

Continuous-Time Hybrid Computation with Programmable Nonlinearities

Ning Guo, Yipeng Huang, Tao Mai, Sharvil Patil, Chi Cao,
Mingoo Seok, Simha Sethumadhavan, Yannis Tsividis
Department of Electrical Engineering
Columbia University, New York, NY, USA

Abstract—We present the first continuous-time hybrid computing unit in 65nm CMOS, capable of solving nonlinear differential equations up to 4th order, and scalable to higher orders. Arbitrary nonlinear functions used in such equations are implemented by a programmable clockless continuous-time 8b hybrid architecture (ADC+SRAM+DAC) with activity-dependent power dissipation. We also demonstrate the use of the unit in a low-power cyber-physical systems application.

Keywords—continuous-time computation; hybrid computation; nonlinear function generation; nonlinear differential equation

I. INTRODUCTION

Emerging cyber-physical systems (CPS) often must operate on a very tight energy budget. Examples include sensor nodes powered by energy harvesting and tiny robots. Such systems often involve considerable computation, which must be done by an embedded, very-low-power computing unit. The required precision is limited, but the solution must be faster than real time and consume very little energy. For example, for an autonomous miniature robot to find optimal tracks, it needs to evaluate future states several times under different control inputs. This involves math operations, notably solving nonlinear differential equations (DEs) describing the system states, which are often functions of continuous time. However, when such equations are solved using traditional discrete-time computing units (e.g. microcontrollers), convergence issues may be encountered due to time discretization.

Recent work [1] has shown that analog computation, when done with today's VLSI technology, has several appealing attributes: computation is completely parallel, with computation time independent of the problem size; no convergence issues exist as no time-discretization is used – a significant advantage of analog computation over discrete-time computation. However, in that fully-analog work, the nonlinearities that could be used in computation were limited to only a few specific types, and approximated nonlinear functions using coarse piecewise linear interpolations introduced considerable errors. In addition, computation accuracy was poor because the input/output offsets of analog computing blocks (except for integrators) were not calibrated. In this paper, we present a hybrid (mixed analog/digital) computing unit that greatly mitigates these issues. Clockless digital circuits make arbitrary nonlinearities possible, and extensive digitally-assisted calibration corrects for analog imperfections. We demonstrate, for the first time, hybrid computation in continuous-time (CT) domain.

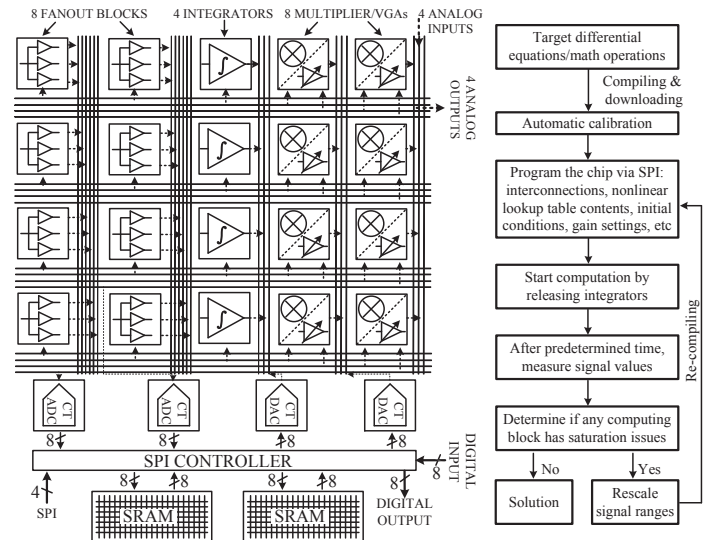


Fig. 1. Hybrid scalable computing unit architecture and its workflow.

II. SYSTEM ARCHITECTURE

The architecture of the chip and its workflow are shown in Fig. 1. As the unit is meant to be scalable, this test chip only includes a sufficient number of blocks to thoroughly test their function and interaction; it can solve nonlinear DEs up to fourth order. To keep interference low, the system is oriented from top to bottom as analog, mixed-signal and digital blocks, with deep n-wells used for substrate isolation. Differential current-mode signals are used for computation, with switch matrices located between blocks to allow programming of the current signal routing, as in [1]. The output of each analog block can be routed to the input of any other analog block. When the current output of one analog block needs to be connected to more than one inputs, fanout blocks are used to make copies of the current signal. The chip is programmed by an external control board over Serial Peripheral Interface (SPI). A given DE to be solved is mapped to a block diagram using some or all of the blocks in Fig. 1, connected in such a way that the system is characterized by the DE, with appropriate time scaling and with initial conditions imposed. The currents corresponding to the state variables in DEs represent the solution which can be directly routed off-chip, or be digitized by on-chip ADCs and sent back to the control board through SPI or parallel digital outputs.

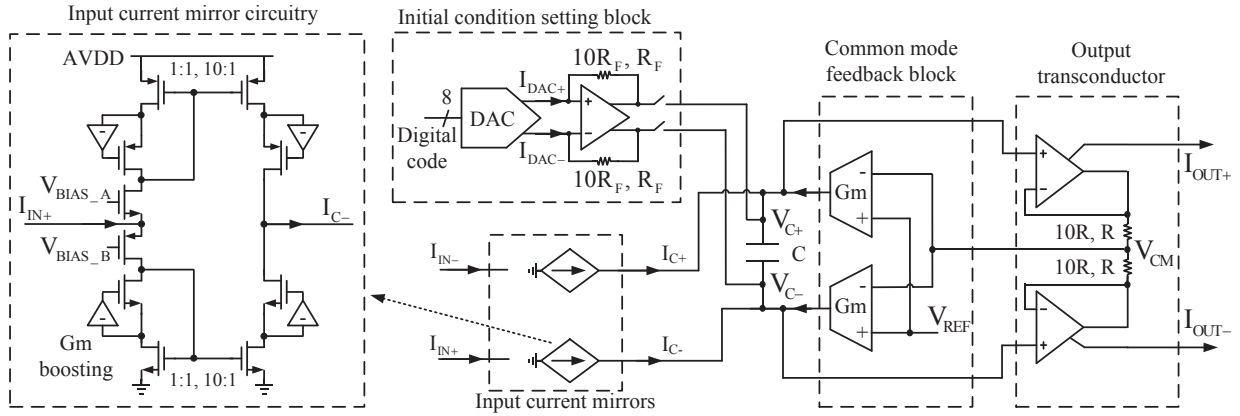


Fig. 2. Integrator core.

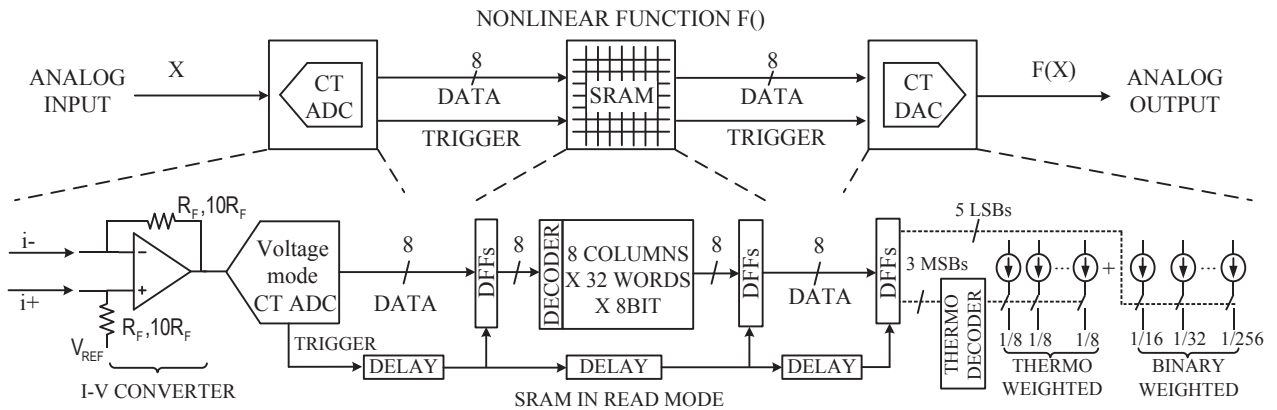


Fig. 3. Continuous-time programmable nonlinear function generator.

All non-analog signals and circuits involved in hybrid computation are CT digital ones, a type that has been demonstrated in signal processors [2]. Such circuits involve digital signals that, although binary, are functions of continuous time, with their time details being an integral part of the signal representation, thus carrying more information than conventional digital signals including asynchronous ones, and avoiding aliasing [2]. To our knowledge, this is the first time that CT digital signals are used in hybrid computation.

III. INTEGRATOR DESIGN DETAILS

The integrator architecture used in the hybrid computing unit is shown in Fig. 2. As compared to the log-domain integrator architecture in [1], the DC gain of the integrator described here is much less sensitive to device mismatches. As shown in Fig. 2, input differential currents are mirrored by a class-AB stage and driven onto a capacitor. The integrated voltage on the capacitor is then converted back to a differential output current by two transconductors, thus allowing interfacing with other current-mode analog blocks. The common-mode feedback block, consisting of two transconductance amplifiers, maintains the capacitor's common-mode voltage with respect to ground. The output transconductor stage consists of two current-copying OTAs connected in negative feedback. The current-copying OTA, based on the design in [3], has two identical outputs; one current output is used for voltage feedback by driving the

resistor load in the common-mode voltage feedback path, as shown in Fig. 2, and the other is used as the integrator core's output.

The input mirrors and the output transconductors have configurable gains, which allow 3 signal ranges ($\pm 0.2\mu\text{A}$, $2\mu\text{A}$, $20\mu\text{A}$) and 3 choices of unity gain frequency (2kHz, 20kHz, 200kHz). Initial conditions on the capacitors are imposed through transimpedance amplifiers driven by 8-bit current DACs. Input and output DC offset currents are digitally calibrated by 6-bit current DACs to enhance the accuracy of the integrator for computation.

IV. PROGRAMMABLE CT NONLINEAR FUNCTION GENERATOR

A key feature of our hybrid computing unit is the programmable nonlinear analog function generator. The architecture is shown in Fig. 3. It consists of a CT ADC, a CT SRAM and a CT DAC. The 8-bit ADC can convert full-scale signals up to 20kHz in two selectable signal ranges ($\pm 2\mu\text{A}$, $20\mu\text{A}$); as shown in Fig. 3, an I-V converter is used at the input stage with two configurable gains ($150\text{mV}/\mu\text{A}$, $15\text{mV}/\mu\text{A}$), followed by a voltage-mode level-crossing ADC (0.6V full scale) with a feedback R-string DAC, similar to that in [2], but using a Gray code counter and a decoder to replace the shift-register arrays in the feedback path, minimizing the digital switching power and noise. The 8-bit address/word length

SRAM design is based on the 10T cell in [4], but with a CT digital data path. In read mode, shown in Fig. 3, the input trigger signal is passed through delay lines; after the data have been read out from the SRAM banks and have settled, the trigger signal triggers the output DFFs and allows the data to go to next stage. The 8-bit DAC uses a segmented current-steering architecture with two configurable ranges ($\pm 2\mu\text{A}$, $20\mu\text{A}$); thermometer coding is used for the 3 MSBs to ensure monotonicity. Glitches generated by the DAC are reduced to negligible levels by the follow-up integrators in the overall computing diagram. Since the conversion of the input and output analog signals is done in CT, this scheme works in real time and avoids introducing time-sampling errors and aliasing into the generated analog functions. Two examples of nonlinear analog function generation and their errors compared to ideal values are shown in Fig. 4(a); the full cycle ($-\pi$ to $+\pi$) sine and sigmoid table lookups have normalized RMS errors of 0.0056 and 0.0076 respectively. The total power dissipation of this nonlinear function generator is signal-dependent, decreasing as the table lookup activity decreases, as shown in Fig. 4(b).

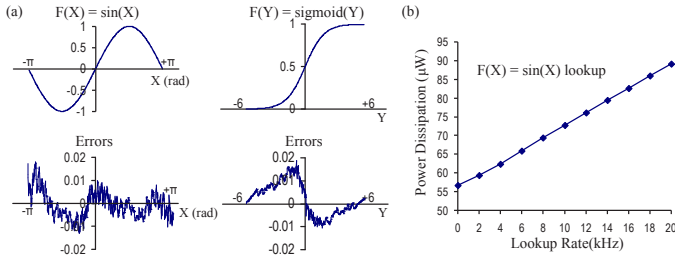


Fig. 4. (a) Nonlinear function lookup example. (b) Power dissipation.

V. MEASUREMENT RESULTS

The test chip (Fig. 5) has been fabricated in TSMC 65nm LP CMOS technology, as compared to $0.25\mu\text{m}$ CMOS used in [1], suggesting that analog/hybrid computation circuits can be scaled down together with digital circuits. The active area is 2.0 mm^2 , including circuits used for testing purposes and general programmability; the area would be considerably smaller if only special-purpose computation tasks were targeted. Input/output offset and gain calibration, by 6-bit thermometer-code current-steering DACs, are done automatically for all analog and mixed-signal blocks upon startup; the accuracy of solving nonlinear DEs is greatly improved by calibration, as shown in Table I. For testing purposes, the logic circuits for automatic calibration are off-chip. With the external SPI clock running at 20 MHz, the entire calibration at startup takes 4.1ms.

A performance summary of the hybrid computing system is shown in Table II. Compared to the previous effort in [1], our chip is approximately more than $18\times$ energy-efficient while providing better accuracy and more functions for computation, thanks to technology scaling and extensive use of class AB operation in analog blocks; a detailed comparison is given in Table III.

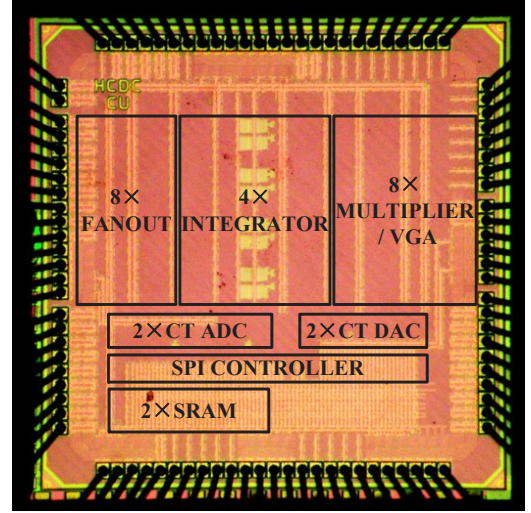


Fig. 5. Chip photo.

TABLE I. ACCURACY IMPROVEMENT FOLLOWING CALIBRATION

No.	DE's physical background	Nonlinearity involved	RMS error* (uncalibrated)	RMS error* (calibrated)
1	Van der Pol oscillator	Multiplication	17.7%	1.9%
2	Large angle motion of pendulum	Trigonometric function (sine)	7.3%	1.5%
3	Mass-spring dampers with Coulomb friction	Sign function	18.0%	0.5%

*Relative to full sale

TABLE II. HYBRID COMPUTING UNIT PERFORMANCE (27°C)

Supply voltage	1.2V	Block name	Power
Technology	TSMC 65nm LP	Fanout ⁴	37 μW
Die area / active area	$3.8\text{ mm}^2 / 2.0\text{ mm}^2$	Integrator ⁴	28 μW
Number of integrators	4	Multiplier ⁴	61 μW
Number of multipliers/VGA	8	VGA ⁴	49 μW
Number of fanout blocks	8	CT ADC ⁵	54 μW / 82 μW
Number of CT ADC	2		
Number of SRAM	2	CT DAC ⁵	4.6 μW / 15 μW
Number of CT DAC	2		
Number of analog inputs/outputs	4/4	SRAM ⁶	20 μW
Digital input/output word length	8 bits	Analog circuits leakage	6.7 μW
Programming interface	SPI	Digital circuits leakage (estimate)	85 μW
Integrator nonlinearity ¹	0.44%		
Fanout nonlinearity ²	0.13%		
VGA/Multiplier nonlinearity ³	0.15%		
ADC+DAC SNDR 1kHz/20kHz	46.3dB/53dB		
DAC DNL/INL	0.73LSB/0.67LSB		

¹ $2\mu\text{A}$ range, full-scale 20kHz sine input.

² RMS deviation from unity gain over $\pm 85\%$ full scale.

³ RMS deviation from unity gain over $\pm 85\%$ full scale in VGA mode.

⁴ $2\mu\text{A}$ range, full-scale sine input.

⁵ $2\mu\text{A}$ range, 1kHz / 20kHz full-scale sine input.

⁶ 20kHz full-scale sine digital input from ADC; SRAM programmed as a linear lookup table.

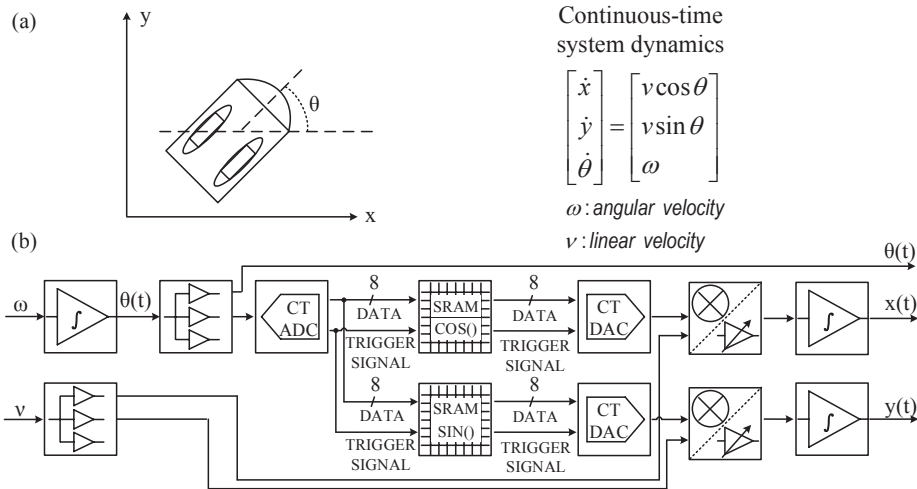


Fig. 6. (a) Differential-drive robot system dynamics. (b) Block diagram for solving system dynamics in the hybrid computing unit.

TABLE III. COMPARISON TO PREVIOUS WORK

	One macro in [1]	Our chip
Supply voltage	2.5V	1.2V
Technology	250nm CMOS	65nm CMOS
Active area (estimate)	6.3 mm ²	2.0 mm ²
Number of function blocks	25	26
Power with all blocks on (estimated)	18.8 mW	1.2 mW
Programming interface	Non-standard	SPI
Programming environment	Simulink	Arduino IDE
Calibration	Integrators only	All blocks, automatic
Computation types	CT analog only	CT analog / CT hybrid
Nonlinearities available for computation	Specific types	Arbitrary
On-chip ADC, SRAM, DAC	N/A	Available
On-chip digital controller	N/A	Available
Shut down of unused blocks	N/A	Available

VI. CPS APPLICATION EXAMPLE

We have successfully tested the chip using a variety of equations. As a CPS application example, we demonstrate modeling the system state of a tiny differential-drive wheeled robot (Fig. 6(a)) using model-predictive control [5]. Under a limited computing energy budget, the robot predicts the system state $(x(t), y(t), \theta(t))$ at a future instant under as many randomized inputs $(\omega(t), v(t))$ as possible, and the best input that minimizes a cost function is chosen as the actuator control. Our chip acts as a system dynamics simulator in this application. The system dynamics equations are mapped to the diagram shown in Fig. 6(b), and we program our chip to implement this diagram. In this example, we apply constant inputs (ω, v) to our chip and solve for the state 0.1s into the future. Our chip solves this in 0.84μs with 0.48nJ energy consumption and with 0.6% RMS error relative to full scale under 5000 random tests, which is acceptable for this application. We do not compare to [1] in this example as trigonometric function generators and digital calibrations are not available in that work.

VII. CONCLUSION

In conclusion, we have presented a new principle, namely CT hybrid computation, and the first reconfigurable CT hybrid computing unit with a scalable architecture in 65nm CMOS technology. The system has the ability to do clockless computation with arbitrary nonlinearities that are implemented by a continuous-time ADC + SRAM + DAC architecture, which demonstrates for the first time the use of CT digital signals in hybrid computation. No time discretization is used, thus avoiding convergence issues. The hybrid system has been successfully demonstrated to solve nonlinear differential equations. Extensive digitally-assisted calibration is used to improve analog computation accuracy. As an illustration of the chip's capabilities, a low-power CPS application example has been presented.

ACKNOWLEDGMENT

We thank Chien-Tang Hu, Doyun Kim, Jianxun Zhu, Teng Yang, Yang Xu, Yu Chen and Zhe Cao for valuable discussions. This work has been supported by National Science Foundation grant CNS 1239134.

REFERENCES

- [1] G. Cowan, R. Melville, and Y. Tsvividis, "A VLSI analog computer / math co-processor for a digital computer", Digest IEEE 2005 ISSCC, pp. 82-83.
- [2] B. Schell and Y. Tsvividis, "A clockless ADC/DSP/DAC system with activity-dependent power dissipation and no aliasing", Digest 2008 IEEE ISSCC, pp. 550-551.
- [3] Ade Putra, T. Hui Teo, and S. Rajinder, "Ultra Low-Power Low-Voltage Integrated Preamplifier Using Class-AB Op-Amp for Biomedical Sensor Application," IEEE International Symposium on Integrated Circuits, 2007.
- [4] D. Kim et al., "A 1.85 fW/bit ultra low leakage 10T SRAM with speed compensation scheme", Proc. IEEE ISCAS, pp. 69-72, May 2011.
- [5] G. Klancar and I. Skrjanc, "Tracking-error model-based predictive control for mobile robots in real time," Robotics and Autonomous Systems, vol. 55, no. 6, pp. 460-469, 2007.