DIXIEME COLLOQUE SUR LE TRAITEMENT DU
SIGNAL ET SES APPLICATIONS

NICE du 20 au 24 MAI 1985

DESIGN OF AN IIR-FILTER BANK FOR DUAL TONE RECEIVERS

**Kari Järvinen, Markku Renfors,
Tapio Saramäki, and Yrjö Neuvo**

**Department of Electrical Engineering, Tampere University of Technology
P.O. Box 527
SF - 33101 Tampere, Finland**

RESUME

Un concept nouveau d'un banc de filtres, symétrique à l'égard d'une certain fréquence nominal est presenté. Par le placement de cet centre de symmétrie à $\omega = \pi/2$ pouvons nous effectivement utiliser les filtres demibandes, decimation par facteur 2, et en effet réaliser seulement la moitié inférieure de ces filtres. Le résultat de ceci est un banc des filtres qui ne demande que un petit nombre de multiplicateurs. En outre, la longueur du programme pour réaliser ce banc de filtres a été substantiellement abregé et la vitesse constante du courant des données par le banc de filtres a été préservé.

Comme un example la réalisation d'un banc de 4 filtres pour les récepteurs du ton double a été étudé. Le banc de filtres a été réalisé en entier par une unité de traitement des signaux Intel 2920 avec la longeur de programme 138 instructions, en laissante 54 instruction pour les choses diverses. 34 de 40 emplacements de mémoire lecture ont été utilisé. Une réalisation conventionelle aurait pris approximativement tours les ressources de la unité de traitement.

SUMMARY

A novel design of a filter bank that is symmetric with respect to a certain center frequency is presented. By placing this center of symmetry at $\omega = \pi/2$ we can effectively utilize half-band filters, decimation by factor of 2, and actually implement only the lower half of the filters. This results in a filter bank that requires only a small number of multipliers. In addition, the length of the program required to implement this filter bank in a signal processor is substantially shortened and a constant data flow rate through the filter bank is preserved.

As an example, the implementation of a 4 filter bank for dual tone receivers is considered. This filter bank has been implemented with one Intel 2920 signal processor with a program length of 138 instructions leaving 54 instructions for other purposes. 34 out of the available 40 RAM locations are used. A conventional 4 filter implementation would have taken approximately all the resources of the processor.

1. INTRODUCTION

In this paper we consider the design and implementation of a specific class of filter banks in which the channels are located symmetrically with respect to a certain center frequency. When implementing this kind of filter banks using digital filters, it is advantageous to select the sampling rate so that the center of symmetry becomes equal to quarter of the sampling rate. With this selection, the higher and lower signal bands can be separated using a pair of lowpass and highpass filters combined with sampling rate reduction by a factor of two. Due to the inherent frequency translation properties of decimation, the higher and lower groups of filters have, at the lower sampling rate, the same specifications. A symmetrically located bandpass filter pair can then be implemented efficiently by multiplexing a single filter.

The bandsplit filter/decimator can be implemented very effectively by using a pair of complementary half-band filters [1] or half-band lattice wave digital filters [2 – 4]. In practice, it takes only a minor part of the computational workload.

This approach has several benefits compared to direct implementation of the bandpass filters: slower processor can be used because the bandpass filters operate at the lower sampling rate, the required coefficient memory is reduced, and only half of the bandpass filters have to be designed. Furthermore, because the relative bandwidths of the bandpass filters are doubled, the finite wordlength effects (coefficient quantization, roundoff noise, and limit cycles) are reduced.

An important application for this type of filter banks can be found in dual tone multifrequency (DTMF) receivers [5]. In the following, we discuss the design and implementation of a four-channel DTMF filter bank using a single Intel 2920 signal processor.

2. THE FILTER BANK

In the specific type of filter banks we are considering, the number of filters N is even. The filters $H_1(z)$, $H_2(z)$, ..., $H_{N/2}(z)$ with center frequency lower than $\pi/2$ are called the lower group and the filters $H_{N/2+1}(z), \ldots, H_N(z)$ with center frequency higher than $\pi/2$ are called the higher group. The frequency responses of the lower and higher group filters must satisfy (see Fig. 1)

$$|H_i(e^{j\omega})| = |H_{N+1-i}(e^{j(\pi-\omega)})|, \quad i = 1, \ldots, N/2. \quad (1)$$

Fig. 2 shows the block diagram of such a filter bank. The band-split filter separates the frequency band of the lower (higher) group through lowpass (highpass) filtering. The sampling rate of the output signals of these filters is then reduced by a factor of two. The actual bandpass filters satisfy the frequency specifications of the lower group filters.

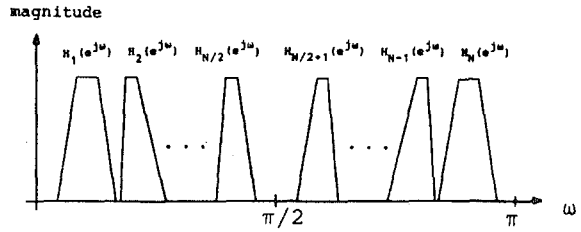


Fig. 1. Example of a symmetric filter bank.

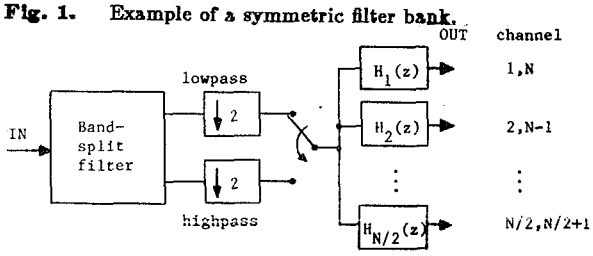


Fig. 2. Block diagram of the filter bank.

The Band-Split Filter

In general, a signal $x(n)$ after it has been decimated by a factor of two contains not only the baseband but also the aliased components due to the sampling rate reduction. The Fourier transform of the decimated signal is given by [6]

$$Y(e^{j\omega'}) = 1/2\{H(e^{j\omega'/2})X(e^{j\omega'/2}) + H(e^{j(\omega'-2\pi)/2})X(e^{j(\omega'-2\pi)/2})\} \quad (2)$$

where $H(z)$ is the transfer function of the decimation filter and ω' is the normalized frequency variable at the lower sampling rate.

For reducing the sampling rate of lowpass signals by a factor of two and to avoid aliasing at the lower sampling rate, it is necessary to filter the original signal before sampling rate reduction with a filter that approximates the ideal digital lowpass characteristics specified by

$$H(e^{j\omega}) = \begin{cases} 1, & |\omega| < \frac{\pi}{2} \\ 0, & \frac{\pi}{2} \leq |\omega| < \pi. \end{cases} \quad (3)$$

The spectrum of the filtered and decimated signal is then given by

$$Y(e^{j\omega'}) = 1/2X(e^{j\omega'/2}). \quad (4)$$

In the case of sampling rate reduction of highpass signals by a factor of two, the function of the decimation filter is to remove the baseband components and pass the aliasing components for further processing. The ideal digital filter isolating the frequency range of interest is specified by

$$\hat{H}(e^{j\omega}) = \begin{cases} 0, & |\omega| < \frac{\pi}{2} \\ 1, & \frac{\pi}{2} \leq |\omega| < \pi. \end{cases} \quad (5)$$

The spectrum of the highpass filtered and decimated signal containing only the aliased components becomes

$$Y(e^{j\omega'}) = 1/2X(e^{j(\omega'-2\pi)/2)}). \quad (6)$$

We can see that, in the filter bank of Fig. 2, the frequency band of a higher group filter H_{N+1-i} is translated down to the frequency band of the lower group filter H_i.

Both the lowpass and highpass decimation filters can be implemented very efficiently using a lattice wave digital filter (WDF) [4]. Because in lattice WDF's not only their transmittances but also their reflectances are available as transfer functions, a single lattice WDF can perform the whole frequency band division into two halves. In other words, the transfer function of a lowpass lattice WDF [2] can always be expressed as a sum of two allpass transfer functions $A(z)$ and $B(z)$,

$$H(z) = \frac{1}{2}(A(z) + B(z)). \tag{7}$$

The corresponding highpass transfer function is obtained through

$$\hat{H}(z) = \frac{1}{2}(A(z) - B(z)). \tag{8}$$

These two transfer functions are complementary in the sense that

$$|H(e^{j\omega})|^2 + |\hat{H}(e^{j\omega})|^2 = 1. \tag{9}$$

In this case we can utilize a half-band lattice WDF [2] which has the further advantage that approximately half of the filter coefficients are zero. This results from the fact that the allpass transfer functions are of the form

$$A(z) = \hat{A}(z^2) \tag{10a}$$

$$B(z) = z^{-1}\hat{B}(z^2). \tag{10b}$$

Furthermore, sampling rate reduction by a factor of two can be combined very efficiently with a half-band lattice WDF [3]. As shown in Fig. 3, operations of filtering and sampling rate reduction may be commuted so that all the computations are carried out at the lower sampling rate.

3. DTMF FILTER BANK

The purpose of the DTMF subsystem is to measure the signal power within four 10 Hz wide frequency bands centered at the frequencies of 780 Hz, 900 Hz, 1020 Hz and 1140 Hz. The maximum allowable passband ripple in each band is 1.0 dB and the minimum attenuation is 37.5 dB in the stopbands starting 60 Hz away from the center frequencies.

In the DTMF filter bank, the center of symmetry is 960 Hz. Consequently, the input sampling rate is selected to be 3840 Hz, and the bandpass filters operate at 1920 Hz sampling rate. The block diagram and ideal frequency characteristics of the DTMF filter bank are shown in Fig. 4. We only need to implement the 780 Hz and 900 Hz bandpass filters. The sampling rate reduction causes spectrum inversion so that the 1020 Hz and 1140 Hz bands can be processed by the 900 Hz and 780 Hz filters, respectively.

It turns out that, due to the limited program memory of the 2920 signal processor and because no subroutine calls can be used [7], the length of the program is one of the most critical factors in our application. Consequently,

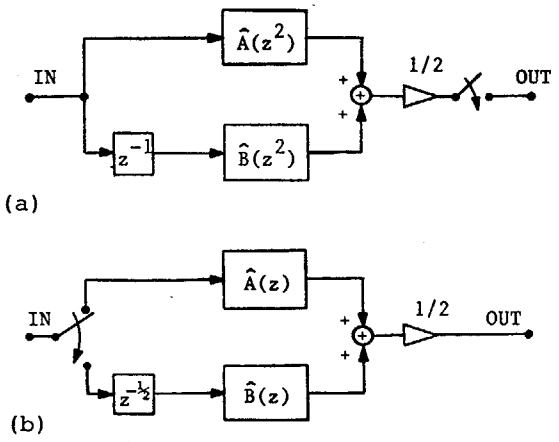


Fig. 3. Equivalent realizations of decimation by 2 using half-band filters. (a) Higher rate. (b) Lower rate.

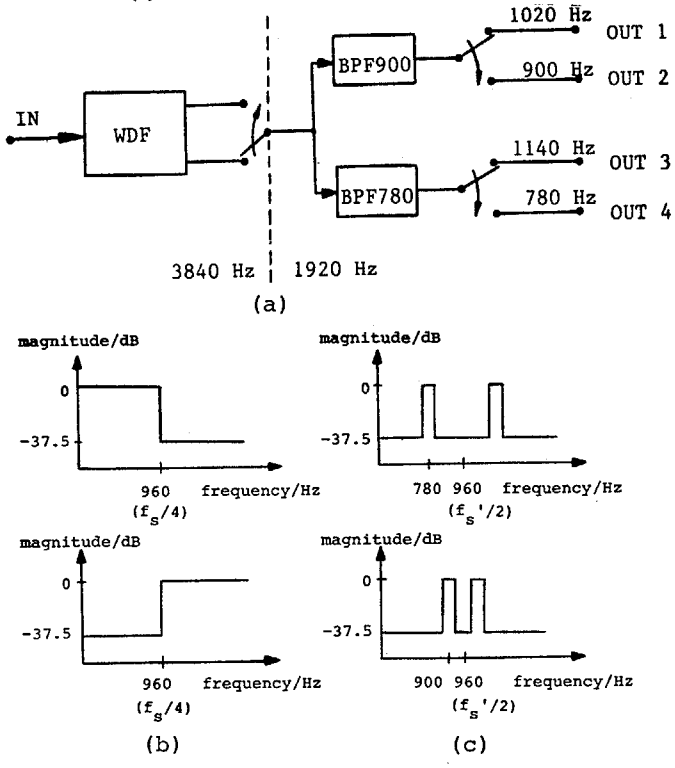


Fig. 4. Realization with sampling rate reduction. (a) Block diagram. (b) Ideal characteristics of the decimation filter. (c) Ideal characteristics of the bandpass filters.

the program cycle time is designed to be twice the output sampling period (*i.e.*, equal to the input sampling period) because then the multiplexing of the bandpass filters can be done without repeating the code which implements them. As will be discussed later, the band-split filter can not be implemented in the most efficient way, but this selection gives the highest overall efficiency.

Filter Approximation

It is sufficient to design only the lowpass part of the band-split filter and the lower group bandpass filters. The structure of the filter bank and properties of half-band filters guarantee that the upper group channels will have appropriate frequency responses if the lower group channels have.

The lowpass part of the band-split filter is designed to have equiripple stopbands at the passbands of the higher group and equiripple passbands at the passbands of the lower group filters. When the stopband attenuation of the band-split filter is at least 37.5 dB, the passband (775 – 905 Hz) ripple is insignificant (in the order of 0.0008 dB) due to the properties of half-band filters. A seventh order (the order must be odd [1, 2]) half-band filter satisfies the spesifications of the band-split filter.

This filter has one pole at the origin and the transfer function is of the form

$$H(z) = \frac{N(z)}{D(z)} = \frac{\sum_{i=0}^{7} a_i z^{-i}}{\sum_{i=0}^{6} b_i z^{-i}}. \qquad (11)$$

$N(z)$ and $D(z)$ were determined such that

(i) $|H(e^{j2\pi fT})|^2$ alternatingly goes through the value 0 and the stopband maximum value at seven points on [1015 Hz, 1025 Hz] $\cup$ [1135 Hz, 1145 Hz] and takes the value 0 at f=1920 Hz.

(ii) $|H(e^{j2\pi fT})|^2$ alternatingly goes through the value 1 and the passband minimum value at the seven points on [775 Hz, 785 Hz] $\cup$ [895 Hz, 905 Hz] and takes the value 1 at f=0 Hz.

(i) ensures that N(z) is a mirror-image polynomial (three zero pairs on the unit circle and one zero at $z = -1$) and (ii) ensures that the numerator polynomial of the complementary highpass design is a anti-mirror-image polynomial (one zero at z=1 and three zero pairs on the unit circle.) This quarantees that the filter satisfying (i) and (ii) is implementable as a sum of two allpass filters [8]. The actual design of the lowpass filter was performed using algorithms described in [9].

The bandpass filters are designed to satisfy the frequency specification of the two lower group channels. They must also provide enough attenuation in the frequency bands 960 – 1015 Hz and 1145 – 1200 Hz so that aliasing to the transition bands of the lower group filters is at acceptable level. The upper stopband edge frequency of the 900 Hz bandpass filter is designed to be only 30 Hz away from the center frequency in order to facilitate the frequency band division. The bandpass filter specifications can be met with fourth-order elliptic designs.

Implementation of the Band-Split Filter

The seventh-order lattice WDF used as the band-split filter is shown in Fig. 5. Only three adaptors are needed and the adaptor coefficients have been quantized to 6 bits. The lowpass and highpass frequency responses of the band-split filter (with quantized coefficients) is shown in Fig. 6.

In the structure of Fig. 5, all computations are carried out at 1920 Hz sampling rate. However, we have selected a computational cycle time which corresponds to 3840 Hz sampling rate and, on the other hand, in the 2920 signal processor it is not possible that the sampling period of some part of the system is higher than the computational period because no jump instructions are available.

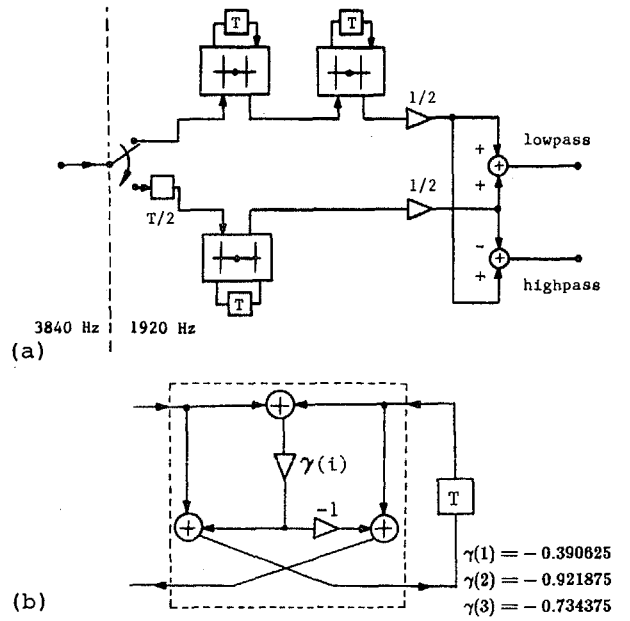


Fig. 5. Implementation of the band-split filter. (a) Block diagram. (b) Signal flow diagram of two-port adaptor with a unit delay.

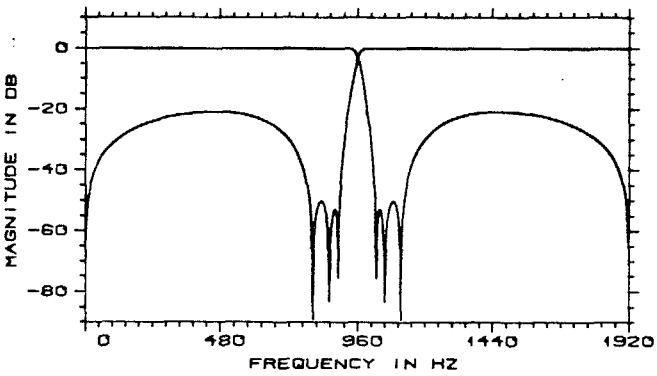


Fig. 6. Magnitude responses of the wave digital filter.

There is not any inconvenience in operating also the decimation filter with the same higher rate, *i.e.*, according to to Fig. 3(a). The advantage of the lower operation rate implementation comes out by the fact that it requires fewer delays, *i.e.*, fewer data memory locations are required for the realization. From the programmers point of view, it would be inconvient to implement the lower operation rate filter straight according to Fig. 5 since the operation of the branches should be prevented during every other program round. However, we notice that it is not necessary to prevent all of the operations from occuring during every other program cycle since it is sufficient if changes in the state variables are prevented. This leads to the structure of Fig. 7. All the computations are carried out at the higher rate but the actual filtering is performed at the lower rate. It has been possible to replace one of the unit delays connected to the adaptors by switches. Therefore, instead of the 7 RAM-memory locations required for realizing the delays of the higher operation rate structure, the implementation of Fig. 7 requires only 5 RAM-memories for that purpose, 3 for the adaptor delays and 2 for carrying information between program cycles.

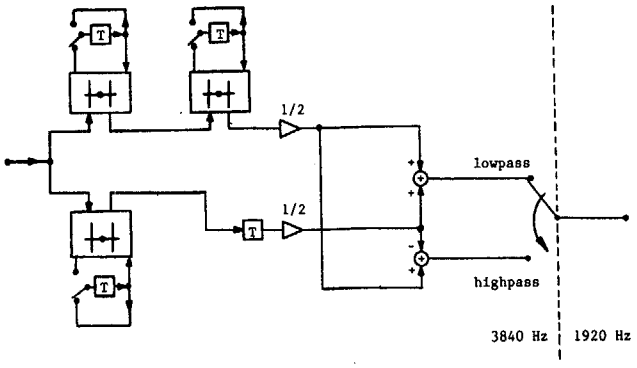
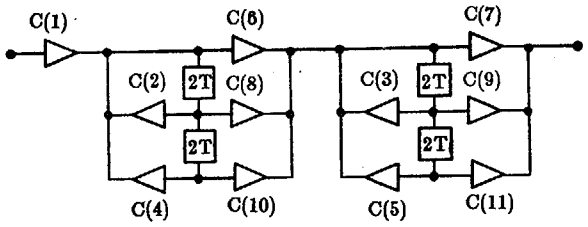


Fig. 7. Block diagram of the decimation filter realization that is chosen to be implemented with the 2920 processor.



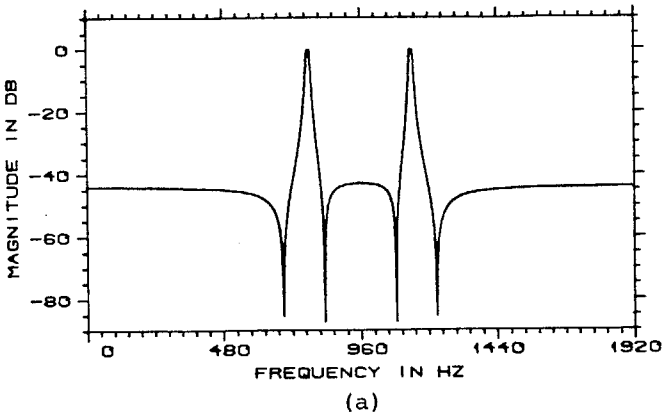
The 780 Hz bandpass filter:

$C(1) = 0.6347656250D - 02$
$C(2) = -0.1662109375D + 01$
$C(3) = -0.1625000000D + 01$
$C(4) = -0.9785156250D + 00$
$C(5) = -0.9765625000D + 00$
$C(6) = 0.3750000000D + 00$
$C(7) = 0.2750000000D + 01$
$C(8) = 0.6875000000D + 00$
$C(9) = 0.3500000000D + 01$
$C(10) = 0.3750000000D + 00$
$C(11) = 0.2750000000D + 01$

The 900 Hz bandpass filter:

$C(1) = 0.4882812500D - 02$
$C(2) = -0.1929687500D + 01$
$C(3) = -0.1947265625D + 01$
$C(4) = -0.9755859375D + 00$
$C(5) = -0.9794921875D + 00$
$C(6) = 0.2500000000D + 00$
$C(7) = 0.8000000000D + 01$
$C(8) = 0.4980468750D + 00$
$C(9) = 0.1400000000D + 02$
$C(10) = 0.2500000000D + 00$
$C(11) = 0.8000000000D + 01$

Fig. 8. Block diagram of the fourth order IIR filter with additional delays used for processing two signals alternatingly.



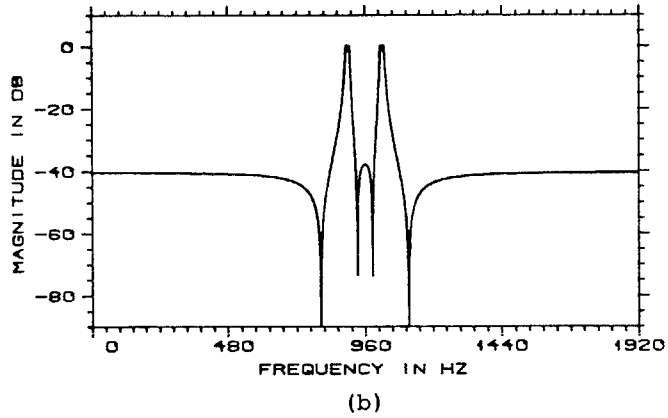


Fig. 9. Magnitude responses of the bandpass filters. (a) The 780 Hz bandpass filter. (b) The 900 Hz bandpass filter.

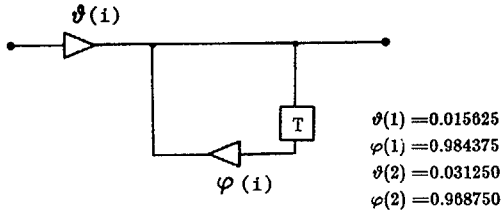


Fig. 10. Block diagram of the lowpass filters LPF1 and LPF2.

The Bandpass Filters

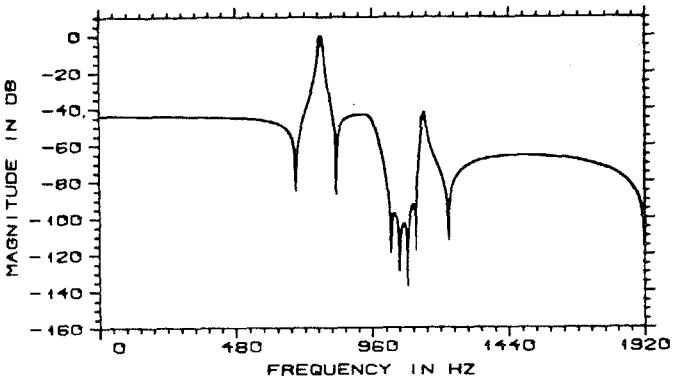
Both bandpass filters are fourth order elliptic IIR filters requiring 11 multipliers. They are implemented as a cascade of two second order sections (Fig. 8). By replacing each delay by two delays, one filter can be used to process two signals alternatingly. The filter coefficients have been rounded to 15 bits.

Better finite wordlength performance could be achieved by using lattice WDF's also for the bandpass filters. The problem is that higher filter orders would be required, since a bandpass lattice WDF must have transmission zeros at ω=0 and at $\omega = \pi$. In this case the filter order would be six, requiring altogether eight additional RAM memory locations for the bandpass filters. On the other hand, the finite wordlength performance of second order sections is sufficient for our application.

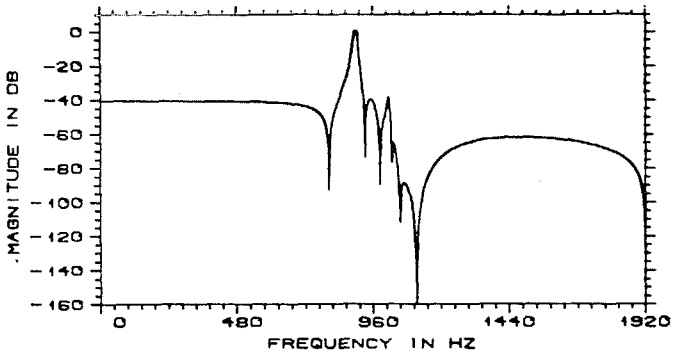
The magnitude responses of the whole filter bank are shown in Fig. 11.

The Power Measurement

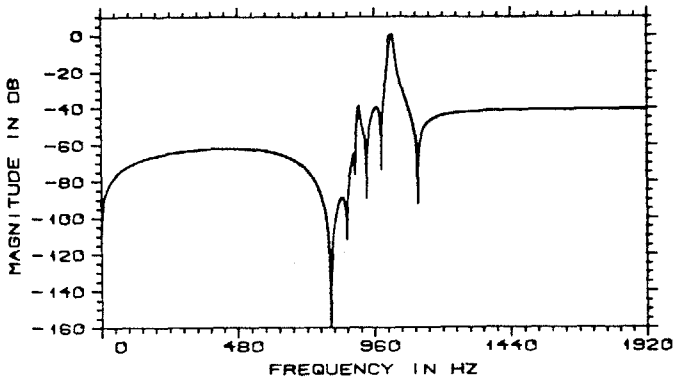
The power in each one of the four frequency bands of interest is measured by full wave rectifying the signals after they have been bandpass filtered and then removing the AC-components by recursive single pole lowpass filters LPF1 and LPF2 shown in Fig. 10. These filters have been designed to eliminate the most significant frequency components from the full wave rectified output signals, 120 Hz at the output of the 900 Hz bandpass filter and 360 Hz at the output of the 780 Hz bandpass filter. These frequency components are attennuated by about 30 dB. Consequently, the DC-component, which brings information about the signal amplitude, is about 33 dB above the most significant AC-component. The narrower we make the band for the lowpass filter, the slower the system becomes, forcing us to make a compromise between speed and accuracy.



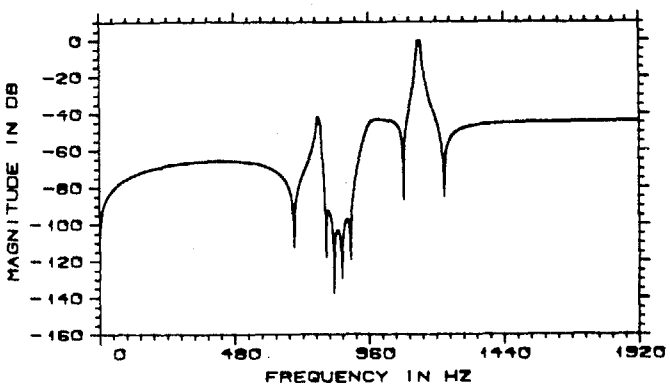
(a)



(b)



(c)



(d)

Fig. 11. Magnitude responses of the whole filter bank. (a) The 780 Hz channel. (b) The 900 Hz channel. (c) The 1020 Hz channel. (d) The 1140 Hz channel.

4. CONCLUSIONS

The overall power measurement system, Fig. 12, has been implemented with one Intel 2920 Analog Signal Processor. The system requires 138 program memory locations leaving 54 instructions for other purposes. A conventional 4 filter implementation would have taken approximately all the program resources of the processor. Altogether 34 out of the available 40 RAM locations are used.

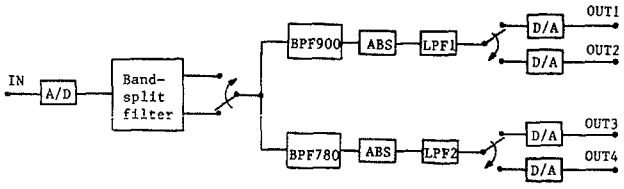


Fig. 12. Block diagram of the whole one chip system.

5. REFERENCES

[1] Y. Neuvo and S. K. Mitra, "Complementary IIR digital filters," *Proc. IEEE Int. Symp. Circuits Syst.* (Montreal, Canada), pp. 234–237, May 1984.

[2] W. Wegener, "Wave digital directional filters with reduced number of multipliers and adders," *Arch. Elek. Ubertragung.*, vol. 33, pp. 239–243, June 1979.

[3] J. A. Nossek and H.-D. Schwartz, "Wave digital lattice filters with applications in communication systems," *Proc. IEEE Int. Symp. Circuits Syst.* (Newport Beach, CA), pp. 845–848, May 1983.

[4] L. Gazsi, "Single chip filter bank with wave digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 709–718, Oct. 1982.

[5] L. Gazsi and H. Dawei, "DTMF band split filters realized by wave digital filters," *Proc. ECCTD'83* (Stuttgart, FRG), pp. 221–223, Sept. 1983.

[6] R. E. Crochiere and L. R. Rabiner, "Interpolation and decimation of digital signals – A tutorial review," *Proc. IEEE*, vol. 69, pp. 300–331, March 1981.

[7] *2920 Assembly Language Manual.* Santa Clara, CA: Intel Corp., 1979.

[8] T. Saramäki, "On the design of digital filters as a sum of two allpass filters," submitted for publication in Electronics Letters.

[9] T. Saramäki, "Design of digital filters requiring a small number of arithmetic operations," Dr. Tech. dissertation, Dep. Elec. Eng., Tampere University of Technology, Finland, May 1981.