

# ***IS-54 Simulation***

## ***Application Report***

***John D. Crockett***

***Elliott D. Hoole***

***Thomas Labno***

***Stephen Popik***

***Wireless Communications Systems — Semiconductor Group***

SPRA135  
October 1994



Printed on Recycled Paper

## **IMPORTANT NOTICE**

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

**TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.**

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

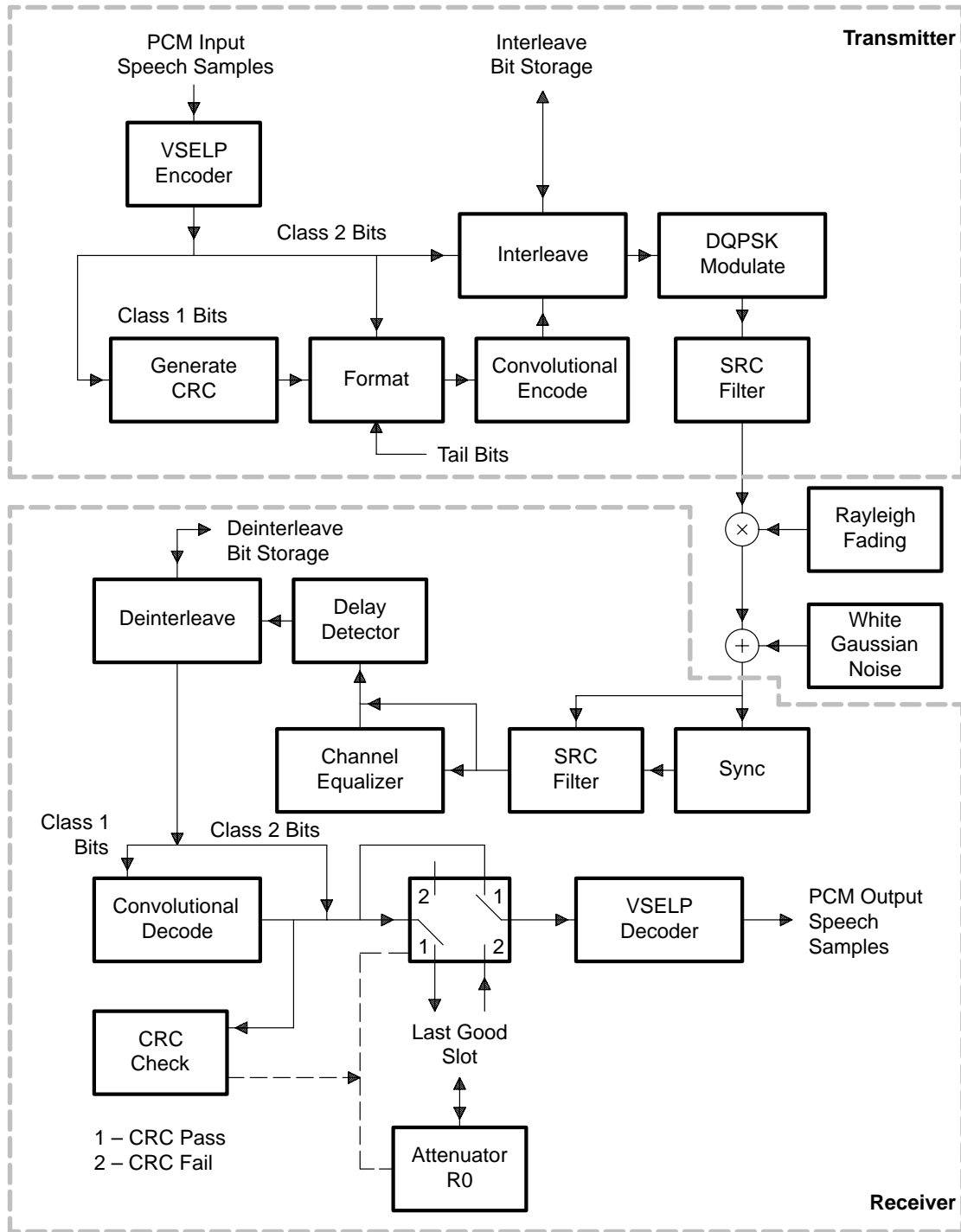
TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

## Introduction

This paper describes a C language simulation of both the transmit and receive baseband processing for a digital cellular telephone that meets the U.S. digital cellular standard (IS-54B). This simulation is needed for two reasons: first, to gain greater understanding of the IS-54 digital cellular standard and the associated digital signal processing required in a terminal that meets this standard with a vision toward efficient implementation on the TMS320 DSPs; second, to gain the capability to evaluate the effect of bit errors on the speech coder (vector sum excited linear prediction, or VSELP) and IS-54 control functions. This necessitated development of a simulation of the IS-54 processing and RF channel. See Figure 1.

The IS-54 standard separates the data bits into class 1 bits and class 2 bits. The class 2 bits are not protected and have less influence on the speech coder than class 1 bits. The class 1 bits are convolutionally encoded so that errors can be detected and corrected. In addition, a cyclic redundancy check (CRC) is calculated on the 12 class 1 bits designated as most perceptually significant. The CRC is also convolutionally encoded for error detection and correction and is used to signify noncorrectable errors in the most perceptually significant 12 bits for special error handling provisions. Consequently, the evaluation of the effect of bit errors on the voice coder must encompass all IS-54 transmit and receive processing functions.

**Figure 1. IS-54B Simulation Processing Block Diagram**



## Description

The IS-54 simulation starts with input speech parameters that are organized into 20-millisecond frames. Each frame is processed through the transmit path, the channel simulation, and the receive path.

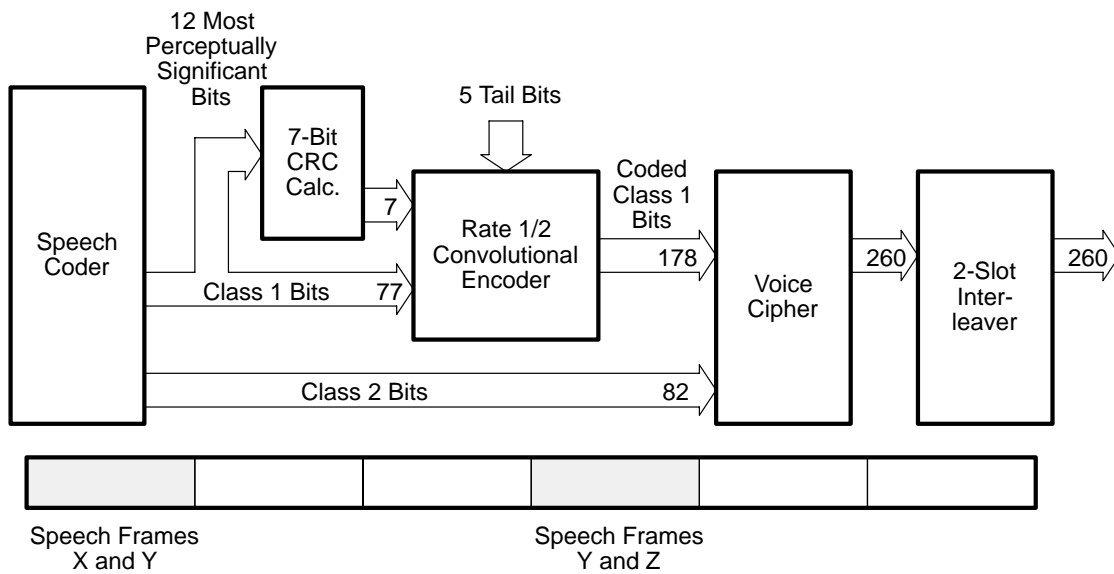
### Transmit Path

A block diagram of the IS-54 simulation is shown in Figure 2. The speech data is read into the simulation from an input speech file. This file is binary pulse-code-modulated 16-bit data. The VSELP encoder is the Motorola standard, which is available from the TIA. The VSELP encoder and decoder are not incorporated into this simulation but are run as a separate program. The output from that program is fed to this simulation, whose output is then used to create final PCM speech data. From the output of the VSELP encoder, the most perceptually significant bits of the encoded speech frame are packed into a binary word for generation of the CRC. The CRC is calculated by first multiplying the input word by  $2^7$  and dividing by a polynomial given in IS-54 as:

$$g_{\text{crc}}(X) = 1 + X + X^2 + X^4 + X^5 + X^7 \quad (1)$$

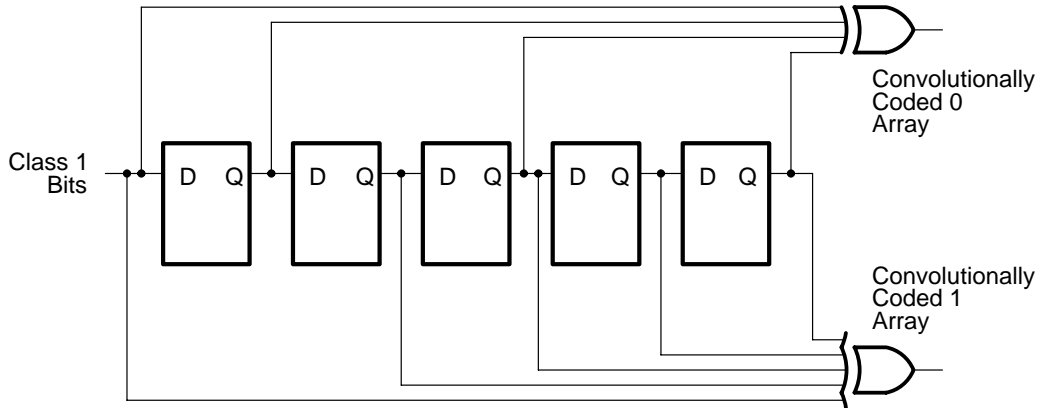
The quotient is discarded and the 7-bit remainder is kept.

**Figure 2. IS-54 Error Encoding and Interleaving**



The CRC, along with the other class 1 bits (IS-54 Table 2.1.3.3.3.4-2) from the VSELP data, is packed into the c11 array [1] to be encoded for forward error correction. The forward error correction is a rate 1/2 convolutional encoder with an initial state of 0x00. This encoder produces two output bits for each bit input. The last five bits fed into the convolutional encoder are tail bits of state 0 to force the encoder to also return to the zero state. A block diagram of the convolutional encoder is shown in Figure 3.

**Figure 3. IS-54 Convolutional Encoding Block Diagram**



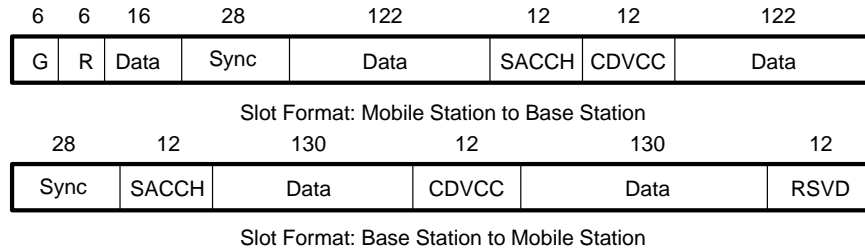
The output from the convolutional encoder, arrays cc0 and cc1 (IS-54 para. 2.1.3.3.3.4), are then packed into a 260-bit slot data array along with the class 2 bits (IS-54 Table 2.1.3.3.4-1). During this packing, the bits are shuffled around within the slot to minimize the probability that a burst error would affect more than one bit in the same vocoder parameter. This is shown in Figure 2 as voice cipher. The 260-bit slot data array is then interleaved with data from the previous frame so that the resultant transmitted burst consists of bits from both the current and previous frames. This interleaving of data across two transmit slots is designed to randomize the burst error across the data bits, thus increasing the probability that bit errors will be detectable and correctable.

The data, which consists of speech and redundant error correction information from two frames, is then formatted to the IS-54 slot format. See Figure 4. This consists of inserting the sync word, the SACCH data, the CDVCC field, and the reserved bits (the CDVCC and the reserved bit fields are filled with 0s in this simulation). The base-to-mobile format is used in order to focus on the processing stream in the handheld terminal.

The IS-54 standard specifies the modulation as  $\pi/4$  differential quadrature phase shift keying (DQPSK). The input data is paired into dibits, allowing for four symbols that specify a phase change from the previous point on the complex plane. Each dibit corresponds to an odd multiple of  $\pi/4$  phase change resulting in an 8-point modulation constellation. These eight points are also called maximum-effect points.

The next function in the transmit path is the square-root-raised-cosine (SRC) filter. This filter is employed on both transmit and receive sections for the composite effect of a raised-cosine filter transfer function. This results in a filter response with nulls at the adjacent symbols in order to minimize intersymbol interference. The transmit SRC filter also includes 4X interpolation. The overall filter response is split between the transmit and receive sections to allow more efficient use of bandwidth due to partial response signaling.

**Figure 4. IS-54 Slot Formats**



<p>G – Guard Time  R – Ramp Time  Data – User Information or FACCH  RSVD – Reserved (Set to 0)</p>	<p>SACCH – Slow Associated Control Channel  CDVCC – Coded Digital Verification Color Code  Sync – Synchronization and Training</p>
--	--

### Channel Model

At this point in an actual IS-54 handset, the data would then be input to the RF stage for modulation of the carrier frequency. Because this is a simulation, we chose to substitute simulated fading and noise generation for the transmit and receive RF portions of the IS-54 processing chain.

In a mobile radio environment, signals from many paths combine at the antenna. Depending on the relationship between the phase angles of the signals, the effect of the combination is interference that can be constructive or destructive. As the mobile radio moves, the relationship between the phase angles changes, causing the signals to be combined randomly and providing a challenge for receiver and system designers. The term for this effect is fading, and because the magnitude of the result occurs in a Rayleigh distribution about the mean value, it is called Rayleigh fading.

A simulator for generating Rayleigh fading was proposed by W. C. Jakes [4]:

```

N = 34;                      /* number of simulated signals */
N0 = 0.5 *(N/2 - 1); /* number of oscillators */
alpha = PI / 4;
V = 55;                      /* vehicle speed in MPH */
Fc = 850.0E+6;               /* carrier frequency */
lambda = 3.0E+8 / Fc; /* carrier wavelength */
wm = 2 * PI * V / lambda;
xc(t) = sqrt(2)*cos(alpha)*cos(wm*t);
xs(t) = sqrt(2)*sin(alpha)*cos(wm*t);
for (n = 1 ; n <= N0 ; n++)
{
    wn = wm * cos(2*PI*n/N);
    xc(t) += 2*cos(PI*n/N0)*cos(wn*t);
    xs(t) += 2*sin(PI*n/N0)*cos(wn*t);
}

```

xc(t) is the in-phase (cosine) component, and xs(t) is the quadrature (sine) component. This model provides a very good approximation of theoretical behavior and is excellent for general use.

Another major impairment to wireless communications is within the radio itself. As the received signal gets weaker, the signal-to-noise ratio decreases, and errors caused by thermal noise in the radio receiver can occur. This noise is characterized by a zero-mean, Gaussian probability density function in the time domain. In the frequency domain, the power spectral density of thermal noise is constant and is called white noise. In a real system, there are filters that limit the bandwidth of the noise, but the power spectral density of the noise is still constant in the filter passband, so it can still be called white. In the receiver, the noise is added to the received signal and is therefore termed additive white Gaussian noise (AWGN).

In the simulation, a Gaussian noise generator is used that generates noise of unit variance and then is scaled to the variance required for the desired signal-to-noise ratio.

### Receive Path

The receive path (receiver) is also shown in Figure 1. Raised-cosine-filtered samples are fed into the sync detector, which looks for the sync word that occurs at the beginning of the slot. The sync detector looks for this sync word over a 4-symbol window, starting two symbols prior to the expected sync point. When the data matches the proper slot sync word, the data is fed into the SRC filter. This filter is the same as the transmit chain SRC filter described on page 36, except that the receive filter performs 4X decimation.

After it is fed through the SRC filter, the data is input to a channel equalizer. As shown in Figure 2, the channel equalizer can be turned either on or off under command of the cellular base station. The channel equalizer is not included in this simulation and is the subject of a separate paper [9].

The delay detection process, also called differential decoding, is the inverse of the differential encoding process in the transmitter. The delay detector computes the amount of phase change between two successive raised-cosine-filtered maximum-effect points. This can be shown easily with the exponential notation for complex numbers. Let  $A * \exp(j * \pi / 2)$  be the current point and  $B * \exp(j * \pi / 4)$  be the previous point. Now multiply the current point by the complex conjugate of the previous point:

$$A * \exp(j * \pi / 2) * B * \exp(-j * \pi / 4) = A * B * \exp(j * \pi / 4)$$

The result is an exponential whose angle is the phase change between the previous and the current points. Because it is the phase change that contains the information bits, the magnitude can be disregarded.

The deinterleave function recombines a frame of speech data from data received from two consecutive receive slots. As discussed in the transmit chain description, the data is interleaved to minimize susceptibility to burst errors. At this time, the data is divided back into the encoded class 1 (cc0 and cc1) bits and the unprotected class 2 bits. The class 1 bits are then fed into a convolutional decoder while the unprotected class 2 bits are held to recombine with the class 1 bits once decoded.

The convolutional decode is performed via the Viterbi algorithm. A two-dimensional array is built that is 89 (the number of bits input to the encoder) columns wide and 32 (the possible number of states of the encoder) rows high. This algorithm calculates the probability of possible paths through the array (which represent the sequence of states through which the encoder would have passed). This probability is added to the cumulative probabilities for each of the possible preceding states to give a cumulative probability for a given trellis position. Then, given that the beginning and ending states of the convolutional encoder are 0 (0 is the initial state and five tail bits of 0 force it back to state 0), the path of maximum probability is selected by tracing through the array from ending state to beginning state. With the path through the trellis known, the input bits are easily obtained. The path of maximum probability should produce the original encoded bit stream, even in the presence of low bit errors.

The CRC value and the 12 most perceptually significant bits are extracted from the decoded class 1 bits. A CRC is recalculated on these 12 bits and compared against the received CRC. This is done to detect the



presence of errors in these 12 bits. If the CRCs match, the received VSELP speech parameters are sent to the VSELP decoder. If they do not match, a state machine (IS-54 para. 2.2.2.2.3.2.) is employed for handling the errors. This state machine stores the last good set of speech parameters for use in cases of repeated CRC errors. The received speech parameters are then fed into the VSELP decoder for speech synthesis.

## Using the Simulation

One of the goals in developing this simulation was to ensure that it is portable across different computing platforms. To this end, every attempt was made to use only ANSI-C compatible calls and syntax. The code was originally developed using Borland C++ 3.1 running on 486/33 ISA PCs. It was tested and modified to make it compatible with the Microsoft Visual C++ 1.0 and Zortech C++ 3.0 compilers, which support ANSI-C compliance.

To run the simulation, a command file, IS54SIM.PRM, is utilized to pass all required information to the program. Additionally, another file, SRC\_FILT.DAT, is required and contains the square-root cosine filter coefficients necessary for the simulation. These files and the simulation program must all reside in the same working directory.

The format of the command file is simple. It is an ASCII file that contains four lines:

1. The desired SNR
2. The assumed vehicle speed
3. The carrier frequency (used in Fading model)
4. The filename for the input speech data that has already been VSELP processed (This file should also be in the working directory.)

The SRC\_FILT.DAT file is also an ASCII file, where each line is a coefficient used by the SRC filter.

After running the simulation (by typing the program name on the system command line), there are seven output files produced, all of which reside in the current working directory. These files are summarized below.

IS54SIM.OUT	An ASCII-Hex version of the 193-bit VSELP data recovered for each frame
RAWTXBIT.OUT	An ASCII-Hex version of the 324-bit formatted TDMA slot prior to transmission
CLTXBIT.OUT	An ASCII-Hex version of the 89 class 1 bits and 82 class 2 bits for the transmit slot
CLRXBITS.OUT	An ASCII-Hex version of the 89 class 1 bits and 82 class 2 bits recovered in the receive slot. Each line of receive data (one per slot) is appended with the current CRC error state (0–7).
RAWRXBITS.OUT	An ASCII-Hex version of the 324-bit formatted TDMA slot prior to decoding. Each line of receive data (one per slot) is appended with the current CRC error state (0–7).

By examining these output files, a user can determine the performance of an IS-54 transmission under varying levels of SNR (degradation in the channel). This program also outputs the number of received frames with valid CRC, the number of frames with invalid CRC, and the bit error rates for each field for CRC-valid frames.

The simulation was compiled and run on IBM-compatible PCs using several compilers. The simulation runs three to six slots per second on a 486DX–33MHz PC.

### Code Availability

The associated program files are available from Texas Instruments TMS320 Bulletin Board System (BBS) at (713) 274-2323. Internet users can access the BBS via anonymous ftp at *ti.com*.

### References

1. *Cellular System: Dual-Mode Mobile Station – Base Station Compatibility Standard*, IS-54B, Telecommunications Industry Association, April 1992.
2. Chishtie, Mansoor A., “U.S. Digital Cellular Error-Correction Coding Algorithm Implementation on the TMS320C5x”, *Telecommunications Applications With the TMS320C5x DSPs*, Texas Instruments Incorporated, 1994, pp. 63–75.
3. Choong, Yong F., *Convolutional Decoder for the IS-54 Error Protected Speech Codes*, Digital Signal Processing Branch, Semiconductor Process and Development Center, Texas Instruments Incorporated, 1992.
4. Jakes, William C., Jr., *Microwave Mobile Communications*, John Wiley & Sons, New York, New York, 1974, pp. 70–76.
5. Proakis, John G., *Digital Communications*, McGraw Hill, New York, New York, 1989.
6. Choong, Yong, and Chishtie, Mansoor A., *Convolutional Encoder for IS-54 Voice Channel C program*, Texas Instruments Incorporated, 1992.
7. Chishtie, Mansoor A., *Viterbi Decoder Algorithm for IS-54 SACCH Control Channel C program*, Texas Instruments Incorporated, 1991.
8. Hartman, Matt, *C Language Version of VSELP Speech Coder*, Systems Research Laboratories, Chicago Corporate Research and Development Center, Motorola, Incorporated, 1990.
9. Hoole, Elliot D., “Channel Equalization for the IS-54 Digital Cellular System With the TMS320C5x”, *Telecommunications Applications With the TMS320C5x DSPs*, Texas Instruments Incorporated, 1994, pp. 177–187.

## **IMPORTANT NOTICE**

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

**TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.**

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.