

Application Note

AN2124

Morse Decoder for the PSoC™

By: Melchor A. Varela Morales

Associated Project: Yes

Associated Part Family: CY8C25xxx, CY8C26xxx

PSoC Designer Version: 3.21

Referenced Application Notes: AN2122, AN2038, AN2032

Summary

The mixture of analog and digital processing capabilities, plus its low cost and ease-of-use, makes PSoC an ideal solution for amateur radio operator's (HAM's) projects. This Application Note describes the development of a Morse decoder based on a PSoC device and an LCD display.

Introduction

The PSoC family of programmable system-on-chip devices is great for HAM's accessory projects because the configurable analog and digital blocks allow users to build complete solutions in only one chip. The software development tools are free and a low-cost emulator is available as well.

As a simple example of the capabilities of PSoC for HAM projects, this Application Note describes the development of a Morse decoder using only the PSoC device and an LCD.

This decoder will automatically decode the Morse symbols and present them on the display using an automatic scroll feature. It features automatic speed tracking and continuous wave (CW) speed display. Applications range from the eaves-dropping of Morse code QSOs to learning and practicing Morse code.

The decoder has an analog front-end composed of an input band-pass filter (BPF) and an 8-bit Delta Sigma analog-to-digital converter (ADC), all implemented using configurable analog blocks in the PSoC architecture. Tone detection and symbol decoding logic are implemented in the software.

Morse Telegraphy

Telegraphy, by on-off keying of a carrier, is the oldest radio modulation system, but is still in use, especially by HAMs. It is also known as continuous wave or CW.

The Morse code character set defines the characters as a series of DIT and DASH symbols. The reference symbol duration is the DIT. A DASH symbol is three DITs long. The space between a pair of symbols is one DIT long. The space between characters is three DITs long, and the space between words is five DITs long.

The speed of Morse Telegraphy is usually expressed in words per minute (WPM), rather than bauds. The following formulae relate WPM to bauds:

$$WPM = 2.4 \times \frac{\text{dots}}{s} \quad (1)$$

$$WPM = 1.2 \times B \quad (2)$$

Where:

- WPM = telegraph speed in words per minute
- 2.4 = a constant calculated by comparing dots per second with plain language Morse code sending the word "PARIS"
- B = telegraph speed in bauds

Thus, a keying speed of 24 dots/s or 12 baud is equal to 10 WPM.

For aural reception in a typical HAM radio receiver, a Morse code RF signal is not completely demodulated to its original DC pulse because only thumping would be heard. Instead, the signal moves down to AF, usually near 700 Hz.

The presented Morse decoder detects the AF signal, in this case tuned to 800 Hz. It measures the on-off keying periods that it interprets and translates them in order to be shown in the display.

Decoder Operation

The LM7805 can be used to power the PSoC by 9V or 12V. The decoder is connected to the receiver speaker output or any other audio output. The design can be easily adapted to use a microphone instead of a direct connection to the receiver, and to use batteries instead of the power supply. (Note that the audio signal must be lower than 5 volts, otherwise the PSoC may be damaged.)

The operation of the decoder is completely automatic, and no user intervention is required. The only requirement is to tune the receiver to a Morse signal and adjust the receiver output level. It is important to adjust the receiver so that the AF Morse signal is centered around 800 Hz. The signal coming from a CW beacon is great to start the operation.

The decoder has a 2 line-by-16 column LCD to present the decoding results. The first line shows the received characters. When the line is complete, upon receiving the following characters, the decoder shifts the display line character-by-character to the left, so it will always display the last 16 characters received. Figure 1 shows the decoder screen:

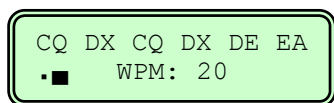


Figure 1. Decoder LCD Display

The second line of the LCD is dedicated to the decoder status. The first column shows the tone-detector status. It displays a dot ('.') when a tone is detected and a space when no tone is detected. The second column displays the signal strength level using a vertical bargraph. The remaining columns display the current WPM rate. The decoder restarts to the default value of 20 WPM, and goes to the idle state when there are no symbols during a 6-second period. In the idle state, the WPM rate is displayed with lower dash symbols.

The decoding capabilities are essentially connected to the quality of the received signal.

The decoder will not decode weak and vanishing signals with bad signal-to-noise ratios. In most cases, performance is quite satisfactory.

The device has been tested with speeds from 10 to 40 WPM, but this range can be extended with further adjustments.

System Architecture

The decoder design consists of three principle blocks: analog front-end, tone detector, and symbol detection logic.

The analog front-end digitizes the input audio signal in order for the signal to be processed by the tone detector logic. The analog front-end is composed basically of an input BPF and an 8-bit Delta Sigma ADC, all implemented using PSoC user modules.

The tone detector is based on a quadrature correlator implemented by software, based on the implementation described in Ref. 1 "Cypress Microsystems AN2122: DTMF Detector."

The symbol detection logic will decode the Morse code characters by measuring and interpreting the on-off periods. The basis of this process is two moving-average filters; one for the 'DIT' period and another for the 'DASH' period, that provide the automatic speed tracking feature.

Tone Detection

The tone detector is based in a quadrature correlator implemented by the software. Correlation is the search for similarities between two sets of data, either a pair of signals (cross-correlation) or self-similarities in the same signal at different times (auto-correlation). Correlation of signals requires the computation of a sum of time-shifted products.

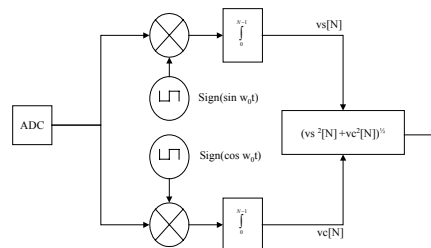


Figure 2. Quadrature Correlator

In this case, the correlator correlates the incoming signal with two reference quadrature signals, which correspond to the sign function of sine and cosine signals of the tone to detect. The sign function mimics a square wave of the tone to detect. The advantage of this correlator is the absence of multiplication operations.

The correlator performs addition when the reference signal is positive and subtraction when the reference signal is negative. The quadrature correlator is illustrated in Figure 2.

After processing N samples, the quadrature components are squared to eliminate incoming signal-phase dependence. The following formulae illustrate the correlator operation:

$$vs[n] = \sum_{n=1}^N x[n] \cdot \text{sign} \left[\sin \left(2\pi \frac{fd}{f_s} n \right) \right] \quad (3)$$

$$vc[n] = \sum_{n=1}^N x[n] \cdot \text{sign} \left[\cos \left(2\pi \frac{fd}{f_s} n \right) \right] \quad (4)$$

$$Z^2 = vs^2[N] + vc^2[N] \quad (5)$$

$vs[n]$ and $vc[n]$ are the sine and cosine correlator components, f_d is the detection frequency, and f_s is the sampling frequency. Sine and cosine Accumulators have to be initialized to zero.

The tone detector parameters, assuming the detection frequency f_d to be 800 Hz, have been selected with the following criteria. The N value has been chosen to have a detection time of about 5 ms in order to provide enough timing resolution, especially with high-speed coding. The sampling frequency has to be at least two times the tone frequency to satisfy the Nyquist criterion. It has been carefully selected in order to have an integer number of samples and signal cycles, taking care of selecting an integer divisor for the ADC clock.

Choosing a sampling frequency of 6465 sps and N fixed to 32, we get a detection frequency of 808 Hz and a detection time of 4.95 ms. In this case, the ADC clock is obtained from the 48M clock and a divisor of 29. For N samples, we get 4 signal cycles and 8 samples per cycle (45° phased). It is possible to implement the sign function by using the sampling counter because the sign function changes every 4 samples.

The sine and cosine functions have a 90°-phase shift, which is exactly 2 counter counts in this case.

Hardware Architecture

The required hardware elements are mostly implemented inside the PSoC device. The external elements are the power supply circuits, the LCD display, and the optional audio amplifier.

Power Supply

The PSoC is powered at 5 volts in order to operate at its maximum speed, 24 MHz. In any case, this design is operating at 12 MHz, so it could be modified to run at 3.3V. The 5-volt supply is generated from a 7805 linear regulator (U4) shown in the schematic (Figure 7). The input is protected against inverse polarity by a series diode (D1). An external DC wall adapter of 9V or 12V is adequate to power the unit.

Audio Amplifier

The audio amplifier (U5) is based on the common LM386. It is powered to 5 volts and the gain is set to 50; that is a comfortable value for this design. The volume control is done by a potentiometer (R5). The output is connected to a small speaker.

LCD Display

The required display is an industry standard LCD display based on Hitachi HD44780, or compatible display driver-bus protocol. The display is connected directly to the PSoC I/O ports and uses the 4-bit interface, so only 7 I/O pins are required. Resistor R1 must be adapted to the specific display used for optimum contrast, because the required bias setting is not standard for all display types.

PSoC

The PSoC device provides most of the required hardware elements. The decoder requires an analog front-end to digitize the input audio signal in order to be processed by the tone detector logic. It is composed of a BPF and an 8-bit Delta Sigma ADC. See Figure 3. Although this design does not require dynamic re-configuration, it has been implemented in order to prepare the design for future features. As a general rule, and even for designs not requiring dynamic re-configuration, I recommend creating designs with a base configuration of the most common user modules, and the required loadable configurations.

In this case, there is a base configuration with the LCD User Module and the DATA configuration implementing the analog front-end.

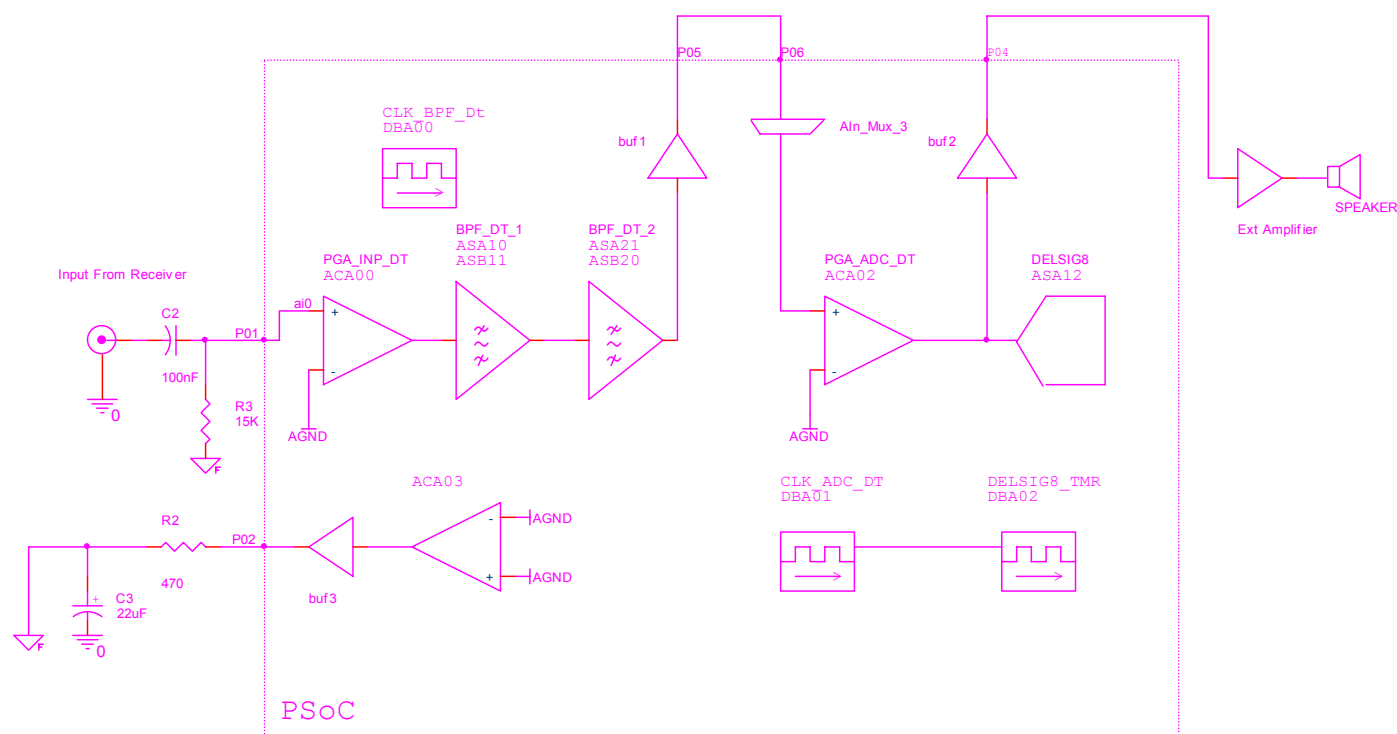


Figure 3. PSoc Configuration

The received AF Morse signal is normally in the 700 to 1000 Hz range. In order to minimize the unwanted frequency components and avoid the aliasing, a four-pole BPF is implemented. The filter has a Chebychev response (0.1dB ripple in the pass band). It has a center frequency of 800 Hz and a bandwidth of 800 Hz. Filter settings are shown in the following table:

Table 1. Filter Settings

Parameter	Low Pole	High Pole
Q	1,717	1,705
C1	3	5
C2	2	1
C3	4	21
C4	13	4
CA	32	32
CB	32	32
N	138	
Sample Clock	43478 Hz	
Gain	4 dB	

The band-pass filter (BPF_DT_1 and BPF_DT_2) is implemented using four switched capacitor blocks (ASA10 and ASB11 for the low pole, and ASA21 and ASB20 for the high pole). The timing reference is set by a Timer 8 User Module, CLK_BPF_DT (DBA00).

The filter input comes from a PGA User Module, PGA_INP_DT (ACA00). The gain is set to 1 (one), but can be adapted to meet the receiver audio output requirements. The input signal is decoupled by an external RC high-pass filter (C2 and R3). The proper DC offset is provided by a PGA User Module (ACA03), which outputs the AGND reference externally.

The filter output is driven externally to P0[5] and returned to the analog input multiplexer 3, AIn_Mux_3, via P0[6] in order that it be connected to the PGA, PGA_ADC_DT (ACA02). This PGA provides the interconnection path between the filter and the ADC, and also the connection to the external amplifier through Buf2 and P0[4].

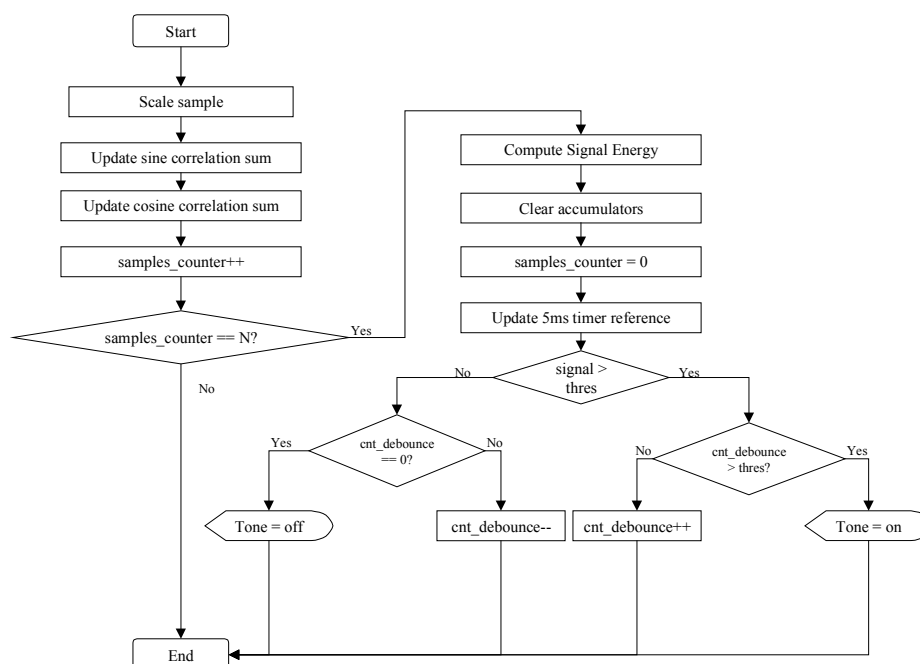


Figure 4. ADC Interrupt Service Routine (ISR)

The ADC is a DELSIG8 User Module. This module uses a switched capacitor (SC) block (ASA12) and a digital block (DBA02) to implement a timer. This configuration allows the proper number of integration cycles and the PSoC decimator to process the single-bit output stream from the analog block.

The ADC timer requires a prescaler in order to set the required sampling frequency: 6465 sps. This timer is implemented using a Timer 8 User Module CLK_ADC_DT that is driven by the 48M clock. Therefore, the divisor is set to 29, according to the sample rate formula for the DELSIG8:

$$SampleRate = \frac{DataClock}{256} \quad (6)$$

Software Architecture

The software is organized around a foreground active level, which performs symbol detection and presentation of tasks. The tone detection logic and timing reference is located in the ADC Interrupt Service Routine (ISR). The ADC ISR is serviced at the sample rate and takes the signal samples to be processed by the tone detection logic. Figure 4 shows the ADC ISR logic.

To provide a higher dynamic range, the tone detector scales the sample and multiplies it by the sign function of the sine and cosine reference signal. The result is accumulated in the sine and cosine Accumulators. The sine and cosine sign references are obtained from the sample's counter by using the third bit of the counter as a sign bit, so that every 4 samples the sign is changed (one reference signal cycle is 8 samples). The 90°-phase shift for the cosine reference is obtained by adding 2 to the sample counter and using the third bit of the counter as the sine.

After the N samples are processed, the energy is calculated by squaring and summing the quadrature components. This is implemented by the integer-multiply macros from Application Notes AN2032 and AN2038. The result is divided by 2^{16} for scaling. The Accumulators are then cleared to start a new computation.

Because the N value corresponds with a time of 5 ms, it is used as a time reference for the symbol detection logic.

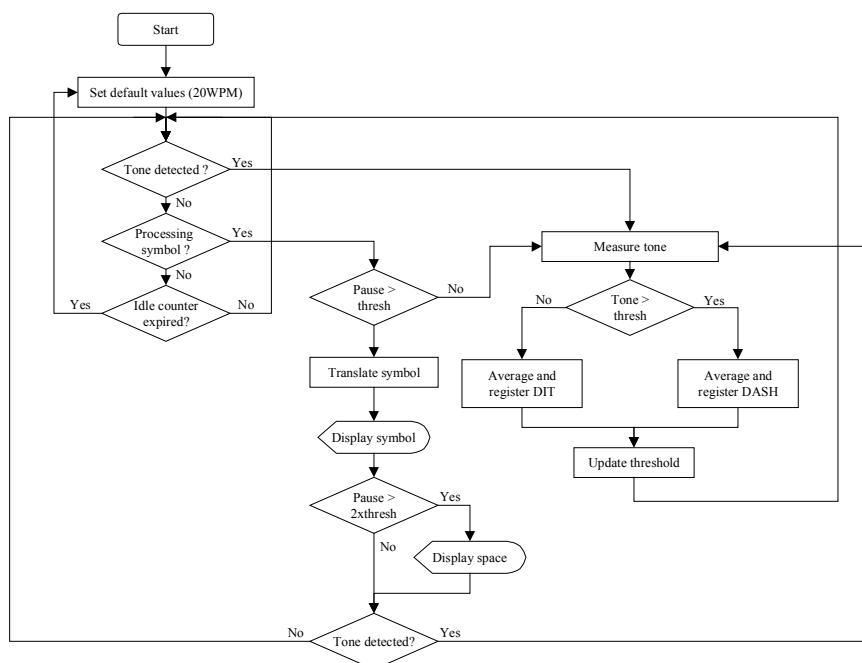


Figure 5. Symbol Decoding Logic

The signal energy value is used for the signal strength level bargraph, and is compared with a fixed value to determine the presence of a tone. The tone is validated using a debounce counter. When the signal energy value is above a decision threshold, a counter is incremented until it reaches an acceptable value. In that case, the tone is accepted as a valid tone. If the signal is below the decision threshold, the counter is decremented until it arrives at zero. In that case, it is considered as the end of the tone.

The symbol decoding logic is illustrated in Figure 5. The decoder is continuously looking for transitions in tone detection in order to measure the tone and silence duration. The base of symbol decoding is two running-average buffers that store the last four values of the DIT and DASH interval samples. The average values of the DIT and DASH periods are used to set up the threshold for consecutive symbols. The threshold is the mean value of the DIT and DASH periods. The threshold is also used for character and word detection by measuring the length of the silences and comparing the values.

Character decoding is done via a look-up table indexed by two control variables. The values of the control variables depend on the previous symbol occurrences. Character decoding is performed when an inter-character or an inter-word silence has been detected. When an inter-word period is detected, an automatic space is displayed.

The WPM status is updated in every character occurrence by accessing the look-up table indexed by the current average DASH period.

When no symbols are detected in a period of about 6 seconds, the decoder resets to the default values in order to start the training with another Morse signal. The default decoding values are set for 20 WPM.

Display control is done by using the LCD User Module API. Besides ordinary text functions, the API provides the ability to display bargraphs that are very useful; in this case, for signal strength level display.

References

1. Cypress MicroSystems AN2122:
DTMF Detector
2. Cypress MicroSystems AN2032:
Unsigned Multiplication
3. Cypress MicroSystems AN2038:
Signed Multi-Byte Multiplication

The references are available on
www.cypress.com.

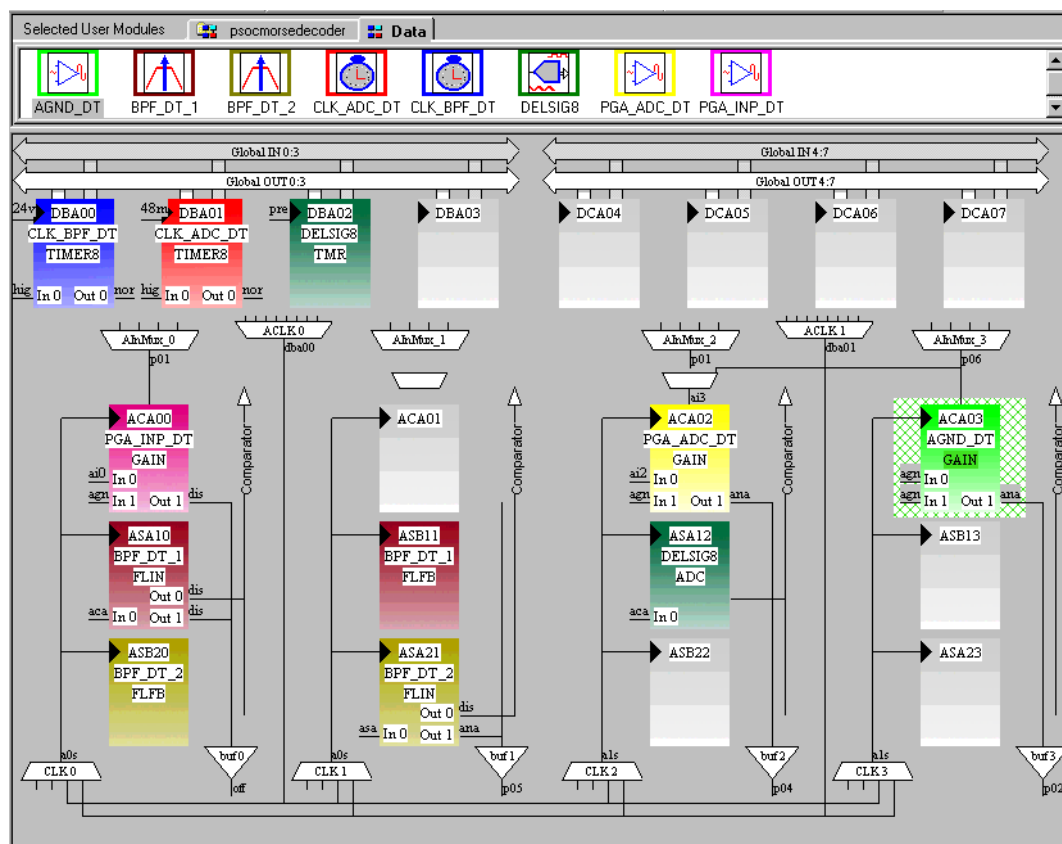


Figure 6. Configuration

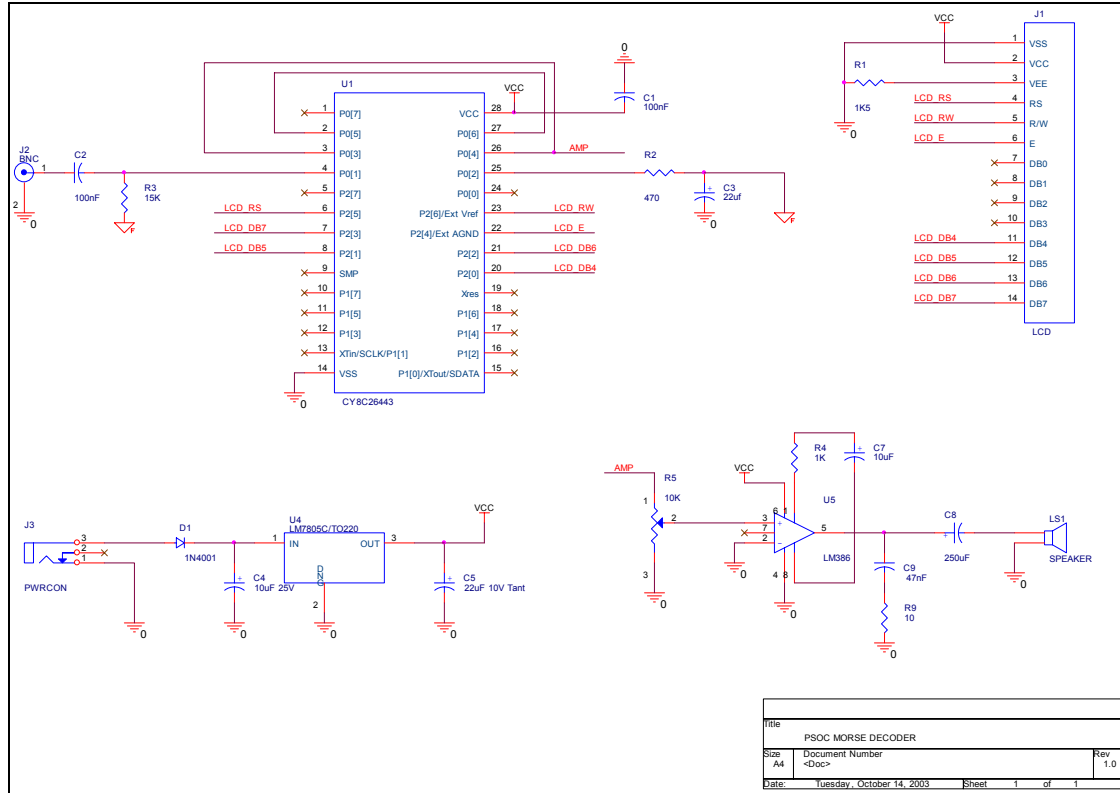


Figure 7. Schematic

Morse Code Character Set

A	• —	Period [.]	• — • — • —	Ä	• — • —
B	— • • •	Comma [,]	— — • • — —	Á, Â, Æ, Å	• — — • —
C	— • — •	Question mark [?]	• • — — • •	Ç	— • — • •
D	— • •	Hyphen [-]	— • • • • —	É, Ê	• • — • •
E	•	Double dash [=]	— • • • —	È	• — • • —
F	• • — •	Colon [:]	— — — • • •	Ê	— • • — •
G	— — •	Semicolon [;]	— • — • — •	Ö, Ó	— — — •
H	• • • •	Left parenthesis [(]	— • — — •	Ñ	— — • — —
I	• •	Right parenthesis [)]	— • — — • —	Ü	• • — —
J	• — — —	Fraction bar [/]	— • • — •	Ž	— — • • —
K	— • —	Quotation marks ["]	• — • • — •		
L	• — • •	Dollar sign [\$]	• • • — • • —		
M	— —	Apostrophe [']	• — — — — •		
N	— •	Underline [_]	• • — — • —		
O	— — —	End of message or cross [+]	• — • — •		
P	• — — •	Paragraph [¶]	• — • — • •		
Q	— — • —	End of work [']	• • • — • —	0	— — — — —
R	• — •	Wait [[]]	• — • • •	1	• — — — —
S	• • •	Understood [*]	• • • — •	2	• • — — —
T	—	Starting signal [➔]	— • — • —	3	• • • — —
U	• • —			4	• • • • —
V	• • • —			5	• • • • •
W	• — —			6	— • • • •
X	— • • —			7	— — • • •
Y	— • — —			8	— — — • •
Z	— — • •			9	— — — — •

About the Author

Name: Melchor A. Varela Morales
Title: R&D Electronics Engineer
Thales e-Transactions España.
Contact: m.varela@computer.org

Cypress MicroSystems, Inc.
2700 162nd Street SW, Building D
Lynnwood, WA 98037
Phone: 800.669.0557
Fax: 425.787.4641

<http://www.cypress.com/> / <http://www.cypress.com/support/mysupport.cfm>

Copyright © 2003-2004 Cypress MicroSystems, Inc. All rights reserved.

PSoC™ (Programmable System-on-Chip) is a trademark of Cypress MicroSystems, Inc.

All other trademarks or registered trademarks referenced herein are property of the respective corporations.

The information contained herein is subject to change without notice.