A Wireless Communication using Bubls

Ashwini.B.Halakerimath
Department of Computer Science and Engineering
AGMRCET-Varur, Hubli, Karnataka, India

Sneha.Vasudev. Dhage
Department of Computer Science and Engineering
AGMRCET-Varur, Hubli, Karnataka, India

Abstract-This article presents a secure wireless communication using Morse algorithm explicitly the setup includes system connected with Relay/Optocoupler. Morse code algorithm is used to convert simple sentences to dash (“-“) and dots (" . "). The setup includes bulb whose intensity varies based on input character. And webcam facing opposite side bulb. When the character is typed Morse code generator encodes it and decodes at opposite side. So as bulb blink, webcam detects it and through simple threshold based video processing and the interface translates the blinks back to sentences. The same arrangement is placed at the other side. Communication is possible only between authenticated users. This sort of communication can’t be decoded, no airtel or Google can store a copy of it so great for security.

Keywords – Morse code encoding, decoding.

I. INTRODUCTION

Wireless communication is a transfer of information between two points that are not connected by electrical conductors. It allows a secure means of communication only between authenticated users. Image and the database of the authenticated users are stored on both systems, when an user tries to communicate the identity of the user is compared with already existing database, if matches with existing database of the user then it allows for communication. If authentication process fails communication cannot proceed. This technique uses Morse code algorithm for data transfer which is used in Bluetooth and Wi-Fi and this type of communication can be used in areas where there is lack of network. When user enters message, message is converted to dash and dots by Morse code generator using Morse code algorithm. The encoded data the bulb changes its intensity for every input character, the video of change in intensity of bulb for the message is captured by webcam on the other side, the recorded video is passed to Morse code generator and generator decodes the message and converts dash and dots back to sentences. When message is delivered, system alerts with beep sound indicating message has reached successfully. The same process is carried out on the other side.
II. MODULE DESCRIPTION

A. WEB-
A simple Webcam setup consists of digital camera attached to computer, typically through USB port. The camera part of the web cam set up is just a digital camera. Webcam software “grabs a frame” from the digital camera at a pre-set interval. And transfers it to another location for viewing. Webcam system with a high frame rate is used for streaming video. Frame rate indicates the number of pictures the software can grab and transfer in one second. For streaming video, a minimum rate of at least 15 frames per second are needed.

Whenever the data is transferred, it is usually done over the network, which uses the air medium. But here, the bulb is used as media to exchange the data in the forms of dash and dots. So the data sent is over, gets converted to signals in the form of energy which make the bulb to glow. When bulb glows, the sensors sends the signals to take image of it. The image captured is processed through an algorithm which decodes the data and converts back to message which was entered as input. This sort of communication increases the accuracy of data sent over the network.

B. OPTO-COUPLER-
Opto-couplers have existed in various forms since the late 1960s. Optocouplers use light to carry information through an isolation barrier. Input signals modulate the output intensity of a light-emitting diode. A photodiode responds to the optical signal by switching an output transistor on and off.

C. BULB-
The Morse Code Torch (light) is used to communicate secret messages just like real spies do. This light is used to send messages to someone who can then decipher them using the included code chart. The torch doubles as a cool key ring. The included shutter card is used to imitate the shutters on a navy signal lamp and can broadcast coded messages to anyone nearby.

III. PROPOSED ALGORITHM

A. MORSE CODE ALGORITHM

If we receive stream of bits like 1011101 we cannot yet decide that this is actually an 'R': it could be that the next two bits would be 01, making the character an 'L'. This is solved easily by considering the inter-character space as part of the character itself. So we say that 'R' is 1011101000, while 'L' is 101110101000.

The code that Morse developed for use with his system went through a few transformations before arriving at the code we’re familiar with today. Initially, Morse code only transmitted numbers. The transmission’s receiver would then have to use a dictionary to translate the numbers into words. But that proved to be tedious. Soon the code was expanded to include letters and even punctuation.

Morse code became extremely important in maritime shipping and aviation. Pilots were required to know how to communicate using Morse code up until the 1990s. Today Morse code is primarily used among amateur radio users. In fact, up until 2007.

The timing specification for Morse code:

- A dash is three times as long as a dot.
- The space between dots/dashes within one character is equally long as a dot.
- The space between two characters is three times as long as a dot.

The space between two words is seven times as long as a dot. So the duration of a dot is the obvious unit of time:
Consistently, we can express the space between words as the bits 0000. For every dot, we have the sequence 10; for every dash, 1110; and at the end, we attach 00 (which, together with the 0 at the end of the last dot or dash, completes the group of three 0s that is the inter-character spacing). Consequently, every character is represented by an even number of bits. Furthermore if we consider only the sequences 10, 11 and 00 occurs; the sequence 01 never occurs! So if we number the bits starting from one, the odd-number bits are more likely to be 1 than the even-numbered bits.

Decoding of Morse signals-the signal path

This consists of the following steps:

1. Multiply the incoming signal with a locally generated carrier at the same frequency as the incoming signal. This mixes the incoming signal to 0 Hz (or nearly 0 Hz, if the local carrier is not exactly at the right frequency). In order not to lose information, two signal paths are created, one after mixing with the carrier itself, and one with the same carrier shifted in phase by 90 degrees. The paths are customarily called I (in-phase) and Q (quadrature).

2. Low-pass filters both I and Q signals. Note that, due to the mixing done in the previous stage, this corresponds to passing only signals which were originally near the carrier frequency: it acts as a band-pass filter. In RSCW, this filter is just a moving average over 48 samples, but it could be replaced by a steeper filter is just a moving average over 48 samples, but it could be replaced by a steeper filter to attenuate signals away from the frequency of interest more. Furthermore, in RSCW this step also does a down sampling: 8000 samples per second enter the filter, but only 1000 samples per second leave it. This is acceptable since all high-frequency components have been removed, and it reduces the computational load of the rest of the algorithm.

3. Calculate a moving average over the duration of exactly 1 bit, again for both the I and Q signals. (Note: a moving average over say 10 samples is just first the average of samples 1, 2, …., 10, next the average of samples 2,3,….11, next over 3,4,….12, and so on.)

4. Since the phase of the incoming carrier is unknown, next calculate a Pythagorean sum of the I and Q contributions. This makes the result independent of the phase difference of the received signal and our local carrier.

The result of this step is the green line in the graphs produced by RSCW.

5. Once per bit, take a sample from the above Pythagorean sum. These samples should be taken at the right moment, namely such that the moving averages have just covered the duration of one entire bit (as opposed to part of one bit and part of the next bit). It is the job of the clock regeneration to find out the right sampling moments; see below.

6. Next, subtract from each sample a number which is the threshold. A suitable threshold is somewhere in between the typical signal level for a 0 (i.e., the noise received while the transmitter is off), and the level for a 1 (i.e., the result of both the transmitter's signal and the noise).

The result of this is (ideally) a signal that is positive if a 1 was transmitted, and negative if a 0 was transmitted. However, due to noise this may not always be the case. In the graphs, this signal is shown in cyan (light blue).

7. Finally, calculate the cross-correlation between these samples and sequences of 1s and 0s (or rather, -1s) corresponding to every possible message. Then the most likely message, given what we have received, is the one that has the largest cross-correlation, so output that message as the result of the decoding process. See below for a more detailed discussion of how to do this practically.
In this paper we have described overview of wireless communication using bulb, now which are used for high range secure communication. The input given are sequence of characters that are converted to be into dash and dots. The encoded data upon reaching the other side gets converted back to sentences. Hence communication is achieved without using wires and also includes simple technology.

REFERENCES

[10] Uvais Qidwai ,Qatar University, C.H.Chen, Qatar University ,2010.” Algorithm approach with Matlab”.