Two-Dimensional Position Detection System With MEMS Accelerometers, Readout Circuitry, and Microprocessor for Padless Mouse Applications

Seunghbae Lee, Student Member, IEEE, Gi-Joon Nam, Member, IEEE, Junseok Chae, Member, IEEE, Hanseup Kim, Student Member, IEEE, and Alan J. Drake, Student Member, IEEE

Abstract—A hybrid two-dimensional position sensing system is designed with microelectromechanical systems (MEMS) for padless mouse applications. The X/Y-axis acceleration of the user's hand movements is measured by two MEMS accelerometer devices. These acceleration values are pulsewidth modulated and converted into (X, Y) coordinates on the screen by integral operations performed on a microprocessor. The overall system consists of four major components: 1) MEMS accelerometers; 2) CMOS analog readout circuitry; 3) an acceleration magnitude extraction module; and 4) a 16-b RISC microprocessor. Fig. 1 presents the block diagram of overall hybrid mouse system. Two MEMS accelerometer devices are employed to measure X- and Y-axis acceleration of the movements from the user's hand. These acceleration values are pulsewidth modulated (PWM) by the CMOS analog readout circuitry and will be converted into (X, Y) coordinates on the screen by performing integral operations on a 16-b RISC microprocessor. In this design, we present a hybrid configuration system where each component is not based on the same substrate and has to be interfaced with others by means of external connection such as wire bonding. The MEMS device is designed in silicon-on-glass (SoG) technology developed at the University of Michigan. The analog readout circuit is designed in a dual-poly, single-metal process, and the digital microprocessor is designed in a 1.5-μm AMI process through MOSIS using SC MOS design rules. We will, however, discuss further the possibility of implementing all of these components as a monolithic system. Another interesting aspect of this design is that it can be handily extended into a three-dimensional (3-D) position detection system by adding an additional MEMS accelerometer to measure Z-axis directional movement.

The remainder of this paper is organized as follows. Section II presents the overall system architecture and explains each module in detail. Section III describes the verification and testing methodology of the system. Section IV presents statistical data of the final design, and Section V describes the future work relevant to monolithic implementation of all the components. Finally, concluding remarks are given in Section VI.

II. SYSTEM OVERVIEW

A. System Requirements and Constraints

Two system requirements stand out for consideration: speed and accuracy. The motion of human hands can be categorized into two kinds: voluntary and involuntary movements [1], [2]. The focus of our position detection system is to sense only voluntary motions, calculate acceleration of those motions, and convert it into position information. As can be seen in Table I, the highest frequency of hand movement is about 25 Hz, which is slow compared to the speed of modern microprocessors. The mouse system must be able to generate (X, Y) coordinate values and...
faster than this rate. Thus, the sampling rate of position information was chosen to be 100 Hz (every 10 ms), and, accordingly, the target clock frequency for the microprocessor was set to 100 kHz, thus providing enough operations (1000 clock pulses) to calculate the \( X/Y \) coordinates per sampling. In terms of movement accuracy, it is reported that the acceleration noise of about 15 mg (g \( m/s^2 \)) is generated when normal people hold their hands as motionless as possible [7]. This value sets the minimum resolution required for an acceleration detection system. Auspiciously, the MEMS accelerometer adopted for the current design provides its finest resolution due to its micrometer-level dimension and measurement. In a MEMS accelerometer, the sensitivity is defined as the capacitance variation per gravity change, and this variation is easily converted to voltage through analog readout circuits. Thus, the combination of MEMS accelerometer’s sensitivity, the gain of analog readout circuit, and its equivalent input noise determines the overall minimum detectable acceleration of the system. Taking this into account, the analog readout circuit design was carefully designed, fully tested for functionality, followed by layout versus schematic (LVS) and parasitic extraction for layout and timing verification. Then, those modules are stitched together to form a super-module at a higher level (bottom-up design). Only after all three verification procedures were satisfied were modules instantiated and used at a higher level of the design hierarchy. The majority of modules—the datapath of the microprocessor, analog circuitry, and MEMS device—are full custom designs while the microprocessor controller was designed and synthesized using Verilog HDL.

### C. Operational Overview

Fig. 1 shows an overall operational block diagram of the mouse system. When a user moves the mouse in an arbitrary direction, the acceleration is decomposed into \( X \)- and \( Y \)-axis components which are measured by a corresponding “MEMS accelerometer.” The measured acceleration value is represented by a differential capacitance between two capacitors in the MEMS device. This capacitance value is fed into the system in two different ways: 1) back to the MEMS device to reset it so that we can measure the next acceleration without any bias and 2) into CMOS analog readout circuitry which modulates the capacitance value into digital pulselength output voltage. The polarity of the pulse, whether it is positive or negative, determines the direction of the acceleration and the width of the pulse is proportional to the magnitude of the input acceleration. An “acceleration magnitude extractor” changes the generated pulselength into the units of system clock frequency, which will be used by a microprocessor to calculate the exact

---

<table>
<thead>
<tr>
<th>Human Hand Motion</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voluntary Motion</td>
<td>Under 2 Hz</td>
</tr>
<tr>
<td>- 99% voluntary movement [3]</td>
<td>1 Hz</td>
</tr>
<tr>
<td>- Writing and drawing motion [5]</td>
<td>As high as 6 Hz</td>
</tr>
<tr>
<td>Involuntary motion [6, 7]</td>
<td>3.8 Hz</td>
</tr>
<tr>
<td>- Normal hand Tremor</td>
<td></td>
</tr>
<tr>
<td>- Parkinson’s Disease</td>
<td></td>
</tr>
</tbody>
</table>

TABLE I
SPEED OF HAND MOTIONS
position coordinates on the screen. In the next section, each subcomponent is explained in details.

D. MEMS Accelerometer

Typically, there are three different types of MEMS accelerometers: piezoresistive [8], tunneling [9], and capacitive [10], [11]. The capacitive accelerometer was chosen for this system because it provides high sensitivity, good dc response, low drift, low temperature sensitivity, low-power dissipation, and a simple structure [12]. For implementation, SoG technology is used to take advantage of its robustness and simplicity of design.

Fig. 2 shows the structure of an SoG lateral accelerometer [10]. A MEMS accelerometer consists of two major components: a proofmass and sensing electrodes. A thick silicon proofmass is connected to a frame, which is anchored to a glass substrate with suspension beams and comb fingers are formed using deep reactive ion etching (DRIE). Since the proofmass is suspended over recess on a glass substrate with serpentine suspension beams, it is free to move with a response to external acceleration. On the other hand, electrodes are attached to the glass substrate.

When external forces are applied to the accelerometer—for example, by the user’s hand movement—the proofmass moves against the forced direction due to an inertia force while the electrodes are stationary. The movement causes capacitance variations between the comb fingers which form parallel-plate capacitors, denoted as \( \Delta C \). One side of the comb fingers generates a positive variation \( (+\Delta C/2) \), and the other side produces a negative one \( (-\Delta C/2) \). The total capacitance change is the difference between these two values \( ((\Delta C/2) - (-\Delta C/2)) \). Therefore, the external acceleration is converted to the differential capacitance variation which can be expressed as

\[
\Delta C(\text{pF/g}) = \frac{9.8 \cdot M}{k \cdot d_0} \cdot C_s
\]

where \( M \) is the mass of a proofmass, \( C_s \) is the rest capacitance, \( d_0 \) is the sensing gap distance, and \( k \) is the spring constant of suspension beams [11]. From (1), it can be expected that a heavy accelerometer with a large number of comb fingers and compliant structure—i.e., small spring constant—provides good sensitivity, although it is difficult to fabricate such accelerometers. The dimension of the fabricated single-crystal silicon proofmass is 2.1 mm \( \times \) 2.4 mm \( \times \) 100 \( \mu \)m (\( W \times L \times H \)) and weighs 0.25 mg. The sense capacity and sensitivity are approximately 16.4 pF and 0.78 pF/g, and the noise floor is 10 \( \mu \)g/Hz. For the 2-D mouse system, two lateral accelerometers are required for sensing X- and Y-axis accelerations. A schematic top view of the MEMS accelerometer in Fig. 2 is shown in Fig. 5(a).

The fabrication process has five steps requiring only three masks. Fig. 3 shows the cross section of the wafer for each fabrication step. In the first step, a recess is formed on the glass substrate [Fig. 3(a)]. Then, the glass wafer with recess is anodically bonded to silicon wafer [Fig. 3(b)]. Next, the silicon wafer is thinned to desired thickness with chemical mechanical polishing (CMP) [Fig. 3(c)]. Metal contacts are evaporated [Fig. 3(d)], and, finally, a deep DRIE etch is performed to release the structure [Fig. 3(e)]. Fig. 4 shows a scanning electron microscope (SEM) picture of a MEMS accelerometer taken after fabrication. MEMS accelerometers can be readily integrated to other circuits using post-CMOS processing [13]–[15].

E. CMOS Analog Readout Circuits

The detected acceleration, which is represented as a differential capacitive variation from the MEMS accelerometer as illustrated in Fig. 5(a), is modulated to pulsewidth by the analog readout circuitry as shown in Fig. 5. In general, the lateral accelerometer presented in Section II-D can be operated either in open-loop mode or closed-loop mode. In closed-loop operational mode, the overall performance such as linearity, dynamic range, and bandwidth are improved [11]. In our design, an electromechanical oversampled modulator is used because it provides direct digital output and force feedback control of the proofmass simultaneously over a wide dynamic range.
Another various class of electromechanical modulators, the switched-capacitor sigma–delta electromechanical modulator [Fig. 5(b)], is employed because it is insensitive to the input parasitic capacitance. An efficient gain- and offset-compensated integrator is obtained by using an offset-storage $C_o$. This technique is called the correlated double sampling (CDS) technique [16] and can reduce $1/f$ noise and offset in op-amp. The circuit uses a single-ended charge integrator to read out the capacitance variation, and the static comparator forms the loop quantizer. Finally, a digital flip-flop samples the output from the comparator and synchronizes its bitstream with the system clock. Since the proofmass can be modeled as a second-order system, the closed-loop system becomes unstable, so a simple lead compensator $H_c(z)$ is inserted in the feedback path for stability.

There are four clock phases in the switched-capacitor interface circuit [Fig. 5(c)]. During the first phase ($\phi_1$), the sense capacitors formed between electrodes and proofmass are reset and the op-amp offset voltage is stored on $C_o$. In the second phase ($\phi_2$), sense capacitors $C_{\text{right}}$ and $C_{\text{left}}$ are charged through $V_+$ and $V_-$. The charge difference between $C_{\text{right}}$ and $C_{\text{left}}$ is integrated on the feedback capacitor $C_i$, while the comparator quantizes the charge. In this phase, the offset voltage stored in $C_o$ is subtracted and thus the output voltage op-amp is compensated. The amplitude of op-amp output voltage corresponds to the magnitude of applied acceleration. For example, if the MEMS device gets positive acceleration, then the proofmass is closer to the left anchor of the MEMS accelerometer and $C_{\text{left}}$ is bigger than $C_{\text{right}}$. Since $C_{\text{left}}$ is connected to the negative reference voltage ($V_-$), the negative net charge is sampled to an integrator, generating positive op-amp output voltage. The capacitance sensitivity defined as the op-amp output voltage due to input capacitance charge ($\Delta C$) is calculated by

$$V_o = \frac{\Delta C}{C_i} \cdot (V_+ - V_-).$$

From (2), the capacitance sensitivity is approximately $0.33$ V/pF. The quantized output is also latched at the end of
In the third phase (φ₂), the output of the flip-flop is fed back to the MEMS accelerometer to place the proof-mass in the null position by electrostatic forces. If the proofmass gets positive (negative) accelerations, the output of comparator will be positive (negative) because of its voltage inversion. This supply voltage is reapplied to the proofmass to make it return to the null position by electrostatic forces. The last phase (φ₃) is used for the control and sensing of the proofmass. The output of the flip-flop forms pulse bitstreams for the next module.

Fig. 6 shows the HSPICE simulation result of the modulation of acceleration magnitudes into the corresponding digital pulse bitstreams (PWM signal). As the acceleration changes, capacitance differences between proofmass and electrodes are sensed and the corresponding charge is integrated with a 90° phase lag. The integrated charges are reflected to the voltage at the output and passed through the comparator generating a PWM signal.

The layout for analog readout circuit shown in Fig. 7 is composed of all the components necessary such as switches, amplifier, comparator, D-flip-flop, and clock generator.

**F. Noise Analysis of Closed-Loop System**

There are several noise sources that affect the resolution (i.e., the minimum detectable input acceleration) of the overall system. In an oversampled electromechanical sigma–delta system, higher sampling rates reduce the quantization noise. Each doubling of the sample frequency results in a 3-dB signal-to-noise ratio (SNR) improvement. Based on the sensitivity of accelerometer and capacitance, the equivalent input noise of acceleration can be calculated. Each noise source now will be described.

1) **Brownian Noise:** The primary mechanical noise source for the device is due to the Brownian motion (Fig. 8) of gas
molecules between comb fingers. The total noise equivalent acceleration (TNEA) \( [m/(s^2\sqrt{Hz})] \) \([12]\) is

\[
TNEA = \frac{\sqrt{4K_BT\Delta f}}{M} = \sqrt{\frac{4K_BT\omega_r}{QM}} \tag{3}
\]

where \( K_B \) is the Boltzmann constant, \( T \) is the temperature in Kelvin, \( D \) is damping coefficient, and \( \omega_r \) is resonance frequency. From (3), it is seen that the MEMS accelerometer has smaller Brownian noise with larger \( Q \) and heavier proofmass. Plugging device data into this equation results in Brownian motion noise around 10 \( \mu g/\sqrt{Hz} \).

2) Amplifier Noise: The readout circuitry utilizes correlated doubling sampling to reduce the input CMOS amplifier flicker noise. However, the amplifier thermal noise [Fig. 9(a)] is amplified by the ratio of total input capacitors, including parasitics in the integrating capacitor. The noise at the output of the amplifier can be expressed by integrating the total noise power and dividing it by the effective noise bandwidth, which is half of the sampling frequency. Therefore, the amplifier output noise voltage due to thermal noise in the op-amp is expressed as

\[
V_{\text{amp-out}} = \sqrt{\frac{16}{3}} \cdot \frac{C_T}{C_i} \cdot \frac{kT}{C_{\text{out}}} \cdot \frac{1}{f_s} \cdot [V/\sqrt{Hz}] \tag{4}
\]

where \( C_T \) is the total input capacitance including parasitics, \( C_{\text{out}} \) is the capacitance of the op-amp output node, and \( f_s \) is the sampling frequency. From this equation, the equivalent input acceleration noise due to amplifier thermal noise is about 0.7 \( \mu g/\sqrt{Hz} \).

3) \( KT/C \) Noise: A major noise source in switched capacitor circuits is \( KT/C \) noise [Fig. 9(b)], which is generated by the CMOS switch thermal noise. The rms voltage noise from thermal switch noise can be calculated by the integration of the bandwidth of switch’s \( RC \) filter. The voltage noise power density due to this switch thermal noise is expressed as

\[
V_{\text{SC-out}} = \sqrt{\frac{KT}{C_{\text{out}}} \cdot \frac{2}{f_s}} \cdot [V/\sqrt{Hz}] \tag{5}
\]

This \( KT/C \) voltage noise is converted to the equivalent input acceleration noise as 0.3 \( \mu g/\sqrt{Hz} \).

4) Mass Residual Motion Noise: The proofmass is being rebalanced by a pulse train, and thus it has a residual motion with small ac amplitude (Fig. 10). The amplitude of this motion can be shown \([17]\) to be approximately equal to

\[
\Delta x = 4a_{\text{max}}/(\pi \delta f)^2 = 8.0 \times 10^{-9} \text{ m with } a_{\text{max}} = 20 \text{ g and } f_s = 100 \text{ kHz.} \]

This residual motion corresponds to an equivalent rms of 78 mg acceleration in this MEMS accelerometer. This acceleration can be randomized due to the varying input, and hence it can be considered to be noise distributed uniformly from dc to the proofmass residual motion frequency \( f_s/4 \). This noise is equivalent to 500 \( \mu g/\sqrt{Hz} \) and becomes dominant due to its low sampling frequency.

5) Dead-Zone Noise: Assuming that the input of the accelerometer is zero, the feedback voltage generates \( \alpha_{\text{max}} \) with frequency \( f_s/4 \) because of electrostatic force. Therefore, the input signal must be large enough to break this dead-zone pattern. The minimum rms input acceleration for this is

\[
\alpha_{\text{deadspace}} = \frac{8\alpha_{\text{max}}(\omega_r/\omega_s)^2}{f_s^2} = 160 \mu g
\]

and should be counted as noise.

6) Total Noise and Minimum Detectable Acceleration: Since all of the noise sources considered here are uncorrelated, the total noise is obtained simply by summing all of the noise sources, resulting in 530 \( \mu g/\sqrt{Hz} \). Therefore, the minimum detectable acceleration is the same as the one calculated by integrating this input acceleration density over the bandwidth of interest, which is \( \sim 100 \text{ Hz.} \) The resultant minimum detectable acceleration is about 5.3 \( mg \), which is below the smallest acceleration a human hand can generate/perceive. From the noise-source consideration, we see that the mass residual noise dominates with order of magnitude. However, this mass residual motion noise can be significantly reduced by increasing the sampling frequency at the cost of power consumption and complexity.

G. Acceleration Magnitude Extractor

The modulated pulse bitstream from the CMOS analog readout circuit is converted into binary data for further processing. Two types of data should be extracted from pulse bitstreams: polarity and magnitude. The polarity indicates the direction of acceleration and can be handily extracted by observing the sign of pulse bitstreams. For example, Fig. 11 illustrates the movements of the mouse and corresponding acceleration values on both the \( X \) and \( Y \) axes. It also shows the possible PWM bit streams. If the value of the pulse is 5 V (digital value “1”), the acceleration is toward the positive direction, and vice versa. The width of pulse bitstreams is proportional to the magnitude of acceleration. The extraction of correct magnitude of acceleration can be achieved via two binary flags (flip-flops) and a binary counter, as shown in Fig. 12. The “C” flag is set to the polarity of current input pulse bitstream synchronized by the system clock while the “P” flag value is the one shifted from the “C” flag flip-flop from the previous clock cycle. The comparator compares the two flag values and, when-
ever two flag values differ—e.g., the polarity of input stream changes—it loads the current value of binary counter into the buffer register called “acceleration register” and, at the same time, it resets the counter. While “C” and “P” flag flip-flops have the same value, the counter keeps increasing or decreasing based on the value of the “C” flag. Thus, the magnitude of generated values is proportional to the degree of the acceleration of movements, and the polarity of the input bitstreams determines the direction of user’s movements. In other words, the positive value of the counter corresponds to the positive acceleration, and vice versa. The “acceleration register” always keeps some static values specifying the previous acceleration magnitude, and the control microprocessor accesses only this register to calculate the current position cursor. Whenever the acceleration register loads a new value from the counter, the “N” flag flip-flop is set to “1,” indicating that the new value has been loaded. The control microprocessor keeps polling “N” flag flip-flops to decide whether it should fetch a new acceleration value or not. The layout for acceleration magnitude extractor is assembled together with core microprocessor and is shown in Fig. 16.

H. Core Control Microprocessor

The core control microprocessor is designed to filter the acceleration output from the analog readout circuits within the bandwidth of interest and to remove the quantization noise folded at higher frequency. This can be done with a simple finite impulse response (FIR) low-pass filter with ~100-Hz cutoff frequency. Also, the microprocessor calculates the position coordinates based upon the magnitude values of the acceleration from the acceleration extract module. The processor is based on RISC concepts and is implemented as a modified two-stage (fetch/decode and execute) pipeline (see Fig. 13). Instruction fetch takes place during the half cycle before the instruction is decoded to allow a full half-cycle for instruction memory access and a full cycle for decoding the instruction. The microprocessor uses a 16-b word and address space and all instructions are single-word. In addition to the basic arithmetic and logic operations, two application-specific instructions are implemented, which will be described below.

The main part of the application program consists of two procedures: polling and integration. For the polling procedure, a special instruction is defined called load counter value (LCNT).
When LCNT is executed, the core control microprocessor examines the value of the “N” flag flip-flop from the previous acceleration extraction module. If the value of the flip-flop is “1,” the value of the acceleration register is transferred into one of registers in the data-path module. At the same time, the “N” flag flip-flop is reset to “0” by the microprocessor to indicate the completion of the transferring operation.

The transferred acceleration magnitude value should be integrated twice so that it is transformed into position coordinates as shown in the following equation:

\[
 v = \int_{t_1}^{t_2} a \, dt \quad \text{and} \quad p = \int_{t_1}^{t_2} v \, dt = \int_{t_1}^{t_2} a \, dt \tag{6}
\]

where \( v, a, \) and \( p \) represent velocity, acceleration, and position coordinates, respectively. Instead of implementing the integration operation, it is performed via two normal addition instructions: \( v = v + a \) and \( p = p + v \). Since the system has abundant instruction cycles for each coordinate calculation, as pointed out in Section II-A, the lack of special-purpose hardware is justified.

The other special instruction is called SROUT, which moves calculated coordinate values to the 16-b system interface bus. To minimize the interface with outside circuitry, instruction memory (ROM) and data memory (RAM) have been integrated into the core microprocessor module. Both ROM and RAM contain 256 words, which is large enough to contain the entire application program.

I. Timing Scheme and Critical Paths

The general timing strategy of the core microprocessor is shown in Fig. 14. The clocking is predominantly positive edge-triggered, except for the program counter. The program counter is a negative edge-triggered component to allow enough instruction memory access time between \( t_1 \) and \( t_2 \). At time \( t_2 \), an instruction is loaded into the instruction register, and, between \( t_2 \) and \( t_3 \), the newly fetched instruction is decoded. In our design, every control signal is latched (as shown at time \( t_3 \)) to provide stabilized control signals for safe execution of the instruction.
during the next clock cycle from $t_3$ to $t_4$. In other words, the entire clock period is dedicated to execute one instruction. Finally, at time $t_4$, the result of the execution is written back to either the register file or the data memory. The rationale behind this timing strategy is to distribute major timing loads to different clock cycles evenly, resulting in the reduced clock period.

Our microprocessor design does not have separate memory address register (MAR) or memory data register (MDR), because both instruction memory (ROM) and data memory (RAM) are integrated together with the processor. Thus, memory access instructions (Load/Store) are virtually the same with a register transfer instructions (Mov/Movi), which simplifies the system architecture.

Through extensive simulation, the paths from the register file to ALU were determined to be critical primarily because the register file is the largest component and the ALU has a ripple-carry adder (see Fig. 15). The ALU delay time is 53.7 ns, which is measured from the rising clock edge to the point when the calculated data are valid on the system data bus. This critical path permits a maximum clock frequency of approximately 18 MHz, which is above the target frequency of 100 kHz. Recalling that the application target frequency dictates the maximum critical delay, the ripple-carry adder showed adequate performance in this regard, and a faster ALU was not necessary.

III. VERIFICATION AND TESTING

Verification and simulation were performed using the Mentor Graphics Tools and Epoch Design Compiler. Mentor’s QuickSim Pro was used for transistor-level and Verilog simulation. Parasitics were extracted from the layout, and Mentor’s Accusim was used for the detailed delay and loading effects. Timing information from the parts generated in Epoch [23] was extracted automatically during the import process to Mentor’s Design Architect. Once the functional models were properly back annotated, timing verification was performed in QuickSim Pro. In order to physically verify the design, full-mask LVS was performed by importing the custom datapath into Epoch and generating a net-list for the entire core that includes the custom and Epoch-synthesized parts.

To facilitate testing, the program counter and instruction register were made scannable. The application-specific instruction SROUT moves the specified register value to 16-b output pins (SOUT[15:0]), allowing us to observe any arbitrary register values of the register file during the testing mode. These two design-for-testability features allow us to verify the contents of instruction/data memory efficiently. By applying instruction addresses at the program counter and observing the values of instruction register via scan-chain testing mode, the correctness of the instruction memory (ROM) were verified. Data memory (RAM) was verified by moving contents of a specific memory

Fig. 16. Microprocessor layout (5.6 mm × 5.8 mm).

Fig. 17. Process flow of monolithic implementation. (a) SOI wafer. (b) Analog/digital CMOS process. (c) Deep trench etching. (d) Release.
Fig. 18. MEMS accelerometer for monolithic implementation.

IV. DESIGN STATISTICS

MEMS accelerometer was designed to have characteristics such as sensitivity with 0.78 pF/g, dimensions as 2.1 mm × 2.4 mm × 100 μm (W × L × H), and a sensing capacitance of 16.4 pF. The analog circuit has an estimated power of 3 mW, capacitance sensitivity of 0.3 V/pF, and die size of 4.5 mm × 2.6 mm. Mechanical and analog simulation shows that the designed padless mouse system can detect acceleration as small as 5.3 mg, which is a sufficiently high resolution for mouse applications.

The microprocessor, as shown in Fig. 16, contains 53,634 transistors, and the die size is 5.6 mm × 5.8 mm. As described in Section II-I, the critical path was from the register file to ALU, and the corresponding maximum clock frequency was approximately 18 MHz. Although maximum achievable clock speed is 18.1 MHz, its operational frequency is same as the sampling frequency of the analog switched-capacitor sigma-delta readout circuits. The power consumption is estimated to be 10 μW at 100-kHz operational frequency.

V. MONOLITHIC IMPLEMENTATION

The hybrid 2-D position detection system has been proposed to combine MEMS accelerometers and digital circuits by means of switched-capacitor circuits using wire bonding. Due to the wire being bonded, however, these different modules in a system must interface with each other at board level. This not only limits the size of a system, but also contributes capacitive or resistive parasitics that may degrade the system performance [24]. For example, stray capacitance and inductance due to wire bonding and bond pads can be up to 1 pF/pad and 1 nH/mm, respectively. To circumvent these drawbacks, a system-on-chip (SoC) design and fabrication, where all of the modules are fabricated in a single wafer, are considered as a monolithic implementation. In other words, all of the necessary modules such as MEMS accelerometers, the analog readout circuit, and the microprocessor are to be integrated on the same wafer without any interfaces between them. The only major interface with outside circuits—other than VDD, GND, clock, and reset signals—is the parallel data bus which transfers positional coordinates calculated from the microprocessor. Since they are fabricated on the same wafer, thus eliminating complex steps for the interface, it greatly reduces the overall costs as well as complex steps for the interface.

With all of these advantages over the hybrid system, however, the monolithic system has several problems such as noise between analog and digital circuits, interconnections between MEMS devices and analog circuits, and so on. One of the main factors for a noise problem in monolithic system stems from the fact that the analog and digital circuits are sharing the same substrate. Although substrate has high resistance, it is not a complete isolator, and thus voltage variations in digital parts are capacitively coupled to analog circuits and thus may cause malfunctions of a system.

The noise problem due to the substrate coupling can be solved by using silicon-on-insulator (SOI) [18]–[22], which isolates one module from others by etching substrate boundaries of each module and making it isolated both physically and electrically. Also, MEMS accelerometers and analog circuitry can be connected to each other by making deep trenches. It is reported that the silicon substrate underneath interconnection materials (such as metals) can be removed in the process of deep trench etching because of a sidewall etching effect [14]. The detailed fabrication process flow for monolithic implementation combining MEMS devices and analog and digital circuits are basically post-CMOS fabrication, as illustrated in Fig. 17. The usual CMOS processes are done using SOI start wafers, as shown in Fig. 17(a) and (b). After CMOS processes, deep trenches are
made for the formation of proofmass of MEMS accelerometers as well as isolation between analog and digital circuitry [Fig. 17(c)]. Fig. 17(d) presents the release step by removing SiO$_2$ with buffered hydrofluoric acid to suspend the proofmass. During the release, the SiO$_2$ between analog/digital circuitry are also etched away. Finally, Fig. 18 shows the mask-level MEMS structure. The close-up view shows an interconnection line, contacts, suspension beams, and etch holes which help releasing proofmass during BHF etch process.

VI. CONCLUSION

In this paper, we presented a novel 2-D position detection system for padless mouse applications. The final system consists of four major components: 1) MEMS accelerometer; 2) analog readout circuit; 3) acceleration magnitude extraction module; and 4) 16-b RISC microprocessor. Each module was carefully designed, laid out, and verified through extensive simulation. The calculated and simulated minimum detectable acceleration is 5.3 mg with a 100-kHz low sampling frequency for low power consumption.

The hybrid multichip nature of the proposed system will limit the system size and contribute capacitive and resistive parasitics that may degrade the system performance. To circumvent these drawbacks, an SoC design and fabrication, where all of the modules are fabricated on a single wafer, is considered as future work in this area. Also, the system can be handily extended into a 3-D position detection system by adding an additional MEMS accelerometer to measure $Z$-axis directional movement.

REFERENCES

Junseok Chae (M’03) received the B.S. degree in metallurgical engineering from Korea University, Seoul, Korea, in 1998 and the M.S. and Ph.D. degrees in electrical engineering and computer science from the University of Michigan, Ann Arbor, in 2000 and 2003, respectively. From 2003, to 2005, he was a Postdoctoral Research Fellow with Wireless Integrated MicroSystems (WIMS), University of Michigan. He joined the faculty of Arizona State University, Tempe, in August 2005, where he is currently an Assistant Professor of electrical engineering. His areas of interests are MEMS sensors, mixed-signal interface electronics, MEMS packaging, and ultrafast pulse (femtosecond) lasers for micro-/nano-structures. He holds several U.S. patents. Dr. Chae, along with his colleagues, was the recipient of the First Place Prize and the Best Paper Award at the Design Automation Conference Student Design Contest in 2001 with the paper titled “Two-dimensional position detection system with MEMS accelerometer for mouse application.” He gave an invited talk at Microsoft Inc. regarding “MEMS technology for consumer electronic applications.”

Hanseup Kim (S’00) received the B.S. degree from Seoul National University, Seoul, Korea, in 1997 and the M.S. degree from the University of Michigan, Ann Arbor, in 2003, both in electrical engineering. He is currently working toward the Ph.D. degree in electrical engineering in microelectromechanical systems (MEMS) at the University of Michigan. He interned at LG Corporate Institute of Technology, Korea, in early 1999, where he participated in developing a microoptical switch. Since entering the graduate program in 1999, he has been developing a micro vacuum pump for use in a miniaturized gas chromatograph as well as several polymer-based microfabrication technologies such as wafer bonding and membrane transfer techniques. Recently, he has also been involved with the development of bidirectional micro loud-speaker and chip-cooling chip using acoustic thrust generation. His research interests include the design and microfabrication of micro actuators and MEMS systems as well as their integrations with circuits. Up to date, he has authored and coauthored 15 papers in refereed journals and conferences and has served as a technical reviewer for the Journal of Microelectromechanical Systems. Mr. Kim was the recipient of the Best Paper and First Place Awards at the 38th Student Design Contest of Design Automation Conference (DAC 2001) with four other coauthors.

Alan J. Drake (S’99) received the B.S. degree in electrical engineering from the University of Arizona, Tucson, in 1997 and the M.S. degree in electrical engineering from the University of Michigan, Ann Arbor, in 2000, where he is currently working toward the Ph.D. degree. Currently, he is with IBM Austin Research, Austin, TX. His research interests include low-power VLSI, resonant clock generation and distribution, and SOI technology.