Implementation of directional Doppler techniques using a digital signal processor

N. Aydin D. H. Evans

Division of Medical Physics, Faculty of Medicine, Leicester University, Leicester, UK

Abstract—*Three methods of deriving directional signals from phase quadrature Doppler signals, using digital techniques, are described. These are the phasing-filter technique, the Weaver receiver technique and the complex FFT. The basic theory behind the three methods is presented, together with the results of digital simulations. Each of the methods has been implemented in real time using a commercially available digital signal-processing board, and their relative processing times are compared. All the methods work well, and the decision to implement one or other in a specific application is likely to rest on secondary factors, such as the need to tape-record the time domain output.*

Keywords—*Complex FFT, Digital complex modulation, Digital signal processing, Directional doppler, Hilbert transform, Spectrum analysis*

Med. & Biol. Eng. & Comput., 1994, **32**, S157–S164

1 Introduction

MOST CONTINUOUS-wave and pulsed-wave ultrasound Doppler devices utilise quadrature phase detection to provide directional information. A number of directional techniques based on quadrature detection have been used and are reviewed by Coghlan and Taylor (COGHLAN and TAYLOR, 1976). Traditionally, these implementations have been based on analogue signal-processing circuits. However, digital signal processors are now readily available which allow complex processing of audio-frequency signals in real time. Digital systems have several advantages over analogue systems; for example, they are easier to set up, less susceptible to aging and environmental variations, less sensitive to noise, and more flexible in general. This paper describes three digital implementations that may be used to derive directional signals from phase quadrature Doppler systems. These techniques are the phasing-filter technique (phase domain processing), the Weaver receiver technique (frequency domain processing) and the Complex FFT.

Quadrature phase detection is implemented by synchronously detecting real and imaginary Doppler-shifted frequencies in two quadrature channels, as shown in Fig. 1. Each of the methods described in this paper work directly on the I and Q outputs.

2 Digital implementations of directional Doppler detectors: theoretical considerations

In this Section, the basic theory behind the three approaches is described without recourse to complicated mathematical expressions. In order to clarify this approach, DSP simulations have been implemented and the results are illustrated graphically. The same data set was used for all simulations, to enable comparisons to be made between the methods.

Correspondence should be addressed to: Professor D. H. Evans, Department of Medical Physics, Leicester Royal Infirmary, Leicester LE1 5WW, UK

First received 7 August 1992 and in final form 5 April 1993

2.1 *Phasing-filter technique (PFT)*

A block diagram of this technique is shown in Fig. 2. The system is based on a wide-band digital Hilbert transform (HT), which produces a 90° phase shift in the input signal. A discrete-time HT is a linear shift-invariant discrete-time system whose ideal frequency response $H(e^{j\omega})$ is given by

$$H(e^{j\omega}) = \begin{cases} -j & 0<\omega<\pi \\ 0 & \omega=0,\pi \\ +j & -\pi<\omega<0 \end{cases} \qquad (1)$$

The corresponding ideal impulse response is given by

$$h(n) = \begin{cases} \dfrac{2\sin^2(\pi n/2)}{\pi n} & n\neq 0 \\ 0 & n=0 \end{cases} \qquad (2)$$

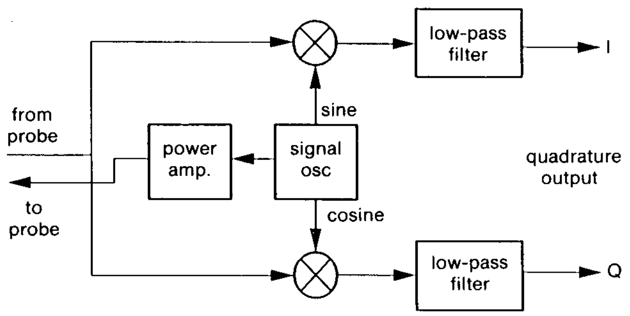


Fig. 1 *Block diagram of a Doppler system based on the quadrature phase detection technique*

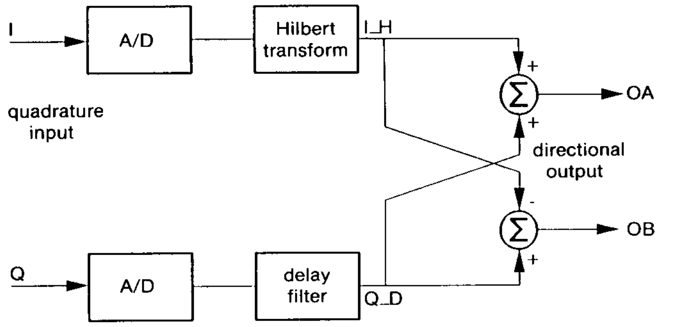


Fig. 2 *Phasing-filter technique (PFT) for use with quadrature phase detection signals*

This ideal response can be approximated by using finite impulse response (FIR) or infinite impulse response (IIR) filters (Ansari, 1987). The main properties of the HT are well documented (Schwartz *et al.*, 1966). In our applications, an FIR-type filter was implemented using the FIR filter design option of the Hypersignal* DSP software. The software implements the Parks-McClellan design algorithm.

The delay filter (DF) is an all-pass filter which introduces a delay that is exactly the same as that introduced by the HT filter. It was also implemented using a basic Parks-McClellan design. The passband area stretches from DC (zero) up to half the sampling rate, which means that the filter is all passband, with no transition bands. Once again the Hypersignal package was used to implement this filter.

Consider a perfect quadrature detection system with ideal quadrature outputs $I = A\cos\omega_a t + B\sin\omega_b t$ and $Q = A\sin\omega_a t + B\cos\omega_b t$, where ω_a represents the signal frequency due to flow in one direction and ω_b represents the signal frequency due to flow in the other. If these signals are applied to the system shown in Fig. 2, separated outputs are obtained. From the properties of the HT, the output of the filter is

$$I\_H = A\sin\omega_a t - B\cos\omega_b t \qquad (3)$$

which is the Hilbert transform of the in-phase component of the input signal.

The output of the DF is

$$Q\_D = A\sin\omega_a t + B\cos\omega_b t \qquad (4)$$

which is the delayed version of the quadrature component of the input signal. Note that in eqns. 3 and 4, the equal time delays between input and output have been ignored.

After addition and subtraction, the results are

$$OA = Q\_D + I\_H = 2A\sin\omega_a t \qquad (5)$$

$$OB = Q\_D - I\_H = 2B\cos\omega_b t \qquad (6)$$

In order to confirm these results, a simulation has been implemented using a commercially available signal-processing software package. Two complex waveforms were created with a sampling frequency of 16 kHz and 512 data points to simulate 1 kHz forward and 2 kHz reverse signals. They were

$$I = 1000\cos(2\pi n1000/16000) + 500\sin(2\pi n2000/16000) \qquad (7)$$

$$Q = 1000\sin(2\pi n1000/16000) + 500\cos(2\pi n2000/16000). \qquad (8)$$

A digital phase shifter was implemented using a digital HT and DF. The signal-processing algorithm was implemented using signal-processing functions of the DSP software package. The results are displayed in both the time and frequency domains in Fig. 3. As can be seen from the Figure, the digital filters introduce a time delay of approximately 6 ms. As the outputs are totally separated, two independent real FFTs are required to produce the frequency domain display.

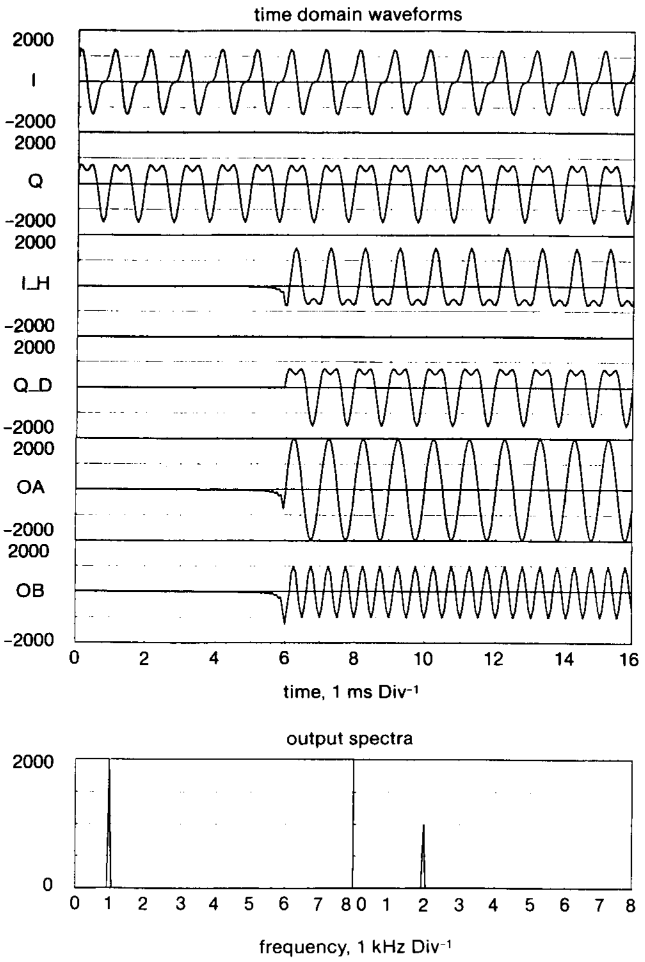


Fig. 3 *Time domain displays of each processing stages and output spectra for the PFT (simulation results)*

2.2 *Weaver receiver technique* (*WRT*)

This system, also known as the IQ demodulator, employs a digitally implemented quadrature mixer. Fig. 4 is a block diagram of the system. The internal digital signal oscillator generates a quadrature waveform, which determines the pilot frequency around which the directional outputs appear. The digital input quadrature signal is mixed with this pilot frequency and the outputs of the mixers added to give a directional output around the pilot frequency.

To explain this mathematically, let the quadrature pilot frequency signals be $P_i = \sin\omega_p t$, $P_q = \cos\omega_p t$. Consider again the input signals to be perfect quadrature signals as defined above:

* Hyperception, Inc., 9550 Skillman LB125, Dallas TX 75243, USA

$$I = A\cos\omega_a t + B\sin\omega_b t,\ Q = A\sin\omega_a t + B\cos\omega_b t$$

After mixing, the results are

$$X1 = I\cdot P_i = \sin\omega_p t(A\cos\omega_a t + B\sin\omega_b t) \quad (9)$$
$$= \frac{A}{2}\sin(\omega_p - \omega_a)t + \frac{A}{2}\sin(\omega_p + \omega_a)t$$
$$+ \frac{B}{2}\cos(\omega_p - \omega_b)t - \frac{B}{2}\cos(\omega_p + \omega_b)t$$

$$X2 = Q\cdot P_q = \cos\omega_p t(A\sin\omega_a t + B\cos\omega_b t) \quad (10)$$
$$= -\frac{A}{2}\sin(\omega_p - \omega_a)t + \frac{A}{2}\sin(\omega_p + \omega_a)t$$
$$+ \frac{B}{2}\cos(\omega_p - \omega_b)t + \frac{B}{2}\cos(\omega_p + \omega_b)t$$

If we add $X1$ and $X2$, the output is

$$OUT = X1 + X2 = A\ \sin(\omega_p + \omega_a)t + B\cos(\omega_p - \omega_b)t \quad (11)$$

This method has a single output, giving directional information around the pilot frequency signal which means that only one real FFT process is needed to produce the output spectra.

To confirm the results obtained above, a simulation of the system has been implemented using signal-processing functions of the DSP software package. The I and Q signals were created as for the PFT (eqns. 7 and 8).

A complex (quadrature) pilot frequency signal with the same sampling frequency and number of data points was created using the same DSP function. The frequency of the pilot signal was set at 4 kHz. The in-phase and quadrature components of the pilot frequency signal were

$$P_i = \sin(2\pi n 4000/16000),\ P_q = \cos(2\pi n 4000/16000) \quad (12)$$

These signals were mixed with the quadrature inputs and the outputs of the two mixers added. The result was an SSB modulated signal, or in other words an SSB frequency-shifted signal. The waveforms at each processing stage and at the output are illustrated in Fig. 5. The spectrum of the output of the system shows clearly that the output is shifted by the 4 kHz pilot signal frequency giving one spectral line at 5 kHz (4 + 1) and another at 2 kHz (4–2). Note that the lower SB is rejected for the 1 kHz signal and the upper SB is rejected for the 2 kHz signal.

2.3 *Complex FFT (CFFT)*

The CFFT is a straightforward technique for deriving directional data as no extra signal-processing is required to produce frequency domain information. A block diagram of the system is shown in Fig. 6. Quadrature input signals I and Q are taken as the real and imaginary parts of a complex time signal. Owing to the properties of the CFFT (BRIGHAM, 1974), if the complex input time signal is in quadrature (i.e. the phase difference between the real and imaginary parts is 90°), the output is directional and the output spectrum is dual-sided.

The complex Fourier Transform (CFT) has some very useful properties which allow the detection of the direction of blood flow when it is applied to quadrature signals. Consider a complex time signal; if its real part is even and its imaginary part odd, then the CFT is real. If its real part is odd and its imaginary part even, then the CFT is imaginary. In order to clarify these properties, let a

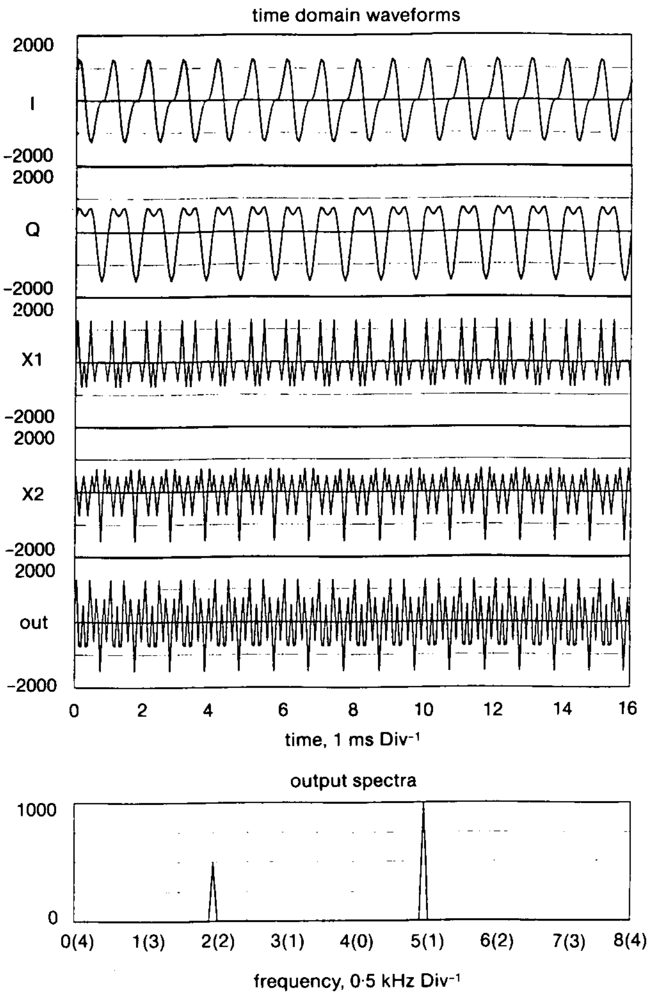


Fig. 5 *Time domain displays of each processing stage and output spectra for WRT (simulation results)*

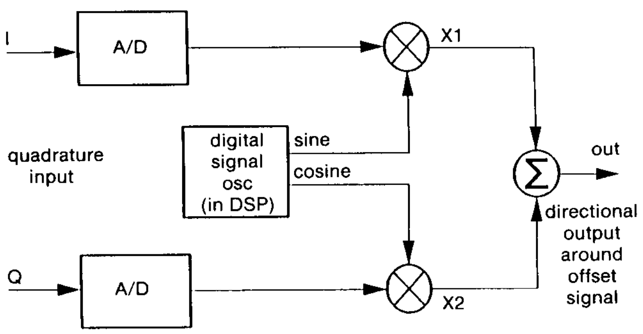


Fig. 4 *Weaver receiver technique (WRT) for use with quadrature phase detected signals*

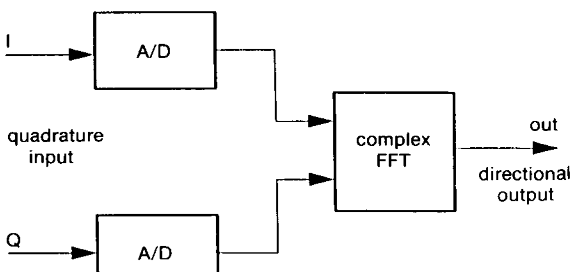


Fig. 6 *Block diagram of the complex FFT (CFFT) for use with quadrature phase detected signal*

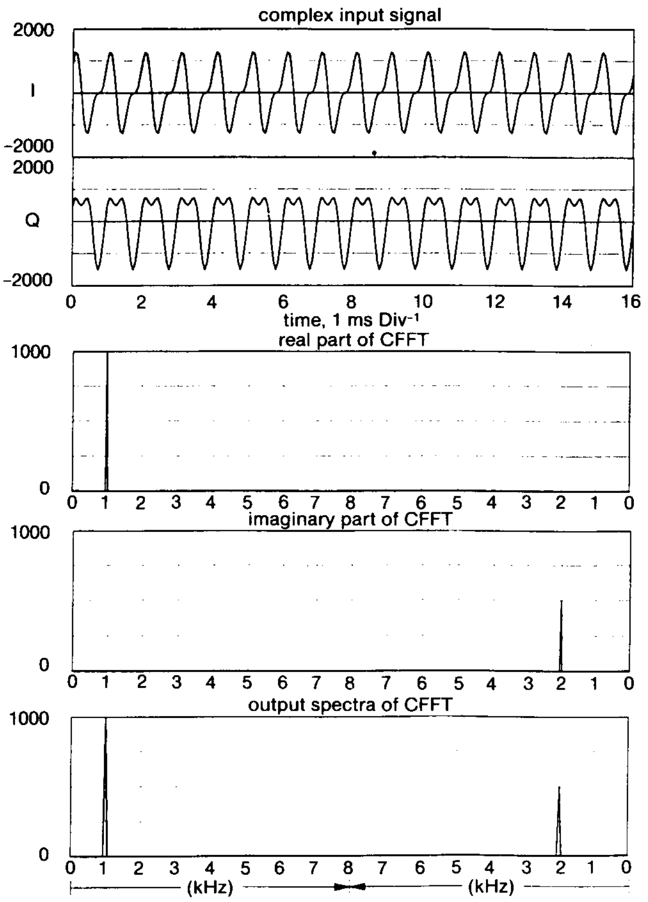


Fig. 7 *Real part, imaginary part and output spectra of the CFFT simulation when the input is complex quadrature signal*

complex time signal including both cases be

$$x(t) = x_a(t) + x_b(t) = x_r(t) + jx_i(t)$$

$$x_a(t) = (A \cos 2\pi f_a t + jA \sin 2\pi f_a t)$$

$$x_b(t) = (B \sin 2\pi f_b t + jB \cos 2\pi f_b t)$$

where x_a represents the first case and x_b represents the second case. Hence

$$x(t) = (A \cos 2\pi f_a t + B \sin 2\pi f_b t) + j(A \sin 2\pi f_a t + B \cos 2\pi f_b t) \quad (13)$$

The Fourier transform of this complex time function is

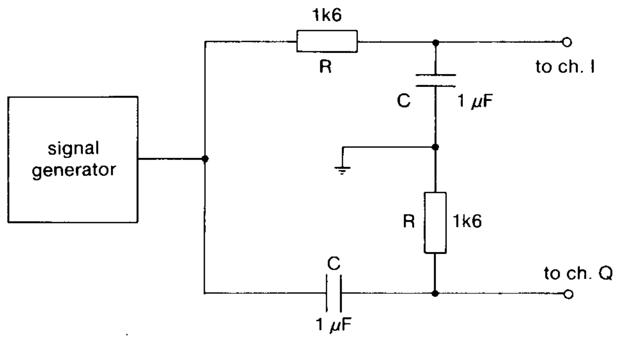


Fig. 8 *R-C network used to produce quadrature signal*

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} (x_r(t) + jx_i(t))e^{-j2\pi ft}dt \quad (14)$$

$$X(f) = \int_{-\infty}^{\infty} A \cos 2\pi f_a t e^{-j2\pi ft}dt + \int_{-\infty}^{\infty} B \sin 2\pi f_b t e^{-j2\pi ft}dt + j\int_{-\infty}^{\infty} A \sin 2\pi f_a t e^{-j2\pi ft}dt + j\int_{-\infty}^{\infty} B \cos 2\pi f_b t e^{-j2\pi ft}dt$$

$$X(f) = \frac{A}{2}\delta(f-f_a) + \frac{A}{2}\delta(f+f_a) - j\frac{B}{2}\delta(f-f_b) + j\frac{B}{2}\delta(f+f_b) + \frac{A}{2}\delta(f-f_a) - \frac{A}{2}\delta(f+f_a) + j\frac{B}{2}\delta(f-f_b) + j\frac{B}{2}\delta(f+f_b)$$

$$X(f) = A\delta(f-f_a) + jB\delta(f+f_b)$$

This result substantiates the above statement about CFT of complex quadrature signals.

In order to verify this result graphically, a complex time signal was created including both cases which satisfy the properties of CFT stated above. This is

$$X = I + jQ = \{1000 \cos(2\pi n 1000/16000) + 500 \sin(2\pi n 2000/16000)\} + j\{1000 \sin(2\pi n 1000/16000) + 500 \cos(2\pi n 2000/16000)\} \quad (15)$$

The CFFT was implemented using the FFT function of the DSP software package. The results obtained from this process are illustrated in Fig. 7. The output of the CFFT of the 1 kHz signal is real, illustrating the first property above; the output of the CFFT of the 2 kHz signal is imaginary, illustrating the second property.

The resultant output spectrum is dual-sided and directional. Unlike the other two methods, the separation in this method is purely in the frequency domain.

3 Practical implementations

The processing platform for the practical implementations was an LSI† DSP32C digital signal-processor board plugged into an 80386 based IBM-AT-compatible computer. This board is built around the AT&T DSP32C digital signal-processor chip, which features 32-bit floating point arithmetic, providing an overall dynamic range in excess of 1500 dB, 1536 × 32 bit on chip memory and an 80 ns instruction time (AT&T Microelectronics, 1990). In addition, two 16 bit A/D and two 16 bit D/A convertors are provided on the board. The total memory of the system is large enough to accommodate signal-processing software. An optional analogue Doppler signal processor that we designed may be used in conjunction with the LSI board. This board utilises a CW Doppler system with quadrature outputs and can be directly connected to the analogue inputs of the DSP board. We have, however,

† Loughborough Sound Images Limited, The Technology Centre, Epinal Way, Loughborough, Leics LE11 0QE, UK

Table 1 Specifications of the HT and DF used in the practical implementation

parameter	units	Hilbert transform	Delay filter
sampling frequency	Hz	16000	16000
centre frequency	Hz	4000	4000
bandwidth	Hz	7500	8000
transition BW1	Hz	250	0
transition BW2	Hz	250	0
stopband attenuation	dB	40	40
filter length	number of taps	37	37
passband ripple	dB	1.5	1.5

used a very simple quadrature signal generator to simulate a Doppler system with quadrature outputs in order to describe these digital techniques as simply as possible. A simple filter pair formed by RC high-pass and low-pass filters was implemented and a high-quality signal generator used. The system is illustrated in Fig. 8. At the cut-off frequency (3 dB point), the high-pass filter introduces +45° and the low-pass filter −45°. The total phase differ-

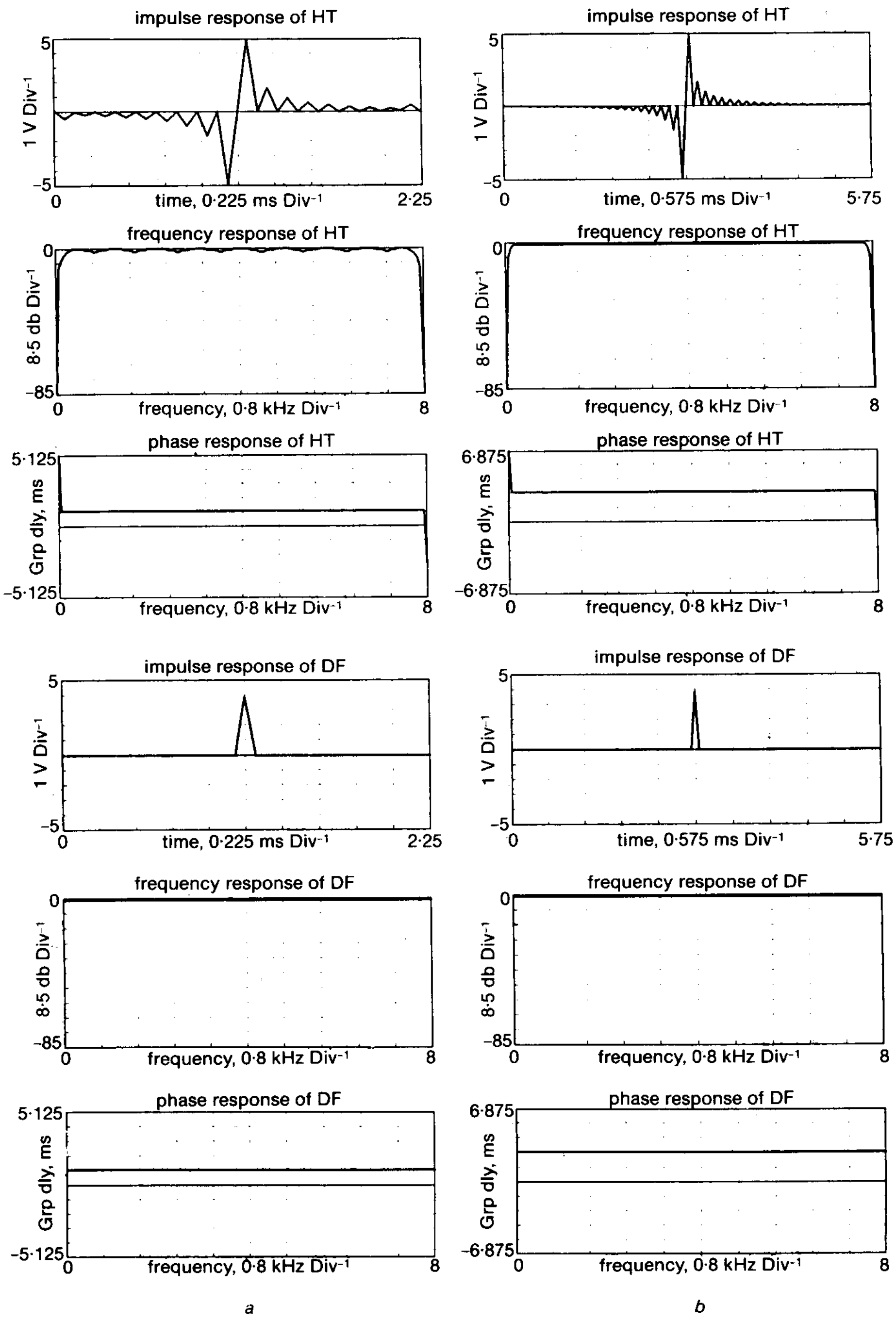


Fig. 9 *Impulse, frequency and phase responses of the Hilbert transform (HT) and delay filter (DF): (a) with 37 taps; (b) with 93 taps*

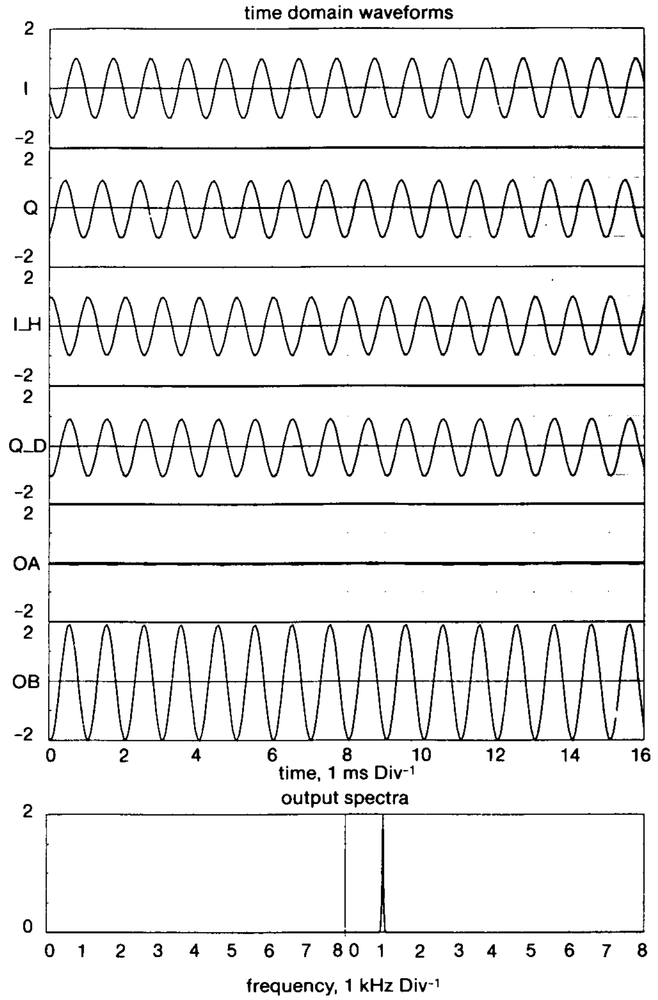


Fig. 10 *Time domain display of each processing stage and output spectra of the PFT implemented practically*

ence is 90°. The frequency was set to 1 kHz and the magnitude to 1 V.

All signal-processing software was written in DSP32C assembly language. The control software was written in C and the results displayed on the computer.

3.1 *Implementation of the phasing-filter technique*

The block diagram shown in Fig. 2 was implemented digitally using filters designed with the filter design option of Hypersignal DSP Software. Details of the design of HT are found elsewhere (Gold *et al.*, 1970; Oppenheim and Schafer, 1975; Rabiner and Schafer, 1974; Taylor, 1983). As mentioned above, a digital HT introduces a time delay. For the correct operation of a quadrature system, the delays in both channels must be equalised, and thus a DF was implemented using the same DSP package. Specifications of the HT and the DF used in our design are shown in Table 1.

The impulse, frequency and phase responses of this HT and DF are shown in Fig. 9*a*. Fig. 9*b* illustrates the responses of another HT and DF showing better specifications with 100 dB stopband attenuation and 0·5 dB passband ripple, but with more taps (93). In practice, no more than 70–75 dB is usually achievable. Our experiments indicate that the HT designed using the specifications listed in Table 1 is sufficient to provide a reasonable separation level, which is around 40 dB between two channels. In real-time applications, it is also important to keep computation time as short as possible, and so we preferred to use the minimum number of taps which produced the required specifications for the directional Doppler system.

Fig. 10 shows the actual output of the system illustrated in Fig. 2. This is an output of the six-channel real-time digital oscilloscope implemented in software on the personal computer. An additional display shows the output spectra of the system. I and Q are sinusoidal quadrature signals digitised by the on-board A/D convertors. The data length is 512 point for each channel. I_H and Q_D are the outputs of the HT and the DF, respectively. It can be seen clearly that the phase difference between I_H and Q_D is 180° because the HT filter introduces a 90° phase shift into the in-phase component I. The last two traces show the separated outputs in the time domain. Output OB is almost twice the input magnitude and OA is very small (at least 40 dB down). This practical result confirms the mathematical and simulation results presented above. The output spectra of the system plotted on a logarithmic scale are shown in Fig. 12*a*. Here the separation level is approximately 50 dB.

3.2 *Implementation of the Weaver receiver technique*

This method has been implemented in real time using 1024 data points for each channel, instead of 512, in order to use the full screen of the computer display after performing the real FFT. The quadrature data were captured by the LSI board and mixed with the digitally generated 4 kHz quadrature pilot frequency signal. The outputs of digital mixers $X1$ and $X2$ were then added. The output is directional around the 4 kHz pilot frequency signal. Here the process is similar to SSB modulation. The waveforms at all processing stages are illustrated in Fig. 11, where I and Q stand for the quadrature input signals, $P1$ and $P2$ for the quadrature pilot frequency signals, $X1$ and $X2$ for

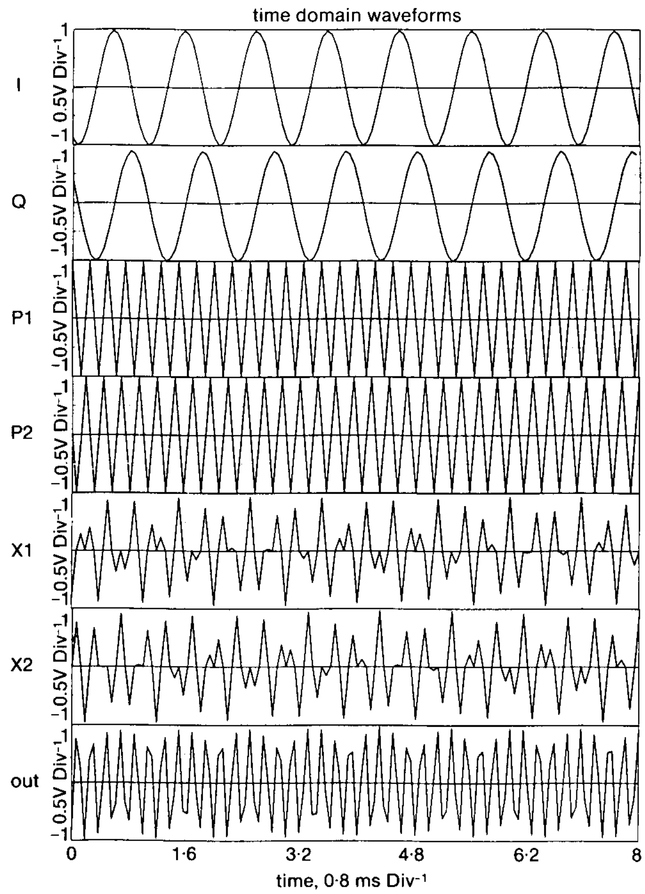


Fig. 11 *Time domain display of each processing stage and output of the WRT implemented practically*

the modulated signals and OUT for the SSB (direcitonal) output signal.

The number of samples for the pilot frequency signal should be equal to the number of samples of input quadrature signal. For example, if 256 data points are sampled by the A/D convertor, then 256 complex data samples must be generated for the pilot frequency signal. In order to minimise cross-talk and avoid possible aliasing, the pilot frequency signal should ideally be set to $F_s/4$. However, it is perfectly valid to use other frequencies, providing they do not cause aliasing which results in misinterpretation of the signal spectra.

Digital quadrature sinusoidal signal generators are more precise than their analogue counterparts, but their word length is finite which produces rounding errors that change both the amplitude and frequency of the generated signal. Improving the performance of digital sinusoidal oscillators has been investigated (ABU-EL-HAIJA and AL-IBRAHIM, 1986). Further discussions of digital sinusoidal oscillators can be found elsewhere (FURUNO *et al.*, 1975; FLIEGE and WINTERMANTEL, 1992).

The resultant output spectrum of this method is shown in Fig. 12*b*.

3.3 *Implementation of complex FFT*

A 512 point complex quadrature signal was digitised by the LSI DSP32C board and placed in the on-board memory. These data were then windowed using a complex Hanning window routine. Finally, the 512 point CFFT was implemented using a CFFT routine. The result is complex and directional, as explained above. The resultant data are rearranged for display on the screen and sent to the PC. In this method, the separation is performed in the frequency domain, and therefore there are no separated time domain outputs. The output of this system in the frequency domain is illustrated in Fig. 12*c*.

4 Results and discussion

Three different methods for the separation of forward and reverse flow signals in an ultrasonic Doppler system have been digitially implemented by software. The processing platform was a powerful 32-bit floating point digital signal-processor capable of processing Doppler signals at up to 150 kHz in real time.

The PFT is based on a HT, and the forward and reverse outputs are totally separated, which means they can be presented audibly as a stereo pair and unlike phase quadrature signals (which are extremely prone to phase distortions) can be separately recorded on a stereo tape recorder (SMALLWOOD, 1985; BUSH and EVANS, 1993). As the output consists of two separated channels, a dual-channel spectrum analyser is required, i.e. two real FFT routines must be implemented to obtain the output spectra. The output signal bandwidth is limited by the HT frequency response. In our case, the bandwidth is 7500 Hz

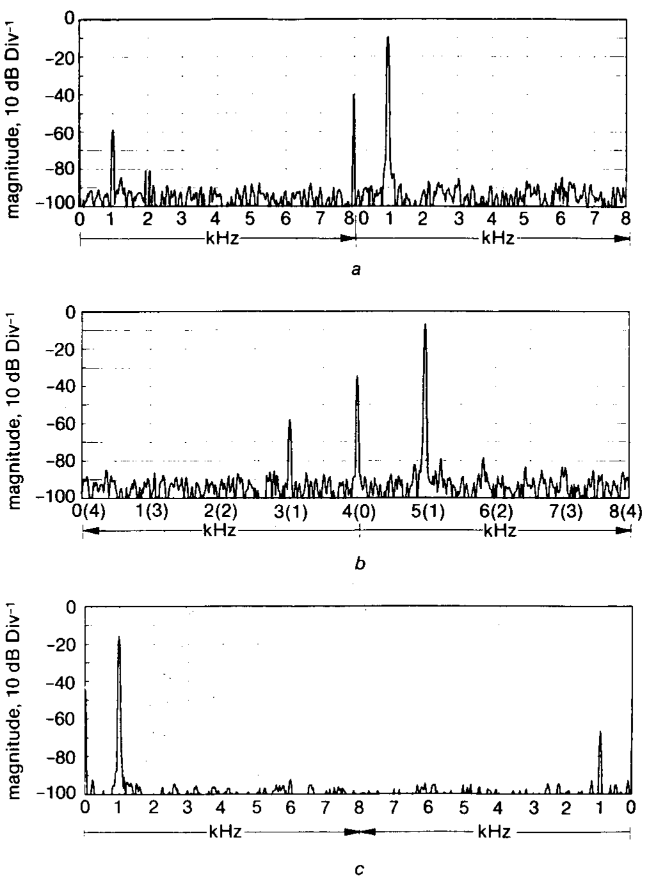


Fig. 12 *Comparative logarithmic scale output spectra of practical implementations for (a) the PFT, (b) the WRT and (c) the CFFT*

for each channel. Output spectra for this system are illustrated in Fig. 12*a*. Here, two channels were displayed side by side as one display. The execution time for the real FFT (including the routine call) is 0.976 ms for 512 data points. The total processing time required by the DSP chip to perform all processing tasks (two filters, two Hanning windows and two FFTs) is 13.25 ms.

The WRT is based on the SSB modulation method (WEAVER, 1956). The output is a lower sideband or an upper sideband around a fixed quadrature pilot signal frequency, depending on the combination of the real and imaginary parts of the quadrature input signal. The output is one channel and requires only one real FFT routine to obtain the output spectrum. The bandwidth for forward and reverse flow signals, however, is half of the full range ($F_s/4$). Fig. 12*b* illustrates the output spectrum for this system, when the input is a 1 kHz complex quadrature signal and the pilot signal frequency is 4 kHz. The execution time for the real FFT (including the routine call) is 2.085 ms for 1024 data points. The total processing time (including the signal generation, the mixing, the windowing and the FFT) is 10.3125 ms. Note that 1024 data

Table 2 Total processing times required by the DSP chip

data length	1024	512	256	128	64	
total	26.6875	13.25	6.5625	3.25	1.5625	PFT
processing	10.3125	5.00	2.4375	1.1875	0.5625	WRT
time, ms	10.50	5.0625	2.4375	1.1875	0.5625	CFFT

PFT = phasing filter technique, CFFT = complex FFT
WRT = Weaver receiver technique

sample were used to get this result. When 512 samples were used, the total processing time was 5 ms.

Fig. 12*c* shows the output spectrum of the CFFT when the input is a 1 kHz complex quadrature signal. Separation is totally in the frequency domain, and so no direct separated time domain output can be obtained using this method. However, further processing may be applied in order to obtain totally separated time domain outputs. The execution time for the CFFT routine (including the routine call) is 1.778 ms for 512 data points. The total processing time is 5.0625 ms.

The processing speeds of these techniques can be increased by decrasing the data length and optimizing the DSP32C source code. When 512 data samples were used for all implementations, the Weaver receiver technique was the fastest. Table 2 shows the total processing times required by the AT&T DSP32C processor to perform all real-time processing tasks for several data lengths.

All the methods discussed here perform well and give a separation level of at least 40 dB, which compares very favourably with analogue circuits even when considerable efforts have been made to optimise their directional separation. As DSP technology has been improving rapidly and getting cheaper, digital techniques are replacing their analogue counterparts. We have shown that implementing Doppler systems digitally using modern DSP processors is highly feasible and eliminates some difficulties associated with analogue systems.

Acknowledgment—This work was supported in part by SERC grant GR/600372, and by the Turkish Ministry of Education.

References

ABU-EL-HAIJE, A. I., and AL-IBRAHIM, M. M. (1986): 'Improving performance of digital sinusoidal oscillators by means of error feedback circuits', *IEEE Trans.*, **CAS-33**, pp. 373–380

ANSARI, R. (1987): 'IIR discrete-time Hilbert transformers', *IEEE Trans.*, **ASSP-35**, pp. 116–119

BRIGHAM, E. O. (1974): 'The fast Fourier transform' (Prentice-Hall, Inc., Englewood Cliffs, New Jersey) pp. 11–49

BUSH, G., and EVANS, D. H. (1993): 'Digital audio tape as a method of storing Doppler ultrasound signals', *Physiol. Meas.* **14**, pp. 381–386

COGHLAN, B. A., and TAYLOR, M. G. (1976): 'Directional Doppler techniques for detection of blood velocities', *Ultrasound Med. Biol.*, **2**, pp. 181–188

FLIEGE, N. J., and WINTERMANTEL, J. (1992): 'Complex digital oscillators and FSK modulators', *IEEE Trans.*, **SP-40**, pp. 333–342

FURUNO, K., MITRA, S. K., HIRANO, K., and ITO, Y. (1975): 'Design of digital sinusoidal oscillators with absolute periodicity', *IEEE Trans.*, **AES-11**, pp. 1286–1299

GOLD, B., OPPENHEIM, A. V., and RADER, C. M. (1970): 'Theory and implementation of discrete Hilbert transform'. *Proc. Symp. Comput. Process. Comm.* (Polytechnic Press), pp. 235–250

OPPENHEIM, A. V., and SCHAFER, R. W. (1975): 'Digital signal processing' (Prentice-Hall, Inc., Englewood Cliffs, New Jersey) pp. 337–366

RABINER, L. R., and SCHAFER, R. W. (1974): 'On the behaviour of minimax FIR digital Hilbert transformers', *Bell Syst. Tech. J.*, **53**, pp. 363–389

SCHWARTZ, M., BENNETT, W. R., and STEIN, S. (1966): 'Communication systems and techniques' (McGraw-Hill, Inc.) pp. 29–35

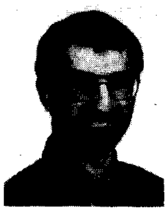
SMALLWOOD, R. H. (1985): 'Recording Doppler blood flow signals on magnetic tape', *Clin. Phys. Physiol. Meas.*, **6**, pp. 357–359

TAYLOR, F. J. (1983). 'Digital filter design handbook' (Marcel Dekker, Inc., New York and Basel), pp. 183–188

WEAVER, D. K. JR. (1956): 'A third method of generation and detection of single-sideband signals', *Proc. IRE*, **44**, pp. 1703–1705

AT&T Microelectronics (1990): WE DSP32C Digital Signal Processor Information Manual

Author's biography



N. Aydin was born in Gemlik, Turkey, in 1962. He obtained his BSc in Electronics and Communication Engineering in 1984, and his MSc in Electronics in 1987, from Yildiz University, Turkey. Between 1986 and 1989, he worked as a Research Associate in Electrical and Electronics Measurements and Systems Lab, at Yildiz University. At present, he is finalising his research work on blood flow measurements for his PhD, in the Medical Physics Division of the Faculty of Medicine, University of Leicester. His research interest includes analogue and digital signal processing, patient monitoring systems and biomedical instrumentation.