Local Packet Detection in SDR based IoT Gateways

G Prabhu
CS18M004
IIT Madras
cs18m004@smail.iitm.ac.in

M Dileep Kumar
CS18M031
IIT Madras
cs18m031@smail.iitm.ac.in

P Ohm Prakash
CS15B030
IIT Madras
cs15b030@smail.iitm.ac.in

Vedant Somani
CS14B053
IIT Madras
cs14b053@smail.iitm.ac.in

***Abstract*—The aim of the second deliverable is to implement the energy based signal detection program in real-time environment in Raspberry Pi device.**

I. Introduction

The gateway radio receivers using SDR and Raspberry Pi (RPi) are used to receive the signal in real-time and send the signal data to the cloud. The noise part of the signal is discarded in the SDR receiver itself and only the signal part is sent to the network to avoid any congestion in the network. Two methods are used to detect the signals in real-time, one is energy based detection and the other method is correlation based detection.

In energy-based detection, the entire signal stream is monitored against varying power levels. The points where the power is higher, is considered to be signal while the rest is discarded as noise. In the first deliverable, both energy based signal detection and correlation based signal detection were implemented using C programming language and the same was tested using the sampled signal file from the RTL-SDR device. The signal input given to these programs are the sampled signal files (in bin format) collected by the RTL-SDR device connected to a laptop/PC. But the project envisages to create a low cost gateway that can be configured in an RPi device with an RTL-SDR device connected. Hence the program for detecting the energy based detection has to be incorporated in the RPi device to make it function in real-time. The energy based detection program now has to detect the signal in real-time. As the real-time signal detection needs the real-time processing of the received sampled values, the driver program of RTL-SDR is the best location to incorporate the energy based detection program.

Once the energy based signal detection is incorporated into the driver source code of RTL-SDR, then the received signal data has to be sent to the cloud through Ethernet port. Socket programming using UDP protocol has to be implemented alongwith the detection algorithm to send the signal to the cloud. This deliverable envisages the processing of signal received by RTL-SDR in realt-time and sending the detecting signal data to the network/laptop.

II. Second Deliverable

A. Aim

Local detection of the signal in the bit stream as received by the RTL-SDR and sending the signal to the cloud using socket programming.

B. Outcomes Expected

Show the efficiency of the energy based local detection in real time. This can be done with respect to demonstrating the sample loss while implementing the algorithm.

III. Procedure And Steps Followed

A. Preparation

As the first step, the Raspbian Operating system was installed in the RPi device to make it functional and able to work as a gateway device. RTL-SDR device drivers were installed and checked for functioning in the RPi device.

B. Incorporating Energy Based Detection in RTL-SDR

The driver source code of RTL-SDR has been modified in the next phase to incorporate the energy based detection program. For this task, the team has studied the source code of the driver to identify the specific location and variables that are used to write the detected samples into a ".bin" file.

Fig. 1. Raspberry Pi Device

In the driver program, `rtl_sdr.c` is the file used to detect the signal and write it to a local file. In this program, `rtl_sdr_callback()` is the function used to read the detected samples and write it to the local file. Without modifying the original C program, a copy of the same was created in the name of `rtl_sdr_energy.c` for the project. Energy based detection program was implemented as a function in the name of findsignal() in the `rtl_sdr_energy.c` program. Once the signal is detected by the function, the samples are read and copied into a buffer. The sample values can be then written to a file or send to a socket program to send it to the network. In the first deliverable, energy based detection program has detected

starting point of the signal and reported the sample number of the same. In the real-time detection, the program has to send the entire signal data to the network. Accordingly, the data between two preamble has been taken as the signal data and the same has been copied to the buffer for further sending. Appropriate modifications were made in CMakeLists.txt file and Makefile.am files to incorporate the changes in the device driver program thereby avoiding any errors while compiling the actual program.

C. Socket Programming

RPi is a memory constrained device and the programs running in it has to be adequately optimized to get a desired output. The socket program written and tested for sending data from one computer to another has to be modified as per the requirement of RPi. Unlike other devices, where the data generated by the system is occasional, the real-time signal receiver generates voluminous data to be sent through the socket. This leads to loss in the signal data during transmission. The loss of signal while transmitting is due to the reason that the socket busy in sending the previous data when the current signal is ready to be sent. Also there is a limitation in the size of data that can be transmitted through socket at a given point of time.

A separate function was written to send the data from RPi device to the cloud. The each received signal samples data was more than 100k bytes, which was leading to loss of samples while transmission through the socket. So the "send" function was modified to split the data into chunks of 50k each and send it to the socket. The RTL-SDR was connected to RPi device which was further connected to the laptop/cloud on Ethernet port. The signal transmitted by LoRa device was sampled and sent by the RPi to the laptop where the received signals from socket are saved in separate signal file in CSV format. A separate signal file is being created for each signal data received.

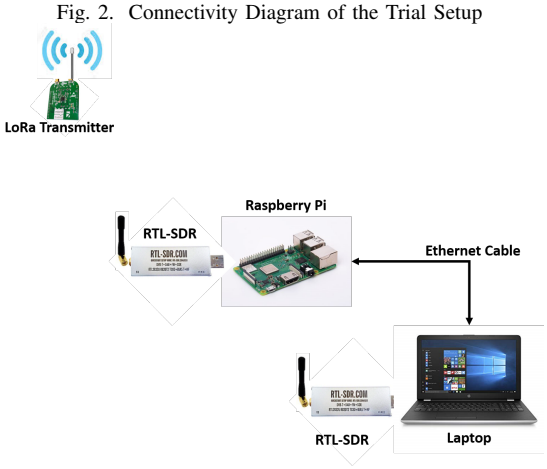
IV. Trials and Observations

A. Efficiency of Energy Based Detection in Real-Time

The efficiency of the energy based detection in real-time was tested using an extra RTL-SDR device which was kept close to the RTL-SDR connected to the RPi device. RPi was taken control using the SSH connection from the laptop and the real-time energy based detection program was run in RPi to detect the signals. The RTL-SDR connected to the laptop was used to receive the signals into a bin file and then detect the signals from the bin file using energy based detection in non-realtime. Both the realtime and non-realtime programs were controlled from the laptop in a time synchronized manner by using "Terminator" software. This setup was done to enable verification of the signal being detected in real-time energy based detection algorithm. Two receivers are kept closeby to avoid any variation in reception due to multi-path propagation.

The block diagram of the trial setup is given below:-

Fig. 2. Connectivity Diagram of the Trial Setup

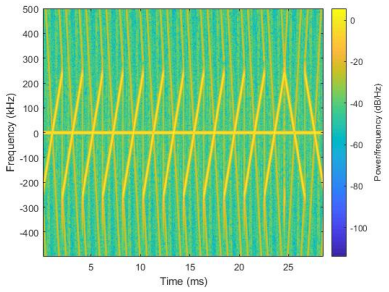


The number of signals detected by real-time and non-real-time programs were checked by changing the following parameters:-

- Varying the interval between signals being transmitted by the LoRa device.
- Increasing the running time of the programs.

Screenshot of the preamble of the received signal plotted in MATLAB using spectogram is given below:-

Fig. 3. Preamble of the Received signal



The total number of signals detected by real-time and non-real-time programs were plotted against the parameters like time interval between the signal transmission and the duration of running the program. The same is produced below for reference.

Fig. 4. Signals detected by varying the time interval between signals

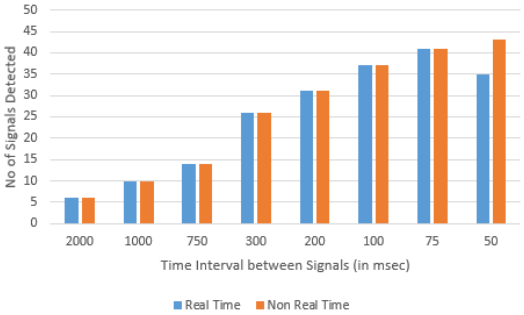
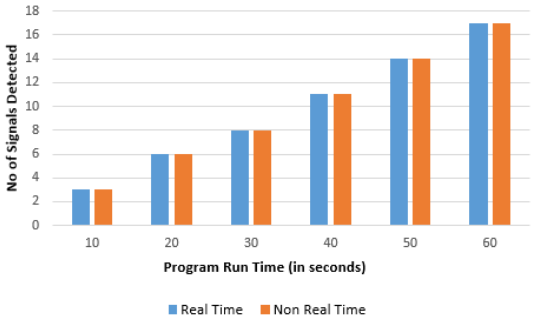


Fig. 5. Signals detected by increasing the duration of the program running time



B. Observations

1. When the time interval between the signals transmitted by the LoRa device was reduced below 50ms, the realtime program was giving packet losses. This packet loss is due to the large amount of data that is being sent to the socket for sending it to the cloud. In lesser time intervals, the frequency of signals received by RTL-SDR increases thereby leading to the increase in number of signals detected per second.

2. The program was run for prolonged duration and under suitable time intervals of signals there was no loss of packets observed.

ACKNOWLEDGMENT

We thank Revathy Narayanan for being a constant help and mentor through the course of this work.

REFERENCES

[1] *Installing Raspbian Operating System Images* https://www.raspberrypi.org/documentation/installation/installing-images/

[2] *Raspberry Pi Basics: installing Raspbian and getting it up and running* https://www.howtoforge.com/tutorial/howto-install-raspbian-on-raspberry-pi/