PRESAMPLING FILTERING

D. D. MCRAE and R. C. DAVIS
Advanced Communications Department
Radiation Incorporated
Melbourne, Florida

Summary Sampled data systems often employ lumped-parameter lowpass filters both prior to and following the sampling operation. The purpose of these filters is to reduce the error between the input and output data waveforms. The present paper discusses the effect of presampling filters on the rms interpolation error for two types of sampled data systems and gives some thumb rules for choosing such filters. The two types of sampled data systems considered are: (1) one employing only zero-order hold interpolation, and (2) one employing zero-order hold followed by the best lowpass lumped-parameter interpolation filter. The resulting expressions for rms interpolation error for sampled data systems employing lumped-parameter filters from a detailed time domain analysis are given.

Background In sampled data systems a time continuous data waveform is represented by periodic time samples. The process of reconstructing an estimate of the original waveform from the time samples is called the interpolation process and the difference between the original and estimated waveforms is called the interpolation error.

It has been recognized for some time that the magnitude of the interpolation error can be affected by use of a linear filter operating upon the original time function before it reaches the sampling device. Although such filtering distorts the wanted signal to some degree, which increases the interpolation error, it also can reduce the interpolation error by removing components which cause aliasing or frequency foldover in the reconstruction process. It has been shown that if the power spectral density of the data to be transmitted is non-increasing the presampling and interpolation filters which minimize the mean-square interpolation error are both ideal sharp cutoff lowpass filters with cutoff at one-half the sampling rate. It can be easily demonstrated that in such a case the removal of the presampling filter will double the mean-square interpolation error. However, the resulting sharp cutoff interpolation filter, by itself, is non-optimum for any data spectra other than ideal sharp cutoff. It follows that the use of an optimum combination of presampling and interpolation filter can never reduce the mean-square interpolation error by two over that obtained by use of the optimum interpolation filter with no presampling filter. Further, if one examines the effect on interpolation error when the presampling filter is restricted to a relatively simple minimum phase filter, the gain available becomes extremely small. This is illustrated in Figure 1 which shows a set of curves of rms error versus sampling rate for first order data, first order presampling filter, and 100% pulse stretching. The parameter on the curves is SR/f_p, the ratio of sampling rate to presampling filter 3 dB frequency. At a given sampling rate then the curves show the variation in rms error for first order data as the first order presampling filter bandwidth is varied. Notice that at a normalized sampling rate of 312, 10% rms error is obtained with no presampling filter and that this is reduced to only 8.4% with the best 1st order presampling filter. For higher order data the improvement over the no presampling filter case is even less than that for first order data.

If little or no improvement is obtained then why bother to use a presampling filter at all? The answer of course is that noise is almost always present along with the data and one would like to suppress the noise without seriously distorting the data. Figure 1 shows that for first order data, a first order presampling filter having a 3 dB frequency down to about one- sixth the sampling rate, can be incorporated before degradation over the no presampling filter case occurs. In the case that the noise frequencies are higher than the significant data frequencies one would like to make the presampling filter bandwidth just as restrictive as possible without seriously degrading the data in order to eliminate the noise. This is just the type information that plots such as that in Figure 1 can give. Various plots like this have been generated for different orders of data in combination with different orders of presampling filters. The data considered have Butterworth spectra and the filters are of the lowpass Butterworth type.

Results Figures 2-5 show the effect of presampling filters on the rms error for first through fourth order Butterworth data with 100% pulse stretching and no interpolation filter. On these figures it is assumed that the sampling rate is chosen such that 10% rms error is obtained with no presampling filter. The curves are plots of the change in rms error (above or below 10%) versus the 3 dB bandwidth (as a fraction of the sampling rate) of first, second, and fourth order Butterworth presampling filters. For the assumption of less than 10% rms error in the no presampling filter case, the curves shrink vertically. Figures 2-5 therefore indicate that in a sampled data system which employs no interpolation filter but uses zero-order hold (100% pulse stretching) a presampling filter can be incorporated at a 3 dB bandwidth of one-fourth the sampling rate with less than 1% degradation in the rms error over the no presampling filter case. Note that this assumes that 10% or less rms error is obtained prior to filtering but is insensitive to data order and is true for at least first through fourth order presampling filters.

Let us next consider the case where a lumped-parameter interpolation filter is used with the zero-order hold. Assume that the designer chooses the interpolation filter as the best one for the given sampling rate with no presampling filter (corresponding to the knee of a curve at the given sampling rate for T_h/T_s = 1.0 in a plot like Figure 11). What then is the effect of including a presampling filter on the rms error performance of the system? This is the question we now attempt to answer.

A set of curves for answering the question just posed was generated using a computer program for the rms error expressions given in the Analysis Section to follow. A typical plot of a set of these curves is shown in Figure 6. Here the rms error is plotted versus sampling rate for third order Butterworth data. The best first order interpolation filter in the absence of presampling filtering was chosen for each sampling rate. The parameter SR/f_p appearing on the curves is the ratio of sampling rate to presampling filter 3 dB frequency. At a given sampling rate the curves show the change in rms error as the first order presampling filter bandwidth changes.

From curves, such as those in Figure 6, for first, third, and fourth order data the effect of presampling filtering on rms error was plotted. This is shown in Figures 7-9. Here it is assumed that the sampling rate is such that 5% rms error is obtained in the case that 100% pulse-stretching is used and the best first order interpolation filter in the absence of a presampling filter is employed. The change in rms error (about the 5% figure) is plotted versus presampling filter 3 dB frequency. For smaller rms errors the curves shrink in the vertical direction. These curves show that if the rms error before presampling filtering is 5% or less, a presampling filter with 3 dB bandwidth-at one-half the sampling rate can be incorporated with less than approximately 1% degradation in error performance. As the presampling filter bandwidth is decreased below onehalf the sampling rate the degradation increases rapidly. For rms errors (before presampling filtering) greater than the 5% for which the curves in Figures 7-9 are drawn, the presampling filter 3 dB bandwidth cannot be reduced much below the sampling rate before serious degradation occurs.

The curves of Figures 7-9 are for the case that the best <u>first</u> order interpolation filter is used. Similar results have also been obtained for the case that the best fourth order Butterworth interpolation filter is used. These considerations then indicate that a sampled data system already employing the best interpolation filter for the given sampling rate should not use a presampling filter 3 dB bandwidth below approximately one-half the sampling rate.

Conclusions The resulting expressions from a time-domain analysis of the rms interpolation error in sampled data systems using lumped-parameter filters are presented in the Analysis Section. These expressions take into account presampling filtering, pulse-stretching, interpolation filtering, and an arbitrary time delay on the input data. A computer program for evaluating these expressions has been used to obtain some interesting results concerning presampling filter assignment for certain sampled data systems.

It was found that a presampling filter with 3 dB bandwidth at one-fourth the sampling rate can be used with less than 1% degradation in rms error performance on a sampled data channel whose rms error is less than 10% using only zero-order hold interpolation. It was found that this statement is insensitive to data order and is true at least for first through fourth order presampling filters. As the presampling filter bandwidth is reduced below one-fourth the sampling rate the degradation in rms error performance was found to increase rapidly. This establishes a restriction on how low the presampling filter bandwidth can be when one is attempting to suppress noise frequencies which are higher than the important data frequencies.

Another sampled data system considered was one which uses zero-order hold followed by the best interpolation filter for the given sampling rate. For such a system it was found that a presampling filter 3 dB bandwidth down to approximately one-half the sampling rate can be used before serious degradation of rms error occurs. For presampling filter bandwidths below one-half the sampling rate the degradation increases rapidly.

Analysis The model for a sampled data system employing lumpedparameter filters for interpolating is shown in Figure 10. These systems quite often use a presampling filter for filtering the data prior to the sampling operation. This filter is indicated in Figure 10 as having an impulse response $h_p(t)$. The output of this filter is then sampled at a periodic rate, $1/T_{s.}$ The sampling operation can be represented as impulsive when the sample width is quite narrow but more often than not the sample value is fed to a "hold" circuit resulting in 100% duty cycle samples (pulse-stretching). Consequently the inputs to the postsampling filter (often called the interpolation filter) are 100% duty cycle pulses whose areas are proportional to the corresponding samples. For convenience in the mathematical analysis the sampling operation is considered impulsive and the effect of pulse-stretching is taken into account by an aperture filter which is combined with the interpolation filter and the combination labeled the "postsampling filter" in Figure 10. This postsampling filter has impulse response $h_o(t)$ obtained by convolving the impulse responses of the aperture filter and the interpolation filter. An arbitrary time delay, T_d, is also allowed in Figure 10 to offset the effect of time delays encountered in the filters. The interpolation error is indicated as $\epsilon(t)$.

The problem then is to analyze the mean-square error of this typical system in terms of the sampling rate and filter parameters. This analysis has been carried out entirely in the time domain* and has yielded expressions which are quite general in that they allow evaluation of the rms error taking into account presampling filtering, pulse-stretching,

* In the past the derivation has usually been performed in the frequency domain since the mechanics are easier to visualize this way.

interpolation filtering, and an arbitrary time delay. Unlike many approaches no assumption is made nor necessary that the phase characteristics of the filters have been compensated to give constant time delay across the passbands. If a filter is specified as an n^{th} order Butterworth filter then it has the phase characteristic of an n^{th} order Butterworth filter and any phase distortion as well as amplitude distortion of the data waveform inherent with such a filter is reflected in the rms error evaluated from the expressions.

This analysis is quite long and tedious so only the resulting expressions are included here.[**] The normalized (with respect to data power) mean integral- squared error obtained can be separated into three terms

$$\frac{\widetilde{\epsilon^2}}{R_f(0)} = 1 - E_2 + E_3$$

where E_2 is given by:

Case 1: $T_d \geq T_h$

$$E_2 = 2K_o K_p \omega_o \omega_p \sum_{m=1}^{M} \sum_{n=1}^{N} \sum_{k=1}^{K} a_m b_n c_k \left\{ \frac{e^{-\alpha_m T_d}\left(e^{+\alpha_m T_h} - 1\right)}{(\beta_n - \alpha_m)(\gamma_k - \alpha_m)\alpha_m T_h} - \frac{2\alpha_m e^{-\beta_n T_d}\left(e^{+\beta_n T_h} - 1\right)}{(\beta_n^2 - \alpha_m^2)(\gamma_k - \beta_n)\beta_n T_h} + \frac{2\alpha_m e^{-\gamma_k T_d}\left(e^{+\gamma_k T_h} - 1\right)}{(\gamma_k^2 - \alpha_m^2)(\gamma_k - \beta_n)\gamma_k T_h} \right\}$$

[**] For those interested a detailed derivation is contained in “A Different Approach to Interpolation Error in Sampled Data Systems”, by R. C. Davis, Radiation Incorporated Technical Report No. 24, Radiation Incorporated, Palm Bay, Florida; February 26, 1968.

Case 2: $T_d < T_h$

$$E_2 = 2K_oK_p\omega_o\omega_p \sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{k=1}^{K} a_m b_n c_k \left\{ \frac{1-e^{-\alpha_m T_d}}{(\beta_n - \alpha_m)(\gamma_k - \alpha_m)\,\alpha_m T_h} \right.$$

$$- \frac{2\alpha_m (1-e^{-\beta_n T_d})}{(\beta_n^2 - \alpha_m^2)(\gamma_k - \beta_n)\,\beta_n T_h} + \frac{2\alpha_m (1-e^{-\gamma_k T_d})}{(\gamma_k^2 - \alpha_m^2)(\gamma_k - \beta_n)\,\gamma_k T_h}$$

$$\left. + \frac{1-e^{-\alpha_m (T_h - T_d)}}{(\beta_n + \alpha_m)(\gamma_k + \alpha_m)\,\alpha_m T_h} \right\}$$

E_3 is given by:

$$E_3 = (K_oK_p\omega_o\omega_p)^2\, T_s \sum_{m=1}^{M}\sum_{n_1=1}^{N}\sum_{n_2=1}^{N}\sum_{k_1=1}^{K}\sum_{k_2=1}^{K} a_m b_{n_1} b_{n_2} c_{k_1} c_{k_2}$$

$$\left\{ \frac{2(e^{\gamma_{k_1} T_h}-1)(1-e^{-\gamma_{k_1} T_h})}{(\gamma_{k_1} T_h)^2\,(\gamma_{k_1} + \gamma_{k_2})} \left[\frac{e^{(\gamma_{k_1} + \alpha_m) T_s}}{(\beta_{n_2} - \alpha_m)(\beta_{n_1} + \alpha_m)\left[1-e^{-(\gamma_{k_1} + \alpha_m) T_s}\right]} \right. \right.$$

$$\left. - \frac{2\alpha_m\, e^{-(\gamma_{k_1} + \beta_{n_2}) T_s}}{(\beta_{n_2}^2 - \alpha_m^2)(\beta_{n_1} + \beta_{n_2})\left[1-e^{-(\gamma_{k_1} + \beta_{n_2}) T_s}\right]} \right]$$

$$+ \left[\frac{1}{\gamma_{k_1}\gamma_{k_2} T_h} - \frac{(1-e^{-\gamma_{k_2} T_h})}{(\gamma_{k_2} T_h)^2\,(\gamma_{k_1} + \gamma_{k_2})} - \frac{(1-e^{-\gamma_{k_1} T_h})}{(\gamma_{k_1} T_h)^2\,(\gamma_{k_1} + \gamma_{k_2})} \right]$$

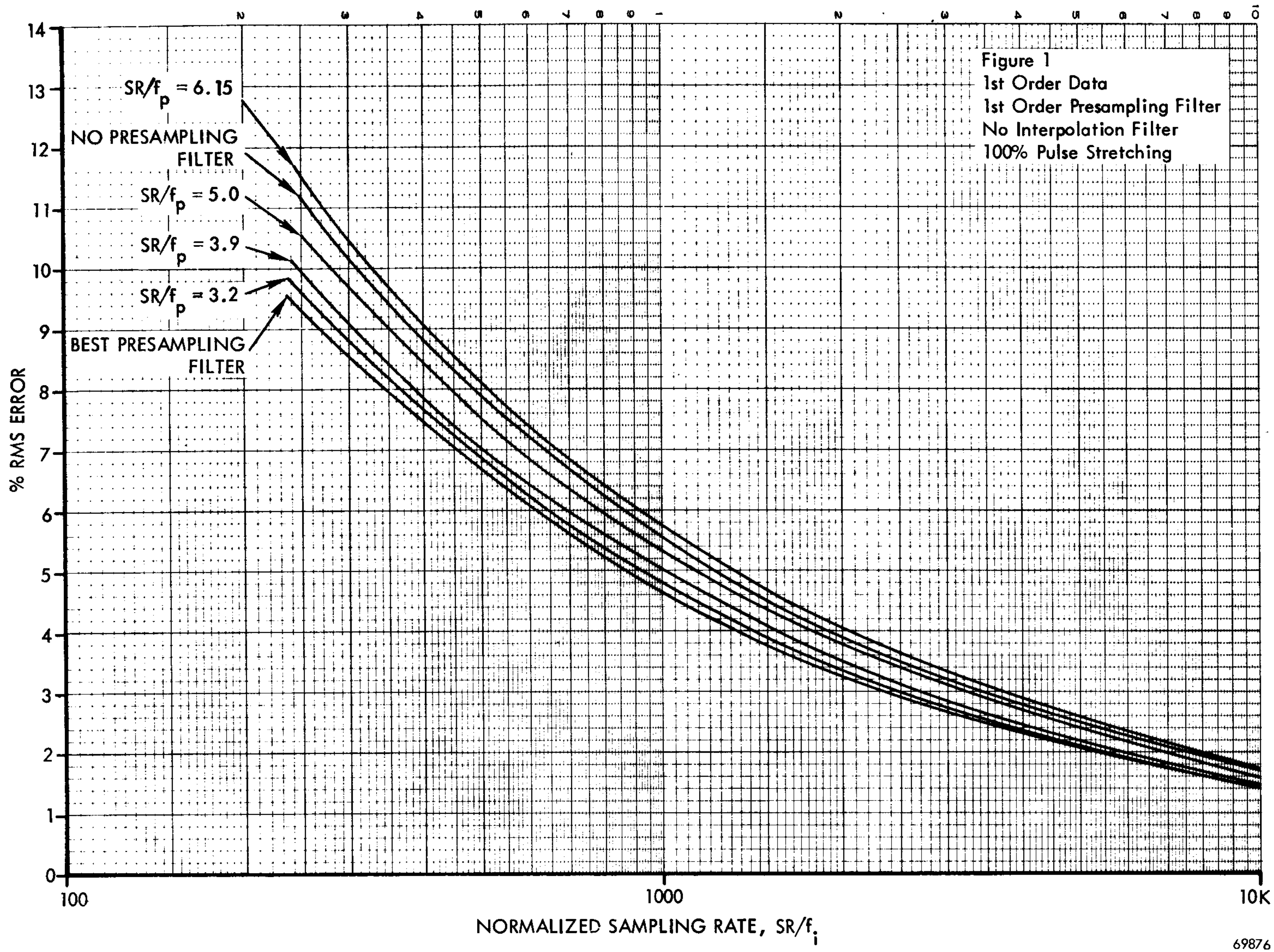
$$\left. \bullet \left[\frac{1}{(\beta_{n_1} - \alpha_m)(\beta_{n_2} + \alpha_m)} - \frac{2\alpha_m}{(\beta_{n_1}^2 - \alpha_m^2)\,(\beta_{n_1} + \beta_{n_2})} \right] \right\}$$

The parameters in the equations are:

T_d = time delay
T_h = duration of hold, $0 < T_h < T_s$. For 100% pulse-stretching, $T_h = T_s$.
T_s = sampling period
K_o = interpolation filter gain
K_p = presampling filter gain
ω_o = interpolation filter "bandwidth"
ω_p = presampling filter "bandwidth"
a_m = residues at left half plane poles of data spectral power density
b_n = residues at poles of presampling filter
c_k = residues at poles of interpolation filter
α_m = left half plane poles of data spectral power density
$-\beta_n$ = poles of presampling filter
γ_k = poles of interpolation filter
M = number of left half plane poles in data spectral power density
N = number of poles in presampling filter
K = number of poles in interpolation filter

The summations are finite sums over the respective poles and the poles and residues are complex quantities in general. Once the pole and zero locations, the pulse- stretching time T_h, the time delay T_d, and the sampling period Ts are all specified, it is only a matter of evaluating the finite multiple sums to determine the mean integral-squared error from these expressions. To do this by hand would indeed be a difficult task for even the simplest filters but a computer is very much suited to such a task. For this reason a FORTRAN language computer program was generated and used to evaluate the expressions for various combinations of filters and data spectra.

A typical set of curves obtained with the computer program is shown in Figure 11. These curves are for data with a third order Butterworth shaped spectrum, a first order interpolation filter (simple RC filter) and no presampling filter. The parameter ω_o/ω_i is the ratio of interpolation filter 3 dB frequency to data 3 dB frequency. The parameter T_h/T_s is the ratio of sample duration to sampling period. If 100% pulse- stretching is employed (which is usually the case), $T_h/T_s = 1.0$. The curves for $T_h/ T_s = 0.01$ approximate the ca se of impul se sample s being f ed to the interpolation filter. It can be seen that the error performance of the sampled data system is significantly modified by the pulse-stretching operation.



14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
% RMS ERROR
100
1000
10K
NORMALIZED SAMPLING RATE, SR/f_i
SR/f_p = 6.15
NO PRESAMPLING FILTER
SR/f_p = 5.0
SR/f_p = 3.9
SR/f_p = 3.2
BEST PRESAMPLING FILTER
Figure 1
1st Order Data
1st Order Presampling Filter
No Interpolation Filter
100% Pulse Stretching

69876

VARIATION OF RMS ERROR VS nTH ORDER BUTTERWORTH PRESAMPLING FILTER BANDWIDTH.
CURVES ASSUME 10% RMS ERROR IN NO PRESAMPLING FILTER CASE.

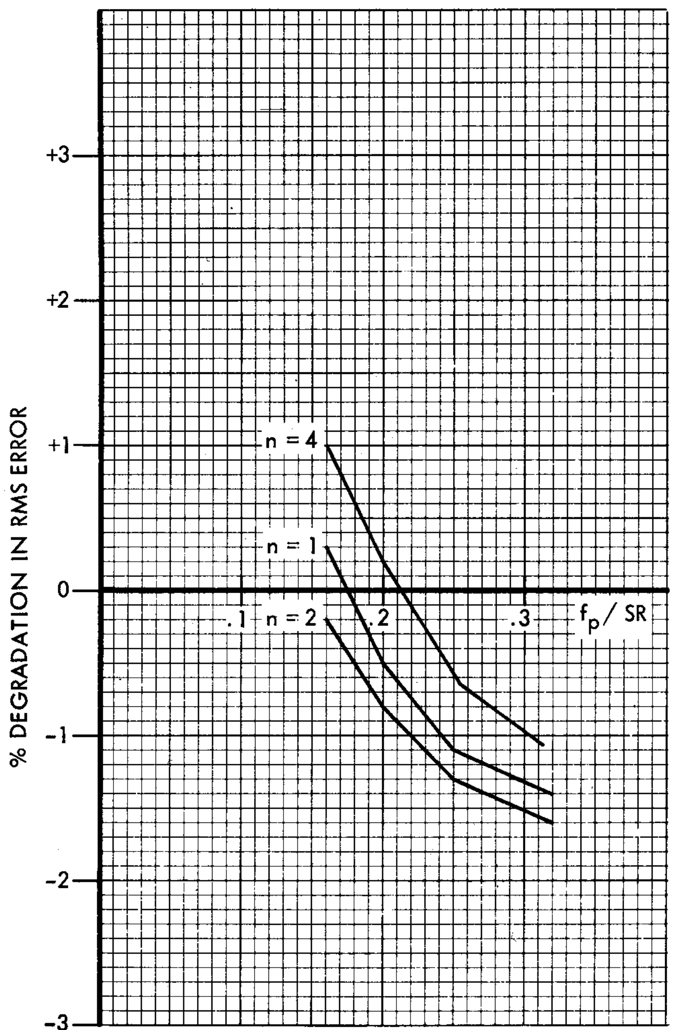


Figure 2– 1st Order Data

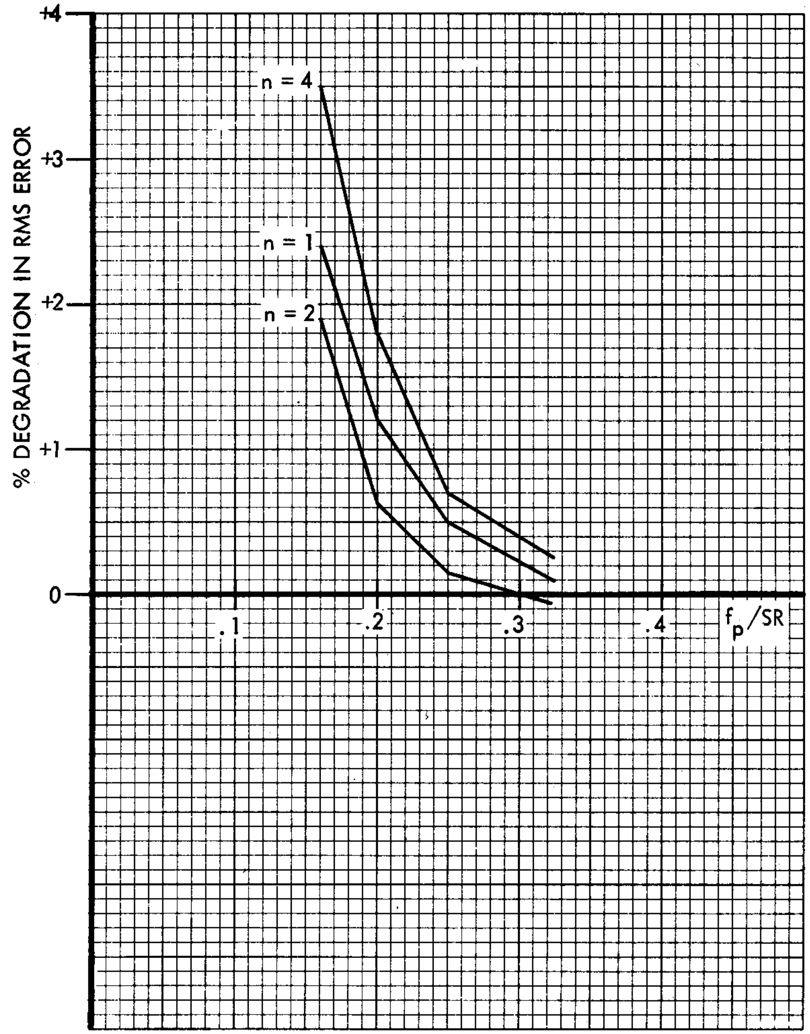


Figure 3– 2nd Order Data

71900

VARIATION OF RMS ERROR VS nTH ORDER BUTTERWORTH PRESAMPLING FILTER BANDWIDTH.
CURVES ASSUME 10% RMS ERROR IN NO PRESAMPLING FILTER CASE.

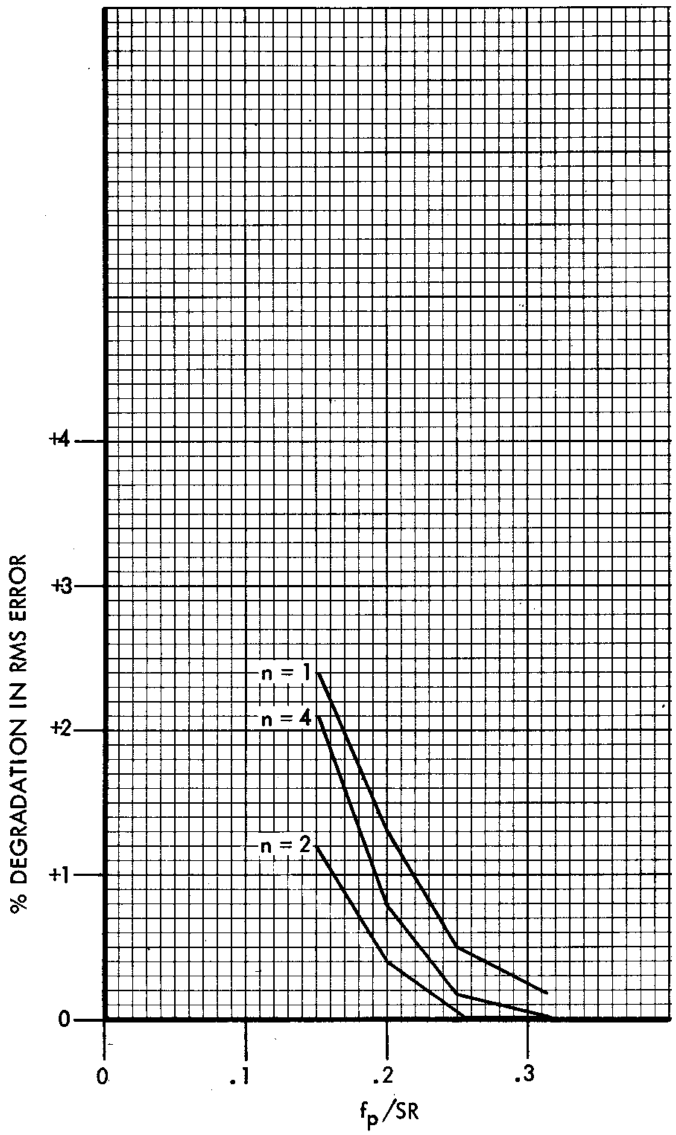


Figure 4. 3rd Order Data

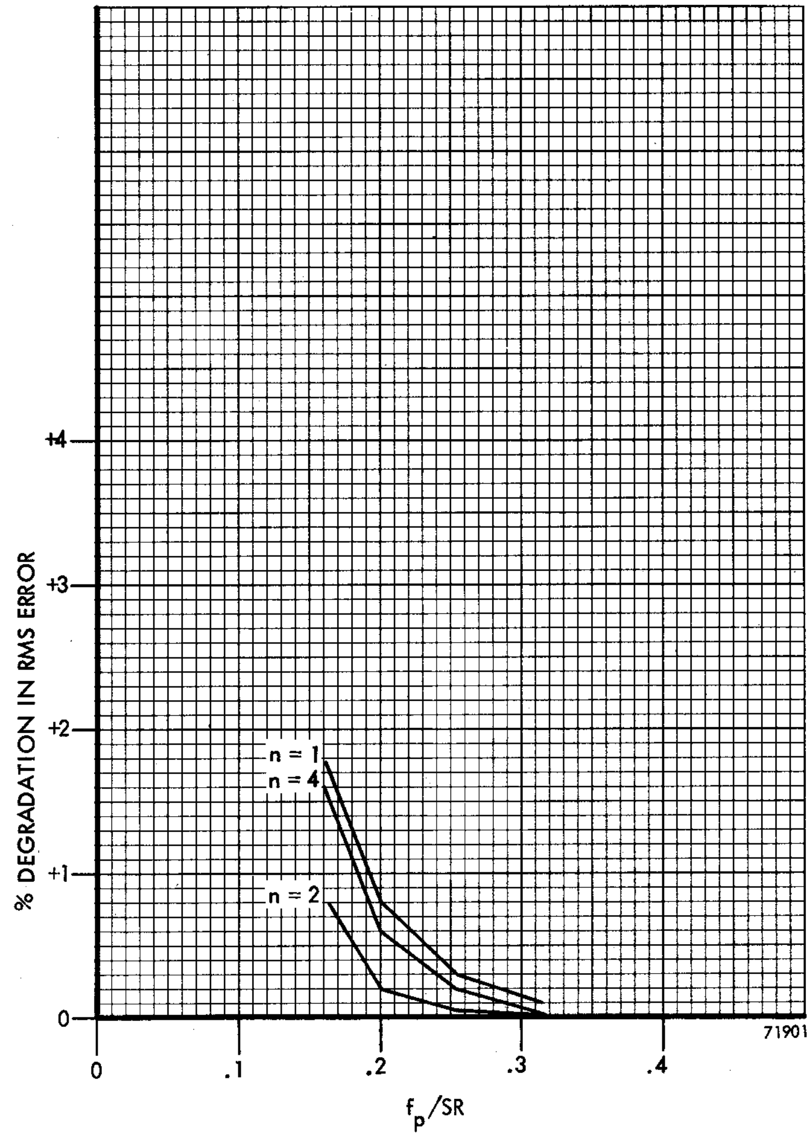


Figure 5. 4th Order Data

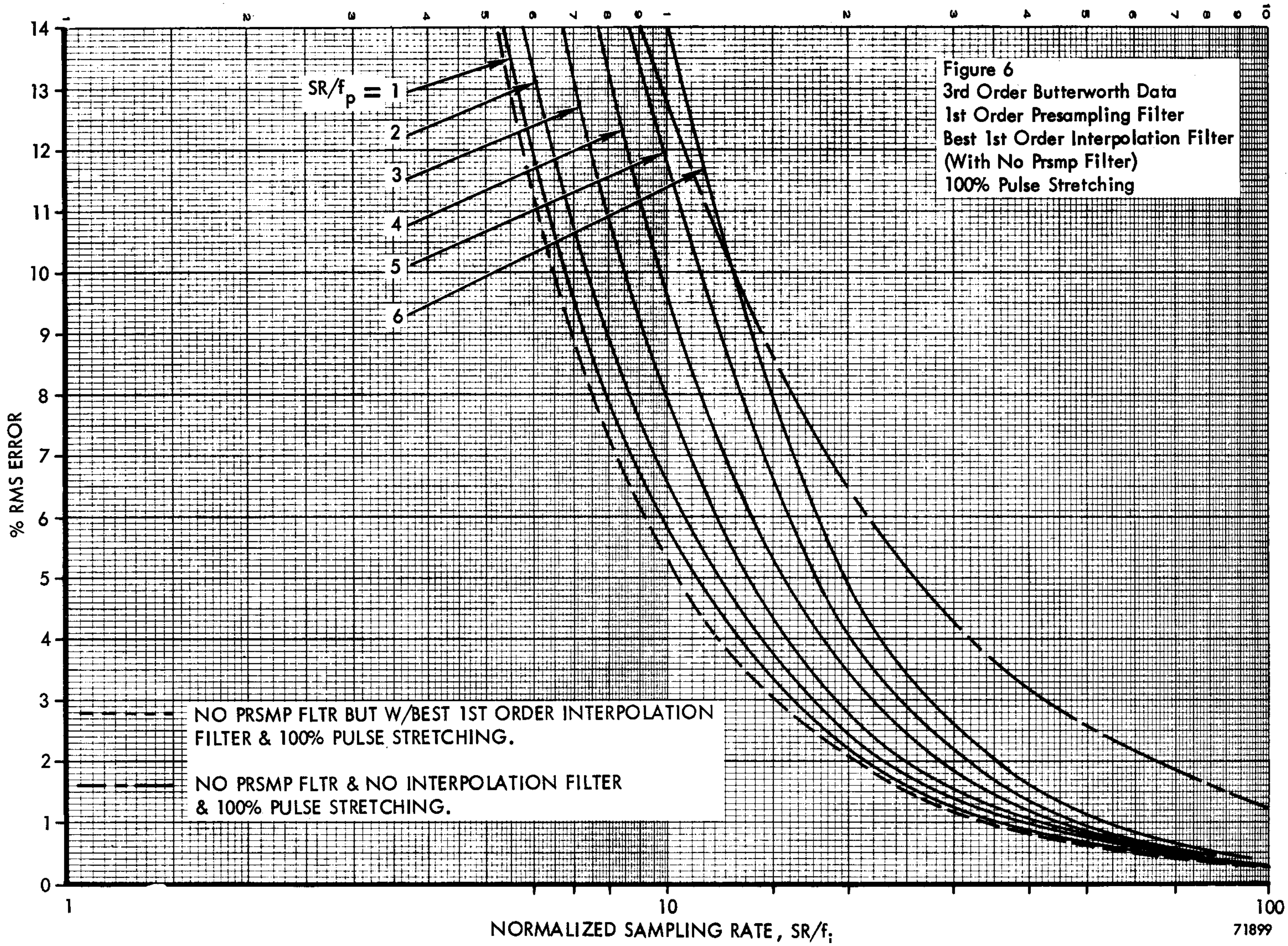


Figure 6
3rd Order Butterworth Data
1st Order Presampling Filter
Best 1st Order Interpolation Filter
(With No Prsmp Filter)
100% Pulse Stretching
SR/f_p = 1
2
3
4
5
6
NO PRSMP FLTR BUT W/BEST 1ST ORDER INTERPOLATION FILTER & 100% PULSE STRETCHING.
NO PRSMP FLTR & NO INTERPOLATION FILTER & 100% PULSE STRETCHING.
% RMS ERROR
NORMALIZED SAMPLING RATE, SR/f_i
71899

VARIATION OF RMS ERROR VS nTH ORDER BUTTERWORTH PRESAMPLING FILTER BANDWIDTH.
CURVES ASSUME 5% RMS ERROR USING THE BEST 1ST ORDER INTERPOLATION FILTER ALONE.

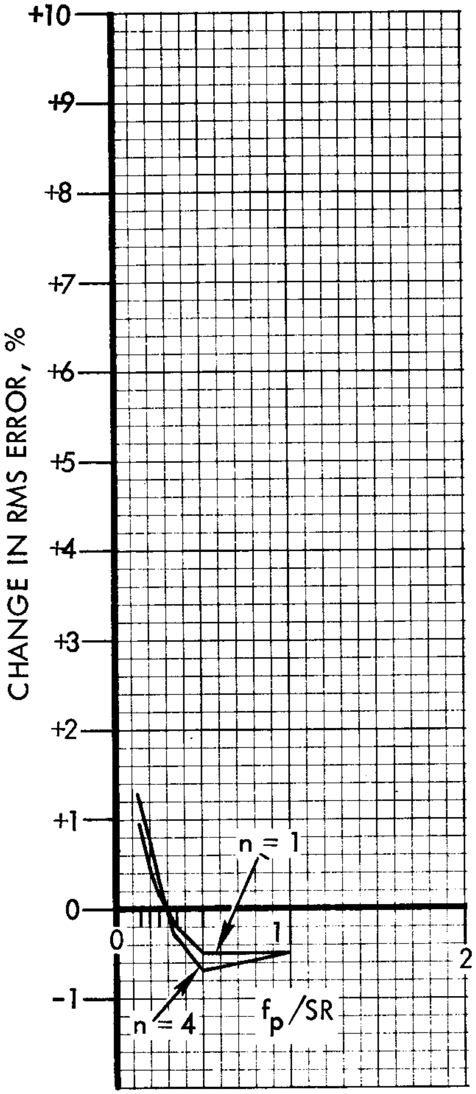


Figure 7 1st Order Data

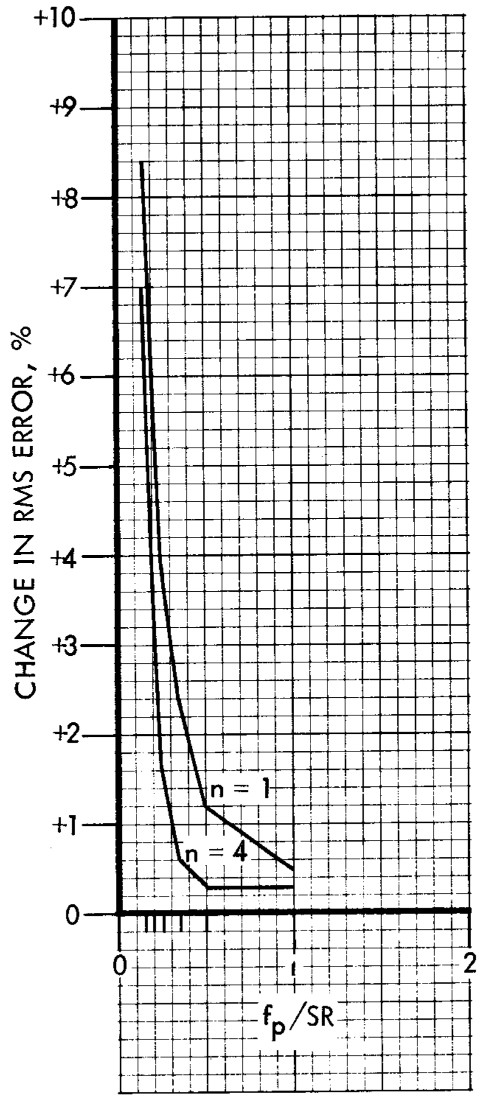


Figure 8 3rd Order Data

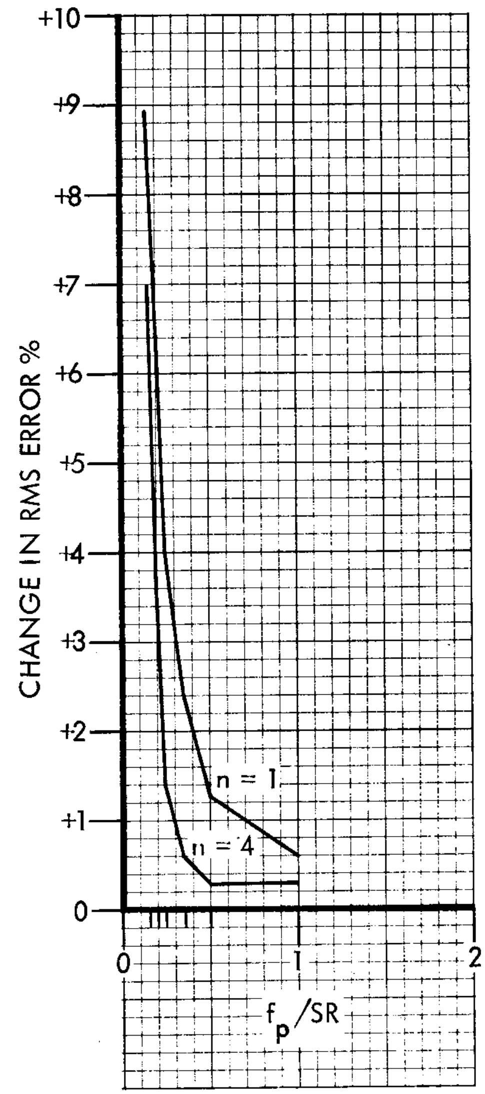


Figure 9 4th Order Data

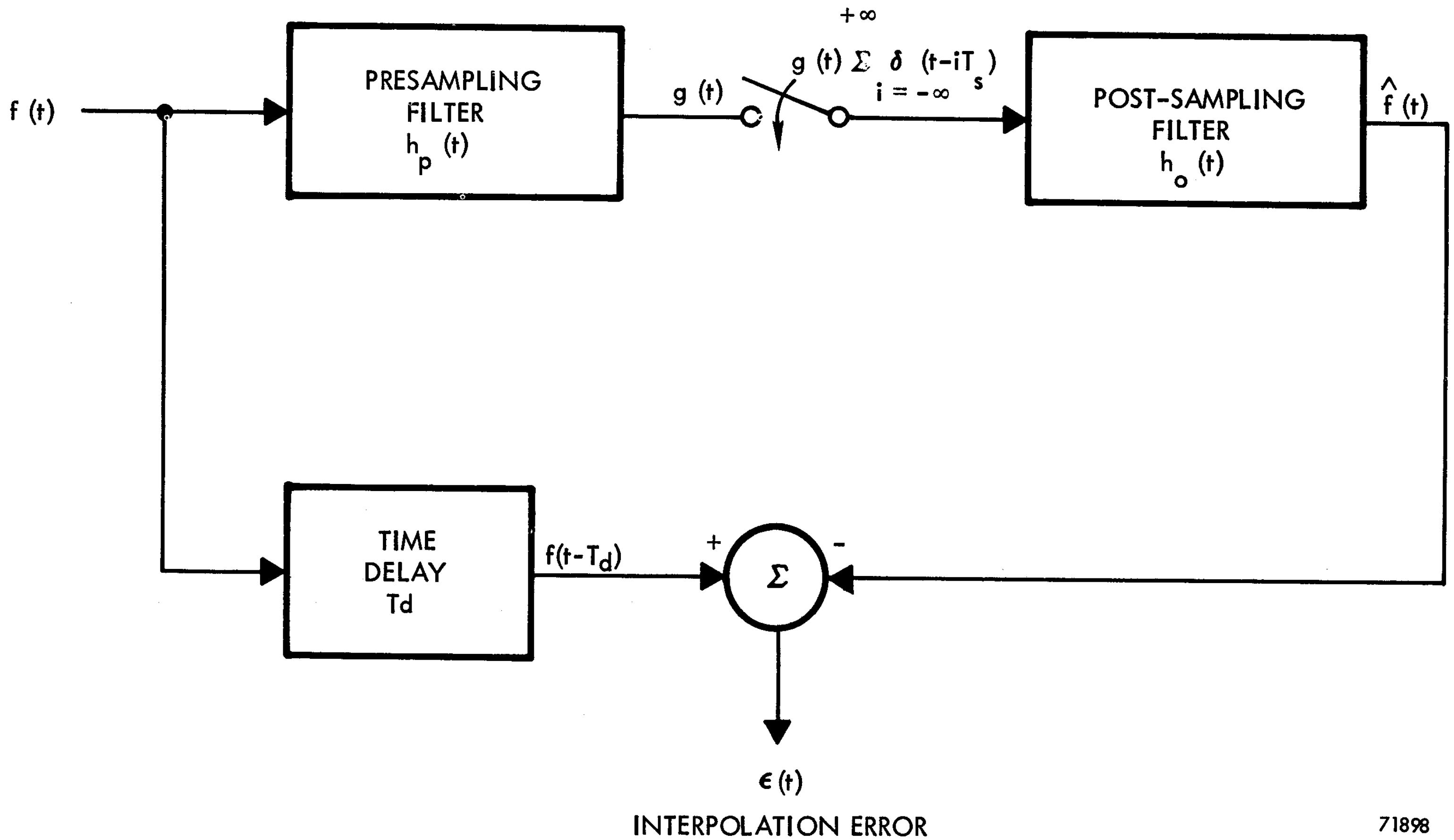


Figure 10. Sampled Data System Model

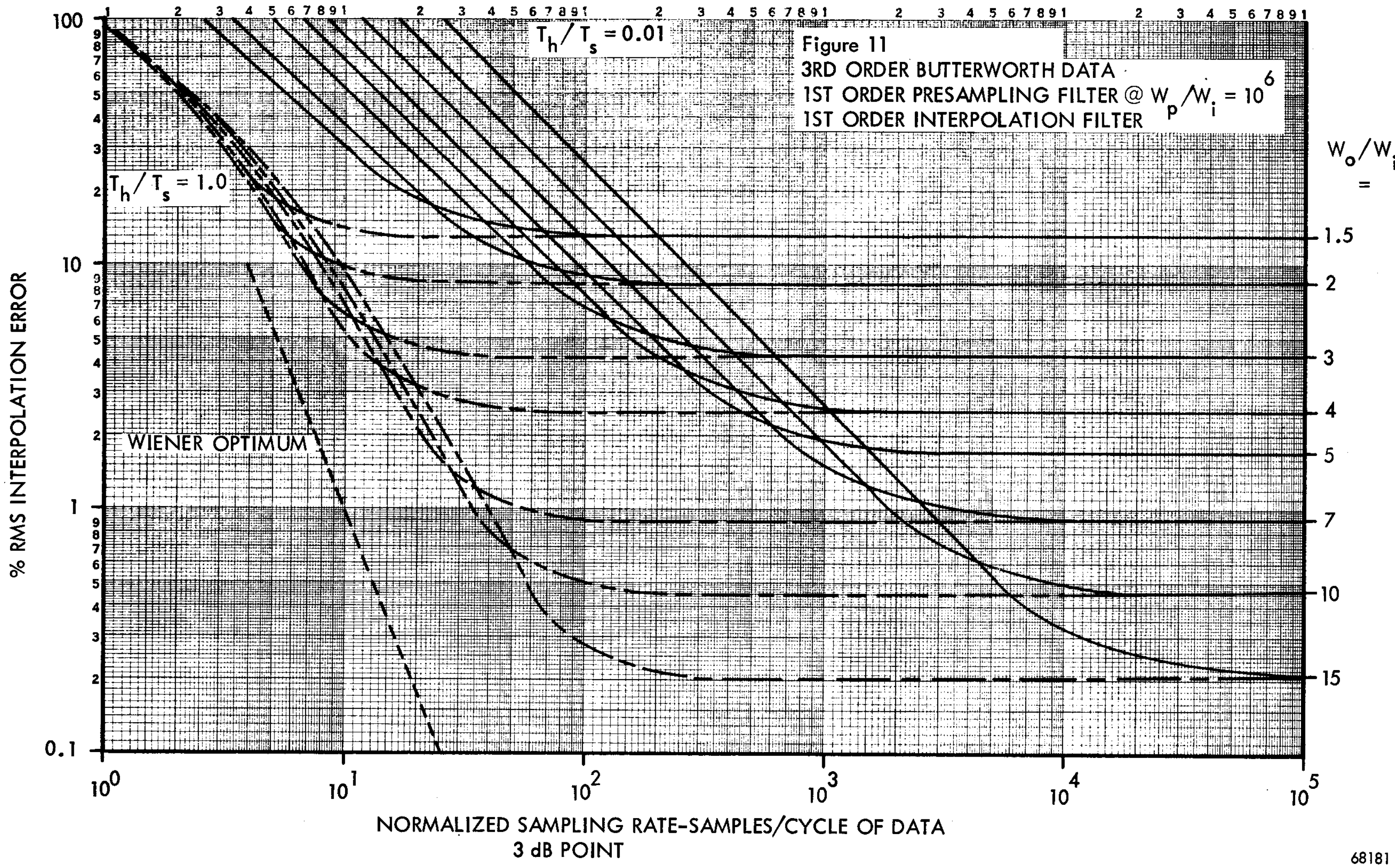


Figure 11
3RD ORDER BUTTERWORTH DATA
1ST ORDER PRESAMPLING FILTER @ $W_p/W_i = 10^6$
1ST ORDER INTERPOLATION FILTER
$T_h/T_s = 0.01$
$T_h/T_s = 1.0$
WIENER OPTIMUM
W_o/W_i =
1.5
2
3
4
5
7
10
15
% RMS INTERPOLATION ERROR
100
10
1
0.1
10^0
10^1
10^2
10^3
10^4
10^5
NORMALIZED SAMPLING RATE-SAMPLES/CYCLE OF DATA
3 dB POINT
68181