A PIC16F84 based CW Decoder

The project in few words

I propose a simple CW decoder wich makes use of a PIC16F84 microprocessor with a 16x2 chars LCD display. It is equipped with an audio frequency input from your receiver, an input for a stright key and an audio output locked to the input signal. It automatically adapts itself to the CW rate and may be employed for learning purpose substituting the traditional tone generator and offering the capability of displaying the keyed code.

Introduction

This project arises from a twofold need connected with CW learning, first of all to enance the performance of the traditional oscillator providing it with a display on wich you can verify the correctness of your keying, and then the need to have an instrument you can couple to the receiver so as to help those who, being still novices, are in a great trouble trying to deal with their early CW QSOs.

However it should be clear that, in my opinion, neither this device can substitute the ear's and brain's interpretation capability nor other similar instruments can do that. At the most they can help in quickening the code learning.

The decoding capabilities are essentially connected to the received signal quality, it must by clear and strong enough, so don't think you can decode a weak and vanishing signal in the QRM, if this is your goal, you should much better make use of your ears. If however the signal is good and stable enough, then this equipment can succeed in doing its job well, adapting also itself to the CW rate, provided that it is sufficiently regular.

Specifications

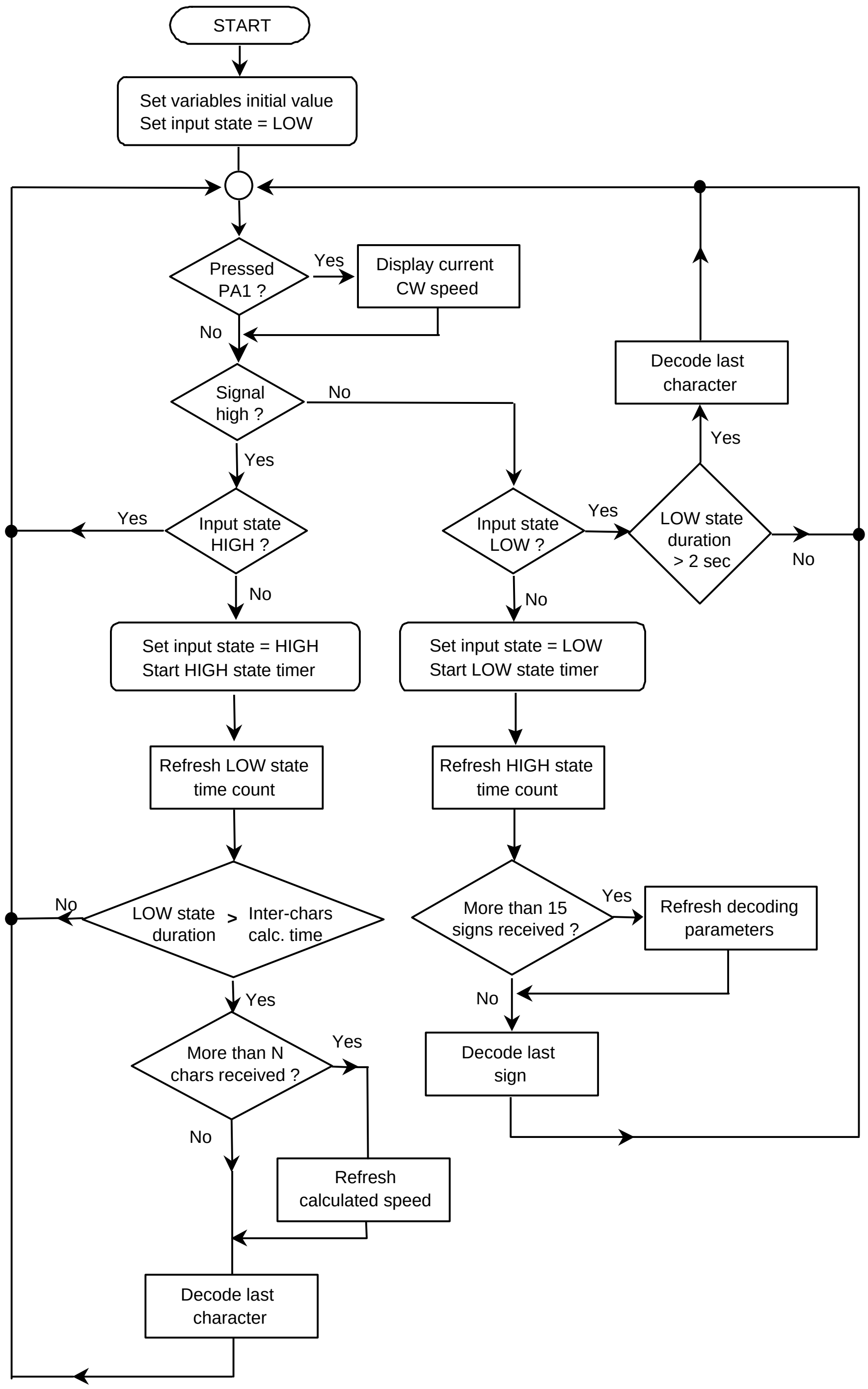
The device is equipped with a 2x16 LCD display , the text shifts from left to the right starting from the end of the second raw. An inter-words automatic spacing function is provided, based on a regular timing of the pauses in the sent code. This function may be inhibited grounding the J pin if the device is used for training purpose or while receiving an improperly sent code. The audio input must be at least 100 mV pp, a clipper is provided to cut large signals. The band width is about 100 Hz and the center frequency may be adjusted between 700 and 1000 Hz by a trimmer. A service push button (P1) displays the keying rate in chars/min, this measure is refreshed every N received characters (N is a settable software parameter). An input is provided for a straight Key, and both inputs (audio and key) activate the code display and the audio monitor function, a LED is operative while receiving code and shows the correct lock to the audio input, these two monitoring functions are very helpful to adjust the receiver tune because of the narrow bandwith of the decoder. The BF monitor can drive a 32 Ohm earphone with the two sides series connected. The Vcc can be supplied by a 9V transistor battery and requires about 15 mA. An external supply (min 9V) is however recommended for long time use. When powered on, the microprocessor is setted for an intermediate keying rate, some characters may be therefore required to reach the lock with the received signal if it is very slow or very fast.

The Decoder Software

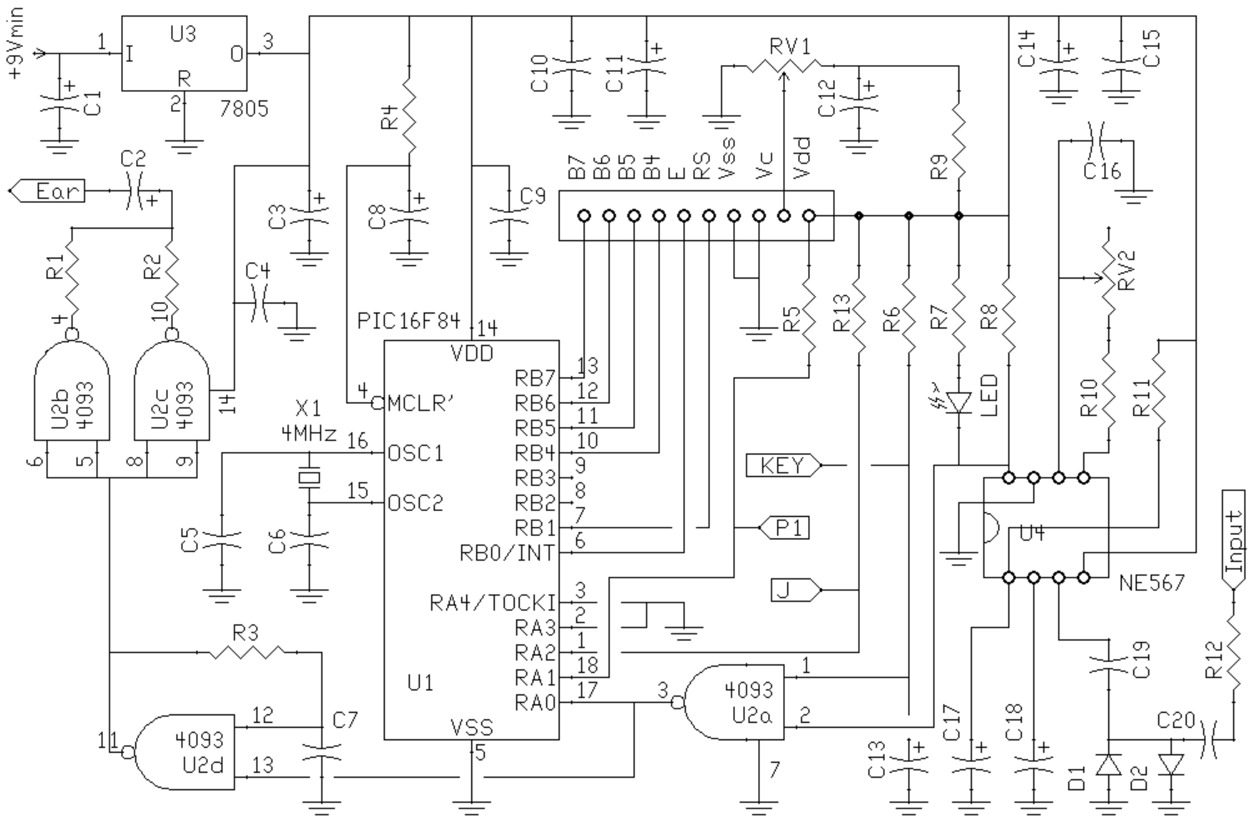
The software I developed makes use of the assembler PIC16 language and run on a PIC16F84 microprocessor. It takes a measurement of the received signal ON and OFF time, obtains some statistical mean values, and calculates three parameters wich are then used for decoding :

- Mean length of the dit/dash cycle
- Mean length of the inter-characters pause
- Mean length of the inter-words pause

A flow chart of the program (macro level) is listed below



Electrical Schematic



R1 : 1.8 KΩ	R11 : 33 KΩ	C8 : 1 µF	C18 : 0.5 µF
R2 : 1.8 KΩ	R12 : 3.3 KΩ	C9 : 100 nF	C19 : 100 nF
R3 : 18 KΩ	R13 : 18 KΩ	C10 : 100 nF	C20 : 100 nF
R4 : 22 KΩ	C1 : 47 µF - 25 Vl	C11 : 220 µF	U1 : PIC16F84
R5 : 18 KΩ	C2 : 4.7 µF	C12 : 22 µF	U2 : 4093 CMOS
R6 : 18 KΩ	C3 : 100 µF	C13 : 1 µF	U3 : 78L05
R7 : 820 Ω	C4 : 100 nF	C14 : 220 µF	U4 : NE567
R8 : 10 KΩ	C5 : 82 pF	C15 : 100 nF	RV1-RV2 : 4.7 KΩ
R9 : 10 KΩ	C6 : 82 pF	C16 : 100 nF	D1-D2 : OA95 - AA118
R10 : 10 KΩ	C7 : 100 nF	C17 : 1.5 µF	X1 : 4 MHz xtal

The schematic appears very simple, actually almost all of the functions are performed by the microprocessor software, while an NE567 tone decoder takes charge of processing the audio input signal. This IC contains a PLL circuit whose lock frequency may be adjusted between 700 and 1000 Hz by the RV2 trimmer.

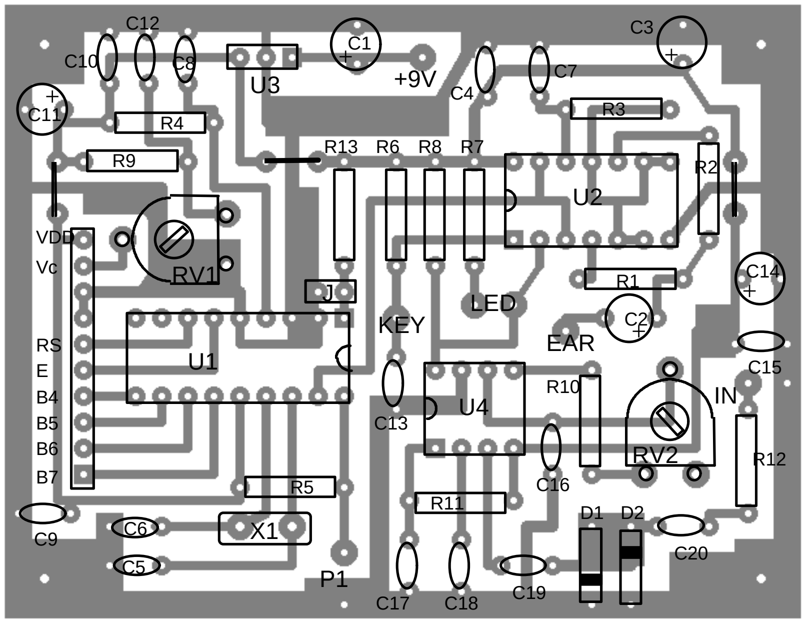
With the listed component values it is obtained a band width of about 100 Hz. The minimum accepted input signal amplitude is 100 mV pp and its time duration would be at least 20 mS. The PLL measured lock delay is about 10 mS.

A clipper is provided to limit the input signal amplitude, it is obtained by 2 germanium diodes (OA95, AA118, …. not critical).

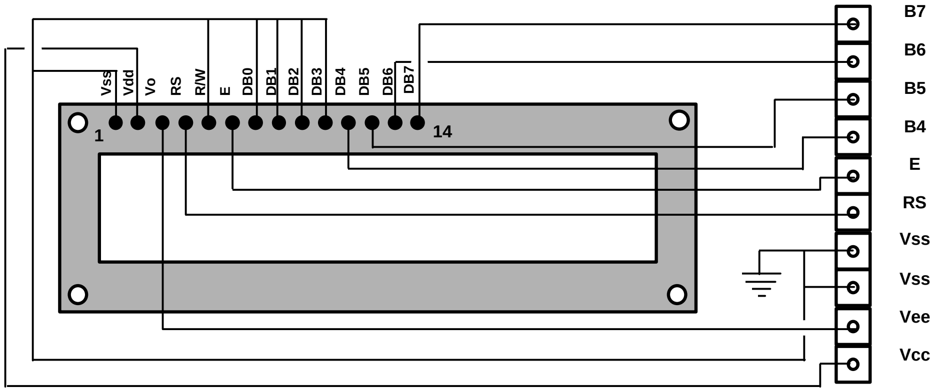
A LED is connected to the pin 8 of the PLL to show the correct frequency lock. The same pin 8 drives one gate (pin 2) of the CMOS trigger NAND 4093 whose output (pin 3) is connected both to the microprocessor PA0 gate and to a second CMOS gate (pin 13)

working as an audio generator. The remaining two 4093 gates are used to implement a buffer capable of driving a medium impedance load (64 Ohm).
A 78L05 regulator supplies both the decoder module and the LCD display, the RV1 trimmer is used to adjust the display brightness.
The only required tuning is an RV2 trimmer adjustment so as to obtain the best frequency lock using an input CW signal strong and clear from your receiver

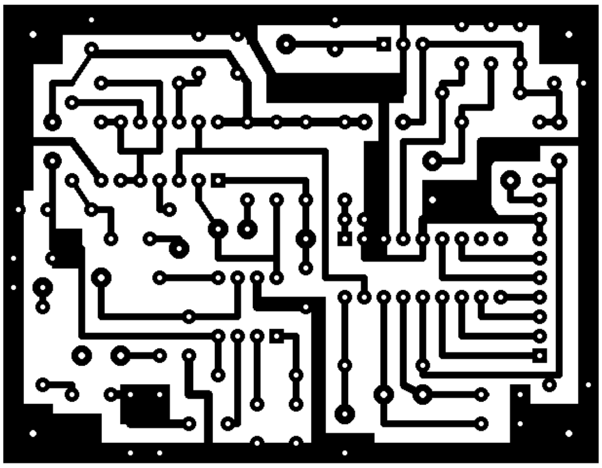
Component Layout



The whole circuit is mounted on a 60x80 mm PCB board. It is recommended to make use of small sized components (ceramic multilayers capacitors, low voltage electrolitics, …) so as to make easier the assembling. The LCD display is connected by a 10 pole flat cable soldered to 2.54 mm standard connectors, as shown below



PCB board (real dimensions)



The PIC16F84 development tool

You may find several commercial development tools for the PIC16F84 micro (see for example the Microchip Starter Kits). However if you are interested in a low cost solution, you can download from the Microchip WEB site **http://www.microchip2.com/index.html** the assembler software MPASM (**http://www.microchip.com/10/Tools/pTools/ MPASM/ index.htm**) and the simulator MPSIM (**http://www.microchip.com/10/Tools/Archive/ index.htm**) together with the technical documentation. Regarding to the hardware programmer construction, it is easy to find many references on Internet (I'll be glad to provide some informations). You can download a shareware version of the CIRCAD software from the Holophase WEB site **http://www.holophase.com/**

Final notes

No particular difficulty should arise in the realization, neither in the finding of the components nor in the assembling of the decoder. If you are interested in obtaining more informations, a copy of the PIC software or the CIRCAD PCB files, you may contact me at my E-mail box : ik3oil@arrl.net.