# Software

Since BlackChat uses a serial interface client to connect to a computer, the only custom software is that operating on the MSP430 microcontroller in the device. This firmware handles formatting messages to be sent through the transmitter, interpreting messages detected by the receiver, as well as interfacing with the user to get messages to transmit and displaying messages that have been received.

## VirtualWire

The firmware uses an MSP430 ported Arduino library written specifically to operate with the type of transceivers used in BlackChat. The super regenerative receiver performs best in a steady state situation where the received signal consistently changes between 0 and 1, so that the DC component is around half of the operating voltage. Outside of this condition, the automatic gain controller does not function properly and signals will not be demodulated properly.

To circumvent this caveat of the receiver design, VirtualWire transmits messages with a fixed stream of 0-1 training pulses to approximate the required steady state condition. Once these pulses are transmitted, a start code is sent, follow by a message length, and finally a 16 bit cyclic redundancy check.

On the receiving end, VirtualWire polls the output of the receiver 8 times per bit period, and uses a majority vote to determine its value. Since the transmitter and receiver are on different clock domains, some type of synchronization must be done so ensure that the 8 samples are taken from the same bit period. Do accomplish this, VirtualWire uses a software PLL to continuously adjust the sample frame to a proper location. If the receiver sees a transition sooner than half way through its current frame, the frame is moved back, and if a transition is detected later than half way through its current frame, the frame is moved forward. Continuously following this algorithm allows the receiver to align with the incoming bit period, and also allows for correcting minor timing differences between the transmitter and receiver.

## User interface