

# The Matched Delay Technique: Theory and Practical Issues

Wentai Liu, Mark Clements, Ralph Cavin III  
Department of Electrical and Computer Engineering  
North Carolina State University  
Raleigh, NC 27695-7911  
e-mail: wentai@mcnc.org  
(919)-515-7347(ph)  
(919)-515-55239(fax)

## 1 Introduction

Because of the advent of very high speed, optical fiber networks, and powerful microprocessors, there is a strong trend toward using a network of processors, rather than a single very expensive processor, for high performance computing. A fundamentally important component of this type of computing architecture is a low cost, high bandwidth network interface for each of the processors. GaAs or ECL technologies currently are used to implement high performance network interfaces. These circuit families have high cost and power consumption relative to CMOS. Also, CMOS has a much better ability to integrate higher level network functionality with the interface circuitry. So far, however, CMOS technologies have not been able to support network bit rates above 100 - 200 megabits per second. Specifically, the serializer/deserializer function is the performance bottleneck in a CMOS implementation of a network interface. Research into new circuit techniques are needed if this bottleneck is to be eliminated and CMOS network interfaces that operate in the gigabit per second range are to be built.

In the past few years, we have developed an innovative circuit technique called the *matched delay* structure. This technique is derived from *wave pipelining*, a timing methodology intensively studied by our group [6, 7]. Using the matched delay technique, we have designed digital sampler and generator chips that operate in the low gigabit per second range. The matched delay sampler and generator perform complementary, or dual functions. The sampler takes a serial digital input data stream, and produces parallel binary words that represent sequential digital samples of the input stream. The generator performs the opposite function. It takes parallel input words that represent the sequential samples of a serial data stream, and creates that stream. The sampler and generator, respectively, are capable of performing the deserializer and serializer functions in a high performance CMOS network interface.

The key advantage of both the sampler and generator is that they are based on a circuit structure which produces a timing resolution that is equal to the difference between two propagation delay values. As a result, the resolution is not limited by the minimum inherent gate delay. This approach allows a particular circuit technology to reach much higher levels of timing resolution than could previously be obtained from that technology. In our experiments, we have found that a resolution of 25 ps is achievable with 1.2  $\mu\text{m}$  CMOS, a relatively inexpensive technology.

There are many other applications which require very fine timing resolution and could potentially benefit from the matched delay technique as well. These include phase-locked and delay-locked loops, data and clock recovery, A/D conversion, and test and measurement equipment such

as network analyzers, BER testers, logic analyzers, VLSI testers, and time interval digitizers.

In the remainder of this white paper, we first describe the structure and operation of the matched delay sampler and generator, and then briefly detail some of our designs which have used them. Next, we discuss some practical issues, such as resolution and accuracy vs. bandwidth, noise isolation, data parallelization, and clock rate, that must be faced when the sampler and generator are to be used in real systems. We have found solutions to most of the practical problems that we have encountered, but further work is needed to refine and expand the usefulness of the sampler and generator. Finally, we propose some basic questions that we believe should guide future research of the matched delay technique.

## 2 The Matched Delay Technique

The fundamental concept of the matched delay technique is that the timing resolution is determined by the difference between two propagation delay values. Both the sampler and generator have a structure based on a pair of tapped delay lines, with a latch at each corresponding pair of taps. The data propagates down one delay chain, and a clock propagates down the other chain, clocking the latch at each stage. The structure can be viewed as consisting of a series of stages, where each stage contains a data delay, a clock delay, and a latch. The difference between the data and clock propagation delays determines the timing interval between adjacent stages. In the sampler, the input of the latch is driven by the data delay chain tap, and in the generator, the latch output drives the data tap. The details of the sampler and generator are described below.

### 2.1 The Matched Delay Sampler

Figure 1 shows a block diagram of the basic matched delay sampler. The sampler has a number of stages, and each stage consists of a data latch, a data delay,  $D_d$ , and a clock delay,  $D_c$ . The data and clock are simultaneously propagated through their respective delay chains. The input to the latch at each stage is driven by the data delay chain tap, and the latch is clocked by the clock delay chain tap, and so each latch samples the local data with the local clock. The effective time interval,  $\Delta t$ , between the samples at adjacent stages is equal to the difference between the data and clock delays:

$$\Delta t = |D_c - D_d|$$

If the sampler has  $N$  stages, then a single clock pulse sent into the clock delay chain will acquire  $N$  consecutive data samples. If a larger number of samples is required, then the sampler must be repetitively clocked, and the data from each clock stored. If the data samples acquired by a repetitive clock are to be consecutive, the clock must have period  $T = N\Delta t$ .

Ideally, we would like to save  $N$  consecutive data bits into an output register every  $T$  seconds. However, the sampler with a repetitive clock does not present all  $N$  consecutive bits at its latch outputs simultaneously. There are two problems that must be solved. First, the delay of the clock at each sampler stage skews the valid times of adjacent latch outputs by  $D_c$ . The skew across a number of stages prevents their outputs from being latched simultaneously. Second, there are several clocks propagating through the sampler at once, and they divide the sampler into *clock sections*. The data samples produced by adjacent clock sections are not necessarily consecutive. In general, only the data samples within a section are consecutive.

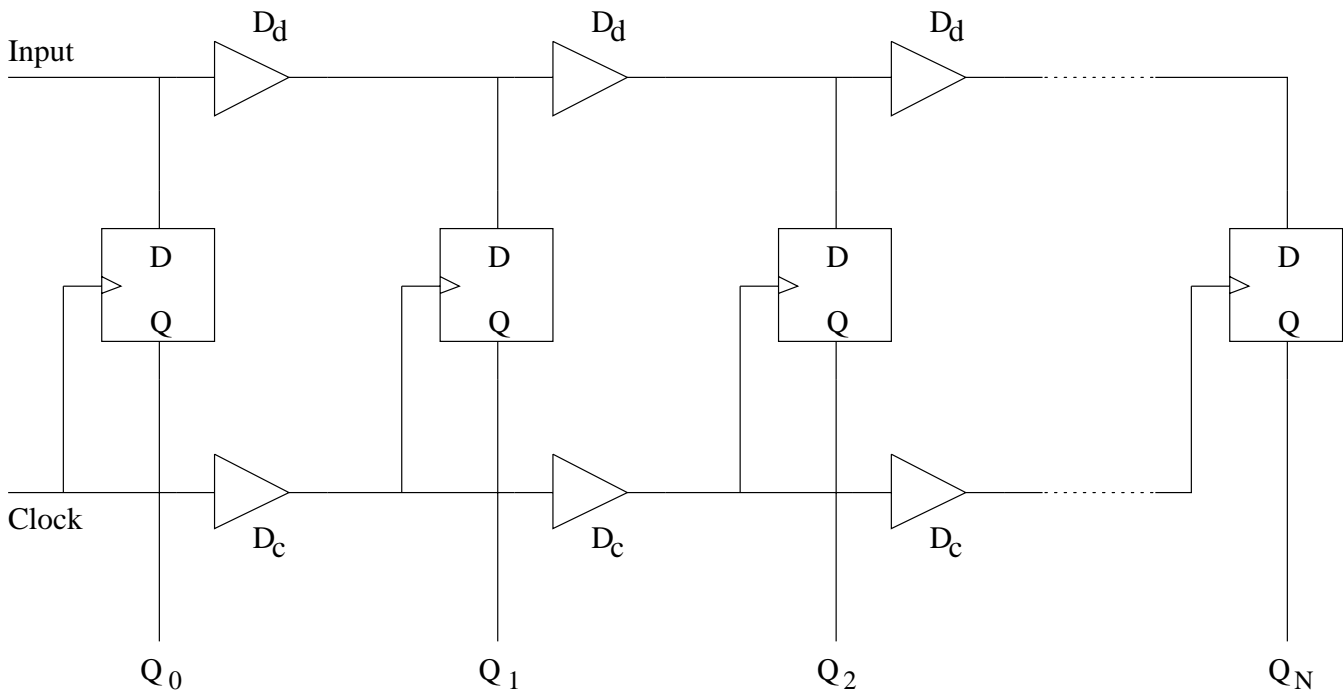


Figure 1: The Basic Matched Delay Sampler

Because the clock signal is delayed at each latch, the data valid times of adjacent latches are skewed by  $D_c$ . By delaying the earlier latch outputs by the appropriate amount, we can re-align the samples to the extent necessary to successfully latch them into the output register. This re-alignment can be achieved by adding a set of deskew latches to delay the upstream half of the channels in each clock section by  $T/2$ . These latches are driven by a clock of the same period,  $T$ , as the sampling clock, but approximately 180 degrees out of phase. Thus, the skew of the deskew latch outputs relative to the downstream channels is mostly canceled. If the phase of the deskew latch clock is optimal, then the data valid window for the output register is determined by the skew among the downstream channels of the section. Figure 2 shows a complete continuous matched delay sampler, and the deskew latches can be seen directly below the clock delay chain.

In the typical sampler,  $\Delta t$  is significantly smaller than  $D_c$ , and so there are several sample clock edges propagating through the sampler at once. Each clock traverses one section during each period  $T$ . The data stored in each section during a particular period is not consecutive with the data of the other sections. There is, however, a constant relationship among the sections that is determined by  $D_d$  and  $D_c$ . The samples from the various sections can be synchronized by delaying the leading sections with clocked FIFO's of the appropriate number of stages. The FIFO stages are driven by a single clock of period  $T$ . In the example sampler, there are four clock cycles present at once. As can be seen in Figure 2, the first, second, third, and fourth sections are delayed by three, two, one, and zero clocks, respectively.

There are two issues that must be considered in the design of a particular matched delay sampler structure. First, once a desired value for the sampling interval,  $\Delta t$ , has been chosen, there is a practical constraint on the nominal values of  $D_d$  and  $D_c$ . If  $D_c$  is not an integral multiple of  $\Delta t$ , then the clocks in the sampler will be out of phase, and the synchronization FIFO for each section will have to be clocked by a separate clock. The complexity of the system will be increased significantly. Second, the input to output latency of the sampler increases with the ratio  $D_c / \Delta t$ .

## 2.2 The Matched Delay Generator

The matched delay generator performs the dual function of the sampler. It has the same topology as the sampler, but the information flows in the opposite direction. The resolution is determined by the propagation delay difference, just as in the sampler. Figure 3 shows the basic generator structure. There are a few differences in the functions of the circuit elements, as compared to the sampler. First, the data delay element is actually an XOR gate, and second, the latch is a toggle latch. One input of the XOR gate is driven by the output or the upstream XOR gate, and the other input is driven by the output of the toggle latch. The XOR gate is used because it will allow both rising and falling edges from upstream to pass through, and will allow the local latch to create a new edge at the time it is clocked. A change at the local input creates an edge, and therefore a toggle latch is used. The clocking works in exactly the same manner as in the sampler.

The toggle latches must be driven by data that indicates the placement of output edges, but we would like to input "sample-like" data, so that the generator performs the dual function of the sampler. Therefore, the input data must be encoded. This encoding is performed by XOR gates that compare each input bit with the adjacent earlier bit. The first bit of each cycle is compared to the last bit of the previous cycle. Anywhere that the two bits differ, the XOR will produce a one and cause the latch to toggle when it is clocked. An edge will be produced in the output data stream. The encoding logic can be seen in Figure 4.

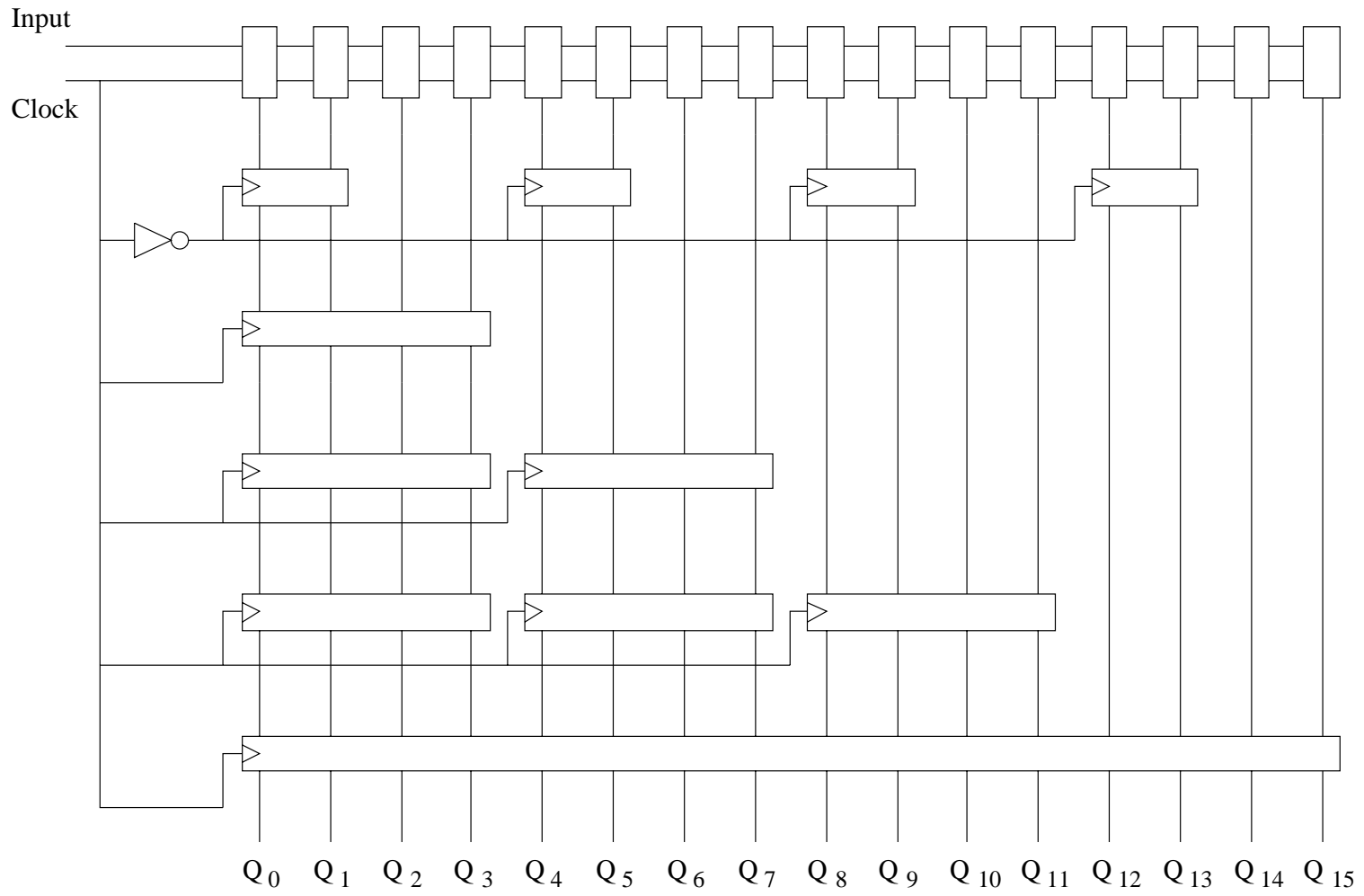


Figure 2: The Continuous Sampler

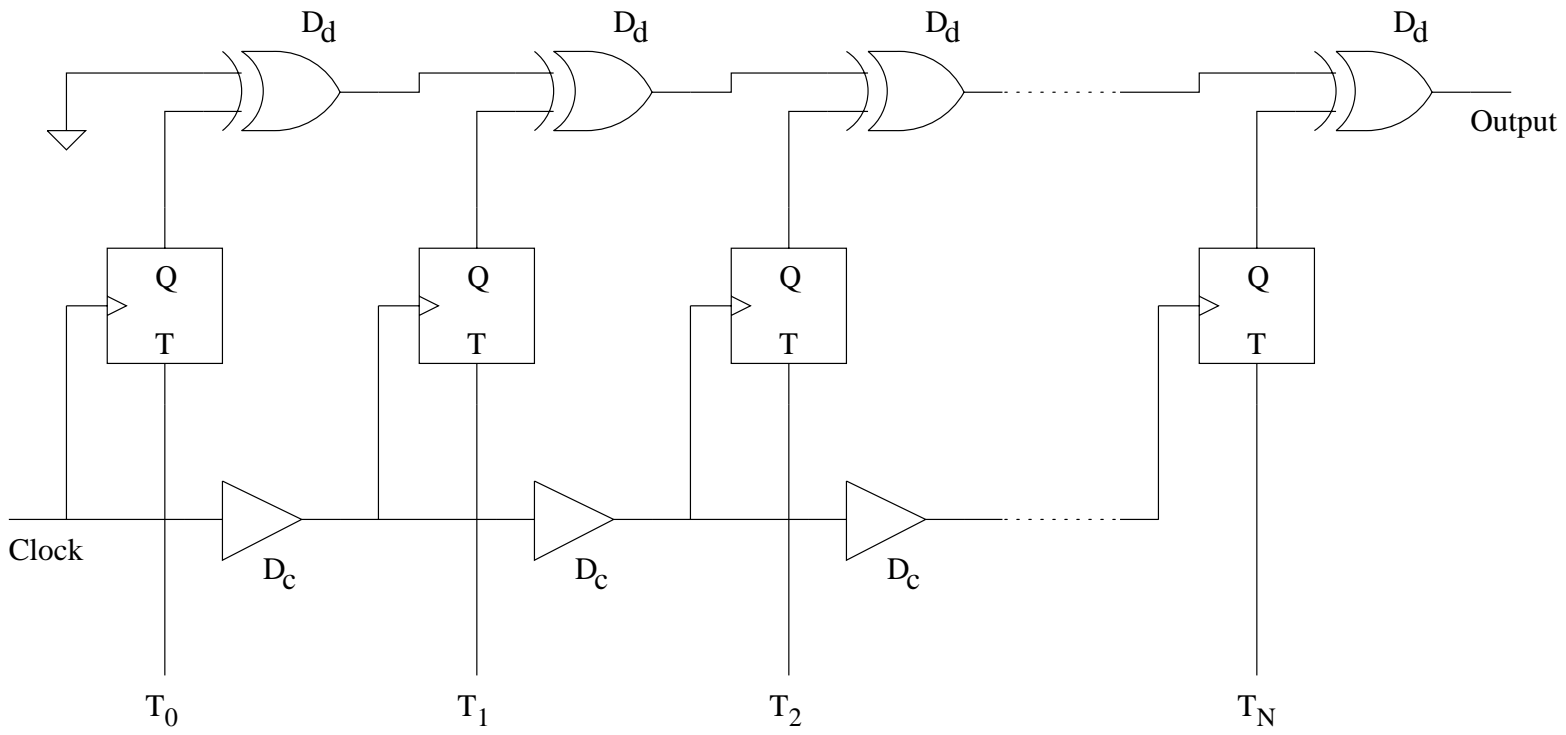


Figure 3: The Basic Matched Delay Generator



In order to produce consecutive data, the generator needs the same additional parts as the sampler. These parts ensure that the encoded edge data reaches each toggle latch at the correct time to be clocked by the same edge through the entire generator. The FIFO's compensate for the fact that multiple clocks are present simultaneously, and the out of phase latches compensate for the clock skew created by the clock delay chain. Figure 5 shows an entire continuous generating structure based on the same data and clock delay values as the example sampler described above. The consecutive data additions to the generator are a reflected version of the additions to the sampler. Note that the delay value choices for the generator are constrained in the same way as for the sampler.

### 3 Previous Results

In this section, we will describe four designs that use the matched delay technique and have already been completed. The first two of these have been fabricated in 1.2  $\mu\text{m}$  CMOS. The third and fourth designs will be fabricated in the next few weeks.

The first design is a test chip for the basic sampler structure with 64 stages [1]. It is designed to have an adjustable timing resolution. This sampler was found to be able to achieve a 25 ps timing resolution and a 1 ns minimum pulse width.

The second project uses a sampler as part of a clock shaping circuit that restores the duty cycle of a distorted clock signal [3]. It also synchronizes the clock to a slower external clock. It operates at frequencies up to 450 MHz and restores the duty cycle to within 45 to 55 percent.

The third project uses a matched delay sampler, with the continuous sampling additions, as a major component in the design of a data and clock recovery circuit for 622 Mbps (OC-12) SONET applications [2]. The design is targeted for 1.2  $\mu\text{m}$  CMOS. The sampler has 64 stages and is clocked at 156 MHz. Our simulations confirm that the circuit reliably achieves 100 ps sampling resolution of the input data, and successfully recovers the clock and data at 622 Mbps.

The final design incorporates a generator that can produce an arbitrary data stream with an edge placement resolution of 100 ps. This chip includes a 512 sample data memory and extensive delay-locked loop based compensation mechanisms to maintain timing accuracy. The generator has 64 stages and is driven by a 156 MHz clock. This chip also will be implemented in 1.2  $\mu\text{m}$  CMOS.

### 4 Practical Issues

In this section, we discuss some practical considerations that affect most potential applications of the matched delay technique. There are several inherent characteristics of the technique which are the sources of its advantages, but which might also present some new problems to the system designer. These are issues that must be carefully explored and characterized during future work.

#### 4.1 Resolution and Accuracy Considerations

A very important characteristic that has a significant influence on the ultimate usefulness of the matched delay technique is timing accuracy. In theory, the difference between the clock and data delays can be made arbitrarily small, and so the timing resolution of the sampler or generator can be increased to an arbitrary level. However, the accuracy of the resolution is limited by the accuracy of the delays of the clock and data delay elements. If the nominal delay difference is set to



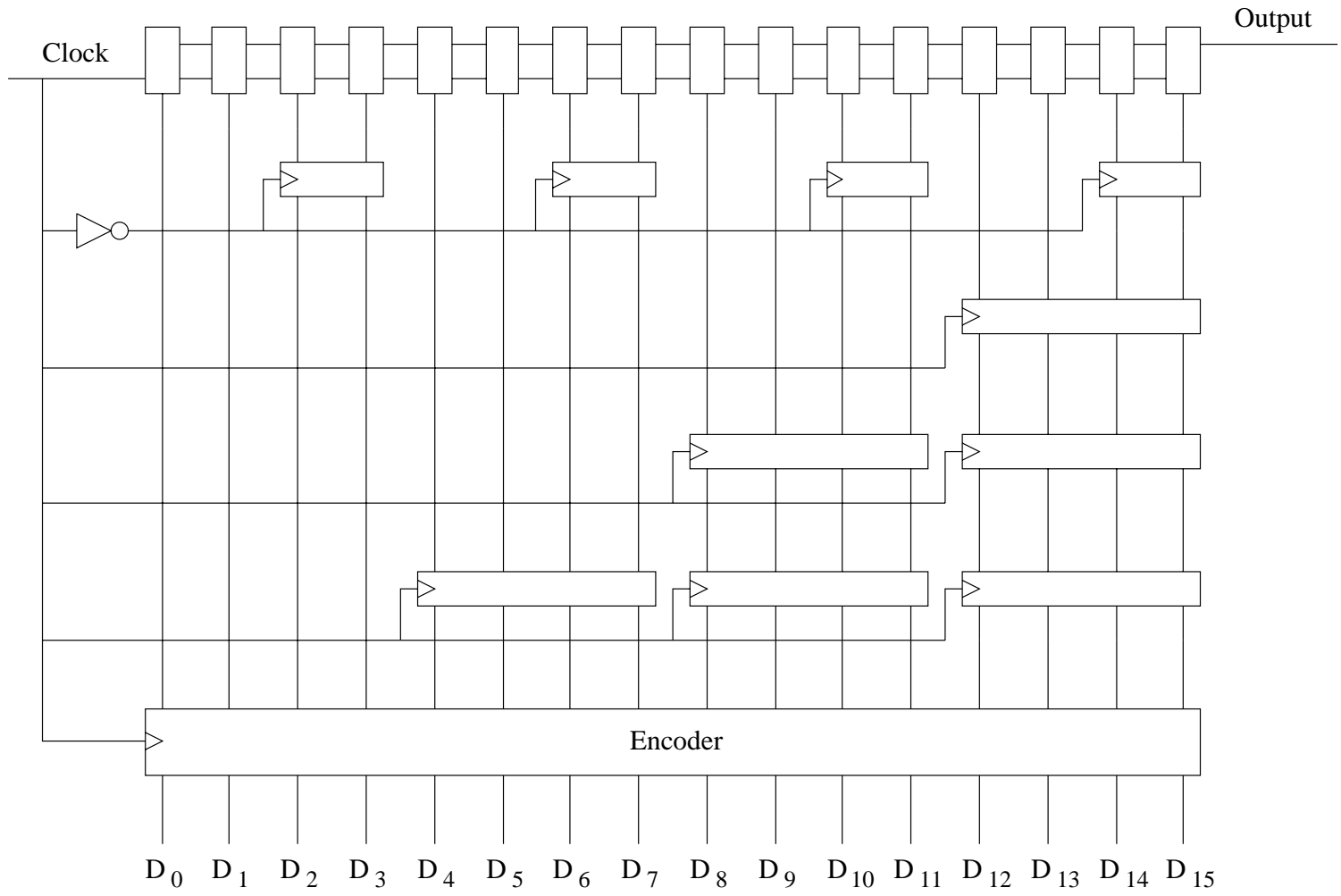


Figure 5: The Continuous Generator

a very small value, then even a tiny error in the delay elements can create a very significant error in the actual difference value.

The first concern related to the accuracy of the delay elements is process and temperature variation. These variations can create a significant error in all the delay elements in a sampler or generator. A means to dynamically compensate for process and temperature in real time is necessary to maintain delay accuracy. By making the delay elements adjustable with a control voltage, and using delay-locked loop techniques, we have been able to successfully maintain delay accuracy in some of our early designs. One aspect of this approach that requires further investigation is the stability of the feedback loop. Additional research into compensation techniques will provide further refinement in the accuracy that can be obtained from the sampler and generator.

An additional facet to the delay accuracy problem appears in the data delay chain of both the sampler and generator. If the data delay element has a data dependency, it will be magnified as the data propagates down the chain. Our primary means of eliminating this data dependency is to use differential logic in the data delay elements. This approach has been successful, but it has disadvantages in power consumption and chip area. Further work is needed to develop power and area efficient delay elements with no data dependency.

Process and temperature gradients on a particular chip can cause the delay to vary from one stage to the next along the delay chain. This effect will introduce timing jitter into the data and clock signals. The only measures that we have taken against this problem are the standard VLSI layout techniques that minimize the effects of gradients. More work is needed to determine if a more advanced gradient compensation approach is practical.

The other major phenomenon that limits resolution accuracy is common to all electrical systems. Noise will cause timing jitter in the delay elements and so create a finite uncertainty in the timing interval between two matched delay stages. Obviously, the noise cannot be eliminated, but it can be minimized using the standard isolation techniques. The potential of the matched delay technique to provide very accurate, very fine timing resolution justifies the cost of very careful noise isolation measures. Further study and experience with the sampler and generator will allow refinement and optimization of our noise minimization approach.

## 4.2 Independence of Resolution and Bandwidth

As described earlier, the resolution of the matched delay devices is determined by the difference between the propagation delays of the delay elements on two parallel tapped delay lines. An individual delay or latch is allowed to cycle at a rate that is much slower than the resolution frequency. As a result, the resolution is not limited to the minimum propagation delay of the circuit elements. In fact, to a large extent, the resolution is independent of the analog bandwidth of the circuit components, and so the sampling or generation rate can exceed the maximum switching rate of the individual components.

With conventional data sampling and generating techniques, obtaining high resolution generally requires using a high performance circuit technology. Therefore, the analog bandwidth usually follows the maximum resolution. The relative independence of resolution and bandwidth provided by the matched delay technique creates the possibility of using CMOS circuit technologies for applications that require high timing resolution but relatively low signal bandwidth.

Most circuit families support higher bandwidth signals on a chip than between chips. The focal point of the bandwidth limitation typically is the package interface. For many applications, the

on-chip bandwidth of CMOS is high enough, but the chip interface bandwidth is insufficient. In these cases, the sampler and generator would allow a high performance system to take advantage of the lower power consumption and higher integration levels of CMOS if the package interfaces could be made fast enough. Using BiCMOS interface circuits, or special CMOS circuit designs, such as small-swing differential CMOS, for the drivers and receivers, may provide enough interface bandwidth. Work is needed to develop improved interface circuits, so that CMOS implementations of the sampler and generator can be used in high performance applications.

Even when very high bandwidth circuits are used, the matched delay technique is most likely to fit best in over-sampling applications. In situations where the timing resolution is comparable to the bandwidth, simpler sampling and generation architectures can be used, Circuit technology choice is the primary consideration in these cases.

### 4.3 Data Parallelization

Another fundamental characteristic of the sampler and generator respectively, is an inherent serial-to-parallel, or parallel-to-serial conversion. This conversion provides two important benefits. First, the data rate that must be handled by the logic that processes the data for the matched delay devices is significantly slower than the external data rate. This factor creates the possibility that an entire high performance system might consist of relatively inexpensive circuitry, because the sampler and generator provide very high timing resolution, and the parallelization allow relatively slow data processing.

The second benefit of the parallelization stems from the fact that many applications, especially in the communications area, need to operate on parallel data. In these cases, conventionally designed systems often contain multiplexors and demultiplexors. The sampler and generator can perform these functions with practically any desired parallelization factor.

The greatly increased number of internal signals that results from the parallelization is a potential problem from a system design perspective, however. A situation that requires a large number of external channels and some central logic that must simultaneously observe or drive all of the corresponding internal signals may result in a difficult interconnection problem. The most important point in preventing this problem is to achieve the correct balance between the degree of parallelization and the the speed of the central logic. Research is needed to develop guidelines for finding this balance.

Multi-chip module (MCM) technology will be a very important solution for this potential problem in many cases. An MCM based system can have a much larger number of connections between chips than a system implemented on one or more printed wiring boards. Also, the signal speeds of those connections can be significantly higher. These factors will allow a matched delay based systems with large numbers of external signals to be designed.

### 4.4 Clock Speed Reduction

A very important advantage of the matched delay technique is the fact that the input clock is slow relative to the sample rate. the sampler and generator do not require a high speed clock to achieve high timing resolution. With conventional techniques, the clock is often the fastest signal used in a system. Matched delays might allow the use of relatively low bandwidth circuit families for some high resolution applications because it is not necessary to bring onto the chip and distribute a very fast clock signal.

Sampling schemes in which either the data or clock are delayed also are able to sample at rates higher than the switching speed of the individual elements, and also do not require a clock with a frequency equivalent to the resolution. However, their resolution is limited by the minimum possible delay element. Since the matched delay resolution is determined by the difference between two propagation delays, the sampler or generator can have a resolution interval that is much smaller than the minimum possible absolute delay.

There is a tradeoff between clock speed and the degree of data parallelization that must be balanced to fit each particular application of the matched delay technique. This tradeoff must also take into account the chip area issue. Further experience in building matched delay based systems will help determine the value of the reduced clock rate.

## 5 Fundamental Questions

In addition to the purely practical issues described above, there are a number of basic questions that must be explored before the matched delay technique can be considered a useful and practical one. First of all, we must develop a detailed characterization of the performance capabilities of the technique, and identify the performance limiting factors. Second, a determination of the levels of performance that can be reached by various circuit technologies is required. Third, an understanding of the costs, relative to other approaches, of using the matched delay technique must be developed. Fourth, a detailed design methodology that provides solutions to the specific problems encountered in applying these techniques is needed. Fifth, we must compare our technique to other approaches on the basis of the above criteria. Finally, an exploration of the classes of applications that might benefit from the matched delay technique must be undertaken.

## 6 Conclusions

In this white paper, we have presented a new technique for achieving very high timing resolution for both the sampling and generation of serial digital data streams. We have outlined the issues that must be examined if the matched delay technique is to be developed into a well defined and useful approach for designing high performance digital systems. We believe that further investigation and development of the matched delay technique will prove its potential for increasing the performance and reducing the cost of many types of high speed digital systems.

## References

- [1] W. van Noije, C. T. Gray, W. Liu, T. A. Hughes, R. K. Cavin, and W. J. Farlow, "CMOS sampler with 1 Gbit/s bandwidth and 25 ps resolution," In *Proc. IEEE Custom Integrated Circuits Conference*, pp. 27.5.1–27.5.4, San Diego, CA, May 1993.
- [2] J. Kang, W. Liu, and R. K. Cavin III, "A Monolithic 625Mb/s Data Recovery Circuit in 1.2 $\mu$ m CMOS," In *Proc. IEEE Custom Integrated Circuits Conference*, pp. 27.3.1–27.3.4, San Diego, CA, May 1994.

- [3] G. Moyer, W. Liu, R. K. Cavin III, and T. Schaffer, “A High Speed CMOS Clock Shaper Using Wave Pipelining,” Technical Report NCSU-VLSI-93-11, North Carolina State University, 1993.
- [4] C. T. Gray, W. Liu, W. A. M. van Noije, T. A. Hughes, and R. K. Cavin, III, “A Sampling Technique and Its CMOS Implementation with 1 Gb/s Bandwidth and 25 ps Resolution,” In *IEEE J. Solid-State Circuits*, vol. 29, no. 3, pp. 340–349, March 1994.
- [5] S. M. Clements, W. Liu, J. Kang, and R. K. Cavin III, “Very high speed continuous sampling using matched delays,” In *Electronics Letters*, vol. 30, no. 6, pp. 463–465, 17th March 1994.
- [6] W. Liu, C. T. Gray, D. Fan, T. Hughes, W. Farlow, and R. K. Cavin, III, “A 250-MHz Wave Pipelined Adder in 2 Micron CMOS” *IEEE J. Solid-State Circuits*, vol. 29, no. 9, pp. 1117–1128, September, 1994.
- [7] C. T. Gray, W. Liu, and R. K. Cavin, III, *Wave Pipelining: Theory and CMOS Implementations*, Kluwer Academic Publishers, October, 1993, ISBN 0-7923-9398-8.