SYSC 3010

TECHNICAL MEMO

**Subject:** RF communication between two Raspberry Pis

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Executive Summary**

The raspberry pi is an affordable fully functional computer on a smaller scale that runs on the Linux OS. It is easily configurable and is a great way to see programming in action. The raspberry pi has several GPIO pins where external components can be attached to perform certain tasks. It can also easily transmit data through the internet. This is possible through the Wi-Fi module installed on modern Raspberry pis. However, it is possible in a certain project that data needs to be transmitted to other devices in an area that doesn’t support internet accessibility. To overcome this problem, one of the possible methods to transmit data is the use of an RF transmitter and RF receiver. This memo will cover the RF communication between two raspberry pis. A 433mhz transmitter and receiver pair will be used

**Method**

RF communication is a low-cost method to transmit data in scenarios where Wi-Fi is not able to. It is the transferring of information between two or more components, hence the transmitter and receiver. The transmitter takes the initial data and modifies that signal using a modulation technique to covert it into a signal which an antenna transmits. The transmitter in this scenario will be on one of the RPIS. The other RPI will have an RF receiver attached to it. The receiver takes the transmitted signal and decodes it to the original data which will be processed by the RPI for whatever purpose. A 433mhz transmitter and receiver pair will be used for this memo.

**Procedure**

To achieve RF communication, three main steps need to be performed:

1. Setting up the necessary software
2. Install WiringPi which is needed to control the pins on the Raspberry Pi. Run the following commands in the terminal:

cd ~/

git clone git://git.drogon.net/wiringPi

cd wiringPi

./build

Executing the last command should have WiringPi installed. To test it, run:

$ gpio readall

Running the command will give you a complete picture of your Pi’s GPIO connector(s) with all the numbering schemes present.

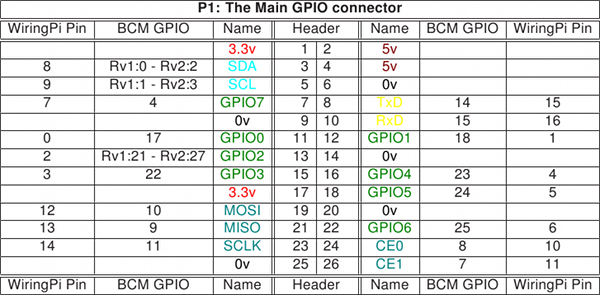


Figure 1: Image displayed in terminal for WiringPi test

1. Install 433Utils which is a library containing encoding and decoding functions necessary for 433mhz rf transmission. To install 433Utils, run the following commands:

cd ~/

git clone git://github.com/ninjablocks/433Utils.git

cd 433Utils/RPi\_utils

make

Executing the last command should have 433Utils installed

1. Setting up the Receiver and transmitter
2. Each GPIO pin can be used to wire the receiver. It is important that you do not connect the pin marked ‘3v3’ to the 5v pin on the raspberry pi as doing so will damage the rpi. Connect the DATA pin to whichever GPIO you choose. Refer to figure 1 for any connections.

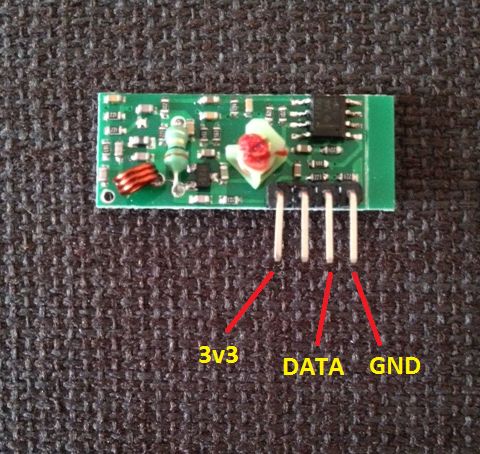


Figure 2: The Receiver module

1. For the transmitter, it is alright to power the it from the 5v pin on the RPI since the data pin is not sending signals to the RPI. An antenna is a useful addition since it increases the range for the transmitter.

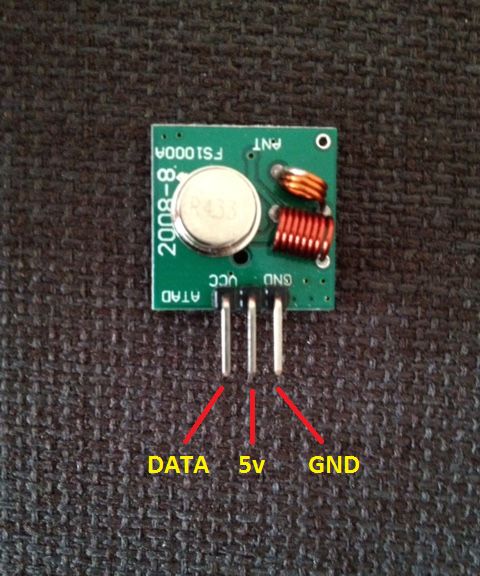


Figure 3: The transmitter module

A reminder that each RPI will have either a transmitter or a receiver. NOT BOTH! For clarity purposes, the image below shows how each RPI will connect to either module. Using a breadboard is recommended for ease of connections.

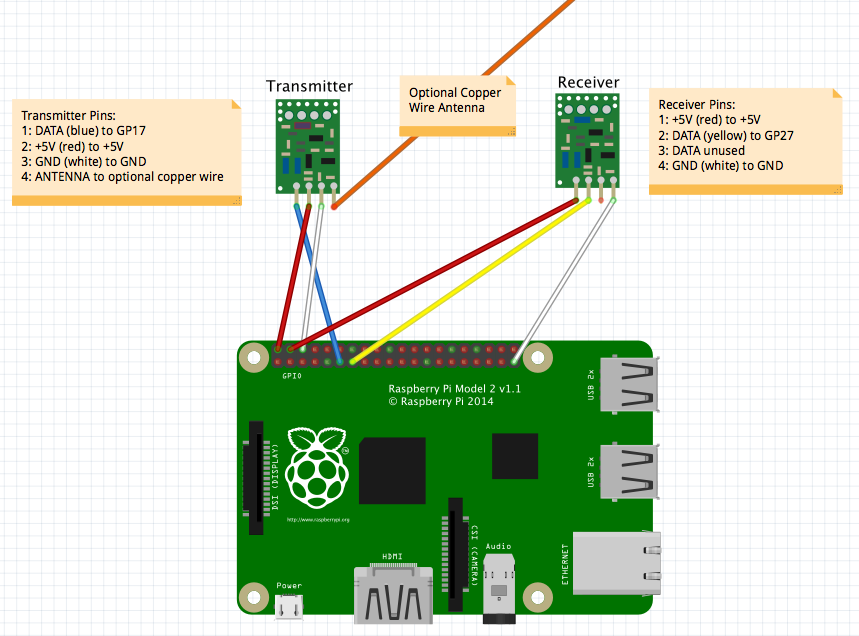


Figure 4: Complete circuit for Transmitter or receiver

1. Running the code

This is the section where we use the library we installed earlier.

1. For the RPI that has the receiver, to start listening for the 433mhz codes start the RFSniffer program from the 433Utils library using the command below:

~/433Utils/RPi\_utils/ sudo.RFSniffer

This will make the receiver wait till data starts transmitting. Once data is received the program will print the data to the terminal.

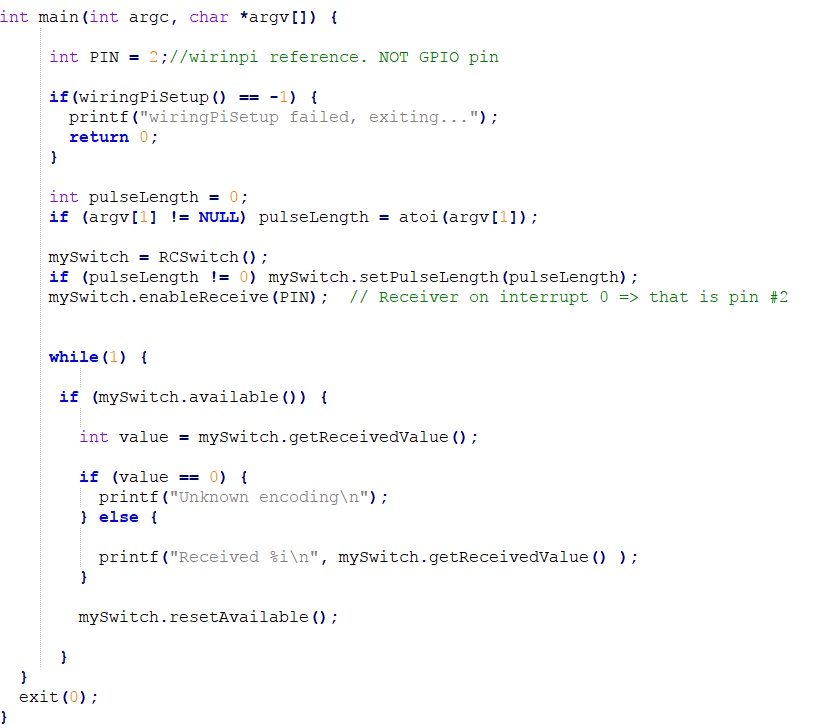


Figure 5: RFSniffer code

1. For the RPI that has the transmitter, to start listening for the 433mhz codes start the codesend program from the 433Utils library using the command below:

~/433Utils/RPi\_utils/ sudo.codesend xxxx

The xxxx represents the arguments that will be sent to the raspberry pi with the receiver. Once data has been sent and received, the output should be:

Received xxxx

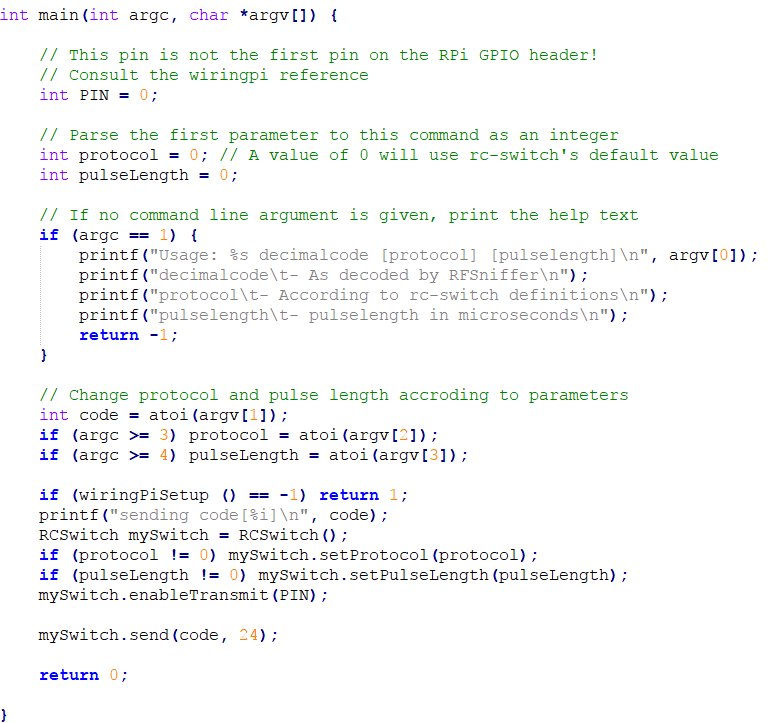


Figure 6: codesend code

The amount of data that can be sent is limited to eight characters at time. It is up to the user to overcome this challenge via methods such as having the first character represent the type of data being sent.

**Conclusion**

To summarize, RF communication is vital in areas that do not provide other means of communication such as Wi-Fi. It is easy to setup and while providing little difficulty in use. The above instructions can be combined with scripts or projects to create automated sending of messages.

**References**

1. <http://www.instructables.com/id/Super-Simple-Raspberry-Pi-433MHz-Home-Automation/>
2. <https://www.princetronics.com/how-to-read-433-mhz-codes-w-raspberry-pi-433-mhz-receiver/>
3. <https://boundless.aerohive.com/technology/wi-fi-back-to-basics-radio-frequency-rf-components.html>
4. <http://wiringpi.com/pins/>