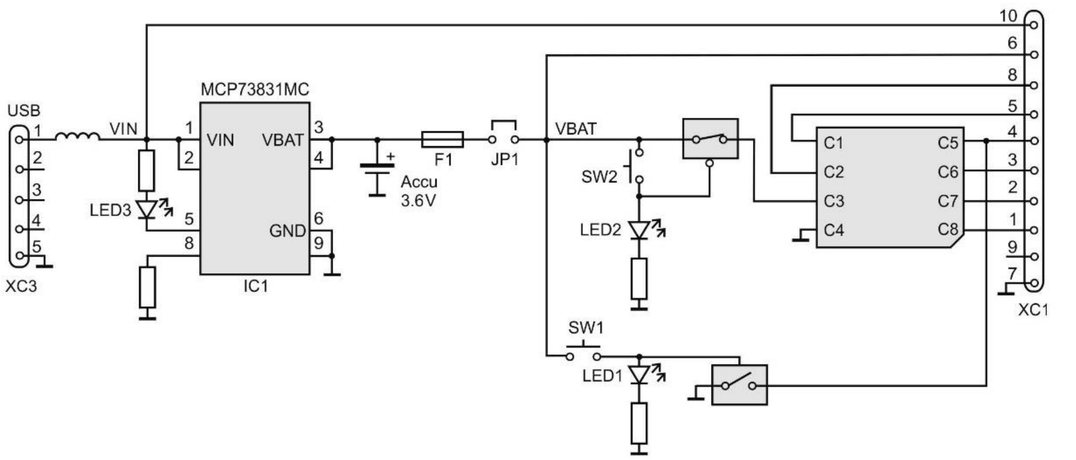
Applications

- Wireless applications development
- Host for IQRF TR modules
- Huge IQMESH networks debug and testing
- Battery powered and portable wireless systems

Key features

- SIM connector with metallic holder for TR module
- 2 pushbuttons, wake-up on button press capability
- 3 indication LEDs
- 6 I/Os
- LiPol accumulator and internal charger. Charged via microUSB connector
- Accumulator over-current protection
- Voltage output to supply low power peripherals, sensors etc.
- Optional DK-PWR-01 power supply board to support operation and charging up to 5 kits available
- Compatible with IQRF DDC kits (Development Daisy Chain)
- Space saving

Simplified circuit diagram



 www.iqrf.org User_Guide_DK-EVAL-04_140801 Page 2

Electrical specifications	*(**Typical values** unless otherwise stated)*
Power supply	
Accumulator	LIP552240, 400 mAh, nominal voltage 3.7 V
External source	
via micro USB connector (with charging)	4.4 V to 6.0 V DC
via XC1 connector, pin 6	According to power supply range of TR module used
Supply current	
sleep	250 nA (powered from accumulator, jumper JP1 disconnected)
charging	50 mA max.
Temperature	0 °C to +45 °C (operating), +10 °C to +25 °C (storage)
Supported TR modules	TR-52B, TR-53B, TR-52D, TR-55D and similar types, in SIM card format
Dimensions	48 mm x 27 mm x 11 mm
Weight	17 g

Absolute maximum ratings

Stresses above those values may cause permanent damage to the device. Exposure to maximum rating conditions for extended periods may affect device reliability.

Power supply (from external source)	6.0 V
Storage temperature	-20 °C to +60 °C

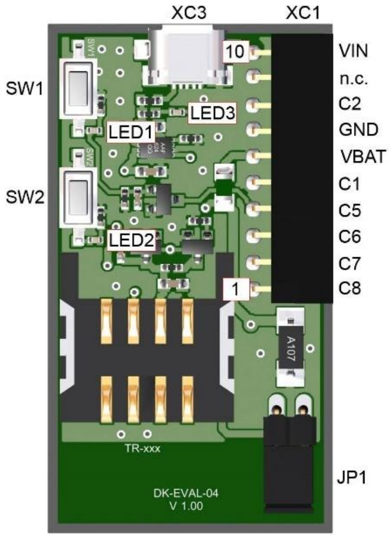
Hardware

Power supply

DK-EVAL-04 is supplied from internal accumulator or from external power source via micro USB connector XC3 which also serves as a charger. The TR module is supplied when jumper JP1 is set. Charging is indicated by LED3. The accumulator is protected against over-current by resettable fuse SN035-16.

The accumulator should be kept charged.

For external power source also connector XC1 (pins 6 and 7) can be used. Jumper JP1 must be disconnected in this case.



Pushbuttons

- User pushbutton SW1
  is connected to pin C5 of the TR module, active low. Therefore, the TR module should have the C5 pin configured as input with internal pull-up. It is arranged by OS by default. Wake-up on change or interrupt on change on this pin can also be configured by user software.
- Reset pushbutton SW2
  TR module is disconnected from power supply when the SW2 pushbutton is pressed.

LEDs

- LED1 and LED2 are on when the appropriate pushbutton is pressed.
- LED3: charging indication. LED3 is on during charging and switched off when fully charged.

SIM connector

Caution: *The TR module can be plugged / unplugged into / from the SIM connector while powered off only.*

Tip: *Use the SW2 pushbutton for this. The TR module is disconnected from power while the SW2 pushbutton is held.*

Supported TR module types

TR	Supported
TR-52D	Yes
TR-53D	Yes
TR-54D v1.01	–
TR-54D v1.02	–
TR-55D	–

TR	Supported
TR-56D	–
TR-58DA	–
TR-52B	Yes
TR-53B	Yes
TR-72D	Yes

Caution: *The unsupported TR modules must not be plugged into DK-EVAL-04 SIM connector.*

Interface connector

Interface connector XC1 makes SIM I/O pins, power supply and ground accessible externally. It is compatible with IQRF DDC development kits.

Caution: *DK-EVAL-04 PCB layout is optimized for space savings during development of network applications with a lot of nodes on a table. Thus, this kit is not intended for range tests using TR modules with internal PCB antenna (e.g. TR-52D**A**, the range would be shortened in this case). For range test use e.g. TR-52D and an external antenna.*

Product information

Pack list

- DK-EVAL-04 kit (without a TR-module)
- Accumulator (soldered) inside
- 1 jumper (power on switch)

Recommended options

• TY-A6A	Power supply with USB A connector
• CAB-USBABMICRO-100	Cable for TY-A6A power supply
• DK-PWR-01	Power supply expansion board (to supply and charge up to 5 DK-EVAL-04 kits)

Ordering code

• DK-EVAL-04	IQRF universal development kit

Document history

• 140801	Chapter *Supported TR module types* added.
• 140124	Revised. Document file renamed from MNDKEVAL04 to User_Guide_DK-EVAL-04.
• 120926	Temperature range corrected.
• 120921	Accumulator voltage range added. Bug in pin 6 description fixed.
• 120208	Bug in description of XC1 and XC3 connectors fixed. New power adapter.
• 110419	Caution concerning range test added
• 110210	First release

B.4 GW-GSM-02

GW-GSM-02

Description

GW-GSM-02 is an IQRF gateway with GSM connectivity intended as an interface between IQRF and GSM/GPRS networks. The user can implement specific functionality by software for internal IQRF TR module.

Main components are 8 b microcontroller, GSM/GPRS transceiver module and IQRF TR module with antennas, FLASH memory and backup accumulator.

GW-GSM-02 can be configured and firmware can be upgraded from PC. Application software for TR module can be uploaded via IQRF IDE4 using the gateway as a programmer.

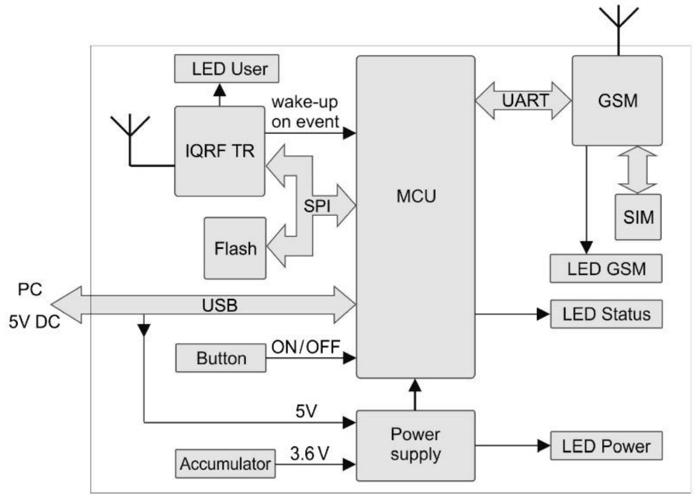
Key features

- TCP Server mode with public IP address
- SMS alarm feature
- Data logging
- IP filtering
- RTCC (real time clock/calendar)
- Backup accumulator
- Very low power consumption in Sleep mode
- Documented application protocol
- Firmware upgrade via PC software
- Programmable application in internal TR module

Applications

- Remote monitoring and control
- Data acquisition / collection
- Datalogger
- Interface to building / home automation
- Remote control of IQRF network
- Connection of more IQRF networks to one PC
- Access to IQRF network from more locations

Block diagram



Electrical specifications

(**typical values** unless otherwise stated)

Power supply	4.8 V – 5.5 V DC
Accumulator	LI14500-1L, 3.7V, 700 mAh
Supply current	
Standby (all peripherals disabled)	12 µA
Operational	
TR and GSM inactive	75 mA
Additional current	
TR active	See datasheet of TR module
GSM transmitting	175 mA
Accumulator charging	450 mA max.
Temperature range	
Operational	0 °C to +60 °C
Accumulator charging	0 °C to +45 °C
Storage	10 °C to +20 °C (recommended)
IQRF	
Frequency bands	868 MHz or 916 MHz (SW selectable)
RF output power	According to TR module, programmable
TR module	TR-54DA
Antenna	PCB antenna built-in TR module
GSM	
frequency bands	900 MHz or 1800 MHz
Antenna	External, SMA, gain 2.15 dBi
Flash memory	1 MB, serial interface SPI, 100 000 erase / write cycles typ.
Data logging	Up to 3200 events
Temperature sensor	Optional, built-in TR module
Dimensions	
without antenna	86 mm x 60 mm x 24 mm
with antenna	104 mm x 60 mm x 60 mm
Weight	90 g

Absolute maximum ratings

Stresses above those values may cause permanent damage to the device. Exposure to maximum rating conditions for extended periods may affect device reliability.

Supply voltage (VCC)	5.5 V
Storage temperature	-20 °C to +60 °C