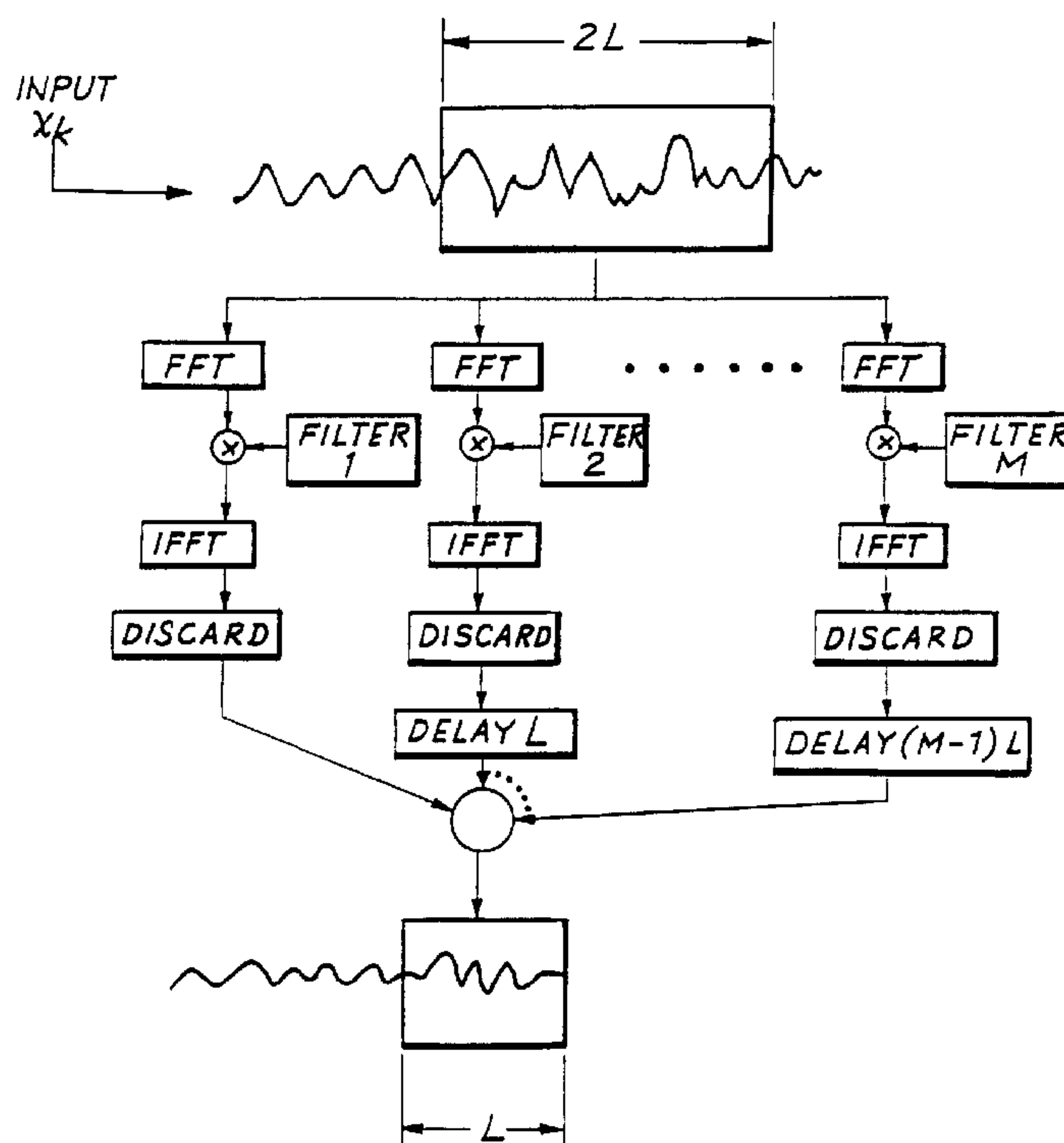




(86) Date de dépôt PCT/PCT Filing Date: 1993/07/05  
 (87) Date publication PCT/PCT Publication Date: 1994/01/20  
 (45) Date de délivrance/Issue Date: 2004/09/07  
 (85) Entrée phase nationale/National Entry: 1995/01/03  
 (86) N° demande PCT/PCT Application No.: AU 1993/000330  
 (87) N° publication PCT/PCT Publication No.: 1994/001933  
 (30) Priorités/Priorities: 1992/07/07 (PL 3387) AU;  
 1992/07/07 (PL 3386) AU; 1993/04/20 (PL 8357) AU

(51) Cl.Int.<sup>6</sup>/Int.Cl.<sup>6</sup> H03H 17/02, G06F 17/14  
 (72) Inventeur/Inventor:  
 MCGRATH, DAVID STANLEY, AU  
 (73) Propriétaire/Owner:  
 LAKE TECHNOLOGY LIMITED, AU  
 (74) Agent: ROBIC

(54) Titre : FILTRE NUMERIQUE DE GRANDE PRECISION ET DE GRANDE EFFICACITE  
 (54) Title: DIGITAL FILTER HAVING HIGH ACCURACY AND EFFICIENCY



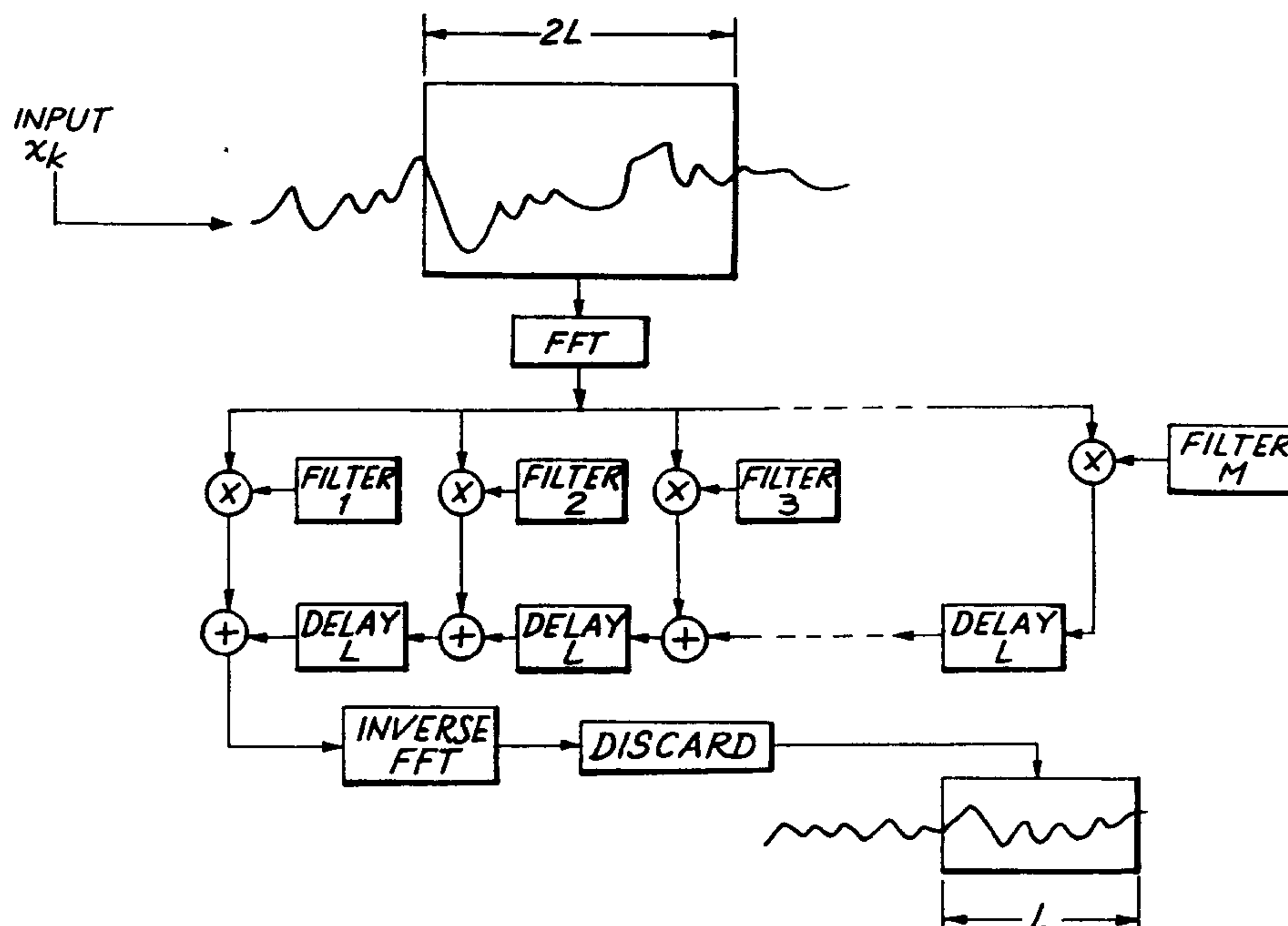
(57) **Abrégé/Abstract:**

Apparatus for and methods of operation of digital filters and certain other electronic digital signal processing devices are provided to improve the accuracy and efficiency of filtering. Particularly, the apparatus and method includes a digital filter with a long impulse response and lower latency, built by operating a number of small component filters (F1, F2, F3) in parallel and combining their outputs by addition, with each component filter operating with a different delay such that the net operation of the ensemble of said component filters is the same as a single filter with a longer impulse response, and the latency of the ensemble is equal to the shortest latency of the component filters. At least one group of the component filters is implemented using a Discrete Fourier Transform technique. A Fourier transform processor adapted to efficiently transform strings of real data is also described.



## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

<b>(51) International Patent Classification<sup>5</sup> :</b>  <b>H03H 17/06, 17/02, G06F 15/332</b>	<b>A1</b>	<b>(11) International Publication Number:</b> <b>WO 94/01933</b>  <b>(43) International Publication Date:</b> 20 January 1994 (20.01.94)
<b>(21) International Application Number:</b> PCT/AU93/00330 <b>(22) International Filing Date:</b> 5 July 1993 (05.07.93)  <b>(30) Priority data:</b> PL 3386 7 July 1992 (07.07.92) AU PL 3387 7 July 1992 (07.07.92) AU PL 8357 20 April 1993 (20.04.93) AU  <b>(71) Applicant (for all designated States except US):</b> LAKE DSP PTY. LIMITED [AU/AU]; 1/166 Maroubra Road, Maroubra, NSW 2035 (AU).  <b>(72) Inventor; and</b> <b>(75) Inventor/Applicant (for US only) :</b> McGRATH, David, Stanley [AU/AU]; 16 Caerleon Crescent, Randwick, NSW 2031 (AU).  <b>(74) Agent:</b> MAXWELL, Peter, Francis; Peter Maxwell & Associates, Blaxland House, 5 Ross Street, North Parramatta, NSW 2151 (AU).	<b>(81) Designated States:</b> AT, AU, BB, BG, BR, BY, CA, CH, CZ, DE, DK, ES, FI, GB, HU, JP, KP, KR, KZ, LK, LU, MG, MN, MW, NL, NO, NZ, PL, PT, RO, RU, SD, SE, SK, UA, US, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).  <div style="text-align: center; font-size: 2em; font-weight: bold;">2139511</div> <p style="text-align: center;">Published With international search report.</p>	

**(54) Title:** DIGITAL FILTER HAVING HIGH ACCURACY AND EFFICIENCY**(57) Abstract**

Apparatus for and methods of operation of digital filters and certain other electronic digital signal processing devices are provided to improve the accuracy and efficiency of filtering. Particularly, the apparatus and method includes a digital filter with a long impulse response and lower latency, built by operating a number of small component filters (F1, F2, F3) in parallel and combining their outputs by addition, with each component filter operating with a different delay such that the net operation of the ensemble of said component filters is the same as a single filter with a longer impulse response, and the latency of the ensemble is equal to the shortest latency of the component filters. At least one group of the component filters is implemented using a Discrete Fourier Transform technique. A Fourier transform processor adapted to efficiently transform strings of real data is also described.

**DIGITAL FILTER HAVING HIGH ACCURACY AND EFFICIENCY****FIELD OF THE INVENTION**

The present invention relates to the art of electronic signal processing, and more particularly but not exclusively, to an electronic filtering environment wherein relatively high accuracy and efficiency is desired and a relatively short flow-through delay (termed "latency") is desired.

**DESCRIPTION OF THE PRIOR ART**

With reference to Fig. 1, electronic filters are utilised to modify the characteristics of an incoming electronic signal so as to provide an output signal which is modified in some defined fashion. In the case of Fig. 1 a "notch" filter is illustrated wherein, in the frequency domain, frequencies in the spectrum of the incoming signal S1 are attenuated in the  $F_1$  to  $F_2$  band so as to produce output signal S2.

Such filters can be implemented from entirely analog components although, in more recent times, there is a preference, in many circumstances, to implement the filter in a digital fashion. Digital implementation can be by means of dedicated digital circuitry or by means of computers (micro processors) programmed to operate as a filter.

Filters have many applications in the field of electronic modelling of real world conditions. For example, filters can be used to provide a model of the acoustic characteristics of rooms or halls. Filters are also used to model deficiencies in systems so as to apply appropriate correction factors for the purpose of removing (cancelling)

imperfections in signals caused by the deficiencies.

Frequently it is desirable that such processing take place in "real time". Also, it is desirable that there is effectively no delay in filtering of a signal generated in a real/live environment so that the modelling/correcting steps performed by the filter are, to all intents and purposes, without any delay being perceptible to the end user.

To achieve this the delay introduced by the filter  $F$  while it performs its filtering function must be reduced to a negligible figure. That is, the time when signal  $S_1$  first presents to filter  $F$  and the time when the results of the filtering by filter  $F$  of the first incoming portion of signal  $S_1$  become available at the output of the filter  $S_2$  must be almost the same. The delay between these two events is hereinafter referred to as the "latency" of the filter system.

Where the filter  $F$  is implemented in a digital manner it may first be necessary to sample the incoming signal  $S_1$  (via an analog to digital converter) then perform the filtering function and then convert the digital signal back to an analog signal (by means of a digital to analog converter). The sampling process takes samples of the incoming signal at discrete time intervals  $t_i$ . The time between each sample is usually the same.

The sampling processing itself introduces finite delays into the system. Additionally, where the filter is implemented by one of the popular fast convolution techniques there is a delay introduced which, in very broad terms, is

**SUBSTITUTE SHEET**

- 3 -

proportional to the accuracy (or length) of the filter.

Mathematically, the filtering operation (that is, the step of imposing the filter characteristic upon the incoming signal S1 so as to produce outgoing signal S2) is known as "convolution" in the time domain. The step of convolution in the time domain becomes a multiply operation in the frequency domain. That is, if the incoming signal S1 is first sampled, then Fourier transformed into the frequency domain, the frequency response of the filter F is vector multiplied with the Fourier transform of the signal S1. The signal is then inverse Fourier transformed to produce a sampled (convolved) output (which can be converted back to analog form if required).

Figure 2 shows the way a convolver (also known as a Finite Impulse Response (FIR) filter) has its impulse response  $\{a_k\}$  measured (for a convolver operating on a treatment of sampled data). For a physical filter,  $a_k$  is zero for all  $k < 0$ . For a general input sequence  $\{x_k\}$ , the filter's output  $\{Y_k\}$  is defined as:

$$Y_k = \sum_{i=0}^{\infty} a_i x_{k-i} \quad (1)$$

A linear filter such as this has a measurable latency,  $d$ , defined as:-

$$\begin{aligned} a_d &\neq 0, \text{ and} \\ a_k &= 0 \text{ for all } k < d \end{aligned} \quad (2)$$

SUBSTITUTE SHEET
------------------

2139511

- 4 -

In other words,  $a_d$  is the first non-zero value in the output sequence. The latency  $d$  is never negative in a physical system. In a similar fashion, for a Finite Impulse Response Filter, we can determine which is the LAST non-zero value in the output sequence. This will give us the length of the impulse response. If we call the length  $l$ , then this means that  $a_{d+l-1}$  is the last non-zero value in the output sequence (see Figure 3).

Typical schemes for implementing FIR filters fall into two categories:-

1. Time domain filters that compute each output sample as each new input sample arrives, thus allowing latencies as low as  $d=0$  or  $d=1$ . Typical filter lengths ( $l$ ) are short.
2. Fast convolution filters that compute a number of output samples in a block. Typical filter lengths ( $l$ ) can be very long. The lowest achievable latency is usually related to the filter length,  $d \approx l \div K$  or

$$K \approx l \div d \quad (3)$$

where  $K$  is a measure of the efficiency of the particular algorithm used. A typical value of  $K$ , for the commonly used fast convolution algorithms such as illustrated in Figs. 4 and 5, is 0.5.

#### BRIEF DESCRIPTION OF THE INVENTION

It is an object of at least a preferred embodiment of the present invention to provide a method and apparatus for performing relatively long convolutions on digital sampled

SUBSTITUTE SHEET
------------------

data so as to provide relatively higher efficiency for a given length than is ordinarily produced with other methods.

In this specification it is assumed that the filter characteristics can be modelled as approximately linear so that the principles of superposition can be applied.

According to the present invention, there is provided a digital filter having an impulse response that includes a low latency and a long filter length, said digital filter comprising a plurality of component filters each having an impulse response including a latency and a filter length, each of said component filter having an input connected to an input of said digital filter, and an output connected to a summing means for combining, by addition, each of said outputs to form an output of said digital filter such that the impulse response of said digital filter is the sum of the component impulse responses and the low latency is equal to the shortest one of the component latencies.

According to the present invention, there is also provided a digital filter with a long impulse response and low latency comprising a smaller number of component filters in parallel and adding means for combining the output of said component filters, with each component filter operating with a different delay such that the net operation of said component filters is the same as a single filter with a longer impulse response, and the latency of said component filters is equal to a shortest latency of the said component filters wherein one or more of said component filters comprise a time-domain finite impulse

-5a-

response filter and multipliers and adders, and the remainder of said component filters comprise a fast convolution means, such that time-domain filter provide the lowest latency portion of said component filters impulse response, and the fast-convolution means provide the longer filter components and said fast-convolution means comprise Discrete Fourier Transformer or the Fast Fourier Transformer.

According to the present invention, there is also provided a digital filter with a long impulse response and low latency comprising a smaller number of component filters in parallel and adding means for combining the output of said component filters, with each component filter operating with a different delay such that the net operation of said component filters is the same as a single filter with a longer impulse response, and the latency of said component filters is equal to a shortest latency of the said component filters wherein one or more of said component filters comprise a time-domain finite impulse response filter and multipliers and adders, and the remainder of said component filters comprise a fast convolution means, such that time-domain filter provide the lowest latency portion of said component filters impulse response, and the fast-convolution means provide the longer filter components and said fast-convolution means comprise a Modified Discrete Fourier Transform processor comprising: means for receiving an input vector  $x(k)$  of length  $N$ , where each element of  $x(k)$  is real;

means for creating a complex vector  $x'(l)$  of length  $N/2$  according to the following:

-5b-

$$x'(l) = [x(l) - jx([l + N/2])]e^{-xj l / N} \text{ for } 0 \leq l < N/2$$

and means for computing an N/2 point Discrete Fourier Transform (DFT) of  $x(l)$  to give the N/2 complex result vector  $Z(p)$ :

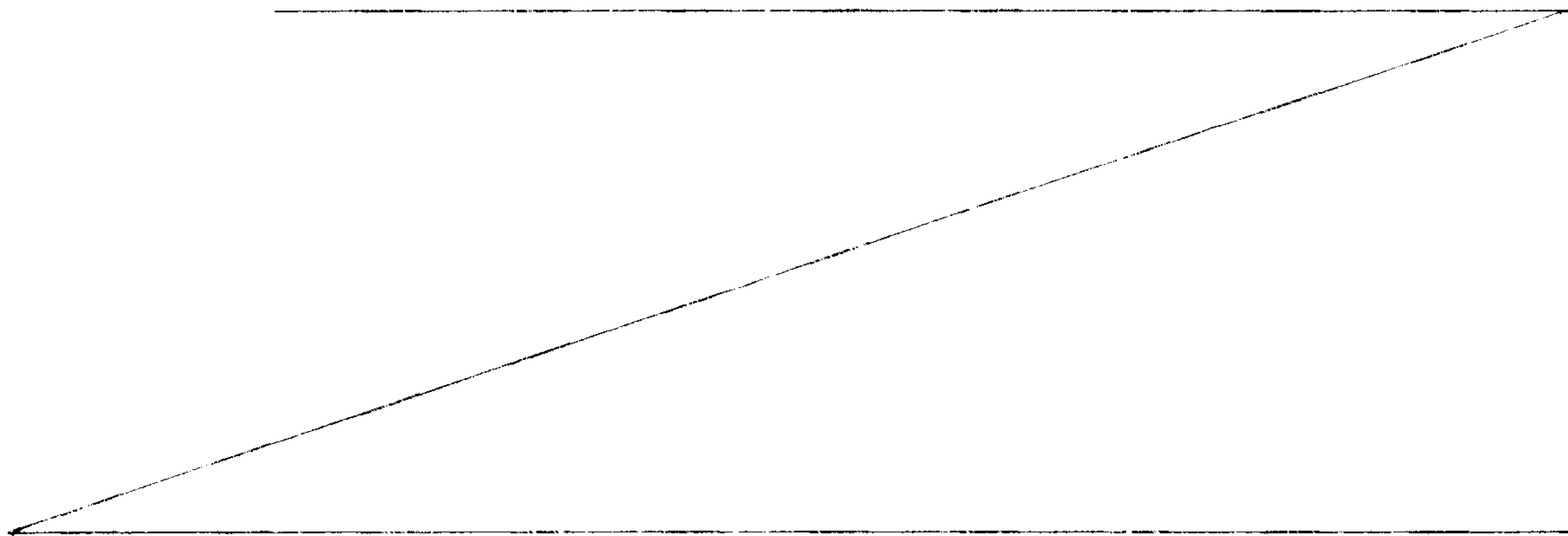
$$Z(p) = DFT_{(n/2)}[x'(l)]$$

and a complementary Inverse Modified Discrete Fourier Transform processor.

Preferably the component filters are implemented in different ways, with some filters adapted to provide low latency, and other filters adapted to provide longer filter lengths, such that the ensemble filter provides both low latency and long impulse response characteristics.

Preferably one or more of the component filters is implemented as a time-domain finite impulse response filter (built with multiply and add operations) and the remainder are implemented using a fast convolution method, such that the time-domain filter(s) provides the lowest latency portion of the ensemble impulse response, and the fast-convolution filter(s) provide the longer filter components.

Preferably the fast-convolution filters are built using the Discrete Fourier Transform or the Fast Fourier Transform.



In an alternative preferred form the fast-convolution filters are built using the Modified Discrete Fourier Transform as described in this specification.

5 Preferably a group or groups of more than one of the component filters are implemented using fast convolution techniques with the component filters in each group using a transform of the same length, and wherein only one transform operation is performed for each group, so that the component filters in each group share the transformed output from the  
10 respective transform operation.

In an alternative preferred form a group or groups of more than one of the component filters are implemented using fast convolution techniques with the component filters in each group using a transform of the same length, and wherein  
15 only one inverse transform operation is performed for each group, so that the output from the component filters in each group is summed before being passed to the respective inverse transform operation.

In a further alternative preferred form a group or  
20 groups of more than one of the component filters are implemented using fast convolution techniques with the component filters in each group using a transform of the same length, and wherein only one transform operation is performed for each group, so that the component filters in each group  
25 share the transformed output from the respective transform operation; and wherein a group or groups of more than one of the component filters are implemented using fast convolution techniques with the component filters in each group using a

**SUBSTITUTE SHEET**

-7-

transform of the same length, and wherein only one inverse transform operation is performed for each group, so that the output from the component filters in each group is summed before being passed to the respective inverse transform operation.

According to the present invention, there is provided a method for building a digital filter with a long impulse response and low latency, built by operating a number of smaller component filters in parallel and  
10 combining their outputs by addition, with each component filter operating with a different delay such that the net operation of the ensemble of said component filters is the same as a single filter with a longer impulse response, and the latency of the ensemble is equal to the shortest latency of said component filters wherein one or more of said component filters is implemented as a time-domain finite impulse response filter, built with multiply and add operations, and the remainder of said component filters are implemented using a fast convolution method, such that  
20 time-domain filter(s) provides the lowest latency portion of the ensemble impulse, and fast convolution filter(s) provide longer filter components, wherein said fast convolution filters are built using the Discrete Fourier Transform or the Fast Fourier Transform.

According to the present invention, there is also provided a method for building a digital filter with a long impulse response and low latency built by operating a number of smaller component filters in parallel and combining their outputs by addition, with each component  
30 filter operating with a different delay such that the net operation of the ensemble of said component filters is the

-7a-

same as a single filter with a longer impulse response, and the latency of the ensemble is equal to the shortest latency of said component filters wherein one or more of said component filters is implemented as a time-domain finite impulse response filter, built with multiply and add operations, and the remainder of said component filters are implemented using a fast convolution method, such that time-domain filter provides the lowest latency portion of the ensemble impulse response, and fast-convolution filter provide the longer filter components, wherein said fast-convolution filters are built using a Modified Discrete Fourier Transform method comprising the steps of:

receiving an input vector  $x(k)$  of length  $N$ , where each element of  $x(k)$  is real;

means for creating a complex vector  $x'(l)$  of length  $N/2$  according to the following:

$$x'(l) = \left[ x(l) - jx\left(l + \frac{N}{2}\right) \right] e^{-\frac{\pi j l}{N}} \text{ for } 0 \leq l < \frac{N}{2}; \text{ and}$$

computing an  $N/2$  point Discrete Fourier Transform (DFT) of  $x'(l)$  to give the  $N/2$  complex result vector  $Z(p)$ :

$$Z(p) = DFT\left(\frac{N}{2}\right)[X'(l),]$$

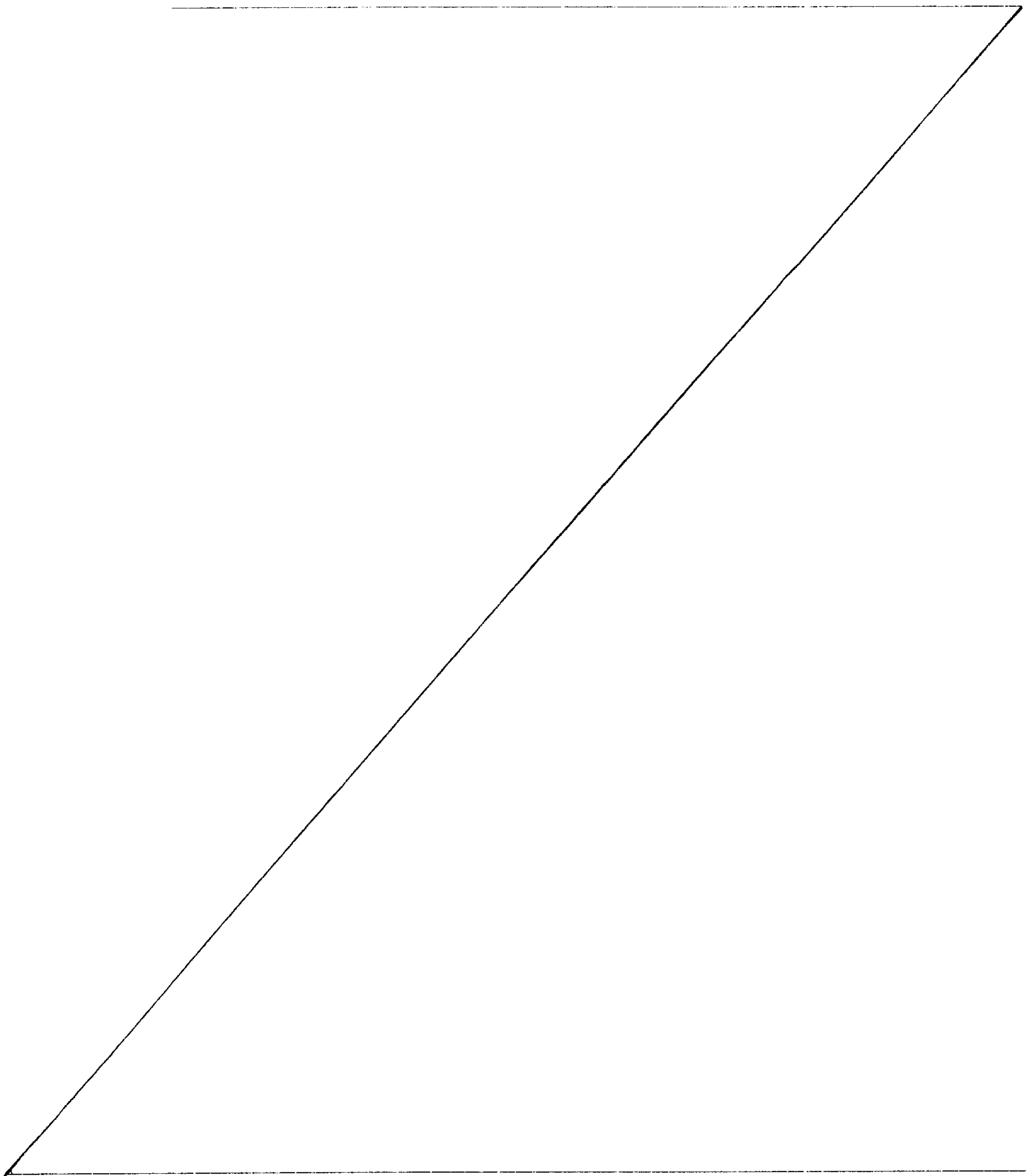
and a complementary Inverse Modified Discrete Fourier Transform method.

Preferably the component filters are implemented in different ways, with some filters adapted to provide low latency, and other filters adapted to provide longer filter lengths, such that the ensemble filter provides both low latency and long impulse response characteristics.

Preferably one or more of the component filters is implemented as a time-domain finite impulse response

-7b-

filter (built with multiply and add operations) and the remainder are implemented using a fast convolution method, such that the time-domain filter(s) provides the lowest latency position of the ensemble impulse response, and the fast-convolution filter(s) provide the long filter components.



Preferably the fast-convolution filters are built using the Discrete Fourier Transform or the Fast Fourier Transform.

Preferably the fast-convolution filters are built using the Modified Discrete Fourier Transform as described in this specification.

Preferably a group or groups of more than one of the component filters are implemented using fast convolution techniques with the component filters in each group using a transform of the same length, and wherein only one transform operation is performed for each group, so that the component filters in each group share the transformed output from the respective transform operation.

In an alternative preferred form a group or groups of more than one of the component filters are implemented using fast convolution techniques with the component filters in each group using a transform of the same length, and wherein only one inverse transform operation is performed for each group, so that the output from the component filters in each group is summed before being passed to the respective inverse transform operation.

In a further alternative preferred form a group or groups of more than one of the component filters are implemented using fast convolution techniques with the component filters in each group using a transform of the same length, and wherein only one transform operation is performed for each group, so that the component filters in each group share the transformed output from the respective transform operation; wherein a group or groups of more than one of the

component filters are implemented using fast convolution techniques with the component filters in each group using a transform of the same length, and wherein only one inverse transform operation is performed for each group, so that the output from the component filters in each group is summed before being passed to the respective inverse transform operation.

According to the present invention, there is provided a digital filter comprising a plurality of filter  
10 modules arranged in parallel and each configured to receive an input signal for filtering, an output of each of said filter modules being summed to produce a composite filtered output signal, each of said filter modules having a corresponding filter length and a corresponding impulse response that is a selected portion of a desired impulse response of said digital filter, wherein the length of each of said filter modules is different, and wherein at least the shortest length filter module is a time-domain low  
20 latency filter configured to process at least first portions of said desired impulse response and the longer length filter module is a fast-convolution longer latency filter configured to process subsequent portions of said desired impulse response.

In yet a further broad form of the invention there is provided a digital filter for filtering overlapped groupings of consecutive samples  $a, b, c, d, \dots$ , said filter comprising a transform processor of length  $m$ , and  $N$  filter processors, each of length  $k$ , an adder adapted to sum the outputs as they are fed in parallel from said  $N$   
30 filter processors, an inverse transform processor of length  $m$ ; said consecutive samples  $a, b, c, d, \dots$  being fed in

-10-

blocks of length  $m$ , each of said blocks being passed through said transform processor and then through each of said  $N$  filter processors with a delay of zero before being passed through a filter processor of said  $N$  filter processors, a delay of  $k$  before being passed through a second filter processor and so on up to a delay of  $(N-1)k$  before being passed through the  $N$ th filter processor; whereby a filter of effective length  $Nk$  is effected with a latency corresponding to that of a conventional filter of length  $k$ .

Preferably, in a further broad form of the invention there is provided a method of implementing a filter with a relatively high length/latency efficiency  $K$ ; said method comprising transforming progressive, consecutive and overlapping position of input data into frequency domain, to produce corresponding transformed data, storing said corresponding transformed data, so as to effect passing said portions of transformed data through a transform processor consecutively; feeding a first transformed portion through a first filter function processor; then feeding said first transformed portion through an  $N-1$ th filter function processor whilst an  $N-1$ th transformed portion is being fed through said first filter function processor; and so on so that a continuously moving  $N$  consecutive blocks of input data are transformed and passed through  $N$  filter function processors; adding the output from said  $N$  filter function processors; inverse transforming said output; performing a discard operation as necessary whereby consecutive portions of filtered output data are produced from said progressive, consecutive and overlapping portions of input data.

-11-

Preferably, in a form of the invention there is provided a method of implementing a filter with a relatively high length/latency efficiency  $K$ ; said method comprising applying a mathematical transform to progressive, consecutive and overlapping portions of input data so as to produce corresponding transformed data; performing a mathematical operation on individual ones (i.e. two or more) of said transformed data; superposing (by addition) the data resulting from said mathematical operations so as to produce resultant data; applying an inverse mathematical transform to said resultant data so as to produce filtered output data.

Preferably each said filter function processor is processing a filter whose Impulses Response is half the length of the transform processor.

Preferably said transform processor is a Fast Fourier Transform Processor and said inverse transform processor is an inverse Fast Fourier Transform Processor and said filter function processor is effected by a multiply operation of transformed input data with an impulse response corresponding to a selected portion of a desired impulse response for the entire filter.

Preferably said method is optimized whereby an approximately equal amount of processing time is spent on Fourier transformation as on filter function (multiply) operation.

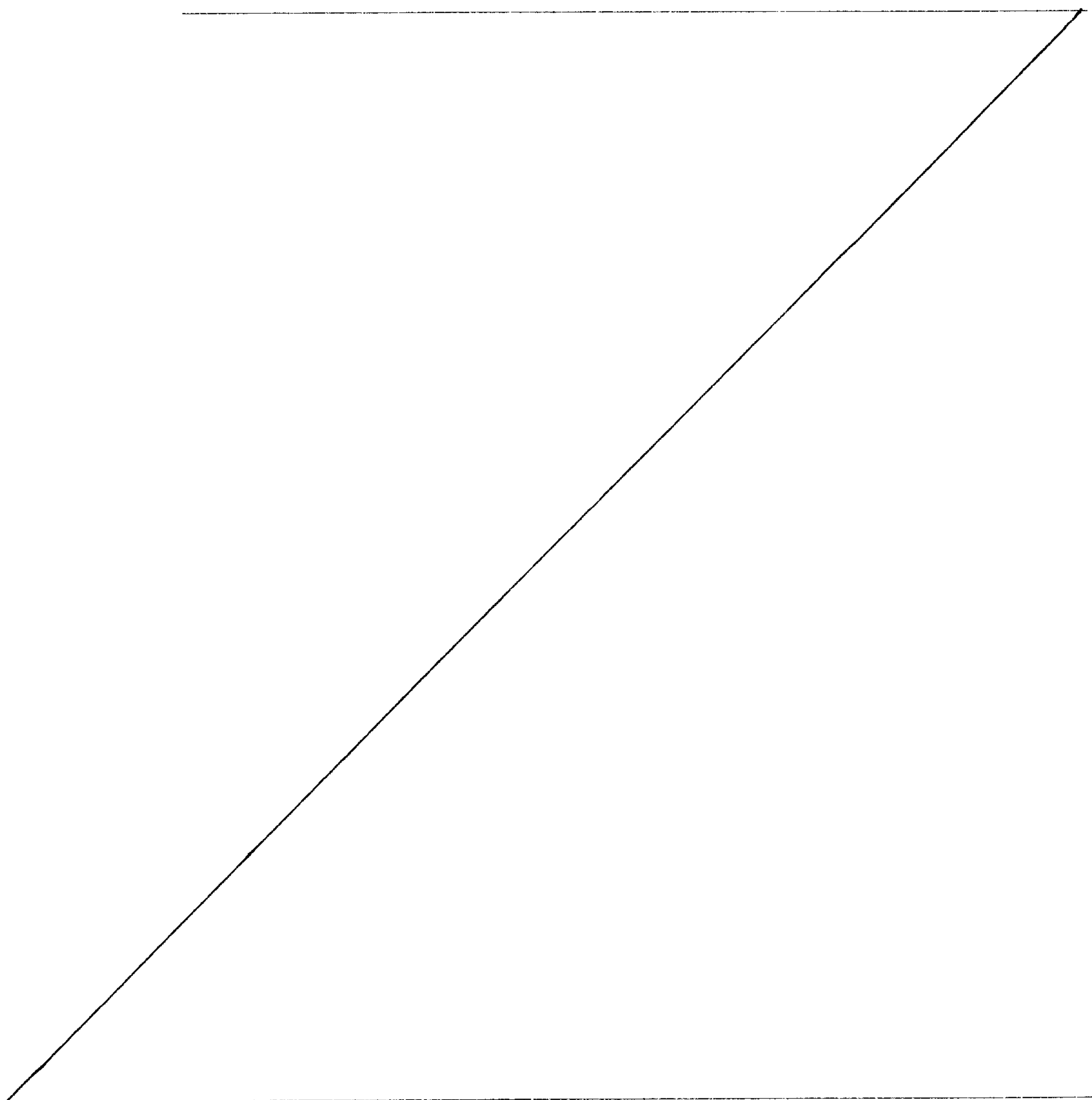
Preferably, in a form of the invention there is provided a method of implementing a filter with a relatively high length/latency efficiency  $K$ ; said method comprising transforming progressive, consecutive and overlapping portions of input data into the frequency

-11a-

domain, performing a mathematical operation on individual ones of said transformed signals, superposing (by addition) the consecutive signals resulting from said mathematical operations, inverse transforming the resultant signal from the frequency domain back to the time domain and outputting said signal.

Preferably said transform is a Fast Fourier Transform and said inverse transform is an Inverse Fast Fourier Transform.

10



Preferably said mathematical operation is a vector multiply of the Fourier transformed input signal segment with the frequency response of the desired (time domain) filter characteristic.

5 Preferably the underlying operation is an overlap and discard operation on successive, overlapping portions of input data.

10 Alternately the underlying operation is an overlap and add operation on successive, overlapping portions of input data.

In a particular preferred form the Fourier transform of any given overlapping block of input samples is taken only once and reused as required.

15 Preferably, the inverse Fourier transform of any given summed grouping of filtered data is performed only once.

20 In a particular preferred form the above described method is utilised to implement at least some of the filter modules of a composite electrical filter; said composite electrical filter comprising a plurality of filter modules arranged to receive in parallel an incoming input signal for filtering; the output from each of said filter modules being summed to produce a composite filtered output signal; each of said filter modules adapted to have an impulse response that is a selected portion of the impulse response of the composite filter.

25

It is preferable to minimise the overlap of the selected portions of impulse response, or to make them not overlap at all.

-13-

Preferably the length of each of said filter modules is different with the characteristic of the shorter length filter modules adapted to process first (or earlier) portions of said impulse response and longer length filter modules adapted to process following (or later) portions of said impulse response.

Preferably said longer length filter modules are adapted to filter progressively longer portions of said impulse response.

10           In a particular preferred form, the shortest module of said plurality of filter modules is a time-domain (low latency) filter whilst additional ones of said filter modules are longer fast-convolution (longer latency) filters implemented using the Fast Fourier Transform method described above or other traditional fast convolution methods.

          It is an object or at least a further preferred embodiment of the present invention to provide a method and apparatus for performing relatively long convolutions on  
20 digital sampled data so as to provide relatively lower latency than is ordinarily incurred with other methods.

          It is assumed that the filter characteristics can be modelled as approximately linear so that the principles of superposition can be applied.

          According to the present invention, there is provided a composite electrical filter comprising a plurality of filter modules arranged to receive in parallel an incoming input signal for filtering, an output from each of said filter modules being summed to produce a composite  
30 filtered output signal, each of said filter modules having an impulse response that is a selected portion of said

-14-

impulse response of the composite filter wherein the number of said filter module is  $N$ , comprising filters  $F_1, F_2 \dots F_N$  and wherein filter modules  $F_1$  is a filter with a very low latency implemented with time domain techniques while all other filter modules  $F_i$  are implemented with fast convolution means and these fast convolution filters  $F_i$  are composed of a sequence of filters, each with longer filter length than its predecessor and hence each with longer latency, but still preserving the property that  $d_{i+1} = d_i + l_i$  whereby it is ensured that said composite filter  $F$  output formed by summing together  $N$  component filter outputs has an impulse response without any "holes" in it.

It is preferably to minimise the overlap of the selected portions of impulse response, or to make them not overlap at all.

Preferably the length of each of said filter modules is different with the characteristic of the shorter length filter modules adapted to process first (or earlier) portions of said impulse response and longer length filter modules adapted to process following (or later) portions of said impulse response.

Preferably said longer length filter modules are adapted to filter progressively longer portions of said impulse response.

In a particular preferred form, the shortest module of said plurality of filter modules is a time-domain (low latency) filter whilst additional ones of said filter modules are longer fast-convolution (longer latency) filters.

Preferably only the shortest of said filter modules is a time domain filter.

-15-

In a further particular preferred form, where the number of said filter modules is  $N$  comprising filters  $F_1, F_2 \dots F_N$  filter module  $F_1$  is a filter with very low latency implemented with time domain techniques whilst all other filter modules  $F_i$  are implemented with fast convolution techniques and these fast convolution filters  $F_i$  are composed of a sequence of filters each with longer filter length than its predecessor and hence each with longer latency, but still preserving the property that  $d_{i+1} = d_i + l_i$  whereby it is ensured that the composite filter  $F$  output formed by summing together the  $N$  component filter outputs has an impulse response without any "holes" in it.

Preferably, in a further broad form of the invention there is provided a method of filtering sampled data so as to achieve a relatively long length but short latency filtering of said data, said method comprising passing said data in parallel through a plurality of sub-filters and summing the output samples from all of said sub-filters to produce filtered sampled data; and wherein the Impulse Response of each of said sub-filters is a selected portion of the desired Impulse Response of the filter characteristic required to produce said filtered sampled data from said sampled data and wherein each said selected portion is selected for each of said sub-filters such that the output from all of said sub-filters, when summed, behaves as if filtered through a filter having said desired Impulse Response.

According to the present invention, there is provided a Fourier transform processor adapted to transform strings of real numbers of use in digital filtering, said processor comprising:

-16-

input means for receiving an input vector  $x(k)$  of length  $N$ , where each element of  $x(k)$  is real;

vector means for creating a complex vector  $x'(l)$  of length  $N/2$  according to the following:

$$x'(l) = \left[ x(l) - jx\left(l + \frac{N}{2}\right) \right] e^{-\frac{\pi j l}{N}} \text{ for } 0 \leq l < \frac{N}{2}; \text{ and}$$

transform means for computing an  $N/2$  point Discrete Fourier Transform (DFT) of  $x'(l)$  to give the  $N/2$  complex result vector  $Z(p)$ :

$$Z(p) = DFT\left(\frac{N}{2}\right)[X'(l),]$$

10 and a complementary Inverse Modified Discrete Fourier Transform.

Preferred embodiments of the invention will now be described with reference to the accompanying drawings wherein:-

**BRIEF DESCRIPTION OF THE DRAWINGS**

- Fig. 1 is a generalised block diagram of a filter operation in the frequency domain,
- Fig. 2 defines the basic terminology used for a convolution filter,
- 20 Fig. 3 defines the latency and length of the filter of Fig. 2,
- Fig. 4 illustrates in a diagrammatic flow chart form, a prior art method of processing sampled data by a Fast Fourier Transform approach,
- Fig. 5 further illustrates the approach of Fig. 4,
- Fig. 6,7,8 develop a method of filtering according to a generalised first embodiment of the invention whereby a relatively high efficiency factor  $K$  can be achieved as compared with the approach of Fig. 4,

- Fig. 9 is a diagram of an embodiment of the invention implementing the method of Fig. 8 where the number of sub-filters is 6,
- 5 Fig. 10 illustrates in block diagram form a summed filter a part of which can be implemented advantageously with the filter of Fig. 8,
- Fig. 11 is a block diagram of the summed filter of Fig. 10 incorporating sub-filters some of which implement the method of Fig. 8,
- 10 Fig. 12 illustrates the manner of processing of an input signal by an example of the filter arrangement of Fig. 10 which utilised five filter portions,
- Fig. 13 illustrates the manner of selection of the filter characteristics of the filter of Fig. 12,
- 15 Fig. 14 is a bloc diagram of an alternative implementation of the summed filter of Fig. 8,
- 20 Fig. 15 illustrates a typical flow of a (prior art) fast convolution algorithm implementation suitable for filters  $F_2 - F_5$  of Fig. 12,
- 25 Fig. 16 illustrates a DFT engine which forms the basis for an explanation of a fourier transform algorithm optimised to process real number strings, and

Fig. 17 is a block diagram of a further embodiment of the invention wherein the summed filter of Fig. 10 is implemented utilizing a Modified Discrete Fourier Transform.

5 DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS OF THE  
INVENTION

1. High Efficiency Filter

Figure 4 illustrates the time-flow of a typical fast-convolution algorithm. This is an overlap-discard algorithm implemented using the Fast Fourier Transform (FFT). 2M words of input data that arrives during time segments a and b is processed fully during time segment c with a forward FFT, a vector multiply, and an inverse FFT. The resulting M words of output data are buffered, to be presented at the output during time segment d. The FFT and inverse FFT (IFFT) are only used to transform the data between the time-domain and frequency domain. The actual filter operation is executed in the vector multiply operation, which actually takes only a small fraction of the total compute time. So, the relevant parameters of the filter of Fig. 4 are:-

Length = M,

Latency = 2M,

and, therefore  $K = 0.5$

25 With reference to Figs. 6, 7 and 8, the rationale behind the method and apparatus according to at least one embodiment of the present invention is derived.

Fig. 6 illustrates a filter of length ML where the filter characteristics of each of the component filters F1,

F<sub>2</sub>. . . . F<sub>n</sub> are separate, discrete, component portions of the desired filter characteristic for the entire filter assembly. The delays L, 2L. . . . (M-1) L are imposed as illustrated so as to recreate, following addition, an output  
5 Y<sub>k</sub> equivalent to that achieved by passing input samples x<sub>k</sub> through a filter having the filter characteristic from which the filter characteristic portions for filters F<sub>1</sub>, F<sub>2</sub>. . . . were derived. Fig. 7 is derived by implementing the filters F<sub>1</sub>, F<sub>2</sub>. . . of Fig. 6 using the Fast Fourier Transform  
10 algorithm of Fig. 5.

With reference to Fig. 8, reorganisation of the filter of Fig. 7 allows the use of only one Fast Fourier Transform module 11 and one inverse Fast Fourier Transform module 12. It is implicit that the Fast Fourier Transform module is  
15 adapted to process a block of samples from input x<sub>k</sub> equal to twice the length of each of the filters Filter 1, Filter 2, Filter 3. . . . . illustrated in Fig. 8.

As previously stated the filter characteristic (impulse response) of each consecutive filter F<sub>1</sub>, F<sub>2</sub>. . . . is taken from  
20 and corresponds to consecutive corresponding portions from the impulse response desired of the entire filter module.

The time delay L before each Fast Fourier transformed block of data is passed through the next filter is equal to half the sample length originally processed by the Fast  
25 Fourier Transform module.

Figure 9 shows the computation of one block of output data, in a similar style to Figure 4, but using the improved length/latency efficiency method derived in Figs. 6, 7 and 8.

**SUBSTITUTE SHEET**

The method of Fig. 8 as used by Fig. 9 is summarised below.

During time segment h, the input data that arrived during time segments f and g is FFT'd and the resulting block of Frequency Domain Input Data is stored for future use. We then compute the next block of Frequency Output Data, which is inverse FFT'd and presented as output during time segment i. The old way of computing fast-convolution simply took the latest block of Frequency Domain Input Data, and multiplied it by a vector that represents the desired filter response, to get the new Frequency Domain Output Data. The improved length/latency efficiency method uses a number of previous Frequency Domain Input Data blocks to compute the new Frequency Domain Output Data block, as shown in Figure 9. In this example, the blocks of filter data are called Filter A, Filter B, ..., Filter F. In this example, the filter implemented is 6 times as long as the filter implemented in Figure 4, but with no greater latency. By comparison with Fig. 4, the relevant parameters of the filter of Fig. 9 are:-

Length =  $6M$ ,

Latency =  $2M$ ,

and, therefore  $K = 3$

Fig. 8 summarises the logic behind the implementation of the embodiment of Fig. 9.

Particularly, it will be noted that the progressive delays  $L$ ,  $2L$ ,  $3L$ , ...,  $(M-1)L$  of Fig. 8 are achieved in Fig. 9 by the taking of delayed, overlapped groupings of consecutive samples a, b, c, d, ...

**SUBSTITUTE SHEET**

The above described filter arrangement can be used advantageously in a low-latency FIR filter arrangement such as illustrated in Fig. 10.

Figure 10 shows an architecture for implementing an FIR filter by adding together N filters. If each filter is characterised as: Filter  $F_i$ , latency  $d_i$ , length  $l_i$ , then generally the N filters are chosen so that their latencies are ordered in ascending order, and furthermore  $d_{i+1} = d_i + l_i$ . This means that the first non-zero value in the impulse response of filter  $F_{i+1}$ , comes immediately following the last non-zero value in the impulse response of filter  $F_i$ . Hence this summation of filters results in a single, longer filter with its impulse response being the sum of the impulse responses of the N component filters.

The important property of this filter is the length/latency efficiency, K, which is higher than any of the component filter efficiencies.

That is, the filter of Fig. 10 uses the technique of adding together several filters to form a new filter which is as long as the sum of the component filter lengths, and whose latency is as short as the latency of the lowest-latency component filter.

Fig. 11 shows an implementation of the low-latency filter 10 of Fig. 10 wherein there are three filter modules  $F_1$ ,  $F_2$ ,  $F_3$ . The first module  $F_1$  is a low-latency ( $d=0$ ) time domain filter whilst filters  $F_2$  and  $F_3$  are implemented according to the embodiment described in respect of Figs. 8 and 9.

2. A low-latency FIR filter

As previously described, Fig. 10 shows an architecture for implementing an FIR filter by adding together N filters.

If each filter is characterised as: Filter  $F_i$ , latency  $d_i$ , length  $l_i$ , then generally the N filters are chosen so that their latencies are ordered in ascending order, and furthermore  $d_{i+1} = d_i + l_i$ . This means that the first non-zero value in the impulse response of filter  $F_{i+1}$ , comes immediately following the last non-zero value in the impulse response of filter  $F_i$ . Hence this summation of filters results in a single, longer filter with its impulse response being the sum of the impulse responses of the N component filters.

The important property of this filter is the length/latency efficiency, K, which is higher than any of the component filter efficiencies.

That is, the filter of Figs. 10 and 12 uses the technique of adding together several filters to form a new filter which is as long as the sum of the component filter lengths, and whose latency is as short as the latency of the lowest-latency component filter.

Particularly, the composite filter assembly of Fig. 12 utilises the technique of combining a first time-domain (low latency) filter with additional fast-convolution (longer latency) filters to maximise filter length while minimising latency. This technique is implemented by adding together N filters,  $F_1, F_2, \dots, F_N$  where  $F_1$  is a filter with very low latency, implemented with time-domain techniques, and the

other filters.  $F_i$  are each implemented with fast-convolution techniques. More specifically, the assembly adopts the technique whereby the  $N-1$  fast-convolution filters,  $F_i$ , are composed of a sequence of filters, each with longer filter length than its predecessor, and hence each with longer latency, but still preserving the property that  $d_{i+1} = d_i + l_i$ . This ensures that the filter,  $F$ , which is made by summing together the  $N$  component filters, has an impulse response without any "holes" in it.

10 With particular reference to Fig. 12 the composite filter  $F$  comprises five filter portions  $F_1$ ,  $F_2$ ,  $F_3$ ,  $F_4$  and  $F_5$ . The impulse response  $a_k$  of the composite filter  $F$  is illustrated at the top of Fig. 12 and has a total sample length of 1024 samples.

15 Filter  $F_1$  has an impulse response comprising the first 64 samples of the impulse response  $a_k$ . That is, the filter has a length of 64 samples. The filter as implemented has a low latency filter (such as is referenced in Motorola document APR 7/D in respect of the DSP 56000 series of  
20 Integrated Circuits). This filter has an effective latency of 0.

The subsequent filters  $F_2$ ,  $F_3$ ,  $F_4$ ,  $F_5$  are implemented using fast convolution digital techniques. Fig. 15 illustrates the basic algorithm for such techniques which  
25 comprises taking the fast Fourier transform of the incoming sampled data, multiplying the transformed data samples by the impulse response of the filter, converting the fast Fourier transformed data samples back to the time domain by use of an

inverse fast Fourier transform and then outputting the data. An overlap/discard method is used whereby only a portion of the output data is utilised.

5 The length and latency of the additional filters F2, F3, F4, F5 is selected according to the rule illustrated diagrammatically in Fig. 13, whereby each filter portion has a latency equal to the sum of the length and the latency of the immediately preceding filter portion.

10 In this case the end result is a filter having a total length of 1024 samples and a latency of 0.

Fig. 14 illustrates a variation of the filter of Fig. 8 wherein delay is introduced after the filter algorithm is applied in the frequency domain.

### 3. Optimized Real String Handling

15 With reference to Fig. 16, a common method of frequency analysis is via the Discrete Fourier Transform (hereafter referred to as the DFT), which can be implemented efficiently in electronic apparatus using the Fast Fourier Transform algorithm (hereafter referred to as the FFT).

20 The DFT is formulated to operate efficiently when its input data and output data are both complex (having a real and imaginary component). When the data input to the DFT is real, the output data from the operation will contain some redundancy, indicating that some of the processing that led to this output data was unnecessary.

25 In this embodiment what is described is a new transform for operating on real numbers in the digital environment, that has many of the same applications as the DFT, but

**SUBSTITUTE SHEET**

without the inefficiencies of the DFT for operation on real numbers. For the purposes of this document, the algorithm described herein will be named the Modified Discrete Fourier Transform (MDFT).

The DFT is computed according to the equation below:

$$X(n) = \sum_{k=0}^{N-1} x(k) e^{\frac{-2\pi jnk}{N}} \quad (1)$$

If the input data  $x(k)$  is real (ie. it has no imaginary component), the output data  $X(n)$  has the following properties:

$$\begin{aligned} X(0) & \in \mathfrak{R} \\ X\left(\frac{N}{2}\right) & \in \mathfrak{R} \\ X(n) & = [X(N-n)]^* \text{ for } 0 < n < \frac{N}{2} \end{aligned} \quad (2)$$

Where the \* operator is used to signify complex conjugation. This means that the imaginary part of  $X(0)$ , the imaginary part of  $X(N/2)$  and all  $\{X(n): N/2 < n < N\}$  are redundant. The process of extracting only the necessary information out of the DFT output is therefore not trivial.

An alternative transform is shown below:

$$Y(n) = \sum_{k=0}^{N-1} x(k) e^{\frac{-2\pi jk(n+\frac{1}{2})}{N}} \quad (3)$$

This is like a standard DFT except that the output vector  $Y(n)$  represents the signal's frequency components at different frequencies to the DFT. The output vector  $Y(n)$  has redundancies (just as the DFT output  $X(n)$  has), except that

**SUBSTITUTE SHEET**

the redundant part of the data is more clearly extracted from  $Y(n)$  than from  $X(n)$ . The redundancy in  $Y(n)$  that results when  $x(k)$  is real can be expressed as follows:

$$Y(n) = [Y(N-1-n)]^* \quad (4)$$

5 This implies that the second half of this vector  $Y(n)$  is simply the complex conjugate of the first half, so that only the first half of the output vector is required to contain all of the information, when  $x(k)$  is real.

10 An alternative view of the above equation is that all of the odd elements of the vector are simply the complex conjugate of the even elements:

$$\begin{aligned} Y(1) &= [Y(N-2)]^* \\ Y(3) &= [Y(N-4)]^* \\ &\dots \\ Y(N-3) &= [Y(2)]^* \\ Y(N-1) &= [Y(0)]^* \end{aligned} \quad (5)$$

15 This means that we only need to compute the even elements of  $Y(n)$  to obtain all of the required information from the modified DFT of the real signal  $x(k)$ . We can name the array  $Z(p)$  the array that contains the even elements from  $Y(n)$ , as follows:

20

$$Z(p) = Y(2p) \text{ for } 0 \leq p < \frac{N}{2} \quad (6)$$

Based on our previous equation for  $Y(n)$ , we get:

$$Z(p) = Y(2p) = \sum_{k=0}^{N-1} x(k) e^{\frac{-2\pi j k (2p + \frac{1}{2})}{N}} \quad (7)$$

which after some manipulation becomes:

$$25 \quad Z(p) = \sum_{k=0}^{\frac{N}{2}-1} [x(k) - jx(k + \frac{N}{2})] e^{\frac{-\pi j k}{N}} e^{\frac{-2\pi j p k}{(N/2)}} \quad (8)$$

if we create an  $N/2$  length complex vector from the  $N$  length real vector:

$$x'(l) = [x(l) - jx(l + \frac{N}{2})]e^{\frac{-\pi j l}{N}} \text{ for } 0 \leq l < N/2 \quad (9)$$

then we can see that:

$$5 \quad Z(p) = DFT_{(N/2)}[x'(l)] \quad (10)$$

This means that we have computed the vector  $Z(p)$  by using a DFT of length  $N/2$ .

We say that  $Z(p) = MDFT[x(k)]$  (where MDFT indicates the Modified Discrete Fourier Transform operator). The procedure to follow for computing  $Z(p)$  is then as follows:

1. Take the input vector  $x(k)$  of length  $N$ , where each element of  $x(k)$  is real.
2. Create the vector  $x'(l)$ , a complex vector of length  $N/2$  by the method of equation (9) above.
- 15 3. Compute the  $N/2$  point DFT of  $x'(l)$  to give the  $N/2$  complex result vector  $Z(p)$ .

The MDFT has many properties that make it useful in similar applications to the DFT. Firstly, it can be used to perform linear convolution in the same way as the DFT. Secondly, it has an inverse transform that looks very similar to the forward transform:

$$x'(k) = IDFT_{(N/2)}[Z(p)] \quad (11)$$

where IDFT indicates the  $N/2$  point Inverse Discrete Fourier

Transform.

The algorithm can be implemented in an electronic apparatus supplied with a set of  $N$  real numbers and producing  $N/2$  complex output numbers, representing the MDFT of the input data. This apparatus uses digital arithmetic elements to perform each of the arithmetic operations as described in the preceding text.

Another embodiment of the present invention is a pair of apparatus, the first of which computes an MDFT as described in the previous paragraph, and the second of which computes an inverse MDFT, using the arithmetic procedures described previously in this document. By passing overlapped blocks of data from a continuous stream of input data through the MDFT computer, then multiplying the  $Z(p)$  coefficients by appropriate filter coefficients, then passing the resulting data through the Inverse MDFT computer, and recombining segments of output data appropriately, a modified Fast Convolution processor can be built.

The above described a modification to the DFT that makes it more useful in a number of applications particularly but not limited to the real time filter applications previously described. All of these extensions to the DFT can also be applied to the FFT algorithm and other fast implementations of the DFT.

25

#### Example 1

Fig. 17 illustrates an implementation of the summed filter of Fig. 11 wherein the Modified Discrete Fourier Transform (MDFT) described immediately above is applied for

the purposes of transforming the data stream into the frequency domain and the corresponding Inverse Modified Discrete Fourier Transform (IMDFT) is applied following application of the filter algorithm and prior to discard for  
5 conversion from the frequency domain.

In filter F2 of Fig. 17, the MDFT takes 64 real words of input and produces 32 complex words of output. The IMDFT takes 32 complex words of input and produces 64 real words of output.

10 In filter F3 of Fig. 17 the MDFT takes 256 real words of input and produces 128 complex words of output. The IMDFT takes 128 complex words of input and produces 256 real words of output.

The filter of Fig. 17 is implemented using a Motorola  
15 DSP 56001 processor incorporating software (bootable from ROM or from another host computer) to implement the algorithm. The delay elements are implemented using a bank of external memory chips comprising three MCM 6206 memory chips.

Data input and output between the analog and digital  
20 domains is effected by an ADC and DAC chip, the Crystal CS 4216, communicating via the synchronous serial communication port of the DSP 56001.

#### INDUSTRIAL APPLICABILITY

Embodiments of the invention may be applied to digital  
25 filters implemented in software, hardware or a combination of both for applications such as audio filtering or electronic modelling of acoustic system characteristics. The method is broadly applicable in the field of signal processing and can

be used to advantage, for example. in: adaptive filtering;  
audio reverberation processing; adaptive echo cancellation;  
spatial processing; virtual reality audio; correlation.  
radar; radar pulse compression; deconvolution; seismic  
5 analysis; telecommunications; pattern recognition; robotics;  
3D acoustic modelling; audio post production (including  
oralisation, auto reverberant matching); audio equalisation;  
compression; sonar; ultrasonics; secure communication  
systems; digital audio broadcast, acoustic analysis;  
10 surveillance; noise cancellation; echo cancellation.

The above describes only some embodiments of the  
present invention and modifications obvious to those skilled  
in the art can be made thereto without departing from the  
scope and spirit of the present invention.

**SUBSTITUTE SHEET**

**WHAT IS CLAIMED IS:**

1. A method for building a digital filter with a long impulse response and low latency, built by operating a number of smaller component filters in parallel and combining their outputs by addition, with each component filter operating with a different delay such that the net operation of the ensemble of said component filters is the same as a single filter with a longer impulse response, and the latency of the ensemble is equal to the shortest  
10 latency of said component filters wherein one or more of said component filters is implemented as a time-domain finite impulse response filter, built with multiply and add operations, and the remainder of said component filters are implemented using a fast convolution method, such that time-domain filter(s) provides the lowest latency portion of the ensemble impulse, and fast convolution filter(s) provide longer filter components, wherein said fast convolution filters are built using the Discrete Fourier Transform or the Fast Fourier Transform.

20 2. A method for building a digital filter with a long impulse response and low latency built by operating a number of smaller component filters in parallel and combining their outputs by addition, with each component filter operating with a different delay such that the net operation of the ensemble of said component filters is the same as a single filter with a longer impulse response, and the latency of the ensemble is equal to the shortest latency of said component filters wherein one or more of said component filters is implemented as a time-domain  
30 finite impulse response filter, built with multiply and add

-32-

operations, and the remainder of said component filters are implemented using a fast convolution method, such that time-domain filter provides the lowest latency portion of the ensemble impulse response, and fast-convolution filter provide the longer filter components, wherein said fast-convolution filters are built using a Modified Discrete Fourier Transform method comprising the steps of:

receiving an input vector  $x(k)$  of length  $N$ , where each element of  $x(k)$  is real;

10 means for creating a complex vector  $x'(l)$  of length  $N/2$  according to the following:

$$x'(l) = \left[ x(l) - jx\left(l + \frac{N}{2}\right) \right] e^{-\frac{\pi j l}{N}} \text{ for } 0 \leq l < \frac{N}{2}; \text{ and}$$

computing an  $N/2$  point Discrete Fourier Transform (DFT) of  $x'(l)$  to give the  $N/2$  complex result vector  $Z(p)$ :

$$Z(p) = DFT\left(\frac{N}{2}\right)[X'(l),]$$

and a complementary Inverse Modified Discrete Fourier Transform method.

3. A composite electrical filter comprising a plurality of filter modules arranged to receive in parallel  
 20 an incoming input signal for filtering, an output from each of said filter modules being summed to produce a composite filtered output signal, each of said filter modules having an impulse response that is a selected portion of said impulse response of the composite filter wherein the number of said filter module is  $N$ , comprising filters  $F_1, F_2, \dots, F_N$  and wherein filter modules  $F_1$  is a filter with a very low latency implemented with time domain techniques while all other filter modules  $F_i$  are implemented with fast

-33-

convolution means and these fast convolution filters  $F_i$  are composed of a sequence of filters, each with longer filter length than its predecessor and hence each with longer latency, but still preserving the property that said composite filter  $F$  output formed by summing together  $N$  component filter outputs has an impulse response without any "holes" in it.

4. A digital filter having an impulse response that includes a low latency and a long filter length, said  
10 digital filter comprising a plurality of component filters each having an impulse response including a latency and a filter length, each of said component filter having an input connected to an input of said digital filter, and an output connected to a summing means for combining, by addition, each of said outputs to form an output of said digital filter such that the impulse response of said digital filter is the sum of the component impulse responses and the low latency is equal to the shortest one of the component latencies.

20 5. The digital filter as claimed in claim 4, wherein at least a first one of said component filters is specifically configured to provide said low latency, and at least another one of said component filters is specifically configured to provide a long filter length.

6. The digital filter as claimed in claim 5 wherein, excepting said first component filter, a first non-zero impulse component of each remaining component filter occurs during or immediately following the impulse response of another one of said component filters.

-34-

7. The digital filter as claimed in claim 6 wherein at least one of said component filters comprises a fast convolution filter having a transform element, a filter element, and an inverse transform element connected in cascade.

8. The digital filter as claimed in claim 7 comprising a first fast convolution filter configured to provide said low latency and at least one other fast convolution filter configured to provide said long filter  
10 length.

9. The digital filter as claimed in claim 7 wherein at least one of said component filters is implemented as a time-domain finite impulse response filter including multiply and add operations and the remainder of said component filters, of which there is at least one, are each implemented using a fast convolution filter such that the time domain filter provides said low latency and said fast convolution filter substantially provide said long filter length.

20 10. The digital filter as claimed in claim 7 wherein at least one of said fast convolution filter comprises a modified convolution filter in which said filter element comprises a plurality of filter sub-elements arranged in parallel and each configured to receive a transformed signal output from said transform element, each of said filter sub-elements supplying a corresponding input of an adding element that outputs to said inverse transform element.

-35-

11. The digital filter as claimed in claim 10 wherein each of said filter sub-element comprises a delay unit that delays application of said transformed signal to a corresponding multiplication unit where said delayed transformed signal is multiplied by filter coefficients preselected to form at least a part of said impulse response, each of said multiplication units outputting to said adding element.

12. The digital filter as claimed in claim 11  
10 wherein said delay units delay said transformed signal by an integer multiple of a predetermined delay period prior to applying same to the corresponding multiplication unit.

13. The digital filter as claimed in claim 12 wherein said filter element comprises  $N-1$  ( $N$  being an integer and  $\leq 2$ ) of said delay units connected in cascade and each of said predetermined delay, a first one of said delay units being input with said transformed signal and  $N$  of said multiplication units, a first one of which being input with said transformed signal and the remainder being  
20 input with the output of a corresponding one of said delay units.

14. The digital filter as claimed in claim 10, comprising two or more of said component filters, each of which are formed as said modified convolution filter.

15. The digital filter as claimed in claim 10 wherein at least one of said component filters is a time-domain filter configured to provide low latency.

-36-

16. The digital filter as claimed in claim 15 wherein said time domain filter comprises zero latency.

17. The digital filter as claimed in claim 16 comprising said zero latency time domain filter, one or more modified convolution filters and one or more fast convolution filters.

18. The digital filter as claimed in claim 15 wherein said time domain filter is a finite impulse response filter.

10 19. The digital filter as claimed in claim 7 wherein said fast-convolution filter(s) perform one of a Discrete Fourier Transform or a Fast Fourier Transform.

20. The digital filter as claimed in claim 7 wherein the fast-convolution filters are built using a Modified Discrete Fourier Transform method comprising the steps of:

receiving an input vector  $x(k)$  of length  $N$ , where each element of  $x(k)$  is real;

20 creating a complex vector  $x'(l)$  of length  $N/2$  according to the following:

$$x'(l) = \left[ x(l) - jx\left(l + \frac{N}{2}\right) \right] e^{-\frac{\pi j l}{N}} \text{ for } 0 \leq l < \frac{N}{2}; \text{ and}$$

computing an  $N/2$  point Discrete Fourier Transform (DFT) of  $x'(l)$  to give the  $N/2$  complex result vector  $Z(p)$ :

$$Z(p) = DFT\left(\frac{N}{2}\right)[X'(l),]$$

-37-

and a complementary Inverse Modified Discrete Fourier Transform.

21. The digital filter as claimed in claim 4 wherein said low latency is sufficiently short so as to permit said digital filter to be used to process live audio signals.

22. A digital filter comprising a plurality of filter modules arranged in parallel and each configured to receive an input signal for filtering, an output of each of said filter modules being summed to produce a composite filtered output signal, each of said filter modules having a corresponding filter length and a corresponding impulse response that is a selected portion of a desired impulse response of said digital filter, wherein the length of each of said filter modules is different, and wherein at least the shortest length filter module is a time-domain low latency filter configured to process at least first portions of said desired impulse response and the longer length filter module is a fast-convolution longer latency filter configured to process subsequent portions of said desired impulse response.

23. The digital filter of claim 22, where the number of said filter modules is  $N$  comprising filters  $F_1, F_2, \dots, F_N$  and wherein filter modules  $F_1$  is a filter with a very low latency implemented with time domain techniques while all other filter modules  $F_i$  ( $2 \leq i \leq N$ ) are implemented with fast convolution techniques, each of said filters  $F_i$  being of longer filter length ( $l$ ) than its predecessor and hence each with longer latency ( $d$ ), but preserving the

-38-

property that  $d_{i+1}=d_i+1_i$ , thereby ensuring that the composite filtered output formed by summing together the N component filter outputs forms a substantially continuous impulse response.

24. A digital filter with a long impulse response and low latency comprising a smaller number of component filters in parallel and adding means for combining the output of said component filters, with each component filter operating with a different delay such that  
10 the net operation of said component filters is the same as a single filter with a longer impulse response, and the latency of said component filters is equal to a shortest latency of the said component filters wherein one or more of said component filters comprise a time-domain finite impulse response filter and multipliers and adders, and the remainder of said component filters comprise a fast convolution means, such that time-domain filter provide the lowest latency portion of said component filters impulse response, and the fast-convolution means provide the longer  
20 filter components and said fast-convolution means comprise Discrete Fourier Transformer or the Fast Fourier Transformer.

25. A digital filter with a long impulse response and low latency comprising a smaller number of component filters in parallel and adding means for combining the output of said component filters, with each component filter operating with a different delay such that the net operation of said component filters is the same as a single filter with a longer impulse response, and the  
30 latency of said component filters is equal to a shortest

-39-

latency of the said component filters wherein one or more of said component filters comprise a time-domain finite impulse response filter and multipliers and adders, and the remainder of said component filters comprise a fast convolution means, such that time-domain filter provide the lowest latency portion of said component filters impulse response, and the fast-convolution means provide the longer filter components and said fast-convolution means comprise a Modified Discrete Fourier Transform processor comprising:  
 10 means for receiving an input vector  $x(k)$  of length  $N$ , where each element of  $x(k)$  is real;

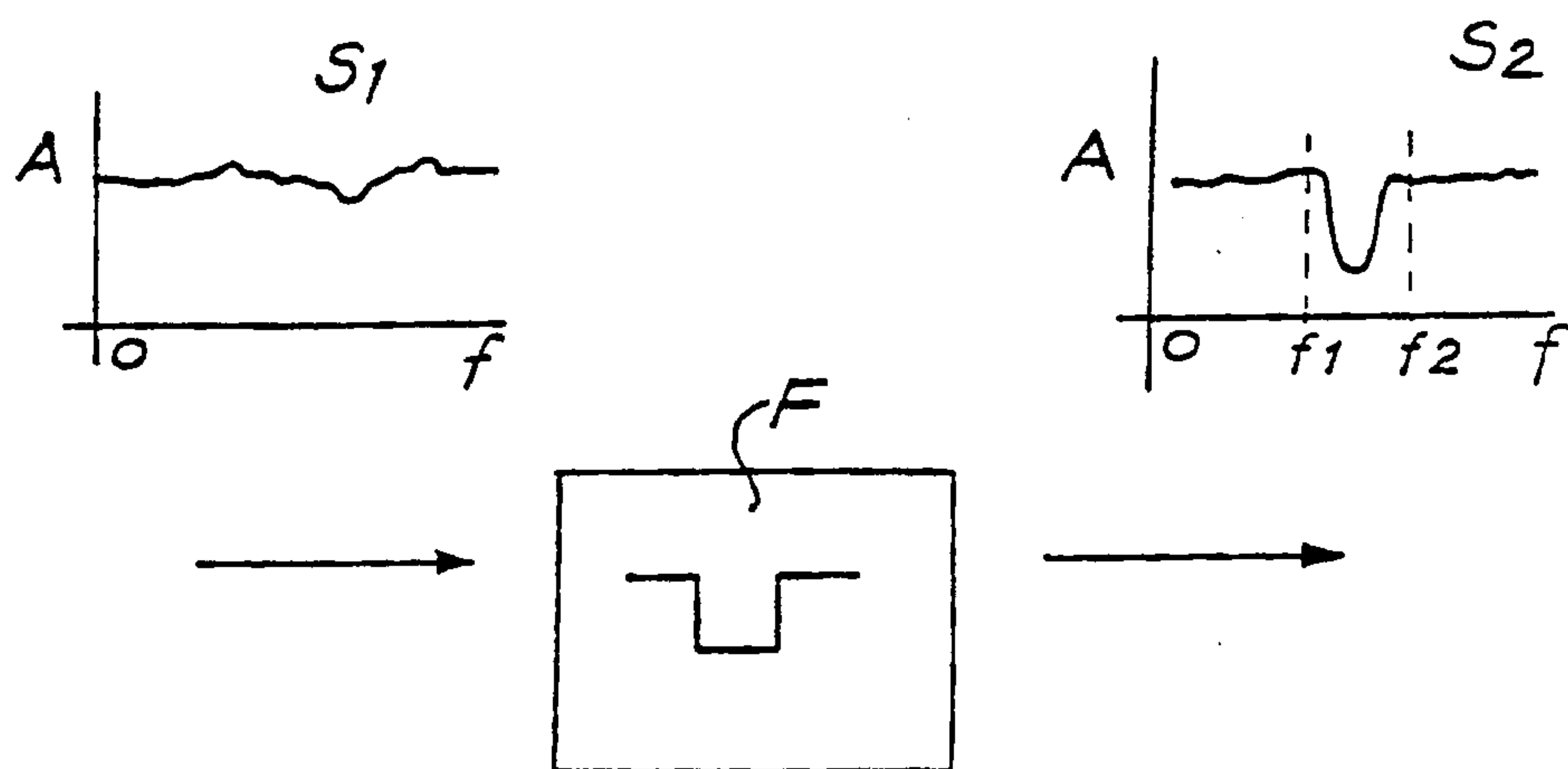
means for creating a complex vector  $x'(l)$  of length  $N/2$  according to the following:

$$x'(l) = [x(l) - jx([l + N/2])]e^{-xjl/N} \text{ for } 0 \leq l < N/2$$

and means for computing an  $N/2$  point Discrete Fourier Transform (DFT) of  $x(l)$  to give the  $N/2$  complex result vector  $Z(p)$ :

$$Z(p) = DFT_{(n/2)}[x'(l)]$$

and a complementary Inverse Modified Discrete  
 20 Fourier Transform processor.



*FIG. 1*

(PRIOR ART)

2/17

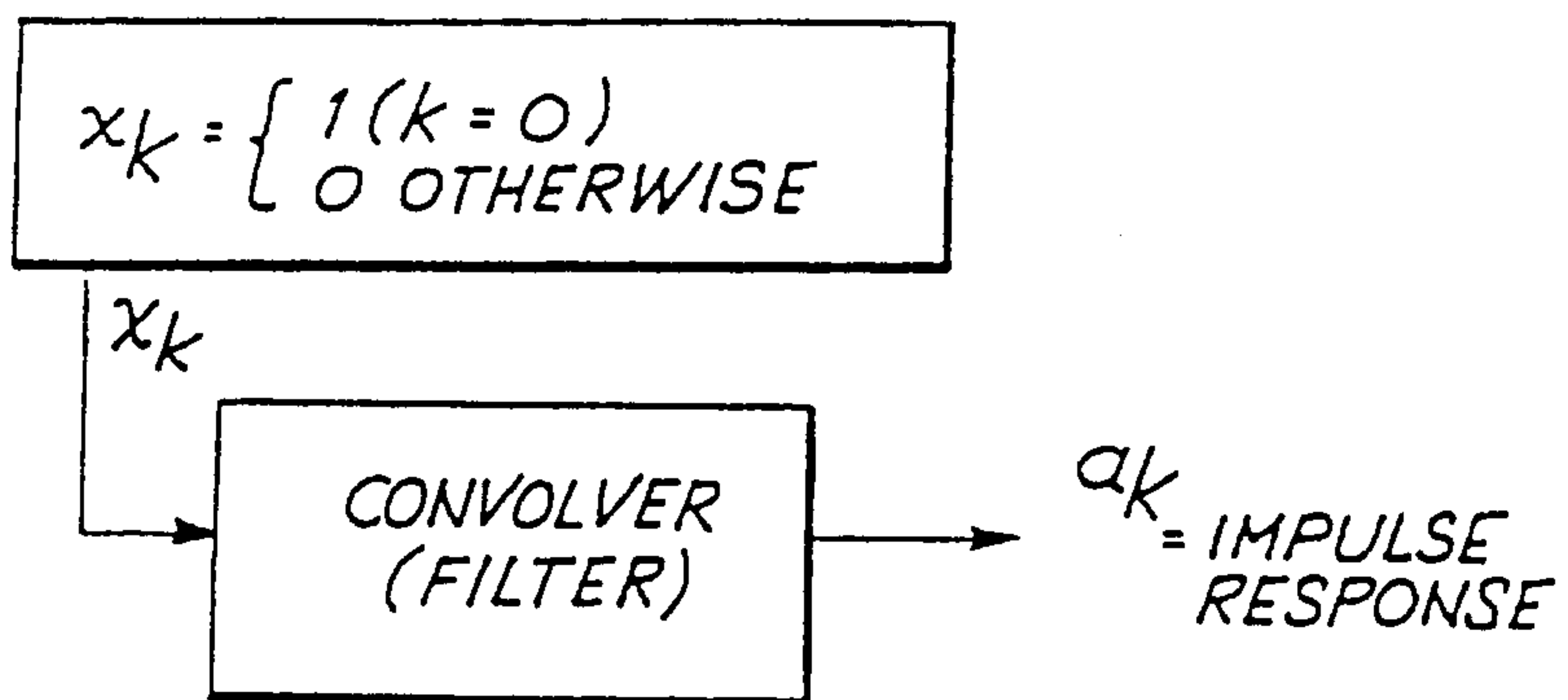


FIG. 2

(PRIOR ART)

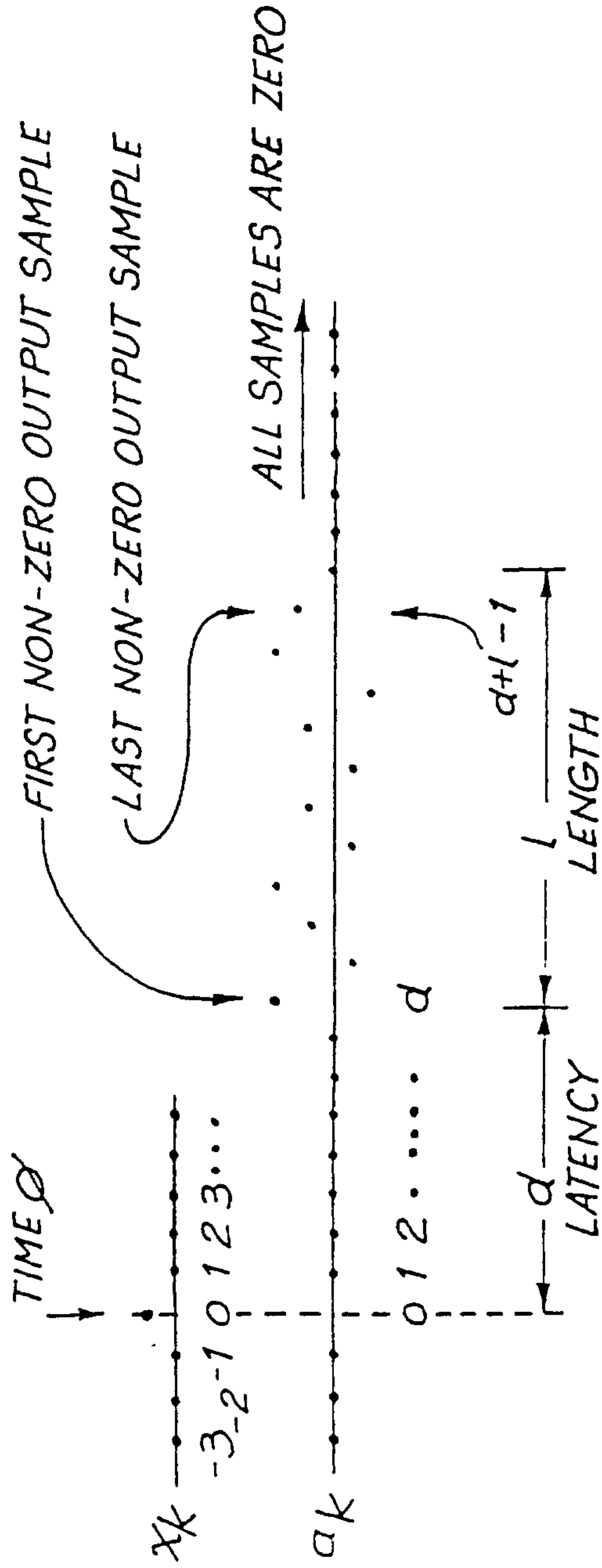


FIG. 3

(PRIOR ART)

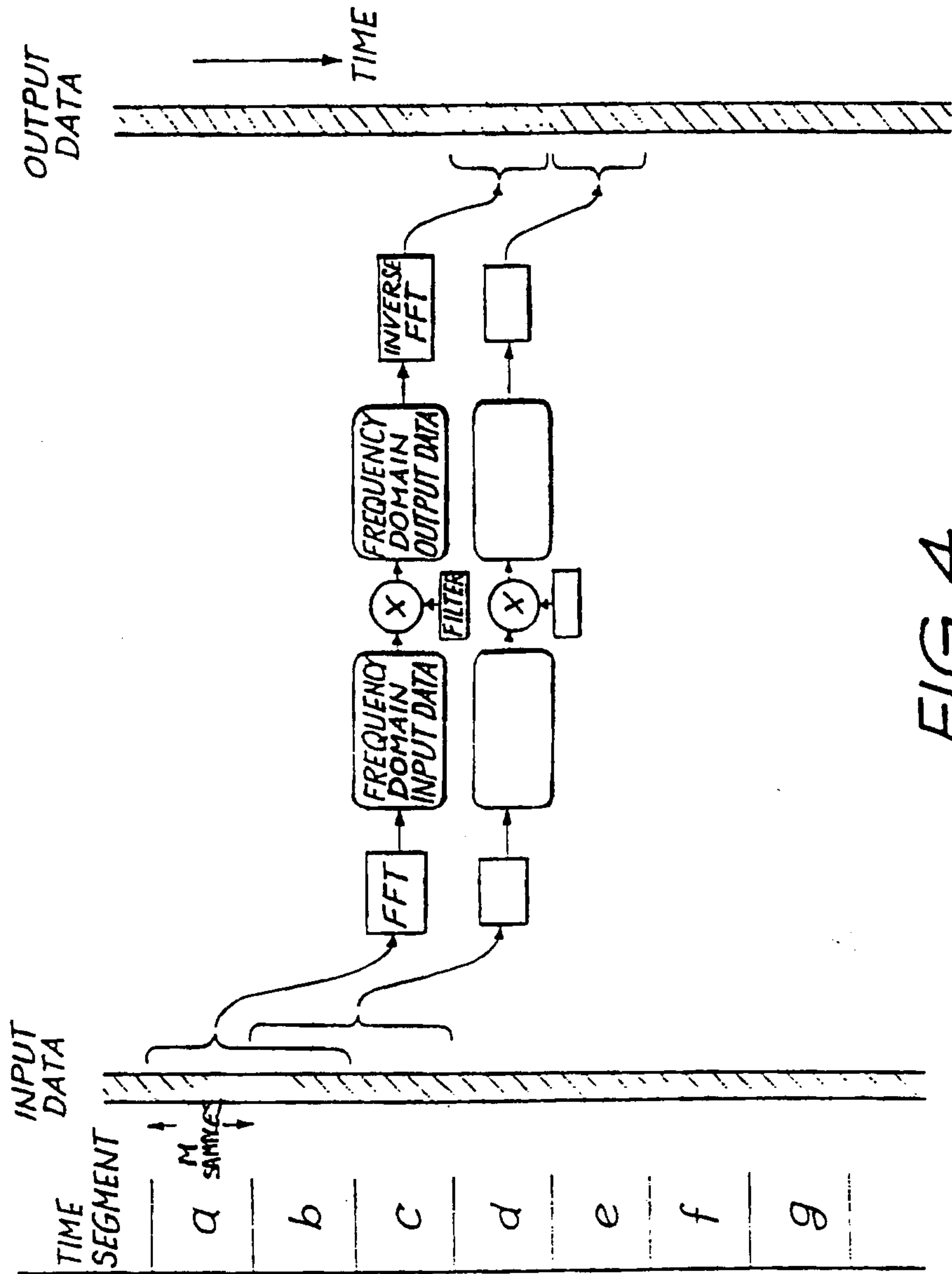


FIG. 4

(PRIOR ART)

5/17

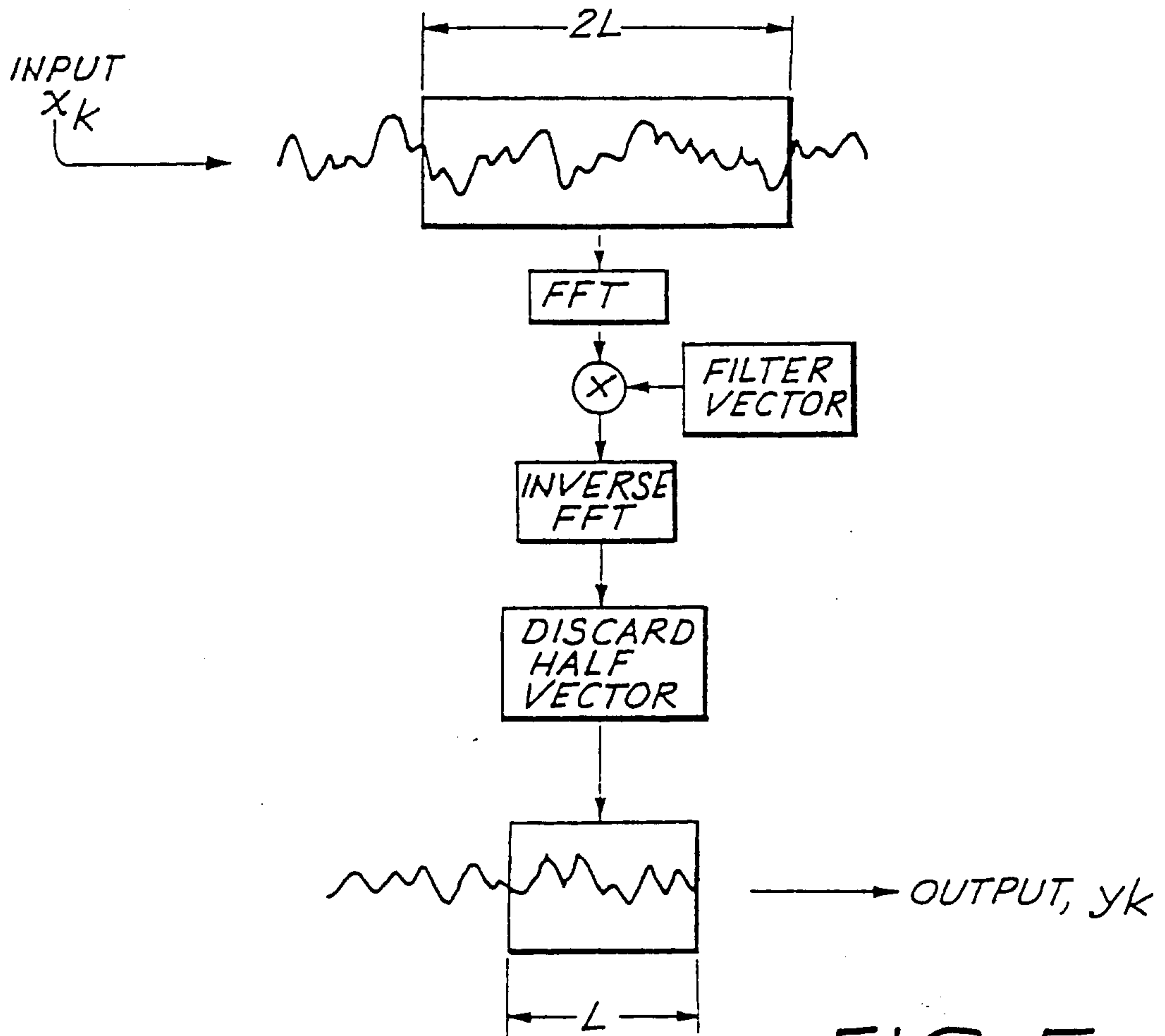


FIG. 5

(PRIOR ART)

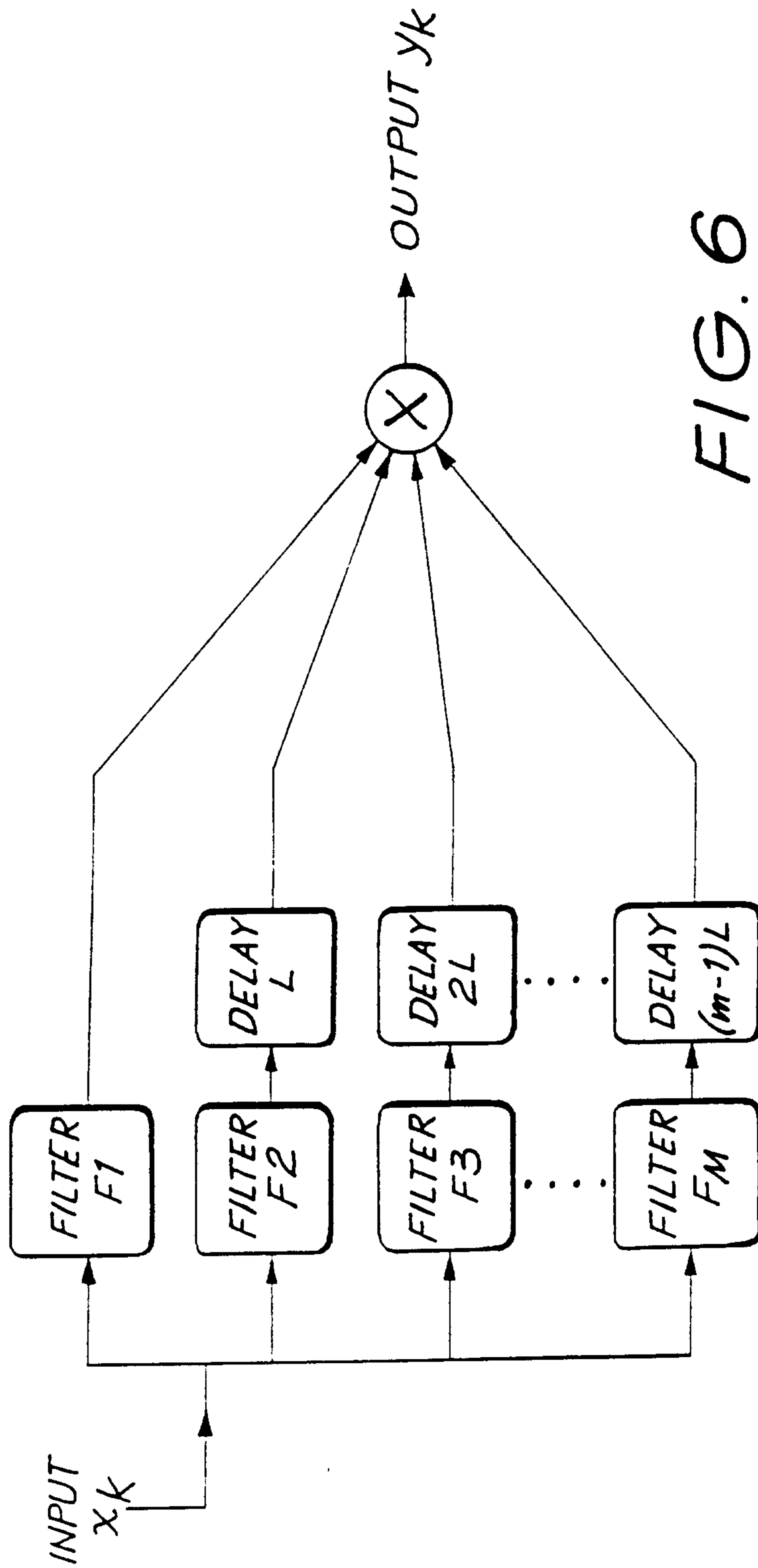


FIG. 6

7/17

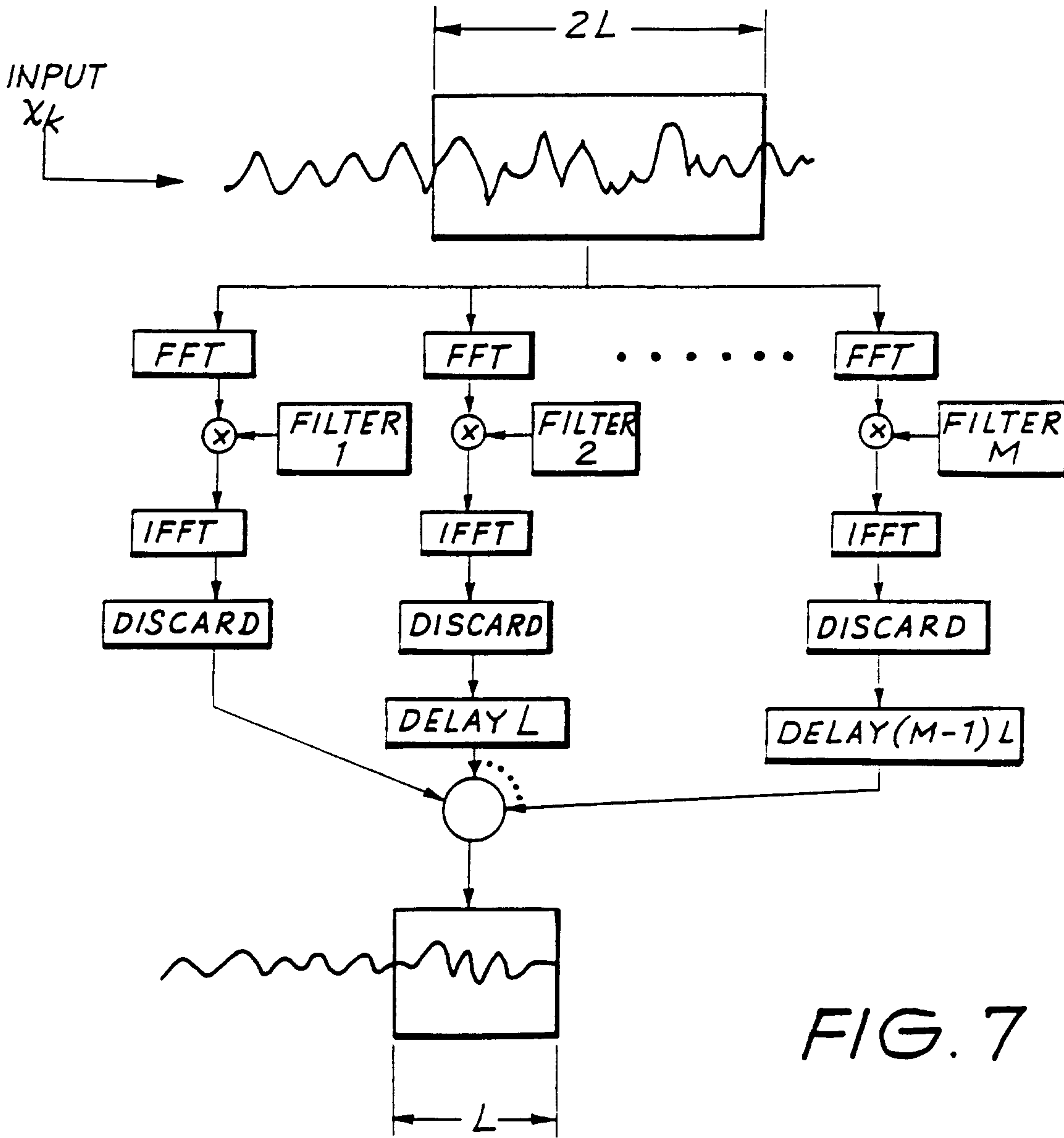


FIG. 7

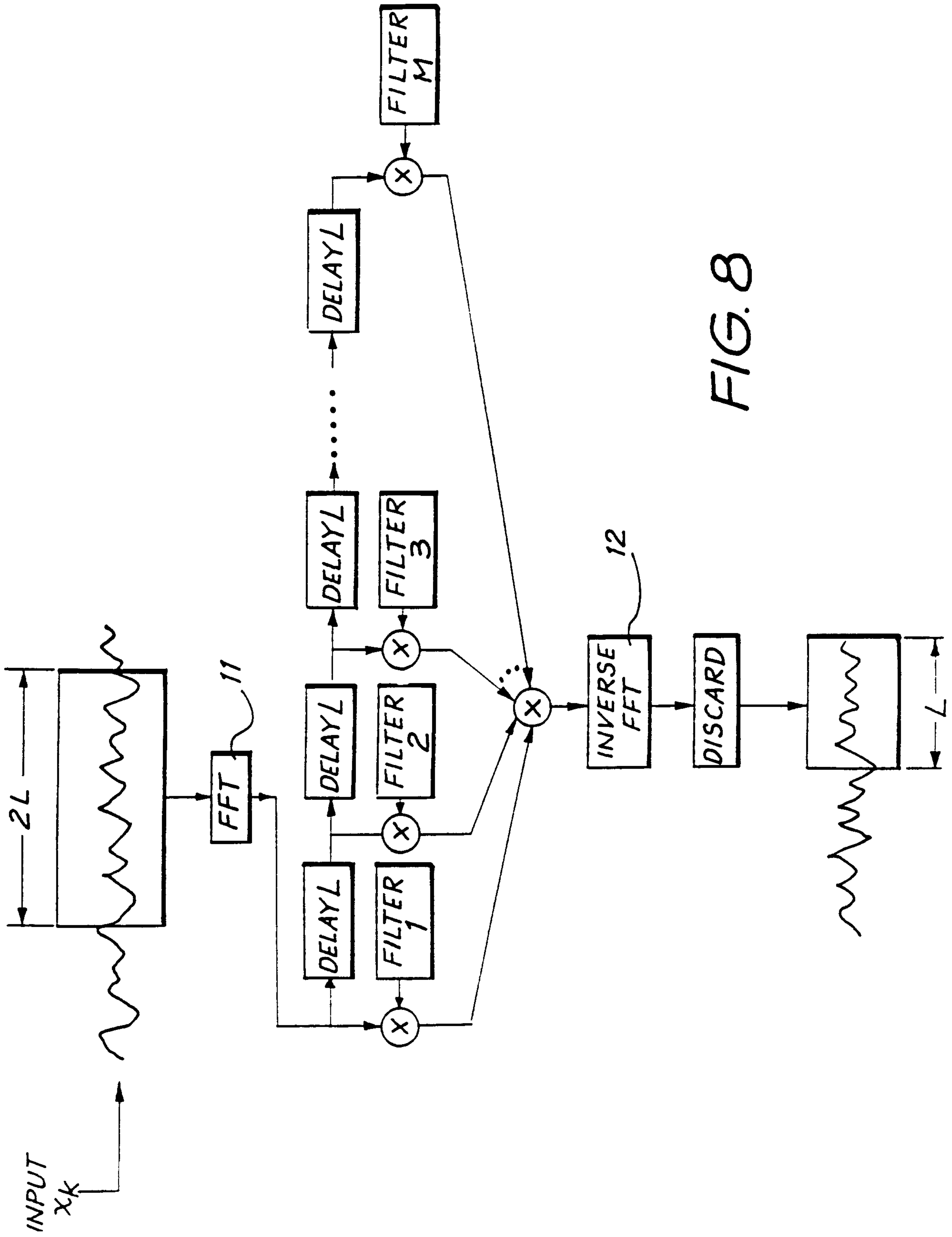


FIG. 8

9/17

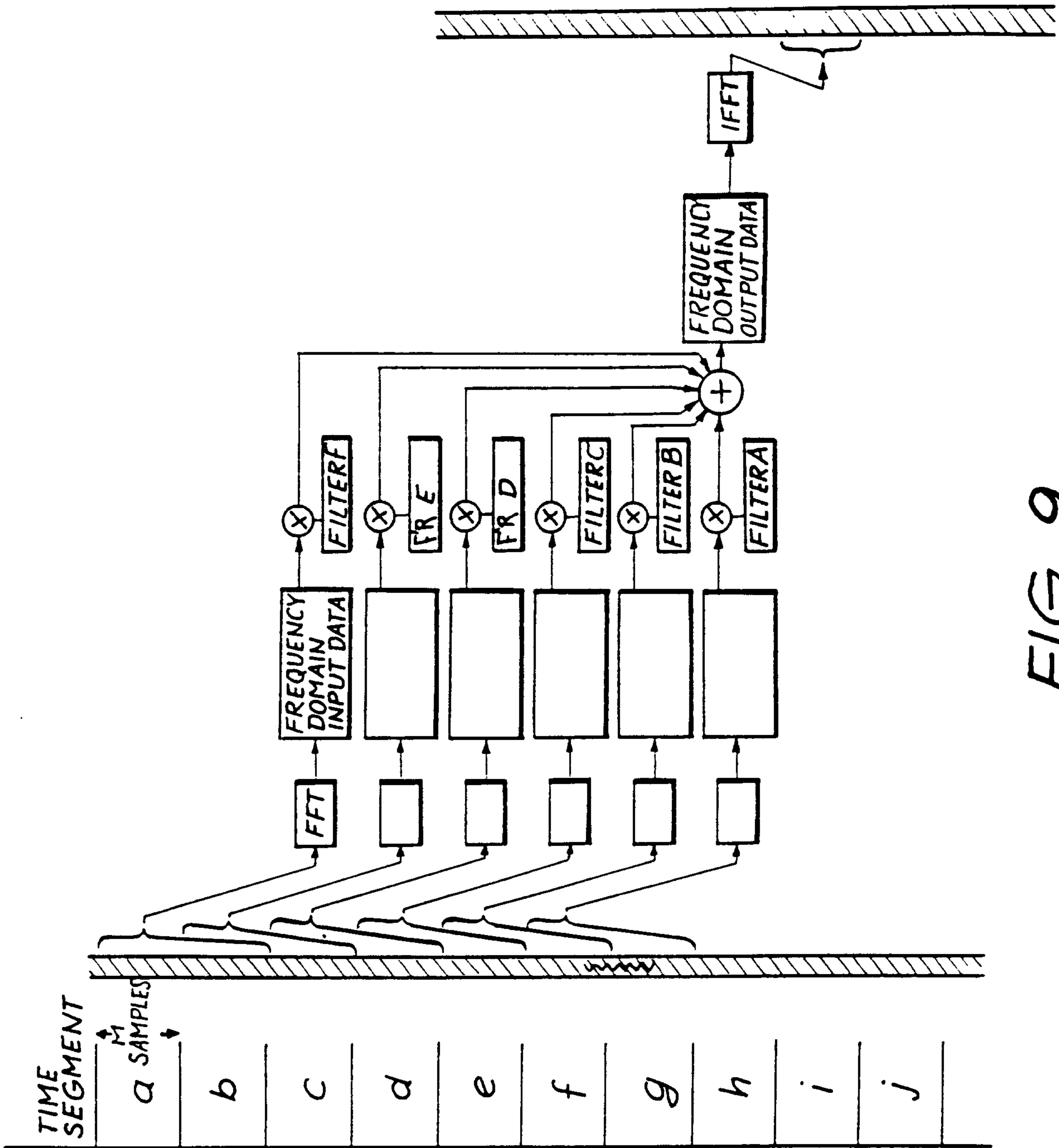


FIG. 9

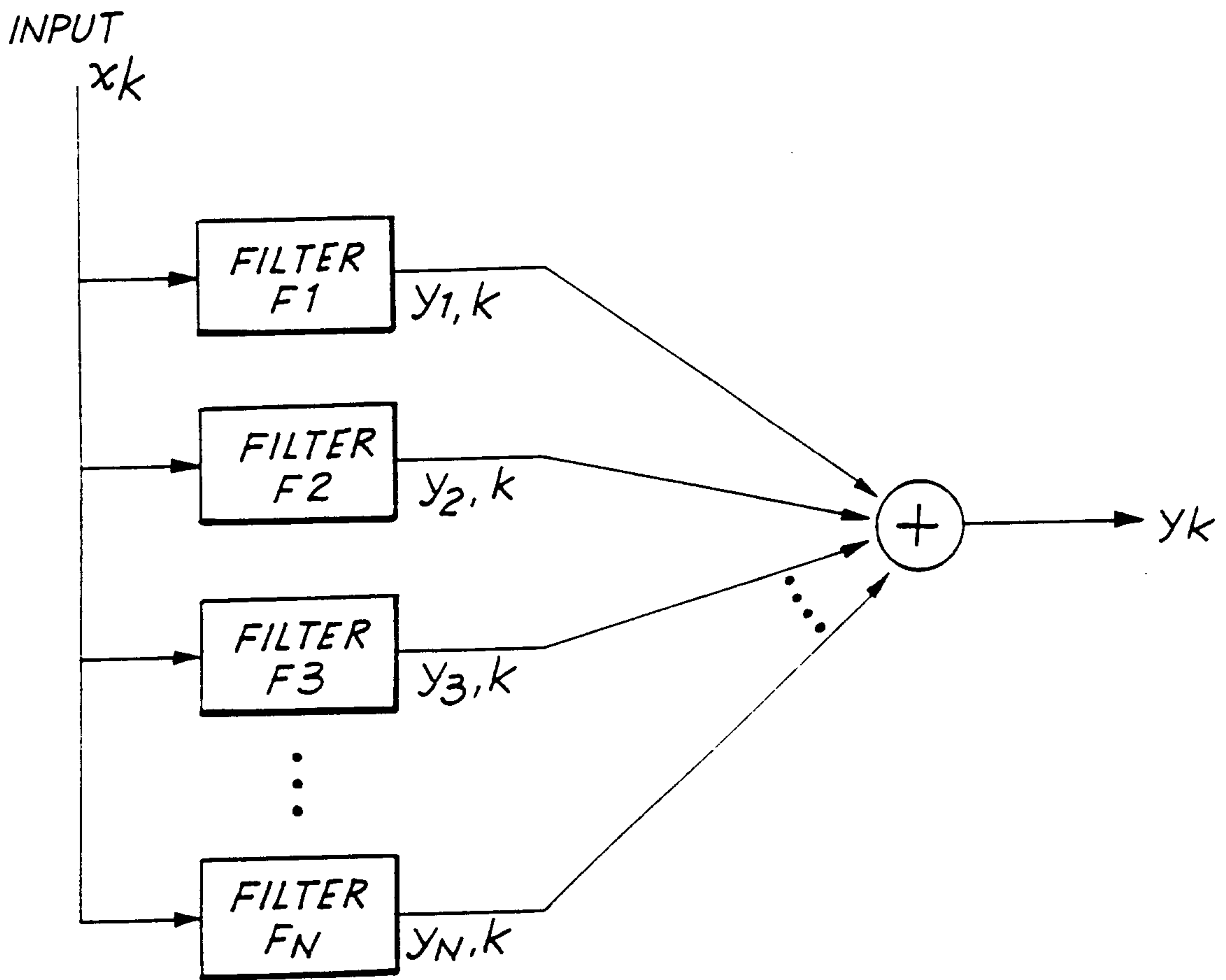


FIG. 10

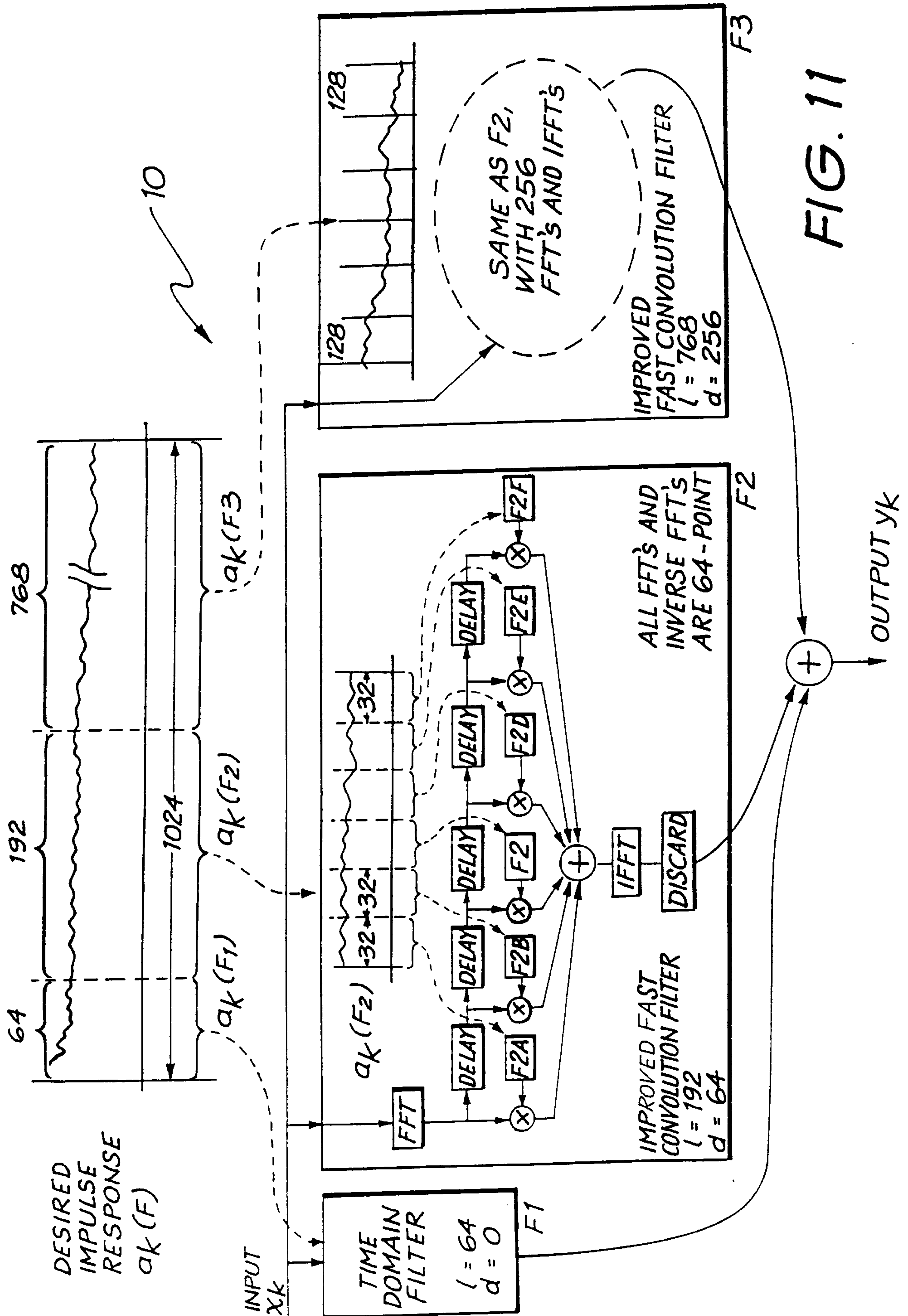


FIG. 11

12/17

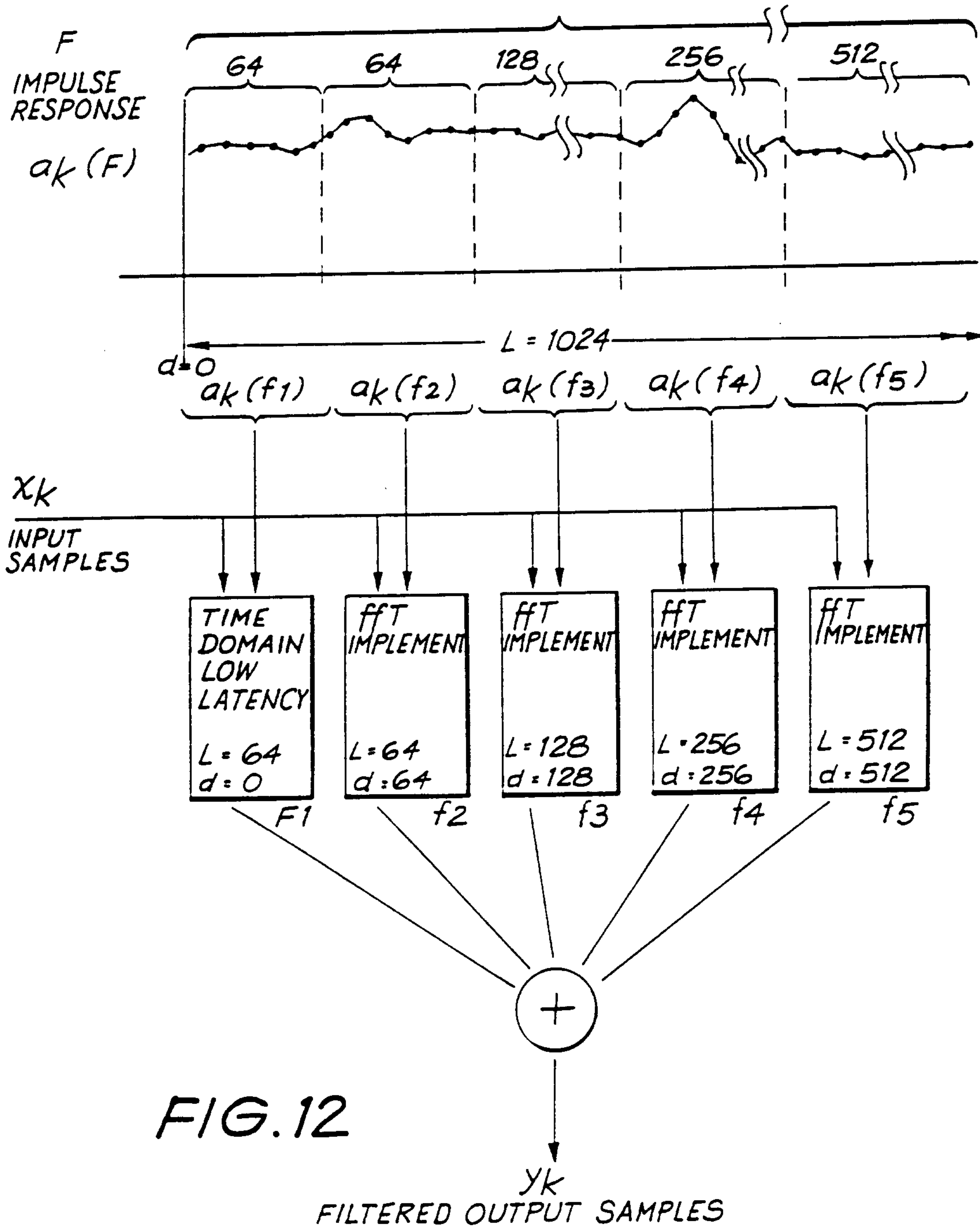
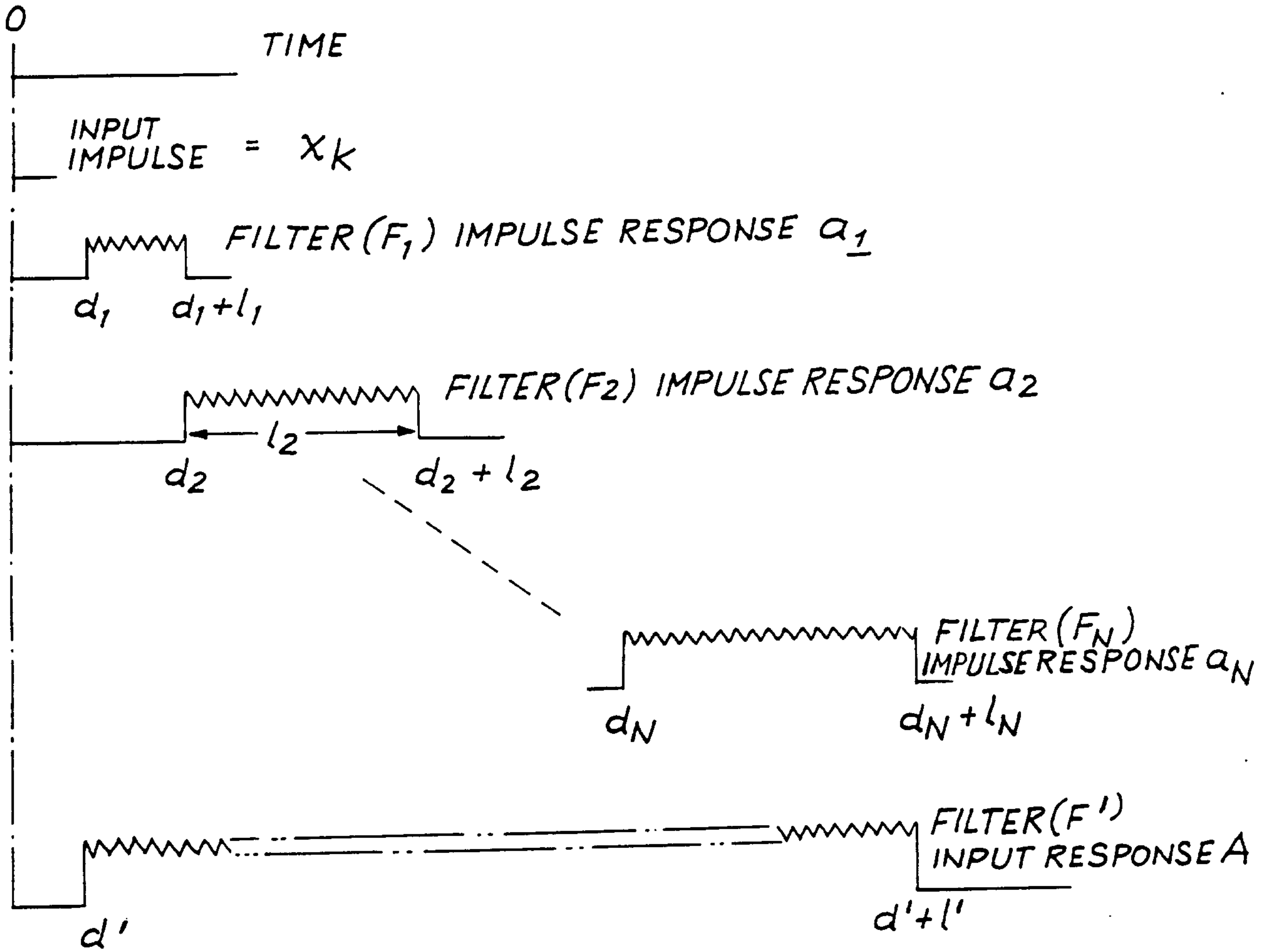


FIG.12



THE FILTER  $F'$  IS MADE BY SUMMING TOGETHER THE OUTPUTS OF FILTERS  $F_1, F_2, \dots, F_N$ .

EACH FILTER  $F_i$  DEFINES OUTPUTS  $y_{i,k}$  THE OUTPUT OF FILTER  $F_i$  AT TIME SAMPLE  $k$ , AS BEING

$$y_{i,k} = \sum_{n=d_i}^{d_i+l_i-1} a_{i,n} x_{k-n}$$

IF WE SET  $y'_k = \sum_{n=1}^N y_{i,k}$

THEN WE HAVE  $y'_k = \sum_{n=d'}^{d'+l'-1} A_n x_{k-n}$

FIG. 13

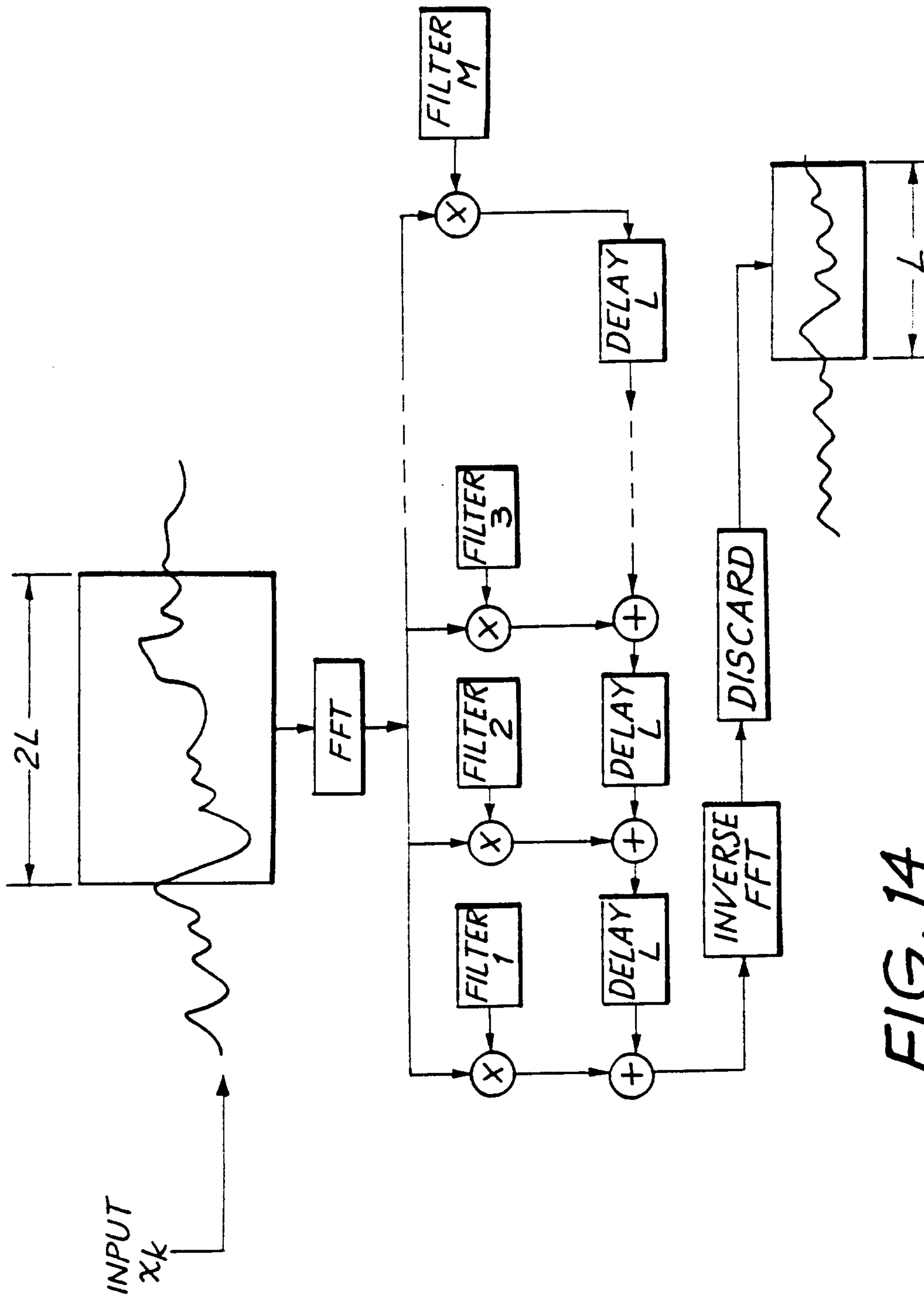


FIG. 14

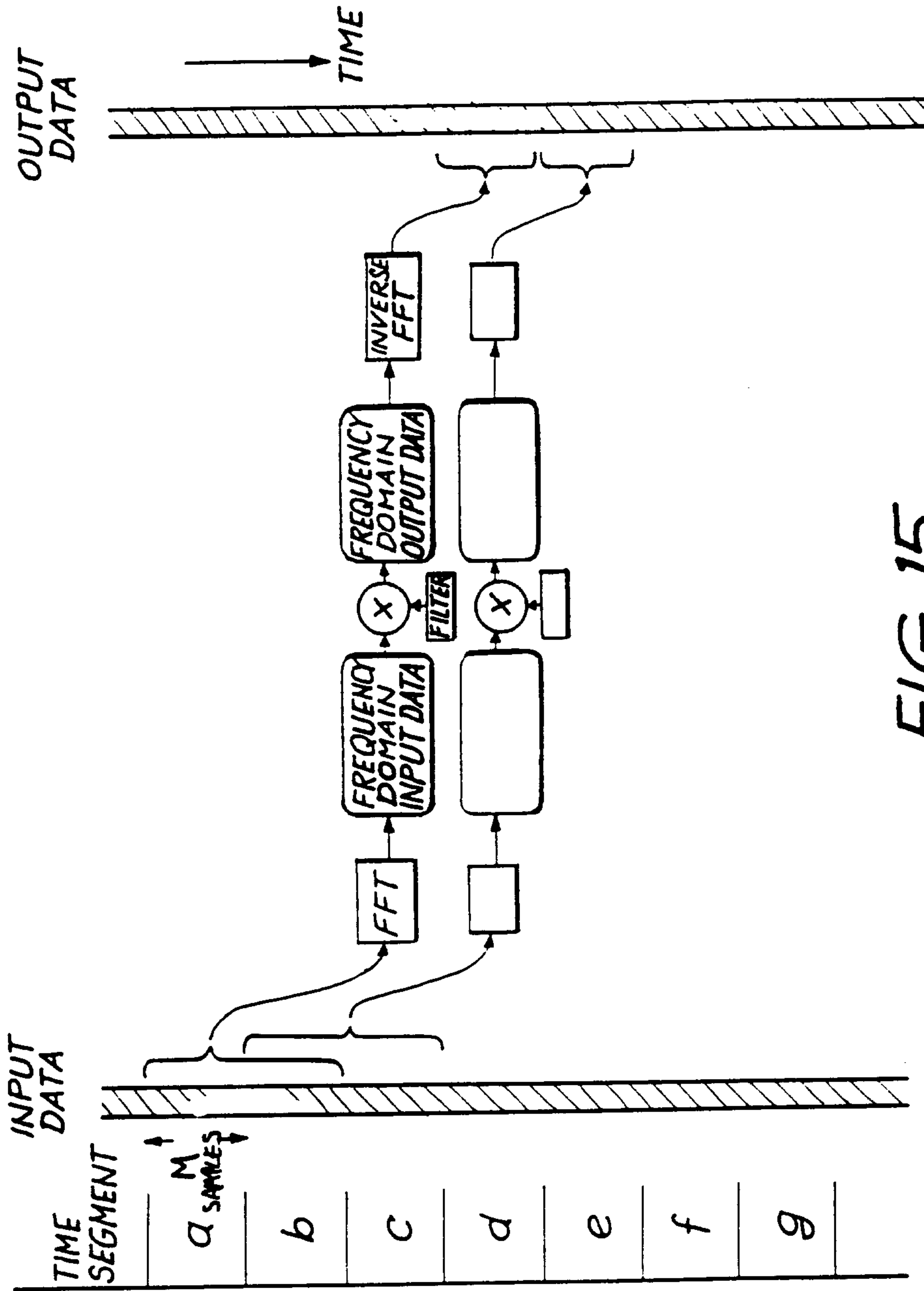


FIG. 15

16/17

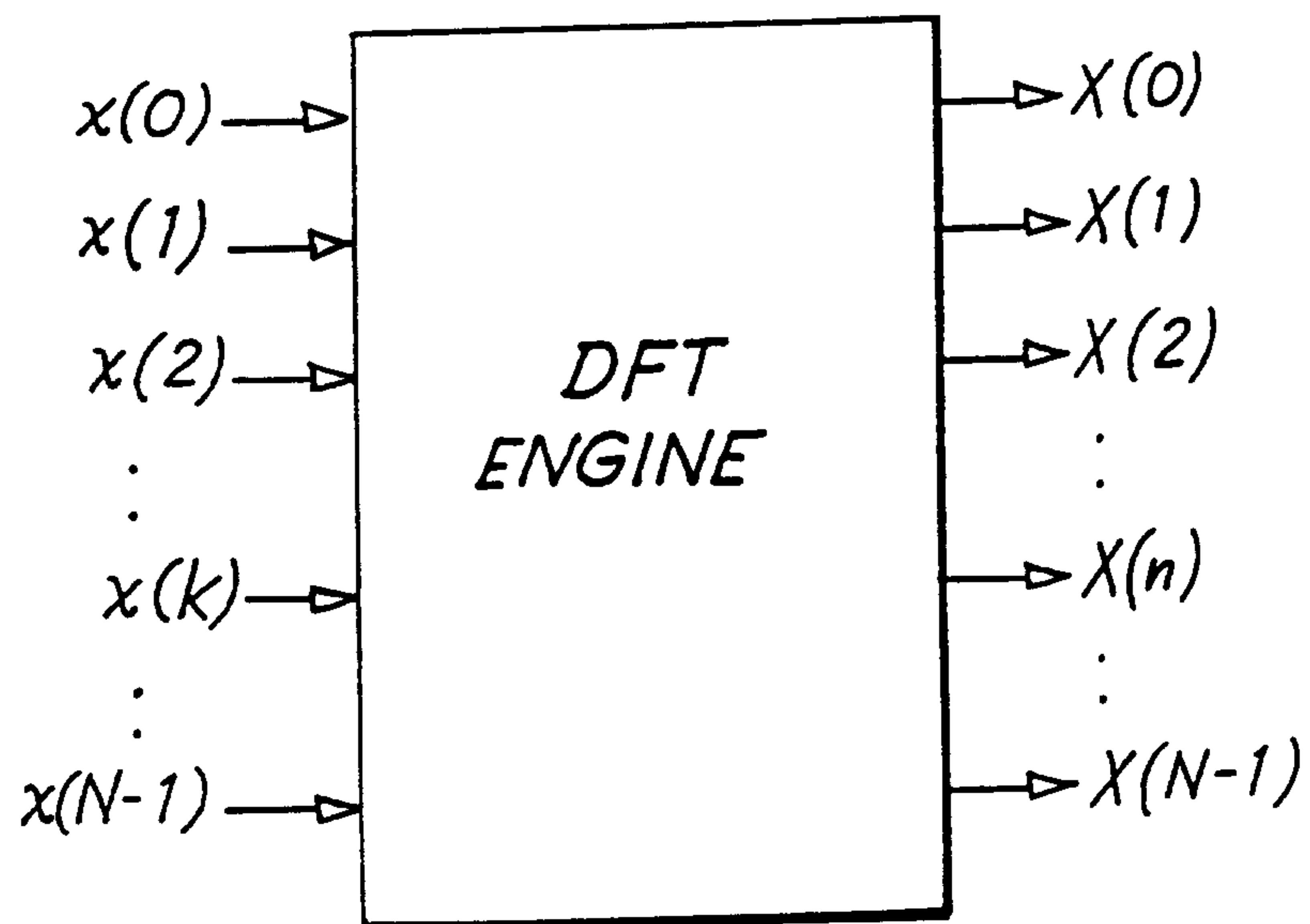


FIG. 16

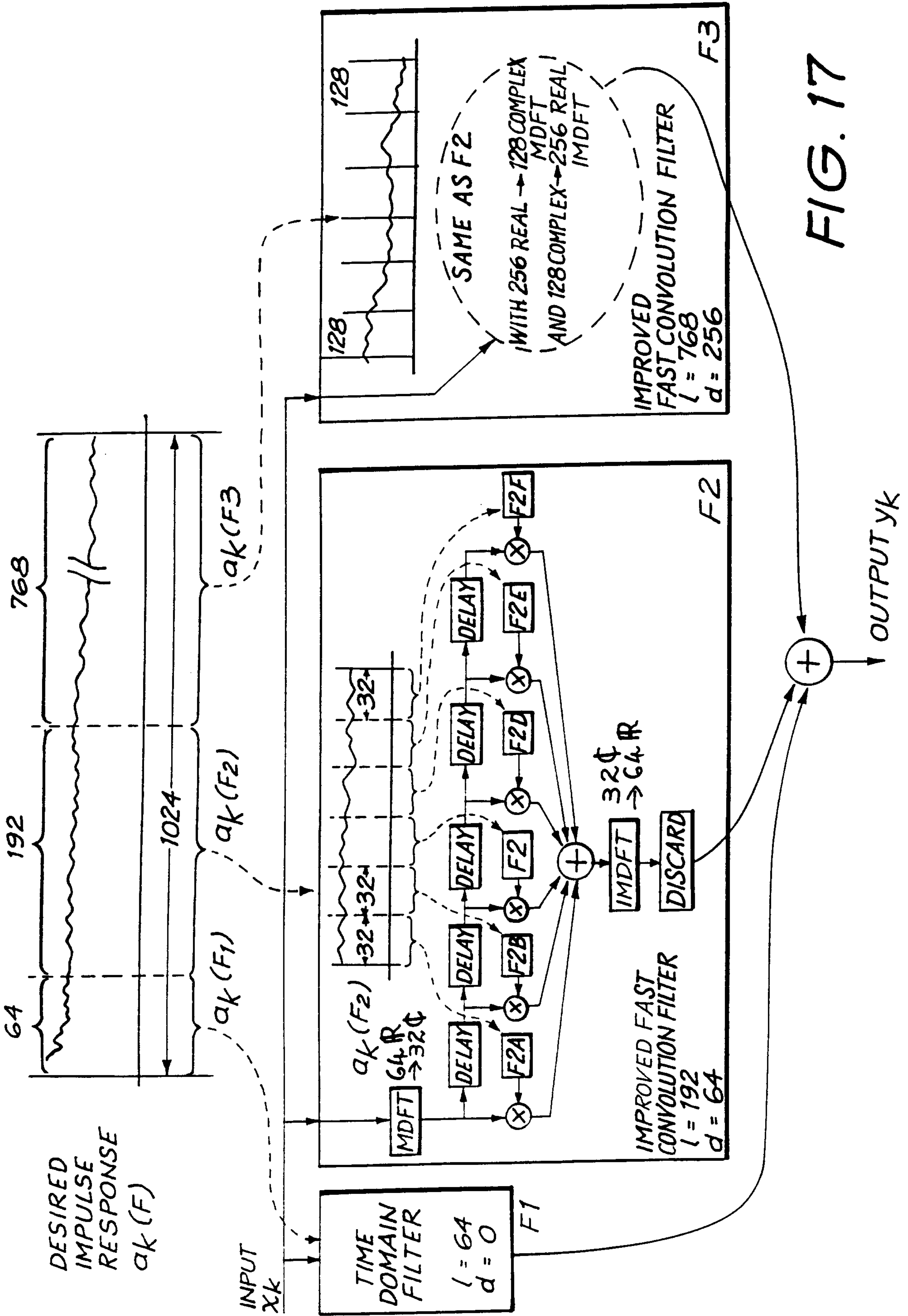


FIG. 17

