

Proceedings of the Institute of Acoustics

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

T.E. Curtis, Admiralty Research Establishment, Portland.

1. INTRODUCTION.

The computing power needed to beamform wide bandwidth signals from large arrays is one of the prime driving factors in the development of high throughput processors for current sonar systems [1], and the processing power needed by beamforming sub-systems is likely to increase to handle larger aperture arrays in future systems.

Space-time beamforming techniques [2] provide a convenient way to handle wide bandwidth signals efficiently but tend to be profligate with memory usage. Figure 1 shows schematically a space-time system for a planar transducer array: digitised element data is written into a random access memory (RAM) store, under the control of a "time-now" count that increments on each store update. Element data is written to the store in locations that map directly to spatial position in the array. The store holds a running time history of the received element signals, equal to the acoustic length of the array and therefore contains all the data necessary to form any real beam.

Beams are formed by summing samples of delayed element data from planes across the store, with read addresses defining the time delay increments from "time-now" for each channel for the required beam steer. For example, summing data from address plane 1 in Figure 1 would form a beam at broadside normal to the array vertical, address plane 2 a beam looking vertically downwards, address plane 3 to endfire, etc.

In practice, each channel in the space-time store consists of the circuit shown schematically in Figure 2: this includes addressing logic, a "time now" count (modulo the time dimension of the RAM store) and an indexed read addressing scheme to generate the necessary delay offsets. The store update and read operations are interleaved, so that samples of the required beams are generated between write updates, but output beams are not necessarily formed at the store update rate, i.e. stored data is usually oversampled. The multiplier is primarily used to weight the array element data with a shading function to control spatial sidelobe levels and for interpolation processing.

The space-time architecture has a number of advantages:

1. it is a direct implementation of a time delay beamformer and handles broad bandwidth signals without the problems of k-space interpolation and of transient jamming encountered in phase or frequency domain realisations

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

2. it can be generalised to handle complex array geometries e.g. conformal or cylindrical arrays, by projecting the address planes onto the required array shape - this can be achieved by simple memory-based look-up tables in practice,

3. the system can handle non-uniform element spacing in a similar manner,

and 4. if array motion is measured, using for example a set of tri-axial accelerometers, array attitude information can be used to modify the read address planes and to inertially stabilise beam pointing directions in space.

However, a practical implementation of the simple system in Figures 1 and 2 is not without its problems. These arise primarily because the data sample rate used for store updating has to be something of a compromise: ideally, in order to minimise the effects of time delay quantisation on beam sidelobe levels, heavily oversampled element data would be stored in the space-time memory but this leads to high input data rates, requiring increased memory access bandwidths, and large capacity data stores.

High input data rates create a number of engineering problems, as they reflect back from the beamformer to the signal acquisition and data transmission systems. Towed arrays, for example, need to use large numbers of transducer channels but the data transmission bandwidth available from the array to the inboard signal processing is limited by the physical constraints of array construction, tow cables, etc. Similar data transmission problems exist with hull mounted arrays, particularly on submarines where the number of hull penetrations, and the signal bandwidth they provide, is limited.

Transmission rates can be reduced by digitising the array data at a frequency approaching the Nyquist rate and interpolating the element data to the required temporal resolution before beamforming. Typically, this interpolation factor will be large, as data sampled at many times the Nyquist rate is required to produce reasonable sidelobe levels. This technique is also very inefficient compared with some form of post-storage interpolation and the processing power required can be a problem unless a high efficiency interpolation algorithm is available.

High capacity stores are not in themselves a problem: production quantities of single chip dynamic RAMs with 4 Mbit storage capacity are available now from several sources [3] and the advanced technologies, e.g. trench and stacked capacitor, to support 16 and 64 Mbit devices in development [4]. However, RAM capacity can be a problem if highly integrated application specific integrated circuit (ASIC) beamformer designs are considered. ASIC technologies do not provide dynamic RAM macro-cells: most provide parameterised static RAM blocks but these are usually implemented using discrete logic gates and do not provide "state-of-the-art" memory densities.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

2. INTERPOLATION REQUIREMENTS.

In a typical system for say a 256 element array sampled at say 8 times the Nyquist rate, around 128 Kbytes of RAM are required for the space-time store, together with some control and arithmetic logic. One obvious approach to reduce data rates and minimise storage is to use post-storage interpolation: data can then be transmitted and stored at low sample rates and the sampled time series data for each element interpolated to generate the necessary delay resolution before combining the array data to form beams.

Eight kilo-bytes of static RAM in an 0.9 micron DLM CMOS ASIC technology [5] occupies about the same silicon area as a couple of 16-bit multipliers. So, if some compact, hardware-efficient post-storage interpolation techniques can be developed for stored data sampled at frequencies approaching the Nyquist rate, saving in silicon area of greater than 6:1 are possible. This ratio might well drop for a full custom design with a DRAM capability, but the commercial advantages of using ASIC technology make the trade-off between data storage capacity and DSP complexity worth considering.

A block schematic for a post-storage interpolation system is shown in Figure 3: fine delay interpolation is assumed to be provided by some DSP technique, and provides a "vernier" delay capability, added to the coarse delay selection available directly by RAM store read address offsetting.

A separate delay interpolator is required for each transducer channel for each beam; for a system with 256 hydrophones generating 256 beams, this equates to 65536 logically separate interpolators, although in practice all channels would be multiplexed through the a smaller number (maybe one) of physical circuits. With signal bandwidths of say 5 KHz, around 10^9 interpolations per second are needed and for sidelobe performance better than say -30 dB, interpolator error bounds of ± 3 degrees in phase and $\pm 5\%$ in amplitude would be typical.

3. INTERPOLATION SYSTEMS.

Various techniques have been reported in the open literature to implement post-storage interpolation systems [6], mainly based on the use of simple linear interpolation or finite impulse response (FIR) filter structures to approximate time delays.

3.1 Linear Interpolation.

Linear interpolation is attractive as it is a simple operation to implement in hardware: it can be considered as a two-point FIR process with a difference equation of the form:-

$$Y_{(k)} \text{int} = \alpha \cdot Y_{(k-1)} + \beta \cdot Y_{(k)}$$

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

and a z-domain transfer function:-

$$H(z) = \alpha \cdot z^{-1} + \beta$$

where α and β are chosen so that magnitude of the transfer function is more or less equal to unity over the frequency band of interest and the rate of change of phase shift with frequency sensibly constant, so that the circuit approximates the required time delay.

Figure 4(a) plots the variation of phase shift with signal frequency (normalised to the clock rate) for this system for various values of α and β and Figure 4(b) the corresponding amplitude vs frequency response. For the error bounds specified, this linear interpolation system can be used at frequencies up to around 0.11 times the sampling clock rate, equating to oversampling ratios of approaching 5 times the Nyquist rate.

3.2 FIR Interpolation.

For sample rates less than this, say 1.25 times the Nyquist rate, longer FIR filters are needed, typically 16- or 32-taps [6]: for the 256 channel, 256 beam system above, this requires in excess of 10^{10} arithmetic operations per second. With current programmable DSP technology, this represents a large system, around 1000 devices. It may be possible to reduce this figure using devices designed specifically for FIR filtering but these are difficult to use in beamforming applications since data is most conveniently interpolated in an element sequential, beam sequential order of each sample time epoch.

With commercially available FIRs [7,8], based on distributed multiplier-accumulator (MACC) architectures, interpolation order causes two main problems. Firstly, serial coefficient loading imposes a time overhead on interpolation filter channel switching (a different set of FIR coefficients is needed for each element in each beam). Secondly, the pipelined architecture used in distributed FIR MACCs has been developed for time sequential data flow applications and pipeline flushing is required between channel changes. Pipelined devices can be utilised in a sample sequential, channel sequential, beam sequential system, i.e. a block time processing system, but this simply amounts to a pre-storage interpolation system (with some added latency), as the intermediate high resolution interpolated data from individual channels needs to be stored before beam summation.

At first sight, these disadvantages could be overcome by integrating the complete storage and interpolator system, using a systolic implementation [9]. However, this would need to process all 16 or 32 time series samples of the element data samples simultaneously and the organisation of the 512 or 1024-bit wide data paths (16 or 32 samples of 16 bits for both data and coefficients) and the space-time store architecture required to achieve this would be an interesting engineering exercise, to say the least.

The ideal interpolator would use a small number of time series samples and a

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

minimum number of internal states that must be stored and restored when switching channels/beams. Linear interpolation between adjacent element samples obviously meets these criteria but it requires high oversampling rates: FIR interpolation performs well but is difficult to implement efficiently in large systems.

The following sections outline a recursive interpolation implementation based on wave digital filter (WDF) all-pass networks. This recursive method has the advantage that it operates with modest oversampling rates (down to around 2.5 times Nyquist directly, down to 1.25 times with some modification) and with algorithmic efficiency about an order of magnitude better than FIR based realisations.

4. ALL-PASS DELAY NETWORKS

Low complexity WDF structures for high specification filter applications have been described in the literature [10,11]. Many of the implementations developed are based around multi-branch all-pass networks, such as that shown in Figure 5; each branch in the network has a transfer function of the form:-

$$H(z^{-1}) = \prod_{j=1}^J \frac{\tau_j + z^{-1}}{1 + \tau_j z^{-1}}$$

The network is all-pass, so the magnitude of the transfer function is equal to unity, but the phase/frequency characteristic of the network is controlled by the filter coefficients, τ_j , and is given by:-

$$\theta_j(\omega) = -2 \cdot \tan^{-1} \left[\frac{1 - \tau_j}{1 + \tau_j} \tan(\omega/2) \right]$$

Consider for example the first order section in Figure 5: the phase response for the network with coefficients in the range 0..1 is plotted in Figure 6(a). It can be seen from the Figure that the phase is sensibly linear over the first part of the curves and controlled by gamma so the circuit can be considered, to a first approximation, as a variable time delay. Phase non-linearity over this range is plotted in Figure 6(b): it can be seen that a phase error of less than +/- 1.2 degrees is maintained for frequencies up to 0.2 times the sampling rate, or for oversampling rates greater than 2.5 times Nyquist.

This recursive system can be used directly to implement a fine delay mechanism

Proceedings of the Institutue of Acoustics

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

for a post-storage interpolation beamformer with data oversampling rates of 2.5 times Nyquist or greater. Computer simulation indicates that sidelobe levels approaching -40 dB are achieved on 32 element arrays with better sidelobe performance on larger arrays: this has been verified in a number of practical hardware applications.

5. PRACTICAL IMPLEMENTATIONS.

As can be seen from Figure 5, there are several circuit topologies that generate the same all-pass response: the circuit in Figure 5(c) is better suited to implementations using standard multiplier/accumulators, whilst those in Figures 5(a) and 5(b) fit well onto ASIC designs.

A well as a number of implementations using "off-the-shelf" DSP MACC components, (see Figure 7), the technique has also been used in several integrated silicon designs. The following paragraphs outline the parameters and performance of a full custom programmable processor, developed as part of the UK VHPIC Application Demonstrator (VAD) programme [12], and a commercial "sea-of-gates" ASIC implementation is described in Section 6.

5.1 The Block Floating-point Programmable Arithmetic Unit (BPPAU).

The BPPAU [13] was developed jointly by MOD and Plessey under the VAD programme. It integrates two identical 24-bit block floating-point arithmetic units onto a single silicon die (Figure 8), approximately 14mm x 11mm containing in excess of 240,000 transistors, fabricated using the 1.4 micron DLM CMOS process at Plessey, Roborough.

It is a general purpose programmable device, specifically designed as a processing node in signal flow networks [14] for high speed sonar signal processing applications. The internal architecture of the chip is based around a WDF data flow path and has been optimised to perform high throughput sonar front-end processing functions (e.g. beamforming, filtering) efficiently. Some bench-marks for the circuit running typical sonar applications are given below [13]:-

Beamforming(with interpolation and window)	32 beams from a 32 element array every 63 μ Secs.
1024-point complex FFT with window	two FFTs in 1.25 mSec

TABLE 1 - PLESSEY BPPAU BENCHMARKS

Operating at clock rates up to 30 MHz, the BPPAU provides in excess of 7.5 million WDF interpolations per second. Figure 9 shows the measured beam response with the device programmed as an all-pass interpolative beamformer:

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

sidelobe levels around -38 dB are obtained with a 16-element array sampled at 2.5 times the Nyquist rate. Figure 9 also plots the beam responses with the interpolation disabled.

6. REDUCING ARRAY SAMPLE RATES.

The oversampling ratio used in the all-pass interpolative beamformer above is defined by the phase linearity of the WDF system. Ideally, it would be convenient to find some low complexity post-storage interpolation method that was phase linear at lower sampling rates. This interpolation need not be variable, as in the WDF system: a simple fixed interpolation to increase the time resolution of stored data by a factor of two would be sufficient, so that a WDF all-pass interpolator system could operate on this interpolated data.

This can be achieved relatively easily: whilst absolute phase linearity can only be realised over a limited frequency band with WDF sections, differential linearity between sections can be exploited.

Differential phase and phase non-linearity for the two path network is shown in Figures 10. Differential phase linearity of better than ± 1.5 degrees is obtained at frequencies up to 0.4 times the sample rate. Consequently, the circuit can be used to interpolate data oversampled at 1.25 times the Nyquist rate by a factor of two, with an error of less than 1.5 degrees.

This system can be used in an architecture like that shown in Figure 11(a) but in practice it is often more efficient to split the interpolation processes and use combination of pre- and post-storage interpolation, as outlined in Figure 11(b).

This all-pass pre-/post-storage interpolator beamforming capability has been included in a programmable WDF filter ASIC, developed primarily for high speed band-shifting applications.

The device, developed in collaboration with GEC (Avionics), was implemented on LSI Logic's LA9000 series gate array (1.5 micron DLM CMOS process), using SDL CAD software. It operates at 30 MHz, with 24-bit fixed point arithmetic: further details of the device are included in reference 15.

Measured beam responses for this system, with a 16 element array and data sampled at 1.25 times the Nyquist rate are shown in Figure 12: the ASIC will calculate in excess of 10 million interpolations per second and processes a 32 element array, generating 32 beams, in 112 μ Secs: around 30 devices are needed for the system application considered in Section 1.

The major drawback with this implementation is that it does not have a linear phase response with frequency - see Figure 10: this is not usually a problem in passive sonars but can be so in active systems using wide-band transmissions and correlation processing. However, in practice, the

Proceedings of the Institutue of Acoustics

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

pre-interpolator phase non-linearity is often swamped by the transducer phase response and correlation replicas are phase corrected for the complete send/receive electronics transfer function when they are generated.

In more sensitive applications, the phase response can be made linear by using the pre-storage interpolation scheme shown in Figure 13 based on linear phase WDF networks [16,17]: measured phase non-linearity and differential non-linearity are shown in Figure 14.

7. DISCUSSION AND CURRENT WORK.

The previous Sections indicate the development of a collection of simple beamforming interpolation methods that can produce compact hardware implementations. The basic all-pass interpolator works with over sampling ratios around half those required for linear two-point interpolation, with a simpler processing algorithm (i.e. it requires one multiply and two adds per interpolate c.f. two multiplies and one add for the linear interpolate). The recursive implementation described provides a more compact and efficient realisation than the that obtained using FIR interpolators.

The application of WDF all-pass networks in beamforming sub-systems have been demonstrated to provide useful savings in a number of systems and enable complexity/storage capacity trade-offs in ASIC implementations.

8. ACKNOWLEDGEMENTS.

The author gratefully acknowledges the considerable input to the work outlined in this paper from Plessey Naval Systems, GEV(Avionics) and Imperial College, London.

Copyright HMSO, 1989.

9. REFERENCES.

- [1] T E CURTIS, A G CONSTANTINIDES and J T WICKENDEN, 'Control Ordered Sonar Hardware - COSH: A Distributed Processor Network for Acoustic Signal Processing', IEE Proc, Part F, 131, 1984.
- [2] T E CURTIS and R J WARD, 'Digital Beamforming for Sonar Systems', IEEE Proc, Part F, 127, 1980.
- [3] see for example:
Hitachi HM514100JP/ZP-8/10/12 Data Sheets, Hitachi Ltd, 1989.
- [4] see for example:
Digest of Technical papers, International Solid-State Circuits Conference, ISSCC89, 1989.
- [5] see for example:
LCA10000, LCA100000 Compacted Array Series Data Sheets, LSI Logic, 1989.
- [6] R G PRIDHAM and R A MUCCI, 'A Novel Approach to Digital Beamforming', JASA, 63(2), 1978.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

- [7] See for example:
IMS A100 Data Sheets and Application Notes, INMOS.
- [8] K J LAMB, 'A High Performance Chipset for Digital Filtering', ERA Seminar on Digital Signal Processing, ERA Report 88-0386, London, 1988.
- [9] see for example:
H T KUNG, 'Design of Systolic Arrays for System Integration', EUROSIP International Conference on Digital Signal Processing, Florence, 1984.
S Y KUNG, 'VLSI Array Processors', Proc International Workshop on Systolic Arrays, Oxford, 1986.
- [10] M G BELLANGER et al; 'Digital Filtering by Polyphase Network: Application to Sample Rate Alteration and Filter Banks', IEEE Trans ASSP, 24, 1976.
- [11] R A VALENZUELA and A G CONSTANTINIDES, 'Digital Signal Processing Schemes for Efficient Interpolation and Decimation', IEE Proc, Part G, 130, 1983.
- [12] D J MARSHALL, 'ASICs for Military Applications', ERA Seminar on Digital Signal Processing, ERA Report 88-0386, London, 1988.
- [13] D WADHAM, 'The Development of a High Performance Acoustic Signal Processor', ERA Seminar on Digital Signal Processing, London, 1989.
- [14] Y S WU et al, 'Architectural Approach to Alternate Low-level Primitive Structures (ALPS) for Acoustic Signal Processing', IEE Proc, Part F, 131, 1984.
- [15] A G CONSTANTINIDES and T E CURTIS, 'High Efficiency Wave Translation Filters for Sonar Band Selection', paper to be presented at this conference.
- [16] M RENFORS and T SARMAKI, 'Recursive N^{th} Band Digital Filters - Part 1: Design and Properties', IEEE Trans CAS, 34, 1987.
- [17] S B BRUTON, 'The Design of Low Complexity Digital Filters for Decimation and Interpolation', MSc Thesis #RNEC-SP-88002, Royal Naval Engineering College, Manadon, Plymouth, 1988.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

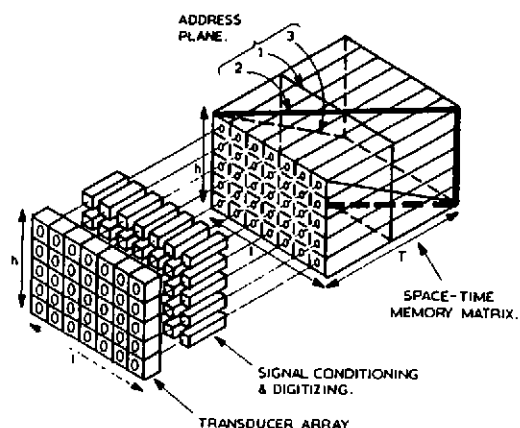


Figure 1

Space-time Beamformer Schematic for Planar Array.

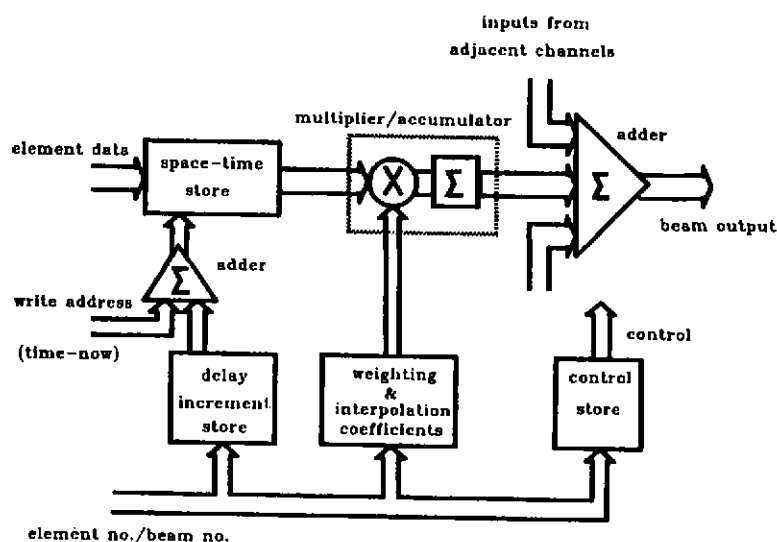


Figure 2 - Channel Schematic for Space/time Store.

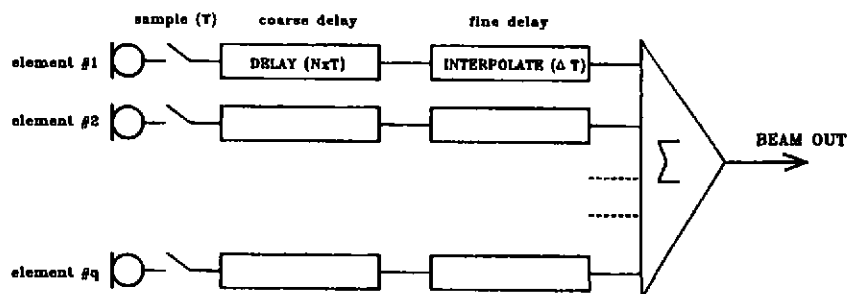


Figure 3 - Post-storage Interpolation Scheme.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

TWO POINT LINEAR INTERPOLATION

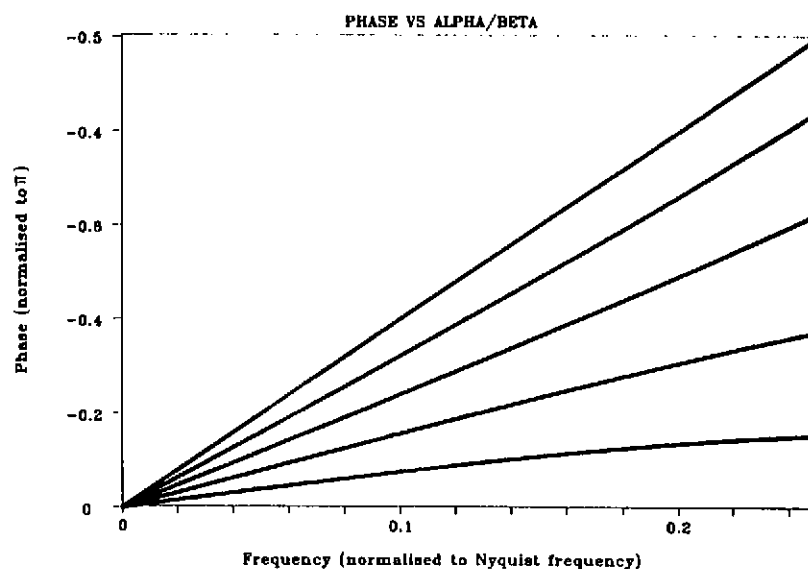


Figure 4(a) - Two-point Linear Interpolator:

Phase response with frequency for a number of values of α and β .

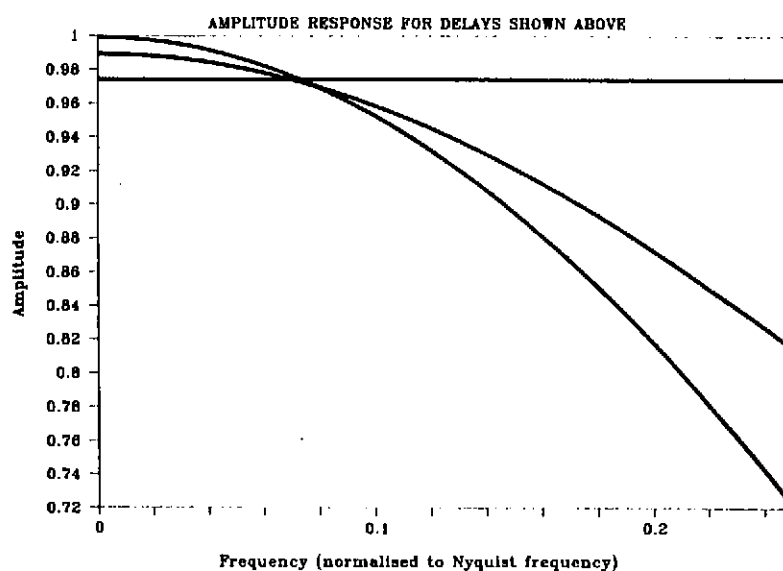


Figure 4(b) - Two-point Linear Interpolator:

Amplitude response for delay sections plotted in Fig 4(a).

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

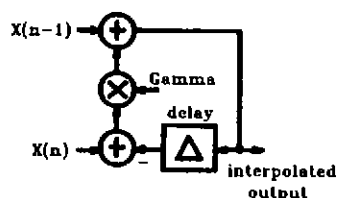


Figure 5(a)

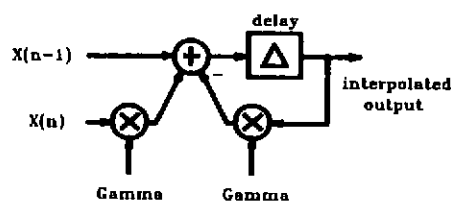


Figure 5(b)

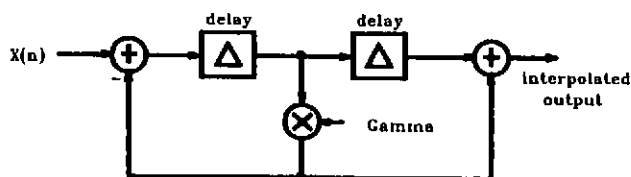


Figure 5(c)

Figure 5 - Wave Digital Filter Section Schematics.

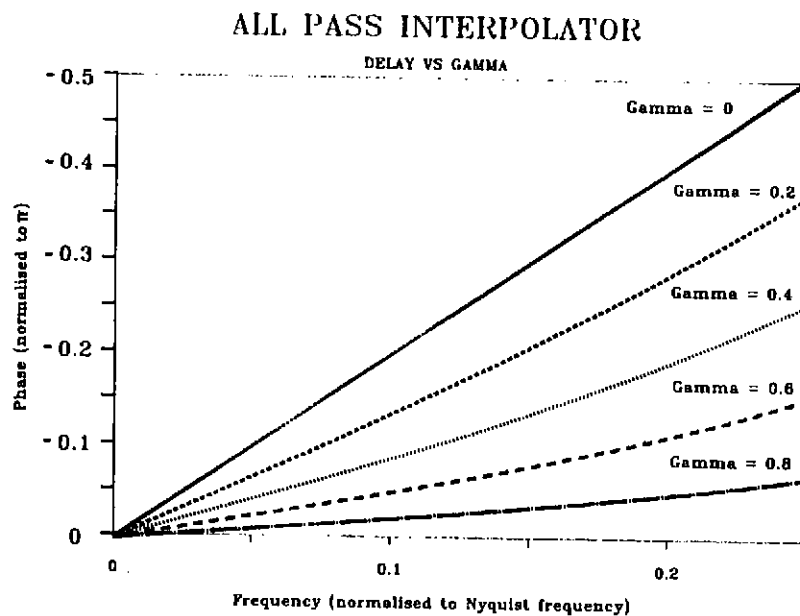


Figure 6(a) - All pass Interpolator:

Phase response with frequency
for a number of values of τ .

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

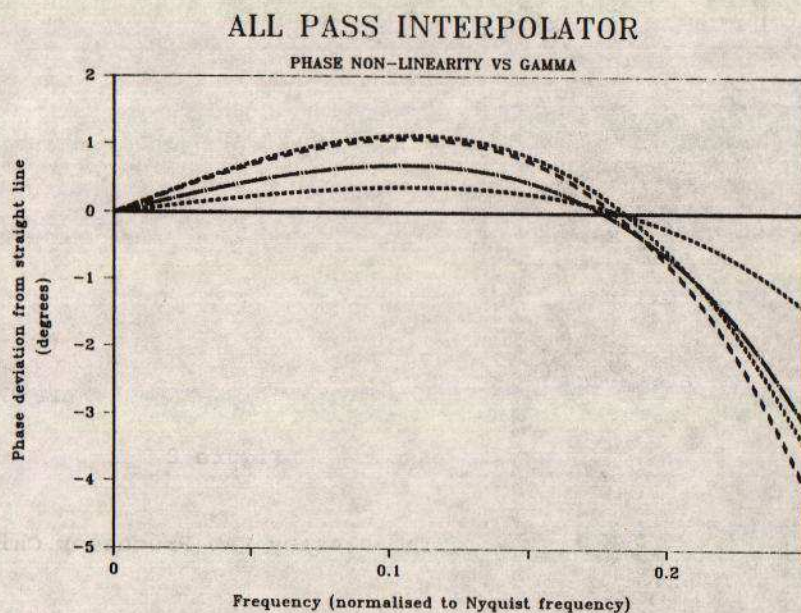


Figure 6(b) - All-pass Interpolator:

Phase non-linearity compared to
straight line fit for delay
sections plotted in Fig 6(a).

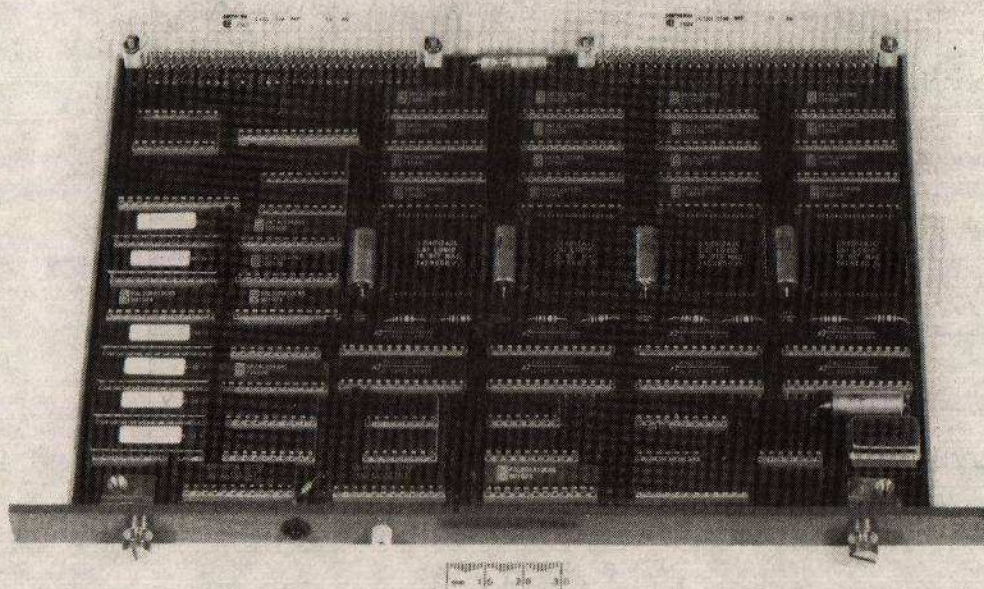


Figure 7 - Photograph of All-pass
Interpolation Beamformer:

The card uses commerciall MACC ICs and
runs at 40 million interpolates/second.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

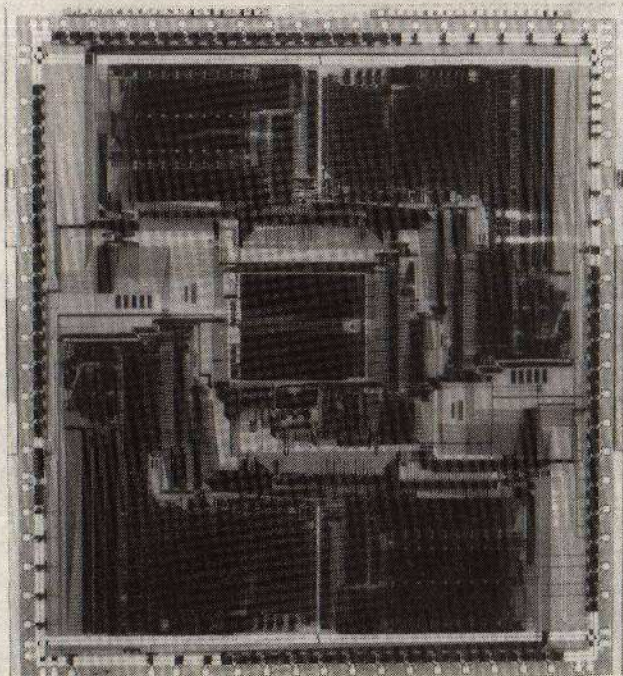
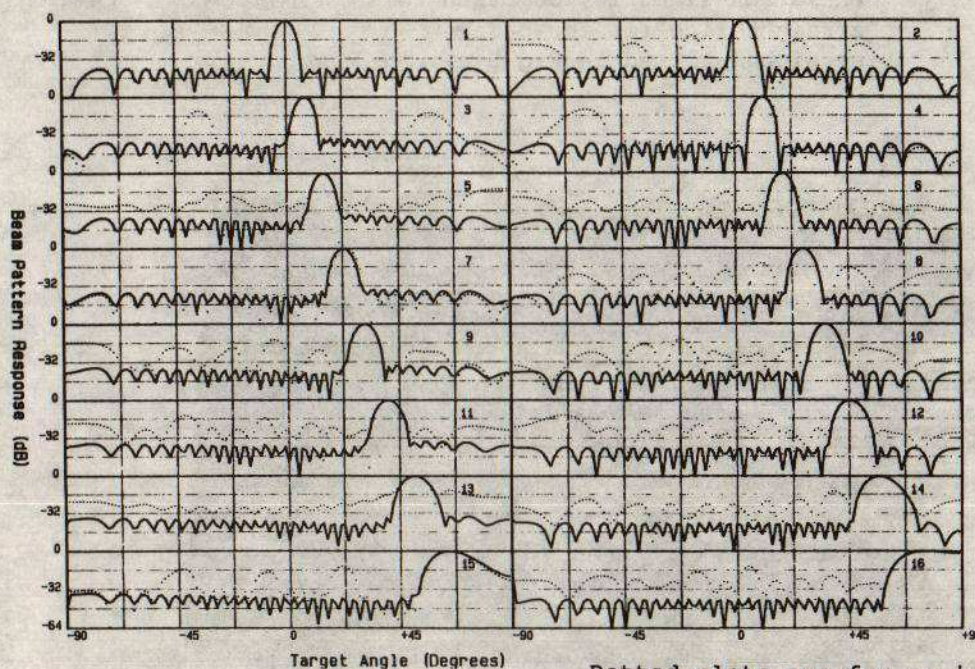


Figure 8

The Plessey VAD Processor Chip:

The two 24-bit BFPAUs can be seen
(top right and bottom left).



Dotted plots are for no interpolation.

Figure 9 - Measured Beampatterns

for a selection of steers for the
BFPAU programmed for all-pass
interpolation beamforming.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

TWO PATH ALL PASS INTERPOLATOR

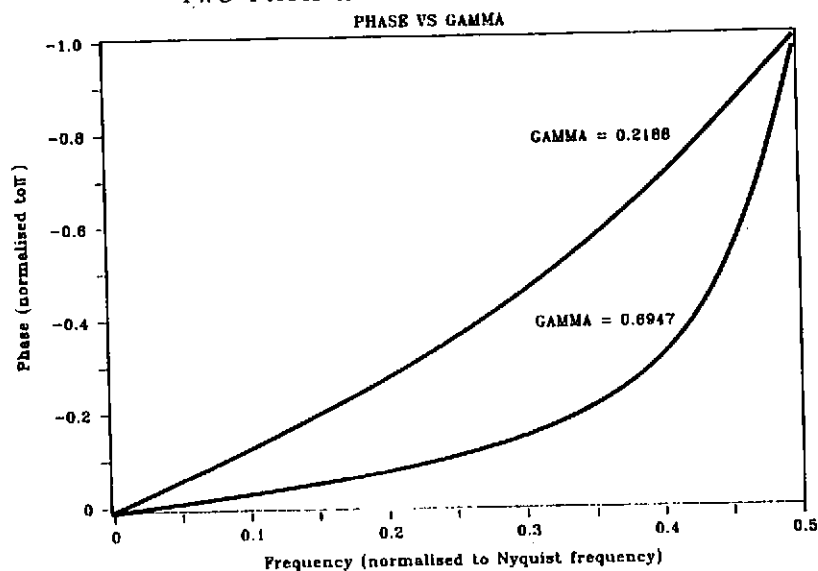


Figure 10(a)

Phase Responses for Two All-pass Interpolator Sections.

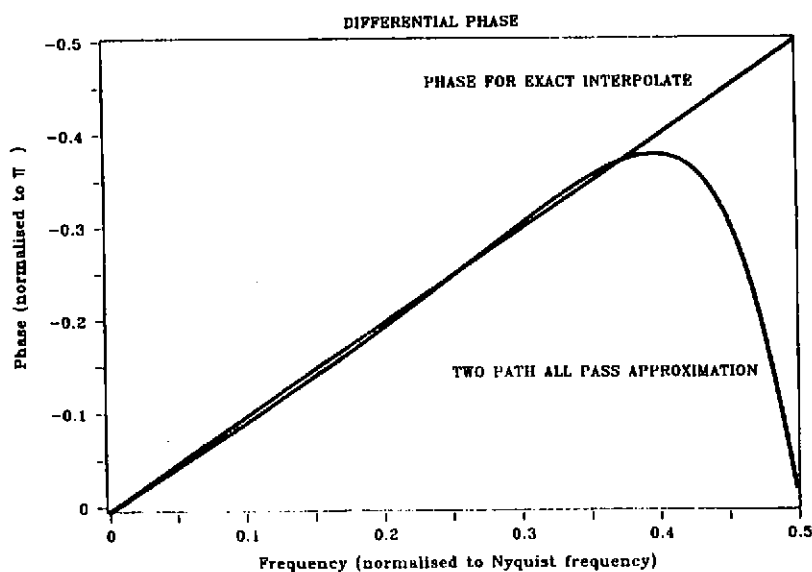


Figure 10(b)

Phase Difference for the Two Interpolator Sections in Figure 10(a).

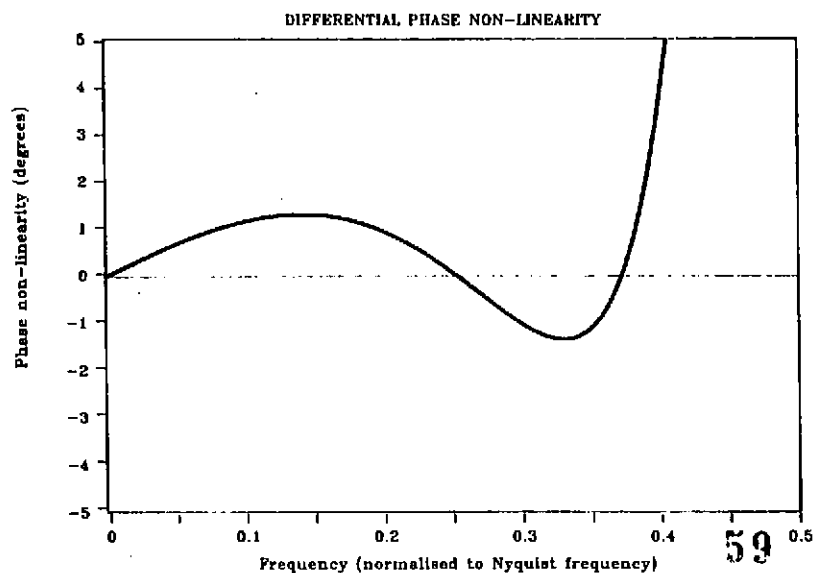


Figure 10(c)

Phase Non-linearity Compared with Exact Interpolation by a Factor of Two.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

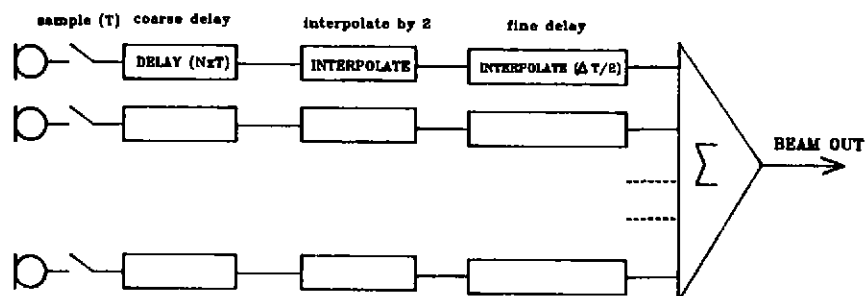


Figure 11(a) - Post-storage, Two Stage Interpolation Scheme.

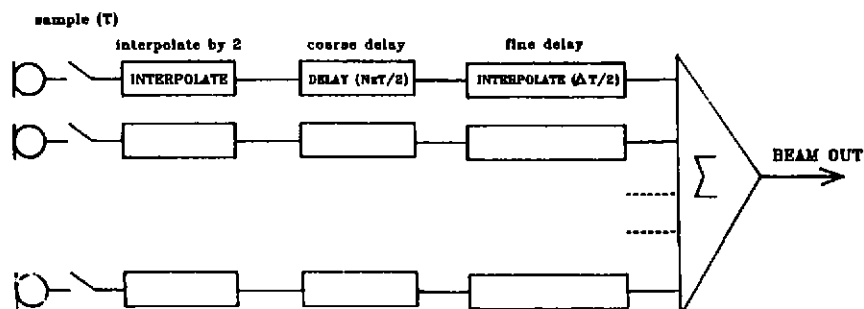


Figure 11(b) - Pre/post-storage Interpolation Scheme

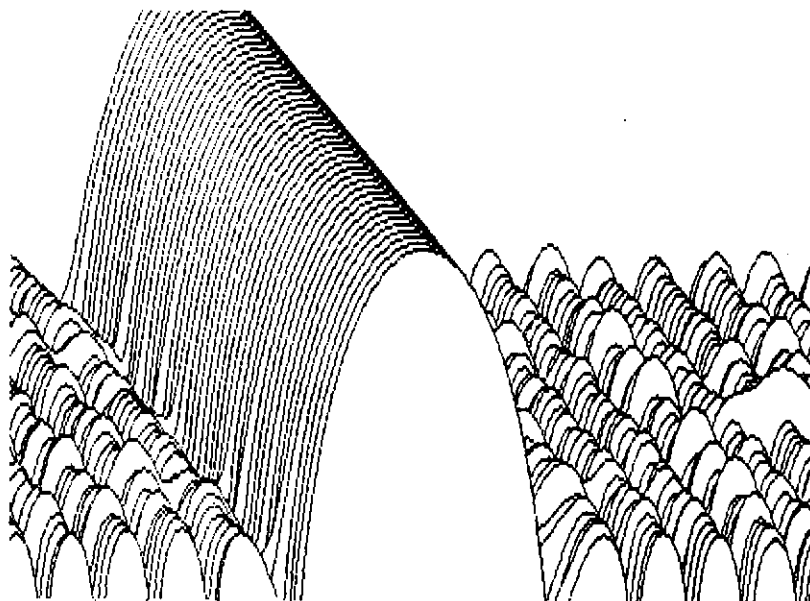


Figure 12 - Beampatterns:

WDF pre/post-storage
interpolation system
realised by WAVE ASIC.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

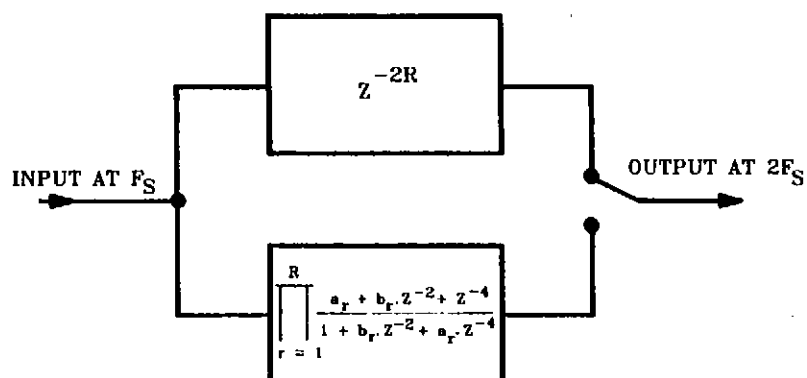


Figure 13 - Linear Phase WDF Schematic

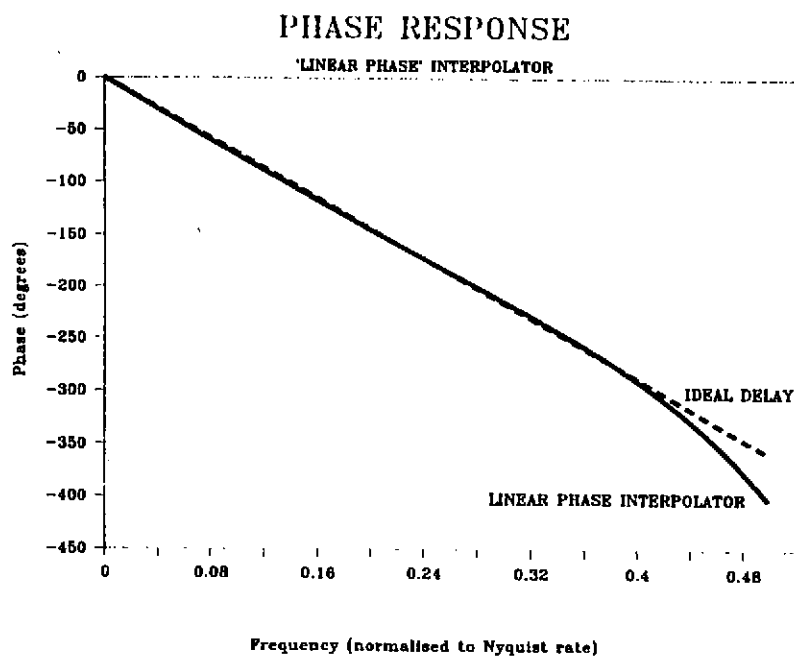


Figure 14(a) - Phase Response and Phase Linearity for System in Figure 13.

SPACE-TIME BEAMFORMING USING RECURSIVE INTERPOLATION

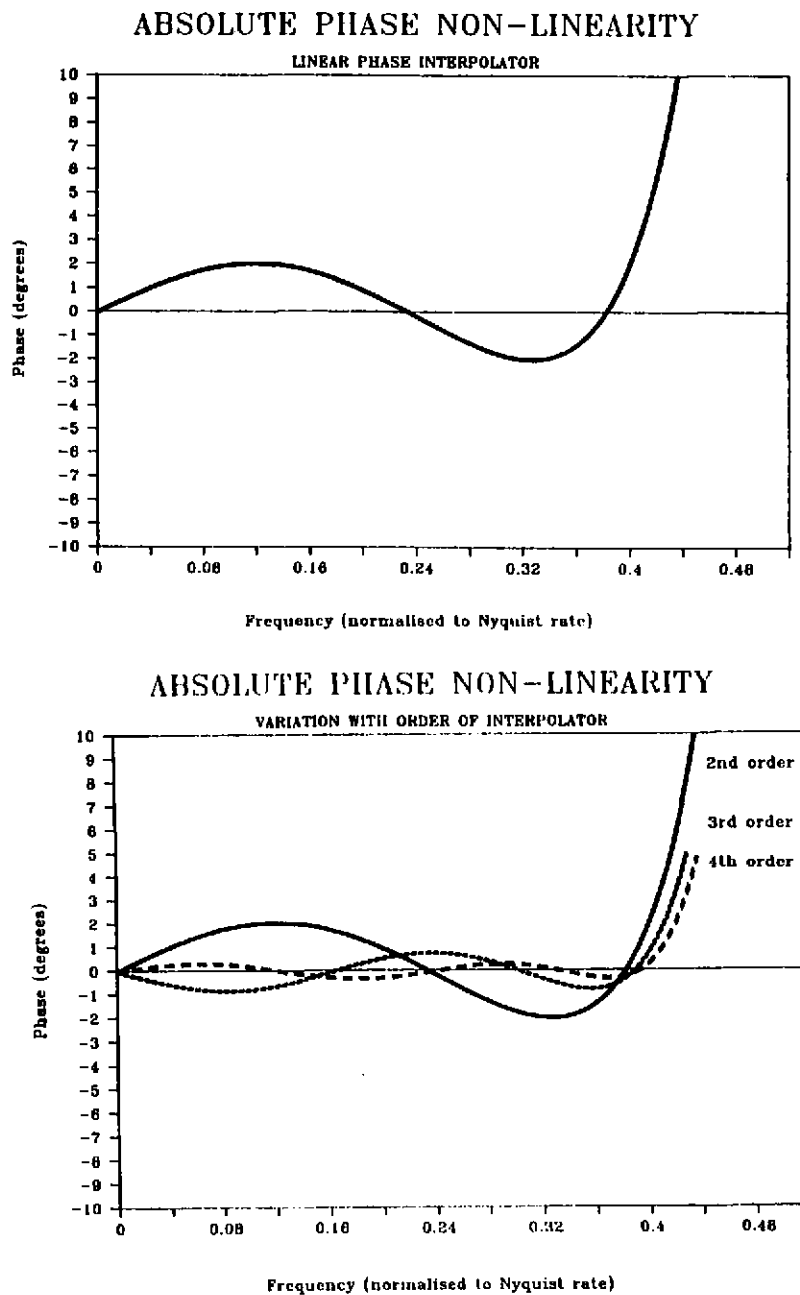


Figure 14(b)- Phase Response and Phase Linearity for System in Figure 13.