

Design for MOSIS Educational Program (Research)

A DSP Enabled Microsystem with MEMS-LC Clocking for Cochlear Implants

Submitted April 6th, 2005



Eric D. Marsman, Robert M. Senger
University of Michigan
Electrical Engineering and Computer Science

Richard B. Brown
University of Utah
Dean of Engineering

1. INTRODUCTION

Increased integration of components into a single SoC will enable performance improvements in remote sensing and bio-medical applications. The microsystem proposed here combines an energy efficient microcontroller (MCU), a low-power DSP core, and a monolithic MEMS-LC clock reference into a single bulk CMOS SoC intended for use in cochlear implants (CIs). By merging all of these components, the area and power consumption of the system will be greatly reduced without sacrificing performance compared to present CI systems. In addition, the same SoC can be utilized in a remote environmental sensor [1] that is being developed concurrently at the University of Michigan as part of the Wireless Integrated MicroSystems (WIMS) center.

The proposed work constitutes a vast extension of a previously designed, fabricated, and functional microsystem described in [2]. The previous design won 1st place in the 40th DAC/ISSCC Student Design Contest. Modifications to the microcontroller architecture and instruction set will be detailed in this proposal. The CI DSP core will be presented as a major addition to the previous SoC. Also, significant performance improvements to the monolithic clocking scheme will be explained.

The next section gives a brief description of CIs, followed by the projected architecture of the microsystem, including the MCU, DSP, and micromachined LC tank clock reference. The last sections briefly describe the *MOSIS Educational Program* specifics including design, verification, testing, process, and packaging plans.

2. COCHLEAR IMPLANTS

CIs are medical devices implanted in patients who have a degeneration or absence of the sensory hair cells in the inner ear. In a properly functioning ear, frequency selective hair cells in the cochlea generate electrical signals that are detected by neurons and transmitted via the auditory nerve to the brain for processing. The purpose of a CI is to bypass non-functional hair cells by directly stimulating the underly-

ing nerve cells. As of 2002, over 59,000 people worldwide had received cochlear implants [3].

The components of a CI include a microphone, signal processor, implant electrodes, and batteries. The microphone, fitted into the patient's ear, captures the sound and converts it to an electrical signal. The signal processor calculates the type and level of stimulation to deliver to the electrodes, which are inserted into the cochlea. Large variations in signal processing algorithms exist due to diverse implementation methods and differing parameters among patients.

Typical commercial CIs use an off-the-shelf software programmable DSP to perform the signal processing. The work proposed here will custom develop a fully integrated DSP core to perform the Continuous Interleaved Sampling (CIS) algorithm shown in Fig. 1. CIS is by far the most popular processing algorithm among patients and manufacturers. Research studies have found it provides the highest speech comprehension rates among patients [4]. As shown in Fig. 1, a high-pass filter (HPF) attenuates vowel sounds before the bandpass filters (BPF) split the signal into n channels ($n=16$ for this design). Following envelope detection, nonlinear dynamic range compression is performed to map the electrical representation of acoustic signals to a representation of current levels. Lastly, non-overlapping bi-phasic pulses are generated and delivered through the stimulation electrodes as current pulses to the cochlea.

There are certain parameters in a CI that must be variable to allow the patient to achieve their maximum performance [5]. The most important of these are the filter cutoff frequencies, compression function, number of channels, channel to electrode assignment, pulse duration, and pulse rate. Our proposed CI microsystem allows individual patients to be custom fitted by programming each of these parameters.

3. PROJECT DESCRIPTION

Fig. 2 shows the proposed microsystem architecture consisting of the digital MCU, DSP core, and the CMOS-MEMS LC tank oscillator to provide an on-chip clock reference. The MCU includes a 3-stage pipeline, 16-bit data path,

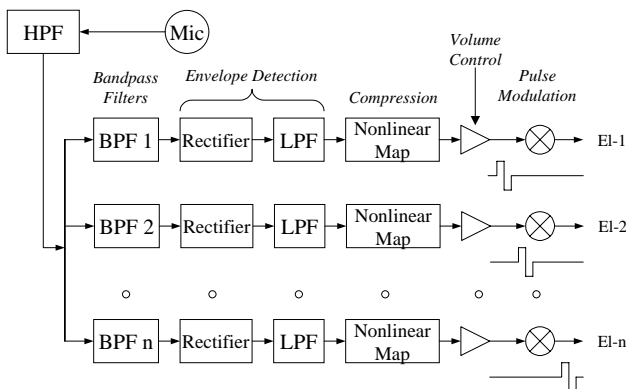


Fig. 1. CIS speech processing algorithm.

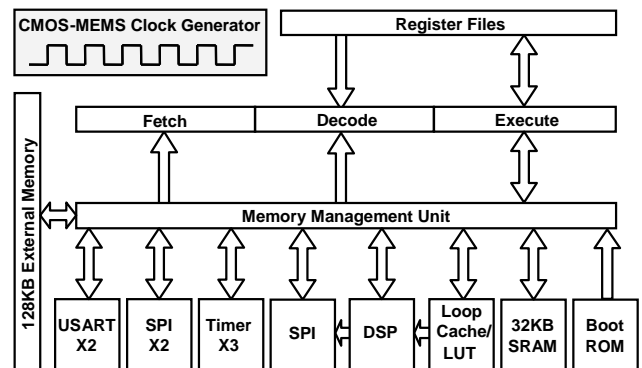


Fig. 2. Microsystem architecture.

a 24-bit unified instruction and data address space, 32KB of on-chip SRAM, and an external memory port supporting up to 128KB. Communication between the DSP and the microcontroller consists of shared memory space for coefficients, control, and data for the DSP and a shared serial peripheral interface (SPI) used for control of the implant electrodes. The interface to the CMOS-MEMS clock reference is a set of software controlled memory mapped registers to select the clock frequency that supplies the microsystem and perform debug and frequency trim. The following paragraphs describe each of the subsystems in detail

A. Microcontroller Architecture

The load-store instruction set architecture (ISA) contains 89 instructions supporting eight addressing modes and single- and multi-word signed arithmetic, shift, logical, and control-flow operations [6]. Instructions in the custom-designed ISA were carefully chosen to minimize decode complexity and power without sacrificing functionality. New instructions will alleviate compiler bottlenecks that existed in the previous ISA, primarily with address register manipulation, and further expand support for multi-word arithmetic. One level of interrupt and subroutine support is available in hardware. Nested interrupts and subroutines are enabled through software control of the hardware stack and frame pointer.

A 3-stage pipeline was chosen to provide adequate performance for remote sensing and bio-medical applications, yet still remain low-power with minimal pipeline hardware overhead. The pipeline utilizes sixteen 16-bit general purpose registers split between two data register windows and twelve 24-bit address registers split between two address register windows. Address and data windows will have separate control bits in the Machine Status Register (MSR) to provide the compiler maximum flexibility when changing windows. The windowing scheme reduces the register encoding field to enable 16-bit instructions while providing additional registers for temporary storage. [7] gives a detailed analysis of the WIMS compiler's efficient utilization of register windows to achieve up to 19% reduction in power consumption and 30% improvement in performance when compared to a non-windowed architecture. Address register manipulation is simplified by dedicated address register instructions or by using special address update modes.

The proposed memory architecture is a banked style with the 32KB of SRAM split into four 8KB banks. This allows instruction and data accesses to occur simultaneously without stalling the machine pipeline as long as they address different banks. This constraint is easily satisfied with minimal organizational control by the software compiler. In addition, this memory configuration allows for unused banks to be shut down on a cycle-by-cycle basis when not being accessed, yielding an overall power savings for the core. The

energy consumption of a banked 4x8KB configuration will be 48.7% less with an area penalty of only 17.2% when compared to a single 32KB bank.

Considerable power savings in the memory architecture are realized by the addition of a low-power, 512-byte loop cache [8]. Unlike traditional caches, the loop cache is a tag-less bank of low-power memory intelligently managed by the WIMS compiler [9]. The cache is intended to contain commonly executed instructions or accessed data, typically found in program loops. Contents of the cache are determined by the compiler and are not under hardware control, as is typical of memory hierarchy caches. It is still possible to change the contents of the loop cache at run time by loading the proper instructions or data into the cache and resuming program execution. The loop cache introduces minimal hardware overhead due to the banked memory structure, but yields significant power savings that were presented in [2].

Direct Memory Access (DMA) support will be added to reduce power and improve performance of frequent memory transfers. The DMA operations facilitate loading/storing groups of address and data registers to SRAM, a common operation required by the compiler to free-up registers and when jumping to subroutines or interrupt handlers. DMA support will also be provided to transfer blocks of code or data from main memory to the loop cache. All DMA accesses will stall the pipeline until the transfer is complete.

An HDL-synthesizeable dynamic clock frequency controller will allow the MCU to modify the system clock frequency on-the-fly to match workload requirements. This novel circuit is entirely coded in Verilog HDL and supports glitch-free clock frequency selection ranging from 78kHz up to 100MHz. The implementation will provide dynamic clock frequency control for both the MCU and the DSP cores separately, allowing each component to select its own clock frequency for optimal performance.

Serial interfaces and timer peripheral components provide the general-purpose functionality required by most embedded systems. A special serial test port facilitates remote, on-site testing of the MCU.

B. Digital Signal Processor Architecture

Fig. 3 is a block diagram of the proposed signed-magnitude fixed-point DSP core. The HPF, BPFs, and LPFs will be implemented as cascaded infinite impulse response (IIR) stages due to the low memory requirements and simplicity of the hardware. They will be 1st, 6th, and 4th order respectively. All filter coefficients will be programmable by the microcontroller to allow for the flexibility required for patient fitting procedures.

The compressed dynamic range, and therefore the number of bits required, enables saving power and area by reduc-

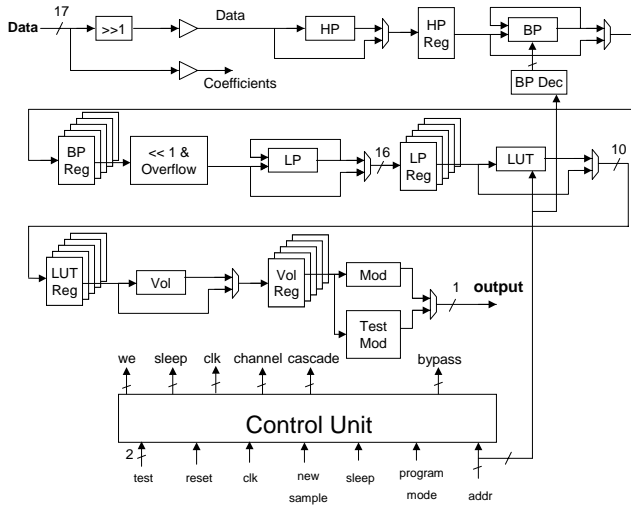


Fig. 3. CIS DSP architecture block diagram.

ing the datapath width from sixteen to ten bits. The parallel nature of the CIS algorithm provides for a significant reduction in hardware by pipelining the datapath and allowing all channels to share the same hardware for filters, LUT, and pulse modulation. The LUT data and the registers that control the pulse characteristics are also programmed via the microcontroller.

The DSP core will have four operating modes. For typical operation, the DSP will be in stimulation mode and will process samples and generate stimulation pulses. Programming mode will allow the microcontroller to set up all filter coefficients, LUT data, and the stimulation profile. Test mode bypasses the datapath stages via multiplexors at each output node to provide observability and controllability over each component in the system. Sleep mode shuts down components to conserve power. While in stimulation mode, any unused datapath stages are shut down through the control unit by utilizing the existing sleep mode circuitry.

C. Clock Generation Architecture

Clock sources for most SoCs and MCUs consist of a low-jitter, off-chip crystal reference with an on-chip phase-locked-loop (PLL) to multiply the off-chip reference frequency. The work proposed here utilizes a complementary, cross-coupled, negative-transconductance LC tank. A design overview of a previous generation of this low-jitter, CMOS compatible reference oscillator is given in [6]. The proposed LC tank implementation will occupy less than 0.3mm^2 of Si area, but will provide a significantly more stable high frequency clock signal than other on-chip clock generation technologies.

Improvements for this work will further reduce the active and standby power consumption. Standby energy will be reduced with the addition of a ring oscillator that can be selected by the microcontroller when a less stable, lower

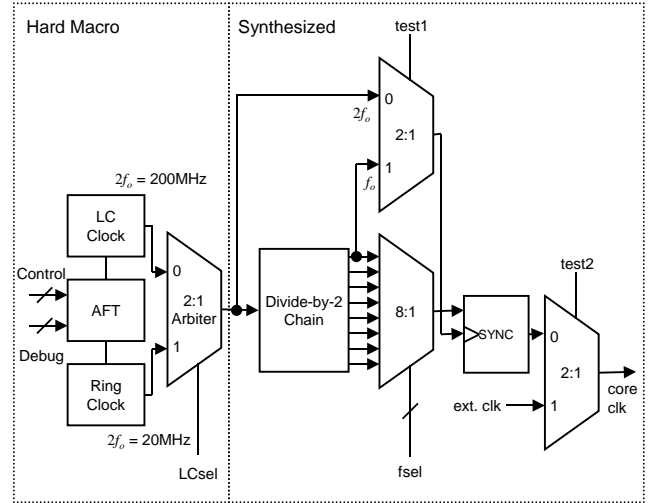


Fig. 4. Clock generation and selection diagram.

power, low frequency clock is desired. The MEMS-LC tank will be powered down, but will incur a 10-20ns wake up delay to select the more stable clock, and therefore increasing the SoC operating frequency. In addition, improvements in temperature compensation and post-fabrication frequency trimming will be implemented [10]. With the proposed hybrid clocking scheme, shown in Fig. 4, neither a PLL nor off-chip crystal are required, thus reducing total system size, cost, and power.

4. FABRICATION PROCESS

The microsystem will be designed in TSMC's $0.18\mu\text{m}$ MM/RF CMOS process with the Thick Top Metal option using the IP design methodology outlined in [11]. We are targeting the run closing May 16th, 2005. The MEMS-LC tank and ring oscillator will be delivered as a full-custom IP block and the microcontroller and DSP core will be developed using a standard ASIC design methodology with logic verification and timing closure. All three components will be treated as separate IP blocks to be integrated at the top level. [12] describes the top level integration in detail, using the previously mentioned IP along with Artisan memory and I/O pad IP.

5. PACKAGING REQUIREMENTS

We request that 25 of the devices be packaged in PGA132M or PGA145M ceramic packages from Kyocera with the remaining 15 in bare die form.

6. ESTIMATED PROJECT SIZE

The die size will be between 3.1 and 3.4mm per side.

7. SIMULATION, TEST, AND CHARACTERIZATION PLANS

Testing of the digital components will be done using an HP82000 D200 tester. The TDS software package supports direct translation of Verilog simulation vectors into control

signals applied to the chip. Focused test cases for each component of the microsystem, randomly generated instruction sequences, and compiled application code will be used for pre-fabrication verification and post-fabrication testing. In addition, performance metrics will be obtained by specific performance measurement code. The clock reference will be verified using direct observation of the clock output pin.

8. CONCLUSION

We have proposed a DSP enabled CI microsystem with a fully integrated MEMS-LC clock reference and a low-power ring oscillator option. Performance improvements explored by this SoC will advance the state-of-the-art in CI technology through reduction in overall CI size and increased battery life. The microsystem architecture remains general enough for use in other remote sensing and bio-implant applications. This work is the culmination of doctoral work for both Eric Marsman and Robert Senger.

9. REFERENCES

- [1] K. Wise, K. Najafi, R. Sacks, and E. Zellers, "A Wireless Integrated Microsystem for Environmental Monitoring," *ISSCC Dig. Tech. Papers*, pp. 434-435, Feb. 2004.
- [2] E. Marsman, *et al.*, "A 16-bit Low-Power Microcontroller with Monolithic MEMS-LC Clocking," *ISCAS*, May 2005.
- [3] National Institute on Deafness and Other Communication Disorders, "Cochlear Implants," <http://www.nidcd.nih.gov/health/hearing/coch.asp>, 2005.
- [4] Cochlear Corporation, "Nucleus Product Data," <http://www.cochlear-americas.com/Professional/141.asp>, 2003.
- [5] B. S. Wilson, *et al.* "Importance of Patient and Processor Variables in Determining Outcomes with Cochlear Implants," *J Speech Hear Res.* 1993,36:373-379.
- [6] R. Senger, *et al.*, "A 16-bit Mixed-signal Microsystem with Integrated CMOS-MEMS Clock Reference," *DAC*, pp. 520-525, Jun. 2003.
- [7] R. Ravindran, *et al.*, "Increasing the Number of Effective Registers in a Low Power Processor Using a Windowed Register File," *CASES*, Oct. 2003.
- [8] N. Bellas, I. Hajj, C. Polychronopoulos, and G. Stamoulis, "Energy and Performance Improvements in Microprocessor Design Using a Loop Cache," *Intl. Conf. on Computer Design*, pp. 378-383, 1999.
- [9] R. Ravindran, *et al.*, "Compiler Managed Dynamic Instruction Placement in a Low-power Code Cache," *Conference on Code-Generation and Optimization*, 2005.
- [10] M. McCorquodale, *Monolithic and Top-Down Clock Synthesis with Micromachined RF Reference*, Ph.D. Dissertation, U. of Michigan, 2004.
- [11] M. McCorquodale, *et al.*, "Microsystem and SoC Design with UMIPS," *IFIP International Conference on VLSI SOC*, Dec. 2003.
- [12] M. McCorquodale, *et al.*, "A Top-down Microsystems Design Methodology and Associated Challenges," 2003 *DATE Designers' Forum*, pp. 292-296, Mar. 2003.