APPLYING THE CONCEPT OF A DIGITAL CHARGE COUPLED DEVICE ARITHMETIC UNIT

Chauncey S. Miller
TRW Systems
Redondo Beach, Calif.

Thomas A. Zimmerman
TRW Systems
Redondo Beach, Calif.

ABSTRACT – Recent discussions have highlighted some of the computational potential of CCD implementations in the digital domain, and some note has been taken of the possibility of applying this potential to a few specific situations.[1] This paper explores the concept and application of a digital CCD arithmetic unit consisting of an arithmetic chip which is electronically reprogrammable to allow the chip to perform a variety of functions. The arithmetic chip contains four multipliers, three adder/subtractors, and a variety of selectable delays and signal routing paths. In discussing the configuration of this chip, the importance of technology items such as yield and general producibility as well as practical items such as pin count and package size is considered. The use of this arithmetic unit in any system is totally dependent on the proper interfacing. As part of this concern, the proper timing of input signals (such as data and multiplier coefficients) and output signals is treated, and it is shown that the internal chip timing requirements can generally be made compatible with anticipated system usage. A detailed discussion is given of the arithmetic unit configuration needed to realize a single-pole recursive filter, a nonrecursive digital filter, a single zero digital filter, and two filter sections useful for voice processing systems. This work has been supported in part by the Naval Research Laboratory, Naval Electronics System Command, Contract No. N000-14-74-C-0068.

INTRODUCTION

The use of charge coupled devices in the digital domain has been receiving increased attention. The digital characteristics of high noise immunity and independence from individual device parameters and parameter variations in combination with the high density implementation capabilities of CCD's make an attractive concept.

In this paper we first discuss a few of the basic constraints that are important in any realistic attempt to employ this concept. Next, we give a brief review of the makeup of a reprogrammable arithmetic unit chip that is capable of being electronically reconfigured so as to perform a number of useful functions. The following section describes the nontrivial considerations associated with properly programming such an arithmetic unit and shows how maximum use can be made of the inherent CCD features. The final section illustrates the wide range of capabilities of this arithmetic unit chip by discussing in some detail several of the functional forms the chip can assume.

SOME FUNDAMENTAL CONSIDERATIONS

Before discussing any particular CCD chip configuration, it is worthwhile to reflect on some of the basic limitations and constraints that appear to exist with today's technology. First, CCD's operate not as wideband amplifiers like conventional MOS or bipolar circuits but on a charge transfer principle. To form a digital adder, several charge transfers must occur before the carry from the least significant bit is available as an input to the next most significant bit. In a 16-bit adder, for example, this operation must be repeated 16 times. To compensate for this, pipelining is used to provide an acceptable speed of operation.

The addends are stored in parallel shift registers with the delay increasing from the least significant bit to the most significant bit. This allows additions to be performed at a rate determined by the delay through 1 bit of addition rather than the delay through the total of 16 bits.

The need to operate CCD arithmetic circuits in a pipeline fashion to attain good speed performance imposes a constraint on the functions which can be realized with CCD LSI. Functions which inherently are of a streaming nature (e.g., the FFT) are best suited to the pipeline approach. Pipelining also implies that a number of shift register stages must be included in the implementation of arithmetic functions to equalize the delay between least significant and most significant bits. Fortunately, shift registers are one of the most efficient functions in terms of chip real estate which can be implemented with CCD's.

The maximum producible size of an LSI chip presents a fundamental limitation to the CCD configuration. With present technology, a square chip 150 mils on a side would be considered small to medium size, while a chip 300 mils on a side is considered large.

One of the most important constraints in configurating CCD chips is the limited number of package pins available. The pin limitation constraint is acutely felt when partitioning the digital functions for implementation with CCD's. Every effort is made to make the digital function or functions complete on a single chip. If the function is split between two chips so that two input and two output lines are needed with 16-bit parallel arithmetic, all of the 64 pins available with standard packages would be used for interconnections with none left for power and ground or control signals. One solution to the pin limitation problem is to time multiplex the input/output signals on the same set of pins. The chip timing must be carefully designed from an overall system viewpoint to assure that all of the time multiplexed signals from several chips appear in the correct sequence.

A REPROGRAMMABLE ARITHMETIC UNIT

By incorporating the characteristic of reprogrammability in an arithmetic unit, a large number of useful signal processing functions can be performed with a relatively small number of computational functions. This is particularly true when it is possible to rearrange the interconnection of these few functions at will. One such situation has been studied recently and has resulted in a concept for a digital CCD arithmetic chip. This concept allocates to a single chip some important arithmetic functions such as addition/subtraction and multiplication. In fact, it can be shown that, by including three such adders/subtractors, four multipliers, and a few delay stages and signal routing gates, an impressive list of signal processing functions can be performed.

Figures 1 and 2 show the interconnection of these functions and the proposed layout for such an arithmetic unit. Note that the chip combines the multipliers and adders with the necessary control gates, inverters, and delays. The estimated overall chip dimensions are 350 x 300 mils. The total active area of the chip is considerably less, and it is probable that this design could be produced with reasonable yield. (This size estimate is based on 7.5 micron photolithography, two-level metal interconnections, and a standard two-phase overlapping gate CCD design.) Table 1 lists the pin usage for the chip, and shows that 64 pins are required for the configuration of Figure 1. The details of programming the arithmetic unit are discussed next.

Table 1. Arithmetic Unit Pin Usage

Power and ground	5
Data input	16
Multiplier	16
Output	16
Clock	1
Clock inhibit	1
Sync (÷5)	1
Accumulator clear	2
Control functions (e.g., FFT add or accumulate filter modes)	6
	64

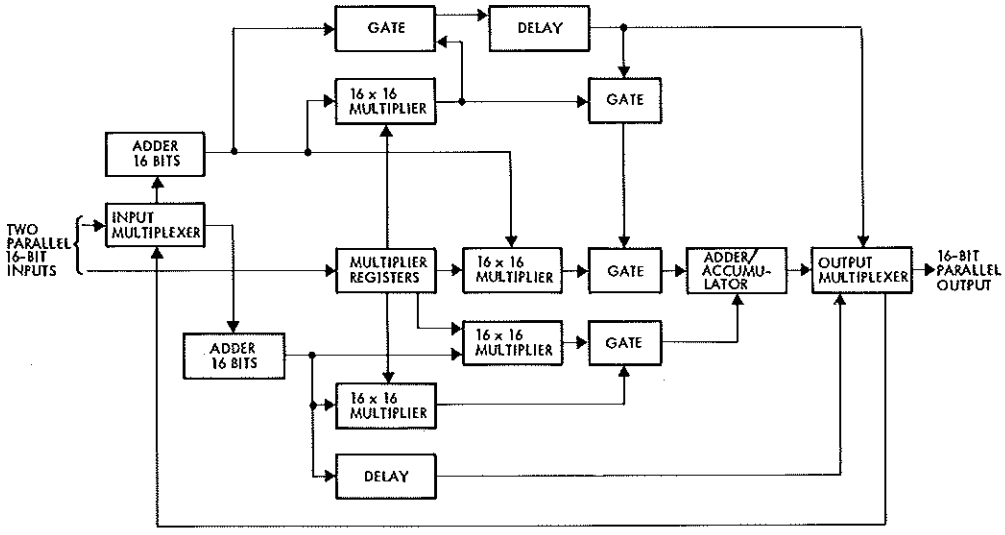


Figure 1. Arithmetic Functions

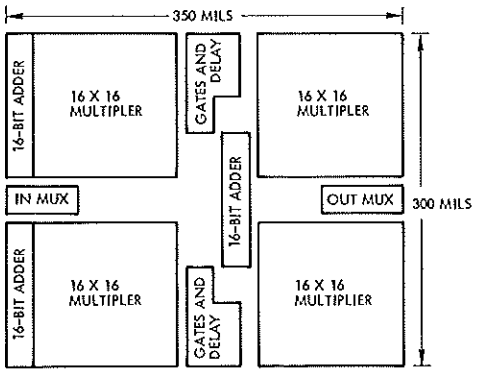


Figure 2. Arithmetic Unit Layout

PROGRAMMING CONSIDERATIONS FOR THE CCD ARITHMETIC UNIT

The utility of the CCD arithmetic unit depends on our ability to reconfigure the chip to perform various functions under program control. Compared to present microprocessors, the CCD arithmetic unit packs five to ten times the computational capability in a single chip. However, due to the fixed layout and the input/output (I/O) limitations imposed by packaging, the CCD arithmetic unit provides less programming flexibility than an array of microprocessor chips with equal computational capacity.

The reconfiguration of the CCD arithmetic unit is accomplished either by activating on-chip controls or by adjusting the inputs so that only the desired function is realized. With reference to Figure 1, the following on-chip functions can be controlled by external binary inputs:

- The two adders at the input can be independently set to either add or subtract
- A delay can be inserted following either the top adder or the top multiplier
- The output adder can be connected either as an adder or as an accumulator
- The internal feedback path from the output back to the input can either be connected or disabled.

It can be seen from Figure 1 that the inputs to the final adder/accumulator are provided by the outputs of two multipliers. This serves to cross-couple the outputs of the two input adder/subtractors which, after multiplication, are summed or differenced, thus completing the complex multiplication required in the FFT kernel or "butterfly" computation. When the CCD arithmetic unit is used for computations

other than the FFT, the cross-coupling can be disabled by making the multiplier coefficient zero. Since there are four multipliers, there are four different (useful) ways of making the interconnections by means of disabling unwanted paths with multiplier coefficients equal to zero.

The programming of the CCD arithmetic unit is strongly influenced by the necessity to time-multiplex the input/output data on the limited number of pins available on a standard package. The charge transfer nature of CCD arithmetic using multiphase clocks requires 2.5 clock intervals for a 1-bit addition or subtraction and five clock intervals for a 1-bit multiplication. It is convenient to arrange the data format so that four clock intervals are used for both the data input/output and the multiplier coefficient input, with the fifth clock interval a "do nothing" state which can be used to synchronize the CCD arithmetic unit with a memory or other external devices. The timing sequences for the various types of computations are indicated in Table 2.

Assuming a 5 MHz clock rate, 1 μsec is required for the five clock intervals in the data format. By the use of pipelining, operations requiring multiplication can be done at a 1 MHz rate. If only addition/subtraction is required, then a 2 MHz throughput can be attained.

It is to be emphasized that these rates are achieved only with pipeline operation. The total delay through the two 16-bit adder/subtractors and the 16x16 multipliers in the signal path amounts to 32 μsec at an assumed 5 MHz clock rate. This factor of 32 represents the difference between pipeline operation and a recursive operation where the output must be available before the next input can be processed. For this reason, computations which are inherently of a streaming nature (e.g., the FFT, correlation, etc.) are particularly well suited to the operation of the CCD arithmetic unit.

INTERLEAVING

When the CCD arithmetic unit is set to operate as a recursive filter, the relatively long propagation delay through the arithmetic functions necessitates the reduction of the sampling rate to below 30 kHz. If only a single filter is being computed, the arithmetic unit is operating very inefficiently because valid inputs appear only in one of the 32 1-μsec intervals constituting the delay through the chip. This situation can be remedied by multiplexing the inputs so that inputs from parallel sources can be interleaved in time. The concept is illustrated in Figure 3 and is basically a conventional time division approach. The delay through the CCD arithmetic unit is divided into 32 1-μsec intervals. The signal from the first source is accepted on the 1st, 33rd, 65th, ..., intervals. The signal from the second source is accepted on the 2nd, 34th, 66th, ..., intervals, and so on. In this way, signals from as many as 32 different sources can be processed simultaneously with one CCD arithmetic unit. By changing the on-chip controls and the multiplier coefficients in synchronism with the inputs, different computations (e.g., single zero filter, double-pole filter, etc.) can be performed on the different inputs.

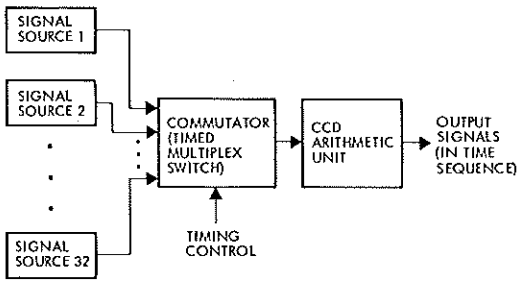


Figure 3. Time Interleaving of Multiple Inputs to the CCD Arithmetic Unit

Another method of obtaining the full computational potential of the CCD arithmetic unit is to use time interleaving to realize a cascade of digital filter sections. This is a very practical procedure because complex digital filters are usually implemented as a cascade of first and second order sections in order to minimize quantization effects. In this case, it is necessary to add an external register as indicated in Figure 4. The external register provides an additional 1 μsec delay needed to apply the output at the n^{th} time interval to the input at the $(n+1)^{th}$ interval. For example, assuming the computation is running continuously and the first section of a digital filter is computed in the first of

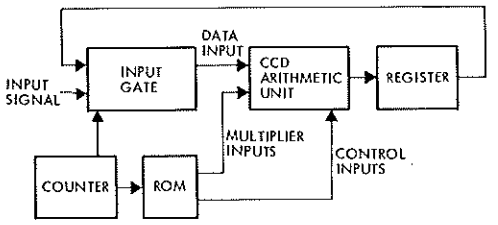


Figure 4. Realization of Cascade by Time Interleaving

32 time intervals, the outputs would appear at the 1st, 33rd, 65th, ..., intervals. The one interval delay provided by the external register allows these values to be applied to the input of a second filter section at the 2nd, 34th, 66th, ..., intervals. Proceeding in this way, a cascade digital filter of 32 different sections can be computed with a single CCD arithmetic unit plus a source of multiplier coefficients and control levels which operates in sync with the arithmetic unit.

From the preceding discussion of processing multiple input signals either from independent sources or as a cascade operation on a single signal, it is obvious that a mixture of the two types of operation is possible. For example, the 32 time intervals could be used to compute eight 4-section filters, four 8-section filters, or any other combination of sources and multiple sections provided the total does not exceed the 32 time intervals available. All that is required is that the timing be carefully controlled and synchronized so that each operation is performed periodically in its assigned time interval.

GENERAL PURPOSE PROGRAMS

Before proceeding with the examples of the different signal processing functions which may be realized with the CCD arithmetic unit, a few comments should be made about operation as a general purpose microprocessor. The primary limitation is the relatively long (32 μsec) propagation delay through the CCD arithmetic unit. This means that programs with many branching points may result in a long execution time. As with the special purpose signal processing functions, programs which can be organized to operate in a pipeline fashion will have the lowest execution time.

The powerful computational capability of the CCD arithmetic unit should not be ignored in general purpose program applications. For example, the parallel adders at the input allow the arithmetic operations and the program instructions to be overlapped in time so that the arithmetic result and its storage location, for example, would appear essentially simultaneously at the output. The adder/subtractor, multiplier, adder/accumulator sequence in the chip layout allows great versatility in computing jump instructions. This combination of powerful arithmetic capability with the unfortunately long propagation delay presents a challenge to the programmer in general purpose applications.

EXAMPLES OF ARITHMETIC UNIT CHIP USAGE

This section discusses several specific configurations of the arithmetic unit chip. A block diagram is shown for each function desired, and this is then compared to the functional realization as achieved by reprogramming the arithmetic chip. The required signal paths and multiplier coefficients are shown where appropriate.

SINGLE ZERO RECURSIVE FILTER

The arithmetic chip can be operated as a single zero recursive digital filter. The block diagram of the single zero recursive filter and the corresponding equations are shown in Figure 5a. A control level is applied to the chip to inhibit the input on the even clock periods (i.e., T_2 and T_4). As shown in Table 2, the data input is accepted during T_1, and a zero value is inserted during T_3. At the multiplier inputs, the sequence is K, 0, 1, and 0. The zero serves to disable the unwanted paths as shown in Figure 5b. The 1 and K multiplier values cause the two single paths to be combined to form the desired output.

TWO SINGLE-POLE RECURSIVE FILTERS

Operation of the arithmetic chip as a pair of single-pole recursive filters is shown in Figure 6a. The organization of the chip allows two independent filters to be computed simultaneously. The two

Table 2. Arithmetic Unit Timing Sequences

Programmable Configuration	Clock Period — Input					Clock Period — Output				
	T_5	T_4	T_3	T_2	T_1	T_5	T_4	T_3	T_2	T_1
FFT										
Data input		Y_2	X_2	Y_1	X_2		CT_2+ST_1	CT_1-ST_1	Y_1+Y_2	X_1+X_2
Multiplier input		-	-	$\sin\theta$	$\cos\theta$		-	$(T_1=X_1-X_2$ $T_2=Y_1-Y_2)$	-	-
Single zero filter										
Data input		-	0	-	X_{in}		-	-	-	Y_{out}
Multiplier input		0	1	0	K		-	-	-	-
Two single-pole filters										
Data input		-	U_{in}	-	X_{in}		-	V_{out}	-	Y_{out}
Multiplier input		0	K_2	0	K_1		-	-	-	-
Single-pole filter with scaled output										
Data input		K_{scale}	0	0	X_{in}		-	$K_{scale}Y_{out}$	-	Y_{out}
Multiplier input		K_{scale}	0	0	K_1		-	-	-	-
Two-pole resonator										
Data input		-	-	-	X_{in}		-	Y_{out}	-	K_{scale}
Multiplier input		K_{scale}	K_1	0	K_2		-	-	-	Y_{out}
Itakura analyzer										
Data input		-	B_{n-1}	-	A_{n-1}		-	B_n	-	A_n
Multiplier input		1	K	K	1		-	-	-	-
Itakura synthesizer (First pass)										
Data input		-	B_m	-	A_{m+1}		-	S_2	-	S_1
Multiplier input		1	K	0	1		-	-	-	-
(Second pass)										
Data input		-	S_2	-	S_1		-	B_{m+1}	-	A_m
Multiplier input		1	0	K	1		-	-	-	-

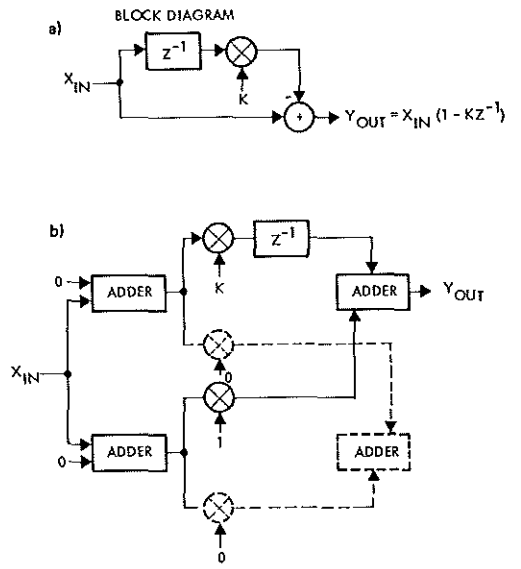


Figure 5. Data Inputs to the CCD Arithmetic Unit

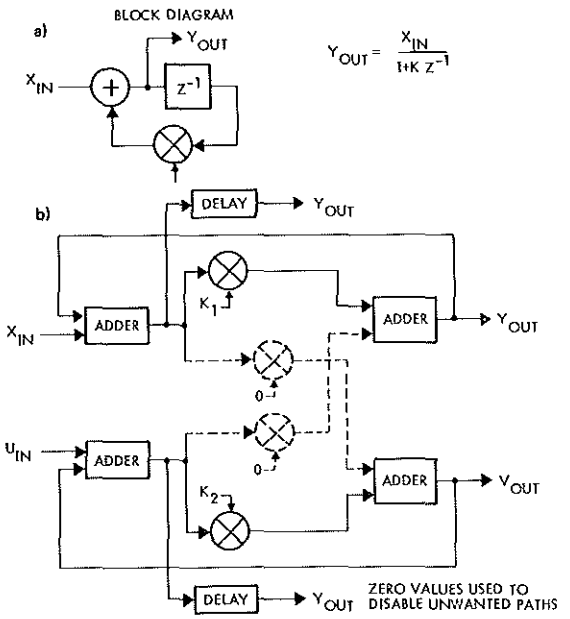


Figure 6. Use of CCD Arithmetic Unit as Parallel Single-Pole Recursive Filters

inputs are applied on the first and third clock pulse intervals. A control level disables the inputs during the even intervals when the outputs are fed back internally to the adder inputs. The multiplier inputs K1 and K2 adjust the time constants of the two filters. Zeros are provided to disable the unwanted paths. The delay required for the recursive filter is provided by the propagation through the multiplier. The subtraction of the output from the input is performed twice by the input adder. After the first subtraction, the difference is applied to the multiplier where it is multiplied by the filter coefficient. After the second subtraction, the difference is delayed one clock period as shown in Figure 6b.

SINGLE-POLE FILTER WITH SCALED OUTPUT

For complex recursive digital filters, it is often desired to provide a weighting or scale factor between adjacent stages. The arithmetic chips can be used to provide a single-pole recursive filter with a scaled output. In this configuration, it is very similar to the one previously described for the two multiplexed single-pole filters with the exception that, in this case, the scaling operation is substituted for the second filtering operation. The operation may be described with the aid of Figure 7. The top half of Figure 7 is identical to Figure 6a and represents a single-pole recursive digital filter. To obtain a scaled output, the same values that were applied to the top adder are also applied to the bottom adder. In this case, the difference formed by the adder is multiplied by the scale factor K_{scale} and is supplied as an output.

LATTICE FILTER – THE ITAKURA ANALYZER SECTION

Linear predictive coding (LPC) is a bandwidth reduction method used for secure voice systems. The linear predictive coder reduces the data rate from a nominal input rate of 60 kilobits to an output rate of less than 5 kilobits. The Itakura procedure is a streaming approach to implement the linear predictive coder with recursive digital filters. The lattice filter

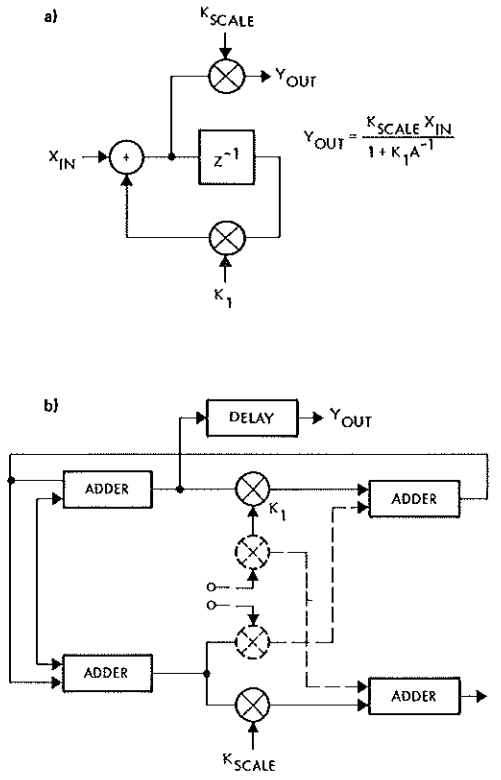


Figure 7. Single-Pole Recursive Filter with Scaled Output

used in the Itakura analyzer is shown in Figure 8a. Inputs to the CCD arithmetic chip are provided during the first and third clock pulses. These correspond to the A and B input in Figure 8a. The multiplier inputs are unity or K, where K is the PARCOR[2] coefficient. In Figure 8b, it is seen that the two outputs, A and B, are obtained by combining the direct (multiplied by unity) and the weighted (multiplied by K) values. Operation of the arithmetic chip in the Itakura analyzer section is particularly straightforward due to the similarity between the complex rotation used in the FFT and the structure of the lattice filter.

THE ITAKURA SYNTHESIZER SECTION

The synthesizer section for the Itakura linear predictive coder employs a lattice filter as shown in Figure 9a. This filter is somewhat more difficult to realize with

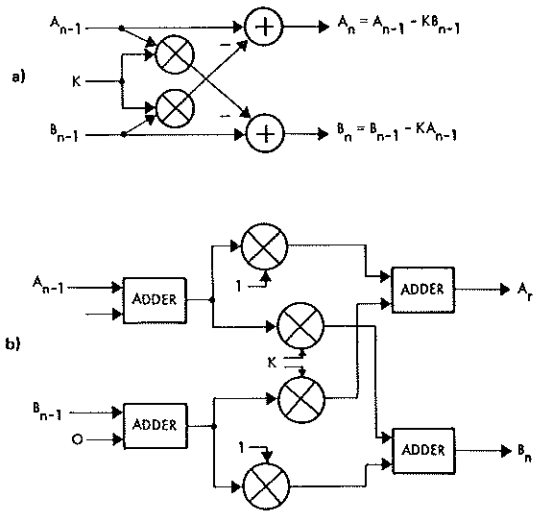
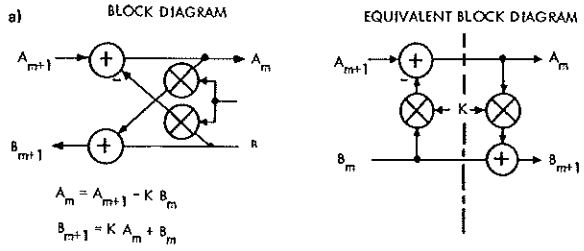


Figure 8. Use of CCD Arithmetic Unit as Lattice Filter (Analyzer)



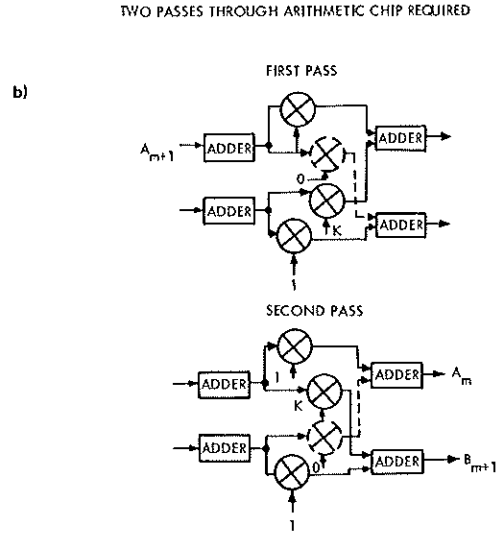


Figure 9. Use of CCD Arithmetic Unit as Lattice Filter (Synthesizer)

the arithmetic chip because two multiplications are required in order to obtain the output. To accomplish this, the lattice filter operation is split into two segments, and two passes through the arithmetic chip are required. Alternatively, two chips can be used – one chip for the first pass and the second chip for the second pass. Referring to Figure 9b, on the first pass, the two inputs A_{m+1} and B_m are applied to the input adder. A_{m+1} is multiplied by unity, and B_m is multiplied by K. The two are added in the top output adder. B_m also appears at the output adder after having been multiplied by unity. In the second pass through the arithmetic chip, the order of the multipliers is changed so that the top input passes through the multiplier, after having been multiplied by unity, and appears at the output as A_m. The lower input is combined with the upper input multiplied by K and appears at the lower adder output as B_{m+1}.

NONRECURSIVE DIGITAL FILTER

As a final example, consider the use of the arithmetic unit in conjunction with a correlator chip, which we described in a previous paper.[1] To use these two chips together, it is necessary to consider the techniques that allow achieving, say, 8-bit accuracy. To achieve 8-bit coefficient accuracy, the signal, s, and the reference, r, consisting of 8-bit words, are partitioned into 4-bit segments

$$s = a + b$$

$$r = c + d$$

with a and c the 4 most significant bits (i.e., ≥ 16) and b and d the 4 least significant bits (i.e., < 16). The multiplication $s \cdot r$ becomes

$$s \cdot r = (a+b)(c+d)$$

$$= ac + bc + ad + bd$$

Thus, four CCD digital correlator chips would be used to form the product comprising a 32-sample signal segment. In addition, three arithmetic unit chips would be needed to sum the four products. Nonrecursive filters with lengths equal to multiples of 32 can be formed by cascading 32-sample sections in a manner similar to that shown in Figure 10.

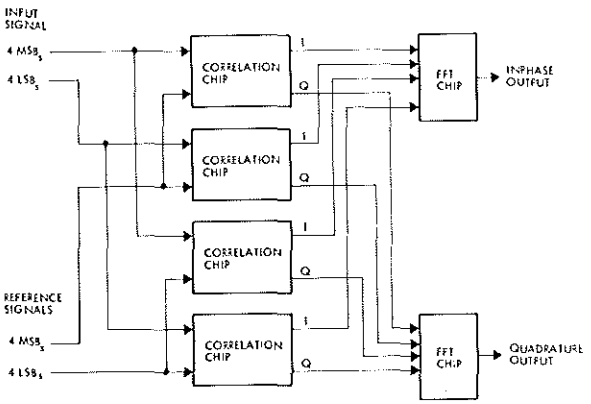


Figure 10. Nonrecursive Digital Filter – 8-Bit Quantization

CONCLUSIONS

The advantages of digital signal processing for most applications are well known and relate to greater accuracy, flexibility, and environmental reliability. However, the growth of digital signal processing has to date been hampered by cost and power limitations even with the use of LSI. The advent of CCD digital processing techniques, as discussed in this paper, has the potential for greatly alleviating these problems and thereby allowing more effective digital signal processing hardware implementations to be achieved in the future. CCD allows high density, low power LSI operation in a full pipelined signal processing architecture. Circuit techniques are now under development towards this objective.

BIBLIOGRAPHY

1. C. S. Miller, T. A. Zimmerman, "The Application of Charge Coupled Devices to Digital Signal Processing," International Communications Conference, San Francisco, California, June 1975.

2. Fumitada Itakura and Shuzo Saito, "On the Optimum Quantization of Feature Parameters in the PARCOR Speech Synthesizer," 1972 IEEE Speech Conference.