# Monolithic Chaotic Communications System

*P. Chiang, W. Dally, E. Lee*

Stanford University
353 Serra Mall
Stanford, CA 94305

## ABSTRACT

A chaotic transmitter and a synchronized chaotic receiver have been designed in a 0.25um CMOS process. Both transmitter and receiver use conversions between voltage and time to implement nonlinear tent-map functions. The transmitter outputs chaotic pulse position modulated (C-PPM) data as a stream of 2.3V p-p pulses of <1ns duration. The chaotic receiver uses the timing information between pulses to regain the chaotic state of the transmitter. The receiver resynchronizes within five cycles in a noise-free environment if lock is lost from the transmitter. The maximum throughput is 20Mb/s and the bit error rate (BER) is < $10^{-9}$. The transceiver is realized on a 2.5mm x 1.5mm die and dissipates 375mW.

## 1. INTRODUCTION

Since Carroll and Pecora showed that two chaotic systems can be synchronized in a drive-response configuration [1], some have recently applied chaos to traditional communications systems. Besides traditional RF systems such as direct sequence spread spectrum, other RF modulation schemes have been investigated which might be more amenable towards chaotic modulation. One such RF modulation scheme is known as ultra-wideband (UWB) impulse radio. In this scheme, data is transmitted with electrical impulses, typically <1ns pulse durations, at a pulse repetition frequency (PRF).

There are many benefits to UWB impulse radio, such as immunity to multipath fading and lower power consumption due to absence of linear power amplifiers. Current technology modulates the spacing between consecutive pulses using a pseudo-random noise (PN) sequence [2]. However, long sequences must be used to reduce the magnitude of spectral lines that might interfere with current narrowband RF systems like Global Positioning System (GPS), increasing acquisition time and complexity.

While there has been much progress in system design applying chaos towards communications, there has been an absence of work on practical VLSI implementations of chaotic communications systems. There has been some previous work integrating chaotic functions and oscillators into a single die. For example, a discrete tent map generator was designed in a 1.2um CMOS technology [3]. Integrated CMOS designs for differential chaos shift keying (DCSK) have been proposed by Delgado-Restituto, Rodriguez-Vazquez, and Porra [4]. Elwakil et al. described a system for chaos generation as well as a monolithic implementation [5]. However, a chip containing both a chaotic transmitter and a synchronized receiver for use in data communications has not been shown to date.

This paper introduces one of the first monolithic chaotic transceivers, with both a chaotic oscillator and a drive-response chaotic receiver integrated onto one die. The transceiver was built based upon the system level design developed by Sushchik et al. [6]. The transceiver uses chaotic pulse position modulation, described in the above paper, to send data at an average rate of 20Mb/s. The nonlinear chaotic behavior is accomplished through conversions between voltage and time.

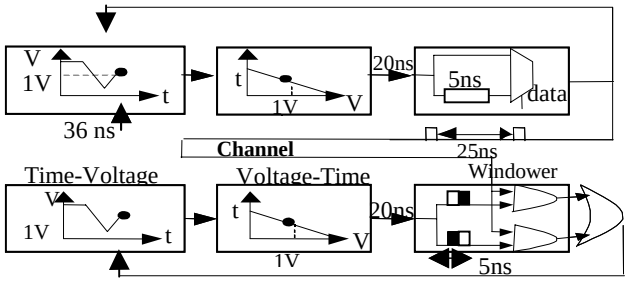


**Figure 1:** Top Level Diagram of Chaotic Transmitter and Drive-Response Receiver

The power spectral density appears uniform, with no apparent spectral lines. Finally, acquisition time to regain transmitter lock takes only 5 cycles in a noise-free environment, faster than conventional PN sequence systems.

## 2. CHIP DESIGN

### 2.1 Transceiver Overview

Figure 1 shows the top-level block diagram of the chaotic transmitter and chaotic receiver. The chaotic transmitter is shown in the top half of Figure 1. The transmitter consists of a chaotic oscillator and a delay modulator. The oscillator generates a pulse train where the time interval between pulses *i-1* and *i*, $t_i$, is related to the preceding time interval, $t_{i-1}$, by a non-linear tent-map function plus a modulation delay, $t_i = f(t_{i-1}) + g(d_i)$, where $f$ is the tent-map function, $g$, is the delay modulation function, and $d_i$ is the $i^{th}$ data bit. The tent map function is realized by a time-to-voltage converter that starts generating a tent waveform on the arrival of pulse *i-1*, and samples the waveform upon the arrival of pulse *i*. The output of this time-to-voltage converter is then input to a voltage-controlled delay element (voltage-to-time converter) to control the position of pulse *i+1*. Data is modulated onto the chaotic pulse stream by adding an additional delay (nominally 5ns) to distinguish between a 1 and a 0.

For example, Figure 1 is annotated with values corresponding to the case where $t_i$ = 36ns, and $d_I$ = 1. The arrival of pulse *i-1* starts the time-to-voltage converter generating the tent-map waveform. 36ns later, the arrival of pulse *i* samples the tent map waveform, resulting in an analog value of 1V. This voltage is in turn used to control the voltage-controlled delay element (voltage-to-time) to give a delay of 20ns. To signal a "1" rather than a zero, an additional 5ns delay is added. Thus, $t_{i+1} = f($36ns$) + g(1)$ = 25ns. This pulse interval, $t_{i+1}$ = 25ns, is then used to determine $t_{i+2}$ in a similar manner.

The receiver implements the same time-to-voltage and voltage-to-time converters as the transmitter, with the same parameter values. The receiver synchronizes with the transmitter by looking for pulses only in a small window where it expects the transmitted pulse to lie. If a pulse is found in the window, the receiver oscillator is locked and uses the pulse interval to predict the window for the next transmitted pulse. If no pulse is detected in the window, synchronization is lost and the receiver listens to all received pulses to resynchronize with the transmitter.

In addition to the transmitter and receiver, a 16-bit asynchronous pseudo-random bit sequence (PRBS) generator and checker are implemented on the die to check BER.

## 2.2 Transmitter Design

### 2.2.1 Voltage-to-Time Converter

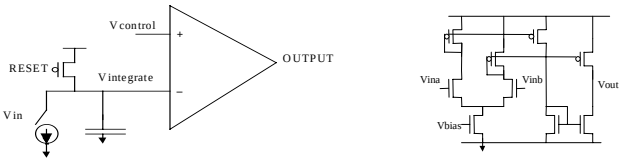


**Figure 2**: Time-to-Voltage Converter, with Opamp Schematic

Figure 2 above shows a schematic of the voltage-controlled delay circuit. The circuit is similar to a single-slope integrating analog-to-digital converter (ADC), except that the output is not quantized by a counter. The circuit integrates a current to generate a voltage ramp and senses when this ramp crosses a trip point with a comparator. The trip point, $V_{control}$ is the sampled voltage from the tent map generator. Before the start of the interval, the ramp voltage, $V_{integrate,}$ is reset to the supply voltage. Then, at the start of the interval, when Vin transitions from low to high, $V_{integrate}$ ramps downward with a slope set by I and C. At the point where $V_{integrate} < V_{control}$, depending on the gain and bandwidth of the opamp, Voutput will switch.

The schematic of the opamp used in the circuit is also shown in Figure 2. The small signal gain is ~22, with a unity gain bandwidth of 485MHz. Also, the opamp must have good common mode dynamic range, as the input common mode may swing from 0.8*Vdd to 0.2*Vdd, which is the dynamic range of the tent map nonlinear generator.

The bandwidth of the opamp is a critical factor in determining the maximum speed of the chaotic oscillator. To obtain a faster bit rate, the discharge rate of the integrator must be increased. However, increasing the discharge rate may force the opamp to operate out of its linear region and the opamp will subsequently slew at the output. Therefore, the input common mode range as well as opamp bandwidth are the inherent speed limitations of this analog chaotic scheme.

### 2.2.2 Tent Map Generator — Time-to-Voltage Converter

The tent map waveform is also generated by integrating current on a capacitor, as shown in Figure 3. Before the start of the interval, the waveform, $V_{tent,}$ is charged to the positive supply. At the start of the interval, switch A closes starting a downward ramp of $V_{tent}$. A fixed delay later, A opens and B closes causing $V_{tent}$ to ramp upward. Finally, at the end of the interval, switch C opens holding the value of the waveform at this point in time on $V_{out}$. Switch C may open at either points (2) or (4), depending upon the time duration between pulses; from this, the nonlinear time intervals between pulses are introduced. Off-chip potentiometers control the values of the current sources on-chip.

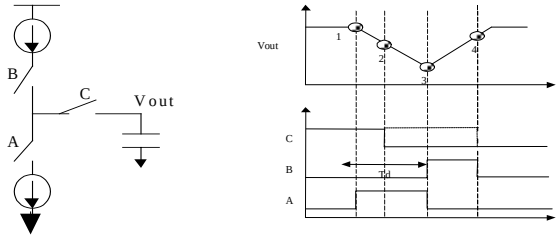


**Figure 3**: Voltage-to-Time Converter: Charge Pump Behavior and Tent Map Example

The dynamics of the chaos depend on both the slope of the integrator in the voltage-controlled delay as well as the slopes of the tent function in the time-to-voltage converter. If the gain of the entire oscillator is less than one, the oscillator will not behave chaotically and the trajectory will converge towards a stable point. At this apex, the system no longer behaves chaotically; the voltage that is obtained will be fed back to produce the same voltage for the next pulse, resulting in a periodic pulse repetition frequency.

### 2.2.3 RS latch structure

To allow the integrators in the time-to-voltage converter and the voltage-controlled delay elements time to reset between pulse intervals the circuits are duplicated as shown in Figure 4. The upper circuit generates even-numbered pulses while the bottom circuit resets. Similarly, the bottom circuit generates odd-numbered pulses while the top circuit resets.

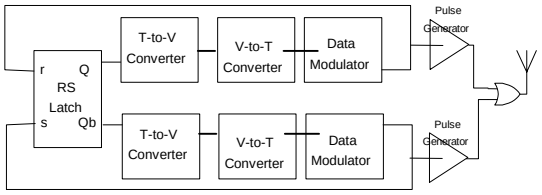


**Figure 4**: Top level of transmitter design, showing multiplexed parallel architecture

## 2.3 Receiver Design

The receiver block diagram, shown in Figure 5, is identical to that of the transmitter except that in place of the data modulator, it includes a windowing circuit. The receiver also includes a state machine that allows it to regain lock with the transmitter after only five pulses in a noise-free channel.

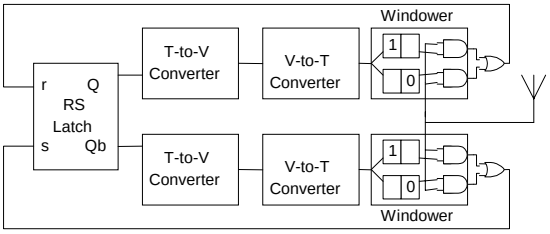


**Figure 5**: Top level of receiver design, showing operation of windowing block

### 2.3.1 Windowing Function

Because the transmitter and receiver implement the same chaotic function, if both are synchronized, the voltage-to-time blocks of the receiver should output pulses at the same points in time as the transmitter. To detect transmitter pulses at these points in time, the receiver windowing logic generates two window pulses: one centered on the expected position of the pulse if the data is a "0", and the other centered on the expected position of the pulse if the data is a "1". The receiver logically ANDs these window pulses with the incoming transmitted pulses. Coincidence results in successful data detection and starts the computation of the next time interval. To allow margin for noise and parameter mismatch, the receiver window is much wider than the duration of the transmitted pulse (5ns vs. 1ns in the present design).

The receiver window is sized to trade off parameter mismatch between transmitter and receiver against noise sensitivity. Making the windows larger gives more tolerance to mismatch, such as current source differences, between the transmitter and receiver. Since the receiver is synchronized to the transmitter, errors do not continuously build in a feedback manner. As long as the amount of inter-die mismatch can be tolerated by properly sized receiver windows, the receiver will remain locked even when the chaotic parameters do not exactly match between transmitter and receiver. However, making the windows too large reduced BER since the receiver becomes more sensitive to impulse noise. Through simulation, it was shown that a window size of 5ns was large enough to cover the 3 sigma (30mV) of opamp offset variation between transceiver on different die, not accounting for impulse noise. Therefore, any parameter mismatch between transmitter and receiver can be compensated for by properly adjusting the value of the modulation delay and the delay and size of the windowing functions.

### 2.3.2 Receiver Synchronization Out of Lock

If no coincidence is detected between a receiver window and a transmitted pulse, the receiver will fall out of synchronization with the transmitter. In this case, the receiver resynchronizes by using incoming pulses to estimate the chaotic state of the transmitter in 5 cycles, as in figure 6.

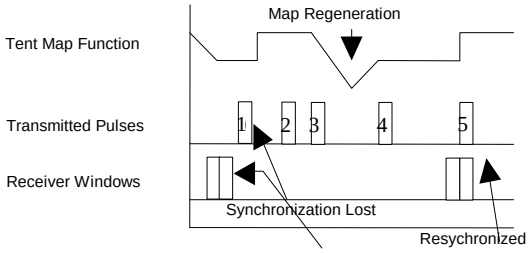


**Figure 6**: Example of receiver losing lock and regaining synchronization

The operation of regaining lock is as follows. Pulse (1) is missed by the receiver. Pulse (2) is used to update a state machine, putting the receiver in a hunting resynchronization mode as opposed to the locked receiver mode. Pulse (3) is used to start regeneration of the tent map, while pulse (4) samples the voltage created by the generator. Notice that the time duration between pulses (3) and (4) create an analog voltage in the receiver which is exactly equivalent to the analog voltage in the transmitter which created the non-linear time duration between pulses (3) and (4) in the first place. Therefore, the receiver has obtained the chaotic state of the transmitter and using the reproduced analog voltage, can predict the location of the next incoming transmitted pulse. Hence, pulse (5) is demodulated and the receiver is again locked.

# 3. EXPERIMENTAL RESULTS

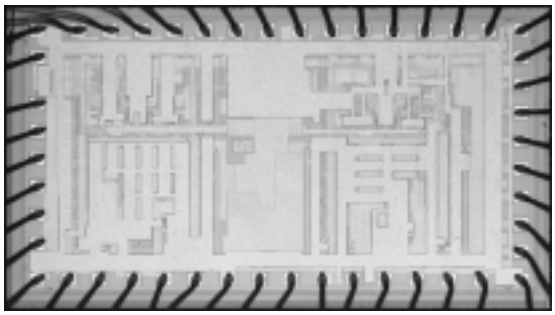

**Figure 7**: Chip photomicrograph.

A prototype chip has been fabricated in National Semiconductor's 0.25um CMOS technology with a nominal 2.5-V supply voltage. The values of the on-chip current sources are controlled by 100k potentiometers on the test board.

**Figure 8**: Dual Tent-Map Generators in Transmitter

Figure 8 shows the dual parallel tent map functions that operate in the transmitter. Each tent map spans approximately 50ns of time and across 2V in magnitude.

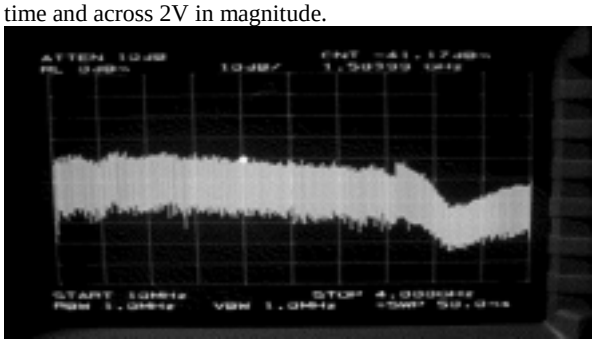

**Figure 9:** Power Spectral Density of Chaotic Pulses of Time Duration < 1ns

Figure 9 above shows the power spectral density of chaotic data impulses between 10MHz and 4GHz, where the pulse width < 1ns. The total average power is spread uniformly in frequency, showing no discernable spectral lines often seen in traditional PN sequence UWB systems.

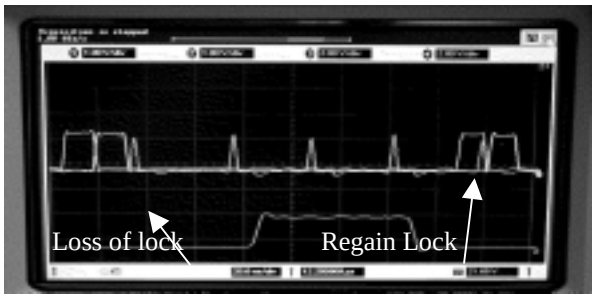


**Figure 10**: Receiver showing loss of lock and regain of synchronization with transmitter

Figure 10 shows the receiver losing synchronization from the transmitter, regain lock and receive data after only 5 transmitted pulses.

BER was measured by using a PRBS at the transmitter, and a PRBS checker at the receiver, where transmission of a single bit error in transmission is detected. BER was measured less than $10^{-9}$, with a very low noise floor (SNR=60dB). Power was substantially larger than expected, due to a mistake in layout, thereby creating a 70 Ohm poly short between power and ground which did not affect circuit operation.

Note that the reported results occur in a noise-free environment, which cannot be assumed in a UWB environment. Therefore, the current implementation of a chaotic impulse transceiver would suffer worse synchronization time as well as higher BER in a real wireless environment. This test chip also does not consider other aspects specific to UWB radio, such as design of a low noise amplifier (LNA), sensitivity to impulse noise, etc. Such problems can be mitigated by more system complexity; i.e. noise sensitivity can be combated by matched filter impulses.

## 4. CONCLUSION

A monolithic chaotic transceiver contains both a transmitter and force-response receiver in a 0.25um CMOS process. This transceiver is based upon a modulation scheme similar to pulse position modulation, for use in UWB impulse radio. The nonlinear tent maps are created on chip using a conversion between voltage and the time domain. A parallel multiplexed architecture of the nonlinear generators is used to ease the design constraints of analog sampling. The receiver is locked to the transmitter by creating windows of where it expects the transmitted pulses to be. These larger windows allow tolerances to account for process mismatch, since jitter does not accumulate in this system. If synchronization is ever lost between the two chaotic oscillators, the receiver will go into a hunt mode to reacquire lock with the transmitter. This sequence takes only 5 cycles in a noise-free environment. A fabricated chip has been designed and tested, with experimental results shown. Future work includes a digital implementation of the non-linear function, which will reduce power consumption and complexity, and increase throughput and robustness.

## 5. REFERENCES


[1] L.M. Pecora, and T.L. Carroll, "Synchronization in chaotic systems," *Phys. Rev. Lett., vol. 64, pp. 821-824, Feb. 1990.*

[2] M.Z. Win and R.A. Scholtz, "Impulse radio: How it works," *IEEE Comm. Letters*, vol. 2, no. 2, pp. 36-38, Feb. 1998.

[3] A. Dornbusch and De Gyvez, J.P. "Chaotic generation of PN sequences: a VLSI Implementation," *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, vol. 5, pp. 454-457, 1999.

[4] M. Delgado-Restituto, et al. "Integrated Circuit Blocks for a DCSK Chaos Radio," *Proceedings of the 1998 IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 473-476, 1998.

[5] A.S. Elwakil, et. al. "A System for Chaos Generation and Its Implementation in Monolithic Form," *Proceedings of the 2000 IEEE International Symposium on Circuits and Systems*, vol. 5, pp. 217-220, 2000.

[6] M. Sushchik, et al., "Chaotic pulse position modulation: A robust method of communicating with chaos," *IEEE Communications Letters*, vol. 4, no. 4, April 2000.