A DESIGN METHOD FOR HALF-BAND FILTERS*

A. N. Willson, Jr. and H. J. Orchard
Electrical Engineering Department, UCLA
405 Hilgard Avenue
Los Angeles, CA 90095-1594 USA
Tel: (310) 825-7400; fax: (310) 206-4061
e-mail: willson@ee.ucla.edu hjo@ea.ucla.edu

ABSTRACT

Earlier methods of design have used iterative approaches such as the well-known Parks-McClellan algorithm or some variant of linear programming. Here we give a direct method of design, using Chebyshev polynomials, which provides a reduction in design time over previous methods by about a factor of ten. The ideal equal-ripple response in both passband and stopband is not achieved exactly, but the error is extremely small and would normally be undetectable in practical realizations.

1 INTRODUCTION

In the implementation of multirate digital systems an important fundamental building block is the half-band filter. It is, of course, well known that half-band filters can easily be designed by simply providing appropriately symmetrical specifications to the popular Parks-McClellan optimal-FIR filter design program, which uses the Remez algorithm. The program then computes the known zero-valued tap coefficients and the center coefficient of value 0.5, along with the unknown tap values that are interleaved with them, as if all the values were unknown at the outset. The errors in the values given by the program for the known coefficients clearly illustrate the (surprisingly large, we thought) amount by which the filter coefficients computed by the Parks-McClellan program are in error.

In [1] (see also problem 4.30 of [2]) a clever method is described for modifying the filter specification that is given to the Parks-McClellan program so that initially it designs a filter of about half the required length. From this the desired half-band filter can be created by simply interleaving the known zero-valued even-index coefficients with the Parks-McClellan-computed odd-index coefficients, scaled by a factor of 0.5, and adding a center-tap coefficient of 0.5. Since the computational requirements of the Parks-McClellan algorithm grow with filter length at a superlinear rate, the design time for computing the tap values when using the *trick* described in [1] proves to be smaller than that for a direct use of the Parks-McClellan program by about a factor of ten.

Use of the above-cited techniques with present computers and workstations probably meets the design requirements for practical half-band filters with acceptable running times. But the question still remains whether there might be an even faster design technique, specially tailored to the design of half-band filters, that can benefit computationally from its special properties in ways that go beyond what is possible when using a general-purpose filter design program such as the Parks-McClellan algorithm.

We have found such a design technique and will describe it in this paper. Rather than improving the computation speed over the Parks-McClellan algorithm by approximately a factor of ten, as with [1], our method speeds up the computing by about a factor of one hundred. Thus, on a personal computer, a design that might take two minutes using the Parks-McClellan algorithm and about 12 seconds with the method of [1] can, in fact, be completed in slightly more than one second using our method.

The design is carried out by representing, at all stages, the polynomials concerned by Chebyshev polynomial expansions. This is done for several reasons. Firstly, the properties of linear-phase FIR filters are naturally describable in terms of Chebyshev polynomials. Secondly, such expansions tend to lead to equal-ripple behaviour by their very nature. And thirdly, when the degree of a polynomial is high, its properties are often very sensitive to the normal coefficients of powers of the variable, whereas they are much less sensitive to the coefficients in its Chebyshev polynomial expansion. It turns out that the slight extra complication of manipulating Chebyshev polynomial expansions is well worth the benefits realized.

The method of design does not produce an *exact* equal-ripple response in the passband and stopband, but the departure from exactness is extremely small, and the result is quite comparable to that in any filter produced by using the Parks-McClellan algorithm, partly because of the minor computational errors inherent in the latter.

*This work was supported by U.S. National Science Foundation Grant MIP-9632698.

2 DESIGN METHOD

In the hope that it will be easier to follow, we shall describe the design in terms of a filter of degree $n = 7$, writing the equations out in full. The extension to the case for general n, where n may be any positive odd integer, should be obvious. The transfer function $H(z)$ of a zero-phase half-band filter is defined by

$$\begin{aligned} H &= h_7 z^7 + h_5 z^5 + h_3 z^3 + h_1 z + h_0 + h_1 z^{-1} + \\ &\quad h_3 z^{-3} + h_5 z^{-5} + h_7 x^{-7} \qquad (1) \\ &= h_0 + h_1(z + z^{-1}) + h_3(z^3 + z^{-3}) + \\ &\quad h_5(z^5 + z^{-5}) + h_7(z^7 + z^{-7}). \qquad (2) \end{aligned}$$

By setting $z = x + jy = e^{j\omega}$, we can express H in terms of the Chebyshev polynomials $T_i(x)$ thus

$$\begin{aligned} H &= h_0 + 2h_1 \cos\omega + 2h_3 \cos 3\omega + 2h_5 \cos 5\omega + \\ &\quad 2h_7 \cos 7\omega \qquad (3) \\ &= h_0 + 2h_1 T_1(x) + 2h_3 T_3(x) + 2h_5 T_5(x) + \\ &\quad 2h_7 T_7(x). \qquad (4) \end{aligned}$$

By using the relation $2xT_m = T_{m+1} + T_{m-1}$ one can easily confirm that (4) is equivalent to

$$H = h_0 + 2x\left[b_0 + b_2 T_2(x) + b_4 T_4(x) + b_6 T_6(x)\right] \qquad (5)$$

where

$$\begin{array}{ll} 2h_1 = 2b_0 + b_2 & 2h_5 = b_4 + b_6 \\ 2h_3 = b_2 + b_4 & 2h_7 = b_6. \end{array} \qquad (6)$$

The useful approximation range of the Chebyshev polynomials with variable $x = \cos\omega$ is $-1 \le x \le 1$ which corresponds to $0 \le \omega \le \pi$. But in this interval the half-band filter has to make parts of two disjoint equal-ripple approximations; one of a passband having unit value centered on $\omega = 0$, and one of a stopband having zero value centered on $\omega = \pi$. The structure of H for the half-band filter, however, guarantees that if an equal-ripple passband is created, then an equal-ripple stopband will accompany it automatically. To simplify the problem we therefore change the variable so that the whole of just the passband occurs centrally in the approximation range of the Chebyshev polynomials in (5) by changing the variable from $x = \cos\omega$ to $y = \sin\omega$, so that $-1 \le y \le 1$ corresponds to $-\pi/2 \le \omega \le \pi/2$.

In the present application, where (5) contains Chebyshev polynomials of only even degree, changing the variable from x to y is achieved very easily by virtue of the relation

$$T_{2k}(x) = (-1)^k T_{2k}(y)$$

which holds, as is the case here, when $x^2 + y^2 = 1$. Applying this to (5) gives

$$\begin{aligned} H &= h_0 + 2\sqrt{1 - y^2}\left[b_0 - b_2 T_2(y) + b_4 T_4(y) - \right. \\ &\quad \left. b_6 T_6(y)\right] \qquad (7) \\ &= h_0 + 2\sqrt{1 - y^2}\, P(y) \qquad (8) \end{aligned}$$

where $P(y)$ is an *even* polynomial of degree 6.

Over the interval $-1 \le y \le 1$ we want H to behave as shown in Fig. 1 with an equal-ripple passband for $-\alpha \le y \le \alpha$. At $y = \pm 1$, $\omega = \pm\pi/2$ and $H = h_0 = 0.5$. We first scale the variable y to a new variable t by $y = \alpha t$ so that $t = \pm 1$ corresponds to $y = \pm\alpha$. Then

$$\begin{aligned} H &= h_0 + 2\sqrt{1 - \alpha^2 t^2}\left[a_0 + a_2 T_2(t) + a_4 T_4(t) + \right. \\ &\quad \left. a_6 T_6(t)\right] \qquad (9) \\ &= h_0 + 2\sqrt{1 - \alpha^2 t^2}\, Q(t) \qquad (10) \end{aligned}$$

is required to make an equal-ripple approximation to unity over $-1 \le t \le 1$. Here $Q(t) = P(\alpha t)$.

Bearing in mind that $h_0 = 0.5$, we approached this by first making the polynomial $Q(t)$ approximate $f(t)$ where

$$f(t) = \frac{1}{4\sqrt{1 - \alpha^2 t^2}}. \qquad (11)$$

An approximation for $Q(t)$ of degree $n - 1$ in t was obtained by truncating a Chebyshev polynomial expansion of $f(t)$ in the hope that this would generate a $Q(t)$ such that $Q(t)/f(t)$ would approximate unity in an equal-ripple fashion. It was not possible to find an analytic expression for the coefficients of the Chebyshev polynomial expansion of $f(t)$, so they were found by standard numerical methods involving sample values [3], [4] of $f(t)$. The number of sample points used had to exceed n in order to generate the required number of coefficients, and it was found that, by using at least $2n$ points, these coefficients settled down to the values one would have obtained by truncating an exact infinite expansion.

At first it was assumed that using $2n$ points would give the best choice for $Q(t)$, and an analysis of the passband response for this case certainly proved it to be remarkably good. The response was almost exactly equal ripple from zero frequency up to at least 75% of the cut-off frequency. Thereafter the ripple size decreased[1] fairly smoothly until it was between five and eight percent smaller at the cut-off. Several simple methods of modifying the coefficients to improve the constancy of the ripple size were tried out, but none proved helpful. Finally, almost by accident, it was discovered that by controlling the number of sample points to some value *between n and $2n$* the ripple size could be kept almost exactly constant.

The optimum value for the number of sample points proved to be that for which the ripple size at $t = 1$ most nearly equalled the ripple size at $t = 0$. These two ripple sizes can easily be found from the a_k coefficients of $Q(t)$

[1] In Chebyshev polynomial approximations, the ripple size normally tends to increase near the band edges, even without help from the Gibbs phenomenon due to discontinuities. Here, the ripples in the approximation of $f(t)$ by $Q(t)$ do increase in size near the band edges, but the increase there in the value of $f(t)$ proves more than enough to offset this when we form the quotient $Q(t)/f(t)$, causing the resulting H to behave as described.

in (9) and the values (± 1) of the Chebyshev polynomials at $t = 1$ and at $t = 0$. Specifically

$$Q(1) = a_0 + a_2 + a_4 + a_6 \tag{12}$$

and

$$Q(0) = a_0 - a_2 + a_4 - a_6 \tag{13}$$

and the ratio of the magnitudes of the ripple sizes at $t = 1$ and at $t = 0$ is then

$$\left| \frac{Q(1)/f(1) - 1}{Q(0)/f(0) - 1} \right| . \tag{14}$$

As one increases the number of sample points from n to $2n$, the quantity in (14) decreases monotonically from a value greater than unity to a value less than unity, and the object is to find the number of sample points corresponding to the value nearest to unity.

The calculation of the a_k coefficients in (9) is the central part of the design and the most intensive computational step. The general formula for the a_k is

$$a_k = \frac{2}{m} \sum_{i=1}^{m} f(\cos\theta_i)\cos k\theta_i \qquad \text{where } \theta_i = \frac{(2i-1)\pi}{2m} \tag{15}$$

and m is the number of sample points whose optimum value we seek. As is customary in this theory, the value of a_0 is one half that given by (15). Here, $f(t)$ is an even function of t, so one can reduce the number of terms in the summation from m to $m/2$ if m is even or to $(m+1)/2$ if m is odd. To compensate for this, all terms in the summation are doubled except, when m is odd, for the one term corresponding to $t = 0$, i.e. $i = (m+1)/2$, for which $\theta_i = \pi/2$.

Finding all the separate a_k by (15) in each step of the search for the best m, merely to get the two linear combinations of them needed in (12) and (13), proves rather wasteful. If one forms the appropriate sum, with respect to k, of the right-hand sides of (15), corresponding to (12) or (13), and interchanges the order of the two summations, then the inner sum, containing just the cosines, can be evaluated analytically. Thus, in general, for (12) we get

$$\frac{1}{2} + \cos 2\theta_i + \cos 4\theta_i + \ldots + \cos 2q\theta_i = \frac{\sin n\theta_i}{2\sin\theta_i} \tag{16}$$

and, for (13)

$$\frac{1}{2} - \cos 2\theta_i + \cos 4\theta_i - \ldots\ (-1)^q \cos 2q\theta_i = (-1)^q \frac{\cos n\theta_i}{2\cos\theta_i} \tag{17}$$

where $2q = n - 1$ is the degree of the polynomial $Q(t)$. Note that the first term in both (16) and (17) corresponds to the factor of $1/2$ associated with the coefficient a_0 mentioned above. When $\theta_i = \pi/2$, the right-hand side of (17) becomes indeterminate through the vanishing of both numerator and denominator, but the quotient of their derivatives then gives the value of the right-hand side as $n/2$. For this same argument, the right-hand side of (16) also simplifies, to $(-1)^q/2$. This approach reduces considerably the effort required to find the quantity in (14) and hence the optimum value of m. The search itself is most simply achieved by using the *bisection method* [3].

After finding the optimum number of sample points one can then compute the separate a_k coefficients using (15). The next step is to change the variable in (9) from t back to y and so obtain the form given in (7), i.e. to find the b_k from the a_k. The obvious, simple way of doing this is to form the regular polynomial equivalent of the expansion in the $T_i(t)$ in (9), scale the variable from t to y and then reform it as an expansion in the $T_i(y)$. But in all practical cases where n is fairly large, this method becomes very ill conditioned and leads to errors in the b_k. To avoid this one has to deal, all the time, with coefficients of Chebyshev polynomials, never resorting to regular polynomial equivalents at any stage. A method of doing this, applicable to the more general case with Chebyshev polynomials of both odd and even degree, was derived and is described in the APPENDIX. When the b_k in (7), and therefore in (5), have been found by this algorithm it is then a trivial matter to find the h_k via the formulas in (6), and the design is completed.

3 NUMERICAL EXAMPLE

Our filter example is required to have $f_c = 0.24$ and provide not less than 60 dB stopband loss. By experimenting with a few values of n one quickly finds that a suitable value is 83. This gives a stopband loss of 61.04 dB and a passband ripple of 0.015 dB. The h_k coefficients, preceded by other relevant data including the number of samples used in the calculation of the a_k coefficients, are all shown in Fig. 2, which is the output of a Fortran program we have written to implement our algorithm.

We have also determined the extent to which the passband and stopband deviate from the ideal equal-ripple response. Over the first half of the passband the maxima and minima are exactly equal, to four decimal digits, at ± 0.0008872. Near the passband edge the maxima and minima depart slightly from this value, though never by more than 0.3 %. With a ripple size of 0.015 dB this departure is completely insignificant. The minimum loss achieved over the stopband departs from the quoted figure of 61.0395 dB by less than 0.026 dB.

4 APPENDIX

We assume that we are given a polynomial $Q(t)$ of degree n expressed as a Chebyshev polynomial expansion thus

$$Q(t) = a_n T_n(t) + a_{n-1} T_{n-1}(t) + a_{n-2} T_{n-2}(t) + \ldots \tag{A1}$$

and that we wish to change the variable from t to y, where $t = \beta y$, so that the interval $-\beta \leq t \leq \beta$ transforms to $-1 \leq y \leq 1$. The parameter β is the reciprocal of the quantity α used in the paper in the relation $y = \alpha t$. $Q(t)$ can then be rewritten

$$\begin{aligned} Q(\beta y) &= P(y) \\ &= b_n T_n(y) + b_{n-1} T_{n-1}(y) + b_{n-2} T_{n-2}(y) + \ldots \end{aligned} \tag{A2}$$

Given the a_k and the parameter β, the problem is to find the b_k.

Each $T_k(t) = T_k(\beta y)$ in (A1) is an even or an odd polynomial in y and can be expressed as a linear combination of the $T_i(y)$ $(i \leq k)$ of like parity thus

$$\begin{aligned} T_k(\beta y) &= d_{k,k} T_k(y) + d_{k-2,k} T_{k-2}(y) + \\ & \quad d_{k-4,k} T_{k-4}(y) + \ldots \end{aligned} \tag{A3}$$

where the last term is $d_{0,k} T_0(y)$ if k is even, or $d_{1,k} T_1(y)$ if k is odd.

The $d_{i,j}$ are polynomials of degree j in the parameter β and are the elements of an upper-triangular matrix $\mathbf{D}$ $(0 \leq i \leq j \leq n)$. If we define the $(n+1)$-vectors $\mathbf{a}$ and $\mathbf{b}$ whose components are the a_k and b_k, $(0 \leq k \leq n)$ then $\mathbf{b} = \mathbf{D}\mathbf{a}$. We can compute the $d_{i,j}$ via a recurrence relation derived from that between three adjacent T_i polynomials, namely

$$T_k(\beta y) = 2\beta y T_{k-1}(\beta y) - T_{k-2}(\beta y). \tag{A4}$$

Expressing the $T_i(\beta y)$ in terms of the $d_{i,j}$ and the $T_j(y)$ as in (A3), and using

$$2x T_j = T_{j-1} + T_{j+1} \tag{A5}$$

we get the recurrence

$$d_{i,j} = \beta(d_{i-1,j-1} + d_{i+1,j-1}) - d_{i,j-2}. \tag{A6a}$$

A special case occurs when i in (A6a) is unity. Then one must use instead

$$d_{1,j} = \beta(2d_{0,j-1} + d_{2,j-1}) - d_{1,j-2}. \tag{A6b}$$

By creating the matrix $\mathbf{D}$ column by column, starting with $d_{1,-1} = \beta$ and $d_{0,0} = 1$ and using (A6), one can build up the b_i in (A2) successively from each a_j in turn as the $d_{i,j}$ from the j-th column are found.

In applying this algorithm to getting the polynomial P from the polynomial Q in the paper, the two vectors $\mathbf{a}$ and $\mathbf{b}$ would have nonzero entries only in the even subscript locations and, as noted previously, the parameter β would be equal to $1/\alpha$.

References

[1] P. P. Vaidyanathan and T. Q. Nguyen, "A "TRICK" for the design of FIR Half-Band Filters," *IEEE Transactions on Circuits and Systems,* vol. CAS-34, pp. 297-300, March 1987.

[2] P. P. Vaidyanathan, *Multirate Systems and Filter Banks.* Englewood Cliffs, New Jersey: Prentice-Hall, 1993.

[3] W. H. Press *et al., Numerical Recipes.* Cambridge, England: Cambridge University Press, 1986. See pp. 148-149.

[4] National Bureau of Standards, *Tables of Chebyshev Polynomials.* Washington, DC: US Government Printing Office, 1952. See p. XVII in the Introduction by Cornelius Lanczos.