A 32-bit Ultrafast Parallel Correlator Using Resonant Tunneling Devices

Shriram Kulkarni, Pinaki Mazumder, and George I. Haddad
Department of Electrical Engineering and Computer Science
The University of Michigan
1301 Beal Avenue, Ann Arbor, MI 48109-2122

Abstract

An ultrafast 32-bit pipelined correlator has been implemented using resonant tunneling diodes (RTDs) and hetero-junction bipolar transistors (HBTs). The negative differential resistance (NDR) characteristics of RTDs is the basis of logic gates with the self-latching property that eliminate pipeline area and delay overheads which limit throughput in conventional technologies. The circuit topology also allows threshold logic functions such as minority/majority to be implemented in a compact manner resulting in reduction of the overall complexity and delay of arbitrary logic circuits. The parallel correlator is an essential component in CDMA transceivers used for the continuous calculation of correlation between an incoming data stream and a PN sequence. Simulation results show that a nano-pipelined correlator can provide an effective throughput of one 32-bit correlation every 100 ps, using minimal hardware, with a power dissipation of 1.5 watts. RTD+HBT based logic gates have been fabricated and the RTD+HBT based correlator is compared with state of the art CMOS implementations.

1. Introduction

Space based communication systems experience high signal-noise (S/N) ratio in the transmission channel and have inherently low power budgets for communication. An added constraint is the requirement of high reliability and security for space to earth transmissions due to their vital nature in supporting military and civilian systems. Spread spectrum communication increases transmission bandwidth by distributing the data signal energy over a large frequency band by use of a pseudo-noise (PN) spreading sequence. The uniqueness of the PN sequence results in receivers being able to detect the transmitted signal even in the high noise environments due to low cross correlation with extraneous transmissions. Thus, the required transmitter power is reduced in spread spectrum systems. Spread spectrum signals have low probability of detection by unintended receivers and hence provide good security. Similarly, the redundancy in the spread spectrum signal allows for reliable communication. Hence, spread spectrum modulation satisfies the constraints imposed by space based communication systems.

The parallel correlator forms an essential component in a digital communication system. Typically, in spread spectrum systems, a parallel correlator computes the correlation of the incoming data stream with a pre-determined pseudo-noise (PN) sequence of a fixed length. This correlation value is used to estimate the output data. For a binary input data stream, the result of such an operation essentially determines whether the output should be 0, 1 or indeterminate. An indeterminate output is primarily caused due to the receiver PN sequence not being the same as the transmitter sequence. Thus, communication between different transceivers can be regulated on the

This work was supported by the U.S. Army Research Office under URI program contract DAAL03-92-G-0109 and by ARPA under contract DAAH04-93-G-0242

basis of PN sequence uniqueness. This provides the capability of rejecting interference from multiple transmission paths and jamming [1].

The correlator as described in this paper is particularly suited for direct sequence spread spectrum systems that use binary phase shift keying as the digital modulation. Figure 1 shows the essential function of the spread spectrum demodulator along with waveforms for desired signal reception and jamming signal rejection. The serial input data stream is shifted with each clock cycle and correlation is performed between the fixed PN sequence and as many stored bits of the input data stream as the length of the PN sequence. The ability of the system to respond only to the spreading code while rejecting others makes it useful in systems that experience jamming and multipath interference. The same feature is the basis of code division multiple access (CDMA) systems that allow multiple users to carry out independent messaging in a single spectrum band. The correlation value between the incoming data stream and the PN sequence has to be generated at each clock cycle. If a purely combinational circuit along with a shift register were chosen to implement the correlator, for long PN sequences, it would result in extremely slow operation due to many levels of logic required for computation of correlation. However, in a bit serial communications application as described in this paper, there is no data dependence and hence deep pipelining schemes can be effectively used to improve the throughput of the correlator.

2. Theoretical development

From a hardware viewpoint, correlation between the two binary streams can be represented as follows.

$$\psi(\tau) = \Sigma(f(t) \oplus g(t-\tau)) \tag{1}$$

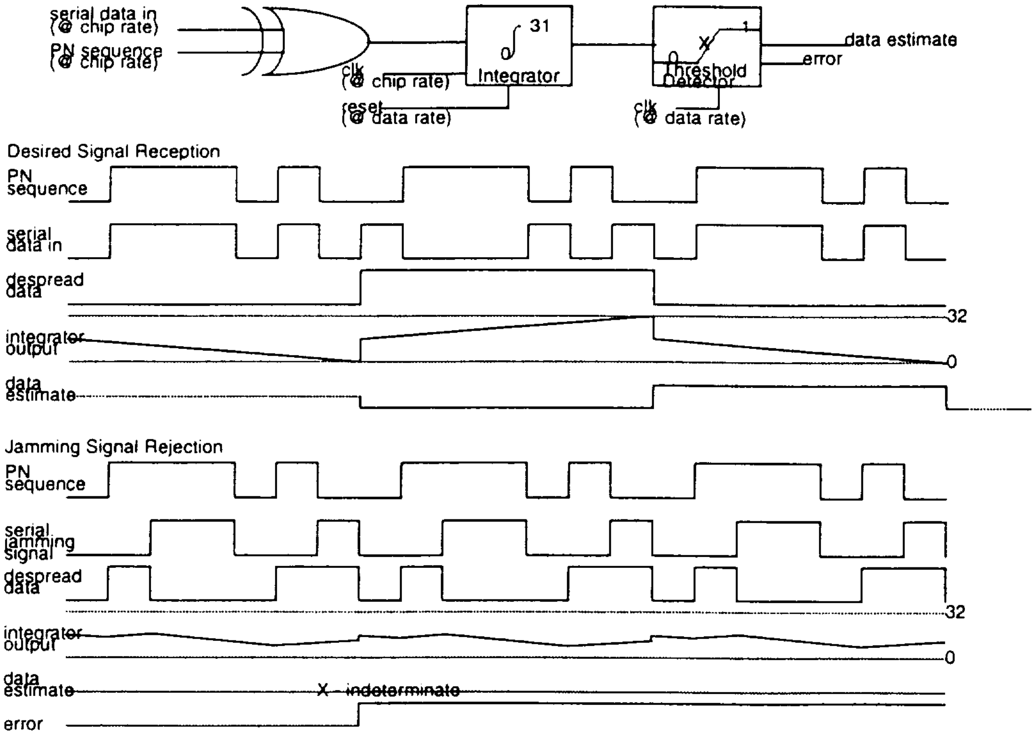


Figure 1. Spread spectrum demodulator

Here, $f(t)$ and $g(t)$ are binary data streams which specifically represent the PN sequence and the input data stream for discussion of the parallel correlator. The XOR operator correlates two binary inputs i.e. it produces a logic 1 output only if the two signals are unlike. The summation of the XOR outputs over the length of the signals gives a measure of the likeness between the two signals. The difference between the number of 1s and the number of 0s in the correlation vector will result in a number that ranges from the negative of the PN sequence length, through 0, up to the positive of the PN sequence length reflecting a 0, indeterminate and 1 output respectively. Thresholds can be set for 0 and 1 detection to account for noise in the channel. The above number, henceforth referred to as the *correlation value*, can also be written as follows.

$$\textit{Correlation Value} = 2\cdot(\Sigma \textit{ of 1s}) - (\textit{PN sequence length}) \qquad (2)$$

3. RTD-HBT logic family

The current-voltage characteristics of an RTD can be approximated by the piecewise linear form shown in Figure 2.

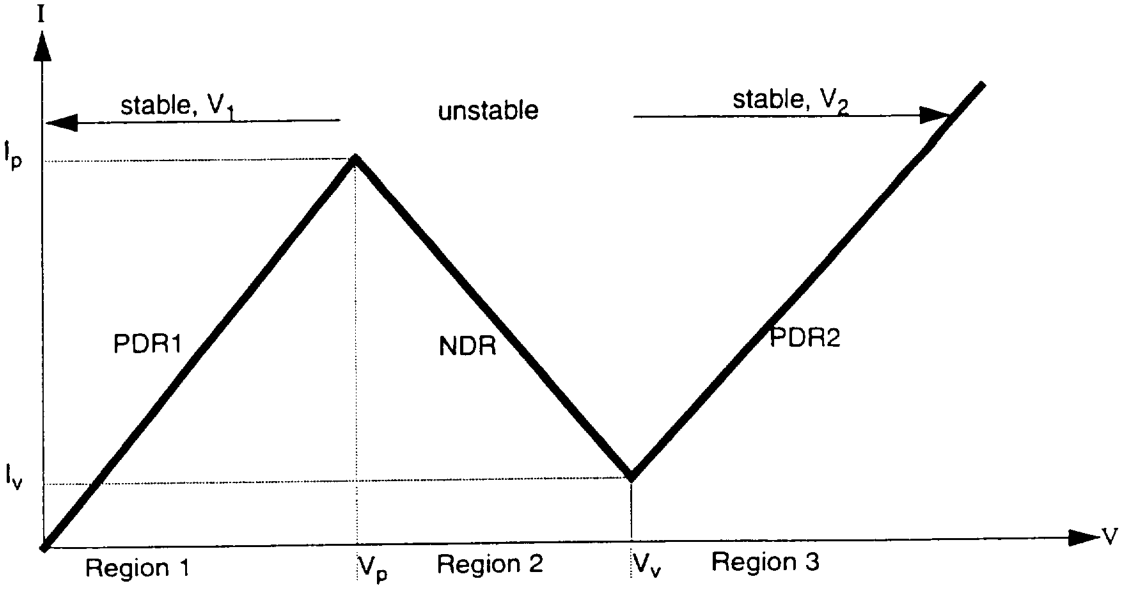


Figure 2. Piecewise approximation of RTD characteristics

As the voltage applied across the device terminals is increased from zero, the current increases until the V_p, the peak voltage of the RTD. The corresponding current is call the peak current, I_p of the RTD. As the voltage across the RTD is increased beyond V_p, the current through the device drops abruptly due to tunneling until the voltage reaches V_v, the valley voltage. The current at this voltage is the valley current, I_v. Beyond V_v, the current starts increasing again. For a current in $[I_v, I_p]$ there are two possible stable voltages; $V_1 \leq V_p$ or $V_2 \geq V_v$. The tunneling characteristic of the RTD facilitates implementation of self latching circuits.

3.1 Bistable mode operation

A binary logic circuit is said to operate in bistable mode when its output is latched, and any change in the input is reflected in the output only when a clock or other evaluation signal is applied. The bistable mode has been used in several earlier technologies, notably in superconducting logic [2]. Superconducting logic typically uses a multi-phase AC power source to periodically reset/evaluate each gate. Similar logic using resonant tunneling devices has been proposed by several authors [3, 4, 5]. The chief disadvantage of these circuits is the requirement of an AC power source whose frequency determines the maximum switching frequency. The RTD+HBT logic circuits described below use a DC power supply and multiphase clocks but the clock signals are not required to supply large amounts of power as in the case of the earlier circuits.

The operating principle of the new bistable element may be understood by considering the circuit shown in Figure 3. There are *m* input transistors and one clock transistor driving a single RTD load. The input transistors can be in either of two states - On, with a collector current of I_H or Off, with no collector current. The clock transistor can be in one of two states - High, with collector current I_{CLKH}, and Quiescent with collector current I_{CLQ}. In addition, there is a global reset state where all the collector currents are 0. When the clock transistor current is at I_{CLKQ}, the load lines in Fig. 1 show that the circuit has two possible stable operating points for every possible input combination. When the clock current is I_{CLKH}, there is exactly one stable operating point for the circuit when *n* or more inputs are high and the sum of the collector currents is $nI_H + I_{CLKH}$. This operating point corresponds to a logic 0 output voltage. Hence this circuit can be operated sequentially to implement any non-weighted threshold logic function $f(x_1, x_2, ..., x_m;n)$, where $f(x_1, x_2, ..., x_m)$ is 1 if and only if $(x_1 + x_2 + ... + x_m) < n$, and $x_1, x_2, ..., x_m$ take on values of either 0 or 1.

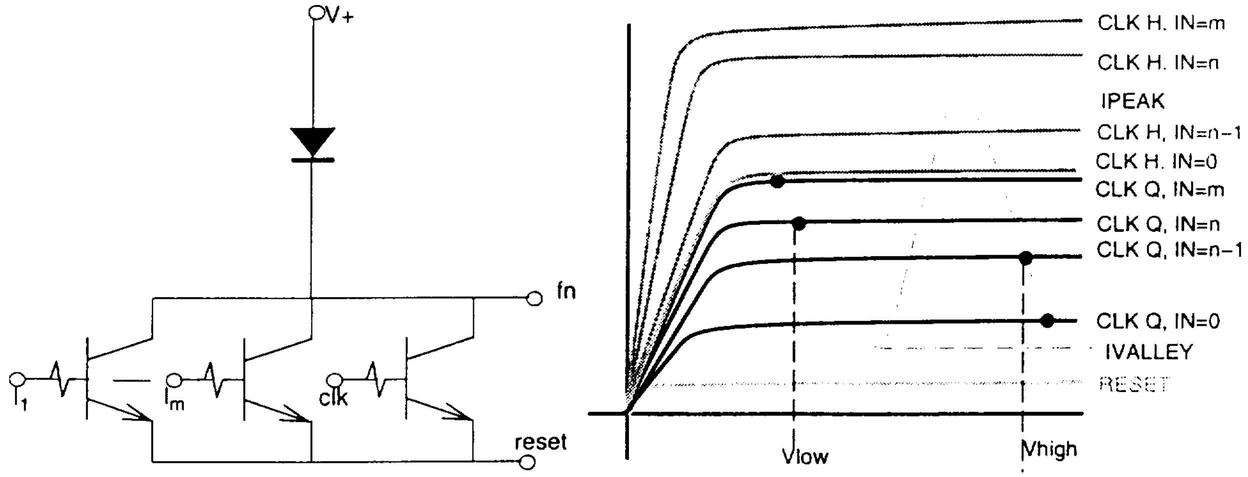


Figure 3. RTD+HBT bistable logic gate operating principle

The operating sequence is as follows:

1. Inputs I_1 through I_m change.
2. The *reset* line goes high forcing all transistors into cut-off. The current through the RTD falls below the valley current, and the *fn* node is pulled high.
3. The *reset* line goes back to 0. The *fn* node remains high.
4. The *clk* signal goes high, causing the total current through the RTD to increase. If more than *n* inputs are high, the current through the RTD exceeds the peak current causing a jump to the second positive differential resistance (PDR) region of the RTD characteristic corresponding to $V_{RTD} > V_{VALLEY}$, where V_{RTD} is the voltage across the RTD and V_{VALLEY} is the valley voltage of the RTD. This results in the *fn* node going low. If less than *n* inputs are high the current through the RTD does not exceed the peak current and the operating point remains in the first PDR region of the RTD,

where $V_{RTD} < V_{PEAK}$, and V_{PEAK} is the RTD peak voltage. Thus, *fn* remains high.

5. The *clk* signal goes to its quiescent state so that the current through the clock transistor is I_{CLKQ}. The output voltage at node *fn* reaches a stable level corresponding to whether the RTD was in the first PDR region or the second PDR region in the previous step of the sequence.

For a three input circuit, three non-trivial threshold functions can be implemented for the cases where $n = 1, 2, 3$. For $n = 1, f_1(x_1, x_2, x_3) = 0$ if and only if 1 or more inputs are high. This corresponds to a NOR function. For $n = 3, f_3(x_1, x_2, x_3) = 0$ if and only if all 3 inputs are high. This corresponds to a NAND function. For $n = 2, f_2(x_1, x_2, x_3) = 0$ if and only if 2 or more inputs are high. This corresponds to an inverted majority or inverted carry function.

Figure 4 shows the simulated traces obtained from NDR-SPICE [6] for an inverter, a three input NOR, and a three input MINORITY gate designed using RTDs and HBTs. It can be seen that the outputs change only on arrival of the clock pulse and hence the circuits are operating in bistable mode. Input and output voltage swings are matched to enable cascaded circuits to function correctly. The signal levels are 1V for logic zero and 2V for logic one.

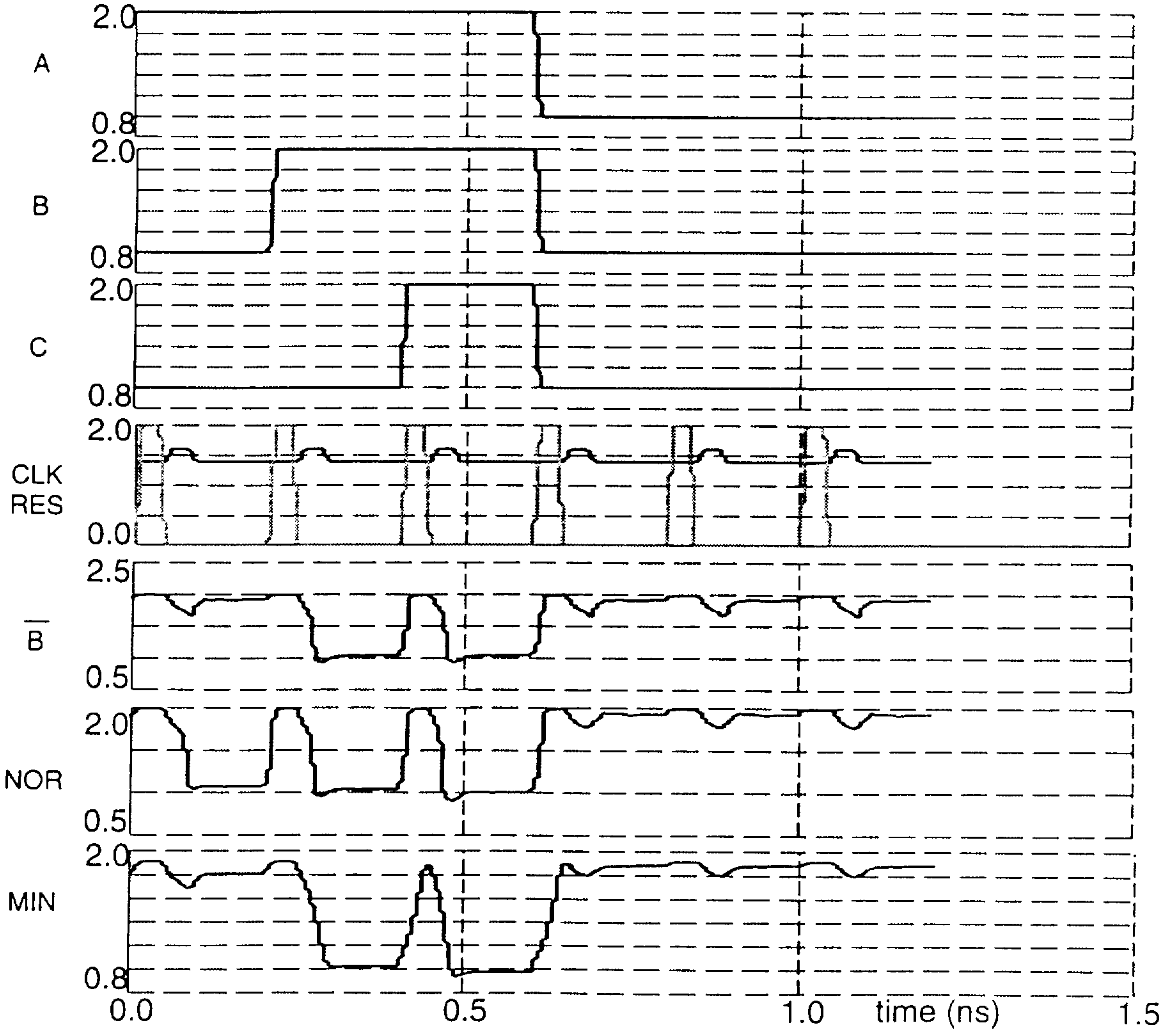


Figure 4. RTD+HBT basic gates simulation

3.2 Design constraints

We now present the design equations for a k input threshold gate with a threshold value of n. Let m be the are of the RTD used and let J_P and J_V represent the peak and valley current densities of the RTD, respectively. I_H, I_{CLKH} and I_{CLKQ} are defined as in section 3.1. The design constraints for the aforementioned gate can be written as:

$$h = mJ_P - (I_{CQ} + kI_H) > 0 \tag{3}$$

$$l = I_{CQ} - mJ_V > 0 \tag{4}$$

$$hh = mJ_P - (I_{CH} + (n\text{-}1)I_H) > 0 \tag{5}$$

$$hl = I_{CH} + nI_H - mJ_P > 0 \tag{6}$$

$$I_{CH} > I_{CQ} > 0 \tag{7}$$

where,

h = quiescent clock, logic high switching margin

l = quiescent clock, logic low switching margin

hh = high clock, logic high switching margin

hl = high clock, logic low switching margin

The design process begins by choosing the input high and low voltages. The input high and low voltages must respectively turn the input transistors on or off. To maintain good noise margins, signal voltage swings should be maximized. However, for cascaded logic stages to operate correctly without resorting to use of level shifters, it is necessary to match the input and output voltage swings. An optimum match resulted in the signal voltage levels being set to 1V for logic 0 and 2V for logic 1. The input transistor size determines the value of I_H. I_H should be small to minimize power consumption and area, but should be large enough to have good switching margins hh and hl. The value of I_{CLKQ} and the area of the RTD are determined from the equations involving I_{CLKQ}. The peak and valley current densities (J_P and J_V) are determined by the growth process, and the RTD area factor m determines the actual currents. The simulations in this paper use an RTD with peak current of 100 μA and a valley current of 25 μA for $m = 1$. Setting $I_{CLKQ} = (m(J_P + J_V) - kI_H)/2$ satisfies both equations (3) and (4), when m is chosen such that $m > 3I_H/(J_P - J_V)$. This also results in the equalization of the switching margins h and l. Choosing $I_{CLKH} = mJ_P - (n - 0.5)I_H$ satisfies the remaining design equations and also equalizes the switching margins hh and hl. The clock line voltages and the clock transistor sizes are determined from the values of I_{CLKQ} and I_{CLKH}. The switching margins for the circuits are $0.5I_H$ or a 50% variation is allowable in the drain current of any one input transistor. When all transistors are systematically larger or smaller, the allowable variation before the circuit malfunctions is $0.5I_H/n$. For a NOR gate, $n = 1$ and the allowable variation is 50%. For a 3-input inverted majority gate the allowable variation is 25% and for a 3-input NAND gate it is 16%. Thus, the switching margin of a NOR gate remains constant with increase in the number of inputs whereas, the switching margin of a NAND gate degrades rapidly with increase in the number of inputs. Thus, the best design margins are provided by the NOR function and the NAND function should be avoided in so far as possible.

3.3 Co-integration of RTDs and HBTs

RTDs and HBTs were integrated on the same wafer to build a 3-input threshold gate with the same topology as the circuit shown in Figure 3. Figure 5 shows a photomicrograph of the integrated circuit. The functionality of the circuit is determined by the input and clock voltages as discussed in section 3.2. By adjusting the values of the supply voltage, input high voltage and clock voltages NAND, NOR and MINORITY functions were tested and the oscilloscope traces are shown in Figure 6. It should be noted that in the correlator design, the signal voltages are fixed and hence functionality of the gates is determined by the device sizes.

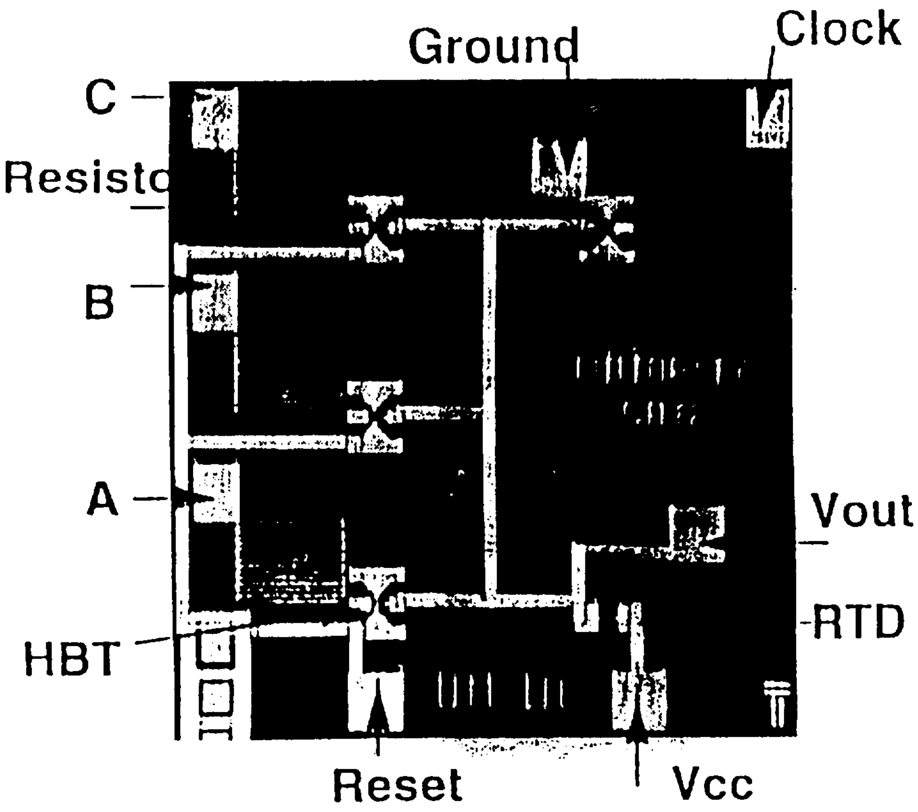


Figure 5. Photomicrograph of RTD+HBT bistable gate

3.4 Pipelined computation

Pipelining is a well studied means of speeding up any computation. An existing combinational block is divided into several sequential stages such that each stage performs a different operation during a particular clock cycle. The drawback of pipelining is that each computation takes the same or more time as nanopipelining [7] but there is an added penalty in the area devoted to the pipeline latches in the circuit.

Consider a combinational block that is composed of n stages with each stage having a delay of t_c. This results in a total delay of $n \cdot t_c$. We could partition the combinational block into k stages from 1 to n, where each stage output is latched. If we assume a latch delay of t_l, the maximum delay of the circuit is now $(n \cdot t_c/k + t_l)$. The throughput of the circuit increases from $1/(n \cdot t_c)$ to $1/(n \cdot t_c/k + t_l)$ but the latency increases from $n \cdot t_c$ to $n \cdot (t_c + t_l)$. Also, if a_c is the area of the combinational block; after pipelining, the area of the circuit increases to $a_c + k \cdot m \cdot a_l$, where a_l is the area of a latch and m is the number latches at each stage. The best possible theoretical throughput would be $1/t_c$ when we have latches at the output of each combinational stage. However, if all combinational stages don't have the same delay, then the maximum achievable throughput with the use of separate pipeline latches is $1/(b \cdot t_c + t_l)$ where $b \cdot t_c$ is the longest combinational stage

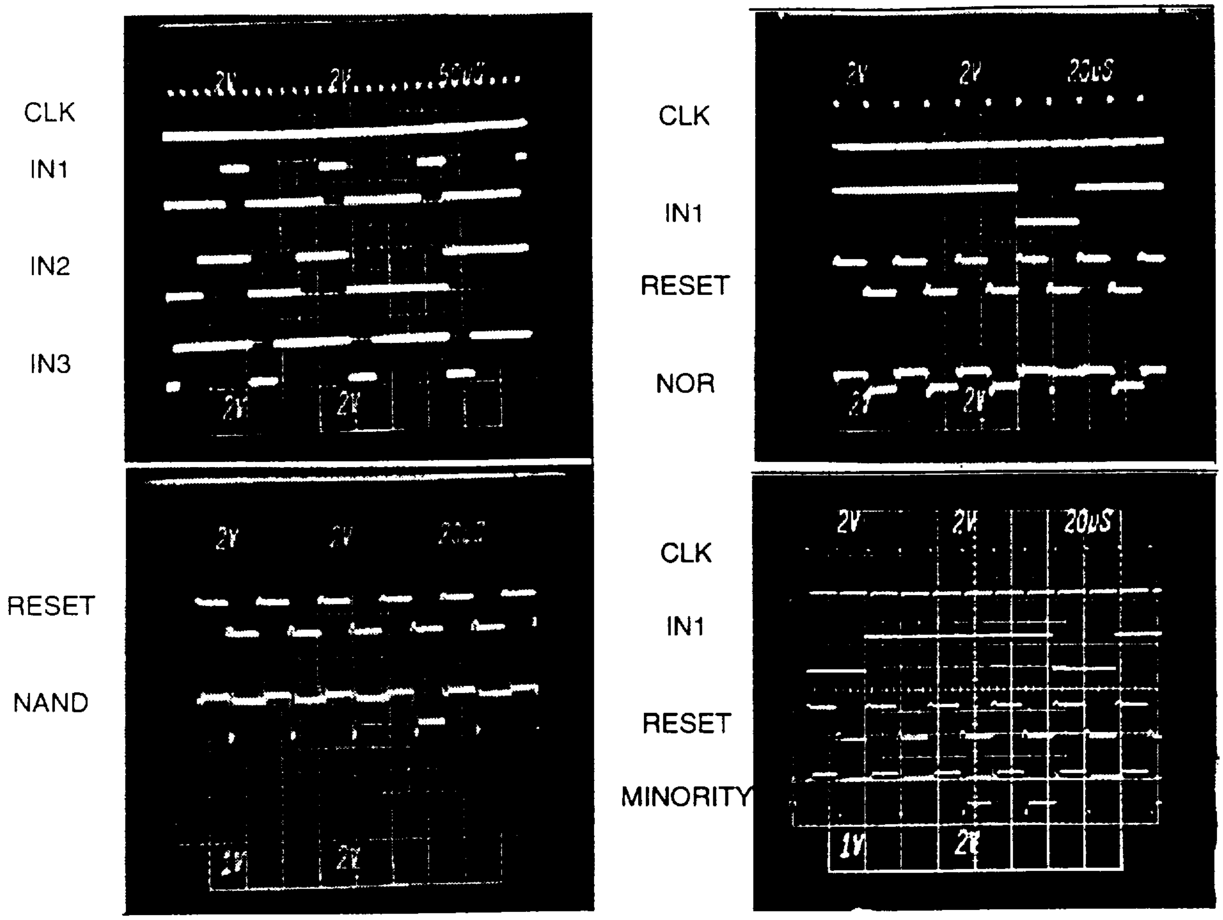


Figure 6. Oscilloscope traces for fabricated RTD+HBT primitive gates

delay. If the latch delay t_l is much larger than the longest combinational delay $b{\cdot}t_c$, it places an upper bound on the maximum achievable throughput of the pipelined circuit. Thus, we see that pipelining using conventional logic results in direct trade-offs between the area of the pipeline latch and the achievable throughput. The use of bistable NDR devices in designing circuits improves the performance of nanopipelined circuits over conventional pipelined circuits because the latch delay, t_l=0. Also, if latency is not of concern, each logic gate can operate in the bistable mode resulting in maximum possible throughput.

3.5 Nanopipelined full adder implementation

The basic bistable logic gates mentioned previously are used to build a nanopipelined full adder that best illustrates the advantages of the NDR logic family. For the parallel correlator, we prefer an adder with complementary sum and carry outputs in order to reduce the number of pipeline stages and hence the latency of the circuit. The complementary sum and carry functions for a 1-bit full adder are written as follows.

$$\bar{S} = \overline{a \oplus b \oplus c_{in}} \qquad (8)$$

$$\bar{C} = \overline{a \cdot b + b \cdot c_{in} + c_{in} \cdot a} \qquad (9)$$

The $\bar{S}$ function is implemented as a three level nanopipelined circuit whereas the $\bar{C}$ function is implemented using a single minority gate. The circuit for the 1-bit full adder is shown in Figure 7. It is apparent that the $\overline{S}$ and $\overline{C}$ outputs are not synchronized with each other. For a single stage of addition, we would need to add two bistable buffers at the $\overline{C}$ output to synchronize the $\overline{S}$ and $\overline{C}$ outputs. However, in the correlator we perform several successive stages of addition and synchronization at each adder will result in increased latency. Hence, synchronization is performed after all stages of addition are complete. For correct operation of the true-bistable logic gates a reset and evaluate pulse is required as mentioned previously. However, when multiple gates are cascaded, as in the implementation of the full adder, a gate must be evaluated only after all its inputs have been correctly evaluated. This requires a two-phase evaluation scheme in which each gate is evaluated in a different phase than its fan-ins and fan-outs. An example timing relationship between phases of consecutive logic blocks for the parallel correlator is illustrated in Figure 8.

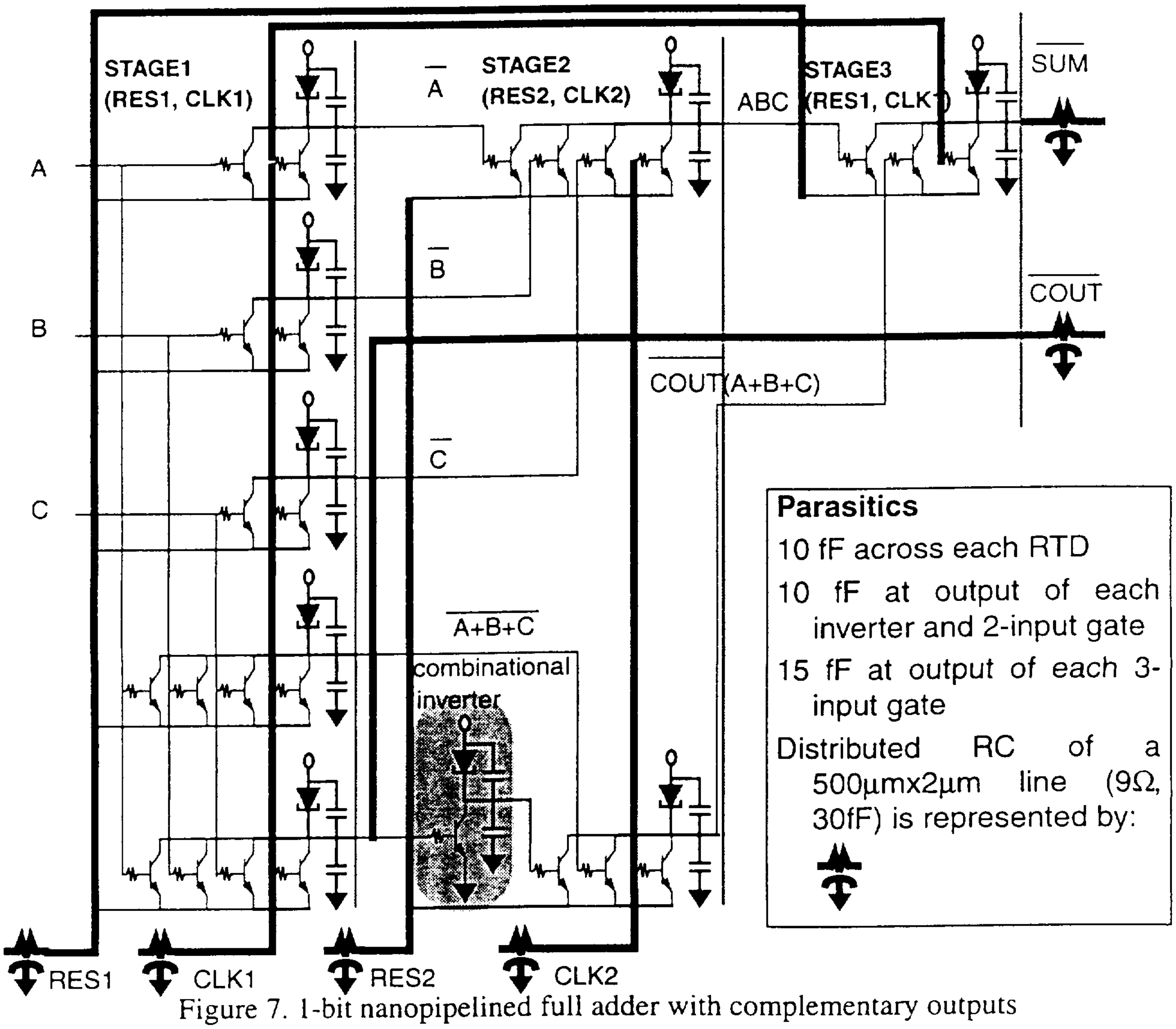


Figure 7. 1-bit nanopipelined full adder with complementary outputs

The *res1* and *clk1* signals form phase1 of the clock whereas *res2* and *clk2* form phase2 of the clock. The two phases of the clock must be non-overlapping. However, the reset and clock signals of a phase may partially overlap as shown in Figure 8. A large overlap period between the aforementioned signals is not desirable since the circuit output is not valid during this time. The simulated output for the 1-bit nanopipelined adder is shown in Figure 9. To project realistic performance, load capacitances and parasitics have been added to the RTDs and HBTs used in the circuit. Also, clock and reset lines are assumed to be global lines with a distributed RC parasitic elements as shown in Figure 7. The circuit outputs are assumed to drive global bus lines across the chip. The two phase clock consisting of *reset1-clock1* and *reset2-clock2* operates at 10 GHz.

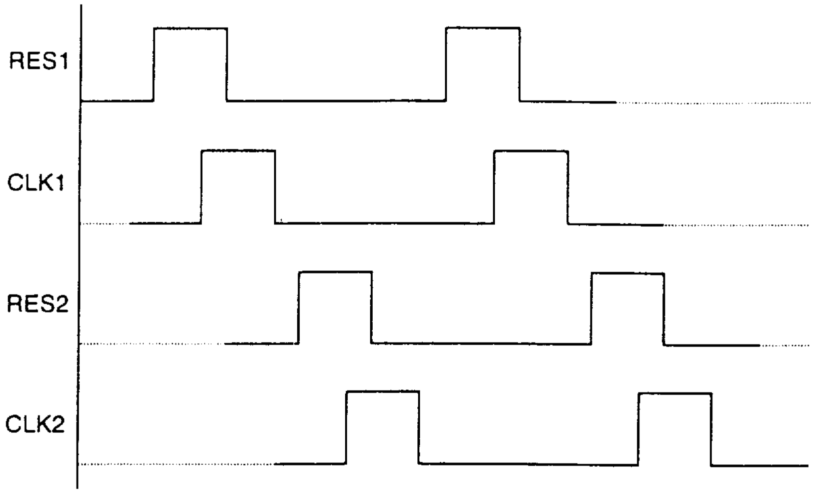


Figure 8. Multiphase timing scheme

4. Correlator implementation

The block diagram of the pipelined correlator is illustrated in Figure 10. A 32-bit latch holds the PN sequence. The input is a serial bit stream which is fed to a 32-bit shift register. The 32-bit latch and 32-bit shift register are each composed of 64 bistable inverters. A pair of cascaded bistable inverters each operating on single, separate phases of the two-phase clock form the basic 1-bit latch. The 32-bit raw correlation vector is generated by performing a bitwise XOR operation on the PN sequence latch output and the most recent 32 bits of the sampled signal available at the shift register output. The raw correlation vector is registered and this forms the input to the pipelined adder network that determines the difference between the number of 1s and 0s in the raw correlation vector. This is the correlation value between the incoming signal and the resident PN sequence and is determined for the 32 most recent data bits at every clock cycle. This value ranges from -32 to +32. The functional description of the correlator is illustrated in the equations (10) through (14).

$$data[31 \leftarrow 0] = \left\{ D^{32}(d_{in}), D^{31}(d_{in}), \ldots, D^{1}(d_{in}) \right\} \tag{10}$$

$$code[31 \leftarrow 0] = \left\{ D^{1}(PN_{31}), D^{1}(PN_{30}), \ldots, D^{1}(PN_{0}) \right\} \tag{11}$$

$$corr[31 \leftarrow 0] = code[31 \leftarrow 0] \oplus data[31 \leftarrow 0] \tag{12}$$

$$sum[5 \leftarrow 0] = \sum_{i=0}^{31} corr[i] \tag{13}$$

$$diff[6 \leftarrow 0] = 32_{d} - 2 \bullet sum[5 \leftarrow 0] \tag{14}$$

Here, $D_i(s)$ represents the value of signal s, i clock cycles prior to the current input.

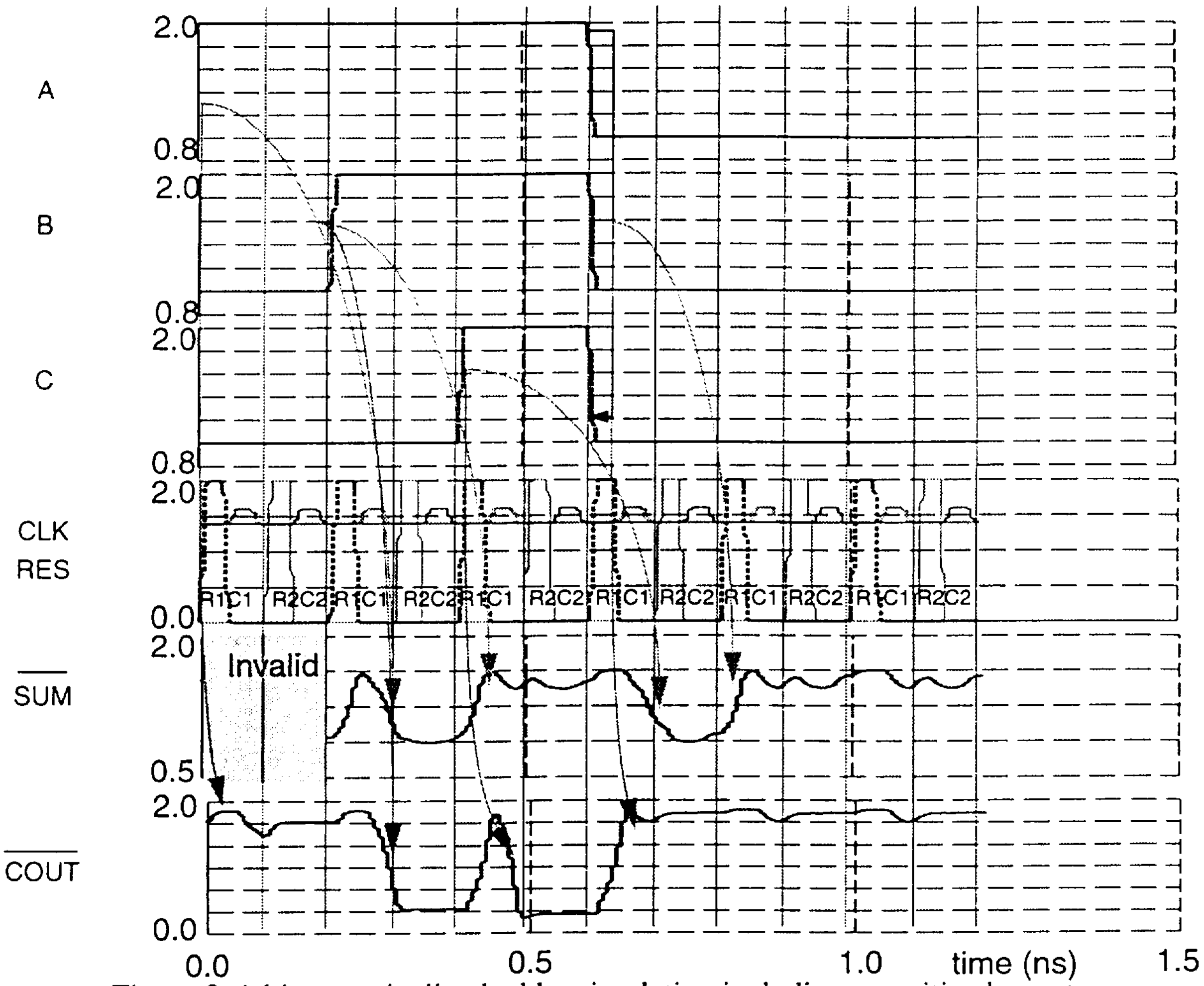


Figure 9. 1-bit nanopipelined adder simulation including parasitic elements

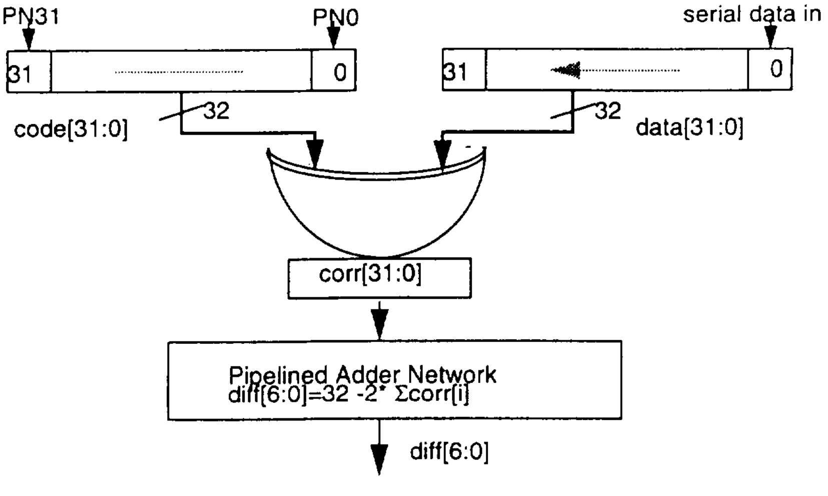


Figure 10. Pipelined correlator block diagram

4.1 Pipelined Adder Network

The adder network consisting of 26 nanopipelined full adders, 11 nanopipelined half adders, and 36 bistable inverters is illustrated in Figure 11. The adders used in the design have complemented sum and carry outputs in order to reduce pipeline latency. The input to the adder network is the raw correlation vector generated by the 32-bit bistable XOR network. The circuit performs eighteen stages of addition to generate a 7-bit result which is the difference between the number of 1s and number of 0s in the correlation vector. Since each stage is nano-pipelined due to use of self latching gates in the bistable adders, the throughput of the circuit is one 32-bit correlation every cycle. However, since the seven bits of the adder network output are not simultaneously generated, bistable inverters are required to synchronize the bits such that all seven bits of a correlation appear in order at the output of the correlator. The least significant bit of the correlation value is always 0 since the difference between the number of 1s and number of 0s in a 32-bit vector is always even. The pipelined adder network essentially sums up the number of 1s in the correlation vector. Bits 0, 1, 2, 3 and 4 of the sum of 1s directly translate to bits 1, 2, 3, 4 and 5 of the difference between number of 1s and number of 0s. Bit 6 of the correlation value is computed while bit 5 of the sum of 1s is being generated by connecting the carry input of the final full adder to V_{dd}. This achieves the 2s complement subtraction required for computing the difference between the number of 1s and number of 0s in the correlation vector. No additional pipe stages are required for this conversion.

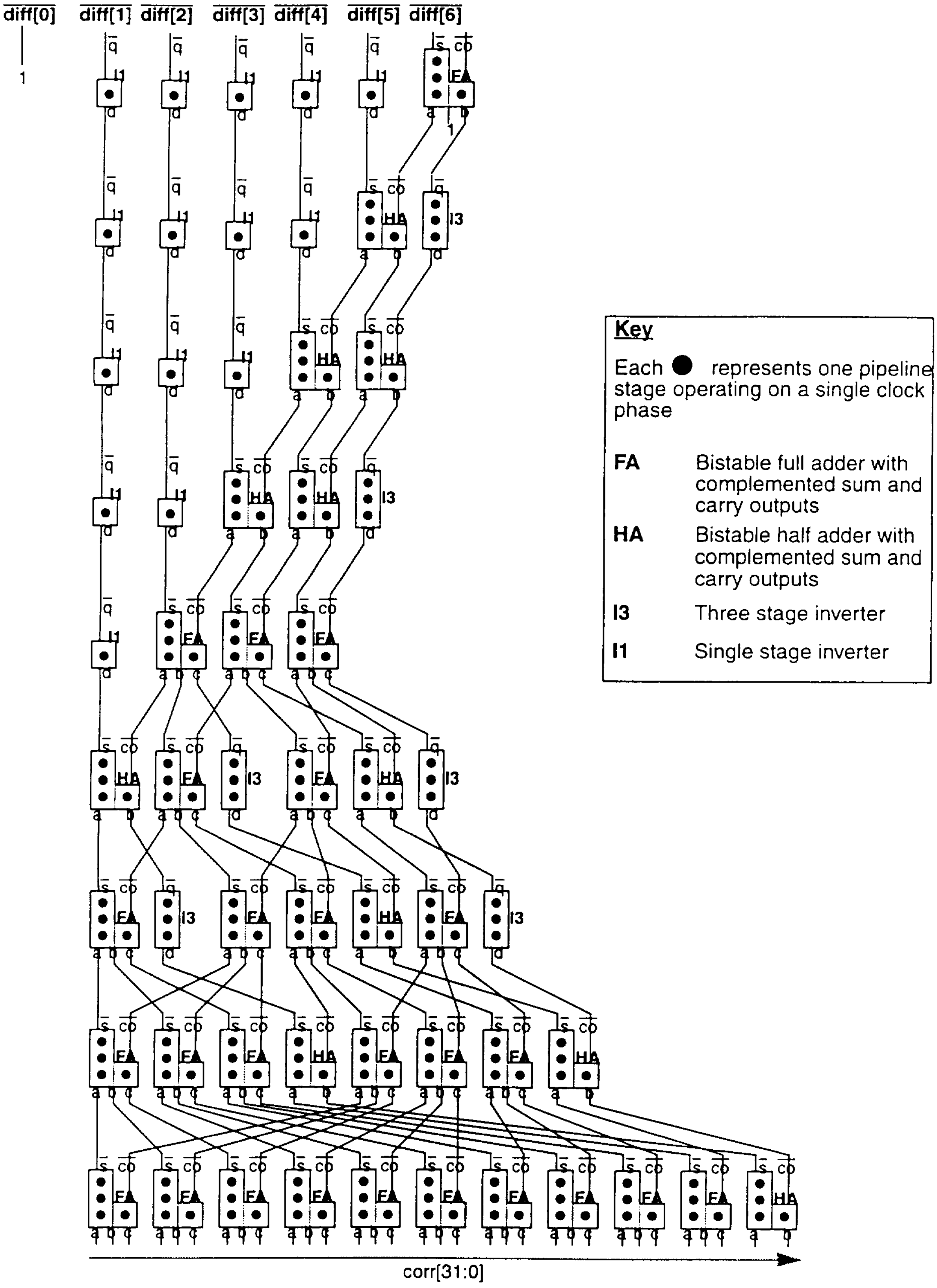
The functional simulation of the 32-bit parallel correlator is shown in Figure 12. The PN sequence for this simulation is chosen to be AAAAAAAA Hex. Note, that this is not an optimum PN sequence but rather is chosen for the ease of illustration of the functionality of the correlator. The input is a pattern of alternating 1s and 0s which results in the 32-bit shift register output toggling between AAAAAAAA Hex and 55555555 Hex at each cycle. This causes the raw correlation vector to alternate between all 1s (FFFFFFFF Hex) and all 0s (00000000 Hex) with each cycle. Thus, the desired correlation difference should be +32 decimal and -32 decimal respectively for the two cases mentioned above. This is seen to be the case in the simulation output. It should be noted that the simulation output reflects changes in the input 10 cycles prior to the output due to pipeline latency. However, the same input pattern has been maintained and is shown in the current plot for the purpose of illustration.

4.2 Comparison with CMOS technology

The correlator designed using RTDs and HBTs is compared with a CMOS implementation using 0.5 micron process technology. The results of the comparison for three circuits - the basic bistable majority gate, the bistable full adder and the 32-bit parallel correlator - are presented in Table 1.

Table 1: Comparison of RTD+HBT circuits with CMOS implementations

Parameter	Bistable Majority		Bistable full adder		32-bit Parallel Correlator	
	CMOS (0.5μ)	RTD+HBT	CMOS (0.5μ)	RTD+HBT	CMOS (0.5μ)	RTD+HBT
Device count	20	5	68	34	6000	2060
Power dissipation	0.7 mW	2 mW	2.3 mW	12 mW	600 mW	1.5 W
Speed	400 MHz	20 GHz	400 MHz	10 GHz	400 MHz	10 GHz
Power-Delay product	1.75 pJ	0.1 pJ	5.75 pJ	1.2 pJ	1.5 nJ	0.15 nJ


$\overline{\text{diff[0]}}$
$\overline{\text{diff[1]}}$
$\overline{\text{diff[2]}}$
$\overline{\text{diff[3]}}$
$\overline{\text{diff[4]}}$
$\overline{\text{diff[5]}}$
$\overline{\text{diff[6]}}$
1
$\overline{q}$
I1
d
$\overline{s}$
$\overline{co}$
FA
HA
I3
a
b
c
corr[31:0]

Key

Each ● represents one pipeline stage operating on a single clock phase

FA	Bistable full adder with complemented sum and carry outputs
HA	Bistable half adder with complemented sum and carry outputs
I3	Three stage inverter
I1	Single stage inverter

Figure 11. Pipelined Adder Network

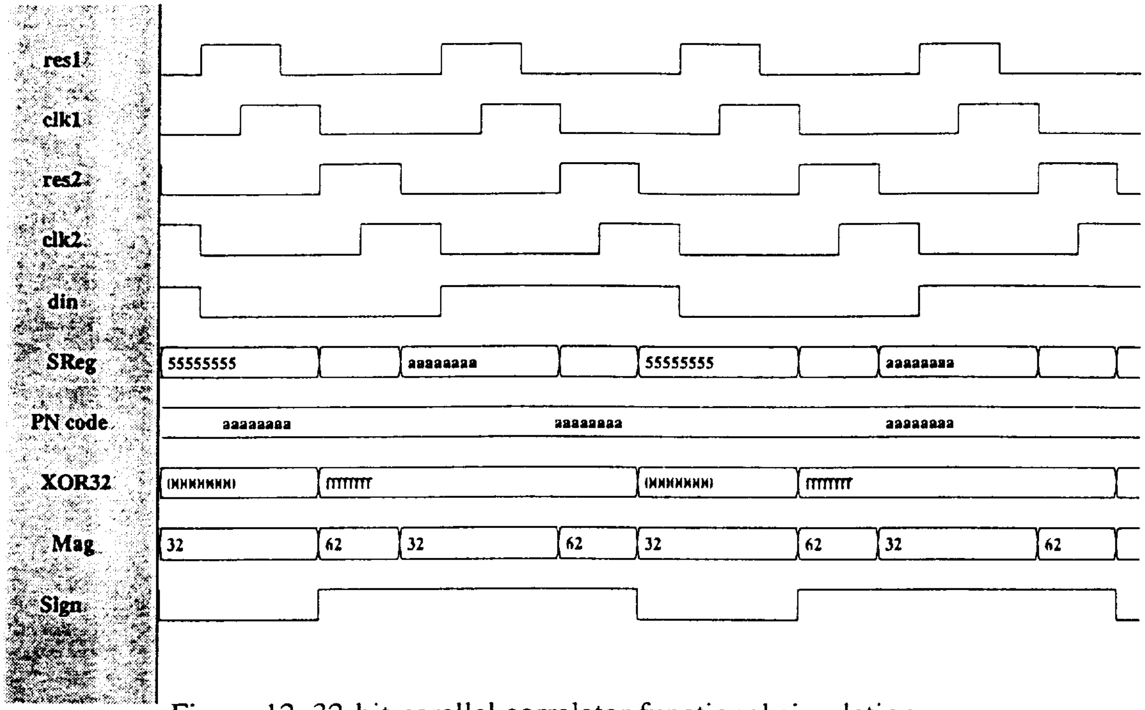


Figure 12. 32-bit parallel correlator functional simulation

The RTD+HBT based correlator offers a tenfold improvement in power-delay product even though it consumes greater absolute power. The fewer number of devices used in the correlator also imply a reduction in wiring lengths and hence parasitics and delays associated with interconnects are much smaller in the RTD+HBT correlator.

Conclusions

The synchronous, sequential nature of true-bistable gates using RTDs and HBTs has been exploited to build a very high speed and compact parallel correlator. Design equations and constraints have been studied and a design methodology for RTD+HBT bistable logic gates is proposed. The bistable nature of the logic gates has demonstrated advantages over conventional logic families by eliminating pipeline area and delay overheads in deep pipelined logic systems resulting in improved throughput and smaller circuit size. The compact implementation of threshold functions allows a single gate carry function which facilitates design of high speed arithmetic and logic functions used in the correlator. Reduction in device count has led to shorter interconnections resulting in reduced parasitic delays. The nanopipelined correlator offers a tenfold lower power-delay product as compared to a state of the art CMOS implementation. The proposed design style has applications in the development of high speed digital communication system architectures to achieve several Gb/s data throughput. In particular, for space based communication systems, nanopipelined RTD+HBT based logic designs offer compact solutions with very low power-delay products.

References

[1] R. E. Ziemer, and R. L. Peterson, *Digital Communications and Spread Spectrum Systems*, New York: Macmallan, 1985

[2] T. VanDuzer, "Superconductor digital ICs," in *VLSI Handbook* (J. DiGiacomo, ed.), New York: McGraw-Hill, 1989.

[3] K. Maezawa and T. Mizutani, "A new resonant tunneling logic gate employing monostable-bistable transition," *Japanese Journal of Applied Physics*, vol. 32, pp. L42-L44, Jan 1993.

[4] S. Mohan, P. Mazumder, R. K. Mains, J. P. sun, and G. I. Haddad, "Logic design based on negative differential resistance characteristics of quantum electronic devices," *IEE Proceedings-G*, vol. 140, pp. 383-391, Dec. 1993.

[5] W. Williamson, III, "High-speed RTD-based logic for systems applications," in *ULTRA electronics program review*, ARPA, Oct. 1994.

[6] P. Mazumder, J. P. Sun, S. Mohan and G. I. Haddad, "DC and transient simulation of resonant tunneling devices in NDR-SPICE," in *21st International Symposium on Compound Semiconductors*, 1994

[7] S Mohan, P. Mazumder, and G. I. Haddad, "Ultra-fast pipelined arithmetic using quantum electronic devices," *IEE Proceedings-E*, vol. 141, pp. 104-110, Mar. 1994.