International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-1, Issue-4, October 2012

Morse Code Reader

N. S. Bakde, A. P. Thakare

Abstract*-- Whilst a cursory scan through the wavebands on a modern "normal" domestic radio receiver may reveal little in the way of Morse code transmission, this communications technique is still very much in use. Tuning in via a "communications receiver" or an older domestic receiver on the short wave (SW) bands will reveal Morse activity. At that time PICs were probably not even a twinkle in the eye of any semiconductor manufacturer. They were certainly not reality. Consequently, the EE design was based on a hardware mark-space ratio detector which fed separate Morse dots, to a pre-PC computer (Commodore PET 32K). This compiled the incoming logic into a binary format, matched it against a lookup table and displayed the results on screen. The design presented here is physically simpler, although the software is considerably more complex. The aspects of this design are different. A handheld unit that can receive Morse code, via audio input (internal microphone) or direct signal connection, and translate it for display on an in-built liquid crystal (LCD) alphanumeric screen. It consists of a PIC microcontroller which provides the interface between the Morse code input and the display unit.

Index terms*--- Major components and requirements, Translation requirements, Binary format, International Morse code, reception rate, circuit diagram, PIC processing, Message display

I. INTRODUCTION

Morse code is a character encoding for transmitting telegraphic information, using standardized sequences of short and long elements to represent the letters, numerals, punctuation and special characters of a given message. The short and long elements can be formed by sounds, marks, or pulses, in on off keying and are commonly known as "*dots*" and "*dashes*" or "*dits*" and "*dahs*". Originally created for Samuel F. B. Morse's electric telegraph in the early 1840s, Morse code was also extensively used for early radio communication beginning in the 1890s. For the first half of the twentieth century, the majority of high-speed international communication was conducted in Morse code, using telegraph lines, undersea cables, and radio circuits. However, the variable length of the Morse characters made it hard to adapt to automated circuits, so for most electronic communication it has been replaced by machine readable formats, such as Baudot code and ASCII.

The most popular current use of Morse code is by amateur radio operators, although it is no longer a requirement for amateur licensing in many countries. It also continues to be used for specialized purposes, including identification of navigational radio beacon and land mobile transmitters. Morse code is designed to be read by humans without a decoding device, making it useful for sending automated digital data in voice channels. For emergency signaling, Morse code can be sent by way of improvised sources that can be easily "*keyed*" on and off.

Revised Manuscript Received on 30 October 2012.
* Correspondence Author
Nilima S. Bakde, Department of Electronics & Telecommunication, Sipna College of Engineering & Technology, Amravati. Maharashtra, India.
A. P. Thakare, Department of Electronics & Telecommunication, Sipna College of Engineering & Technology, Amravati. Maharashtra, India.

II. MAJOR COMPONENTS AND REQUIREMENTS

There are three main aspects to this new design. It comprises:

- A handheld unit that can receive Morse code, via audio input (internal microphone) or direct signal connection, and translate it for display on an in-built liquid crystal (LCD) alphanumeric screen. The received Morse signals are also available as pulses (OV/5V logic) for external use via a separate connection. Signals re-modulated at approximately 1kHz can be output to high-impedance headphones. With a suitably connected Morse key. Signals can be input manually.
- Using a PC computer, Windows-based software can input the signal being repeated from the handheld unit, convert and display the code on the PC monitor, and store the translation to disk for future examination as a text file.
- The PC software can additionally be used to output Morse code to the hand held unit, for display on its screen, or monitoring as an audio signal. There are several modes of code output from the PC: translation of a text file to Morse, direct keying of alphanumeric characters for immediate translation to Morse, use of the keyboard as a Morse key will the duration of key presses simulate; Morse dots and dashes.

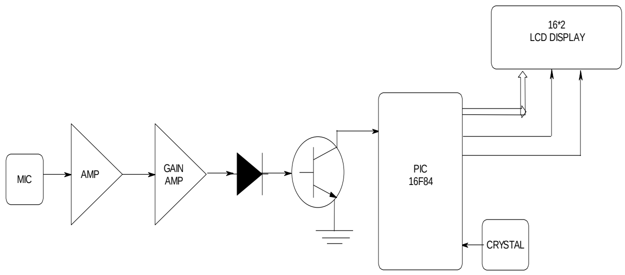

Fig. 2.1: Block Diagram of Morse code Detector.

Several other features are also included in the PC software, as will be described later. The handheld unit can be used on its own. It is not necessary to use it with a computer. Various aspects of the PC software can be used on their own, too, without the need for the handheld unit. In principle, Morse code signals at normal logic levels (0V to 5V pulses) can be directly input to the computer from other sources. The system can be used as a learning aid by those who wish to expand their understanding of Morse. It will also satisfy the curiosity of those who just want to "eavesdrop" on what radio operators are saying to each other.

III. TRANSLATION REQUIREMENTS

International Morse code uses the dot-dash combinations listed in Table 1. Conventionally, a dot is known as "DIT", and a dash as "DAH".

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Whilst the rate of code transmission is up to the Morse operator, the relative duration of the DITs, DAHs and associated spaces has been established by international agreement:

- The DIT is the basic unit of length
- The DAH is equal in length to three DITs
- The space between the DITs and DAHs within a character (letter) is equal to one DIT
- The space between characters in a word is equal to three DITs
- The space between words is equal to seven DITs

These are the basic requirements that any human operator or translation software must observe.

The sending of Morse signals can take many forms, ranging from audio and radio transmission, modulation of light (e.g. Aldis lamps and torches), varying electrical pulse levels (e.g. sending to a computer), to bashing the water pipe if the sender is incarcerated at "Her Majesty's Pleasure"! In audio and radio transmission, the technique is to turn the modulation of a carrier frequency (CW - continuous wave) on and off at the required rate. In audio work, the received signal is already within the audio range of the listener. Radio signals must, of course, be demodulated to become an equivalently pulsed audio signal. There are no set rules regarding the audio frequency of Morse signals, but they must, naturally, lie in the range most likely to be heard clearly, at about 1 kHz, for example. Automatic decoding equipment, therefore, must be able to accept Morse signals as a pulse-modulated frequency. It must also be able to recognized unmodulated pulse levels originating from a voltage simply being switched on and off.

The equipment must be capable of differentiating DITs from DAHs, and letter spaces from word spaces, irrespective of the rate at which the Morse signals are being received. Ideally, it should detect if the transmission rate changes and then readjust its DIT-DAH criteria correctly recognized DIT-DAH ratios. The unit described now makes its own adjustment, typically within about eight to 16 key presses (DITs and DAHs) being received.

Thus, all you need to do is place the unit near the loudspeaker of a radio receiver, or directly plug it into the coded signal source, and observe the unit displaying the received code as an intelligible text message.

The term "intelligible" is used loosely, of course. The unit won't translate from Swahili into English, for instance! It will simply show the letters being received.

IV. BINARY FORMAT

If you examine Morse codes as though DITs are logic 0 and DAHs are logic 1, a binary coded pattern will be seen. Converting from binary to decimal reveals a snag, however. There are some Morse codes that have one or more "leading" DITs, i.e. leading zeros. For example, take the letters E, I. S and H. which are Morse coded as DIT, DIT-DIT. D1T-DIT-DIT and DIT-DIT-DIT-DIT (the phrase *Elephants In Straw Hats Ten Miles Off* was that taught to the author to remember these four and their three DAH counterparts T, M, O - DAH. DAH-DAH, DAH-DAH-DAH!).

With each DIT as logic 0. the binary value of each of the first four letters converts to zero decimal. Not a helpful fact if regarding Morse codes as being true binary symbols.

The answer is to also take note of the number of key presses (DITs or DAHs -call them binary bits) in a coded letter. Now each code can be allocated two decimal numbers, its length as well as its binary value. Separate lookup tables can now be used, each dedicated to a particular code length, and then to the binary value.

International Morse Code

1. A dash is equal to three dots.
2. The space between parts of the same letter is equal to one dot.
3. The space between two letters is equal to three dots.
4. The space between two words is equal to seven dots.

A	• —	U	• • —
B	— • • •	V	• • • —
C	— • — •	W	• — —
D	— • •	X	— • • —
E	•	Y	— • — —
F	• • — •	Z	— — • •
G	— — •		
H	• • • •		
I	• •		
J	• — — —		
K	— • —	1	• — — — —
L	• — • •	2	• • — — —
M	— —	3	• • • — —
N	— •	4	• • • • —
O	— — —	5	• • • • •
P	• — — •	6	— • • • •
Q	— — • —	7	— — • • •
R	• — •	8	— — — • •
S	• • •	9	— — — — •
T	—	0	— — — — —

V. RECEPTION RATE

It will be obvious that the software must have a "base-timing" value against which it assesses DIT, DAH and space lengths. Such lengths depend on the sending operator's keying speed, which can vary considerably between operators. A novice might send at, say, only five words per minute (WPM). An experienced operator could even be sending at 50 WPM (about 25 WPM is a more typical rate).

The software assesses the sending rate by looking for the shorter pulses (the DITs). Initially, a temporary reference value is set to a high timing number, greater than the expected incoming pulse lengths. For a cycle covering the next 16 key press pulses, each pulse timing length is compared with this reference. If it is less, the reference is set to the same value as the pulse.

The comparison is repeated for all 16 key presses. It is then assumed that the reference value is that representing a DIT. The DAH and space values referred to earlier are then set in respect to this value. Again the reference value is set higher than the expected incoming pulse lengths and the cycle repeats.

Simultaneously with the reference value comparisons, each incoming key press is compared against the current DIT, DAH and space lengths, and each code sequence compiled as an equivalent binary value and in relation to its bit count. During the letter spaces the equivalent character is found from the respective lookup table and displayed on screen. If a word space is found, a space character is also sent to the screen. DIT length comparison, of course, is not fool-proof and noise or sporadic changes of operator keying rate may cause temporary misinterpretation of incoming codes, probably signified by a sequence of the letter T being seen. Usually, a recovery from such instances is made within 16 key presses.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

It was also found that when feeding the unit with computer-generated codes, slippage could still occasionally occur.

This is due to the PC monitoring other aspects of its system even though it is also running the Morse program. One PC in particular was excessively prone to this. It periodically decides that it wants to check all sorts of things on the hard drive and the floppies, thoroughly disrupting Visual Basic (and Quick-Basic) timings. The reason cannot be found (the machine came to the author second-hand).

Visual Basic does not allow internal "interrupts" to be stopped. They can be stopped if a machine code program is being run, as the author used to do when using QB with an m/c sub-routine, but he has not yet found a way to integrate *m/c* (8086 assembly dialect) with VB. (Advice from anyone who does know would be appreciated!)

Using a PC as the Morse source, translation rates in excess of 50 WPM were achieved with the PIC unit.

VI. CIRCUIT DIAGRAM

The complete circuit diagram for the *EPE* Morse Code Reader is shown in Fig. 6.1. Not much to it! Basically, Morse signals are input, amplified, translated by a PIC16F84 microcontroller and displayed on the LCD screen. Microphone MTC1 is a miniature electrets type which receives its power via resistor Rl and allows the unit to be placed near the speaker of a radio receiver to pick up Morse signals without any physical connection to it.

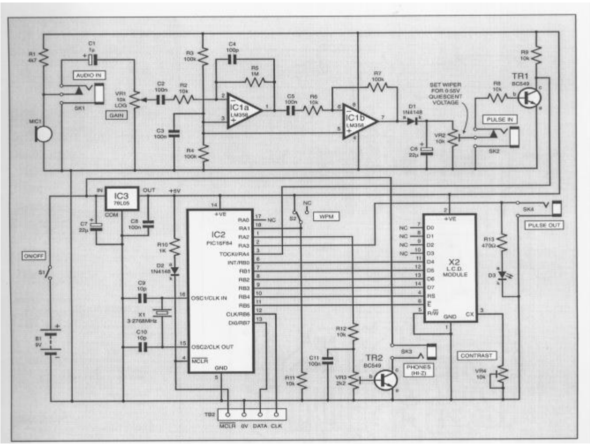

Fig.6.1: Circuit Diagram of Morse Code Detector using PIC 16F84

Socket SKI enables direct connection to, say, a radio receiver's audio output socket, low level or line-level. The microphone is automatically disconnected in this instance. Signals from the selected source are a.c. coupled to level control VR1 and fed to the amplification stage around 1C la. The gain is set at about 100 by resistors R2 and R5. The values of capacitors C2 and C4 respectively give a bit of bass and treble cut to the audio (although not totally eliminate) false triggering by any out-of-band noise on the signal.

From 1C la, the signal is ac coupled to the second amplification stage, around TClb. Here the gain is set at around 10 by resistors R6 and R7. Resistors R3 and R4 provide a midway bias level to both stages.

The next stage extracts the Morse pulse "envelope" from the audio carrier signal. In the presence of pulses (DITs and DAHs), capacitor C6 is held charged via diode Dl. When each pulse ceases, C6 discharges through preset VR2.

For as long as the voltage on the wiper of VR2 is above about 0-6V, transistor TR1 is turned on into full saturation, i.e. its collector voltage is at 0V. When each pulse ceases, the collector voltage returns high, to 5V (the power rail voltage).

Socket SK2 allows external (unmodulated) Morse pulses to be input in place of the audio signal (from a Morse key or computer, for example). Their amplitude should swing between 0V and greater than about 0-6V. The software automatically compensates for the signal inversion by TR1.

VII. PIC PROCESSING

The output from TR1 is coupled to the Schmitt trigger input, RA4, of PIC microcontroller IC2. The software monitors the status of the input, from which information Morse pulse lengths are assessed.

The status of pin RA4 is copied by software (suitably re-inverted) to pin RA3. This allows demodulated pulses to be sent, via socket SK4, to other equipment, such as a PC which itself can decode signals into characters and display them on its screen. While developing the software, the author actually coupled two PCs to the PIC, one transmitting to it, the other receiving from it.

The pulsed signal from RA3 also drives an LED D3, via ballast resistor R13. This serves as an additional Morse code monitor. At lower transmission rates, the relative DIT and DAH lengths can be observed, and even used as a visual source of Morse axle.

Provision has been made for received Morse signals to be repeated as re-modulated audio tones (at roughly l-3kHz) via high impedance headphones, connected via socket SK3. This facility is included as part of the unit's "learning" aspect, so that code transmissions simulated by the computer can be listened to and so test the listener's ability to mentally decode them.

The tones are output from pin RA2, sent via level control VR3 to transistor TR2, to which the headphones can be connected via socket SK3. It is stressed that high-impedance headphones (e.g. at least 40 ohms) *must* be used. The use of low impedance 'phones or a loudspeaker will kill the transistor (which has a rating of about 100mA).

Capacitor Cl I smoothes some of the harshness of the audio square wave - no attempt has been made to provide a "musi-cal" tone! In fact, using the software to generate the tone while carrying out other activities does not allow a precise audio frequency. Whilst the dominant frequency is about l - 3 kHz, other underlying tones are just noticeable. If you would prefer to listen to "cleaner" tones and you have an existing audio oscillator that can be keyed by voltage level changes, it could be driven via the pulse output at SK4.

VIII. MESSAGE DISPLAY

Visual display of the decoded Morse signals is via the 2-line 16-character (per line) alphanumeric LCD, X2. This is operated in standard 4-bit mode, with contrast setting performed by preset VR4.

Switch S2 causes the LCD. to show either two lines of message, or one line plus WPM data on the other.

In 2-line mode, the message is compiled on the lower line, the characters being placed consecutively from left to right across 16 "cells".

On each 16th character the whole lower line is transferred to the top, the lower one cleared and message compilation starts again from the left. The message is not stored after being lost from the display.

In WPM mode, the WPM count is assessed every 60 seconds and output to the left of the upper line. The lower line shows the message as it progresses, with it being cleared after each 16 characters.

At the right of the upper line is displayed a sub-count of the words received since the last one-minute display occurred. It is updated after each batch of 16 characters has been received.

IX. APPLICATIONS

Representation of SOS-Morse code:

An important application is signaling for help through SOS or ...---... . This can be sent many ways: keying a radio on and off, flashing a mirror, toggling a flashlight, and similar methods.

Morse code as an Assistive Technology:

Morse code has been employed as an assistive technology, helping people with a variety of disabilities to communicate. Morse can be sent by persons with severe motion disabilities, as long as they have some minimal motor control. In some cases this means alternately blowing into and sucking on a plastic tube ("puff and sip" interface). People with severe motion disabilities in addition to sensory disabilities (e.g. people who are also deaf or blind) can receive Morse through a skin buzzer.

X. CONCLUSION

The most popular current use of Morse code is by amateur radio operators, although it is no longer a requirement for amateur licensing in many countries. It also continues to be used for specialized purposes, including identification of navigational radio beacon and land mobile transmitters. Morse code is designed to be read by humans without a decoding device, making it useful for sending automated digital data in voice channels. For emergency signaling, Morse code can be sent by way of improvised sources that can be easily "keyed" on and off. MORSE is not dead! With modern communications systems abounding, it may seem so to the uninitiated, but in fact it is "alive and keying".

REFERENCES

1. How To Restore Telegraph Keys: W. R. Smith, W4PAL
2. Perera's Telegraph Collectors Guide (2nd. Edition)
3. Telegraph Collectors Reference (New 2nd. Edition)
4. Principles Of Telegraphy - N. N. Biswas
5. Arnold, G. (Ed.). (1994). Morse 2000 Conference. Morsum Magnificat. Issue 34, 7-8.
6. Gross, K. & Henderson, K. (1992). Comparison of Morse Code Software Programs. Presentation and handout at closing the Gap International Conference.
7. Western Digital My *Book* - Wikipedia, the encyclopedia
8. www.electronic-engineering.ch
9. www.books.google.co.in
10. www.hamradio.cc
11. www.trash.net
12. www.texttospeechblog.com

AUTHOR PROFILE

Ms. Nilima. S. Bakde, graduated from Government College of Engineering in Electronics & Telecommunication from SGBAU, Amravati university. She is a student of M. E. (Digital Electronics) from Sipna Collage of Engineering & Technology, Amravati (India). She is life member of IETE and ISTE.

Prof. Ajay P. Thakare, graduated from Govt. COE, Pune and received Master's Degree in 1996 in Electronics Engineering from SGBAU, Amravati University, India. He has 22 years of teaching experience & currently working as Professor and Head in the Department of Electronics & Telecommunication at Sipna's College of Engineering & Technology, Amravati (India). He worked as Assistant Professor and Section Head of Communication Tech. Department in the Defence Engineering College Debre Zeit, Addis Ababa, East Africa for over 3 & 1/2 years. His main research interests include Signal processing; Electromagnetic, Smart Antenna, Microwave Communication. He is pursuing research in the area of feed network for antenna beam formation. He is a Fellow-member of IE (I), IETE, ISTE and IEEE (USA). He has over 20 publications to his credit in National and International Journals & Conferences.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)