PERFORMANCE ANALYSIS OF SMART AUTOMATION SYSTEMS

Progress Report
In fulfillment of the requirements for the
NU 302 R&D Project
At NIIT University

Submitted by:
Abhinav Parihar, Hitika Handa and S R Rahul
Electronics and Communications

Karanam Rakesh, Kandula Mohan Sai and Dandu Geet Kamal Tej
Computer Science

NIIT University
Neemrana
Rajasthan

CERTIFICATE

This is to certify that the present research work entitled " Performance Analysis of Smart Automation Systems” being submitted to NIIT University, Neemrana, Rajasthan, in the fulfillment of the requirements for the course at NIIT University, Neemrana, embodies authentic and faithful record of original research carried out by Abhinav Parihar, Hitika Handa and S R Rahul of B Tech (Electronics and Communications) and Karanam Rakesh, Kandula Mohan Sai and Dandu Geet Kamal Tej of B Tech (Computer Science) at NIIT University, Neemrana,. She /He has worked under our supervision and that the matter embodied in this project work has not been submitted, in part or full, as a project report for any course of NIIT University, Neemrana or any other university.

(Jetendra Joshi)

LIST OF FIGURES

LIST OF TABLES

CONTENTS

Rationale of work

The build out of the Internet of Things is outpacing desktop and mobile computing. By 2020, over 50 billion intelligent devices (Cisco) will connect to and exchange information over the Internet with an economic impact of nearly US$2 trillion. This huge cohort of “things” comprises staggering diversity, from recognizable computers to infrastructure devices to sensors, light switches, and thermostats.

The impact will span the gamut of industries and applications – medical, agriculture, manufacturing, consumer electronics, transportation, and energy. Like the existing Internet, the emerging IoT will rely upon and instigate adoption of ofpe source software (OSS) technologies and open standards.

With the range of applications and constituents, divergent visions exist for building out and benefiting from the IoT. Some see the IoT as an incremental extension of existing computing technologies and methods (including open source); others herald the IoT as a revolution that will reinvent the IT industry and spur a major paradigm shift (Figure 1).

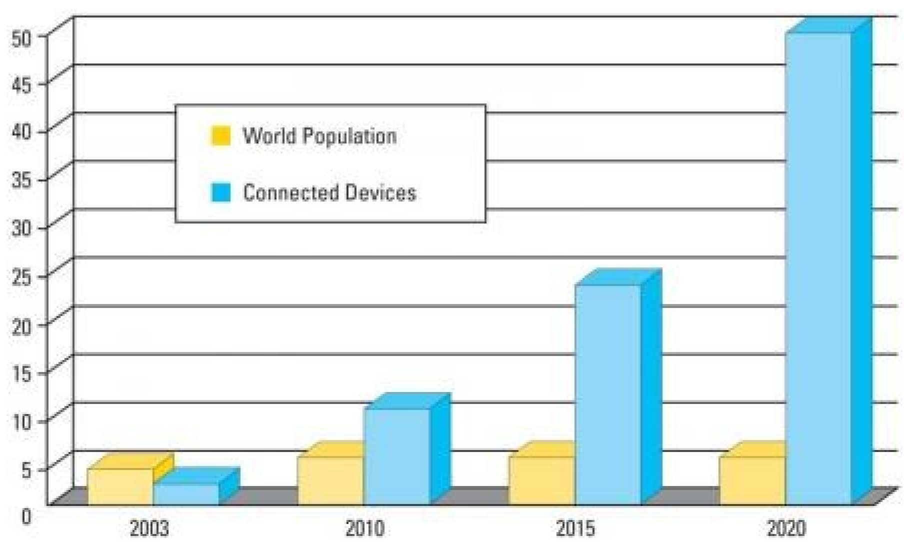


Fig 1. Growth in Human Populations and Connected Devices through 2020 (Cisco)

The buzz about smart cities are catching up. Where 8 years back the people were talking only about smart phones, through a series of parallel technological developments globally the concept of smart cities is rising.

The primary drive for automation IoT is to significantly reduce operating expenditures when automation devices, sensors and actuators become Internet-enabled devices. It's the next huge leap in productivity because there are major advantages to be derived from the acquisition and organization of previously unthinkable amounts of data. New Enterprise Manufacturing Intelligence software (EMI) brings manufacturing-related data together from many sources for reporting, analysis, visual summaries and passing data between enterprise-level and plant-floor systems.

With the increasing use of fieldbus within control networks and the spread of software intelligence, some think that IoT is already emerging in industry. But, the progress that's occurred is just a small fraction of what can and will happen over the next decade. The inflection point will occur when literally everything is connected with inexpensive and easy-to-install wireless networks.

Industrial IoT must be self-organizing, self-configuring, self-healing, scalable to large sizes, with very low energy consumption, low cost, simple to install and based on global standards. That's a tall order, which current automation network standards simply cannot meet. In my opinion, with the spread of IoT, the ZigBee over IEEE 802.15.4 standard, currently languishing with minimal market share, will emerge to mainstream prominence.

What is a Smart City?

It's a city outfitted with high-tech communication capabilities. It uses digital technology to enhance performance and well-being, to reduce costs and resource consumption, and to engage more effectively and actively with its citizens.

The idea of smart city came into formulation owing to the need to accommodate rapid urbanization of the age. Interest in smart cities continues to grow, driven by a range of socioeconomic and technological developments across the globe. It is due to the increasing number of smart cities that established suppliers from energy, transport, buildings, and government sectors are moving into the smart city market, while startups are addressing a range of emerging opportunities in the same field.

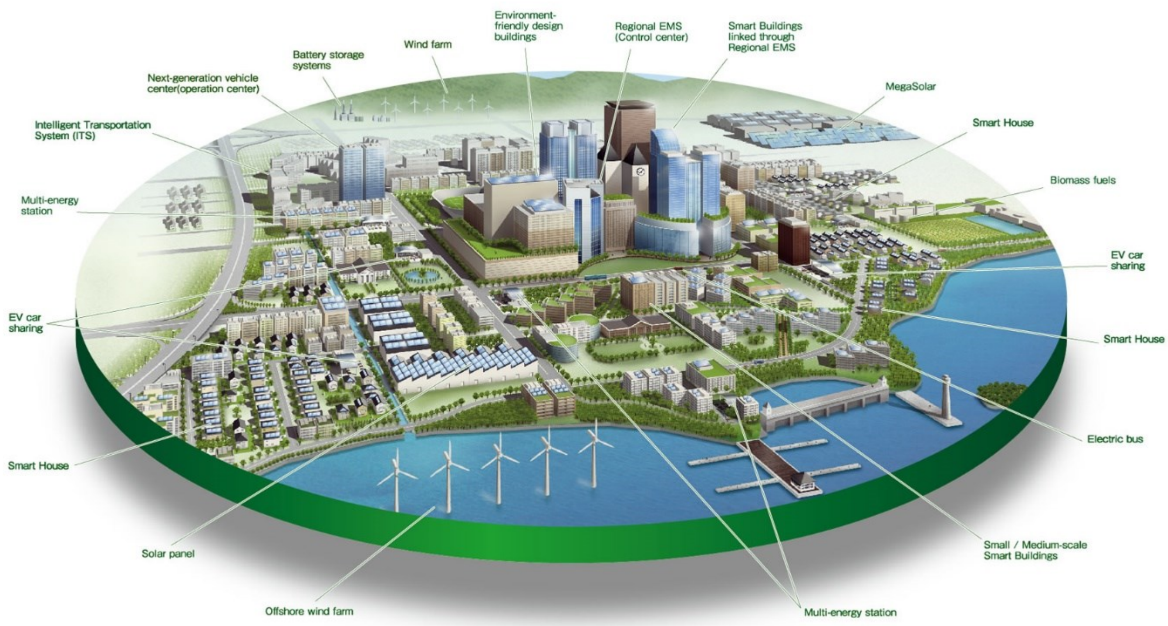


Fig 2. Smart Environment / Smart City

What is the Scope of Smart Cities in India?

India is drawing on the development of smart cities at the global level. Prime Minister Narendra Modi's vision 'Digital India', has a plan to build 100 smart cities across the country. Modi in his speech said, "Cities in the past were built on riverbanks. They are now built along highways. But in the future, they will be built based on availability of optical fiber networks and next-generation infrastructure."

Digital India envisages making India a leader in digitally delivering services in the health, education, banking sectors. Modi announced an investment of $1.2 billion in smart cities with more funding coming from private sectors and abroad.

Prime Minister Narendra Modi's vision "Digital India," has set an ambitious plan to build 100 smart cities across the country.

Modi in his speech quoted, “Cities in the past were built on riverbanks. They are now built along highways. But in the future, they will be built based on availability of optical fiber networks and next-generation infrastructure.”

As the Internet population continues to grow from 60 million in 2009 to 190 million in 2014, it is estimated that the potential growth will be over 550 million users by 2018. The number of Internet users in rural areas will touch 210 million by 2018, aiding India’s internet user base to cross 500 million by 2018.

The Internet class of 2018 will be more rural, older, more gender-equal, more mobile, and more vernacular than their counterparts of today. Rural users which will be the Internet population of the future, is expected to rise from 29 percent in 2013 to between 40 and 50 percent in 2018. Thus, the rural area seems to be catching up with digital quiet swiftly. By 2018, the rural population will stand on an equal footing with the urban population in terms of internet usage and accessibility.

Therefore, the smart city project needs to be designed wisely, considering the local population as the key point. Also, it has to be careful as to not widen the already present gulf between the rural and the urban class. A nation should progress taking all its citizens along, irrespective of class differences, and catering to everyone’s need equally.

As much as the vision of smart cities is incredible and rational, its implementation on India at the given socio-economic condition, might be a tad bit difficult. However, as a fast developing economy, the country needs to keep up with the global standards. Hence, the execution of this plan could make India take a major leap in the race of development.

Home automation gives you access to control devices in your home from a mobile device anywhere in the world. The term may be used for isolated programmable devices, like thermostats and sprinkler systems, but home automation more accurately describes homes in which nearly everything -- lights, appliances, electrical outlets, heating and cooling systems -- are hooked up to a remotely controllable network. From a home security perspective, this also includes your alarm system, and all of the doors, windows, locks, smoke detectors, surveillance cameras and any other sensors that are linked to it.

Until fairly recently, automated central control of building-wide systems was found only in larger commercial buildings and expensive homes. Typically involving only lighting, heating and cooling systems, building automation rarely provided more than basic control, monitoring and scheduling functions and was accessible only from specific control points within the building itself.

Home automation is a step toward what is referred to as the "Internet of Things," in which everything has an assigned IP address, and can be monitored and accessed remotely.

The first and most obvious beneficiaries of this approach are "smart" devices and appliances that can be connected to a local area network, via Ethernet or Wi-Fi. However, electrical systems and even individual points, like light switches and electrical outlets, were also integrated into home automation networks, and businesses have even explored the potential of IP-based inventory tracking. Although the day is still far off when you'll be able to use your mobile browser to track down a lost sock, home networks are capable of including an increasing number of devices and systems.

BI Intelligence, Business Insider's premium research service, expects the number of smart home devices shipped will grow from 83 million in 2015 to 193 million in 2020. This includes all smart appliances (washers, dryers, refrigerators, etc.), smart home safety and security systems (sensors, monitors, cameras, and alarm systems), and smart home energy equipment, like smart thermostats and smart lighting.

The first and most obvious benefit to smart homes is convenience, as more connected devices can handle more operations (lighting, temperature, etc.) and frees up the resident to perform other tasks.

But beyond this, smart home IoT devices can help reduce costs and conserve energy. In our example above, you'd have a comfortable and cool apartment when you get home, but you could also leave your air conditioner off when you're not home, which would lower your electric bill and reduce energy consumption. Smart lights would function in a similar way.

Of course, there are disadvantages, as well. Smart home devices are typically more expensive than their non-connected counterparts, so consumers would certainly feel the hit in their wallets at first.

Consider than connected LED bulbs cost $15 on average, compared to $8 for no connected LED bulbs. However, the cost of these connected bulbs has dropped in the last two years, so the prices of smart home IoT devices could decline even further and make them more affordable to the average consumer.

Today's automation systems applications are cumbersome and complex to develop. Each new device requires too much customization and maintenance just to perform the basic tasks. Connecting disparate devices and systems entails a high level of engineering complexity, because of different data formats, diverse networks, incompatible IP addressing schemes, many operating systems, and so on.

In order for the market to really take off several key technology hurdles will need to be addressed: truly ubiquitous device communications standards and new software architecture to support massively peer-to-peer, complex event-driven data management. The present-day lack of interoperability with distributed, real-time device data remains a huge hurdle.

Water resource management is a very important issue from several angles such as development of water bodies for future, protection of available water bodies from pollution and over exploitation and to prevent disputes. A paramount issue is water-its availability, quality and management. Extensive hydrological information is necessary to develop water resources and protect them.

Literature Review

An increasing number of home automation systems using wireless devices compete for the radio access in the same space and time. Lately, a stressing trend consists of aggregating home automation systems to save energy consumption, while at the same time avoiding wireless interference. This article [1] proposes virtualization, open software deployment, and separation of radio and higher layers as the response to the increasing expandability of home automation systems combined with the increasing number of technologies for connecting wireless devices. A system has been developed, containing three different technologies: ZigBee, Idsecom, and 6LoWPAN simultaneously working over a virtualization platform with access to a common antenna. The evaluation tests performed on the system validate the solution and separately show the performance capacity of virtualization platform, software (ZigBee, Idsecom and 6LoWPAN) nodes, and 802.15.4 wireless antennas.

In this research paper [2] the working experiences which may support other researchers to achieve success without wasting resources are reported. Moreover, presented research analyses of some unusual observations made in a proto-typed smart home monitoring system have been discussed. Packet delivery ratio (PDR) and latency are among a few reliability issues that have been analyzed and addressed in this article to make the system more robust. With their experimental research and analysis the PDR of a node which is at far distance from coordinator degraded more as compare to the nodes near to coordinator in a multi-hop network. The optimum spacing varies from one to another environment, so the spacing between nodes and deployment environment should be properly analyzed by repeating sufficient number of experiment. Additionally, this paper has reported some of our working experiences with using ZigBee protocol (XBee RF module) as communication devices. According to classical theory receive delay is a function of sampling rate and packet length; moreover it is based on spacing between sensor nodes. For developing smart home though significant successes have been achieved using

ZigBee, it is expected more successes if the issues are taken care of during the design stages.

Connectivity is probably the most basic building block of the IoT paradigm. Up to now, the two main approaches to provide data access to things have been based on either multi-hop mesh networks using short-range communication technologies in the unlicensed spectrum, or long-range legacy cellular technologies, mainly 2G/GSM/GPRS, operating in the corresponding licensed frequency bands. Recently, these reference models have been challenged by a new type of wireless connectivity, characterized by low-rate, long-range transmission technologies in the unlicensed sub-gigahertz frequency bands, used to realize access networks with star topology referred to as low-power WANs (LPWANs). In this article [3], they introduced this new approach to provide connectivity in the IoT scenario, discussing its advantages over the established paradigms in terms of efficiency, effectiveness, and architectural design, particularly for typical smart city applications.

The last decade saw the emergence of the Internet of Things (IoT) paradigm, which aims to connect any object to the Internet. In this context, a new type of wireless communication network emerged known as Low-Power Wire-Area Network (LPWAN). By contrast to well-known short range and multi-hop wireless networks, LPWAN networks allow long range communications at a low bit rate. Furthermore, LPWAN networks are considered to be integrated into 5G. Among LPWAN networks, the LoRaWAN technology gains more and more interest from the research and industrial communities. In this article [4], they have led a thorough experimental performance evaluation of LoRaWAN in an indoor environment. From this study, we quantify the limits of this technology and expose the merits of using LoRaWAN for IoT communications in the context of 5G.

Wireless sensor networks (WSNs) are largely used to monitor rural environments and urban environments. Indeed, WSNs have a strong potential thanks to their low cost, their autonomy in energy, and the disposability of the individual sensors. Wireless sensor networks have also found many applications with the Internet of Things paradigm, where the Internet is coupled with the physical world through sensors and actuators. The IEEE 802.15.4 standard, initially designed in 2003, is by far the main standard for wireless communications in WSNs. Several researchers have analyzed the performance of the MAC layer of IEEE 802.15.4, as it has a direct impact on the applications using this standard. Most of the analytical analyses propose a Markov chain model, adapted to different cases. IEEE 802.15.4 has only short-range capabilities: the communication range traditionally varies between tens of meters (in indoor environment) to up to one hundred meters (in outdoor environment and favorable conditions, such as line of sight). To compensate for this limitation, it is possible to deploy more nodes in order to achieve network connectivity. This comes at an extra cost, which might be prohibitive in large-scale applications such as volcano monitoring, forest monitoring, or large urban deployments. Recently, long-range low-power wireless standards have emerged, such as Sigfox, Ingenu and LoRaWAN. These standards enable one-hop communications ranging from 2 km in dense urban environments to 30 km in rural environments. These standards bring high expectations, as they create new possibilities for low-power, largescale monitoring applications. Since most of these standards are proprietary, few research works have focused on them. In this paper [5], they studied the MAC layer of the LoRaWAN standard (which is open and freely available), and we focus on the on-the-air activation procedure, which allows nodes to join an existing network. To do this, they proposed a Markov chain model of the activation procedure, and we derive two important performance metrics: the expected delay to complete the activation, and the required energy. The study is based on the EU863-870 MHz regional setting (also other regional settings are discussed in the paper) for a single gateway. To the best of our knowledge, this is the first analytical study of the MAC layer of LoRaWAN.

Objectives

- Performance Analysis of various devices in physical layer by building a testbed of Home Automation system.
- Study of different performance parameters of an automation system.
- Analysis of different performance parameters of an automation system using various communication protocols.
    - Nrf
    - Bluetooth
    - BLE
    - LoRa
- To design Smart Water Monitoring Solution and Smart Garbage Management using LoRa.

Methodology

Arduino

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs like light on a sensor, turning on an LED. You cantell your board what to do by sending a set of instructions to the microcontroller on the board.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for iot applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs.

Fig 3. Arduino

Arduino features:

1. MEMORY:

The Arduino Uno has 32 KB memory. It comes with 2 KB of SRAM and also 1 KB of EEPROM.

2. CLOCK SPEED:

The performance of this controller is based on its clock speed. The Clock speed of the Arduino is 16 Mhz so it can perform a particular task faster than the other processor or controller.

3. USB INTERFACE:

Most important feature of Arduino Uno is USB connectivity. It means if we want to operate Arduino with PC, then we can do that and data communication between PC and Arduino become easy.

4. INPUT OUTPUT VOLTAGE:

The Arduino Uno can be powered via the USB connection or with an external power supply. If we are using external power then we can supply 6 to 20 volts. Arduino works on 5 volts.

5. INPUT OUTPUT PINS:

Each of the 14 digital pins on the Uno can be used as an input or output. 6 pins out of 14 can be used as PWM output. 6 pins can be used as analog pins.

6. COMMUNICATION:

Arduino board supports I2C and SPI communication. The Arduino software includes wire library for I2C and SPI library for the SPI communication.

Advantages of Arduino:

1. INEXPENSIVE:

Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than Rs.3000.

2. CROSS PLATFORM:

The Arduino Software (IDE) runs on Windows and Linux operating systems. Most microcontroller systems are limited to Windows.

3. SIMPLE PROGRAMMABLE PLATFORM:

The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well.so students learning to program in that environment will be familiar with how the Arduino IDE works.

4. OPEN SOURCE AND EXTENSIBLE SOFTWARE:

The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.

5. OPEN SOURCE AND EXTENSIBLE HARDWARE:

The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it.

Raspberry Pi

A Raspberry Pi is a credit card-sized computer originally designed for education, inspired by the 1981 BBC Micro. Creator Eben Upton's goal was to create a low-cost device that would improve programming skills and hardware understanding at the pre-university level. But thanks to its small size and accessible price, it was quickly adopted by tinkerers, makers, and electronics enthusiasts for projects that require more than a basic microcontroller.

The Raspberry Pi is slower than a modern laptop or desktop but is still a complete Linux computer and can provide all the expected abilities that implies, at a low-power consumption level.

There are a two Raspberry Pi models, the A and the B, named after the aforementioned BBC Micro, which was also released in a Model A and a Model B. The A comes with 256MB of RAM and one USB port. It is cheaper and uses less power than the B. The current model B comes with a second USB port, an ethernet port for connection to a network, and 512MB of RAM.

The Raspberry Pi was designed for the Linux operating system, and many Linux distributions now have a version optimized for the Raspberry Pi.

The Raspberry Pi is not the only small computing device out there. The Arduino is another hobbyist board, which is geared towards those wanting to build out electronics projects.

The full specs for the Raspberry Pi 3 include:

- CPU: Quad-core 64-bit ARM Cortex A53 clocked at 1.2 GHz
- GPU: 400MHz VideoCore IV multimedia
- Memory: 1GB LPDDR2-900 SDRAM (i.e. 900MHz)

- USB ports: 4
- Video outputs: HDMI, composite video (PAL and NTSC) via 3.5 mm jack
- Network: 10/100Mbps Ethernet and 802.11n Wireless LAN
- Peripherals: 17 GPIO plus specific functions, and HAT ID bus
- Bluetooth: 4.1
- Power source: 5 V via MicroUSB or GPIO header
- Size: 85.60mm × 56.5mm
- Weight: 45g (1.6 oz)

Advantages:

Although Raspberry Pi is as small as the size of a credit card, it works as if a normal computer at a relatively low price. it is possible to work as a low-cost server to handle light internal or web traffic. Grouping a set of Raspberry Pi to work as a server is more cost- effective than a normal server. If all light traffic servers are changed into Raspberry Pi, it can certainly minimize an enterprise’s budget.

Disadvantages:

Even though Raspberry Pi can perform different tasks, there are some limitations due to its hardware. Because of its processor, it cannot run X86 operating systems. Some common ones like Windows and Linux distros are not compatible. In addition, some applications which require high demands on CPU processing are off-limits. Users must not use normal computer standards to judge Raspberry Pi. It can work as a personal computer, but cannot replace it.

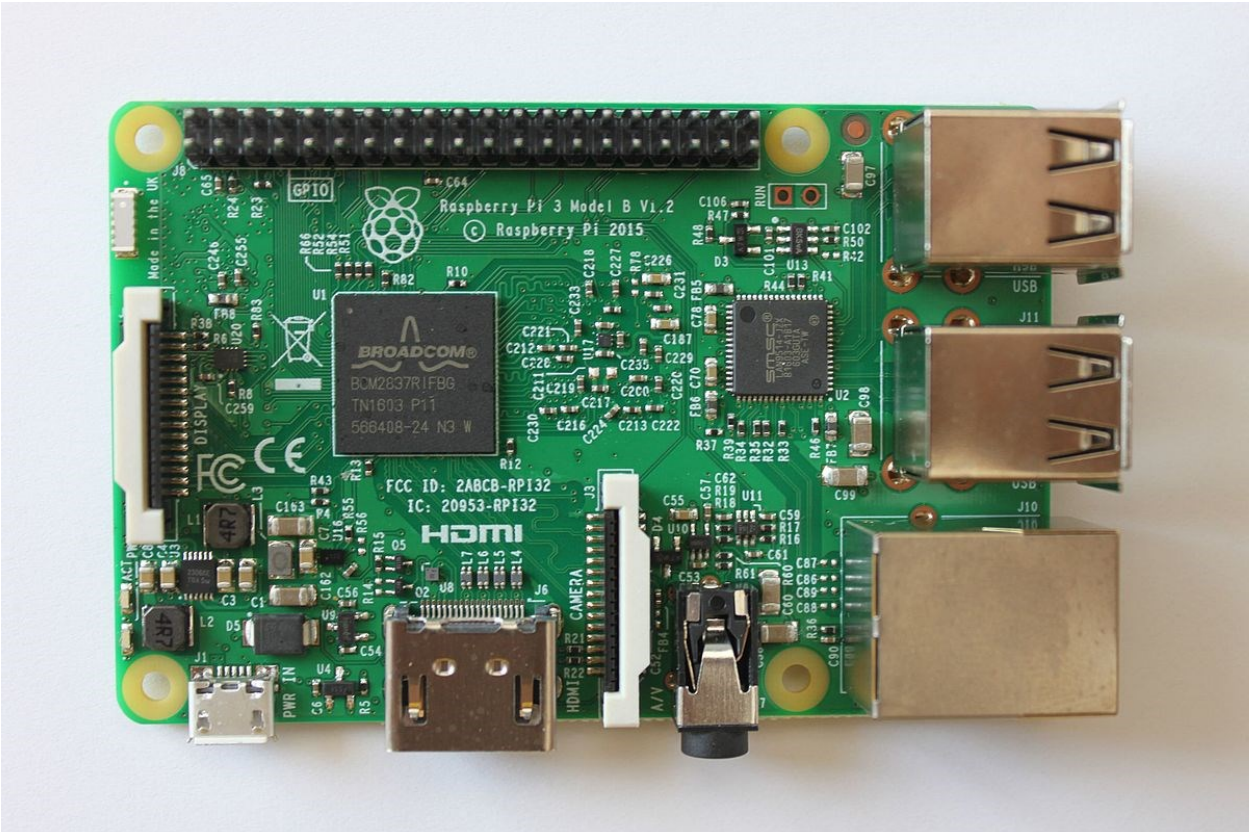


Fig 4. Raspberry Pi 3

NRF24L01

The nRF24L01 is a highly integrated, ultra-low power (ULP) 2Mbps RF transceiver IC for the 2.4GHz ISM (Industrial, Scientific and Medical) band. With peak RX/TX currents lower than 14mA, a sub µA power down mode, advanced power management, and a 1.9 to 3.6V supply range, the nRF24L01 provides a true ULP solution enabling months to years of battery lifetime when running on coin cells or AA/AAA batteries.

The nRF24L01 integrates a complete 2.4GHz RF transceiver, RF synthesizer, and baseband logic supporting a high-speed SPI interface for the application controller. No external loop filter, resonators, or VCO varactor diodes are required, only a low cost ±60ppm crystal, matching circuitry, and antenna.

The Nordic nRF24L01 is available in a compact 20-pin 4 x 4mm QFN package.

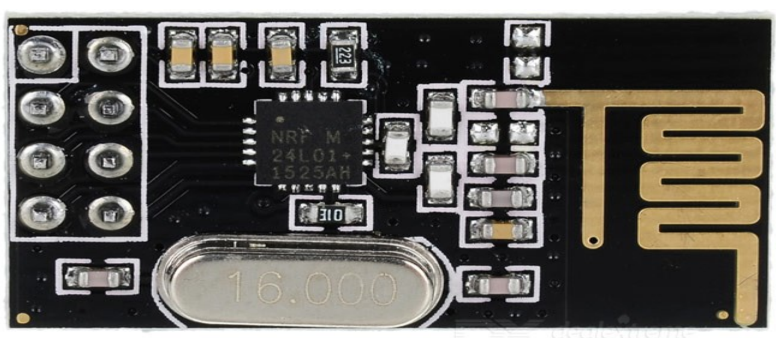


Fig. 5 Nrf24l01

SPECIFICATIONS:

- Ultra low power 2.4GHz RF Transceiver IC
- GFSK modulation, 1 or 2MHz bandwidth
- 0, -6, -12, and -18dBm programmable TX output power
- Configurable on-air data rate of 1Mbps or 2Mbps
- -82dBm RX sensitivity at 2Mbps
- -85dBm RX sensitivity at 1Mbps
- Compatible with a 16MHz ±60ppm crystal
- Automatic packed detection and validation
- Dynamic payload length, 1 to 32Bytes
- Auto retransmit
- Auto Acknowledgment with optional payload
- Internal linear voltage regulator
- 1.9 to 3.6V supply range
- SPI, up to 10Mbps

Power Problems:

Many users have had trouble getting the nRF24L01 modules to work. Many times the problem is that the 3.3V Power to the module does not have enough current capability, or current surges cause problems. Here are suggestions:

1.) Connect a 3.3 uF to 10 uF (Microfarad) capacitor directly on the module from +3.3V to Gnd (Watch + and - !) [Some users say 10 uF in parallel with 0.1uF is best] This is especially important if you are connecting the module with jumper wires. Or you are using the regular Arduino UNO which provides only 50 mA at 3.3V

2.) Use the RF24 Library from TMRH20 (below) and set power low to minimize power requirements: radio.setPALevel(RF24_PA_MIN); Space the two radios about a meter apart. After you have things working, and if you know you have enough 3.3V current (up to 250 mA or more) then try higher power. The possibilities are: RF24_PA_MIN, RF24_PA_LOW, RF24_PA_HIGH and RF24_PA_MAX

Range:

Range is very dependent on the situation and is much more with clear line of sight outdoors than indoors with effects of walls and materials. The usual distance quoted by different suppliers for the low-power version module with the single chip is 200 Feet or 100 Meters. This is for open spaaaaaace between units operating at a Data Rate of 250KHz. Indoors the range will be less due to walls etc... The example with radio.setPALevel(RF24_PA_LOW); will be only 10 feet or so. But reliable

Here are details of the Pinout and connections to Arduino:

Signal	RF module pin	cable colour	"Base Module" pin	Arduino pin for RF24 Library
GND	1	Brown	GND	GND *
VCC	2	Red	VCC	3.3V *
CE	3	Orange	CE	9
CSN	4	Yellow	CSN	10
SCK	5	Green	SCK	13
MOSI	6	Blue	MO	11
MISO	7	Violet	MI	12
IRQ	8	Gray	IRQ	2

Bluetooth

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm.

Specifications:

Hardware Features:

- Typical -80dBm sensitivity
- Up to +4dBm RF transmit power
- Low Power 1.8V Operation ,1.8 to 3.6V I/O
- PIO control
- UART interface with programmable baud rate
- With integrated antenna
- With edge connector

Software Features:

- Default Baud rate: 38400, Data bits:8, Stop bit:1,Parity:No parity, Data control: has Supported baud rate: 9600,19200,38400,57600,115200,230400,460800.
- Given a rising pulse in PIO0, device will be disconnected.
- Status instruction port PIO1: low-disconnected, high-connected;

- PIO10 and PIO11 can be connected to red and blue led separately. When master and slave are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s.
- Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing PINCODE:"0000" as default
- Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

Sections of the BT Board:

- The Green HC-05 sub-module is soldered on top of the Blue BT Board
- The HC-05 module includes the Radio and Memory chips, 26 MHz crystal, antenna and RF matching network.
- The right section of the BT Board has connection pins for power and signals as well as a 5V to 3.3V Regulator, LED, and level shifting.

Fig 6 BT board

HC-05 PinOut:

KEY: If brought HIGH before power is applied, forces AT Command Setup Mode. LED blinks slowly (2 seconds)

VCC: +5 Power

GND: System / Arduino Ground

TXD: Transmit Serial Data from HC-05 to Arduino Serial Receive. NOTE: 3.3V HIGH level: OK for Arduino

RXD: Receive Serial Data from Arduino Serial Transmit

STATE: Tells if connected or not

COMMAND and DATA TRANSFER MODES:

The module has two modes of operation, Command Mode where we can send AT commands to it and Data Mode where it transmits and receives data to another bluetooth module.

The default mode is DATA Mode, and this is the default configuration, that may work fine for many applications:

- Baud Rate: 9600 bps, Data : 8 bits, Stop Bits: 1 bit, Parity : None, Handshake: None
- Passkey: 1234
- Device Name: HC-05

Command Mode Commands:

The format of commands is:

- Always starts with "AT"
- Then "+" followed by <ParameterName>
- Then either:
  - ? (returns current value of parameter)
  - = (New Value of parameter)

A few examples:

- AT (AT Test command. Should respond with OK)
- AT+VERSION? (show the firmware version)
- AT+UART=9600,0,0 (Set baud rate to 9600, 1 stop bit, no parity)

Bluetooth Master Mode:

To configure the module as Bluetooth Master and to pair with another Bluetooth module follow these steps. First we need to put the module into command mode as above by pulling the CMD pin high before power on.

Enter these commands in order:

- AT+RMAAD Clear any paired devices
- AT+ROLE=1 Set mode to Master
- AT+RESET After changing role, reset is required
- AT+CMODE=0 Allow connection to any address (I have been told this is wrong and CMODE=1 sets "any address"
- AT+INQM=0,5,5 Inquire mode - Standard, stop after 5 devices found or after 5 seconds
- AT+PSWD=1234 Set PIN. Should be same as slave device
- AT+INIT Start Serial Port Profile (SPP) (If Error(17) returned - ignore as profile already loaded)
- AT+INQ Start searching for devices

A list of devices found will be displayed, one of which is the slave module. The format of the output is

+INQ:address,type,signal

The address of the module is what we need and is in the format 0123:4:567890

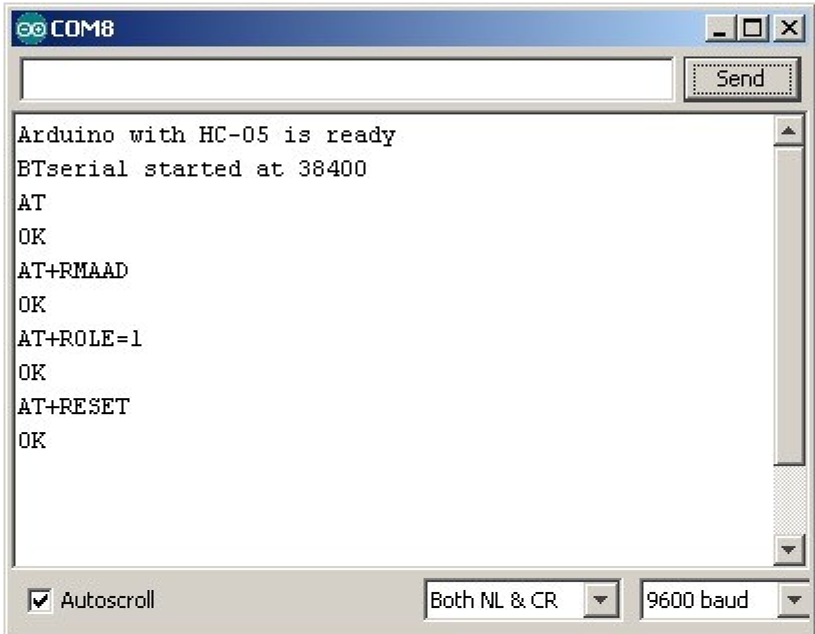


Fig7 master configuration

Slave Mode:

The HC-05 bluetooth module can also act as a slave. There are fewer commands to set this up:

- AT+ORGL Reset to defaults
- AT+RMAAD Clear any paired devices
- AT+ROLE=0 Set mode to SLAVE
- AT+ADDR Display SLAVE address

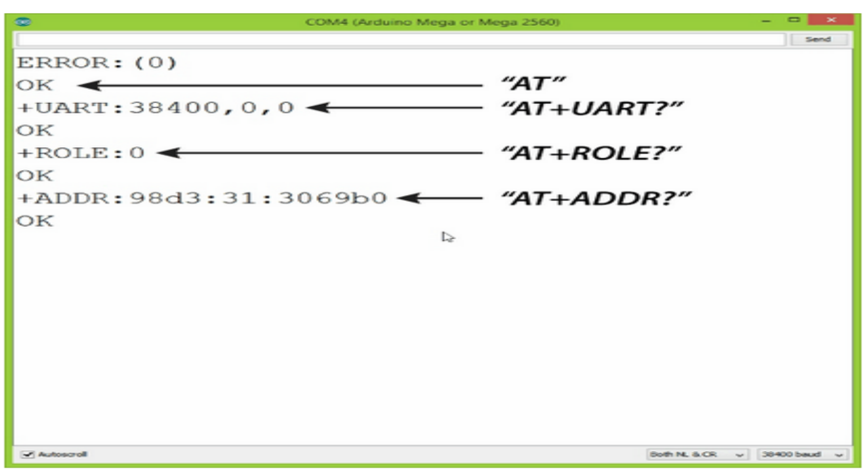


Fig8 slave configuration

Basic AT commands:

Command	Return	Parameter	Description
AT	OK	None	Test
AT+VERSION?	+VERSION:<Param> OK	Param: Version number	Get the soft version
AT+ORGL	OK	None	Restore default status
AT+ADDR?	+ADDR: <Param> OK	Param: Bluetooth address	Get module Bluetooth address
AT+NAME=<Param>	OK	Param: Bluetooth device name	Set device's name
AT+NAME?	+NAME:<Param> OK	Param: Bluetooth device name	Inquire device's name
AT+ROLE=<Param>	OK	Param:0=Slave role; 1=Master role; 2=Slave-Loop role	Set module role
AT+ ROLE?	+ ROLE:<Param>	Param:0=Slave role; 1=Master role; 2=Slave-Loop role	Inquire module role

AT+UART=<Param>,<Param2>,<Param3>	OK	Param1: baud rate(bits/s); Param2: stop bit; Param3: parity bit	Set serial parameter
AT+ UART?	+UART=<Param>,<Param2>,<Param3> OK	Param1: baud rate(bits/s); Param2: stop bit; Param3: parity bit	Inquire serial parameter

BLUETOOTH LOW ENERGY (BLE)

BLE is also called as Bluetooth 4.0.BLE is not just a another version of Bluetooth it's a whole new technology It has some benefits as well as drawbacks over Bluetooth 2.0.The following is the picture of BLE module .

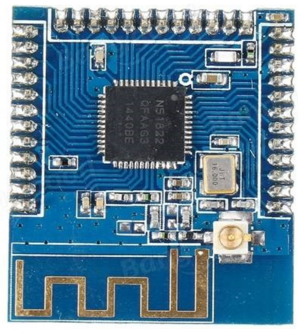


Fig9 BLE module

Features:

- Bluetooth v4.2 compliant protocol stack
- Generic Access Profile (GAP) Features
- Broadcaster, Observer, Peripheral and Central roles
- Supports role reversal between Peripheral and Central
- User-defined advertising data
- Bonding support for up to four devices
- Security modes 1 and 2

Components:

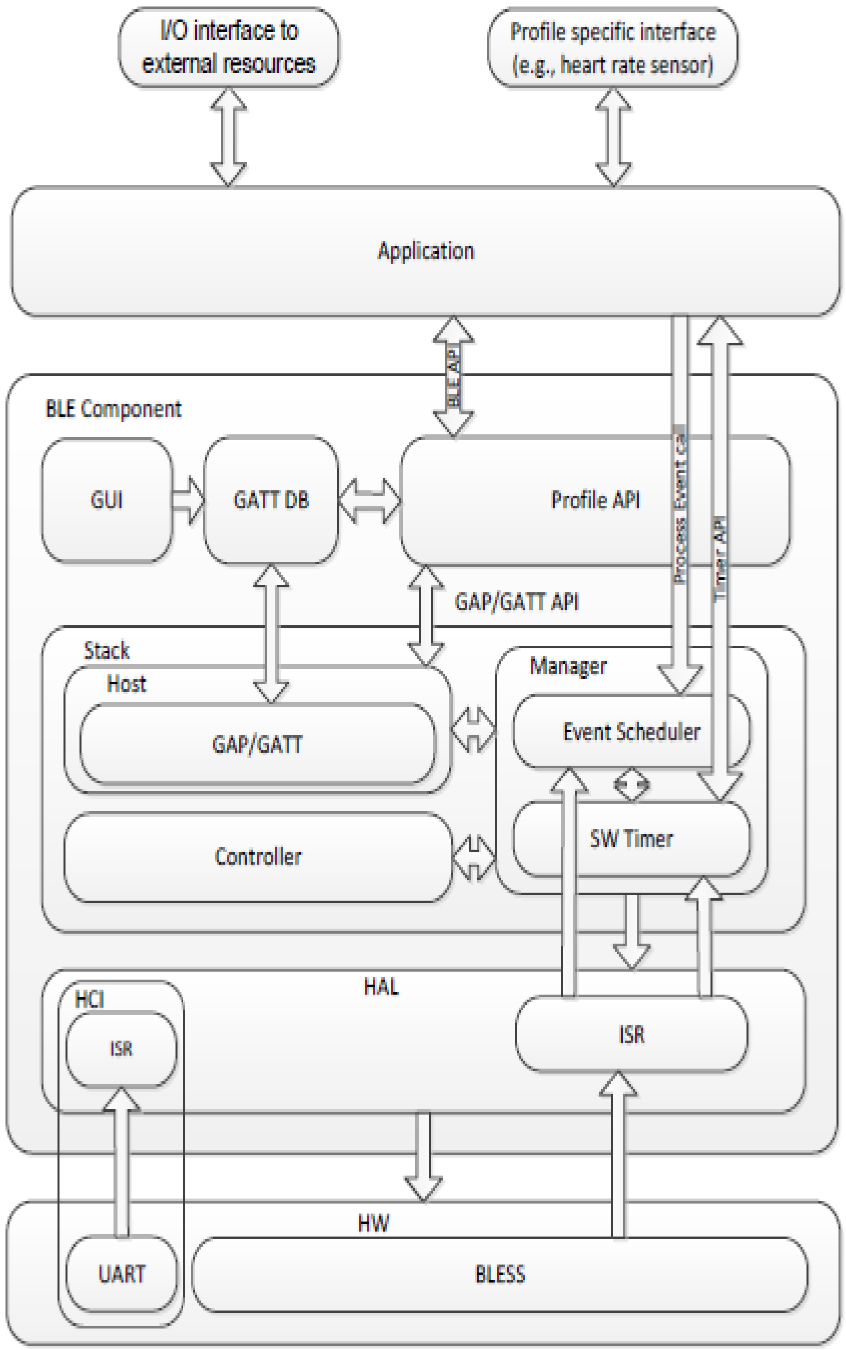


Fig10 BLE compoents

Some of the important differences between Bluetooth 2.1 and Bluetooth 4.0:

	Bluetooth V2.1	Bluetooth Low Energy
Standardization Body	Bluetooth SIG	Bluetooth SIG
Range	~30 m (class 2)	~50 m
Frequency	2.4–2.5 GHz	2.4–2.5 GHz
Bit Rate	1–3 Mbit/s	~200 kbit/s
Set-Up Time	<6 s	<0.003 s
Voice Capable?	Yes	No
Max Output Power	+20 dBm	+10 dBm
Modulation Scheme	GFSK	GFSK
Modulation Index	0.35	0.5
Number of Channels	79	40
Channel Bandwidth	1 MHz	2 MHz

Fig11 BLE Vs Bluetooth

LOW POWER:

The spec sheet on BLE is a little disheartening at first. It's slower than regular BT by quite a bit, to the point that voice and other audio connections aren't possible. But, well, this:

Bluetooth low energy single-mode chips consume less power than dual-mode chips and are optimized to run off a coin cell battery for a year or more.

BLE hardware will come in two flavors: single mode and dual mode. Dual mode hardware is a hybrid of traditional and BLE, and can operate in both high-power and low-power modes. (In the case of a phone, one for typical Bluetooth duties, like audio streaming, and the other for something like connecting to a heartrate monitor, or forwarding text messages to a watch or tablet.)

BLE RANGE CALCULATIONS:

Antenna: The antenna is installed inside of the block-wall building on the second floor.

Receiver: Data will be transmitted down, with the receiver one meter off the ground.

With one antenna on the second floor, the expected Bluetooth Low Energy range is 77 meters.

There are many factors affecting Bluetooth range, typically:

1) The output power of the transmitter
2) The sensitivity of the receiver
3) Physical obstacles in the transmission path
4) The antennas

DATA TRANSFER:

As BLE is designed to consume LOW ENERGY. Due to this it cannot transfer data at very high speed. Because of low energy it cannot be used to stream any audio or video. Usually the bit rate in BLE is 200kbps whereas in Bluetooth 2.0 it is 1 to 3 Mbps.

LoPy

The LoPy can act as both a LoRa Nano Gateway and a multi-bearer (LoRa, WiFi and BLE) development platform. It is programmable with MicroPython and the Pymakr IDE for fast IoT application development, easy programming in-field and extra resilience with network failover. The best blend of speed to deployment and access to new LPWAN networks rolling out across Europe, USA, Africa and India.

With LoRa, Wifi and BLE, the LoPy is the only triple network MicroPython enabled micro controller on the market today – the perfect enterprise grade IoT platform for your connected Things. With the latest Espressif chipset the LoPy offers a perfect combination of power, friendliness and flexibility.

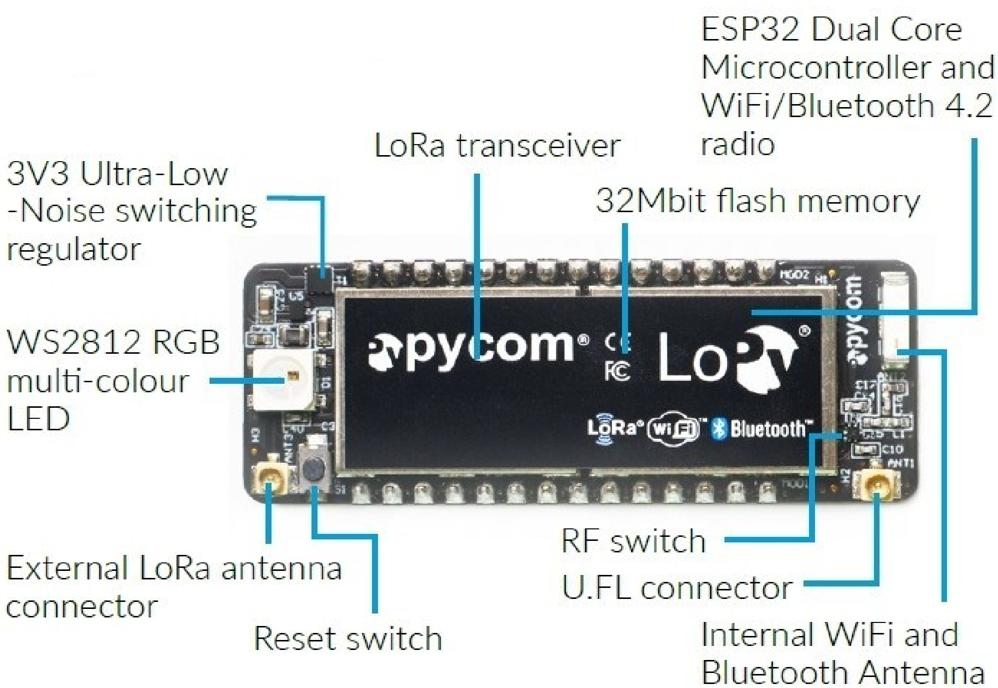


Fig12 lopy module

How Does LoraWAN Work?

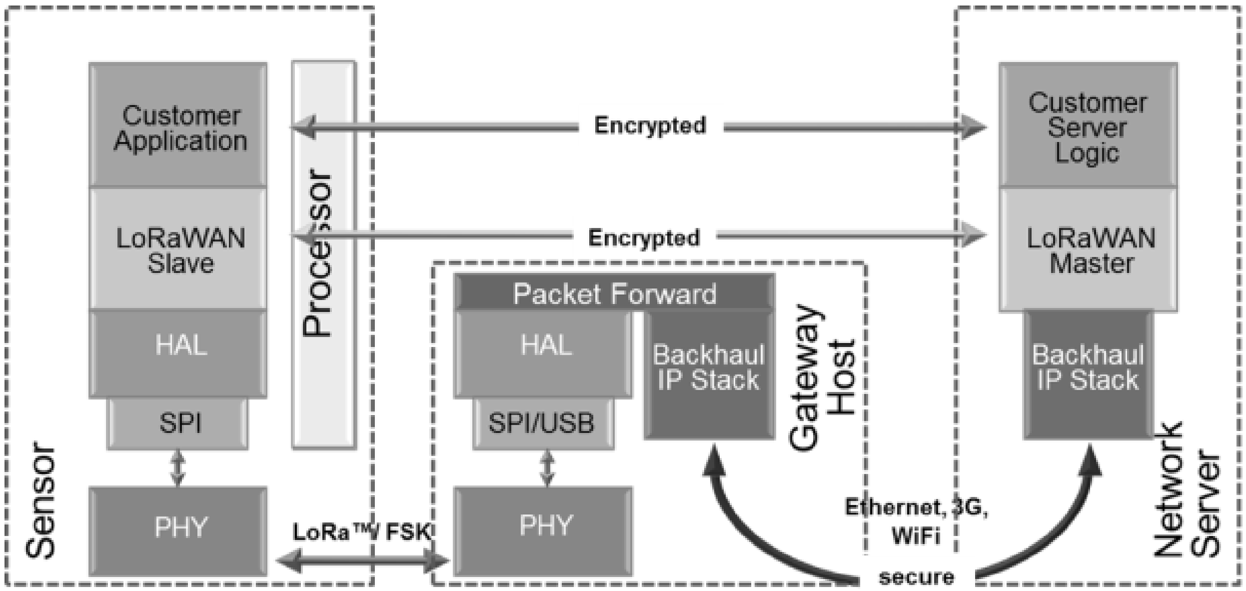


Fig13 Lorawan working

Frequency Bands:

- Lora WAN operates in unlicensed radio spectrum.
- It is similar to WiFi, which uses the 2.4GHz and 5GHz ISM bands
- Lora WAN uses lower radio frequencies with a longer range.
- This poses a challenge for LoraWAN that tries to be as uniform as possible in all different regions of the world. As a result, Lora WAN is specified for a number of bands for these regions.
- A major portion of the 868 MHz LoRa frequency channels are in the license free 865 MHz – 867 MHz band in India. Seven out of the eight channels are in the license free band in India.

Create and connect your things everywhere. Fast.

- Powerful CPU, BLE and state of the art WiFi radio.
- Can also double up as a Nano LoRa gateway
- MicroPython enabled, use the Linux of IoT for 10x faster programming
- Super small and lightweight
- Fits in a standard breadboard (with headers)
- Ultra-low power usage: a fraction compared to other connected micro controllers

SPECIFICATIONS

Processing

– Espressif ESP32
– Dual processor and WiFi radio system on chip
– Network processor handles the WiFi connectivity and the IPv6 stack
– Main processor is entirely free to run the user application
– An extra ULP-coprocessor that can monitor GPIOs, the ADC channels and control most of the internal peripherals during deep-sleep mode while only consuming 25uA

Operating Frequencies

– 868 MHz (Europe) at +14dBm maximum
– 915 MHz (North and South America, Australia and New Zealand) at +20dBm maximum

Range Specification

– Node range: Up to 40km

– Nano-Gateway: Up to 22km

– Nano-Gateway Capacity: Up to 100 nodes

Use the Pymakr IDE

Super easy code editor to write your Python scripts

Quick Verification

For easy and fast debugging use the interactive shell that is accessible through telnet or one of the serial ports

Easy Upload

Upload your scripts and any other files you want to the LoPy via the FTP server

Locally or remotely

Reset the LoPy (you can do it locally, or remotely via Telnet)

Mechanical

– Size: 55mm x 20mm x 3.5mm

– Operating temperature: -40 to 85 degrees celsius

– 3V3 Ultra-Low-Noise switching regulator

– WS2812 RGB multi-colour LED

– External LoRa antenna connector

– Reset switch

– LoRa transceiver

– 32Mbit flash Memory

– ESP32 Dual Core Micro controller and WiFi/Bluetooth 4.2 radio

– RF switch

– U.FL connector

– Internal WiFi and Bluetooth Antenna

Interfaces

– 2 x UART, 2 x SPI, I2C, I2S, micro SD card

– Analog channels: 8x12 it ADCs

– Timers: 4x16 bit with PWM and input capture

– DMA on all peripherals

– GPIO: Up to 24

Security and Certifications

– SSL/TLS support

– WPA Enterprise security

– FCC – 2AJMTWIPY2R

– CE 0700

Memory

– RAM: 512KB

– External flash 4MB

– Hardware floating point acceleration

– Python multi-threading

Hash / encryption

SHA, MD5, DES, AES

Wifi

802.11b/g/n 16 mbps

Bluetooth

Low energy and classic

RTC

Running at 32KHz

Power

– Input: 3.3V – 5.5V

– 3v3 output capable of sourcing up to 400mA

– Wi-Fi: 12mA in active mode, 5uA standby

– Lora: 15mA in active mode, 1-0uA in standby

LoRa Specification

– Semtech LoRa transceiver SX1272

– LoRaWAN stack

– Class A and C devices

Performance Parameters (Qos):

- **Network Delay** is an important design and performance characteristic of a computer network. The delay of a network specifies how long it takes for a bit of data to travel across the network from one node or endpoint to another.

- **Throughput** is the rate of successful message delivery over a communication channel.

- **Reliability** is the overall consistency of a measure. A measure is said to have a high reliability if it produces similar results under consistent conditions.

    - A **reliable** protocol provides reliability properties with respect to the delivery of data to the intended recipient(s), as opposed to an unreliable protocol, which does not provide notifications to the sender as to the delivery of transmitted data

- **Network latency is** the time from the source sending a packet to the destination receiving it.

Ultrasonic Ranging Module HC - SR04

Product Features:

- Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:
  - Using IO trigger for at least 10us high level signal,
  - The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
  - IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.

Fig14 HCSR04 module

Results

Arduino or Raspberry Pi?

- The Arduino employs an 8-bit AT mega series microcontroller.
- The Raspberry Pi is based around a 32-bit ARM processor.
- The Arduino is typically clocked at between 8-16MHz and with 2-8kB of RAM.
- The RPi can be clocked at up to 1GHz and may have up to 512MB of RAM.
- On top of which the Pi has a GPU and video outputs, Ethernet as standard and USB host ports.
- The Pi has a clear advantage for complex networked and high performance embedded applications and those which involve driving a video display or USB peripherals.
- Arduino will only consume micro-watts when in sleep mode compared to the watts drawn by an idle Raspberry Pi running Linux.

Home Automation System:

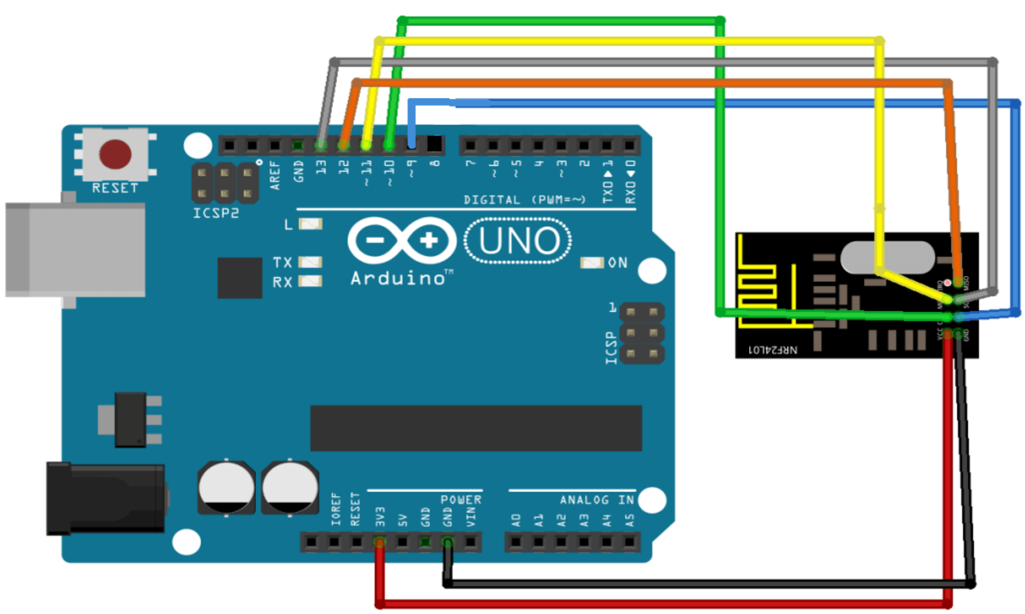


Fig15 arduino nrf connection

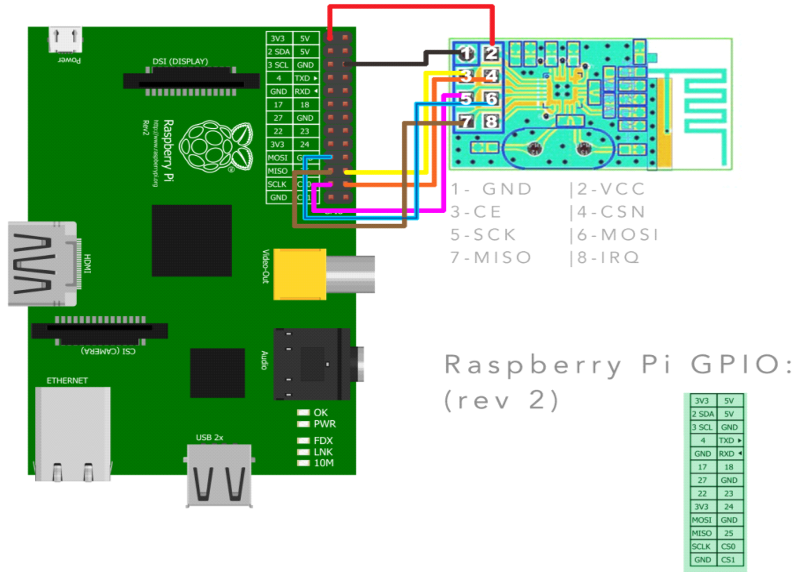


Fig16 Rasberrypi nrf connection

Software:

- Our plan to communicate between these two is to use a application to run a binary ‘sudo ./remote -m 81‘ and ‘sudo ./remote -m 80‘ sending a message to the Arduino with the text 81 and 80 .
- The message is composed by pin identifier (8) and action to perform (0|1|2).
- After the action has been read, all the other numbers identify the pin which the action refers to.

Flow model:

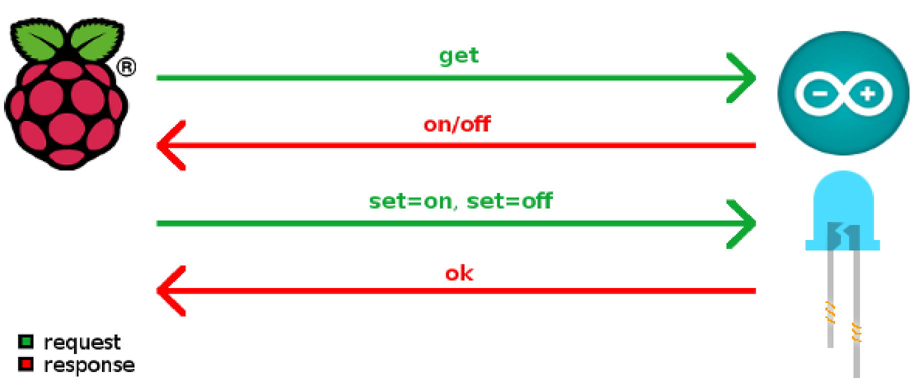


Fig17 flow model

Sketching:

- We used the library package 'Optimized RF24 NRF24L01 Radio Library for Arduino' - despite the name, this package also includes the libraries that we're going to use within the Raspberry.
- So, download the library, import it into the Arduino (Sketch->Import library).
- Prepare the environment at the Raspberry:
    - *$ git clone https://github.com/TMRh20/RF24.git*
    - *$ cd RF24*
    - *$ sudo make install*

NRF24L01 with Arduino and Raspberry Pi

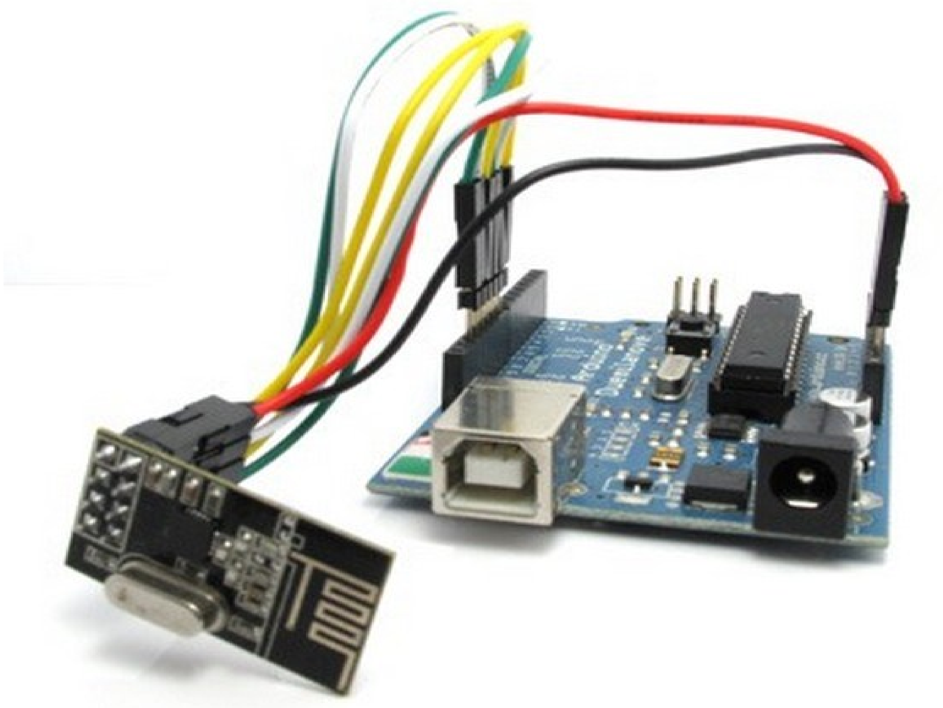


Fig18 arduino and nrf

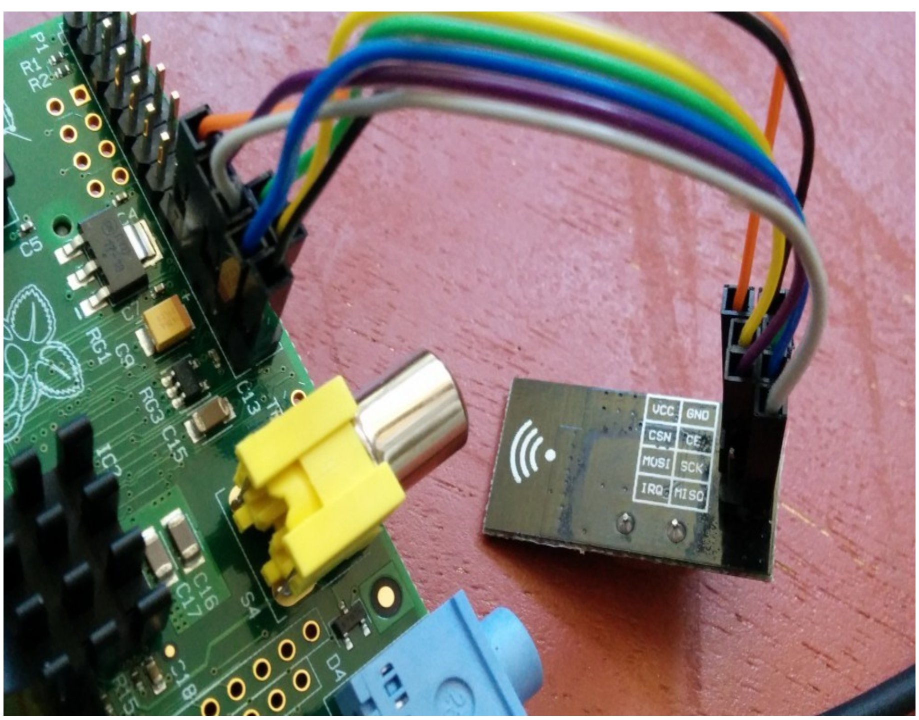


Fig19 rasberrypi and nrf

NRF24L01 setup for Arduino:

In Arduino

```
RF24/examples/GettingStarted/
ROLE: Pong back
*** PRESS 'T' to begin transmitting to the other node
STATUS           = 0x0e RX_DR=0 TX_DS=0 MAX_RT=0 RX_P_NO=7 TX_FULL=0
RX_ADDR_P0-1      = 0xf0f0f0f0e1 0xf0f0f0f0d2
RX_ADDR_P2-5      = 0xc3 0xc4 0xc5 0xc6
TX_ADDR          = 0xf0f0f0f0e1
RX_PW_P0-6        = 0x20 0x20 0x00 0x00 0x00 0x00
EN_AA            = 0x3f
EN_RXADDR         = 0x03
RF_CH            = 0x4c
RF_SETUP          = 0x07
CONFIG           = 0x0f
DYNPD/FEATURE     = 0x00 0x00
Data Rate        = 1MBPS
Model            = nRF24L01+
CRC Length       = 16 bits
PA Power         = PA_HIGH
```

Fig20 nrf arduino setup

NRF24L01 setup for Raspberry Pi:

In the Raspberry

```
================ SPI Configuration ================
CSN Pin          = CE0 (PI Hardware Driven)
CE Pin           = Custom GPIO25
Clock Speed      = 8 Mhz
================ NRF Configuration ================
STATUS           = 0x0e RX_DR=0 TX_DS=0 MAX_RT=0 RX_P_NO=7 TX_FULL=0
RX_ADDR_P0-1     = 0x65646f4e32 0x65646f4e31
RX_ADDR_P2-5     = 0xc3 0xc4 0xc5 0xc6
TX_ADDR          = 0x65646f4e32
RX_PW_P0-6       = 0x20 0x20 0x00 0x00 0x00 0x00
EN_AA            = 0x3f
EN_RXADDR        = 0x02
RF_CH            = 0x4c
RF_SETUP         = 0x07
CONFIG           = 0x0e
DYNPD/FEATURE    = 0x00 0x00
Data Rate        = 1MBPS
Model            = nRF24L01+
CRC Length       = 16 bits
PA Power         = PA_MAX

 ************ Role Setup ***********
Choose a role: Enter 0 for pong_back, 1 for ping_out (CTRL+C to exit)
```

Fig21 nrf rasberrypi setup

NRF24L01 Performance Analysis:

Range	Latency	Packets Acknowledged	Throughput
less than 1 meter	8 ms	118 packets	99%
100 meters	11ms	94 packets	96%
150 meters	23 ms	57 packets	95%
200 meters	45 ms	14 packets	81%

Mode	Power Consumption
Transmission	11.3mA
Reception	13.1mA
Standby	900nA

NRF24L01 Output:

```
RF24/examples/GettingStarted
*** PRESS 'T' to begin transmitting to the other node
*** CHANGING TO TRANSMIT ROLE -- PRESS 'R' TO SWITCH BACK
Now sending
Sent 523513060, Got response 523511232, Round-trip delay 1828 microseconds
Now sending
Sent 524519544, Got response 524517728, Round-trip delay 1816 microseconds
```

Fig 22 nrf output

Bluetooth with Arduino and Raspberry Pi

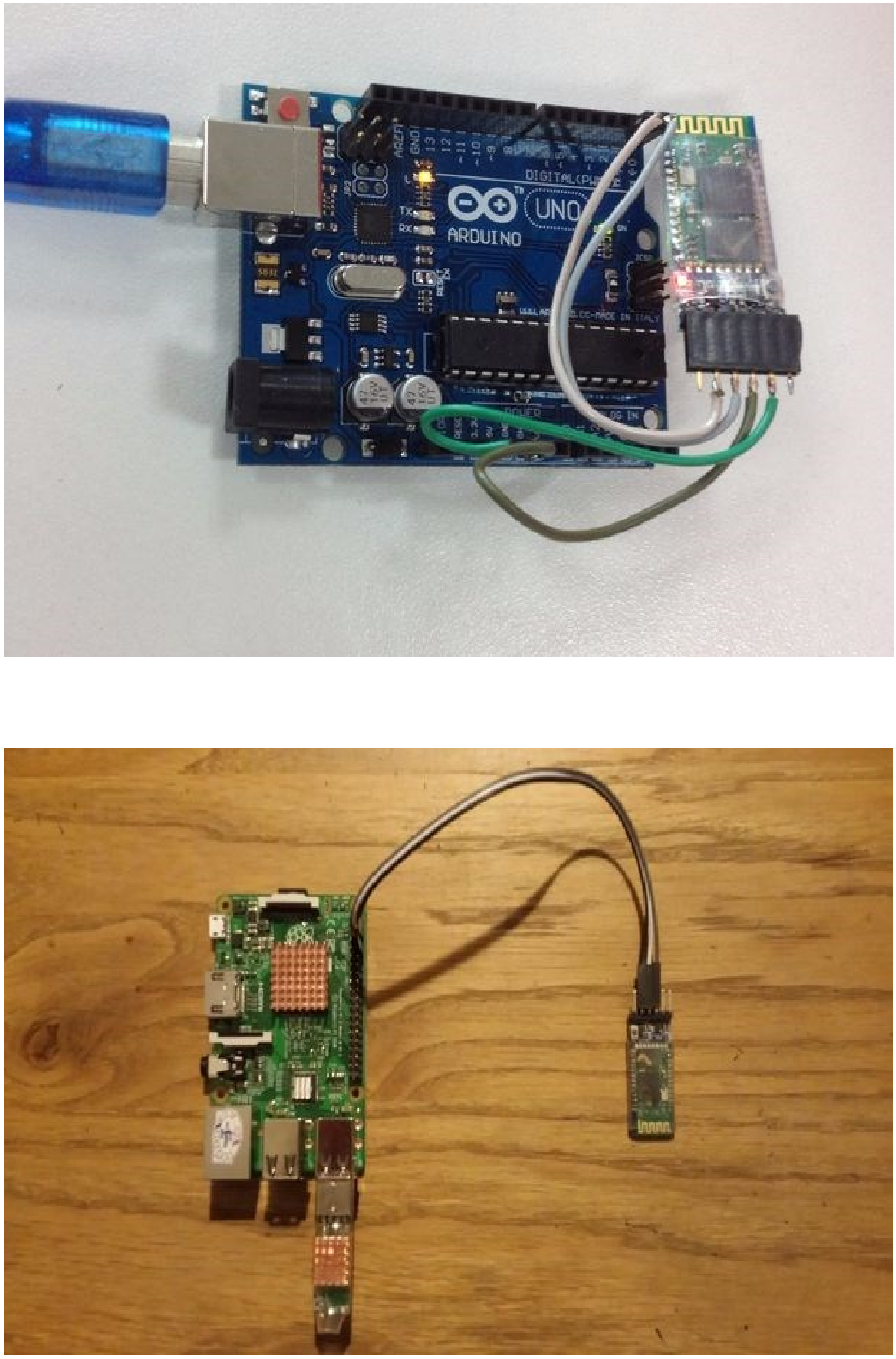


Fig23 Bluetooth with 1)arduino 2)rasberrypi

Bluetooth Performance Analysis:

Range	Latency	Throughput	Power Consumption
Less than 1 meter	220 ms	99%	40mA
10 meters	265ms	98%	41mA

Bluetooth Output:

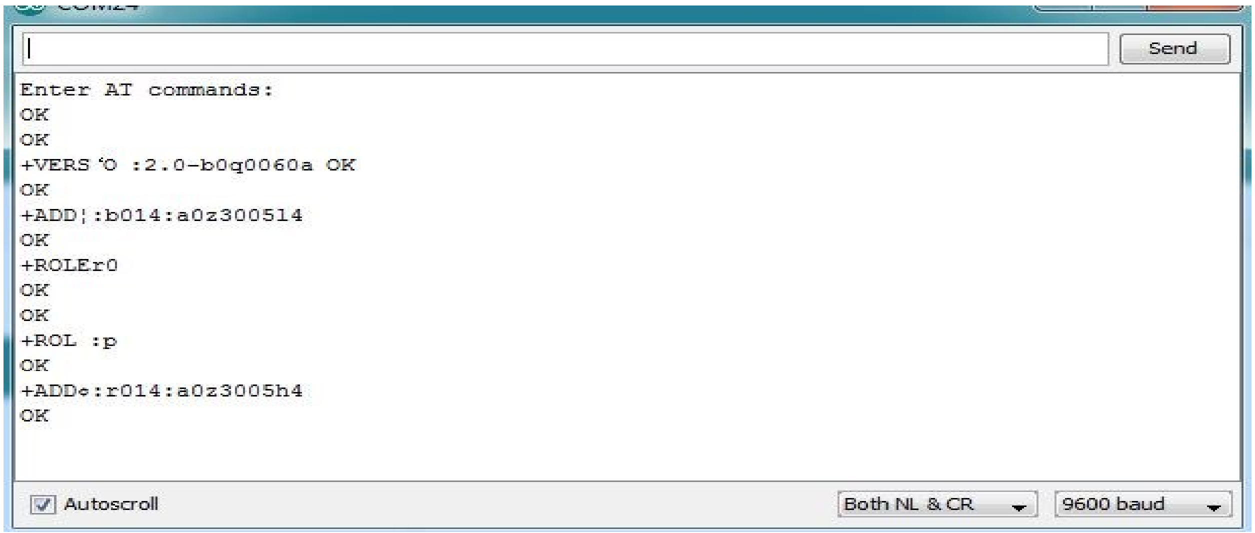


Fig24 Bluetooth output

BLE (Bluetooth Low Energy)

BLE Performance Analysis:

Range	Latency	Throughput	Power Consumption
Less than 1 meter	206 ms	99%	14mA
10 meters	225ms	98%	15mA
50 meters	270ms	98%	15mA

LoPy (Pycom)

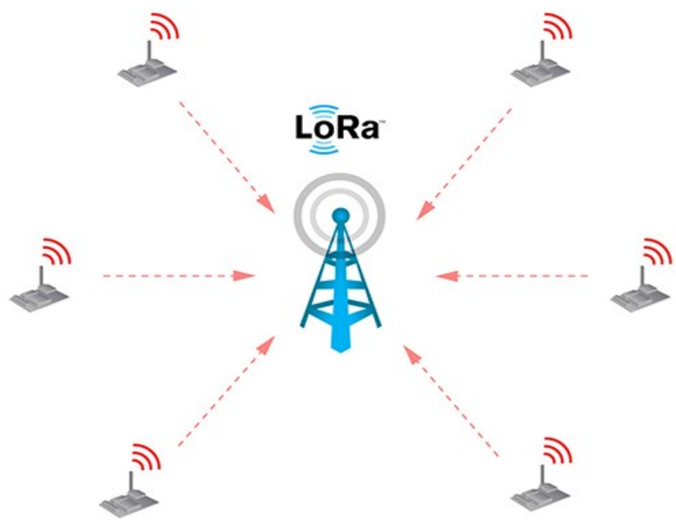
LoPy Performance Analysis:

State	Power Consumption
ON	2.8 mA
Transmitting data	38.9 mA
Receiving data	14.2 mA

Transmit mode	Latency (10–Byte Packet)	Throughput
Send unconfirmed at 5470 bps	2.7 seconds	98%
Send unconfirmed at 250 bps	4.2 seconds	99%
Send confirmed at 5470 bps	1.6 seconds	97%
Send confirmed at 250 bps	4.1 seconds	99%

LoPy (Unconfirmed):

```
{
    uint8_t port = 1;
    char data[]  = "010203040506070809";
    LoRaWAN.sendUnconfirmed( port, data);
}
```



LoPy (Confirmed):

```
{
    uint8_t port = 1;
    char data[]  = "010203040506070809";
    LoRaWAN.sendConfirmed( port, data);
}
```

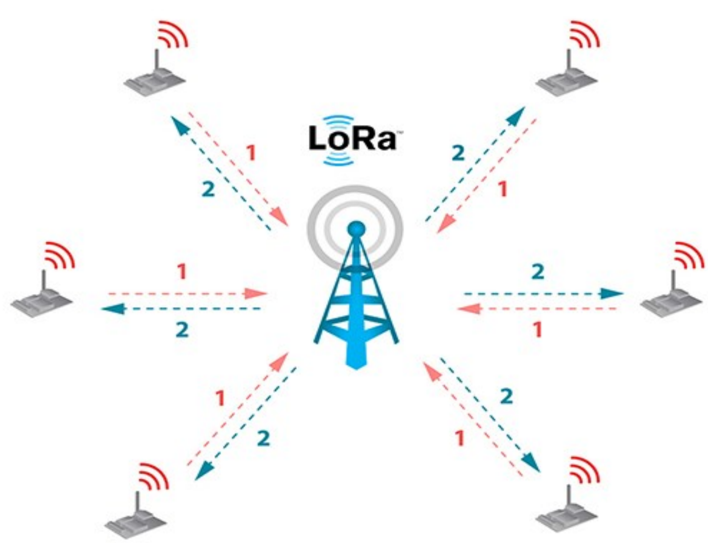
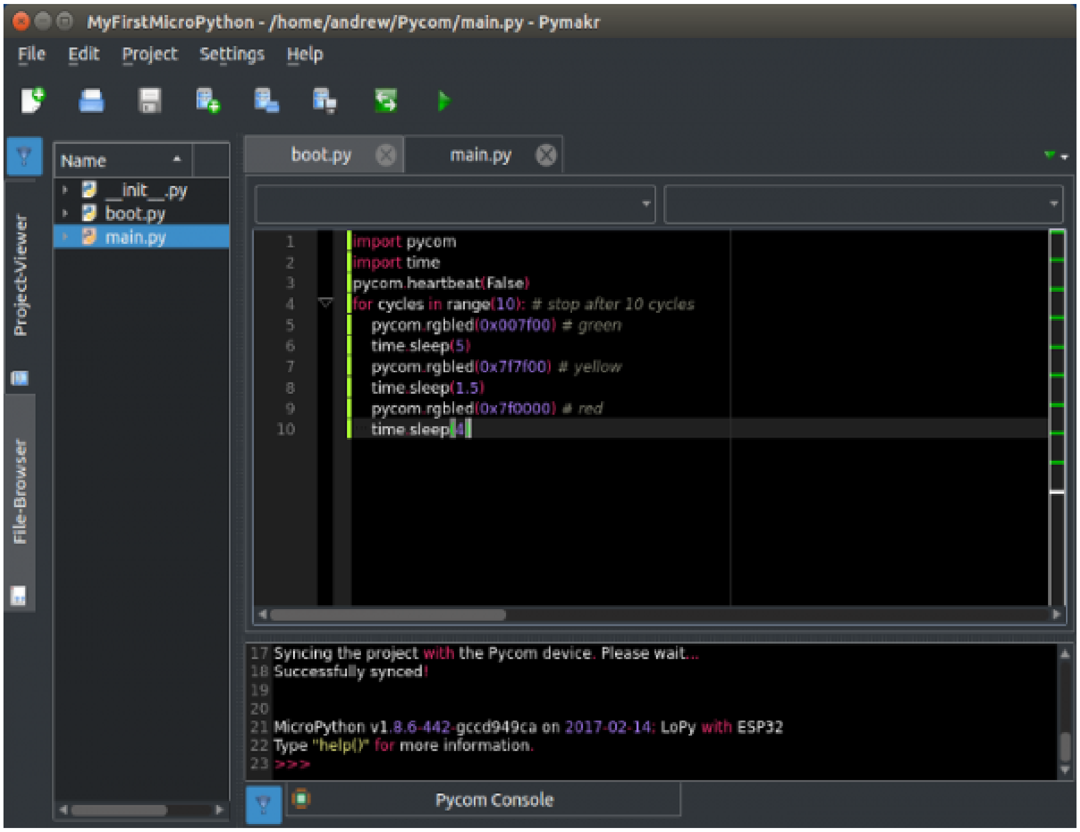


Fig25 a) lopy unconfirmed b) lopy confirmed

LoPy Output:



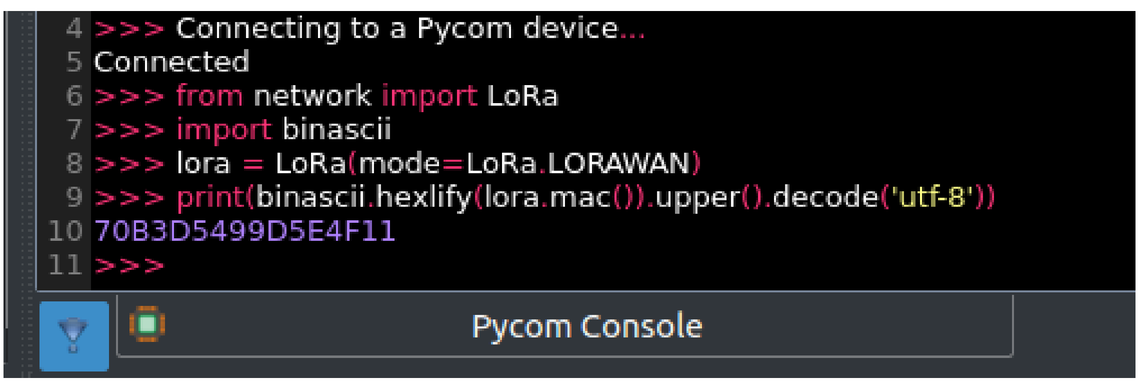


Fig26 lopy output

The Things Network

- It is a global community of 12776 people over 82 countries building a global Internet of Things data network.
- It uses a long range and low power radio frequency protocol called LoRaWAN and for short range Bluetooth 4.2.
- The technology allows for things to talk to the internet without 3G or WiFi. So no Wi-Fi codes and no mobile subscriptions.
- In order to provision our node on TTN, we need to first get the factory configured Device EUI, which is simple enough to do and involves entering a few lines of Python into the REPL (interpreter prompt). With this we can then register the LoPy against our test application via TTN Console.

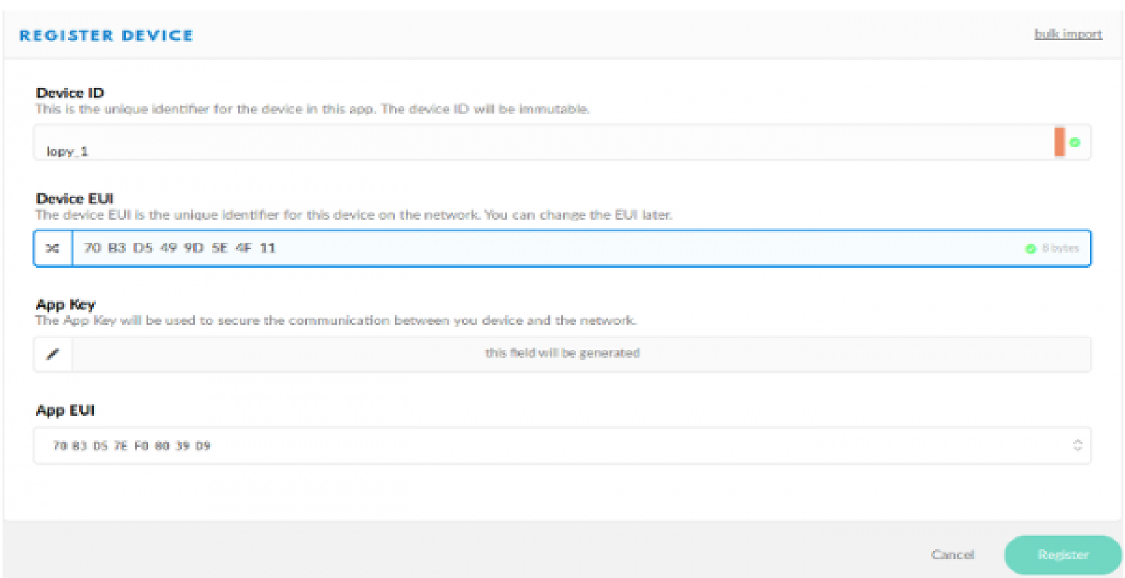


Fig27 Things network register window

- In India, there are 8 communities based in different cities, Bengaluru, Chennai, Coimbatore, Kochi, Mumbai, Mysore, New Delhi and Pune.
- Bengaluru is the official community of The Things Network in India.

Smart Garbage Management

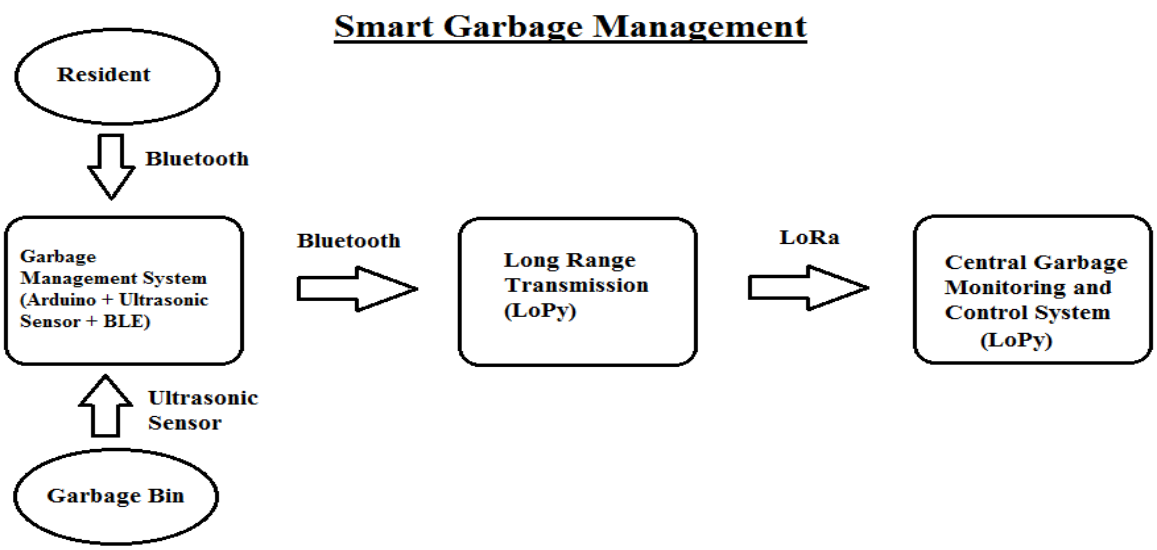


Fig 28 smart garbage management

Problem Scenario:

- The scenario is if the garbage bin is filled, it should be informed to the central garbage control system which will far from the house or the locality.
- If a resident has any complaint or he/she intentionally need to clear the thrash due to some reasons like bad odour, before the garbage bin is filled.

Solution Proposed:

- We used an Arduino microcontroller and LoPy module to interface the sensors and communication protocols.
- We used an ultrasonic sensor to know whether the garbage bin filled.
- If filled, it sends a message to the LoPy module using a BLE module.

- As Lopy is a Long Range communication module, it immediately sends this message to the Central Management System (LoPy) using LoRa protocol.
- If an individual is willing to call the service, he can send it through an app in the mobile which uses Bluetooth.

Smart Water Management

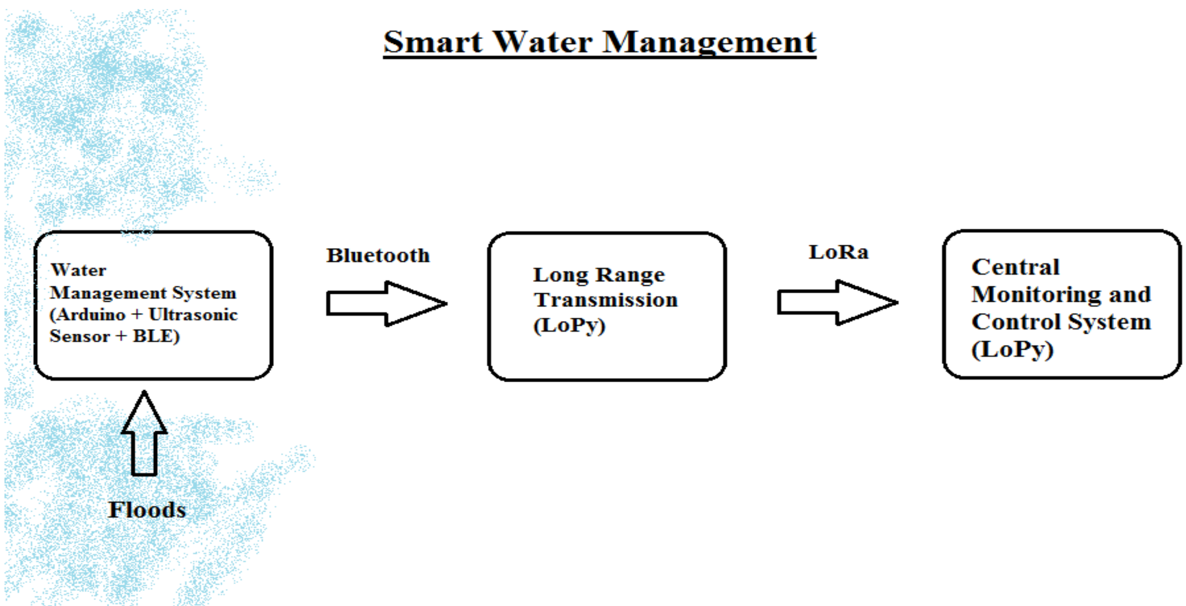


Fig 29 smart water management

Problem Scenario:

- When floods occur many lives near the coastal region are in danger.
- As they travel at a high rate of speed we need to quickly send alert to evacuate a large area near the region.
- We need a long range communication to take care of this.

Solution Proposed:

- We used an Arduino microcontroller and LoPy module to interface the sensors and communication protocols.
- We used an ultrasonic sensor to know whether the water is approaching beyond the threshold level.
- If crossed the value, it sends a message to the LoPy module using a BLE module.
- As LoPy is a Long Range communication module, it immediately sends this message to the Central Management System (LoPy) using LoRa protocol.
- An immediate action can be taken and the surrounding region can be evacuated.

Hardware:

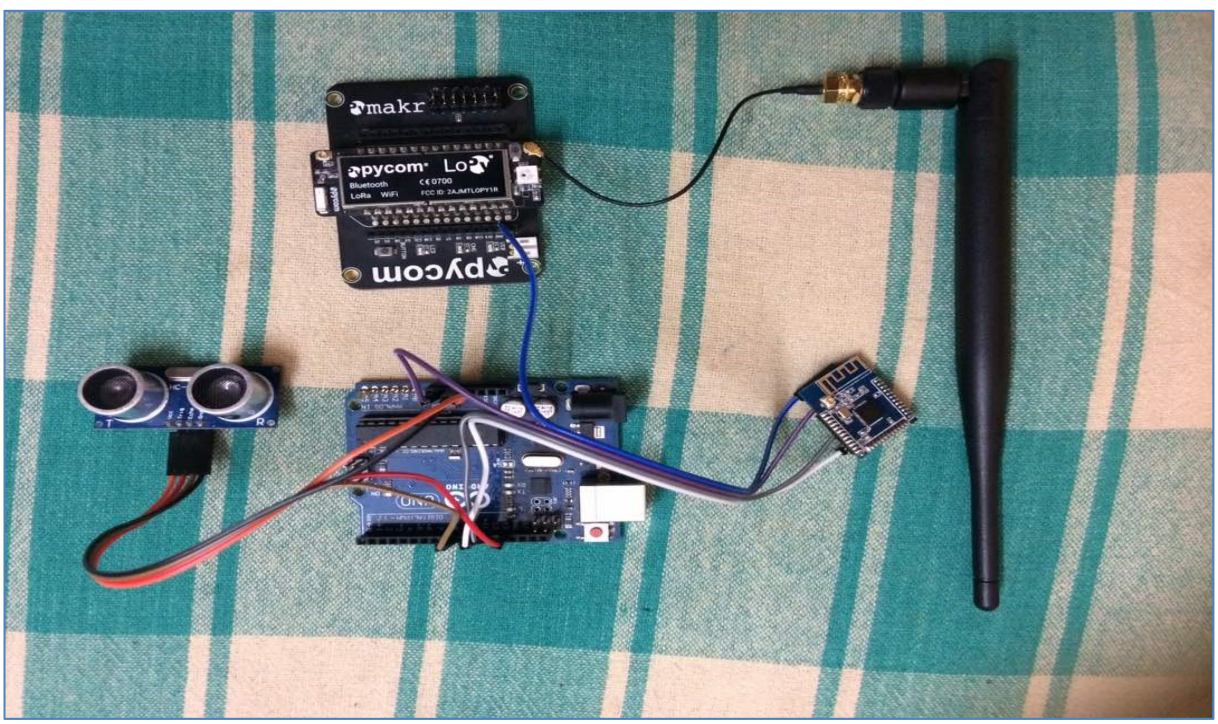


Fig30 hardware solution

Software:

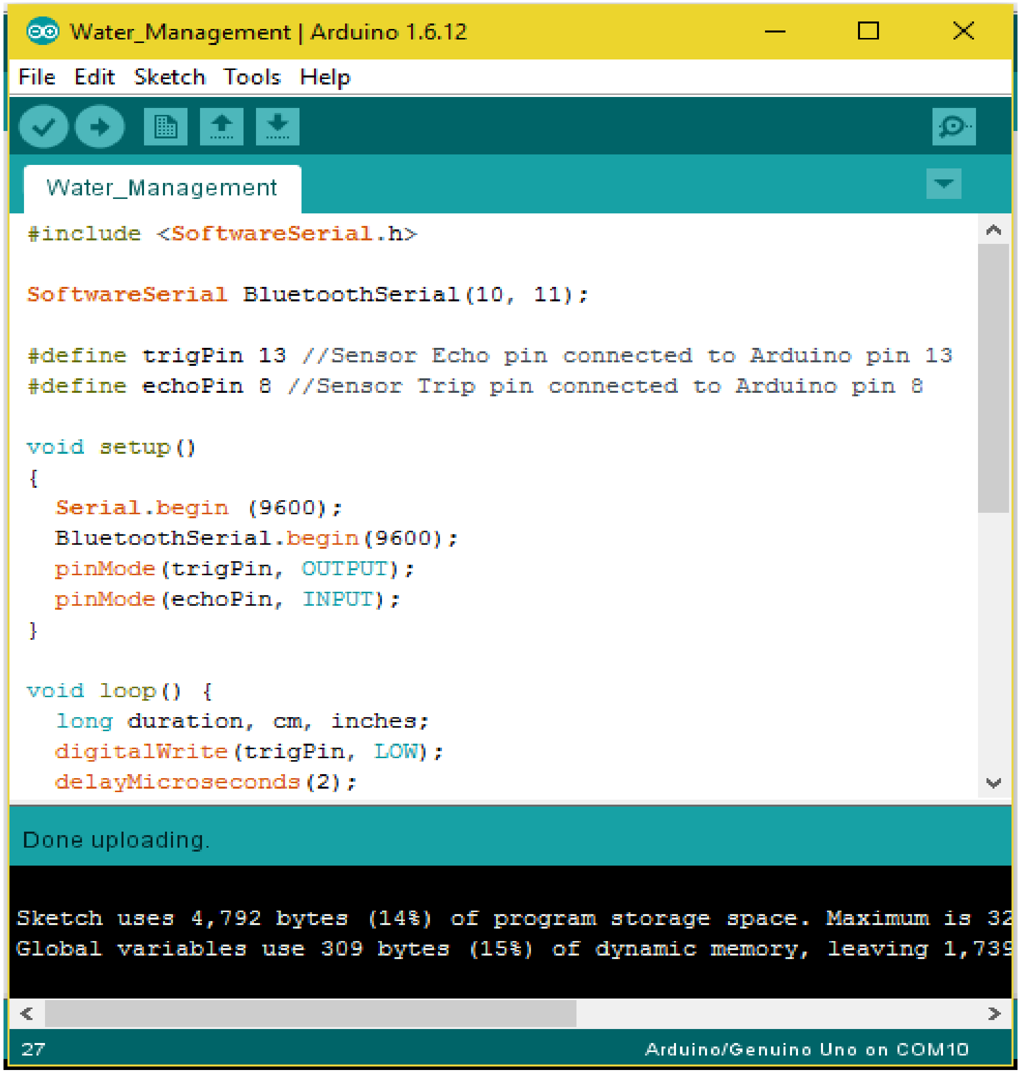


Fig31 a) software solution

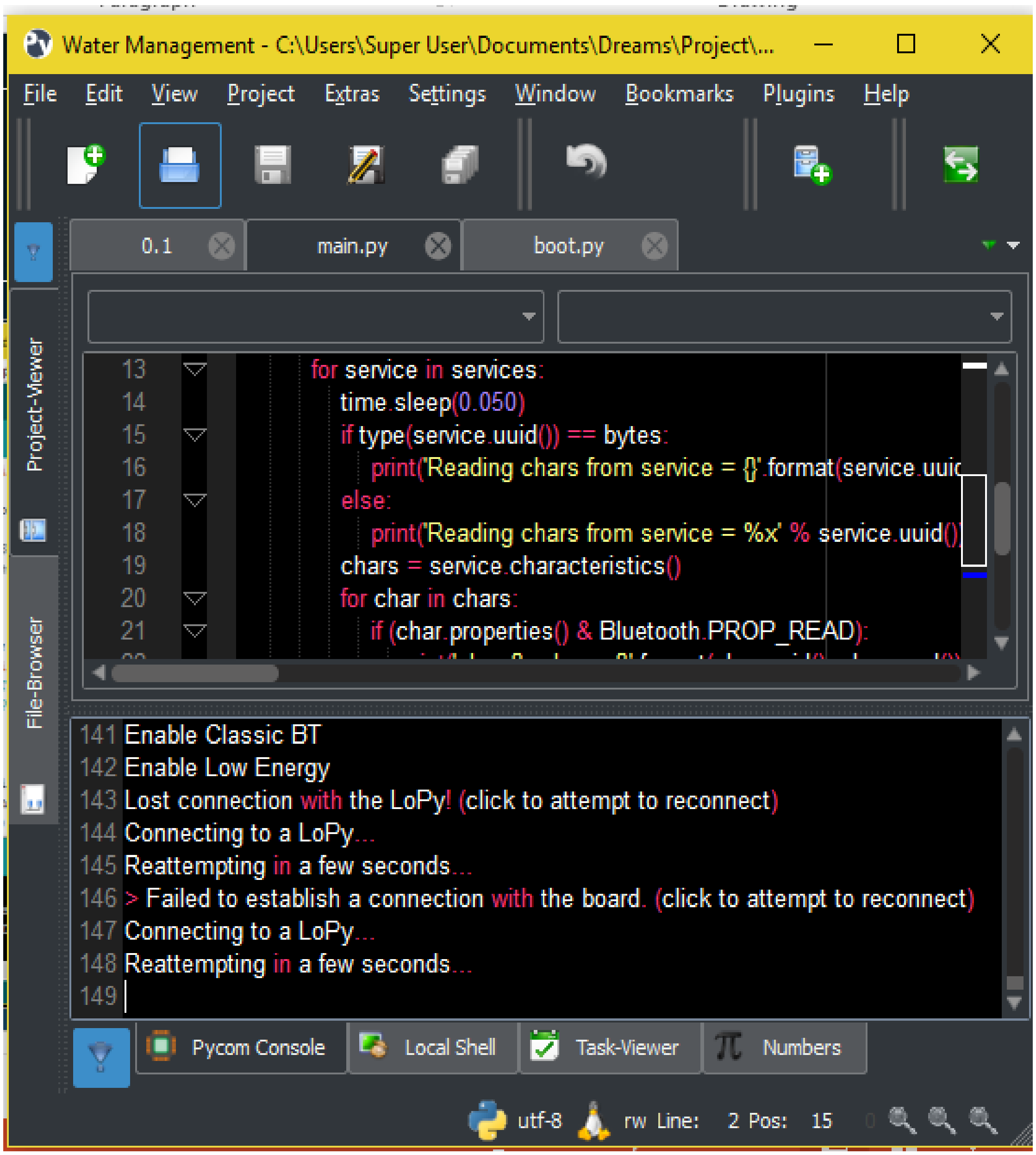
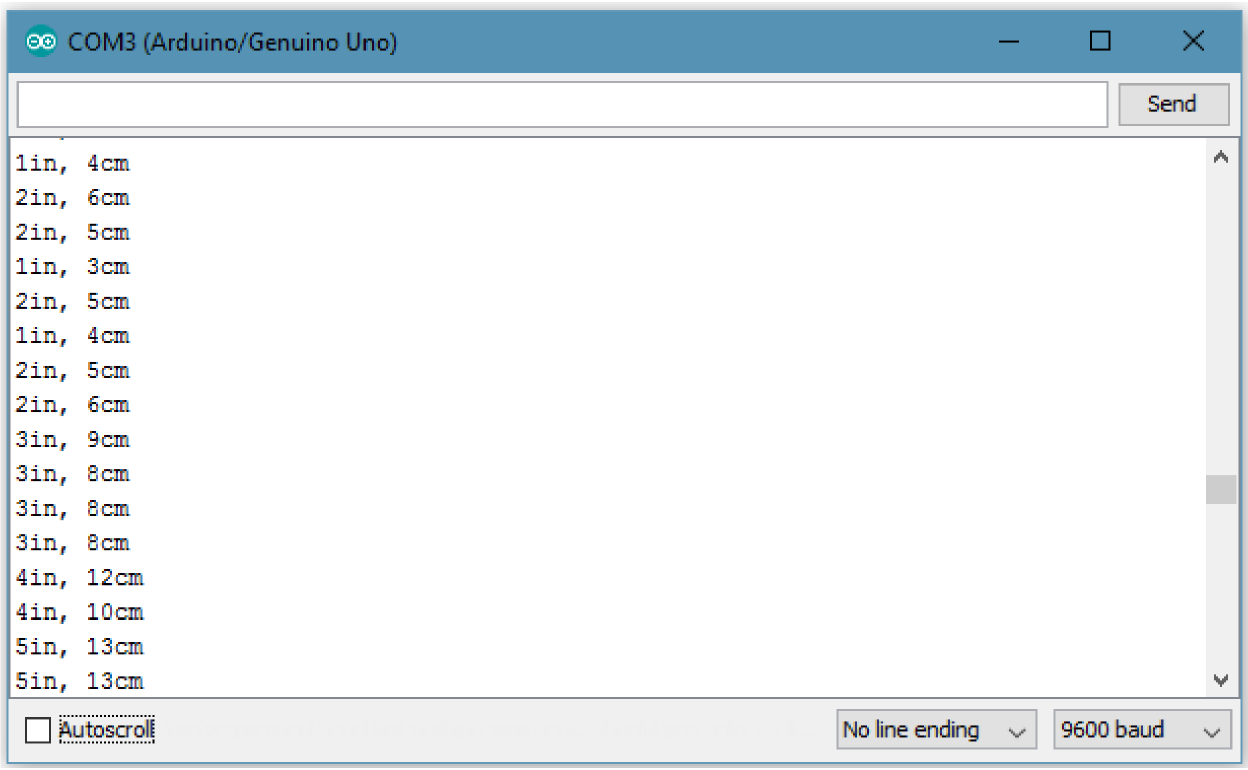


Fig31 b) software solution

Results:



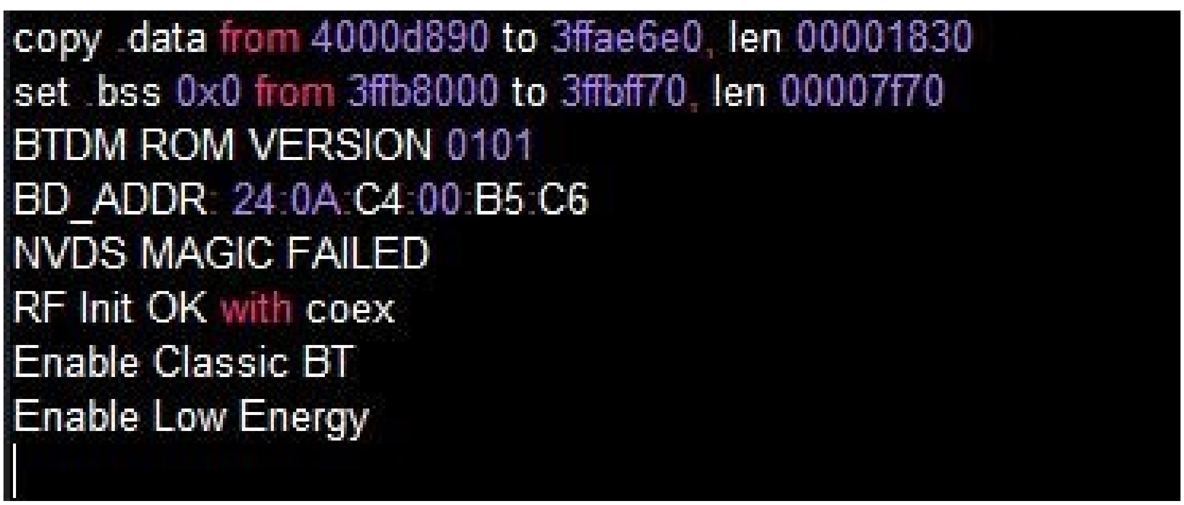


Fig32 a) output

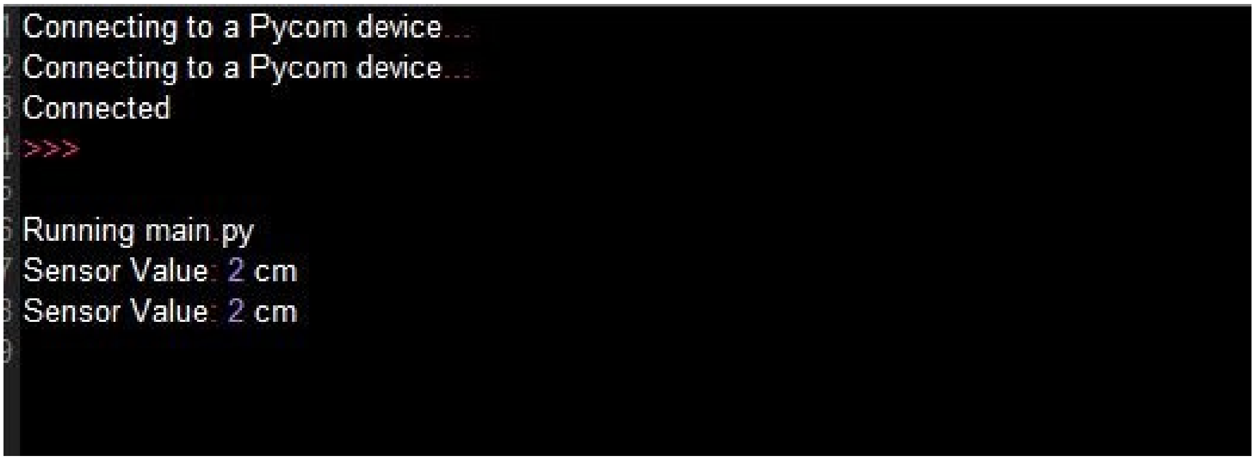


Fig32 b) output

Summary

Our first part of the research included our rationale of work where we looked onto

- Internet of Things.
- Smart Automation Systems.
    - Smart City.
    - Residential Automation.
    - Commercial Automation.
- Performance Enhancement of Physical layer.
    - Arduino.
    - Raspberry Pi.
    - Lora WAN

Here we concluded that **Internet of Things (IoT)** mostly about physical objects and concepts communicating with each other.**Internet of Everything (IoE)** is what brings in network intelligence to bind all these concepts into a cohesive system.

SMART Environment: A physical world that is richly and invisibly interwoven with sensors, actuators, displays, and computational elements.
They are embedded seamlessly in the everyday objects of our lives, and connected through a continuous network.

SMART City: A smart city is an urban development vision to integrate multiple information and communication technology (ICT) and Internet of Things (IoT) solutions.

Commercial Automation : It is the automatic centralized control of building's conditions through a building automation system (BAS). Heating, ventilation, air conditioning and lighting.
The objectives of building automation are improved occupant comfort, efficient operation of building systems, and reduction in energy consumption

We continued our study on the PHYSICAL LAYER.

Physical Layer: Physical layer is the lowest layer of OSI Model. It is responsible for sending bits from one computer to another. This layer deals with the physical connection to the network and with transmission and reception of signals.

Our OBJECTIVE was to

- Performance Analysis of various devices in physical layer by building a testbed of Home Automation system.
- Analysis of different performance parameters of an automation system using various communication protocols.
    - Nrf
    - Bluetooth
    - BLE
    - LoRa
- To design Smart Water Monitoring Solution and Smart Garbage Management using LoRa.

In our METHODOLOGY we

- We designed a testbed of Home Automation for performance analysis of physical layer.
    - Arduino.
    - Raspberry Pi.
- We studied and analysed the different performance parameters related to automation systems.
- Then we designed a testbed prototype using best protocols and devices on basis of analysed results.
- **At last we implemented Smart Water Monitoring Solution using LoRa.**

We studied these performance parameters

- In electrical engineering, **power consumption** often refers to the electrical energy over time supplied to operate an electrical appliance.
- **Network latency is** the time from the source sending a packet to the destination receiving it.
- **Range,** Every sensor is designed to work over a specified range. The design ranges are usually fixed, and if exceeded, result in permanent damage to or the sensor may not respond.
- **Throughput** is the rate of successful message delivery over a communication channel.
- The **frequency** at which a component, circuit, device, piece of equipment, or system operates

We computed results on the following

- We built testbeds of Arduino, Raspberry Pi and LoPy for analysis.
    - Nrf
    - Bluetooth
    - BLE
    - LoRa
        - ➢ Power Consumption.
            - Latency.
            - Range.
            - Throughput.
            - Frequency.
- We built prototypes of Smart Water Monitoring Solution and Smart Garbage Management using LoRa.

On the basis of the results we arrived at the conclusion that the lopy module was the best option for low power long distance communication and in turn we used the lopy module to build our final prototypes of Smart Water Monitoring Solution and Smart Garbage Management using LoRa. The only drawback was the cost of the lopy gateway compared to Bluetooth, ble and nrf modules.

Future Works

- We are going to work on other application scenarios specifically in Intelligent Vehicular Networks.
- LoRaWAN uses the AES 128 algorithm to encrypt messages, which is a non-optimal encryption that can be deciphered by someone willing enough. So, we are going to work on these security aspects.
- We are also working on a research paper based on our proposed solution.

References

- *[1] Jordi Mongay Batalla, George Mastorakis and Constandinos X. Mavromoustakis, "On cohabitating networking technologies with common wireless access for home automation system purposes", IEEE Wireless Communications (Volume: 23, Issue: 5, October 2016).*

- *[2] Sean Dieter Tebje Kelly, Nagendra Kumar Suryadevara and Subhas Chandra Mukhopadhyay, "Towards the Implementation of IoT for Environmental Condition Monitoring in Homes", IEEE Sensors Journal (Volume: 13, Issue: 10, Oct. 2013).*

- *[3] Marco Centenaro, Lorenzo Vangelista and Andrea Zanella, "Long-range communications in unlicensed bands: the rising stars in the IoT and smart city scenarios", IEEE Wireless Communications (Volume: 23, Issue: 5, October 2016).*

- *[4] Pierre Neumann, Julien Montavont and Thomas Noël, "Indoor deployment of low-power wide area networks (LPWAN): A LoRaWAN case study", 2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob).*

- [5] *Jo¨el Toussaint, Nancy El Rachkidy and Alexandre Guitton, "Performance analysis of the on-the-air activation in LoRaWAN", Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2016 IEEE 7th Annual.*