Digital Text Communication System Using On-Off Keying with Pixie QRP Transceivers and Raspberry Pi

Version 1.0(3/4/26): updates will be forthcoming….

---

1. Project Objective

The objective of this project is to design, implement, and experimentally evaluate a digital text communication system using two Pixie QRP CW transceivers and two Raspberry Pi single-board computers. The system will transmit binary data using direct **On-Off Keying (OOK)**, where the Raspberry Pi keys the transmitter to represent digital information. The Pixie is a very low cost kit that can be found for 5-15CAD e.g. search amazon or aliexpress for pixie transceiver! At the end of the project you will be able to send text between two Rasberry Pis using a wire antenna over distances of 10-15km i.e. you can communicate with your friends within Ottawa and beyond without using internet or cell phones!

2. System Overview

Two identical stations will communicate over an HF amateur band. Each station consists of:

- Pixie QRP CW transceiver
- Raspberry Pi
- Opto-isolated GPIO keying interface
- USB audio interface (receive path)
- Terminal-based user interface

Binary data will be transmitted by switching the RF carrier on and off, corresponding to logical “1” and “0”.

3. System Block Diagram Description

Transmitter Chain

User Interface → Encoder → Framing & CRC → Bit Timing Control → GPIO Driver → Isolation Circuit → Pixie Transmitter → Antenna

Block Descriptions:

1. **User Interface**
   A terminal program running on the Raspberry Pi accepts text input from the keyboard.
2. **Encoder**
   Converts ASCII characters into binary representation.

3. **Framing & CRC Module**
   Adds:
   - Preamble for synchronization
   - Start delimiter
   - Payload length
   - CRC checksum
   - End marker
4. **Bit Timing Control**
   Generates fixed-duration bit intervals (e.g., 33–100 ms per bit). Ensures consistent symbol timing.
5. **GPIO Driver**
   Outputs digital HIGH/LOW signals corresponding to binary 1 and 0.
6. **Isolation Circuit (Optocoupler)**
   Electrically isolates the Raspberry Pi from the Pixie keying circuit to prevent damage and ground loops.
7. **Pixie Transmitter**
   Produces RF carrier when keyed (binary "1") and no carrier when unkeyed (binary "0").
8. **Antenna**
   Radiates the OOK-modulated RF signal.

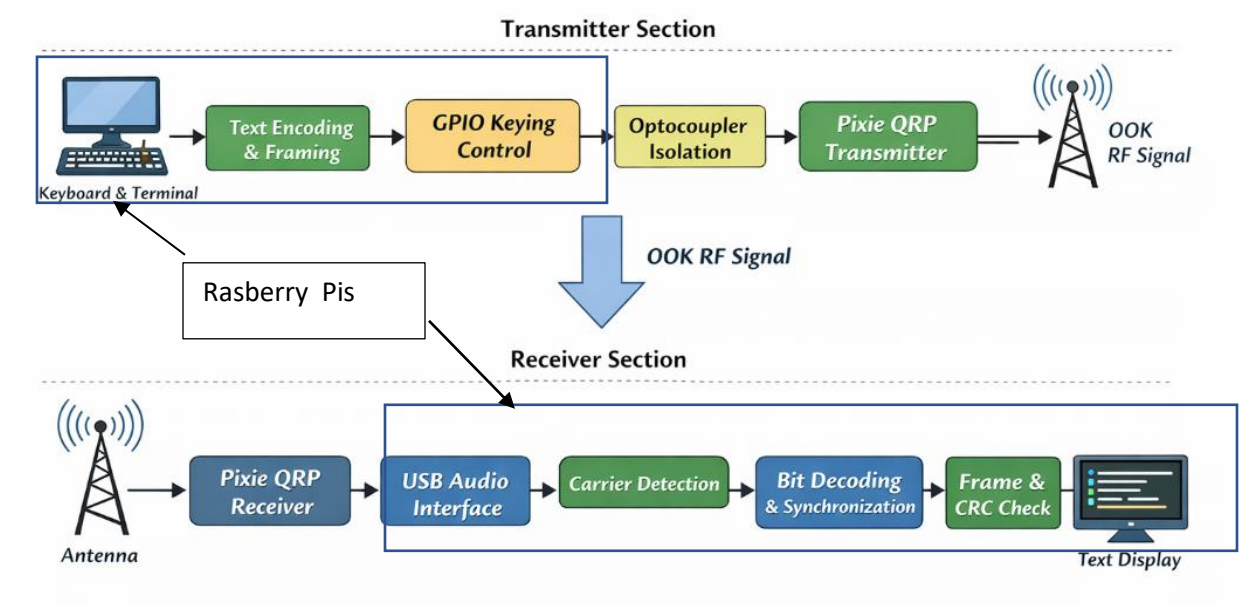
Receiver Chain

Antenna → Pixie Receiver → Audio Output → USB Sound Interface → Carrier Detection → Bit Synchronization → Frame Decoder → Text Display

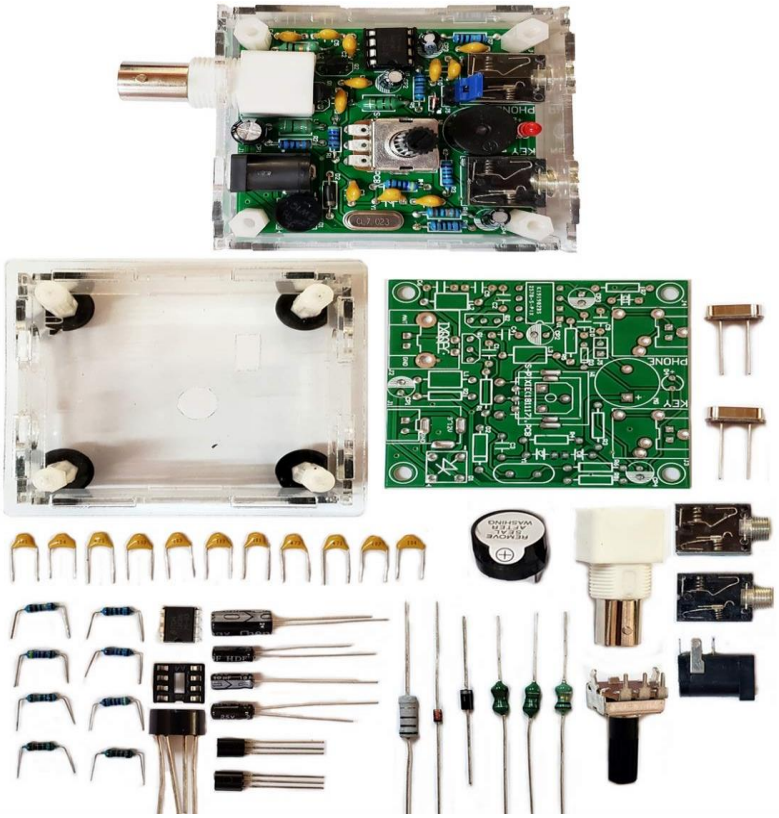
Block Descriptions:

1. **Antenna**
   Receives incoming RF signal.
2. **Pixie Receiver**
   Demodulates RF into an audio tone corresponding to carrier presence.
3. **Audio Output**
   Provides baseband audio signal to Raspberry Pi.
4. **USB Sound Interface**
   Converts analog audio to digital samples.
5. **Carrier Detection Module**
   Performs envelope detection and amplitude thresholding to determine:
   - Carrier present → binary "1"
   - Carrier absent → binary "0"
6. **Bit Synchronization**
   Recovers timing using the preamble pattern to align sampling with transmitted bit intervals.
7. **Frame Decoder & CRC Check**
   Extracts payload and verifies data integrity.

8. **Text Display Module**
   Reconstructs ASCII characters and displays received message.



9. Here is how the Pixie transceiver kit looks like. It costs between 6-12CAD depending on where you order from!.



In this project you will learn how to build an electronic kit from scratch, analyse its operation, develop the necessary software to interface a Rasberry Pi and understand the basics of digital communications not to mention the fun to use the system to communicate with your friends all over Ottawa (with a good antenna)!