An Analog VLSI Low-Power Envelope Periodicity Detector

Hisham Abdalla, *Member, IEEE,* and Timothy K. Horiuchi, *Member, IEEE*

***Abstract*—The low-power, low-cost detection of voices or engine rumble is a desirable function in many different applications. Typical approaches involving frequency-domain computation are quite computationally intensive and require a significant power budget. In an effort to construct a very low-power detector capable of acting as a wake-up signal for other systems, we have designed a low-power (less than 1.8-μW) subthreshold analog very large-scale integration circuit that detects periodicity in the time-domain envelope of the acoustic signal. The circuit was fabricated in a commercially available 2-poly 1.5-μm CMOS process and occupies an area of about 0.242 mm^2.**

***Index Terms*—Interspike intervals, periodicity detection, smart sensors, subthreshold, voice detection.**

I. Introduction

VOICES and engines typically have a strong harmonic acoustic signature that drives most frequency-domain voice and engine detection algorithms. In smart sensor networks where power consumption is a major constraint [1], having an ultra-low-power "intelligent" technique of turning on the sensor to check for changes in the surrounding environment could dramatically reduce power consumption. While low-power cochlea chips with inter-frequency comparisons [2] can be used to detect unknown harmonic signals, a dedicated time-domain approach would likely result in even lower power consumption. Recently, Goldberg *et al.* demonstrated a low-power time-domain approach based on autocorrelation [3]. In our system, however, we use a time-domain-based approach that utilizes the observation that the periodicity of the time-domain envelope occurs at the fundamental frequency [4]–[6]. By applying a peak detector and measuring the period of the envelope peaks, a sufficient number of consecutively *matching periods* is interpreted as the presence of a periodic signal of interest. By limiting the range of frequencies that are acceptable for the periodicity, detection specific to human voices or other periodic signals is possible. This *algorithm* rejects impulse events, wind noise, and can be implemented to be intensity independent. Clearly, as this type of algorithm will trigger on pure tones, repetitive impulses, or non-voice or non-vehicular harmonic sounds in the right range, this is best used as a wake-up signal for more sophisticated algorithms or when false positives are acceptable. While the primary focus of this work is the design and testing of our very large-scale integration (VLSI) circuit, we consider and discuss the effects of signal-to-noise ratio (SNR), bandwidth, and other parameters on this algorithm.

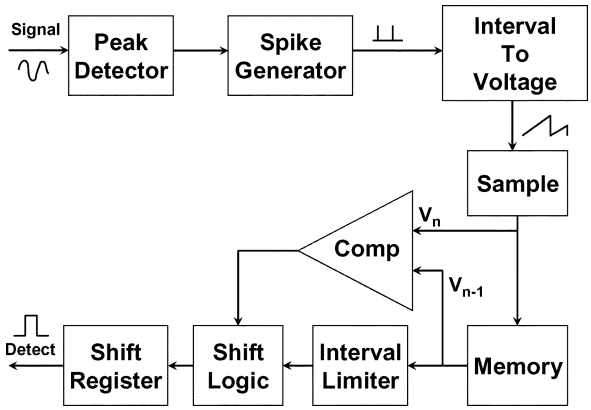

Fig. 1. System block diagram of the peak-periodicity detection chip.

II. System

Fig. 1 shows the block diagram of the circuit. A microphone signal, externally amplified and lowpass filtered, is sent into the chip to a peak-detector/spike-generator circuit. The "spike generator" produces a voltage spike at the peaks of the signal envelope. The "interval-to-voltage" stage linearly converts (via a voltage ramp) the interspike intervals into a voltage. This interval voltage is sampled (V_n) and compared ("comp" block) with the previous interval (V_{n-1}) that is stored in "memory." The sampled interval is subsequently transferred to memory in the period between spikes. Because we are only interested in a limited range of frequencies for a particular application, a "valid interval" signal is generated by the "interval limiter" circuit. Consecutive valid intervals that match generate a "hit" pulse that is used to advance a shift-register-based counter. Output taps on the shift register indicate when three consecutive hits and five consecutive hits have occurred. When non-matching intervals are encountered, the counter is reset. In the experimental data we show, the *rate* of three consecutive hits or five consecutive hits above a threshold rate is interpreted to be a detection event, however, a single occurrence of three consecutive hits or five

Manuscript received July 28, 2004; revised December 1, 2004. This work was supported by the Defense Advanced Research Projects Agency under Contract "Air Coupled Microsensors-0001400C0315." This paper was recommended by Associate Editor T. S. Lande.

H. Abdalla is with the Department of Electrical and Computer Engineering and Institute for Systems Research, University of Maryland, College Park, MD 20742 USA (e-mail: hisham@glue.umd.edu).

T. K. Horiuchi is with the Department of Electrical and Computer Engineering and the Institute for Systems Research at the University of Maryland, College Park, MD 20742 USA, and also with the Neuroscience and Cognitive Sciences Program University of Maryland, College Park, MD 20742 USA (e-mail: timmer@isr.umd.edu).

Digital Object Identifier 10.1109/TCSI.2005.852203

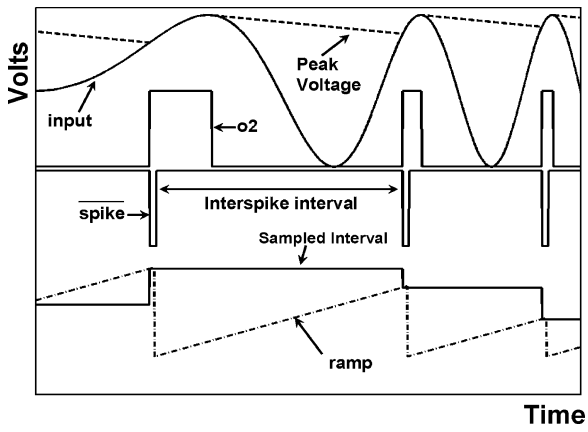

Fig. 2. Timing diagram of the first four stages: the *peak voltage* (dotted line) follows the peaks of the signal. $o2$ is an internal signal of the peak detector. The rising edge of $o2$ marks the onset of a peak in the envelope and is used to generate a spike. The *ramp* (dotted line) converts the *interspike interval* into a voltage. S*ampled interval* is the voltage representation of the current interspike interval. *ramp* is then reset to encode a new interval.

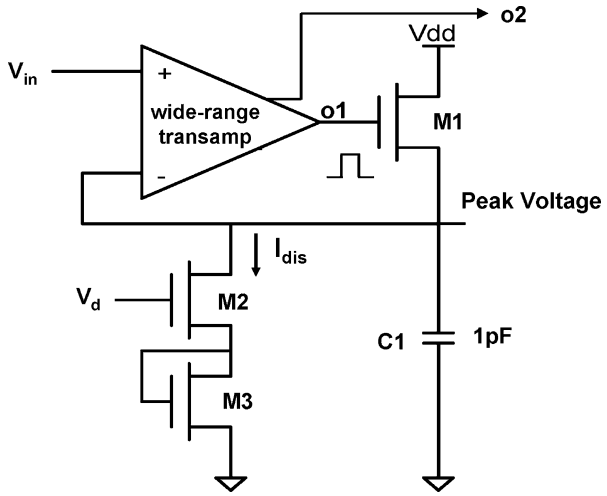

Fig. 3. Peak detector with constant current discharging controlled by V_d. The schematic of the wide-range transamp is shown in Fig. 4. The transamp has two similar outputs $o1$ and $o2$. The $o2$ output is used in the edge detector circuit in Fig. 5.

consecutive hits can also be used. Not shown in the block diagram is the "timing signals" stage which is responsible for generating the signals necessary to sequence the sampling, voltage conversion and decision-making processes following each peak in the signal. Fig. 2 shows the timing diagram for the first four stages.

We acknowledge that the falling edge of $o2$ is the obvious edge to use as an indication of the absolute time of the signal's peak; our goal, however, is to estimate the period of the envelope which depends on the relative timing of the spikes (interspike interval). Our SPICE simulations showed that during the time that the peak voltage tracked the input; it was possible to get oscillations on the digital pulse $o2$. We also note that at the onset of the tracking period, the input has its maximum slope (relative to the rest of the tracking period). Therefore, we designed M1 (Fig. 3) to have a large W/L and we chose to use the rising edge of $o2$ where the charging signal is strongest and $o2$ was the most reliable for producing a sharp transition. Although we

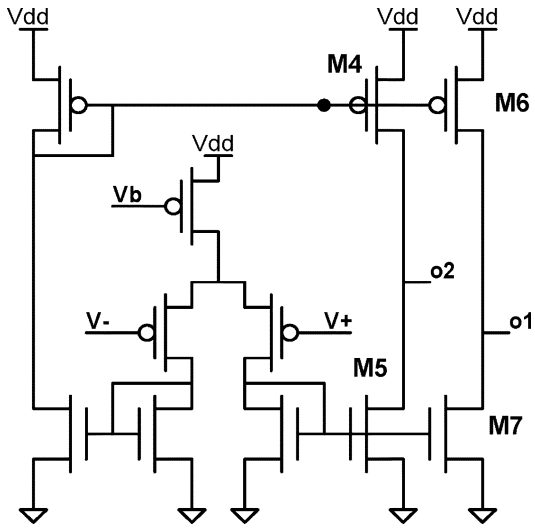

Fig. 4. Dual-output wide-range transconductance amplifier used to generate digital pulses at peaks in the signal's envelope. $o2$ has a larger transconductance than $o1$ because transistors M4/M5 have a larger W/L ratio than M6/M7.

are not truly measuring the interpeak interval, for clean periodic signals we reliably obtain the same period measurement.

III. Circuits

In an effort to reduce power and to match the longer time intervals of interest in the audio frequency range using integrated capacitors, most of the transistors are biased in the subthreshold region of operation where the saturation drain current equation is exponential and the drain–source saturation voltage is about 100 mV. The drain current I_d is given by

$$I_d = I_0 e^{\frac{\kappa V_{\rm gb} - V_{\rm sb}}{V_T}} \tag{1}$$

where I_o is the saturation current, κ is the body effect, V_T is the thermal voltage, $V_{\rm gb}$ is the gate-to-bulk voltage, and $V_{\rm sb}$ is the source-to-bulk voltage [7]. All circuits described below were fabricated in a commercially available 2-poly 1.5-μm CMOS process. The chip occupies an area of about 0.242 mm^2 (read-out circuitry not included) and was operated at a supply voltage (V_{dd}) of 3 V and consumes less than 1.8 μW.

A. Peak Detector

The first stage of our circuit is the peak detector shown in Fig. 3. As the capacitor's voltage drops below the input voltage $V_{\rm in}$, the wide-range transconductance amplifier (transamp) [8] turns M1 on and C1 is charged to $V_{\rm in}$.

When the input drops below the capacitor's voltage, M1 is turned off and the capacitor is linearly discharged by the current source formed by transistors M2/M3. The discharging current $I_{\rm dis}$ is given by the following relation:

$$I_{\rm dis} = I_0 e^{\left(\frac{\kappa^2}{1+\kappa}\right)\frac{V_d}{V_T}}. \tag{2}$$

For proper operation of the peak detector, the discharge rate should be small enough to prevent the peak detector from tracking the input. The fixed linear discharge of C1 makes the chip's performance amplitude-dependent. The discharging current can, however, be made adaptive by externally controlling V_d. Fig. 4 shows the wide-range transconductance amplifier

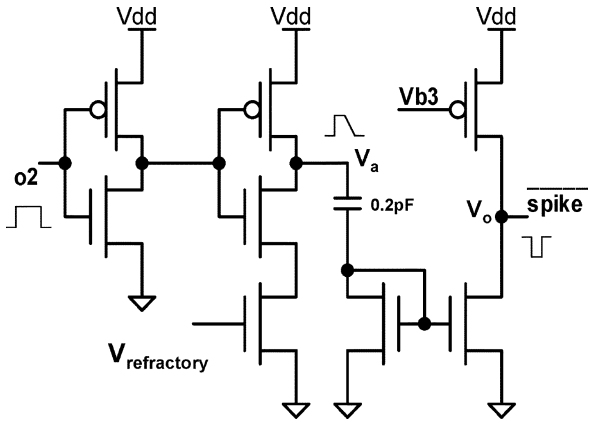

Fig. 5. Edge detector/spike generator for generating spikes on the rising edges of $o2$. V_a must return to its low state before another spike can be generated resulting in a refractory period.

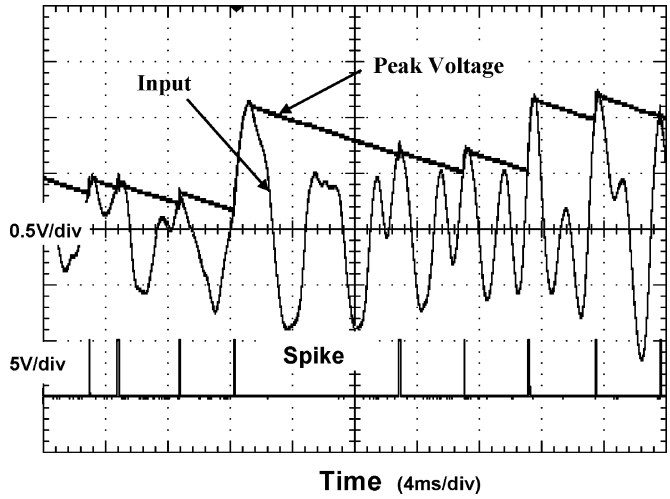

Fig. 6. Oscilloscope trace shows the peak voltage and the spikes that correspond to the peaks of the envelope of the input signal.

used in the peak detector. Since M1 is used as a source follower, the output $o1$ will not, in general, swing to V_{dd} because it will stop when the peak value is attained on the amplifier's input. The second output $o2$ is designed to hit the rail to provide a digital signal when the peak detector is charging the capacitor. Transistors M6/M7 have a W/L ratio of $4\ \mu\text{m}/10.4\ \mu\text{m}$ while that for M4/M5 is $10.4\ \mu\text{m}/10.4\ \mu\text{m}$. The second output $o2$, therefore, has a larger transconductance and will swing from rail to rail effectively generating a voltage digital pulse. Fig. 2 illustrates the simulated response of the peak detector to a sinusoidal input, the transamp's outputs $o1$ and $o2$ will be high while the peak voltage tracks the input.

B. Spike Generator

The rising edge of the peak detector's output $o2$ marks the onset of the peak of the envelope. This edge will be used to generate a voltage spike. The edge detector circuit shown in Fig. 5 is used as our spike generator. On the rising edge of $o2$, the voltage V_a swings from ground to V_{dd}, the capacitor sends a pulse of current down the current mirror pulling V_o low, thus generating a voltage spike. The width of the spike can be controlled by V_{b3} and was set to be approximately 270 μs. In order to generate a second spike, V_a must first return to its low state before which the mirrored capacitive current is too small to pull V_o low. Therefore, similar to biological neurons, the circuit has a *refractory period* T_{refr}; after a spike has been generated, another spike cannot be generated for a certain period of time [9]. In our circuit, after a spike has been generated, $V_{\text{refractory}}$ controls how quickly V_a returns to its low state thereby controlling the refractory period. We can think of the rising edges of $o2$ as a train of input spikes. If a second input spike arrives before V_a has returned to its low state, V_a will be pulled up to V_{dd}, resetting the "refractory timer." Therefore, after a spike has been generated, if the inter-peak intervals are less than the refractory period, this circuit will remain in its "refractory state" and no longer generate spikes. We note that the width of the spike pulse itself ($\cong$0.3 ms) imposes a lower bound on the refractory period. The choice of a particular value for T_{refr} is application dependent and must be less than the smallest period (highest frequency) of interest. Fig. 6 shows the peak voltage and the spikes that correspond to the peaks of the envelope of the input signal.

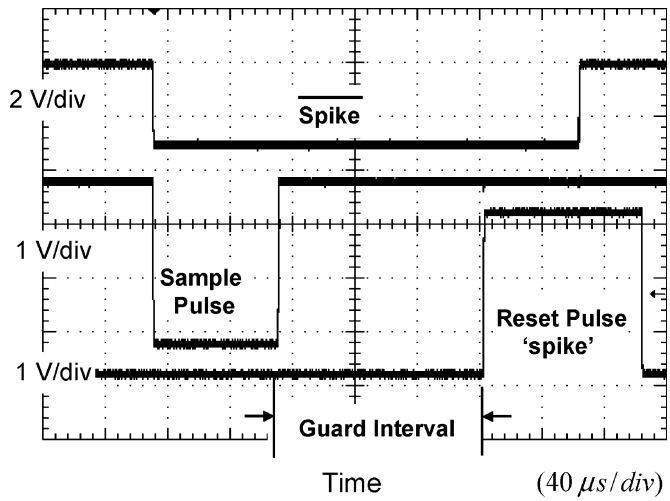

Fig. 7. Oscilloscope trace of the timing signals used to sequence and control sampling, comparison and memory storage.

C. Timing Signals

The proper sequencing and time duration of the ramp voltage sampling and reset events is a crucial task. The sampling pulse must be long enough to complete the sampling process and must end before the onset of the reset pulse. The *reset* pulse must be long enough to allow the complete discharge of C3 (see Fig. 10, shown later). The comparison between the new interval voltage (V_n) and the previous interval voltage, (V_{n-1}) must occur (in the interval comparator circuit) after the *sample* pulse and before the onset of the *reset* pulse. Therefore, a guard interval that can be adjusted to meet these requirements has been included in the design of the circuit. Fig. 7 shows an oscilloscope trace of the timing signals that are generated with every new spike. The circuits that generate the *sample* and *reset* pulses are shown in Figs. 8 and 9, respectively. $\overline{\text{spike}}$ begins high, holding V_x low and "*turn off*" low.

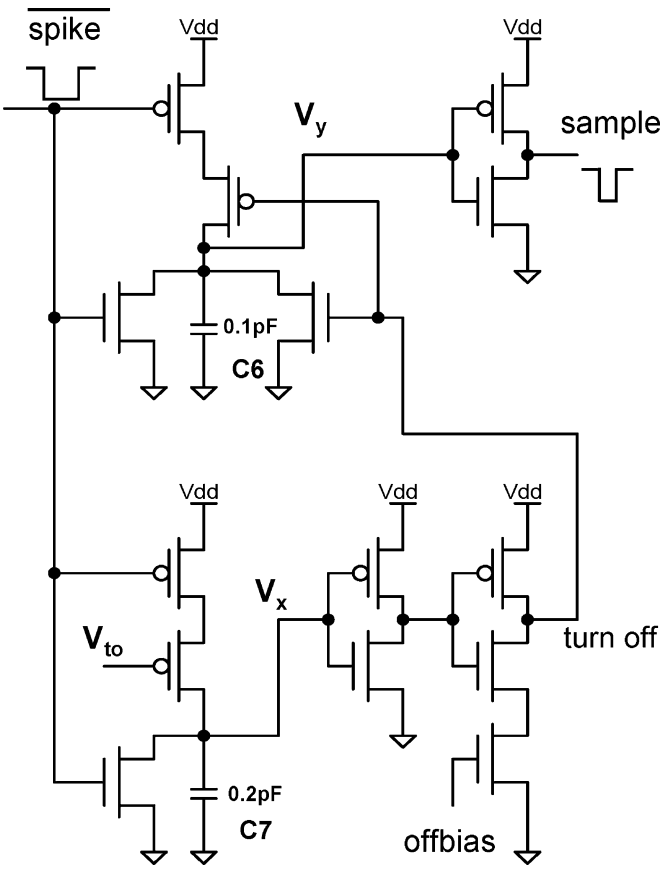

Fig. 8. Timing signal generator: generating the sample pulse.

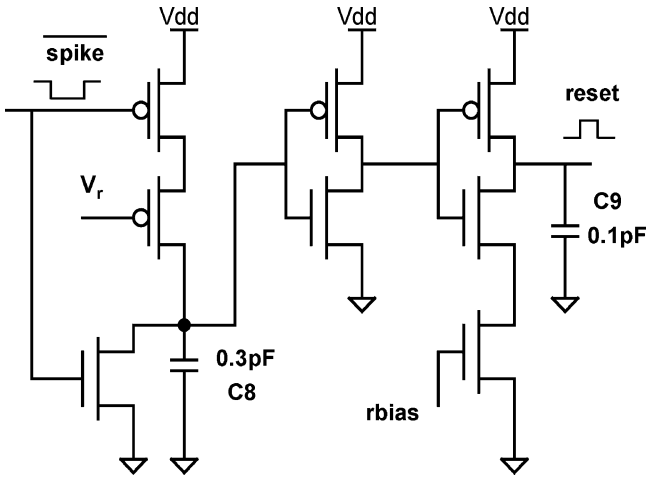

Fig. 9. Timing signal generator: generating the reset pulse.

The falling edge of $\overline{\text{spike}}$ starts the process of charging capacitors C6, C7 (Fig. 8), and C8 (Fig. 9). The charging rate is determined by the charging currents as well as the size of the capacitors. As each capacitor's voltage crosses the threshold of an inverter, a pulse is generated. The circuit in Fig. 8 is responsible for generating the sample pulse. V_y is pulled high abruptly on the falling edge of $\overline{\text{spike}}$ which pulls down *sample* and starts the sampling process. The sampling process continues until the voltage ramp V_x causes the *turn off* pulse to become high. C6 is then discharged which results in *sample* switching back to its high state. The rate at which C7 charges up can be controlled

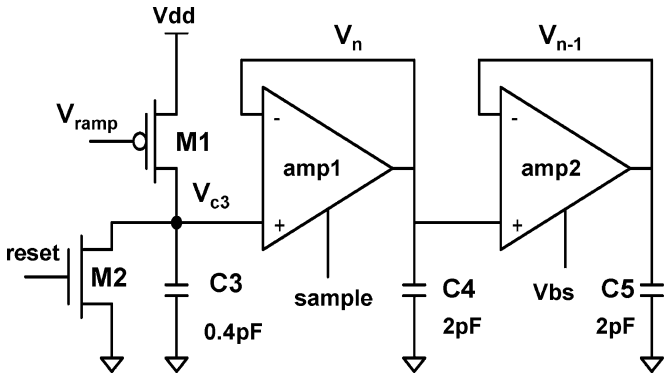

Fig. 10. Interval-to-voltage converter, sampling and storing of intervals. The constant current source M1 charges C3 producing a voltage proportional to the interspike interval. This voltage is sampled and stored in C4 and is slowly copied and stored in C5.

by the bias voltage V_{to} which in turn controls the width of the sampling pulse. The circuit in Fig. 9 generates the reset pulse. The voltage ramp developed across C8 will turn on the *reset* pulse. The bias voltage V_r controls how fast the ramp rises to threshold. This delayed reset turn-on is used to adjust the separation between the sampling and reset pulses. The voltage *rbias* can be used to extend the width of the reset pulse beyond the rising edge of $\overline{\text{spike}}$.

D. Voltage Conversion, Sampling and Storage

Using a technique very similar to [10], we converted the interspike intervals into voltages by sampling (and holding) a fixed-slope voltage ramp. The interval is then stored for further processing. All three operations are shown in Fig. 10. The constant current source formed by transistor M1 and fixed bias V_{ramp} charges C3. The voltage ramp V_{c3} is used to convert an interspike interval to a voltage. As a spike arrives, the ramp is sampled and the capacitor C4 stores this new interspike interval V_n to be compared (in the interval comparator stage) with the preceding interval V_{n-1} stored on the capacitor C5. The *sample* pulse that is generated with every new spike turns the follower amp1 on for a short period of time, sampling the linear ramp. The *reset* pulse that follows the *sample* pulse discharges C3 through M2 in preparation for a new interval-to-voltage conversion. The sampled voltage V_n is temporarily stored in C4. The follower amp2 is weakly biased so that its output slowly follows its input. In effect, this allows the sampled interval V_n to be slowly transferred to memory as V_{n-1} on C5. Since the current charging C3 is constant, the voltage on the capacitor increases linearly with time. Therefore, an interspike interval is linearly transformed into a voltage. Mathematically

$$V_n = mT_n \tag{3}$$

where m is the slope of the voltage ramp V_{c3}, T is the interspike interval, V is the voltage representation of the interspike interval, and n is the time index. The slope of the ramp m is controlled by V_{ramp}. Because the dynamic range is constrained by the power rails, the ramp voltage V_{c3} saturates for long intervals. If V_{sat} is the saturation voltage, then the interval T_{sat} at which the transformation begins to saturate is given by

$$T_{\text{sat}} = \frac{V_{\text{sat}}}{m}. \tag{4}$$

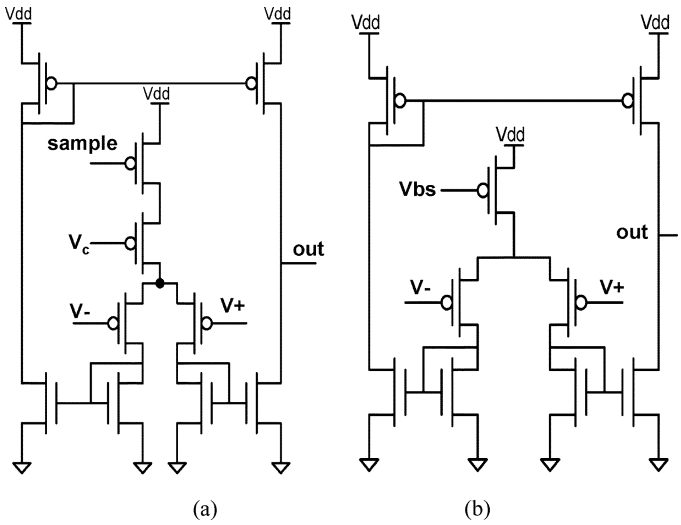

Fig. 11. Wide-range transconductance amplifiers used in sampling and storage. (a) Amp1. (b) Amp2.

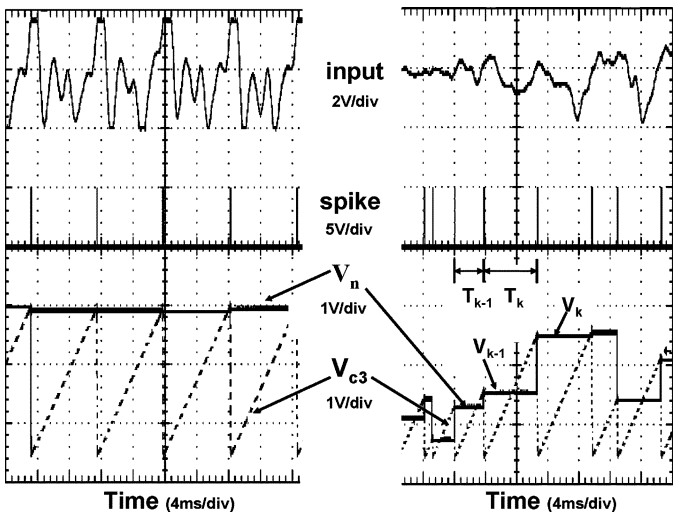

Fig. 12. Two example stimuli: (left) harmonic complex and (right) irregular noise-like signal. Oscilloscope traces show the spike waveform, the interval-to-voltage conversion and the sampling processes. V_n remains constant throughout the interspike interval and gets updated with every new spike.

Fig. 11 shows the implementation of the wide-range transconductance amplifiers amp1 and amp2. Because *sample* swings down to ground, the transistor with control voltage Vc [see Fig. 11(a)] was added to limit the current flow. Fig. 12 shows oscilloscope traces that demonstrate the voltage conversion process and the sampling process. As a spike comes in, the ramp voltage V_{c3} is first sampled then reset, V_{c3} then charges linearly with time encoding the new interspike interval. When another spike arrives, the encoded interval is sampled and the voltage V_{c3} is reset repeating the whole process.

The sampled voltage interval V_n remains constant throughout the interspike interval and gets updated with every new spike via the sampling process.

E. Interval Limiter

In the case of a noisy input, it is possible to get very small interspike intervals that match causing false alarms. Saturation of the voltage transformation for long intervals is another concern and may also result in false alarms. It is therefore desirable to have a circuit that defines a range of "valid interspike intervals." This validation circuit is shown in Fig. 13. As long as the stored "interval" V_{n-1} satisfies the inequality $V_L < V_{n-1} < V_H$, where V_L and V_H are externally defined by the user, the output transistors M1 and M2 will be on. The output of the circuit will be low, indicating a "valid interval." Equivalently in the time (interval) domain, we find that the circuit evaluates the inequality $T_{\min} < T_{n-1} < T_{\max}$, where $T_{\min}$ and $T_{\max}$ are the minimum and maximum valid intervals, respectively, and are given by

$$T_{\min} = \frac{V_L}{m} \quad T_{\max} = \frac{V_H}{m}. \tag{5}$$

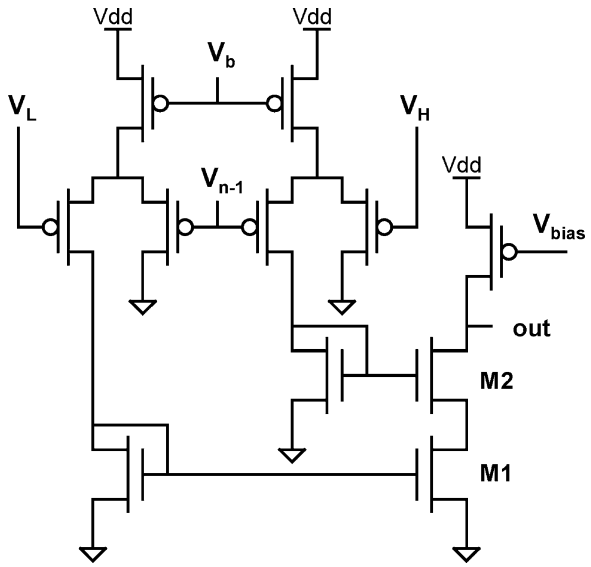

Fig. 13. Interval limiter. If stored interval V_{n-1} is valid, both M1 and M2 will be turned on and *out* will be low indicating a valid interval.

Fig. 14 illustrates the linear interval-to-voltage transformation that saturates for intervals longer than T_{sat}. Since no interval T_n can be less than T_{refr}, T_{refr} must be chosen to be less than $T_{\min}$.

F. Interval Comparator

The interval comparator is the heart of the decision making process. It must decide whether the two consecutive interspike intervals, T_n and T_{n-1} match; a match being defined as satisfying the inequality $|T_n - T_{n-1}| < \varepsilon_t$, where ε_t is the accuracy (or tolerance) of the comparison. Equivalently in the voltage domain is $|V_n - V_{n-1}| < \varepsilon_V$ where $\varepsilon_v = m\varepsilon_t$ and m is the slope of the transformation. Fig. 15 is the circuit that evaluates this inequality. V_b sets a current I_b in the differential pair that is split into I_1 and I_2 which are mirrored to the two outer branches. V_{th} sets a current $I_{\text{th}} < I_b/2$. When the input voltages match, both I_1 and I_2 are strong enough to hold V_x and V_y low, thereby turning off M1 and M2 and the output is high, indicating matched intervals. When the inputs do not match, most of I_b is steered in one of the two branches, I_{th}, in turn, pulls either V_x or V_y high and the output voltage is pulled down, indicating non-matching intervals. The decision must be completed before the onset of the reset pulse (i.e., in less than 200 μs). This causes ε_v to be a function of V_{th2}. Table I shows the measured

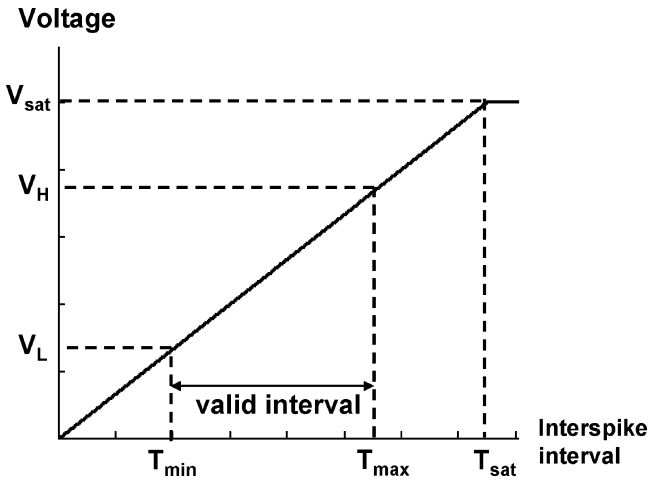

Fig. 14. Interval-to-voltage transformation and the "valid interval" range as determined by V_L and V_H.

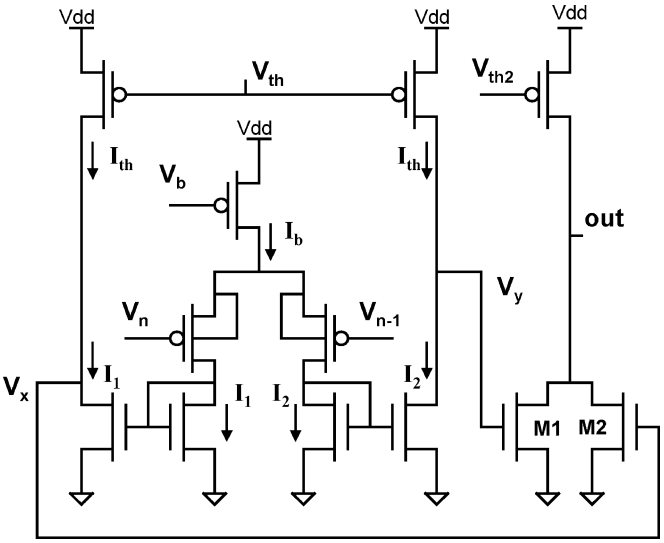

Fig. 15. Interval comparator. If the two consecutive intervals do not match, I_b is steered in one of the branches of the differential amplifier,V_x (or V_y) is pulled up high turning on M2 (or M1) pulling down *out*.

TABLE I
ACCURACY (ε_v) OF THE COMPARATOR (FOR FIXED V_b AND $V_{\rm th2}$) AS A FUNCTION OF $\Delta V = V_{\rm th} - V_b$

ΔV (mV)	49	61	72	83	96	105	114
$\mathcal{E}_v$ (mV)	30	50	73	86	106	125	144

threshold ε_v as a function of $(V_{\rm th} - V_b)$. Since V_n eventually becomes V_{n-1}, the output of this circuit spends most of its time in the matching (high) state. This output is only used, however, after the sampling process has ended and during the time when the reset pulse is high. When the reset pulse is high, if the comparator decides that the two intervals do not match, all five shift registers are reset. The value of $V_{\rm th2}$ also affects the time that it takes for the output to switch back to its high state.

G. Shift Register and Shift Logic

The shift register block shown in Fig. 1 is a five-state shift register. When two consecutive intervals match and satisfy the interval-limiter condition we call it a "hit." If all five registers contain zeros (i.e., "reset"), a "one" gets loaded into the first register only when a "hit" occurs. The "shift logic" block will either generate a clock signal advancing the "one" to a new state if a new "hit" occurs, otherwise, it will generate a reset pulse clearing the entire shift register.

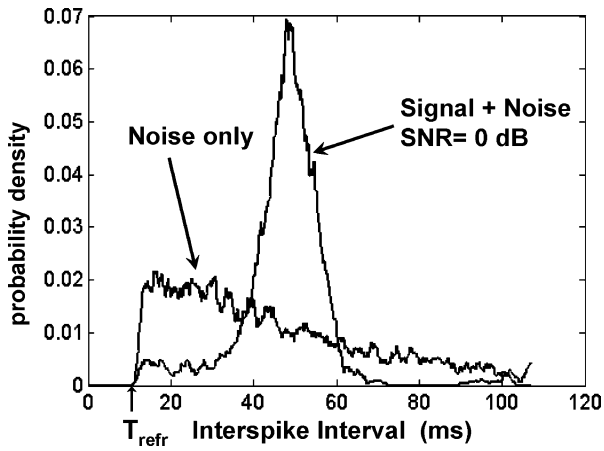

Fig. 16. Measured pdf of two stimuli. (a) Noise only. (b) 20-Hz sinewave signal plus noise. The refractory period was 11.8 ms.

H. Power Consumption

The power consumption of the chip was measured in two different modes; the "testing" mode and the "stand-alone" mode. The on-chip instrumentation circuitry used to monitor various internal signals was powered in the testing mode but was turned off in the stand-alone mode. In both cases, the power was measured with a band-limited (at 1 kHz) white Gaussian noise input and a 40-Hz sinusoidal input. In the testing mode, the power consumption was approximately 15 and 21 μW for the noise and sinusoidal inputs, respectively. In the stand-alone mode, the circuit's normal mode of operation, the power consumption was approximately 1.8 and 0.9 μW for the noise and sinusoidal inputs, respectively.

IV. MATHEMATICAL ANALYSIS AND MODELING

Similar to a passive sonar detection system where the receiver listens for the acoustic signature of engines, propellers and other elements in an underwater vehicle, this problem can be modeled as detecting a sample function from a random process of target sounds buried in noise [11]. Regardless of whether the received signal is noise only or signal plus noise, the only information we have about the received signal after peak detection lies in the interspike intervals. The decision procedure must be based upon the *difference in the statistical properties* of these interspike intervals. In the following analysis, we use real data measured from our chip. Signals were generated using MATLAB (Mathworks Inc.) and presented to the chip through a sound card. The MATLAB Instrument Control Toolbox was used to capture the desired data from the oscilloscope. Fig. 16 illustrates the measured probability distribution function (pdf) of the interspike intervals for two different cases. In the first case, white Gaussian noise was low-pass filtered at 1000 Hz and presented to the chip (relative noise power was -28 dB). In the second case, a 20-Hz

sine wave was added to the background noise at an SNR of 0 dB. In both cases, the refractory period was 11.8 ms and the duration of the signal was 80 s. As the SNR increases, the pdf of interspike intervals for the signal plus noise case converges to an impulse located at the period of the sine wave.

The *algorithm* that we have adopted to distinguish between the two types of inputs is based upon the assumption that the interspike intervals generated from the peaks of the envelope of the signal of interest will be centered at the period of the envelope and that the deviation from this center will be relatively small and will decrease with the increase of the SNR. The chip counts what we have called "hits." A hit is the simultaneous occurrence (intersection) of two events. The first, denoted as A is the event that the stored interval V_{n-1} lies within the acceptance range defined by V_L and V_H. That is

$$A = \{V_L < V_{n-1} < V_H\}. \tag{6}$$

The second event, denoted as B is the event that the two consecutive intervals V_n and V_{n-1} are equal to within a constant ε_v, that is

$$B = \{|V_n - V_{n-1}| < \varepsilon_v\}. \tag{7}$$

We can now define the probability of a hit as the probability of the intersection of the two events A and B

$$p(\text{hit}) = p(A \cap B). \tag{8}$$

It is straightforward to show that the probability of a hit is upper bounded by the smaller probability of the two events A and B, that is

$$p(\text{hit}) \leq \min\left(p(A), p(B)\right). \tag{9}$$

As more and more interspike intervals fall in the acceptance range, $p(A)$ approaches unity and $p(\text{hit})$ approaches $p(B)$. The probability of the event A can be computed from the pdf of interspike intervals by evaluating the integral

$$p(A) = \int_{T_{\min}}^{T_{\max}} f(x)dx \tag{10}$$

where $f(x)$ is the pdf of the interspike intervals.

Before we move on to analyze the effect of different circuit parameters, we introduce a simple dimensionless measure of variability in the interspike interval distribution known as the coefficient of variation C_v [12], [13], defined as

$$C_v = \frac{\sigma_{T_n}}{E(T_n)} \tag{11}$$

where σ_{T_n} is the standard deviation of the interspike interval distribution and $E(T_n)$ is the mean interspike interval. The interspike intervals in our chip obey the inequality

$$T_{\text{refr}} \leq T_n \leq T_{\text{sat}}. \tag{12}$$

Taking the expectation of this inequality we get

$$T_{\text{refr}} \leq E(T_n) \leq T_{\text{sat}} \tag{13}$$

where T_{refr} and T_{sat} are constant. Since $E(T_n)$ is upper bounded, it follows that the probability of the event B is related to C_v, in the following sense:

$$C_v \to 0 \Rightarrow \sigma_{T_n} \to 0 \Rightarrow p(B) \to 1. \tag{14}$$

We begin by first studying noise.

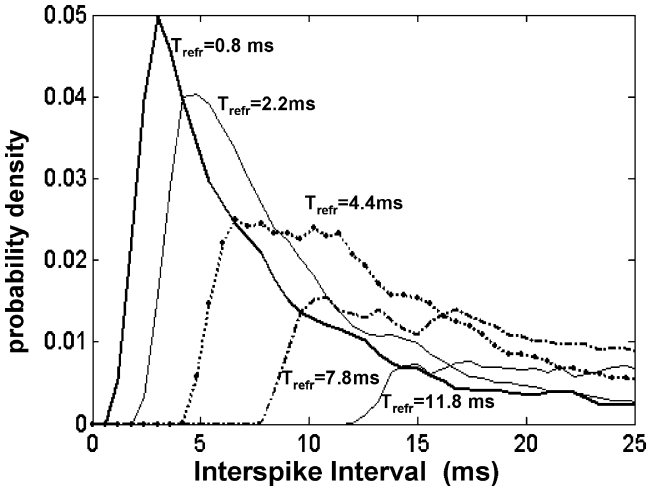

Fig. 17. Effect of refractory period on the pdf of interspike intervals.

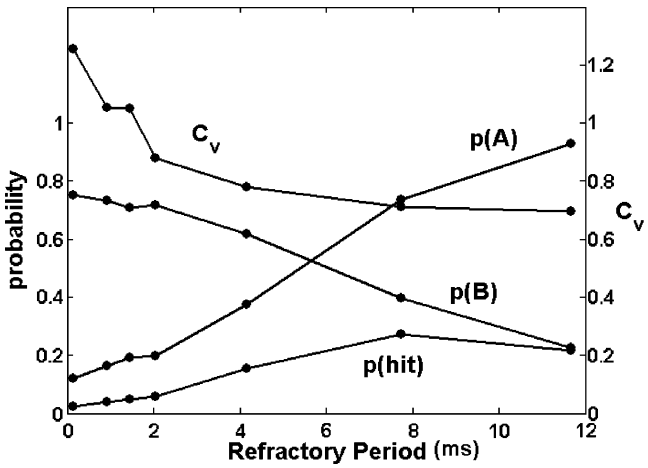

Fig. 18. Effect of the refractory period on $p(\text{hit})$ and C_v. The greater the refractory period the more regular the spike train the smaller C_v is. As more of the interspike intervals fall in the valid interval region. $p(A)$ approaches unity and $p(\text{hit})$ approaches $p(B)$.

A. Noise Only

The distribution of interspike intervals of noise that are generated from the peak-detection process depend on four parameters: refractory period, bandwidth, linear decay rate of the peak detector, and noise power. The effect of these parameters on the measured pdf of the interspike intervals will be shown.

1) Refractory Period: The effect of the refractory period, T_{refr}, on the pdf of noise interspike intervals is shown in Fig. 17.

The first effect is that the interval distribution is zero for intervals less than T_{refr}. The second effect is that the pdfs spread out as T_{refr} is increased and are not just shifted versions of one another. This can be attributed to the behavior of the refractory circuit; after a spike is generated, the circuit will remain in its refractory state (thus longer intervals) if subsequent inter-peak intervals are less than the refractory period. As a result, stimuli that contain many short inter-peak intervals will have a broader, flatter pdf as these short intervals get connected onto other intervals. Fig. 18 illustrates how the increase of T_{refr} affects several parameters. On one hand, more of the noise intervals fall in the valid interval region increasing $p(A)$. On the other hand, however, the increase in the spread of the distribution means fewer consecutive intervals will match resulting in the decrease

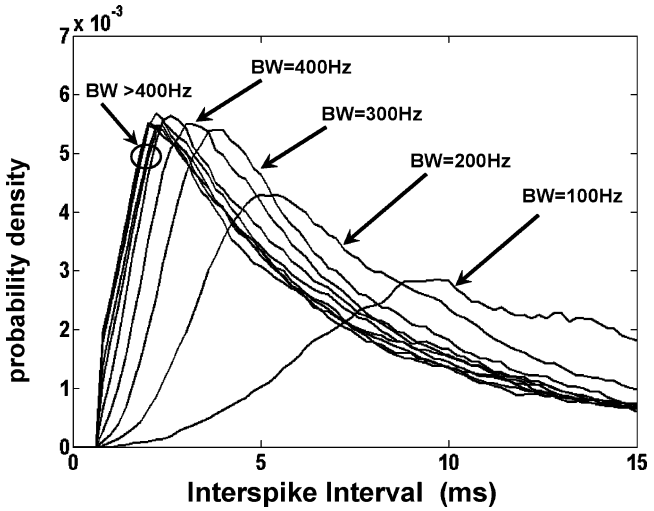

Fig. 19. Effect of bandwidth on the pdf of noise interspike intervals. No significant change in pdf for bandwidths greater than 400 Hz.

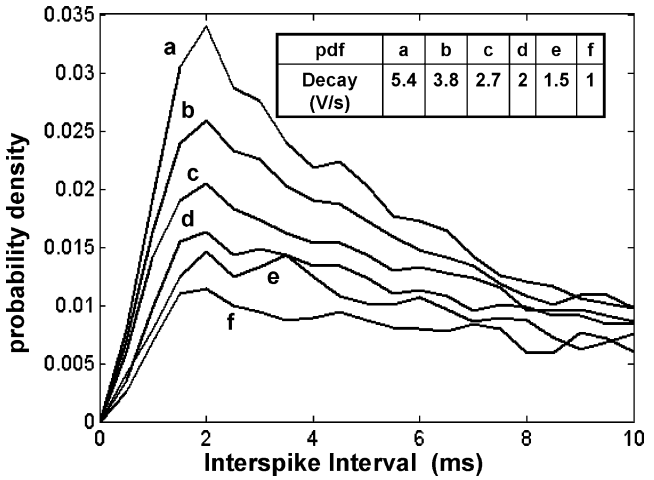

Fig. 20. Effect of decay rate (in volts per second) on the pdf of noise interspike intervals.

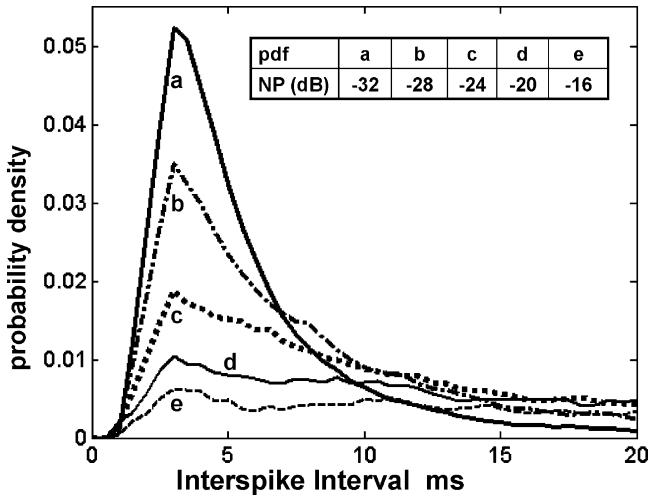

Fig. 21. Effect of noise power (NP) on the pdf of noise interspike intervals.

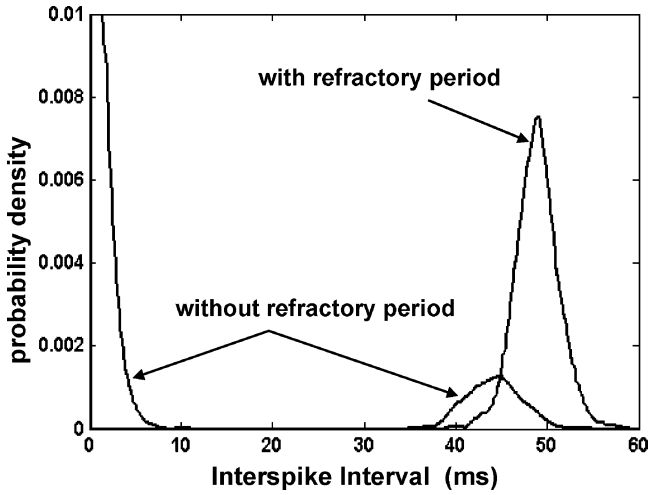

Fig. 22. Effect of the refractory period ($T_{\rm refr} = 11.8$ ms) on the pdf of interspike intervals in the signal plus noise case. The signal is a 20 Hz sinusoid ($\rm SNR = 15$ dB).

of $p(B)$. The net effect on $p(\rm hit)$ is also shown. The decrease in C_v is in agreement with the intuitive idea that the spike train becomes more regular as the refractory period is increased.

The pdfs shown in Figs. 19–21 were measured with a refractory period less than or equal to 1 ms.

2) Bandwidth: Decreasing the bandwidth of the noisy input means that the envelope of the waveform will become "smoother" and the probability of generating multiple spikes off the same peak decreases. Fig. 19 shows that for bandwidths greater than 400 Hz, there is no significant change in the pdf of the interspike intervals (for this set of parameters).

3) Decay Rate of Peak Detector: Qualitatively, increasing the decay rate of the peak detector causes the peak detector's output to encounter a peak earlier than it would have with a smaller decay. The effect this has on the pdf of the interspike intervals is illustrated in Fig. 20. Decreasing the decay rate causes more intervals to fall in the valid interval range increasing $p(A)$. The spread in the pdf, however, causes a decrease in $p(B)$. The maximum $p(\rm hit)$ that was measured for this set of data was 0.11.

4) Noise Power: Increasing the noise power increases the probability of occurrence of large impulses of noise. Because the fixed linear decay rate of the peak detector does not scale with amplitude, the time it takes for the peak voltage to encounter another peak of the envelope and thus generate a spike increases. Fig. 21 shows that increasing the noise power increases the relative probability of getting a long interval.

B. Signal Plus Noise

For simplicity we will use a sinusoidal waveform as our signal of interest. The input filtering bandwidth must be wide enough to pass our signal of interest. The period T of our signal must satisfy the following inequality $T_{\rm refr} < T_{\min} < T < T_{\max}$. To successfully detect periodicity, only one spike should be generated at the peak of the envelope. Without a refractory period, it is possible to get more than one spike at the peaks of the signal. This is likely to occur in a noisy environment but can occur even when the SNR is relatively high. Fig. 22 illustrates the effect of the refractory period on the pdf of interspike intervals of a 20-Hz

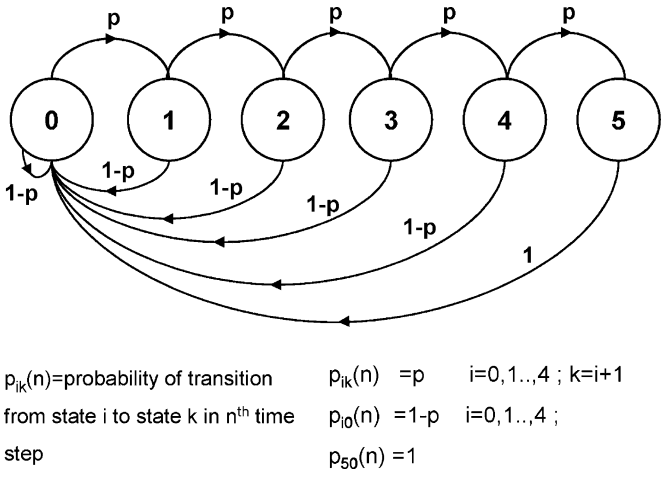

Fig. 23. State diagram of the 5 shift-register-based counter used to count consecutive hits.

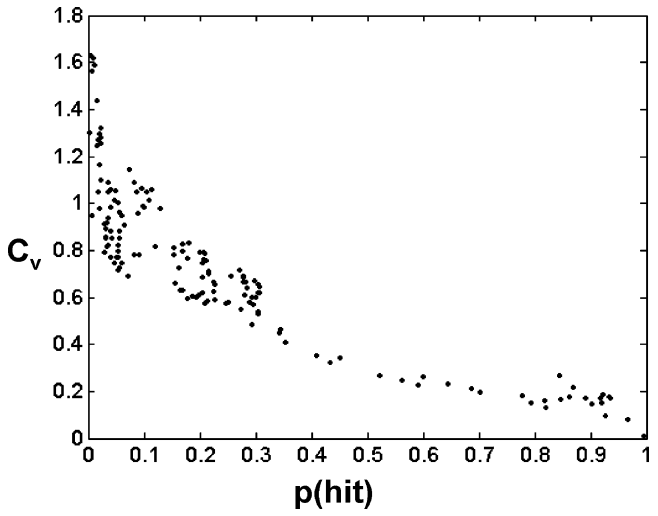

Fig. 24. Coefficient of variation versus $p(\mathrm{hit})$. The more regular the spike train the smaller the coefficient of variation and the higher the probability of getting two consecutive intervals to match.

sinusoidal signal with a SNR of 15 dB. In the absence of a refractory period (other than the 0.3-ms refractory period caused by the width of the spike itself), multiple spikes are produced in the region where the peak-voltage "tracks" the peak of the sinusoid; this explains the non-zero probability of getting small intervals. In addition, any long intervals obtained will tend to be smaller than the actual period of the sinusoid which explains why the small "bump" in the pdf is not centered at 50 ms.

Having a refractory period (11.8 ms in this case) suppresses spiking for a short time after a spike has been generated. The effect this has on the pdf of interspike intervals is also shown in Fig. 22. Heuristically, if we assume that the peak detector will track the sinusoidal input for one eighth of the period, then, in order for the refractory period to be effective in suppressing spiking T must satisfy the inequality $8T_{\mathrm{refr}} > T$. This may set a larger lower bound on the frequency range than that imposed by $T_{\max}$.

C. Modeling the Counter

The shift-register-based counter used in our circuit (5 registers) can be modeled as a finite Markov Chain. There are six states X0, X1, ..., X5. The X0 state corresponds to all five registers being reset. The states X1, X2, ..., X5 correspond to registers 1 to 5 respectively. When a hit occurs, the first register (state X1) is loaded with a logic 1, if another hit occurs, the logic 1 is shifted to the second register (state X2) and so on, if a spike occurs and there is no match, the whole counter is reset back to zero (state X0). The counter can count up to five consecutive hits. The state diagram for this chain is shown in Fig. 23.

The chain can be characterized by its state transition matrix P. The p_{ik} entry in the matrix denotes the probability of transitioning from state i to state k. Two assumptions have been made regarding this model. The first assumption is that the chain is homogenous, that is, the transition probabilities are independent of the time step n. The second assumption, which is generally not true but has been made to simplify the analysis, is that all the forward transition probabilities p_{ik} have been assumed to be equal

$$p_{ik} = p \qquad i = 0, 1, \ldots, 4; \quad k = i + 1 \tag{15}$$

where $p = p(\mathrm{hit})$. The chain is irreducible and aperiodic, and a theorem [14] holds that

$$p_{ik}(n) \to \frac{1}{\mu_k} \quad \text{as} \quad n \to \infty, \qquad \text{for all } i \text{ and } k \tag{16}$$

where μ_k is the mean recurrence time for state k. Therefore, $p_{ik}(\infty)$ can thus be considered as the normalized rate of hits reaching the kth state for $k > 0$. Using this model, p was scanned from 0 to 1 and $p_{ik}(\infty)$ was computed numerically in each case by evaluating P^n for a large value of n (all entries in the kth column become equal). The simulation results are presented in Fig. 25. Note that when p equals 1, the counter behaves like a six-state ring counter, and the normalized rate of hits for all states equals 1/6.

V. Results

A. Testing With Noise and Sinewaves

Fig. 24 illustrates the measured relationship between the coefficient of variation C_v and the probability of getting two consecutive intervals to match $p(\mathrm{hit})$. Both C_v and $p(\mathrm{hit})$ were computed from chip data. In some cases the input was noise only with different settings for bandwidth, decay rate, noise power, and refractory period (as described in Section IV). This explains the wide variation in C_v. In other cases the input had a 20 Hz sinusoid added to the background noise with different SNRs. The increase in $p(\mathrm{hit})$ as C_v decreases indicates that the *algorithm* is successful in predicting regularity in the spike train.

Fig. 25 illustrates the mean rate of hits measured at the outputs of the third and fifth registers (states) plotted as a function of $p(\mathrm{hit})$. The individual dots are measurements from the chip while the solid lines are from the Markov model simulation where the normalized rate was multiplied by 20 to account for the 20-Hz frequency of the sinusoidal input. We notice that for medium values of $p(\mathrm{hit})$ our measurements are greater that the simulated values. This could be attributed to the fact that the transitional probabilities depend on the instantaneous value

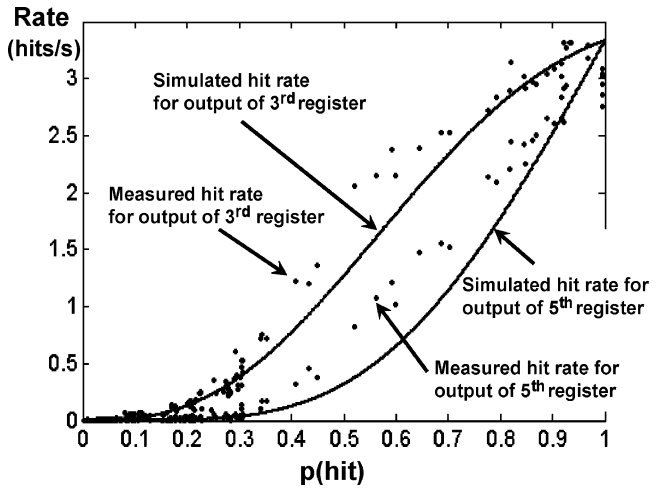

Fig. 25. Measured and simulated rate of hits at the outputs of the third and fifth taps of the shift register.

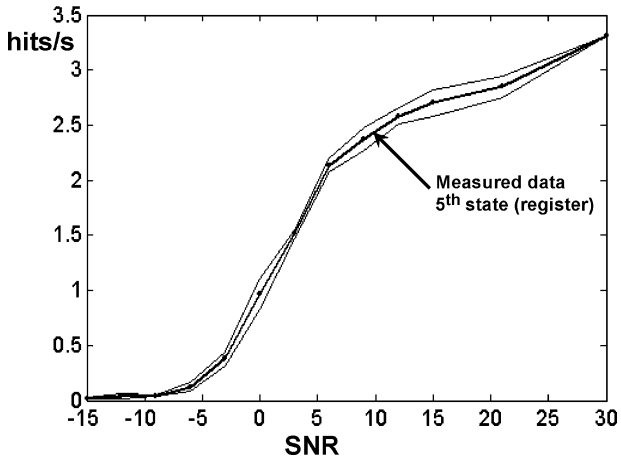

Fig. 26. Measured rate (dark line) of hits at the output of the fifth register as a function of SNR for a 20 Hz sinusoidal input with $T_{\rm refr} = 11.8$ ms. The two thinner lines represent the one standard deviation boundary.

of the interspike interval; therefore, the chain is not homogeneous. This also means that the transitional probabilities are not equal. Another possibility is that due to the inability to directly measure the accuracy of the *interval comparator* circuit and device mismatch, we may have underestimated the value of $p(\text{hit})$ computed from our measurements. Note that as $p(\text{hit})$ approaches 1, the rate of hits measured at the third and fifth states converge to 3.33 (20 Hz/6) confirming the six-state ring counter behavior.

Figs. 24 and 25 serve to illustrate that the chip has successfully implemented the algorithm; higher regularity in the spike trains results in a higher probability of getting a hit which in turn results in a higher rate of measuring hits at the third and fifth outputs of the shift register. Fig. 26 shows the measured rate of hits at the output of the fifth register as a function of SNR. The input is a 20-Hz sinusoidal input added to a background white Gaussian noise (relative noise power is -28 dB), low-pass filtered at 1000 Hz. A refractory period of 11.8 ms was used. It is predicted from the plot that the sinusoid with a SNR as low as -5 dB can be detected.

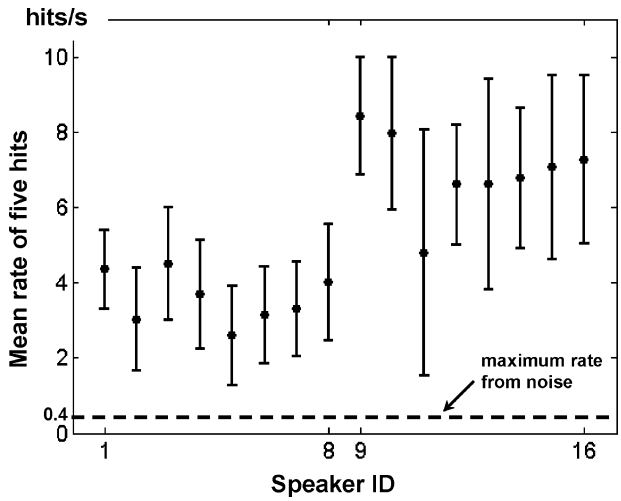

Fig. 27. Mean rate of five hits for test sentences taken from the TIMIT speech database. Speakers 1 through 8 are male speakers while speakers 9 through 16 are female speakers.

B. Real World Signals

We have tested the periodicity detection chip tuned for voice frequencies and for vehicle engine frequencies. For the voice testing, the refractory period was 1.8 ms, the measured minimum and maximum allowable frequencies (set by V_H and V_L) were 38 and 248 Hz, respectively. The core test set of the TIMIT speech database [15] was used to evaluate the performance of the chip as a voice detector. One male speaker and one female speaker from each of the eight dialects were selected providing a test set of sixteen speakers. The total number of sentences used in the test was 160 (ten per speaker). The mean duration of the sentences was approximately 3 s. The sentences were downsampled to 2 kHz and repeated five times. The number of three consecutive hits and five consecutive hits was recorded for each trial and the mean rate of hits was computed for each speaker. Fig. 27 represents the mean rate of five consecutive hits for each speaker. Speakers 1 through 8 were male speakers while speakers 9 through 16 were female speakers. Because of their higher pitch, female speakers tend to produce a greater number of hits than males. The quality of the recording was such that no false alarms were generated for a noise signal with the same power as that of the background noise in the recording. Therefore, by setting the threshold rate sufficiently low, 100% of the speakers can be detected with no false alarms. Fig. 28 shows the effect of adding noise to four different sentences spoken by four different speakers. The speakers that were chosen were the two males and two females that had the greatest and smallest rate of five consecutive hits. With a threshold of as low as 0.5 hits/s, the sentences can be detected down to a SNR of 15 dB. To compare the response to noise input, the chip was tested with white Gaussian noise having a bandwidth of 1 kHz and a duration of 20 s. The noise power was scanned from -60 dB to -5 dB in steps of 5 dB, it was also tested at -3 dB and -1 dB. No spikes were observed for noise powers less than -40 dB. The greatest recorded false-alarm rate for the five consecutive hit output was 0.4 hits/s.

Fig. 29 shows the response of the chip to the speech utterance "zero," the three consecutive hit output is shown to trigger in the

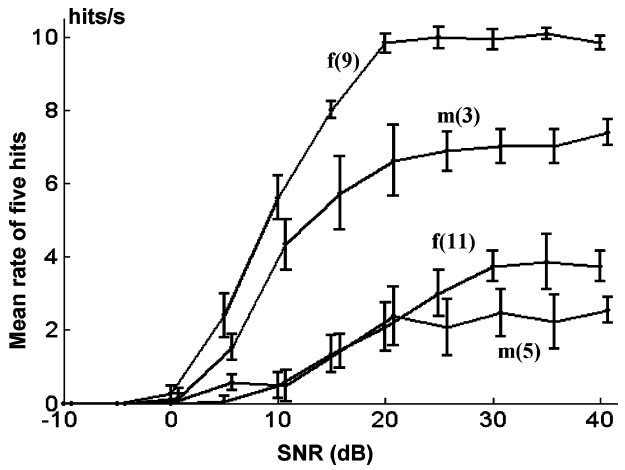

Fig. 28. The effect of adding noise to four different sentences spoken by four different speakers. "m" and "f" denote gender while the number in the parenthesis is the speaker ID. With a threshold of 0.5 hits/s, detection is possible down to a SNR of 15 dB.

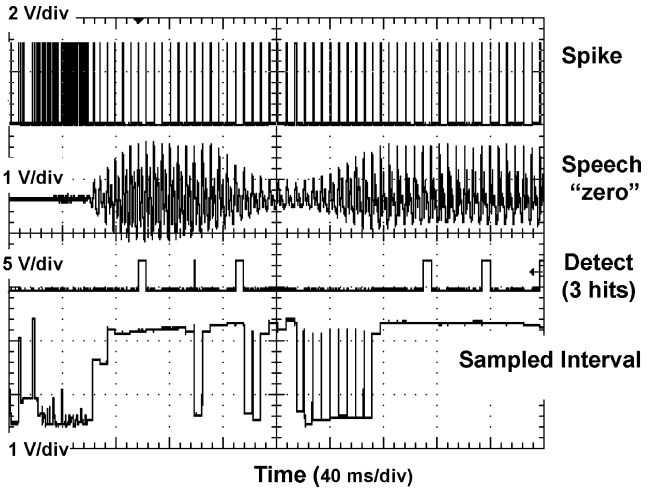

Fig. 29. Response to the speech utterance "zero." No detection during "noisy" onset. Detects the voiced segments. Sampled interval can be used to estimate pitch.

voiced segments of the utterance and not in the noisy onset. Because the Sampled Interval voltage V_n is directly related to the interspike (inter-peak) interval, the pitch in the voiced segments can be computed from V_n.

In the vehicle detection case, although extensive testing was not done, vehicle detection is possible based on observations of the system response to the recordings of vehicles. We show that the chip functions properly in this low frequency range. The chip parameters were set so that the minimum and maximum allowable frequencies would be around 7 and 45 Hz, respectively. Fig. 30 shows the response of the chip to a recording of the sound of a vehicle. The recording had a sampling frequency of 1 kHz. White Gaussian noise was added to the recording at a SNR of 20 dB. The pitch of the envelope in this example as estimated from the interspike interval is approximately 35 Hz. The spectrum (not shown) of the signal shows a peak at 70 and 105 Hz, thus the fundamental frequency (35 Hz) is missing.

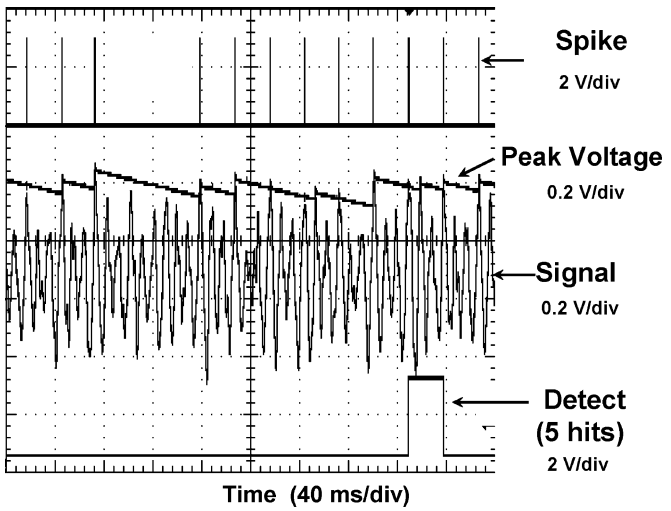

Fig. 30. Response to a recording of the sound of a vehicle.

VI. Conclusion

We have designed and fabricated an analog VLSI low-power circuit that successfully implements an algorithm for detecting periodicities in the time-domain envelope of a signal. We have tested the chip with speech, pure tones, and informally tested it with vehicle signals. The chip's input was band-limited at 1 kHz and was tested with target signals having pitches as low as 7 Hz and as high as 250 Hz. The overall system has been mathematically modeled and the chip measurements were compared against the model. The chip was fabricated in a commercially-available 2-poly 1.5-μm CMOS process and occupies an area of about 0.242 mm^2. The chip was operated at a supply voltage (V_{dd}) of 3 V and consumes less than 1.8 μW. It consists of 215 transistors and nine capacitors. The analog processing blocks of the circuit account for 92 of the transistors; the logic and shift register blocks account for the rest.

Acknowledgment

The authors thank S. Shamma for early discussions of the approach to this work and the Signal Systems Corporation for providing various acoustic recordings during the design period of this work; J. Simon for his critical review and constructive comments; and the MOSIS fabrication service for its continuing support in providing fabrication facilities.

References

[1] D. E. Culler and H. Mulder, "Smart sensors to network the world," *Sci. Amer.*, pp. 85–91, Jun. 2004.
[2] A. van Schaik, "An analog VLSI model of periodicity extraction in the human auditory system," *Anal. Integr. Circuits Signal Process.*, vol. 26, no. 2, pp. 157–177, Feb. 2001.
[3] D. H. Goldberg and A. G. Andreou *et al.*, "A wake-up detector for an acoustic surveillance sensor network: algorithm and VLSI implementation," in *Proc. Int. Symp. Information Processing in Sensor Networks*, Apr. 2004, pp. 134–141.
[4] L. O. Dolansky, "An instantaneous pitch-period indicator," *J. Acoust. Soc. Amer.*, vol. 27, pp. 67–72, Jan. 1955.
[5] M. Filip, "Envelope periodicity detection," *J. Acoust. Soc. Amer.*, vol. 45, pp. 719–732, Mar. 1968.
[6] B. Gold and L. Rabiner, "Parallel processing techniques for estimating pitch periods of speech in the time domain," *J. Acoust. Soc. Amer.*, vol. 46, pp. 442–448, Aug. 1969.

[7] M. Godfrey, "CMOS device modeling for subthreshold circuits," *IEEE Trans.Circuits Syst.*, vol. 39, no. 8, pp. 532–539, Aug. 1992.
[8] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.
[9] J. E. Dowling, *Neurons and Networks: An Introduction to Neuroscience*. Cambridge, MA: Harvard Univ. Press, 1992.
[10] N. Kumar *et al.*, "Auditory feature extraction using self-timed, continuous-time discrete-signal processing circuits," *IEEE Trans.Circuits Syst. II, Analog Digit. Signal Process.*, vol. 44, no. 9, pp. 723–728, Sep. 1997.
[11] H. L. Van Trees, *Detection, Estimation, and Modulation Theory: Part 1*. New York: Wiley, 2001.
[12] C. Koch and I. Segev, *Methods in Neuronal Modeling*. Cambridge, MA: MIT Press, 1998, ch. 9.
[13] D. H. Johnson, "Point process models of single-neuron discharges," *J. Comp. Neurosci.*, vol. 3, no. 4, pp. 275–299, 1996.
[14] G. R. Grimmett and D. R. Strizaker, *Probability and Random Processes*. New York: Oxford Univ. Press, 1992, ch. 6.
[15] "The DARPA TIMIT acoustic-phonetic continuous speech corpus," DARPA, Washington, DC, NIST Speech Disc 1-1.1, NTIS order number PB91-505065, 1990.

Hisham Abdalla (S'94–M'05) received the B.S. degree in electrical engineering from Alexandria University, Alexandria, Egypt, in 1997 and the M.S. degree in electrical engineering from the University of Maryland at College Park, in 2001. He is currently working toward the Ph.D. degree at the Computational Sensorimotor Systems Laboratory, at the same university.

He held summer internship positions at AT&T and the Naval Research Laboratory in 1995 and 2000, respectively. His general research interests are analog very large-scale integration (VLSI) and computational neuroscience.

Timothy K. Horiuchi (M'99) received the B.S. degree in electrical engineering and the Ph.D. degree in computation and neural Systems from the California Institute of Technology, Pasadena, CA, in 1989 and 1997, respectively.

Following a postdoctoral period at the Johns Hopkins University, Baltimore, MD, he joined the University of Maryland, College Park, as an Assistant Professor in 1999 as a part of the Microelectronics Group in the Electrical and Computer Engineering Department, where he is also a Co-Director of the Computational Sensorimotor Systems Laboratory and is a member of the Neurosciences and Cognitive Sciences Program at the University of Maryland. He is one of the directors of the annual Telluride Neuromorphic Engineering Workshop and is involved in the growth of this international research community. His general research interests are in computational neuroscience and the implementation of neural circuit architectures in very large-scale integration (VLSI)-based processors.