

Part 2 - Hands On

4. Practicum: LoRa Implementation on ESP32

A. Hardware Setup (Wiring)

Connect the SX1276/RFM95 module to the ESP32 using the standard SPI configuration:

LoRa Pin	ESP32 GPIO	Function
GND	GND	Ground
3.3V	3.3V	Power (Do not use 5V)
MISO	GPIO 19	SPI Master In Slave Out
MOSI	GPIO 23	SPI Master Out Slave In
SCK	GPIO 18	SPI Clock
NSS/CS	GPIO 5	Chip Select
RST	GPIO 14	Reset
DIO0	GPIO 26	Interrupt (IRQ)

B. Software Preparation

1. Install the "LoRa by Sandeep Mistry" library via the Arduino Library Manager.
2. Ensure the frequency (`433E6` , `868E6` , or `915E6`) matches your hardware module.

C. Experiment 1: Basic Sender & Receiver

Sender Code:

```
#include <SPI.h>
#include <LoRa.h>

// Pin Definitions
#define SS_PIN 5
```

```
#define RST_PIN 14
#define DI00_PIN 26
#define BAND 923E6

int counter = 0;

void setup() {
  Serial.begin(115200);
  while (!Serial);
  Serial.println("LoRa Sender");
  LoRa.setPins(SS_PIN, RST_PIN, DI00_PIN);

  if (!LoRa.begin(BAND)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
}

void loop() {
  Serial.print("Sending packet: ");
  Serial.println(counter);

  LoRa.beginPacket();
  LoRa.print("hello ");
  LoRa.print(counter);
  LoRa.endPacket();

  counter++;
  delay(5000);
}
```

Revision #4

Created 2025-11-22 13:47:03 UTC by ED

Updated 2025-11-23 15:41:00 UTC by ED