CMOS Implementation of Nonlinear Spectral-Line Timing Recovery in Digital Data-Communication Systems

Un-Ku Moon, *Senior Member, IEEE,* and Gang Huang

Abstract—In many of the digital communication systems where a form of passband modulation scheme is used, carrierless amplitude and phase modulation (CAP) or quadrature and amplitude modulation for example, the signal waveform does not contain a baud-rate spectral line. This paper describes analog and all-digital implementations of timing recovery using the nonlinear spectral-line method. The analog implementation of the timing-recovery integrated circuit was fabricated in 0.9-μm CMOS process and verified to meet all the requirements for a system utilizing the CAP modulation scheme, and initial results of the all-digital implementation confirm an even better performance that is process independent. The 51.84-MHz recovered clock allows the receiver to achieve better than a 10^{-10} bit-error rate (BER).

Index Terms—Carrierless, carrierless amplitude and phase modulation (CAP), CMOS, digital communication, nonlinear spectral line, quadrature and amplitude modulation (QAM), timing recovery.

I. INTRODUCTION

THE CRITICAL function and the necessity of timing recovery in various synchronous pulse-amplitude modulation (PAM) systems is widely discussed and re-emphasized in generality as well as in specific applications [1], [4]–[8], [14], [16]. The authors, Agazzi *et al.* [7], Mueller and Müller [8], Lee and Messerschmitt [2], and Sklar [3] do a nice job of summarizing some of the commonly known timing-recovery methods such as spectral line, threshold crossing, sampled-derivative, early-late gate, minimum mean-square error, maximum-likelihood estimation, wave difference, and baud-rate sampling. However, when a focus is given to the CMOS integrated circuit (IC) implementation of digital communications systems, the most practical implementation seems to be either an oversampling system where a sample-rate conversion technique can be efficiently implemented, or a seemingly more straightforward deductive timing-recovery implementation using the spectral-line method.

The sample-rate conversion technique is attractive in the sense that the implementation is completely digital and robust in nature, but the computation can be too costly for a system where the luxury of oversampling is limited by technology. In the low oversampling applications, the complex computations involved in the high order polynomial curve fitting [9], [10] will require an inefficient amount of hardware for sufficient accuracy, and the implementation of sample-rate conversion technique would typically require multiple parallel finite-impulse response (FIR) filters with continuously changing coefficients [11]. Perhaps practical implementation of the sample-rate conversion technique is limited to applications such as high-bit-rate / asymmetrical digital subcriber line (HDSL/ADSL) systems with enough oversampling, and other systems which may tolerate higher bit-error rate (BER).

In the ATM/LAN and switched digital video (SDV) applications, also commonly referred to as fiber to the curb (FTTC), the need for a high data rate does not allow much oversampling [12]. This is a natural arena for the use of deductive timing-recovery incorporating nonlinear spectral-line method. The nonlinear spectral-line method is able to extract the symbol rate from a wide band data/signal which does not contain the baud-rate spectral line. Since the symbol rate is needed for the purpose of timing-recovery, it is commonly achieved first by applying a nonlinear function which draws out the higher moments of the signal that are periodic at the symbol rate. Then, the signal is further processed through a bandpass filter (BPF) and phase-locked loop (PLL) combination to recover signal timing.

In this paper, the necessary timing-recovery functions are implemented in the analog domain [17], and an alternative all-digital implementation is also presented. This paper expands on the work presented in [17] while adding a more robust all-digital implementation design details. The analog timing-recovery block includes a multiplier for the squaring nonlinear function, a self-tuning BPF, and a PLL using an external voltage controlled crystal oscillator (VCXO). The all-digital implementation similarly processes the signal by taking the output of the analog-to-digital converter that is inherent to the signal path of the receiver itself. The following sections will include a summary of the nonlinear spectral-line method (Section II); differences between quadrature and amplitude modulation (QAM) and carrierless amplitude and phase modulation (CAP) modulation schemes (Section III); the overall structure and sub-block details of the IC implementation (Section IV) including the squarer, the self-tuned BPF, and PLL; all-digital timing-recovery implementation detail (Section V); the simulation method and results (Section VI); and some meaningful measurement results (Section VII).

II. NONLINEAR SPECTRAL-LINE METHOD

The key principle in nonlinear spectral-line timing recovery is that a memoryless nonlinearity $f(\cdot)$ is applied to the incoming signal $s(t)$ so that the expected value of the output $E[r(t)]$ where $r(t) = f(s(t))$, is nonzero and periodic with the symbol period T [2], [5], [6], [14]. This is due to the fact that if we treat the phase of the signal (signals in modulation schemes such as QAM, PAM, and CAP) as a random parameter, the signal represents a cyclostationary process. In a simpler case where $E(s(t))$ is already nonzero and periodic with period T (i.e., cyclostationary), the necessary spectral line at the symbol rate exists and a nonlinearity does not need to be applied. Such a condition qualifies as a *linear* spectral-line method [2].

Typical ways of applying a memoryless nonlinearity to the zero-mean input signal are using a class of absolute value circuits (i.e., rectifiers), a squarer, and even fourth-power circuits [15]. Regardless of the method used, it must elicit the cyclostationary behavior that is contained at the higher moments of the incoming signal. Perhaps the most common and clearly explained implementation is the squarer. The analog IC implementation uses this method, sometimes known as envelope-derived timing recovery [2], [5].

In the process of understanding the functions of the memoryless nonlinearity in the context of timing recovery, it is generally "safe" to dismiss any distinction between different in-phase and quadrature data-transmission systems such as QAM and CAP which can fall under passband PAM [2], and further dismiss any differences between baseband and passband PAM [2], [5]. This is well justified given that some sort of band-limiting prefilter normally exists (typically inherent part of the transmission path), and the output of the nonlinear function, $f(\cdot)$, is filtered by a BPF.

The timing-recovery IC operates on a received passband signal (applies to either QAM and CAP in the context of spectral line)

$$s(t) = \text{Re}\left\{ \sum_{n=-\infty}^{\infty} c_n g(t - nT) e^{j\omega_c t} \right\} \quad (1)$$

where $c_n = a_n + jb_n$ is the complex input data sequence and $g(t)$ is the filtered baseband pulse shape modulated by the carrier ω_c. T is the symbol period. The squared signal $r(t) = s(t)^2$ has a desired timing tone (symbol rate ω_s) whose power is [5]

$$R(\omega)\big|_{(\omega=\omega_s)} = \sum_{k=0}^{N-1} |C(k\Omega)|^2 G(k\Omega) G(\omega_s - k\Omega). \quad (2)$$

In (2), $C(k\Omega) = \sum_{n=0}^{N-1} c_n e^{j\Omega n}$, where $\Omega = \omega_s/N$, is the discrete Fourier transform (DFT) of the complex input data sequence c_n with period N. $G(\omega)$ is the spectrum of $g(t)$.

The DFT is invoked because of the periodic nature of c_n. However, in the limiting case of perfectly random and uncorrelated c_n, $N \to \infty$, $|C(\omega)|^2 \to$ constant, and (2) approaches the form

$$R(\omega)\big|_{(\omega=\omega_s)} = \bar{c}^2 \int_{-\infty}^{\infty} G(\omega) G(\omega_s - \omega) d\omega. \quad (3)$$

From Fig. 1, it is clear that the amount of overlap between $G(\omega)$ and $G(\omega_s - \omega)$ directly impacts the strength of the desired timing
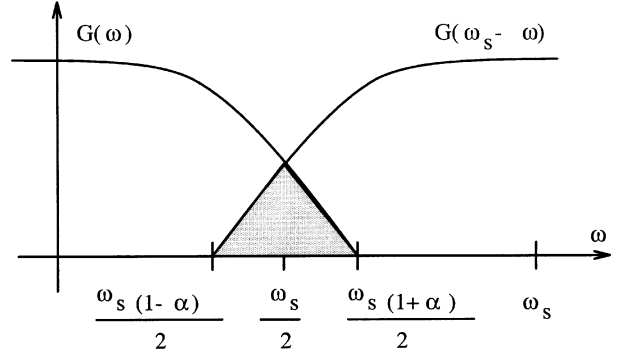

Fig. 1.   Effect of excess bandwidth on symbol tone extraction.

tone. This spectral overlap is a function of the excess bandwidth parameter α, which can vary from 0 to 1 (0 to 100%). Note that in the extreme case of $\alpha = 0$, $G(\omega)$ and $G(\omega_s - \omega)$ are disjoint, and no timing tone is generated. In most applications, signal is transmitted with as high as 100% excess bandwidth [12] and as small as 20%–30%. Even in some extreme cases such as 0% excess bandwidth, absolute value circuits as well as a fourth-power circuits have been shown to work [15].

III. QAM AND CAP

There exists a mild but common difficulty in resolving differences and similarities between CAP and QAM schemes. We will briefly make the distinction here for the benefit of the reader.

A. QAM

QAM is familiar to most and the most common form of mathematical expression is identical to (1). Recalling the equation

$$s(t) = \text{Re}\left\{ \sum_{n=-\infty}^{\infty} c_n g(t - nT) e^{j\omega_c t} \right\}.$$

As once noted, $c_n = a_n + jb_n$ is the complex data sequence, $g(t)$ is the filtered baseband pulse, T is the symbol period, and $e^{j\omega_c t}$ represents modulation to passband by carrier ω_c. The implication in the distinction that is being made here is that the modulation term $e^{j\omega_c t}$ to passband is accomplished by using analog sine and cosine (in-phase and quadrature) waves of the desired carrier frequency.

B. CAP

Simply speaking, CAP is a digital form of passband modulation applied to QAM. Instead of using an analog carrier (sine and cosine), the equivalent modulation to passband is typically implemented digitally using two FIR filters (one each for in-phase and quadrature portions). The output of the FIR filters are summed and transmitted via a higher resolution digital-to-analog converter.

One may regard the CAP modulation scheme as just another implementation of QAM, a part of analog implementation simply shifted into digital domain. We have no objections to that view. The fine difference in the actual signal transmitted is described here, and the receiver needs to know this fine distinction in order to demodulate/decode the data properly.

Due to fixed coefficients of in-phase and quadrature FIR filters, each symbol in the CAP transmitter experiences the same *phase-aligned* carrier modulation. Unlike the QAM where the symbols (serially transmitted back-to-back) are continuously modulated by an analog (does not have to be analog, of course) carrier, each symbol in CAP will experience the same starting phase of the carrier in the time domain. The mathematical distinction is the addition of the multiplier $e^{-j\omega_c nT}$, resulting in

$$s(t) = \mathrm{Re}\left\{ \sum_{n=-\infty}^{\infty} c_n g(t-nT) e^{j\omega_c(t-nT)} \right\}. \qquad (4)$$

If the multiplier is merged with data c_n, it results in the new complex data sequence $\hat{c} = c_n e^{-j\omega_c nT}$. Thus, (4) may be rewritten in the form of QAM like the (1)

$$s(t) = \mathrm{Re}\left\{ \sum_{n=-\infty}^{\infty} \hat{c}_n g(t-nT) e^{j\omega_c t} \right\}. \qquad (5)$$

If a QAM receiver were to demodulate this transmitted CAP signal, the constellation of the demodulated data sequence would rotate according to the additional phasor portion (exponential term) of

$$\hat{c}_n = c_n e^{-j\omega_c nT} = c_n e^{-j2\pi f_c nT} = c_n e^{-j2\pi n(T/T_c)}. \qquad (6)$$

This is the fine difference between CAP and QAM. Note that there are special cases where this difference disappears. One example is when T (symbol period) is equal (or integer multiple) to T_c (carrier period).

Finally, here is one of the many situations where the fine difference may be critical. Consider a case where the symbol rate ($\omega_s = 1/T$) is 2/3 of the carrier frequency ($\omega_c = 1/T_c$), i.e., $T/T_c = 3/2$. The phasor term $e^{-j2\pi n(T/T_c)}$, then, is equal to $e^{-j3\pi n} = e^{-j\pi n}$. Note that the phasor alternates between $+1$ and -1. The net result is that if the CAP signal is assumed QAM by the receiver, every other symbol will be mistaken as the opposite point (opposite side through origin) in the constellation.

IV. ANALOG TIMING-RECOVERY IMPLEMENTATION

The overall structure of the analog IC implementation is shown in Fig. 2. This timing recovery is realized in the continuous-time (except for the inherent sampling nature of the phase detector) domain. To minimize sensitivity to the surrounding noisy environment, all circuits are fully differential wherever possible. As discussed in the Section III, the squaring function is used to apply a memoryless nonlinearity to the incoming signal. The subsequent BPF eliminates a large amount of undesired tones (noise) surrounding the symbol frequency (12.96 MHz). The input to the PLL is a slicer which turns this "noisy tone" into a "jittery clock" (which can have missing or extra edges) at this symbol rate, and the PLL locks onto this jittery input clock and filters out the input jitter using an external VCXO. After the symbol timing is recovered, the jitter-free clock is fed into an application-specific receiver equalizer which processes the incoming CAP signal. The overall timing-recovery system is configured such that one or more of the sub-blocks may be bypassed and piped through external components for comparison and for flexibility in
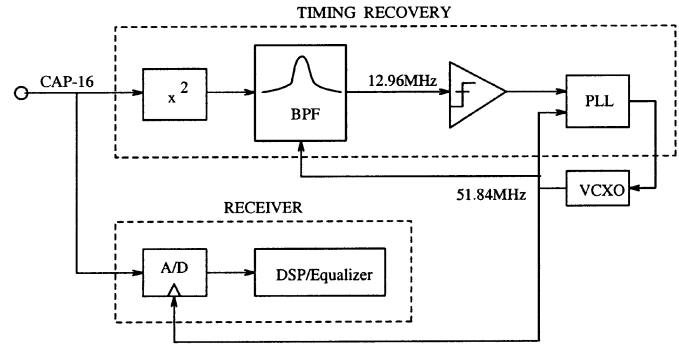

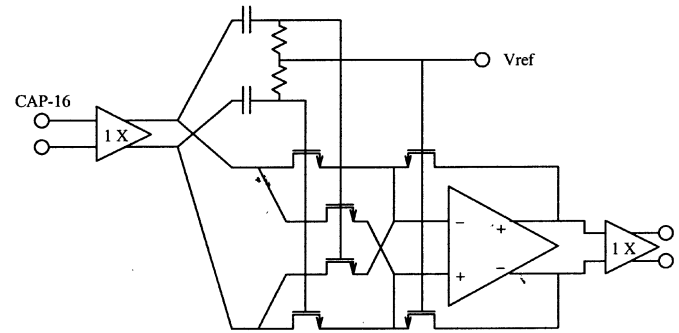Fig. 2. Overall analog timing-recovery structure.

Fig. 3. Squaring function by using a multiplier.

system configuration to accommodate for any unanticipated modifications.

A. Squarer (Multiplier)

The squaring function is implemented with a set of cross-coupled MOSFETs operating in triode [18], as shown in Fig. 3. The input and output buffers are source-follower stages to remove the resistive load requirement from the previous stage, and to have a large enough driving capability at the output. The op-amp symbol represents a differential single-transistor gain stage with simple common-mode feedback. This portion of the design was kept simple to achieve a large bandwidth so that the squaring function would not be hindered by a possible limitation in the signal bandwidth. The overall signal path has more than 100-MHz bandwidth for either inputs (X or Y) according to simulation results. But more importantly, it turns out that the bandwidth on the squaring/multiplying function is limited only by the input buffers and the cross-coupled MOSFETs (current output). The rest of the bandwidth limitation only applies as a spectrum shaping function after the nonlinear function has been applied, which only requires a minimum of 12.96-MHz bandwidth in this application. For the set of cross-coupled MOSFETs alone, even a gigahertz range of input bandwidth can be achieved by placing an enhancement capacitor across the virtual grounds [19]. Lastly, the feedback resistors are also implemented with triode MOSFETs to match the input triode devices for superior gain stability over process and temperature variation. For this system where the magnitude of the extracted symbol tone is commonly very small, stabilized amplitude/gain helps a great deal in maintaining the consistent quality of the symbol tone that is essential for successful timing recovery.
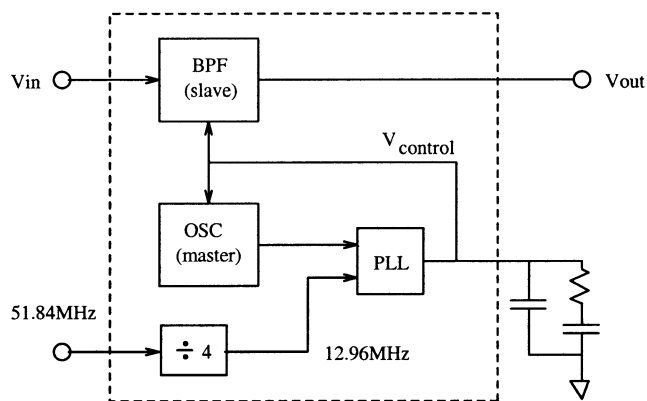
Fig. 4. Self-tuned BPF.

Fig. 5. Biquad configuration for BPF and oscillator.

Fig. 6. Measured BPF frequency response.

B. Self-Tuned BPF

The BPF and the automatic tuning circuit structures are similar to Khoury's G_m/C filter design [20]. For this application, the two blocks used for the voltage-controlled oscillator (VCO) and the BPF are identical as shown in Fig. 4. For the VCO, the signal input is disabled, and for the filter, the positive feedback is disabled. The self-tuned BPF structure is shown in Fig. 5. This should allow a better matching between the master (VCO) and slave (filter) in achieving an accurate automatic-tuning.

The VCO oscillates due to the positive feedback at the signal's zero-crossing [20]. This is done by the limited range in the transconductance of the positive feedback input. At larger signal swings of either polarity, the positive feedback is removed, until the signal swings back near zero. At zero crossing, a net positive feedback is realized. One can conceptually visualize the operation by picturing a pendulum swinging back and forth at its natural frequency (G_m/C) while a small horizontal force is applied in the direction of movement at the instance where the arm is lined up vertically.

Implanted in these VCO and filter biquad blocks is a small range of digital tunability should it be needed for improved tuning accuracy. Fig. 6 displays the BPF's measured frequency response for a few digital settings. For the few dozen devices characterized, the center frequency of the filter at its default setting was within $\pm 5\%$ of 12.96 MHz. The quality factor (Q) and the frequency accuracy (f_c) of the BPF affects the final jitter of the recovered clock. Having a smaller Q requires a better PLL implementation (more noise gets through), and while a larger Q filters out more noise, it requires a more accurate f_c in order to not lose the timing tone. Such tradeoffs must be considered in optimizing performance over process, temperature, and matching tolerances. Reported discrete (manually-tuned) implementations have up to $Q = 50$[14], while it it impractical for IC design. Some prefiltering techniques (before the squarer) can further aid in reducing the jitter [2], [5], [14], but with an added cost of increased complexity that is quite difficult to achieve in analog circuit implementation.

Despite the limited amount of bandpass filtering that can be provided on chip with only a reasonable amount of complexity, it provides an important enhancement to the timing-recovery function. Bypassing this bandpass filtering (the IC design enable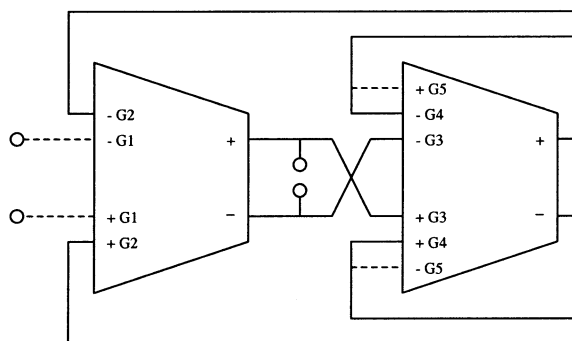d this option), for example, can nearly double the peak-to-peak jitter of the recovered clock, leading to an unacceptably high BER.
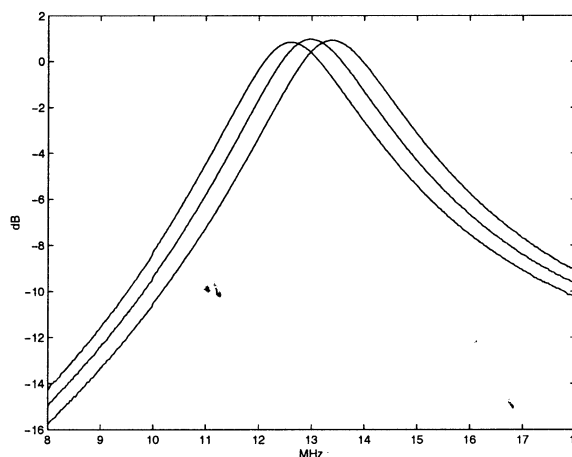
C. PLL

The PLL requires a very narrow-band jitter transfer function. It must filter out the majority of the wide-band jitter still contained in the signal processed through the squarer and the BPF. The unavoidable imperfections of the preceding analog signal processing stages require a VCO with a very narrow tuning range. Use of a VCXO aids in this matter. An oscillator equivalent to Lucent Technologies' 154-type quartz crystal oscillator with a tuning range of ± 100 ppm for 1–4-V input control range (5-V supply) is used to make up the complete PLL. There is still room for significant improvement in this analog implementation by improving the nonlinear function (squarer) and bandpass filtering portions and using an integrated VCO with sufficiently narrow tuning range. The overall performance naturally depends on the combination of these trade-offs in the design of these sub-blocks.

As shown in Fig. 7, the input to the PLL is first squared up by a high-speed slicer (zero-threshold comparator). The phase detector inside the PLL block is a simple XOR gate. It is important to use a nonedge-triggered phase detector because the input to the phase detector is very jittery with multiple edges (or no edge) per symbol period.[1] Simulation results as well as

---

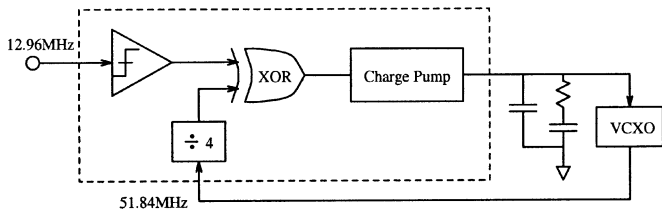[1]What we mean by a nonedge-triggered phase detector is a phase-frequency detector (PFD) which has memory.

Fig. 7. PLL using VCXO.

lab measurements have verified this phenomenon. Use of a common edge-triggered PFD would have failed under these conditions. The buffered XOR output drives a charge pump, which is filtered by external components. A digital PLL implementation similar to [22] was also verified to function properly both in simulation and measurement, but the XOR and charge-pump combination was chosen for its simplicity and smaller chip area.

V. DIGITAL TIMING-RECOVERY IMPLEMENTATION

Having discussed the drawback of the limited Q of the BPF used in the analog implementation, it seems quite natural for one to look for a better way of implementing the same function. Even if external inductors and capacitors are used to achieve a reasonable improvement, the precision of component values is still lacking in comparison to high precision infinite-impulse response (IIR) or FIR filters which operate off a stable sampling period (clock frequency). The digital filters are not necessarily without problems. They can require a large number of high-speed multipliers, which means larger chip area and power consumption. Despite such possible problems, the benefits of stable and precise high-Q filtering of digital filters are very attractive in comparison to a slew of other problems in the analog IC design. In Section IV, we have noted that the process, voltage, and temperature variations of analog IC components create additional challenges in the timing-recovery implementation. For example, because of the center frequency inaccuracy of the BPF, we were not even able to consider the possibility of using an extra high-Q BPF without adding much complication and costly production/testing procedures. In this digital implementation, a new and more efficient approach is taken to implement a high-Q bandpass filter. The main advantage is that it can achieve arbitrarily high order (high Q) filters without using any multipliers and the complexity increases minimally as the order of the filter increases.

A general all-digital timing-recovery system is shown in Fig. 8. It resembles much of the analog implementation as one would presume. The analog-to-digital converter can now be shared with the receiver equalizer, since the necessary timing-recovery signal processing is now performed in the digital domain. After the nonlinearity is applied via an absolute-value circuit, a transversal resonator [23]–[25] is used to filter out a clean symbol frequency tone. A register is then used to sample the filtered tone at the symbol rate (phase-detecting function), which is then passed through a digital loop filter before being fed into the VCO via a simple digital-to-analog conversion.

A. Absolute-Value Operator

The nonlinear device used here is an absolute-value operator. It simplifies the digital implementation and reduces the precision requirement. For CAP applications, the absolute value algorithm performs almost as well as the squarer for a large excess bandwidth system, but the absolute-value function also has an advantage of allowing the extraction of the symbol tone even with the extreme zero excess bandwidth condition. Since the arithmetics are done in 1's complement, the absolute value function is implemented by simply inverting the non-MSB bits if the MSB is "1." The non-MSB bits are then passed on to the following transversal resonator.

B. Transversal Resonator

The filter (resonator) is based on the assumption that after the nonlinear operation on the signal, there is a sine wave component at the symbol frequency. As an example, with $T/4$ sampling, a transversal resonator with a large number of taps can be used to extract the signal. The resonator can be designed such that the coefficients are the samples of a sine wave with frequency $1/T$. A simplest sampled sine wave with frequency 1/4 of the sampling frequency can be represented by the sequence

$$\ldots, -1, 0, 1, 0, -1, 0, 1, 0, -1, 0, 1, 0, -1, \ldots.$$

The transfer function can then be expressed by the Z transform

$$
\begin{aligned}
H(z) &= 1 - z^{-2} + z^{-4} - z^{-6} + z^{-8} - z^{-10} + \cdots \\
&= (1 - z^{-2}) + (z^{-4} - z^{-6}) + \cdots + (z^{-4m} - z^{-4m-2})
\end{aligned}
\tag{7}
$$

where m represents the duration of the impulse response of the filter in number of symbol periods plus one. This is in fact a FIR filter of order $4m + 2$ with nonzero coefficients, and all nonzero coefficients are either 1 or -1. This implies that only additions and subtractions are needed in the implementation. However, the number of additions and subtractions is equal to $2(m + 1)$, and for a large m (to achieve a high Q), the amount of additions and subtractions can still be prohibitively large. We can further simplify by rewriting the transfer function in the following closed form

$$
\begin{aligned}
H(z) &= \frac{Y(z)}{X(z)} \\
&= (1 - z^{-2}) + (z^{-4} - z^{-6}) + \cdots + (z^{-4m} - z^{-4m-2}) \\
&= \frac{(1 - z^{-4(m+1)})}{(1 + z^{-2})}.
\end{aligned}
\tag{8}
$$

The corresponding time-domain difference equation can be written as

$$y(n) + y(n-2) = x(n) - x(n - 4(m+1))$$

or

$$y(n) = x(n) - x(n - 4m - 4) - y(n-2). \tag{9}$$

It becomes obvious that there is an equivalent implementation in the IIR form which requires only two adders and two delay lines that hold $4m + 4$ words and two words. The delay line may be implemented with a first-in-first-out (FIFO), a RAM, or
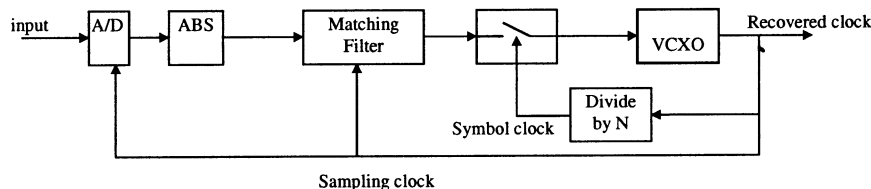
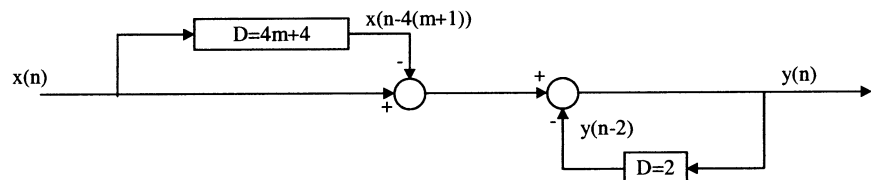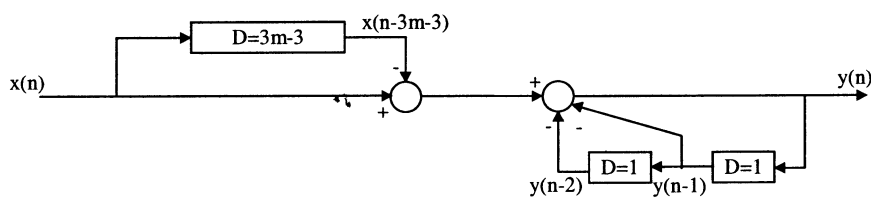Fig. 8. General digital timing-recovery circuit.

Fig. 9. Block diagram of IIR implementation of $H(z)$.

Fig. 10. $T/3$ transversal resonator block diagram.

a parallel shift register, if feasible for a specific design. The IIR block diagram is shown in Fig. 9.

The transversal resonator discussed so far only works if the symbol rate is 1/4 of the sampling rate. Say, for a $T/3$ CAP application, the filter transfer function can be constructed based on the sampled sine wave sequence of

$$\ldots, 1, -1, 0, 1, -1, 0, 1, -1, 0, \ldots$$

which allows us to write the transfer function as

$$H(z) = (1 - z^{-1}) + (z^{-3} - z^{-4}) \cdots + (z^{-3m} - z^{-3m-1})$$
$$= \frac{(1 - z^{-3m-3})}{(1 + z^{-1} + z^{-2})}. \quad (10)$$

The time-domain difference equation is

$$y(n) + y(n - 1) + y(n - 2) = x(n) - x(n - 3m - 3)$$

or

$$y(n) = x(n) - x(n - 3m - 3)$$
$$- y(n - 1) - y(n - 2). \quad (11)$$

Shown in Fig. 10 is the block diagram of the $T/3$ IIR implementation. In general, one can derive the IIR filter coefficients starting from the desired impulse response, convert the impulse response to the FIR transfer function, and obtain the closed form transfer function of the IIR filter.

It should be pointed out that IIR implementation is equivalent to the FIR implementation if the computation in the IIR is executed in exact integers. Therefore, care must be taken so that no bits are rounded and no overflow ever occurs during the operation. The accuracy in the equivalent resonant frequency in the filter is solely determined by the oscillator. Given a VCXO with
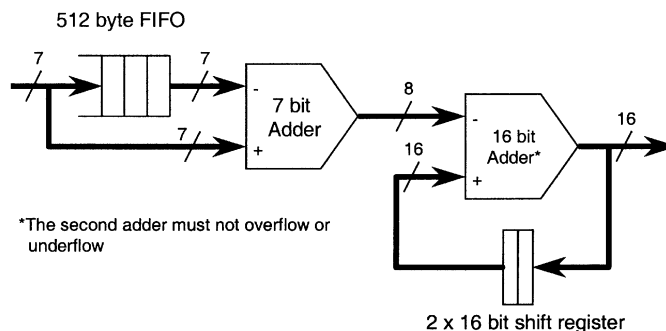

Fig. 11. IIR implementation.

$\pm$ 100 ppm, for example, the filter automatically adjusts itself to the exact symbol frequency once the oscillator is locked.

Fig. 11 shows more details of the IIR filter implementation using 512 byte FIFO and 2×16-bit shift register. The illustration is for the case where 8-bit analog-to-digital converter is used (or 8 MSBs of A/D).

C. Digital PLL (DPLL)

Fig. 12 provides a more detailed overall block diagram of the all-digital implementation. Right-shift and truncation are equivalent to multiplying by a small number, thus a small gain in the *direct* path of the PLL loop is created while an up-down counter is used for the *integral* path. The phase detector function is realized by a simple register which samples the filtered tone at the symbol rate that is coming from the output of the transveral resonator. As illustrated in Fig. 13, the 16-bit data coming out the filter is basically sampled by the register. The net result is a close-to-linear phase detection due to the fact that the output
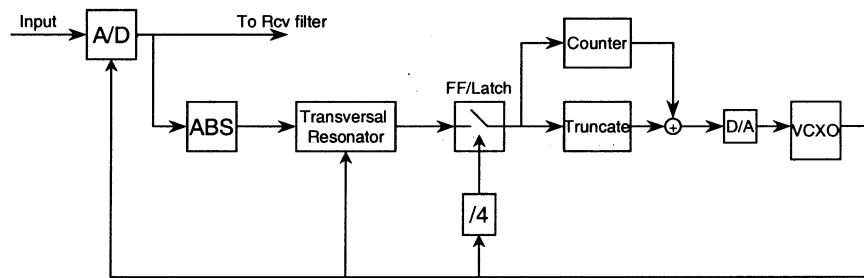
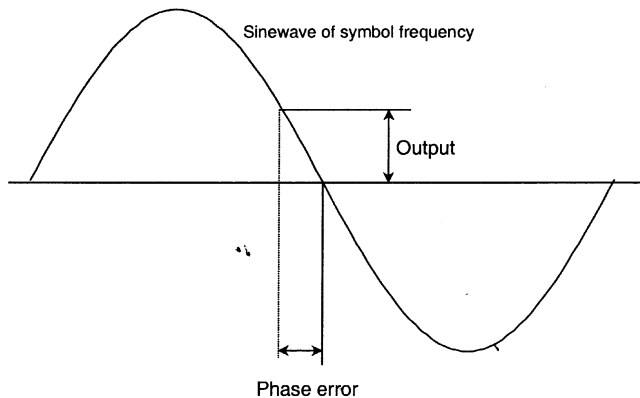Fig. 12. Overall digital timing-recovery implementation.

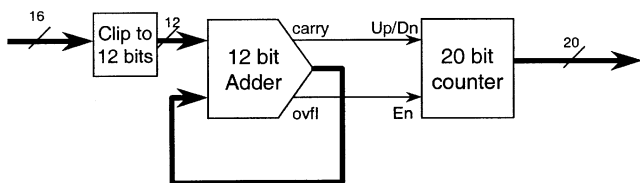Fig. 13. Sampler/register as a nearly-linear phase detector.

Fig. 14. Integral path implementation.

of the filter is approximately a sinewave. This is feasible given the extremely high Q of the all-digital BPF (transversal resonator). Details of the loop filter in the integral path is shown in Fig. 14. Only a clipped 12-bit data is processed. This clipping simply limits the phase detector gain for a large phase error. The 20-bit counter ensures a highly-damped PLL loop dynamics. Fig. 15 shows how the integral and the direct paths are merged before being processed through a digital-to-analog converter which drives an external VCXO. While the output of the integral path (counter) is truncated to 8 bits, the direct path clips the filter output to 14 bits before being truncated to 8 bits. Two 8-bit data are added (and truncated if necessary) before being fed into the D/A. Note that damping characteristics may be modified by using different clipping levels in the two path and number of bits used in various blocks such as the up-down counter.

D. Alternate Approach to Digital-to-Analog Conversion

Even though the use of an 8 to 10-bit level D/A for driving the external VCXO is straightforward and found to be robust in the lab environment, an alternate method may be employed in

this application. Because of the inherent low-pass characteristic that is part of the control signal path of the VCXO, the use of pulse-width modulation as implemented in Fig. 16 is found to be sufficient for the needed jitter requirement. Lab results have shown only a small amount of degredation in comparison to a standard D/A. A simple unsigned 8-bit adder is able to perform this function. Because of the nature of digital overflow, the average output duty cycle is $\text{Input}/256 \cdot 100\%$ (assuming unsigned input). Highest resolution is achieved by running the adder at maximum clock frequency (51.84 MHz).

VI. SIMULATION RESULTS

The simulation results in this section illustrate the intended performance of the analog and all-digital timing-recovery implementation. Giving it a realistic simulation environment, and verifying anticipated results that were reasonable, allowed a greater degree of confidence for achieving successful first silicon. This is especially true in the analog IC implementation.

A. Analog Timing Recovery

For the analog IC version, various simulations are carried out using combination of C-code, MATLAB, and Lucent Technologies' version of SPICE (ADVICE).

The transmitted CAP-16 (16-point constellation) signal was created by using a pseudo-random number generator for the data sequence, and appropriately piping them through a set of FIR filters for in-phase and quadrature paths. The summed output is then passed through an FIR transmission line model. Taking into account the effects of discrete-time FIR transmission line model, the output of an ideal 8-bit digital-to-analog converter is passed through an all-pass L-C smoothing filter. Additive white-Gaussian noise (AWGN) yielding a signal-to-noise ratio (SNR) of 20 dB is also injected for an approximate worst-case condition for an application in mind. A snapshot of the incoming signal spectrum for a 50% excess bandwidth system is shown in Fig. 17.

Using the signal of Fig. 17 as the input to the timing-recovery circuit, SPICE simulation is used to verify the regeneration of the symbol tone by squaring and filtering, as shown in Fig. 18. The center frequency inaccuracy of the BPF is intentionally incorporated into the simulation. Fig. 19 visually illustrates the transformation of the incoming signal through these circuit blocks in the time-domain. Note when the output of the BPF is sliced before entering the phase detector, it may contain irregular edges. For the analog timing recovery, it is necessary
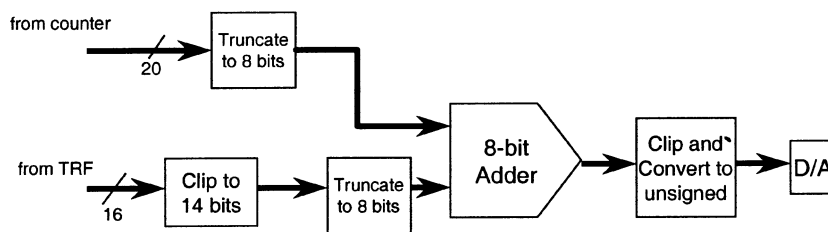
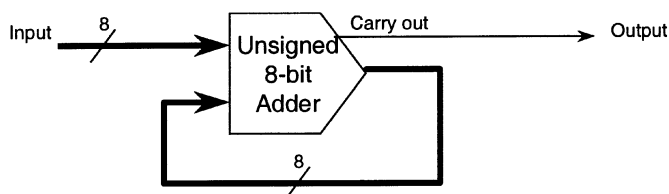Fig. 15.   Loop filter implementation combining integral and direct paths.

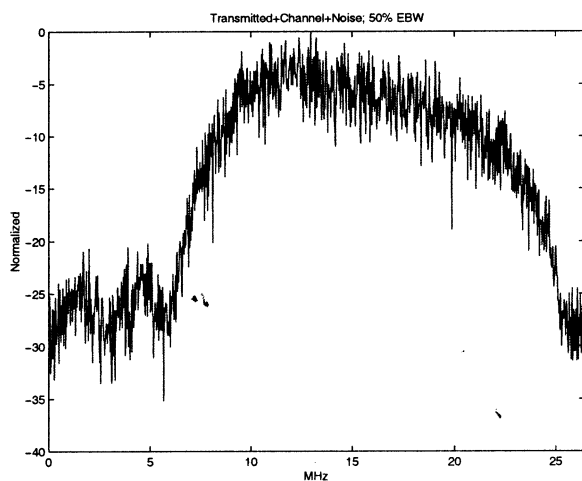Fig. 16.   D/A conversion using pulse-width modulation.

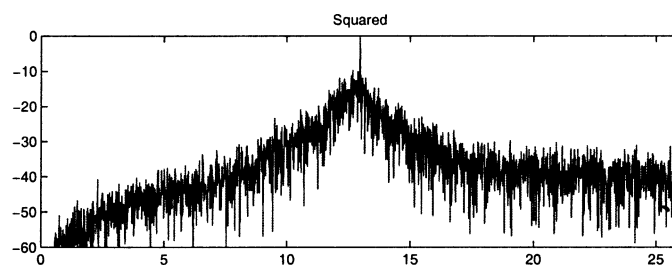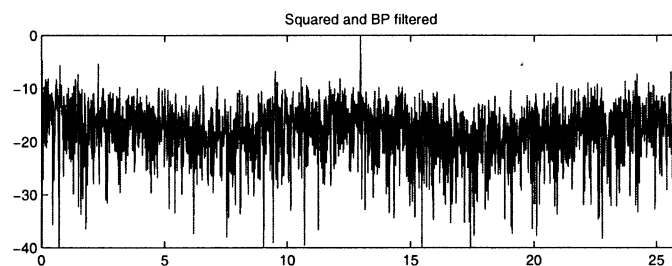Fig. 17.   Simulated spectrum of incoming CAP signal.

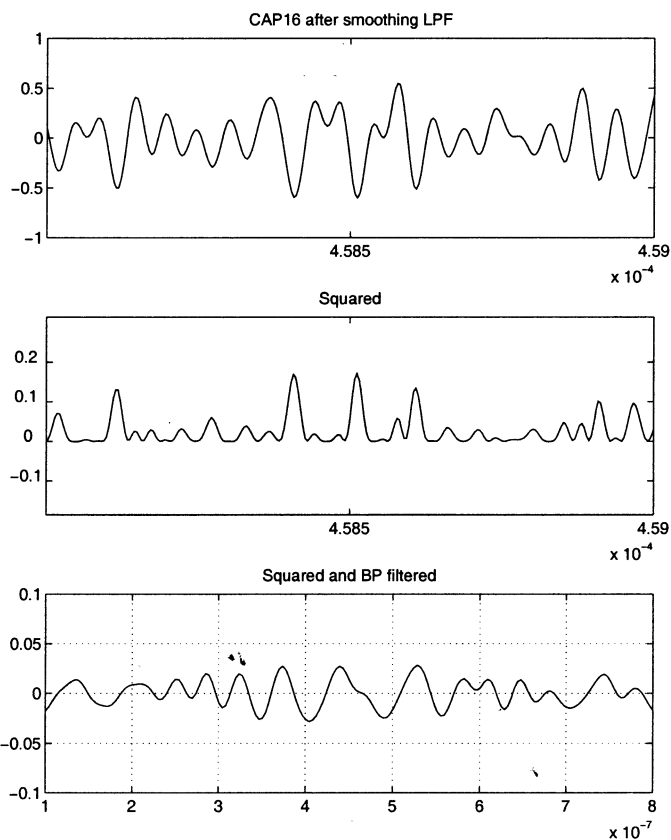Fig. 18.   Simulated spectrum after (a) squarer (b) BPF.

Fig. 19.   Simulated (a) CAP input signal (b) after squaring (c) after BPF.

to adopt a nonedge-triggered phase detector for this reason.[2] If the quality factor (Q) of the BPF can be increased sufficiently (assuming an accurate center frequency for now), the severity of this jitter problem at the input of the phase detector would naturally diminish, and may even allow the use of an edge-triggered PFD. This would be the case in the digital timing-recovery implementation

B. Digital Timing Recovery

Using the signal of Fig. 17 as the input to the digital timing-recovery circuit, MATLAB simulation is used to emulate 8-bit accurate A/D sampling. The simulation results of the symbol tone regeneration by absolute value function (invert non-MSB bits in 1's complement) and filtering via transversal resonator is shown in Fig. 20. Note the precise center frequency of the resonator. Fig. 21 illustrates the visual transformation the sampled time-domain signal experiences. With such precise tone exiting the BPF (transversal resonator), the rest of the timing-recovery functions becomes rudimentary.

[2]To be precise, we mean *memory-less* phase detector is necessary.
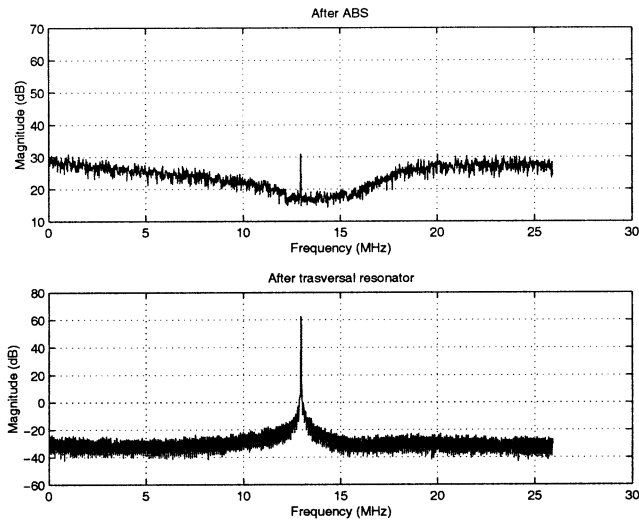
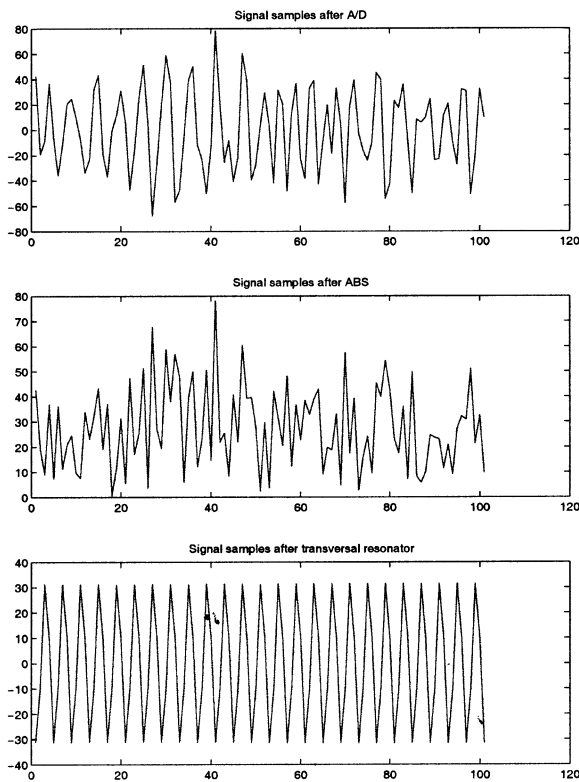Fig. 20.   Simulated digital spectrum (a) after ABS (b) after resonator.

Fig. 22.   Measured spectrum of incoming CAP signal.

Fig. 21.   Simulated (a) sampled input signal (b) after ABS (c) after resonator.
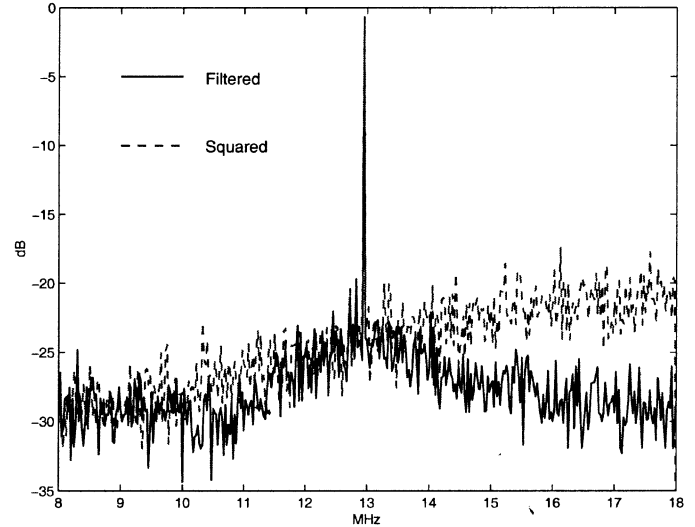
Fig. 23.   Measured spectrum after squarer and BPF.

VII. MEASUREMENT RESULTS

Presented in the following are the measurement results, verifying the intended and anticipated performance of the timing recovery. The simulation results prior to fabrication have shown sufficient agreement with the measurements.

The measured spectrum of the incoming signal for a 50% excess bandwidth system is shown in Fig. 22. An approximate amount crosstalk is also added in the link, which reduces the equivalent input SNR to about 24 dB (calculated/estimated from the channel measurements). The channel used in the measurement is a bundle of 100-m long category-5 twisted-pair with a second twisted-pair (worst-case pair) generating crosstalk.
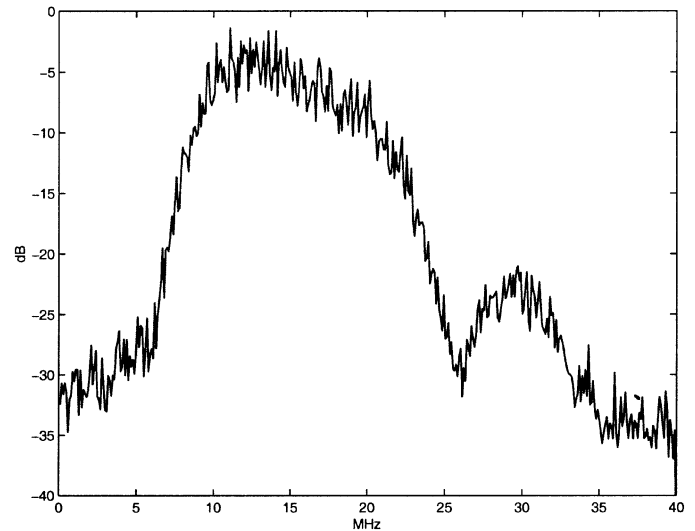
In the analog timing-recovery IC, the lab measurements displaying the regeneration of the symbol tone by squaring and filtering is shown in Fig. 23. With the help of a narrowly tunable VCXO, the measured jitter transfer function of the PLL demonstrates about 500 Hz -3 dB corner frequency. Fig. 24 display this measured PLL transfer function and its measured output jitter spectrum, which follows a similar shape over frequency. Shown in Figs. 25 and 26 are the recovered clock used by a CAP-16 receive equalizer and the constellation of the demodulated/received data that was sent over a maximum length (worst-case) test-link. The recovered clock, measuring about 1.4 ns peak-to-peak absolute jitter, meets the BER requirement of 10^{-10} for an application such as [12]. The analog timing-recovery prototype IC is fabricated in a 0.9-μm CMOS process, and the die photograph of the timing-recovery portion taking up an area of 0.9 mm $\times$ 4.5 mm is shown in Fig. 27.

The analog implementation is already being used in a larger IC system incorporating all the necessary blocks. However, the all-digital implementation is expected to work for greater
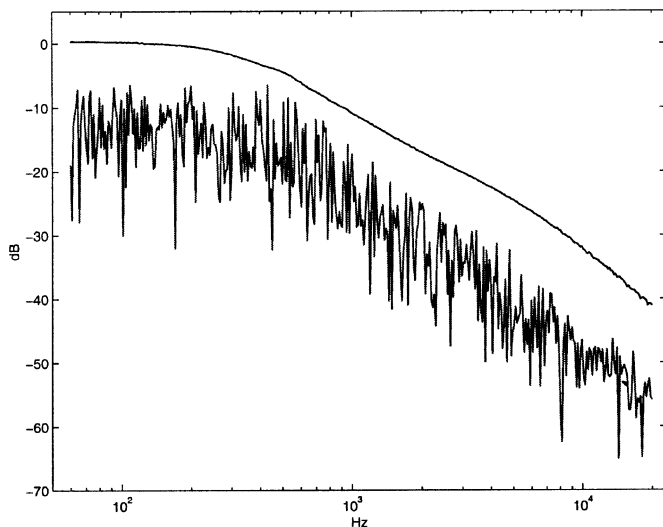
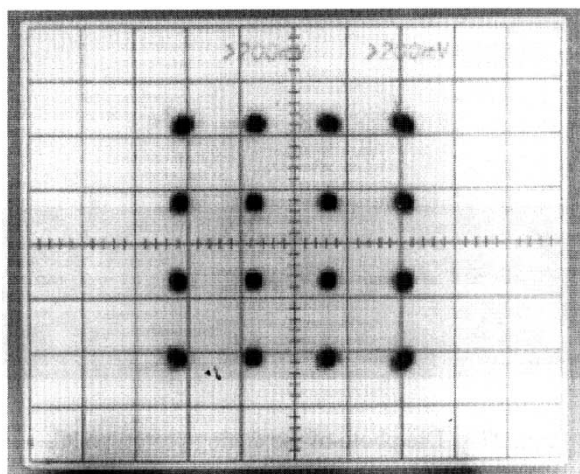Fig. 24. Measured jitter transfer function and jitter spectrum.

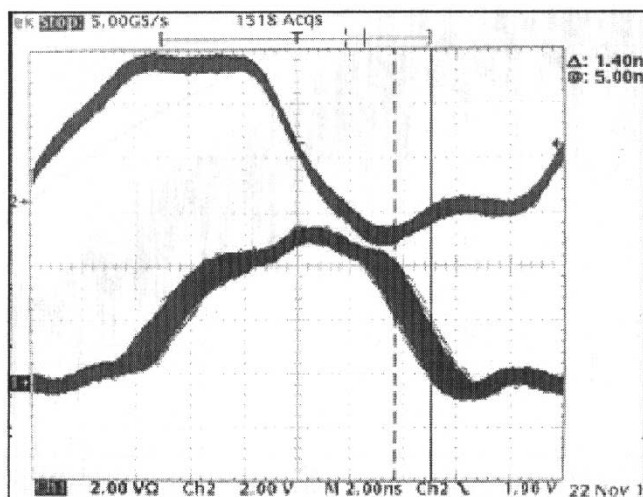Fig. 25. Measured received CAP-16 constellation.

Fig. 26. Measured recovered clock at sample rate (51.84 MHz).

transmission lengths and for reduced excess bandwidth while yielding better jitter and BER performance. Initial measurements of the all-digital implementation under the same testing
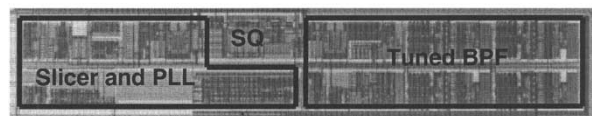

Fig. 27. Die photo of the analog prototype IC.

conditions have displayed about 1 ns peak-to-peak jitter in the recovered clock. When a standard D/A (not pulse-width modulated as discussed in Section V.D) is used to drive the VCXO, the peak-to-peak jitter (directly correlated to the system performance) improved further.

VIII. CONCLUSION

Timing-recovery implementations using nonlinear spectral-line method was presented. An all-analog implementation in 0.9-μm CMOS technology successfully demonstrates the required performance for applications using Carrierless AM/PM [12]. Justification for the IC's applicability in high-speed digital communication systems using synchronous multi-level pulse amplitude modulation scheme has been summarized. The three key functional blocks in the analog timing-recovery structure, squarer, BPF, and PLL, were described, and design tradeoffs were discussed. Some ways of improving the analog timing-recovery process such as prefiltering before the squarer, and higher Q with more accurate f_c of the bandpass post-filter have been mentioned and referenced. The initial results of an efficient all-digital implementation have demonstrated an even better performance that is inherently robust. Digital implementation is free from the environment and process fluctuations due to the scalable and consistently reproducible nature of all-digital ICs. The specifics of reduced complexity and area efficient implementation of transversal resonator (BPF) and digital PLL was presented.

REFERENCES

[1] R. D. Gitlin, J. F. Hayes, and S. B. Weinstein, *Data Communications Principles*. New York: Plenum, 1992.

[2] E. A. Lee and D. G. Messerschmitt, *Digital Communication*, 2nd ed. Boston, MA: Kluwer, 1994.

[3] B. Sklar, *Digital Communications, Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[4] N. A. D'Andrea and U. Mengali, "A simulations study of clock recovery in QPSK and 9QPRS systems," *IEEE Trans. Comm.*, vol. COM-33, pp. 1139–1142, Oct.r 1985.

[5] R. D. Gitlin and J. F. Hayes, "Timing recovery and scramblers in data transmission," *Bell Syst. Tech. J.*, vol. 54, no. 3, pp. 569–593, 1975.

[6] W. R. Bennett, "Statistics of regenerative digital transmission," *Bell Syst. Tech. J.*, vol. 37, no. 11, pp. 1501–1542, 1958.

[7] O. Agazzi, C. J. Tzeng, D. G. Messerschmitt, and D. A. Hodges, "Timing recovery in digital subscriber loops," *IEEE Trans. Comm.*, vol. COM-33, pp. 558–569, June 1985.

[8] K. H. Mueller and M. Müller, "Timing recovery in digital synchronous data receivers," *IEEE Trans. Comm.*, vol. COM-24, pp. 516–531, May 1976.

 [9] G. Liu and C. Wei, "A new variable fractional sample delay filter with nonlinear interpolation," *IEEE Trans. Circuits Syst. II*, vol. 39, pp. 123–126, Feb. 1992.

[10] U. Moon and B. Song, "Background digital calibration techniques for pipelined ADCs," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 102–109, Feb. 1997.

[11] C. W. Farrow, "A continuously variable digital delay element," in *IEEE Int. Symp. Circuits Syst.*, vol. 3, June 1988, pp. 2641–2645.

[12] G. H. Im, D. D. Harman, G. Huang, A. V. Mandzik, M. H. Nguyen, and J. J. Werner, "51.84 Mb/s 16-CAP ATM LAN standard," *IEEE J. Select. Areas Comm.*, vol. 13, pp. 620–632, May 1995.

[13] Y. Takasaki, "Timing extraction in baseband pulse transmission," *IEEE Trans. Comm.*, vol. COM-20, pp. 877–884, Oct. 1972.

[14] L. E. Franks and J. P. Bubrouski, "Statistical properties of timing jitter in a PAM timing-recovery scheme," *IEEE Trans. Comm.*, vol. COM-22, pp. 913–920, July 1974.

[15] J. E. Mazo, "Jitter comparison of tones generated by squaring and by fourth-power circuits," *Bell Syst. Tech. J.*, vol. 57, no. 5/6, pp. 1489–1498, 1978.

[16] R. D. Gitlin and J. Salz, "Timing recovery in PAM systems," *Bell Syst. Tech. J.*, vol. 50, no. 5/6, pp. 1645–1669, 1971.

[17] U. Moon, A. Mastrocola, J. Alsayegh, and S. Werner, "Timing recovery in CMOS using nonlinear spectral-line method," in *Proc. IEEE Custom Int. Circuits Conf. (CICC)*, May 1996, pp. 13–16.

[18] B.-S. Song, "CMOS RF circuits for data communications applications," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 310–317, Apr. 1986.

[19] J. Crols and M. S. J. Steyaert, "A 1.5 GHz highly linear CMOS down-conversion mixer," *IEEE J. Solid-State Circuits*, vol. 30, pp. 736–742, July 1995.

[20] J. Khoury, "Design of a 15-MHz CMOS continuous-time filter with on-chip tuning," *IEEE J. Solid-State Circuits*, vol. 26, pp. 1988–1997, Dec. 1991.

[21] J. Scott, R. Starke, R. Ramachandran, D. Pietruszynski, S. Bell, K. Mc-Clellan, and K. Thompson, "A 16 Mb/s data detector and timing-recovery circuit for token ring LAN," in *Dig. Tech. Papers ISSCC'89*, Feb. 1989, pp. 150–151.

[22] J. F. Ewen, A. X. Widmer, M. Soyuer, K. R. Wrenner, B. Parker, and H. A. Ainspan, "Single-chip 1062 Mbaud CMOS transceiver for serial data communication," in *Dig. Tech. Papers ISSCC'95*, Feb. 1995, pp. 32–33.

[23] P. A. Lynn, "Online digital filters for biological signals: Some fast designs for a small computer," *Med. Biol. Eng. Comput.*, vol. 15, pp. 534–540, 1977.

[24] N. R. Malik, "Microcomputer realization of Lynn's fast digital-filtering design," *Med. Biol. Eng. Comput.*, vol. 18, pp. 632–642, 1980.

[25] P. A. Lynn, "Transversal resonator digital filters: Fast and flexible online processors for biological signal," *Med. Biol. Eng. Comput.*, vol. 21, pp. 718–730, 1983.

Un-Ku Moon (S'92–M'94–SM'99) received the B.S. degree from University of Washington, Seattle, the M.Eng. degree from Cornell University, Ithaca, New York, and the Ph.D. degree from University of Illinois, Urbana-Champaign, all in electrical engineering, in 1987, 1989, and 1994, respectively.

From February 1988 to August 1989, he was a Member of Technical Staff at AT&T Bell Laboratories, Reading, PA. During his stay at the University of Illinois, Urbana-Champaign, he taught a microelectronics course from August 1992 to December 1993. From February 1994 to January 1998, he was a Member of Technical Staff at Lucent Technologies Bell Laboratories, Allentown, PA. Since January 1998, he has been with Oregon State University, Corvallis. His interest has been in the area of analog and mixed analog-digital integrated circuits. His past work includes highly linear and tunable continuous-time filters, telecommunication circuits including timing recovery and analog-to-digital converters, and switched-capacitor circuits.

Prof. Moon is a recepient of the National Science Foundation CAREER Award in 2002, and the Engelbrecht Young Faculty Award from Oregon State University College of Engineering in 2002. He has been an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: ANALOG AND DIGITAL SIGNAL PROCESSING since January 2003, now called IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS. He also serves as a member of the IEEE Custom Integrated Circuits Conference Technical Program Committee and Analog Signal Processing Program Comittee of the IEEE International Symposium on Circuits and Systems.

G. Huang received the B.S.E.E. (with the highest distinction), and the M.S.E.E. degrees from the University of Iowa, Iowa City, in 1983 and 1985, respectively.

From 1984 to 1987, he was a design engineer at Unisys Corporation, where he worked on various VLSI designs for communications equipment. In 1987, he joined AT&T Bell Laboratories, Holmdel, NJ, as a Member of Technical Staff. He is currently with the Advanced Communications Technologies organization of Bell Laboratories, Holmdel, NJ. His present work includes signal processing for wireless systems, channel modeling, and neural networks.