D7-33

N86-28282

TDA Progress Report 42-85

January–March 1986

13151

# A Systolic Architecture for the Correlation and Accumulation of Digital Sequences

L. J. Deutsch and C. R. Lahmeyer
Communications Systems Research Section

*This article describes a fully systolic architecture for the implementation of digital sequence correlator/accumulators. These devices consist of a two-dimensional array of processing elements that are conceived for efficient fabrication in Very Large Scale Integrated (VLSI) circuits. A custom VLSI chip that was implemented using these concepts is described. The chip, which contains a four-lag three-level sequence correlator and four bits of accumulation with overflow detection, was designed using the Integrated UNIX-Based Computer Aided Design (CAD) System. Applications of such devices include the synchronization of coded telemetry data, alignment of both real time and non-real time Very Large Baseline Interferometry (VLBI) signals, and the implementation of digital filters and processors of many types.*

## I. Introduction

One of the most common signal processing operations that is used in conjunction with digital signals is correlation. Suppose that $a_i$ and $b_i$ are two sequences of real numbers (where $i$ is an integer $-\infty < i < \infty$). Then the $l$ lag correlation of these two sequences can be defined by

$$C_l\,[a,b](i) = \sum_{k=-l}^{0} a_k\, b_{k+i} \qquad (1)$$

This is not the most general definition of correlation, but it will be sufficient for this article. If the two sequences $a$ and $b$ are identical, then this is called an autocorrelation. If not, it is called a cross-correlation.

The correlation operator, as defined in Eq. (1), is a measure of the amount of agreement between the two sequences as a function of their relative offset. There are, in fact, many important applications that make use of this observation. In Very Large Baseline Interferometry (VLBI) (Ref. 1), for example, received signals from several antenna sites must be correlated in order to determine the relative time differences between the receivers. In Symbol Stream Combining (Ref. 2), the relative timing is not important, but correlators are used to sum the various signals in a maximum likelihood manner in the presence of noise. Telemetry coding synchronization relies on the detection, using correlators, of fixed binary sequences in encoded data (Ref. 3) or of statistical trends produced by certain error correcting codes (Ref. 4). Finally, Eq. (1) is also the basic equation for a Finite Impulse Response (FIR) digital filter (Ref. 5). This means that a digital correlator may be used in filtering applications as well.

There is some confusion in the digital electronics industry as to exactly what constitutes a correlator. Some parts manufacturers produce correlators that only produce the partial products

$$p_j(i) = a_j b_{j+i} \tag{2}$$

A part that also performs the summation in Eq. (1) is sometimes called a correlator/accumulator. In this article, the portion of the system that performs the partial products will sometimes be referred to as a correlator so as not to be confused with the accumulator portion.

Because digital correlator/accumulators in many applications in data acquisition and advanced tracking techniques require large numbers of lags and high speeds, it was decided that an architecture suitable for implementation in Very Large Scale Integrated (VLSI) circuits was needed. This article presents the results of this research effort to date. Indeed, a very efficient architecture has been developed for these devices that takes advantage of the techniques of systolic arrays (Ref. 6). The algorithms are explained in Section II. The implementation of these algorithms in VLSI is described in Section III.

In order to test these algorithms and architectures, a small correlator/accumulator chip was designed, fabricated, and tested. The chip, called SMLCOR, implements a four lag correlator of three-level input sequences. It also contains a fully pipelined set of four bit accumulators with overflow detection circuitry. The chip was fabricated in a 4.0-$\mu$ NMOS technology and was found to be fully functional in the initial fabrication run. The design and testing of this chip are described in Section IV.

Finally, a very large VLSI chip using this architecture is currently in fabrication. This chip comprises a 32-lag complex correlator with phase rotation circuitry and 24 bits of accumulation. It is one of the largest chips to be designed at the Jet Propulsion Laboratory (JPL) to this date. It is described in Section IV. This chip will be used as part of the Advanced Decoding System that is currently under development.

## II. A Systolic Algorithm for Correlators With Accumulation

The basic architecture for systolic correlators as developed by S. Y. Kung (Ref. 7) is now well known. It is shown in Fig. 1. The basic idea is to take the two digital sequences that are to be correlated and pump them into the circuit from the two sides. As they shift through the single bit delay elements (the boxes labeled "D" in the figure) they are brought into various alignments. The multiplication elements (labeled with a cross in the figure) are then used to form the $p$ results. Because this circuit is fully pipelined, it has the potential for very high speed applications. It is also modular and therefore suitable for implementation on VLSI chips.

The one drawback of this circuit is that it produces only the even index correlation coefficients, $p_{2i}$. It is clear that the odd numbered coefficients may be generated using a second circuit with one of the sequences delayed an extra bit time before entering. A slightly different architecture, shown in Fig. 2, could also be used to generate all the coefficients. This architecture is called broadcasting because of the fact that one of the sequences must be broadcast to all the multipliers at once. This has the disadvantage that the broadcast signal must contend with large fan-out and power problems which could result in slower circuit operation.

It was decided that the chips that would be implemented would incorporate both the systolic and broadcast architectures and an external mode switch for selection between them. Some high-speed applications, such as VLBI, do not require all the coefficients and could take advantage of the systolic architecture. Other applications, such as coding synchronization, do require all the coefficients but do not have to run at the very high speeds. These can use the broadcast system. Finally, those applications that require both high speed and high spatial resolution can use two chips to generate the entire set of coefficients.

The architecture for the pipelined accumulators is shown in Fig. 3. The basic idea here is to take the output of a correlator cell (i.e., one of the $p$'s) and pass it to a first stage that comprises a conventional accumulator. This first stage implements just enough bits of accumulation to generate a single bit output (called the carry) to the rest of the circuit. It can be thought of as a conversion unit that takes an input signal that may be many bits in width and scales it in time to a one-bit signal. It also adds a bias to the signal so that negative numbers in the correlation can be accumulated as postive numbers only.

Following the first stage, the accumulator consists of identical cells that are each single-bit adders. At any time, the result of the accumulation appears in the delay elements with the least significant bit in the uppermost element.

In order to speed up the operation of the accumulators, an additional delay element is added between the stages of accumulations. The results as contained in the delay elements are now skewed in time. This must be taken into account in the use of these circuits in actual applications.

Because of the pipelined nature of the accumulators, the sums should be read out using the following procedure. First, the two data inputs to the correlator should be forced to zero for a few clock times in order to let the partial sums trickle down the pipe. Then the reset signal should be applied. This will zero the registers in the accumulator while allowing the sums to be shifted out to the right. After all the sums have been read out (this takes four clock cycles in SMLCOR), the reset signal can be removed and new data can be sent to the correlators.

In the next section, the design of a small correlator/accumulator using these concepts is examined in detail.

## III. The SMLCOR 4–Lag Correlator/ Accumulator

In order to test some of the above concepts, a small correlator/accumulator circuit was implemented on a VLSI chip. The chip, called SMLCOR (for "Small Correlator"), was designed using the Integrated UNIX-Based Computer Aided Design (CAD) system (Ref. 8). The sequence inputs to SMLCOR are three level. This means that they can assume the values in the set {-1, 0, 1} only. This was done for two reasons. First, most applications that are being considered for these chips (such as VLBI and code synchronization) do not require any more accuracy than this. Second, three-level real multipliers are very efficient to implement since the set {-1, 0, 1} is closed under multiplication. It takes two bits to represent the three levels. This is a bit inefficient since four levels could be represented with this number of bits as well. However, the added complexity needed to implement four-level multipliers would more than outweigh this waste.

The system of representation for the three-level numbers is as follows:

| Number | Representation |
|---|---|
| -1 | 10 |
| 0 | 00 |
| 1 | 01 |

The representation 11 is not allowed, and it can be used for detecting certain failure conditions in the operation of chips using this sytem.

A block diagram of the correlator portion of SMLCOR is shown in Fig. 4. Notice the addition of the select gates. These are used to determine whether the circuit is in systolic or broadcast mode according to the input signal mode. The multipliers are implemented as Programmable Logic Arrays (PLAs) (Ref. 9). They were generated automatically from a set of Boolean equations in a matter of seconds, thus reducing the design time of SMLCOR considerably. Although PLAs are, in general, not an efficient method of implementing fast logic, in this case, the multipliers are small enough that there would be little difference in performance between the PLA and a full custom design.

The implementation of the accumulators is represented by Fig. 5. Some logic (notably the readout logic) has been omitted for clarity. The first stage accumlation is built from two blocks. The first block, called data converter, takes a three-level input and produces an output according to the following truth table:

| Input | Output |
|---|---|
| 10 | 00 |
| 00 | 10 |
| 01 | 11 |

In this way, the number of ones that is output reflects the magnitude of the correlation coefficient. This serves the purpose of biasing the number system to take care of negative numbers. These ones need only to be summed in time to produce the desired (biased) result.

The second portion of the first stage accumulation is a full adder and delay that constitute a 0, 1, 2 accumulator. The carry from this circuit is then fed to the remainder of the array.

The rest of the accumulator is identical to that in Fig. 3 with the addition of the overflow detection circuit at the bottom of the figure. This is used to detect when the capacity of the accumulator has been exceeded. Four bits of accumulation were implemented in SMLCOR.

There is also a method of reading out the results that is not shown in the figures. A reset signal is used to zero the appropriate registers, and the next four clock times are then used to shift the data out of the accumulator delay elements.

A layout of SMLCOR is shown in Fig. 6. SMLCOR was fabricated using a 4-$\mu$ NMOS technology. It was tested using the Digital Microcircuit Functionality Tester (DMFT) (Ref. 10) and found to be fully functional with no additional fabrication iterations required.

## IV. Ongoing Work: BIGCOR

Since the results obtained from SMLCOR were very encouraging, the decision was made to implement a full version with the new architecture. A much larger correlator/accumulator chip, called BIGCOR, was designed using the same techniques as in SMLCOR. This was particularly easy to accomplish as the basic cells from SMLCOR could be used unaltered in the new design.

In order to make BIGCOR useful to a large variety of applications, it has been designed with 32 lags and 24 bits of accumulation. In addition, BIGCOR can rotate the results of the correlation in the complex plane by using a phase rotation circuit. The same phase correction number is applied to all the lags simultaneously using a broadcast technique. The complex results are then accumulated using two pipeline accumulators for each lag.

Although BIGCOR is conceptually only a little more complex than SMLCOR, it is certainly one of the largest VLSI chips yet designed at JPL. It contains over 60,000 transistors. The design of BIGCOR has been completed. In addition, the design has been checked and simulated. It is now in fabrication. BIGCOR is being implemented in a 3-$\mu$ minimum feature size NMOS technology. It should run at 8 MHz in the systolic mode.

## V. Conclusions

An efficient architecture has been developed for the VLSI implementation of systolic correlators and accumulators. The architecture is easily extensible in both the number of lags and the number of bits of accumulation. Furthermore, the concepts have been put to practice in the implementation of SMLCOR, a fully functional 4-lag correlator/accumulator chip. The cells that were developed for SMLCOR may be used in the design of similar chips of varying parameters.

The design of a large enough correlator/accumulator chip to be practical in a large number of digital signal processing applications has also been demonstrated. The chip, BIGCOR, is now in fabrication.

# References


1. Yuen, J. (ed.), *Deep Space Telecommunications Systems Engineering*, Jet Propulsion Laboratory, Pasadena, Calif., pp. 135–138, July 1982.

2. Hurd, W. J., Reder, L. J., and Russell, M. D., "Symbol-Stream Combiner: Description and Demonstration Plans," *TDA Progress Report 42-78*, Jet Propulsion Laboratory, Pasadena, Calif., pp. 115–121, April–June 1984.

3. Deutsch, L. J., "Reed-Solomon Code Synchronization Revisited," *TDA Progress Report 42-80*, Jet Propulsion Laboratory, Pasadena, Calif., pp. 91–96, October–December 1984.

4. Lordem, G., McEliece, R., and Swanson, L., "Node Synchronization for the Viterbi Decoder," *TDA Progress Report 42-73*, Jet Propulsion Laboratory, Pasadena, Calif., pp. 22–35, January–March 1983.

5. Hamming, R. W., *Digital Filters*, Englewood Cliffs, N.J.: Prentice-Hall Inc., pp. 180–195, 1977.

6. Kung, H. T., "Systolic Arrays (for VLSI)," *Sparse Matrix Proc.*, 1978, SIAM, pp. 256–282, 1979.

7. Kung, S. Y., "VLSI Signal Processing: From Transversal Filtering to Concurrent Array Processing," *VLSI in Modern Signal Processing*, Englewood Cliffs, N.J.: Prentice-Hall Inc., pp. 127–152, 1985.

8. Deutsch, L. J., "An Integrated UNIX-Based CAD System for the Design and Testing of Custom VLSI Chips," *TDA Progress Report 42-81*, Jet Propulsion Laboratory, Pasadena, Calif., pp. 51–62, January–March 1985.

9. Mead, C. and Conway, L., *Introduction to VLSI Systems*, Menlo Park, California: Addison-Wesley Publishing Company, pp. 115–127, 1980.

10. Olson, E. M., and Deutsch, L. J., "A System for the Functional Testing and Simulation of Custom and Semicustom VLSI Chips," *TDA Progress Report*, to appear.

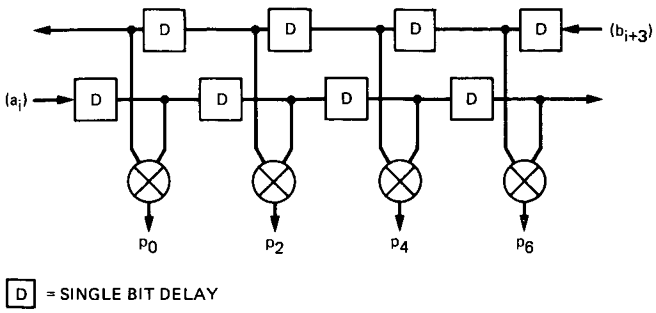


**Fig. 1. Systolic architecture for a digital correlator**

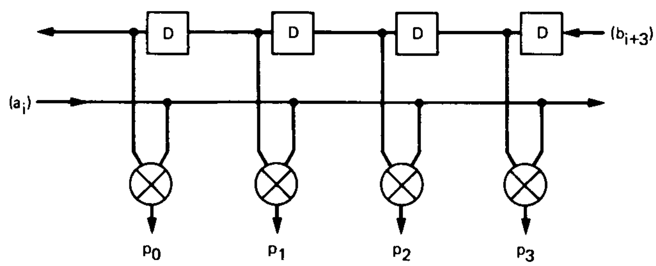


**Fig. 2. Broadcast architecture for a digital correlator**

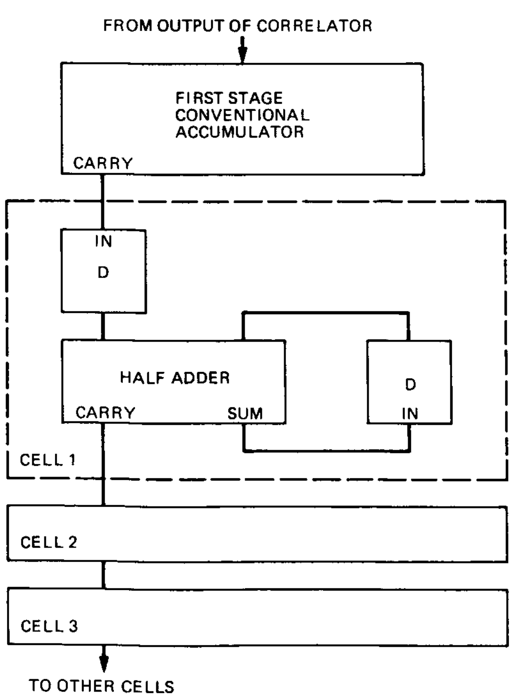


**Fig. 3. Design of the pipelined accumulators**

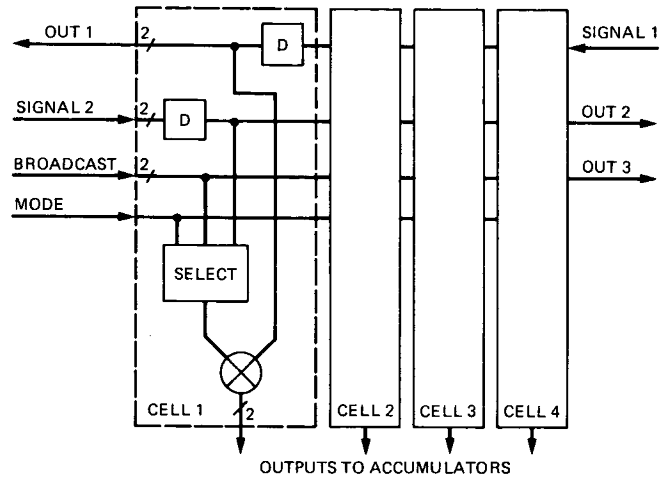


**Fig. 4. Correlator design in SMLCOR**

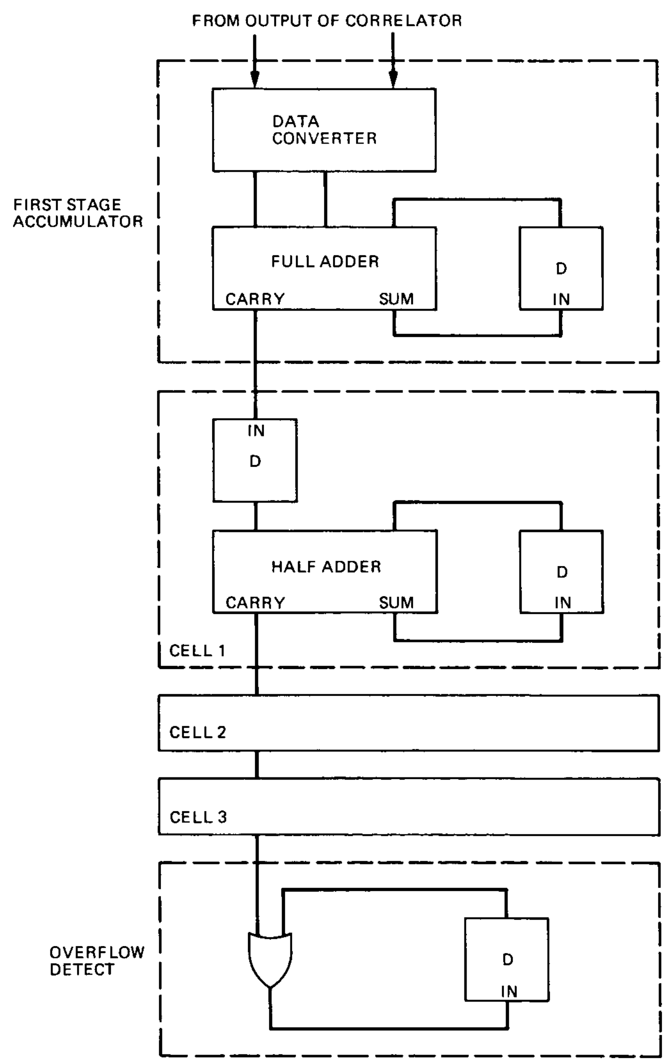


**Fig. 5. The pipelined accumulators in SMLCOR**

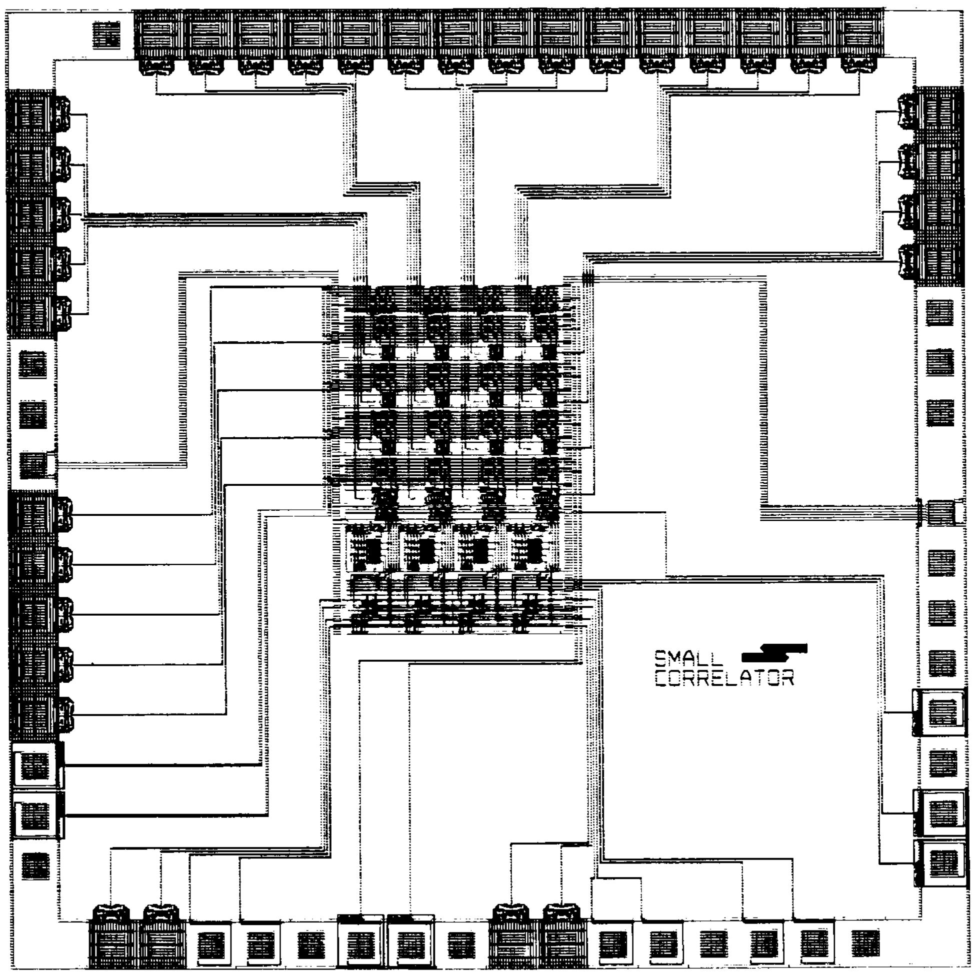


**Fig. 6. Layout of the SMLCOR correlator chip**