DIGITAL DECODING IN RANGE SAFETY RECEIVERS

Mark Dapore
Electrical Engineer
Cincinnati Electronics Corporation
Aerospace Division
Cincinnati, Ohio 45241

ABSTRACT

Command Destruct Receivers are used on all launch vehicles for the purpose of initiating flight termination. The up-link command signal is a UHF carrier frequency modulated with Inter-Range Instrumentation Group (IRIG) audio tones. Typically three commands are used to control and terminate the flight: ARM, DESTRUCT, and OPTIONAL command. The combination and sequence of specified IRIG tones determines the command to be issued.

The decoder portion of the command destruct receiver determines the presence and sequence of IRIG tones. Through the use of a microprocessor and digital signal-processing-technology, Cincinnati Electronics has designed and developed a digital decoder which detects four IRIG tones simultaneously. The digital decoder provides stabilities and repeatabilities unattainable with analog decoders. Reliability and reproducibility are also superior when using the digital approach through the elimination of select and variable components. This improved performance of the decoder is a result of crystal controlled digital filtering and a constant false-alarm rate provided by the microprocessor. The digital filtering is achieved through Finite Impulse Response digital filters.

There are a number of additional features provided by the digital decoder. Self-testing of portions of the receiver and correcting for temperature drift can be performed by the decoder. The digital decoder can have its command sequences reprogrammed in the field allowing separate control and termination of each vehicle in a multiple vehicle launch. Also, acceptance test data, date of manufacture, total time unit has been powered up, and other data can be stored inside the Command Destruct Receiver and recalled when needed.

INTRODUCTION

Every spacecraft launch vehicle and many missile test vehicles require a flight-termination subsystem for the purpose of range safety. If the flight of the vehicle were to deviate from its intended path and violate predetermined safety limits, the Range Safety Officer would issue a command to the vehicle to terminate its flight. Through the efforts of the Range Safety Group of the Range Commanders Council and its ad hoc Committee for Flight Termination Systems, a degree of standardization in system definition has been achieved over the years. Typically, standard IRIG audio tones are frequency modulated on a UHF carrier in a particular sequence to issue the flight termination command.

The piece of equipment on board the vehicle that initiates flight termination is the range safety receiver which receives, demodulates, and decodes the command message. This paper describes a microprocessor-based method of audio tone detection and message decoding recently developed at Cincinnati Electronics Corporation. This concept is presently used in several models of Flight Termination Receivers.

COMMAND FORMAT

The command format consists of 4 IRIG range-safety channels: Channel 1 at 7500 Hz, Channel 2 at 8460 Hz, Channel 4 at 10760 Hz, and Channel 5 at 12140 Hz. Channel 1 and 5 simultaneously represent an Arm Command while Channel 2 and 5 simultaneously represent the Optional Command. Once armed, only Channel 1 is required to retain the arm. An Arm Command with the addition of Channel 2 and the removal of Channel 5 results in a Destruct Command. Channel 4 is the check channel.

The tones must be present a minimum of 15 milliseconds to be recognized as a command and will continue to be recognized as long as the tones are present. Each tone filter has a ±1% minimum 2 dB bandwidth and a ±4% maximum 20dB bandwidth requirement.

HARDWARE

Because Range Safety Receivers are airborne equipment, small size and low power are high priorities. Throughout the design and development of the digital decoder, chip count as well as power dissipation were important considerations. Another important factor in the design which improves the reliability as well as reducing labor required to align the decoder was to eliminate the use of select and trimmable components. Also, with no hand alignment necessary, all decoders built are identical in performance. The block diagram for the decoder is shown in Figure 1.

The audio signal from the receiver portion of the radio is passed through the anti-aliasing filter to reduce signals outside of the band of interest. As shown in figure 1, the signal is then digitized with an A/D converter and fed to the microprocessor which filters and decodes the signal. Signal power in each of the IRIG tones is calculated by the microprocessor. If the correct tones are present and their magnitudes exceed the threshold, the corresponding command is issued. The digital decoder detects 4 IRIG tones simultaneously as well as performing signal strength telemetry corrections. The filtering and detection of tones is achieved with software algorithms.

The anti-aliasing filter rejects signals outside of the band 7500-12140 Hz. All images and aliases are eliminated by the four pole anti-aliasing filter. Because all the IRIG tones to be detected fall within the passband, the anti-aliasing filter does not compromise the stability or accuracy of the digital filters.

The A/D converter digitizes the audio waveform at a predetermined sampling rate and supplies the digital samples to the microprocessor. The A/D converter used is an 8 bit flash converter with a conversion time of 100 nanoseconds which eliminates the need for a sample-and-hold amplifier. The 8 bits provide a 48 dB resolution of the audio waveform. The converter is CMOS to reduce current consumption to a minimum value.

The central controller of the decoder is the microprocessor. The microprocessor has an on-board multiplier and adder specifically for digital signal processing as well as on-board RAM for storage of samples. The 32 bit accumulator provides 192 dB of dynamic range for the digital filters. A crystal controlled clock within the microprocessor controls the sampling rate as well as the execution speed of the digital signal-processing-routines. To determine the presence of specific tones the microprocessor retrieves the audio samples from the A/D converter and applies them to the digital filter algorithm. If the correct tones are present, the appropriate command is issued through fail-safe output circuitry. The complete system is single-point fail safe, meaning that the failure of any component, hardware, or software will not cause an inadvertent output.

The sampling rate is controlled precisely by a software programmable timer. The 5 MHz crystal-controlled system clock is divided by 371 to maintain an exact sampling frequency of 13477 Hz. The sampling frequency controls the center frequency and cutoff frequencies of the filter. Because the sampling frequency is crystal controlled, the filter passbands drift only ±1 Hz over temperature variations, compared to ±100 Hz in analog systems.

ADDITIONS OF EXTRA FEATURES

With an analog multiplexer preceding the A/D converter the digital decoder can become a more versatile device. The signal strength telemetry signal and a temperature sensor can be monitored and adjusted to compensate for temperature effects. The resulting telemetry curve is accurate to within ±1 dB over the RF carrier input range of −107 to −53 dBm. Various analog points throughout the radio are monitored to provide a self-check for the radio. Power supplies are monitored and corrected for drift over temperature.

An EEPROM is added to the system to provide the capability of secure codes. The secure code is entered into the system through a code insertion device and is stored in the EEPROM. Different codes may be entered on each receiver for a multiple vehicle launch, enabling each Command Destruct Receiver to be controlled separately while using only one carrier frequency. The EEPROM is where date of manufacture, serial number, and acceptance test data for the unit are stored and can be recalled whenever desired. A log of running time of the unit is also stored in the EEPROM and updated by the microprocessor automatically every few seconds.

DIGITAL SIGNAL PROCESSING

Several digital-signal-processing methods were considered for use in IRIG tone detection. The methods include Infinite Impulse Response (recursive) and Finite Impulse Response (non-recursive). Because Infinite Impulse Response filters pose a possible instability problem due to their recursive nature, the development focused on Finite Impulse Response filters.

Finite Impulse Response filters are attractive in relation to IRIG tone detection because computations need not take place between each sample. Samples can be stored in a “block” and and the power in the signal calculated only when the block is full.

Because the four IRIG tones are frequency unrelated (due to the greatest common denomonator of the tones being only 20 Hz), the computational timesaving techniques of the Fast Fourier Transform (FFT) can not be used. The Discreet Fourier Transform technique was investigated and resulted in a number of important findings. Sampling frequency and number of samples taken is directly related to transition width of the filter. A reduction of the sampling frequency by a factor of 2 results in a reduction of the transition width of the filter by a factor of 2. Also a twofold increase in the number of samples taken reduces the transition width twofold. Therefore, selection of the minimum sampling rate has two desired effects: the transition width of the filter is reduced, and the number of samples needed to be processed is reduced.

FOURIER SERIES APPROACH

The Fourier Series Approach to designing FIR filters (1) is achieved by using coefficients from equation 1.

$$c(m) = \int_{a}^{b} A \cos m \, Pi \, v \, dv \qquad (1)$$

where m is a number from 0 to N
N is the number of coefficients in the filter
v is the normalized frequency
a is the normalized low cutoff frequency
b is the normalized high cutoff frequency
A is the magnitude of the filter response
Pi is 3.1415

therefore

$$c(m) = \frac{A \sin b \, m \, Pi}{m \, Pi} - \frac{A \sin a \, m \, Pi}{m \, Pi} \quad \text{for } m=-N/2 \text{ to } +N/2 \qquad (2)$$

The coefficients are symetrical around c(0) which saves on the amount of memory needed in the system. Only half of the coefficients are actually stored in the system but they are used twice in the calculation of a filter response. The Fourier Series Approach yields filters with flatter passbands and narrower transition bands than the Discreet Fourier Transform or the Fast Fourier Transform. It also results in filters which provide the best response for the amount of computation time available.

A filter design program using the Fourier Series Method was developed at Cincinnati Electronics to aid in evaluation of the filters. The program designed filters based on the cutoff frequencies, sampling frequency, and window parameters entered by the user. After the design was complete the program plotted the response of the filter. A program of this type greatly aids in the evaluation and the design of digital filters. Figure 2 is a plot from the computer-aided design program.

SAMPLING RATE

The four tones which require detection are shown below.

IRIG TONE	FREQUENCY
Channel 1	7500 Hz
Channel 2	8460 Hz
Channel 4	10760 Hz
Channel 5	12140 Hz

Nyquist Sampling Theorem states that to prevent aliasing the sampling rate must be at least twice the highest sampled frequency. In this instance the highest frequency is 12140 Hz which would result in a sampling frequency of at least 24280 HZ. However, the bandwidth of frequencies to be filtered is only 12140-7500 or 4640 Hz. Therefore, it is desired to pick a sampling frequency that is at least 2 x 4640 and has all image or "alias" responses outside of the band from 7500 to 12140 Hz. The sampling frequency selected is 13477 Hz which is slightly higher than Channel 5. This sampling frequency provides a computational time savings of 4 times compared to a sampling frequency of 24280 Hz.

IMAGES

Because of the sampling rate chosen, a number of image frequencies are produced outside of the 7500 to 12140 Hz band. These images or aliases are filtered out with the anti-aliasing filter. The alias tones both above and below the band are shown below.

IRIG TONE	FREQUENCY	LOW ALIAS	HIGH ALIAS
Channel 1	7500 Hz	5977 Hz	19454 Hz
Channel 2	8460 Hz	5017 Hz	18494 Hz
Channel 4	10760 Hz	2717 Hz	16194 Hz
Channel 5	12140 Hz	1337 Hz	14814 Hz

The two critical alias tones are the low alias for Channel 1 (5977 Hz), and the high alias tone for Channel 5 (14814 Hz). These tones must be filtered out by the anti-aliasing filter while allowing the four channel frequencies to pass. A 4-pole, 0.32-cubic-inch anti-aliasing filter is used to provide the correct band selectivity.

WINDOWS

A number of windows were investigated in an effort to suppress sidelobe response to a minimum level. Finite Impulse Response filters have a −13 dB sidelobe because of Gibbs phenomenon (2). By applying a window to the filter the sidelobes can be suppressed to very low levels. The windows investigated are the Hamming, Hanning, Blackman, Kaiser, Rectangular, and Hamming Deviation.

The rectangular window has sidelobe rejection of less than 20 dB which does not meet the filter performance required. The Hanning window and Hamming window provide excellent sidelobe rejection, (near 60 dB), but widen the transition bands of the filter beyond acceptable limits. The Hamming Deviation Window is a shaped Hamming window. By changing the shape of the Hamming window the user has the ability to trade transition bandwidth for sidelobe rejection level. The most desirable results were obtained with 45 dB sidelobe rejection and a transition band of 100 Hz. The Hamming Deviation window when applied to the Fourier Series Approach digital filter resulted in a response which has the correct passbands, stopbands, and sidelobe rejection required. See Figure 2.

The Kaiser window (3) uses Bessel functions to shape the samples and provide side lobe rejection. The Kaiser window also exhibits satisfactory performance with the ability to trade transition bandwidth for side lobe rejection level. See Figure 3. The windows produce no additional computation time because the window values are multiplied by the coefficients when designed and stored in the decoder as a single coefficient. Both the Kaiser window and Hamming Deviation window have been used in the decoder with virtually identical performance.

SIGNAL PROCESSING SOFTWARE

The signal-processing software is implemented to detect IRIG tones 1, 2, 4, and 5. Tone presence is determined when a filter output exceeds the detection threshold. The software was written in assembly language and is approximately 2K words in length.

The top level flowchart for the signal-processing routine is show in Figure 4. After system power-up the microprocessor initializes and checks the decoder. All command outputs are off during initialization and self-check. The RAM and ROM of the decoder are checked to insure that all parts of the system are functional. The timer, which controls the sampling rate, is next initialized. The timer initialization involves programming two of its internal counters to provide a 3-pulse string every 74.2 microseconds. The three-pulse string causes the A/D converter to take a samples at the rate of 13477 per second. The third counter in the timer is initialized to act as a “watchdog”. This counter will reset the microprocessor if the failure of a component causes the system to be caught in an indefinite loop. After the timer has been initialized, the interrupt is enabled to allow the collection of audio samples. The initialization procedure is completed approximately 550 microseconds after power-up.

INTERRUPT ROUTINE

The interrupt-routine flowchart is shown in Figure 5. The interrupt routine is entered after the A/D converter takes a sample and strobes the interrupt pin on the microprocessor. The microprocessor reads the sample from the A/D converter and converts it from a 1's complement to a 2's complement representation. The sample is then stored in the 50-sample buffer and control returns to the interrupted program. When the buffer is full, the buffer-full flag is set.

A full buffer flag directs the program to enter the filtering portion of the routine. The filter routine program flow is illustrated in Figure 6. The filters use 200 samples to provide the desired 2 dB and 20 dB responses. However, because of the limited RAM available onboard the microprocessor (144 words), only 50 samples are processed at one time and a matrix set up to keep a record of the results. The matrix used is shown in figure 7. Each IRIG tone will have a matrix in RAM. The output of the filter at any one time is simply the result of adding the diagonal of the matrix. This method reduces the number of samples in the microprocessor at any one time to 50 which erases the requirements for additional (external) RAM. Because the calculations occur every 50 samples, an output for each filter is available at 3.71 millisecond intervals.

The noise power in the audio signal is calculated by filtering the audio samples through a FIR filter centered at 9200 Hz. There are no desired signals at this frequency so the output of this filter is a representation of the noise power. The noise representation is used to raise or lower the IRIG tone-detection threshold. With poor signal-to-noise ratios the threshold is raised and with good signal-to-noise ratios the threshold is lowered to achieve an extremely low constant False Alarm Rate. The noise FIR filter response is shown in Figure 8.

CALCULATIONS OF MAGNITUDES

Presence of a specific tone is determined by comparing its magnitude to the computed threshold. Common practice to determine magnitude is to calculate an in-phase and quadrature channel and determine the magnitude from the following:

$$\text{Mag}=[(\text{In-Phase})^2 + (\text{Quadature})^2]^{1/2} \tag{3}$$

Unfortunately, for a 200 sample filter the in-phase calculation requires 200 multiply-accumulate operations, and the quadrature calculation requires another 200 multiply-accumulate operations. Also, 400 coefficients must be stored in the ROM for calculations.

Another approach was discovered which requires calculation of the in-phase channel only. The output of a series of n in-phase calculations is represented by the following:

$$\text{output} = M \sin [a + (2\,Pi\ x\ f)/f_s\ ] \text{ for } x=0 \text{ to } n \tag{4}$$

where f_s = Sampling frequency
f = Frequency applied to digital filter
Pi = 3.1415

The magnitude of the applied waveforms can be determined by any two consecutive outputs designated Out 1 and out 2. See the appendix for the derivation of equations which lead to determination of the magnitudes. By use of only the in-phase channel, computational time is saved as well as the memory required to store all of the quadrature coefficients. When the magnitude of a tone exceeds the detection threshold it is recognized as a valid tone and used in the command detection process.

RESULTS AND CONCLUSIONS

The digital decoder was breadboarded and tested and met all performance requirements for range safety decoders. Measured performance values are shown below.

Test	Measured Value	Requirement
2 dB Bandwidth	±1.7%	±1% min.
20 dB Bandwidth	±2.6%	±4% max.
Deviation Sensitivity	±10 KHz	±4 khz min.
Pickup Time	1-12 ms. *	15 ms. max.
Dropout Time	1-13.5 ms. *	50 ms. max.
Ajacent Channel Rejection	45 dB	35 dB min.
Power Consumption	1.7 Watts	----
Size	4" x 5" x 0.4"	----
Alignment	None	----

* Software programmable

It is important to note that the electrical performance parameters can be changed simply by modifying the software. Also, the command format and the IRIG tones to be filtered can be modified easily with software changes.

The development and testing of the digital decoder has verified that the performance, versatility, and reliability are clearly superior compared to analog type decoders. Finite Impulse Response filters have been shown to provide excellent filtering performance as well as superior noise immunity and a very low, constant, false-alarm rate. Select and trimmable components are eliminated which erases the need to align the decoder resulting in reduced production time, improved reliability, improved performance, and reduced cost. Work is already being performed to reduce the size and power of the decoder by 75% through the use of surface-mount components, hybrids, and CMOS VLSI.

REFERENCES

1. Rabiner, Lawrence - Gold, Bernard, THEORY AND APPLICATION OF DIGITAL SIGNAL PROCESSING, Prentice-Hall Inc, Edgewood Cliffs, New Jersey, 1975, page 88.

2. Stanley, William, DIGITAL SIGNAL PROCESSING, Reston Publishing Company, Reston, Virginia, 1975, pages 211-221.

3. Kaiser, J, "Design Methods for Sampled Data Filters", Proceedings of First Allerton Conference on Circuit and System Theory, November 1963, pages 221-236.

Appendix

Derivation of magnitude

Two consecutive outputs of the in-phase calculation are given by

$$\text{Out1} = M \sin a \qquad (5)$$
$$\text{Out2} = M \sin (a + (2\,\text{Pi}\, f / f_s)) \qquad (6)$$

where M is the magnitude of the sine wave
f is the center frequency of the filter
f_s is the sampling frequency

By trigonometric identity sin(x+y) = sin(x)cos(y)+ cos(y)sin(x)

$$M \cos a = \frac{\text{Out2} - M \sin a \cos (2\,\text{Pi}\, f / f_s)}{\sin (2\,\text{Pi}\, f / f_s)} \qquad (7)$$

Since f, f_s , and Pi are constants

$$k1 = \frac{\cos (2\,\text{Pi}\, f / f_s)}{\sin (2\,\text{Pi}\, f / f_s)} \qquad (8)$$

$$k2 = \frac{1}{\sin (2\,\text{Pi}\, f / f_s)} \qquad (9)$$

M cos a becomes

$$M \cos a = k2\ Out2 - k1\ Out1 \quad (10)$$

M becomes

$$M = ((M \cos a)^2 + (M \sin a)^2)^{1/2} \quad (11)$$

Therefore

$$M = ((k2\ Out2 - k1\ Out1)^2 + (Out1)^2)^{1/2} \quad (12)$$

By the use of equation (12) only the in-phase channel need be calculated to determine the magnitude of the applied tone. This saves memory as well as computational time.

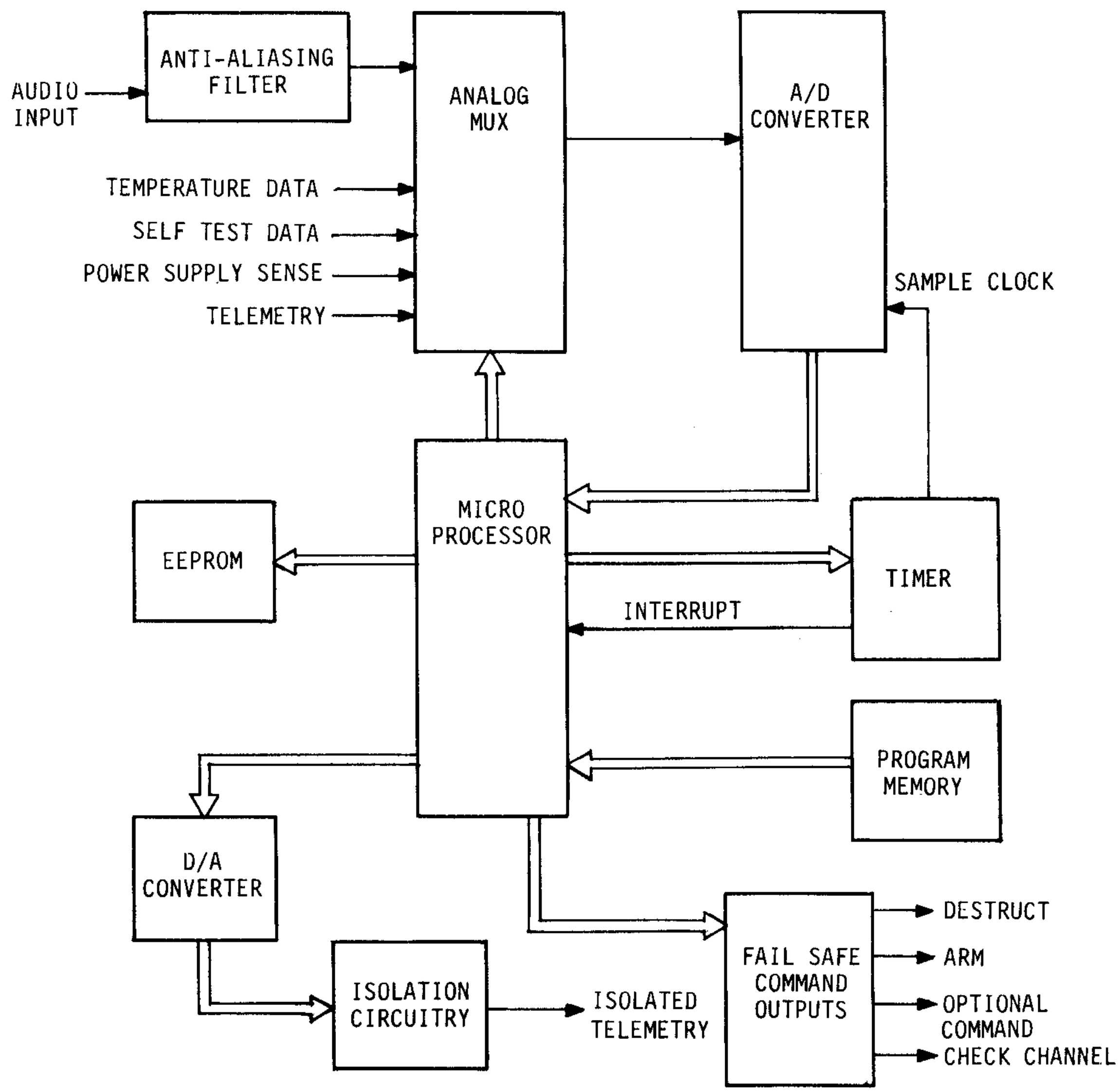

Figure 1. Block Diagram

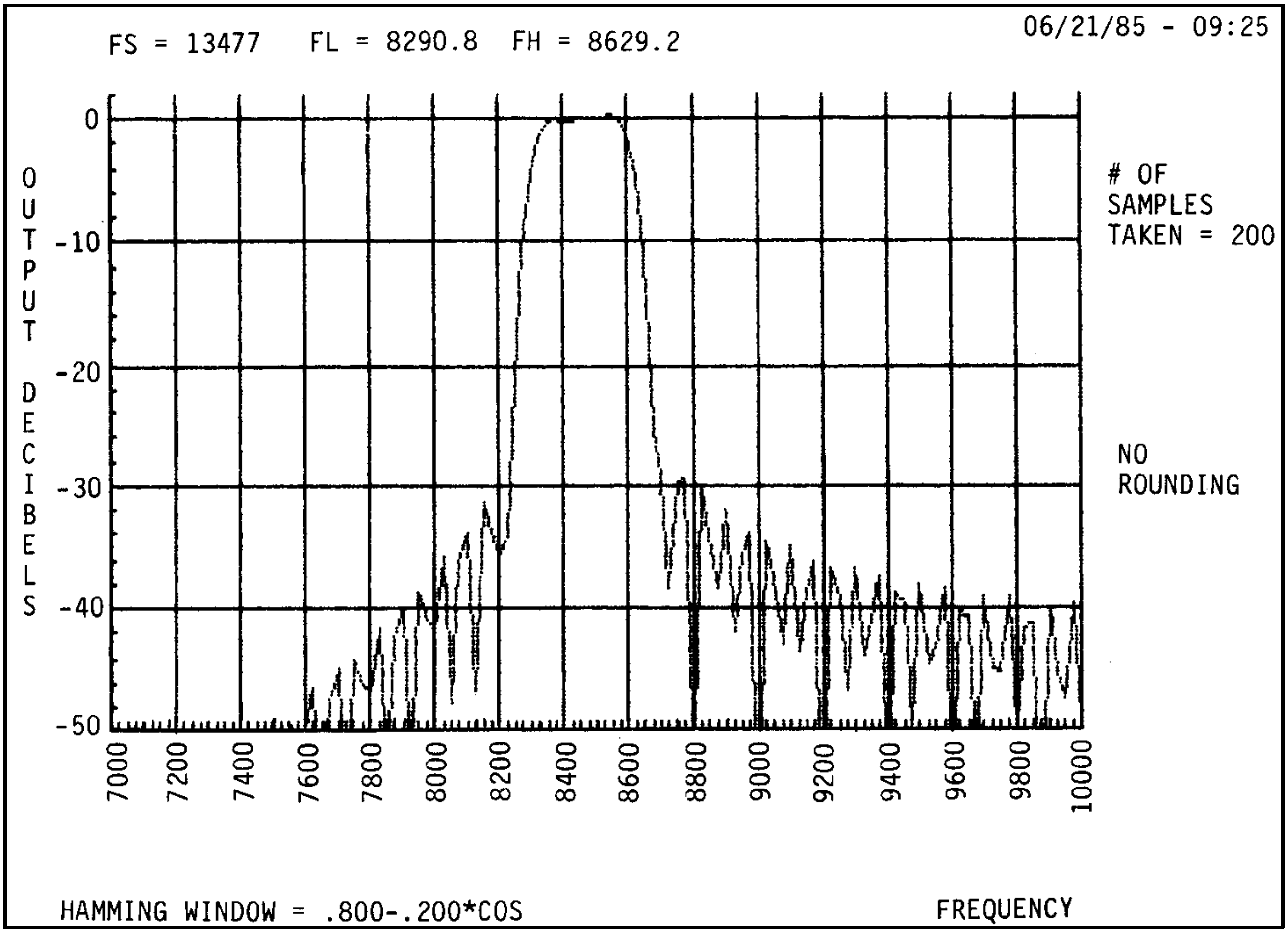

Figure 2. IRIG Channel 2 Filter Response with Hamming Window

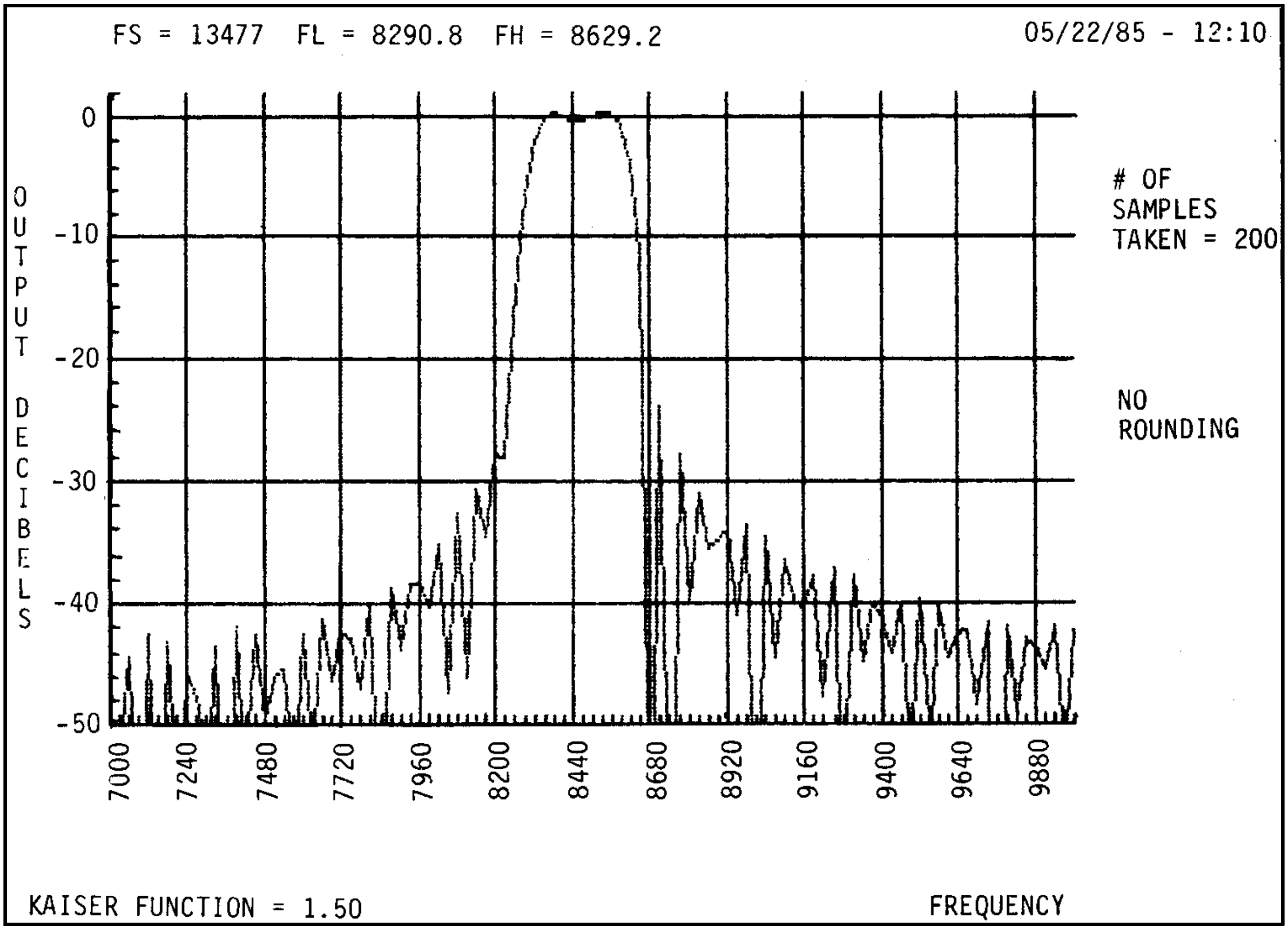

Figure 3. IRIG Channel 2 Filter Response with Kaiser Window

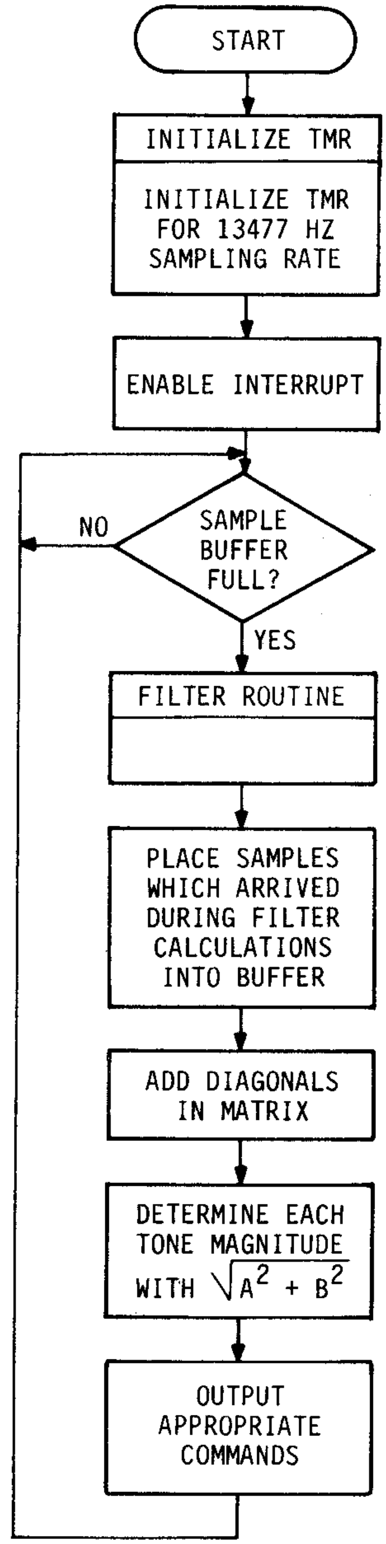

Figure 4. Top Level Signal Processing Flowchart

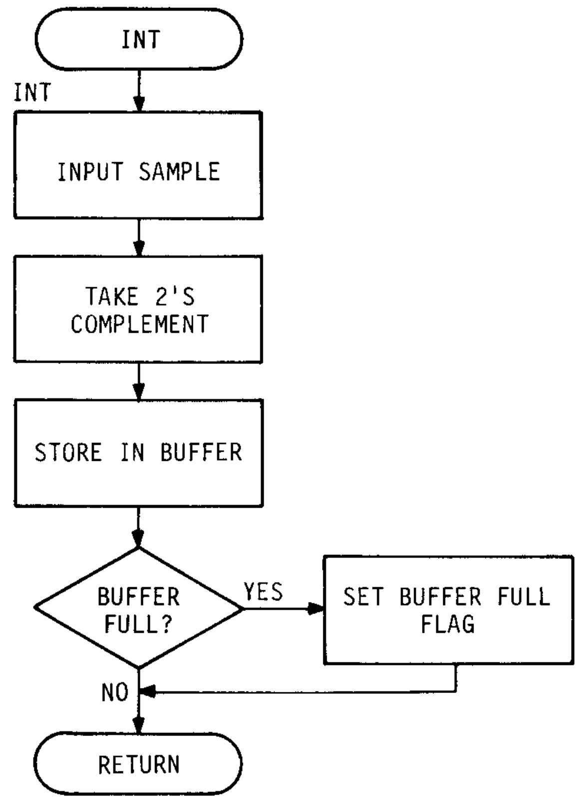

Figure 5. Interrupt Routine Flowchart

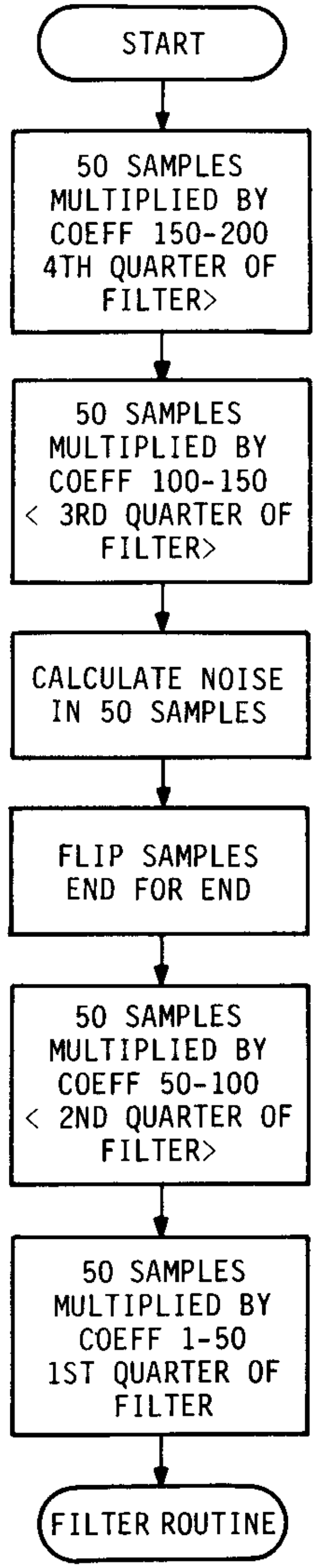

Figure 6. Filter Routine Flowchart

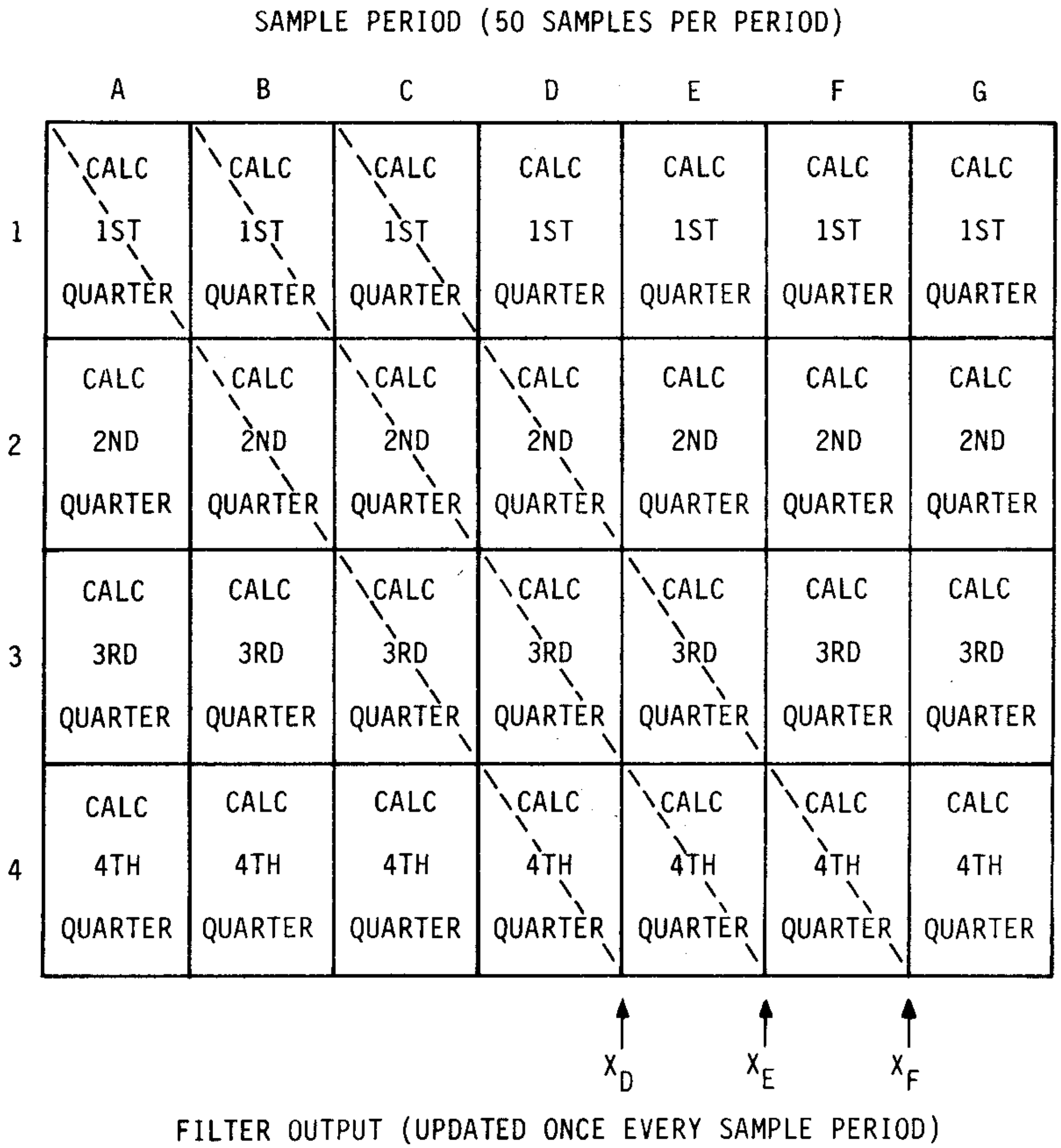

Figure 7. Filter Calculation Matrix

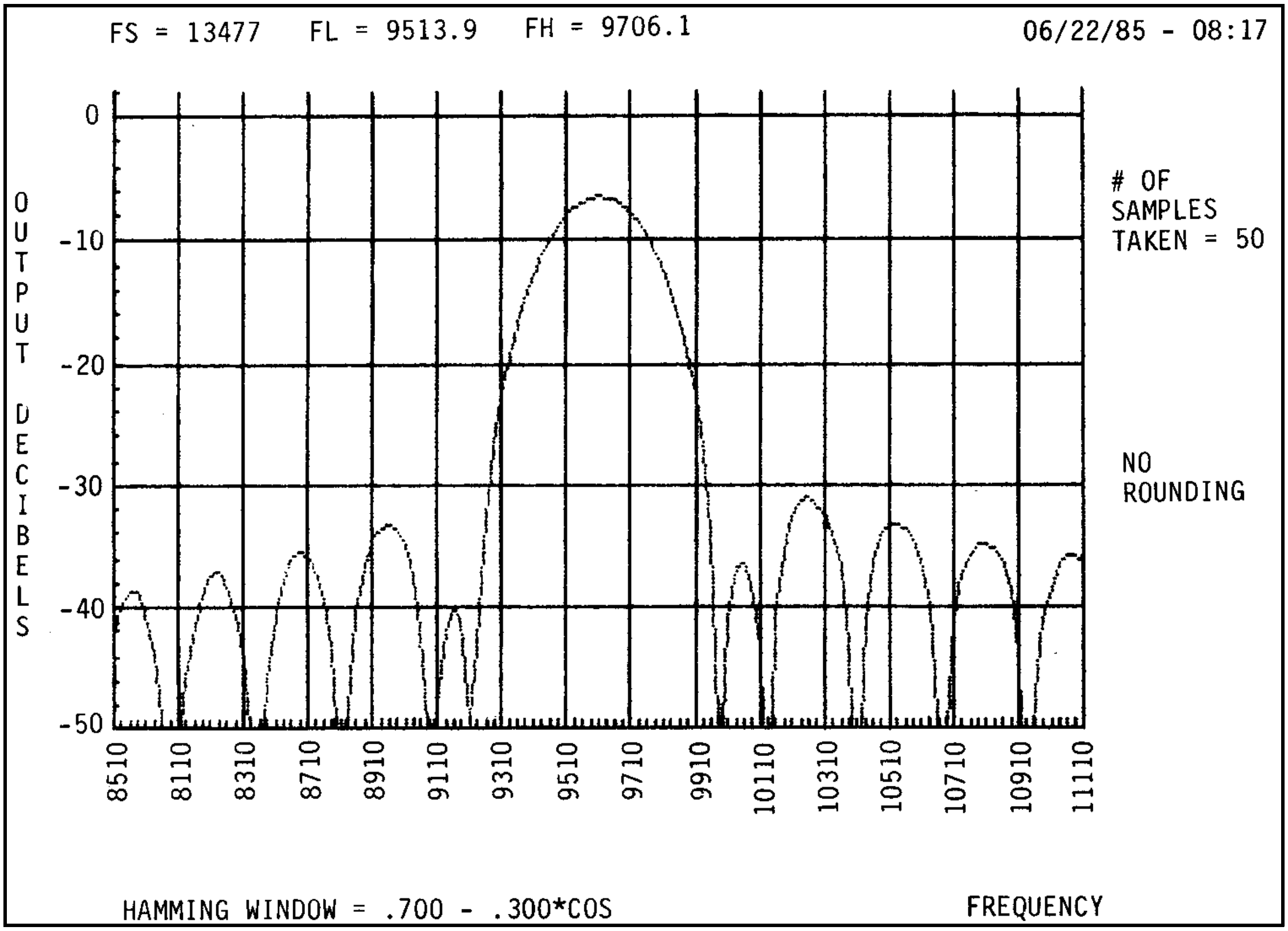

Figure 8. Noise Filter Response