**#include <SPI.h> //Import SPI library**

**#include <RH\_RF95.h> // RF95 from RadioHead Library**

**#define RFM95\_CS 10 //CS if Lora connected to pin 10**

**#define RFM95\_RST 9 //RST of Lora connected to pin 9**

**#define RFM95\_INT 2 //INT of Lora connected to pin 2**

**// Change to 434.0 or other frequency, must match RX's freq!**

**#define RF95\_FREQ 434.0**

**// Singleton instance of the radio driver**

**RH\_RF95 rf95(RFM95\_CS, RFM95\_INT);**

Inside the*setup*function we will reset the **LoRa module by pulling its reset pin to low** for 10 milli second to start fresh. Then we **initialize it with the module** that we created earlier using Radio head library. Then, we **set the frequency and transmission power for the LoRa server**. Higher the transmission more distance your packets will travel but will consume more power.

**void setup()**

**{**

**Serial.begin(9600);**

**// Reset LoRa Module**

**pinMode(RFM95\_RST, OUTPUT);**

**digitalWrite(RFM95\_RST, LOW);**

**delay(10);**

**digitalWrite(RFM95\_RST, HIGH);**

**delay(10);**

**//Initialize LoRa Module**

**while (!rf95.init()) {**

**Serial.println("LoRa radio init failed");**

**while (1);**

**}**

**//Set the default frequency 434.0MHz**

**if (!rf95.setFrequency(RF95\_FREQ)) {**

**Serial.println("setFrequency failed");**

**while (1);**

**}**

**rf95.setTxPower(18); //Transmission power of the Lora Module**

**}**

**Inside the infinite *loop* function, we simply have to send the data packet through the LoRa module.**This data can be anything like sensor value of user command. But for simplicity we will send char value 0 to 9 for every 1 second interval and then initialize the value back to 0 after reaching 9. Note that the values can be sent only in a char array format and the type of data should be unit8\_t that is 1 byte at a time. The code to do the same is shown below

**void loop()**

**{**

**Serial.print("Send: ");**

**char radiopacket[1] = char(value)};**

**rf95.send((uint8\_t \*)radiopacket, 1);**

**delay(1000);**

**value++;**

**if (value > '9')**

**value = 48;**

**}**