

Encrypted Walkie-Talkies

Description

The goal of the project is to build a pair of walkie-talkies that have a digitized and encrypted audio stream being sent wirelessly and in real time between the two. Both devices should be able to transmit and receive the correct audio, while other devices without the encryption key will not be able to.

Hardware

[Audio Module-Teensy Audio BD](#)

[RF Module-RFM69](#)

MSP430 - Interface with all other modules and responsible for encryption

Speaker - Play the received message

Microphone - Used to input audio stream

Pushbutton - For triggering the interrupt which start transmission

Antenna - Transmit and receive audio stream

LCD Display -

Enclosure

Device Features and Flow

Transmit

The user will press a button to trigger an interrupt that will open an audio streaming session to the other walkie-talkie. Then the user's voice will be sampled through a microphone into the MSP430FR6989's ADC using 8-bit samples at 8kHz. These samples will be pushed into the AES accelerator onboard the MSP430, and then sent to the RF module via SPI commands. The RF module will be set up to transmit data via FSK using a data rate sufficient to send the encrypted audio in real time.

Receive

The receiving platform will awake from sleep on an interrupt from the RF module when it detects an incoming preamble. The data stream from the RF module will be pulled to the MSP430, decrypted using the AES accelerator, and then send it to the audio module with I2C commands. That module will be configured to receive the unencrypted audio stream, and output it to an attached speaker.

MSP430 Features

The project will incorporate the ADC for microphone input, the AES hardware acceleration for encryption, interrupts for triggering user input, as well as SPI and I2C buses to send and receive data. Additional features can be added if the time permits.

Project Division

Wireless Communications/RF Module

Dylan

Audio Module Configuration and Communications

Tony

LCD display, Push-button

Vishal

ADC, AES, and State Machine
Juan