## APPENDIX A – DIGITAL FILTER FOR RECEIVED SIGNALS

As discussed in section 4.4.1.3, there were occasions during the measurement of the shear wave velocity using bender elements when the received signal of the shear wave was difficult to interpret due to electrical noise. In these cases, a digital filter was used to filter high and low frequency electrical noise. Based on an approach used by Filz (1992), the signals were filtered using the program MathCAD. An example of the MathCAD file used to digitally filter the received signals is shown in this appendix.

## Digital Filtering of Shear Wave Velocity Data - page 1

The data file (**.dat) obtained from the Tektronix Oscilloscope is in the following format:

| | |
|---|---|
| 0 | N.....Number of data points in record |
| 1 | dt.....timestep between data points |
| 2 | 50.....no. data pt. which was the trigger |
| 3 | 0.49.....trigger offset |
| 4-503 | .....data points |

$N := READ(c325)$

$N = 500$

Reads the first no. in file ( ).

$a := 0..N+3$ $b := 4..N+3$

$x_a := READ(c325)$

Reads the contents of file into the vector x.

$dt := x_1$ $c := 504..515$

dt is the timestep from the record.

$x_c := 0.00236$ $d := 4..515$

$dt = 4 \cdot 10^{-6}$ $i := 0..511$

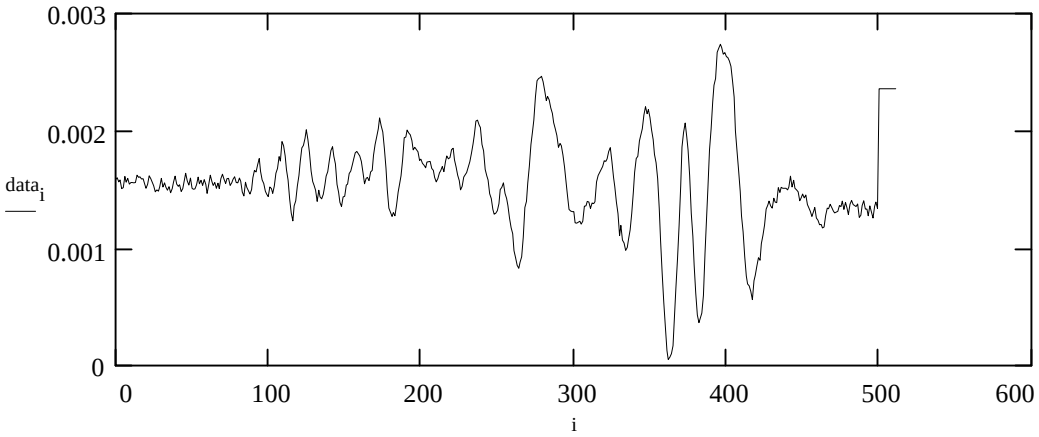
Zeros are added to the end of the record to make a total of 512 data points (req'd for the FFT).

$WRITE(temp) := x_d$

$data_i := READ(temp)$

The 512 data point file is written to the file chris.dat.

### Unfiltered Received Signal




The signal in the time domain is converted to the frequency domain by performing a Fast Fourier Transform.

$ft := fft(data)$ Fourier transform of the time signal.

$n := 512$

$j := 0..\frac{n}{2}$ $freq_j := \frac{j}{n \cdot dt}$ Frequency content (obtained from timestep)

## Digital Filtering of Shear Wave Velocity Data - page 2

Filters are applied to the frequency domain. The lowpass filter lets frequencies below the value pass through. The highpass filter lets frequencies above the value pass through. The bandpass filter lets only a range of frequencies pass through.

$lowpass := 11000 \quad highpass := 10000 \quad bandlow := 11000 \quad bandhigh := 1000$

$ft1_j := ft_j \cdot \Phi(lowpass - freq_j) \qquad ft2_j := ft_j \cdot \Phi(freq_j - highpass)$

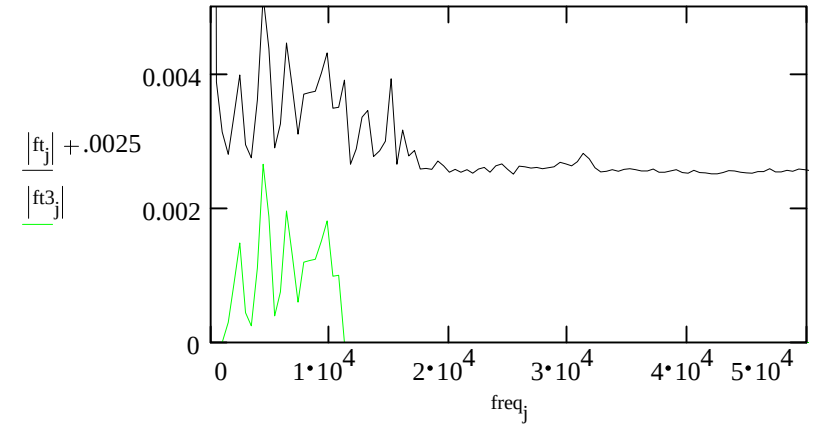
$ft3_j := ft_j \cdot \Phi(bandlow - freq_j) \cdot \Phi(freq_j - bandhigh)$

The filtering is carried out using a Heaviside step function (phi). This function is equal to 1 when ( )>0 and equal to 0 otherwise.

$low := ifft(ft1) \qquad high := ifft(ft2) \qquad band := ifft(ft3)$

### Plot of Frequency Domain

$WRITEPRN(chris) := band_i$




Raw data (black)
Bandpass (green)
Highpass (blue)
Lowpass (red)

### Filtered Data