Grove - 433MHz Simple RF Link Kit

This kit is used for one way wireless communication at a frequency of 433MHz and includes a transmitter module and a receiver module. The twig configuration of this kit allows for around 40 meters of transmitting distance indoors, or around 100 meters outside.

Features

- GROVE compatible interface.
- Uses ASK (Amplitude Shift Keying) Modulation.
- One way communication.

Application Ideas

- Remote control
- Remote automation
- Alarm

Mechanic Dimensions

- Receiver: 24mm by 42mm
- Transmitter: 20mm by 24mm

Usage

The transmitter and receiver modules both rely on a single wire for communication. Though using the UART supplied by the arduino platform can work, it is recommended, instead, to use the VirtualWire library which uses Amplitude Shift Keying for modulation which provides better communication.

Both the transmitter and receiver modules require three wires: Vcc, Ground, and signal. Pin 2 of both parts of the kit are not connected.

Hardware Installation

Connect the Transmitter module to Digital I/O 2 of the [Stem_-_Base_Shield](#) on the arduino being used for transmission.

Connect the Transmitter module to Digital I/O 2 of the [Stem_-_Base_Shield](#) on the receiving arduino.

Programming

Sample code for Transmitter:

```
#include <VirtualWire.h>

int RF_TX_PIN = 2;

void setup()
{
  vw_set_tx_pin(RF_TX_PIN); // Setup transmit pin
  vw_setup(2000); // Transmission speed in bits per second.
}

void loop()
{
  const char *msg = "hello";
  vw_send((uint8_t *)msg, strlen(msg));  // Send 'hello' every 400ms.
  delay(400);

}
```

Sample code for Receiver:

```
#include <VirtualWire.h>

int RF_RX_PIN = 2;

void setup()
{
  Serial.begin(9600);
  Serial.println("setup");
  vw_set_rx_pin(RF_RX_PIN);  // Setup receive pin.
  vw_setup(2000); // Transmission speed in bits per second.
  vw_rx_start(); // Start the PLL receiver.
}

void loop()
{
  uint8_t buf[VW_MAX_MESSAGE_LEN];
  uint8_t buflen = VW_MAX_MESSAGE_LEN;
  if(vw_get_message(buf, &buflen)) // non-blocking I/O
  {
    int i;
    // Message with a good checksum received, dump HEX
    Serial.print("Got: ");
    for(i = 0; i < buflen; ++i)
    {
      Serial.print(buf[i], HEX);
      Serial.print(" ");
    }
```

```
    Serial.println("");
  }
}
```