A Fully Integrated Battery-Powered System-on-Chip in 40-nm CMOS for Closed-Loop Control of Insect-Scale Pico-Aerial Vehicle

Xuan Zhang, *Member, IEEE*, Mario Lok, *Student Member, IEEE*, Tao Tong, *Student Member, IEEE*,
Sae Kyu Lee, *Student Member, IEEE*, Brandon Reagen, *Student Member, IEEE*,
Simon Chaput, *Student Member, IEEE*, Pierre-Emile J. Duhamel, *Student Member, IEEE*,
Robert J. Wood, *Senior Member, IEEE*, David Brooks, *Fellow, IEEE*, and Gu-Yeon Wei, *Member, IEEE*

Abstract—We demonstrate a fully integrated system-on-chip (SoC) optimized for insect-scale flapping-wing pico-aerial vehicles. The SoC is able to meet the stringent weight, power, and real-time performance demands of autonomous flight for a bee-sized robot. The entire integrated system with embedded voltage regulation, data conversion, clock generation, as well as both general-purpose and accelerated computing units, weighs less than 3 mg after die thinning. It is self-contained and can be powered directly off of a lithium battery. Measured results show open-loop wing flapping controlled by the SoC and improved energy efficiency through the use of hardware acceleration and supply resilience through the use of adaptive clocking.

Index Terms—Dynamic voltage scaling, hardware accelerators, integrated voltage regulation, micro-aerial vehicle, system-on-chip (SoC).

I. INTRODUCTION

RECENT successful demonstrations [1] of unmanned aerial vehicle technology have fueled increasing interest in micro-aerial vehicles. These systems are envisioned for various applications, including reconnaissance, hazardous environment exploration, and search and rescue. Taking the pursuit of miniaturization to the next level, researchers are now working on "pico-aerial" vehicles (PAVs) that have a maximum takeoff mass of 500 mg or less and a maximum dimension of 5 cm or less [2]. One such prototype PAV is an insect-scale flapping-wing robot called the RoboBee, currently

X. Zhang was with Harvard University, Cambridge, MA 02138 USA. She is now with Washington University in St. Louis, St. Louis, MO 63130 USA (e-mail: xuan.zhang@wustl.edu).

M. Lok, S. K. Lee, B. Reagen, S. Chaput, P.-E. J. Duhamel, R. J. Wood, D. Brooks, and G.-Y. Wei are with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 USA.

T. Tong was with Harvard University, Cambridge, MA 02138 USA. He is now with Kolmostar Inc., Fremont, CA 94539 USA.

being developed at Harvard University. Recent demonstration of the RoboBee achieved controlled flight—hovering and maneuvering along three axes [1], but it was carried out using an external motion capture system, a bench-top high-voltage amplifier to energize piezoelectric (PZT) actuators that flap its wings, and a desktop computer for computation. The ultimate goal of autonomous flight requires converting these external test equipment into customized components within the robot's tight payload budget. Toward this end, we have developed a battery-powered multi-chip system optimized for insect-scale flapping wing robots [3] consisting of an energy-efficient system-on-chip (SoC) and a lightweight high-voltage power electronics unit (PEU) chip. This paper focuses the design of the SoC that acts like the central "brain" for the micro-robot to process sensor data in real time and coordinate wing flapping motions via control pulses to the actuators.

Since the total payload the RoboBee can carry is approximately 120 mg, weight becomes the central design constraint in our SoC design, and perhaps one of the most challenging. In PAV applications, other design metrics, such as performance and power, can be translated into weight. For example, autonomous aerial vehicles must perform certain amounts of computation within a real-time deadline to sustain stable closed-loop flight, and this minimum performance requirement will dictate the minimum power required. The higher the power consumption, the larger battery capacity required to sustain a fixed flight time. Therefore, higher performance and higher power invariably translate into a heavier battery that consumes a larger fraction of the limited PAV weight budget. After surveying currently available battery technologies and low power micro-processor solution space, we concluded that a general-purpose computing model is insufficient to deliver the power/performance demanded by the RoboBee, and specialized digital computation is needed.

We have adopted several techniques to significantly reduce the total weight of the system: a fully integrated voltage regulator (IVR) is implemented [4], avoiding the additional weight of an external voltage regulator module (VRM) and its associated discrete passive components; application specific hardware acceleration is used to reduce dynamic power and boost real-time performance [3], and supply noise resilient adaptive clocking scheme makes use of fully integrated clock generators and avoids external crystal oscillators [5].

The silicon prototype we developed not only fulfills the functionalities required by the RoboBee system, but can also serves as a general single-chip digital control solution for other PAV platforms.

Similar fully or highly integrated customized SoC approaches for targeted applications have been employed to meet the performance demands and satisfy extremely stringent power and form factor constraints. Examples can be found in contact-lens glucose monitoring [6], intraocular epirential prosthesis [7], energy-autonomous sensor node [8], ECG-based arrhythmia monitoring [9], and implantable seizure control device [10]. In these systems, power saving and size reduction are achieved via monolithic system integration of functionalities, including energy harvesting/power conversion, clock generation, and data conversion. Among the first attempts of an integrated solution for autonomous PAV applications, our SoC has incorporated similar integration strategies shared by these previous projects.

This paper is organized as follows. A brief overview of the RoboBee system architecture and an introduction of the unique design considerations in the PAV SoC are presented in Section II. We then describe the detailed circuit implementation in Section III. Finally, Section IV summarizes the measurement results from the fabricated chip in a 40-nm CMOS process.

II. RoboBee System Architecture

Despite its potential, miniaturization of autonomous aerial vehicles presents unique technological challenges. The most prominent one is how to design the electronic system that embodies all of the essential components, including power storage and conversion, wing actuation, vision sensing and processing, and autonomous flight control, within the size and weight limit of the pico-robot. In this section, we first give an overview of the RoboBee system and then derive the design considerations and performance specifications associated with its central digital controller.

A. RoboBee Overview

The RoboBee system was conceived as a technology exploration to alleviate the global crisis of declining pollinator populations. This biologically inspired pico-robot has similar size to a honey bee. Pictured in Fig. 1, the mechanical body of the RoboBee weighs less than 120 mg and can generate lift forces on the same order by wing flapping. It is orders-of-magnitude smaller than any previous autonomous flying robots, such as the nano-hummingbird [11].

RoboBee's wing motion is enabled by energizing the PZT bimorphs with sinusoidal waveforms near its resonance. A dual-actuator design can independently drive each wing. These actuators have higher power density and more robust mechanical performance compared with other competing options [12]. Equipped with the independently controlled dual-actuator wings, RoboBee enjoys three-rotational degrees of freedom in its aerial maneuvers. Fig. 2 shows the drive signals needed to achieve three degrees of freedom flight: roll torque is generated by asymmetric amplitudes in the two wings to induce differential thrust force; pitch torque by asymmetric
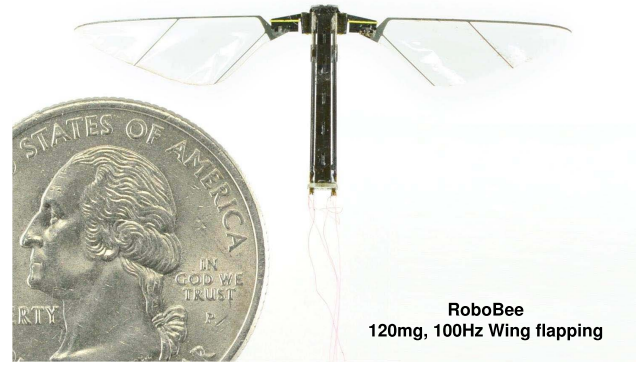

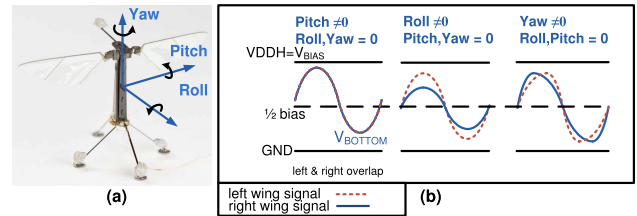Fig. 1. RoboBee is a biologically inspired bee-sized micro-robot.

Fig. 2. (a) Pitch, roll, and yaw rotation in RoboBee. (b) Desirable drive signal waveforms to achieve three-degree-of-freedom maneuver.

TABLE I
RoboBee Payload Estimate With Different Integration Solutions

Functional Components	Weight (mg)		
	Discrete Module	BrainSoC Integration	
		No Thinning	Die Thinning
Voltage Regulation	15 [17]	12	3
Digital Computation	220 [15]		
Clock Generation	3 [16]		
Data Conversion	20 [14]		
Power Electronics	80 [12,13]	43.5 [21]	32.25 [21]
Optical Flow Sensor	25 [19]		
Inertial Meas. Unit	23.5 [18]		
Total	306.5	104	83.75

offsets in the drive sinusoid waveforms to move the mean stroke angle, thus mimicking the method observed in fruit flies; yaw torque by modulating stroke velocity with distorted sinusoid waveforms to induce an imbalanced drag force per stroke cycle.

It is important to note that the RoboBee project is highly exploratory and involves many moving pieces at the same time. Therefore, our iterative design process is mainly guided by qualitative principles with a few first-order quantitative analysis. The design process starts with the mechanical structure. Once the RoboBee mechanical body design stabilizes, we are able to deduce the maximum payload, which in our case is 120 mg. Then, we estimate the non-battery payload in the RoboBee system, and as illustrated in Table I, our estimate highlights how the weight constraint becomes the central design consideration. It is clear that using off-the-shelf discrete components for each of the functional component would
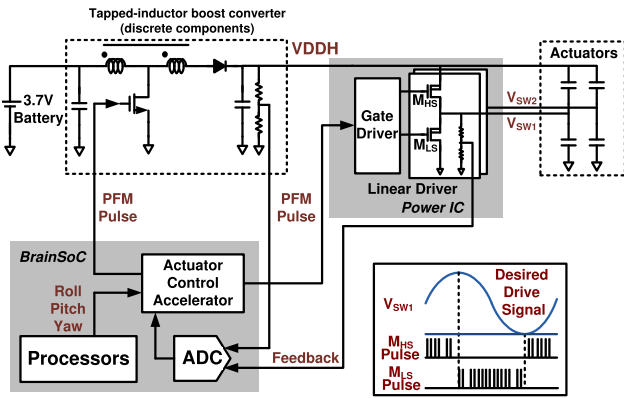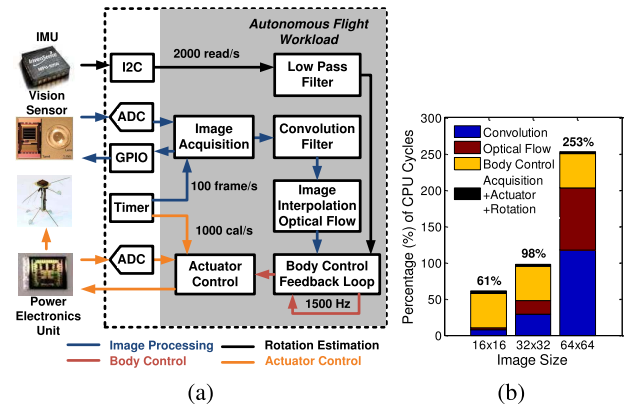
Fig. 3.   Two-stage design of the PEU.

Fig. 4.   (a) Concurrent workloads for RoboBee's autonomous flight. (b) Aggregated cycle count of the workloads as a percentage of M0 cycles at 200 MHz.

overwhelm the total payload of 120 mg, and on-chip integration of the electronic system that facilitates the full operation of RoboBee is imperative. In our design, we adopt a multi-chip strategy that consists of an SoC, optly named the *BrainSoC*, and a PEU with a separate power IC fabricated in high-voltage process. With the multi-chip integration, we are able to bring down the total weight of the payload to 104 mg, and with die thinning, an additional 20.25-mg weight can be shaved off, leaving the remaining 36.25 mg of the payload budget for battery.

Integrating lightweight high-energy-density battery onboard the RoboBee is still active research, because existing off-the-shelf battery technology cannot meet the weight and energy density requirements simultaneously. On the one hand, lithium-ion polymer battery has high energy density (360~875 mJ/mg) yet high minimum weight ($>$500 mg), and on the other hand, solid-state battery has low minimum weight ($<$20 mg) yet low energy density (10~36 mJ/mg) [20]. To decouple the development of the BrainSoC and the battery, we set a target energy density around 150 mJ/mg as a goal for our battery team, and use it as a guideline for setting the power budget and estimating the flight time. For example, given a fixed battery of 36.25 mg, if the RoboBee system consumes 100 mW on average, it can sustain flight for 54.4 s.

B. Actuator Driver

The PEU takes a 3.7-V battery voltage and generates 200–300-V sinusoidal signals to drive the PZT actuators at 80–120 Hz to match the mechanical resonant frequency of the wing structure. The top and bottom layers of the PZT actuator require 180° out of phase drive signals to produce maximum oscillatory behavior. The PEU [21] adopts a two-stage design, as shown in Fig. 3. The first stage is an off-chip tapped-inductor boost converter using discrete components to generate a 200–300-V high-voltage supply VDDH. The second stage, implemented via a 3.75-mg power integrated circuit chip, comprises two high-voltage half-bridge driver channels that operate off of VDDH, connecting to the middle node of the PZT actuators. Each linear drive circuit is controlled via pulse-frequency-modulation (PFM) pulses to produce sinusoidal drive signals [21], [22]. The PFM pulses are generated by a dedicated actuator control accelerator in the BrainSoC,

leveraging the abundance of digital logic and an embedded analog-to-digital converter (ADC).

C. Autonomous Flight Workload

In its most basic form, the required computational workload consists of four tasks that must complete within strict real-time constraints to sustain stable flight of the RoboBee.

1) *Image Processing:* Raw images are acquired from a specialized low power vision sensor [19] one pixel at a time and then are fed to an image processing pipeline consisting of convolution filter and optical flow. The first stage of the image processing uses convolution filter for noise removal and edge sharpening, and the second stage optical flow algorithm is able to estimate velocity and extrapolate the robot position based on velocity.

2) *Rotation Estimation:* An inertial measurement unit (IMU) supplements the vision sensor for rotation estimation with low-pass filtered gyroscope readings.

3) *Body Control:* The flight control algorithm relies on an adaptive proportional–integral derivative (PID) controller. It takes the position and rotation information from the sensors to determine the action needed to stabilize and move the robot.

4) *Actuator Control:* To perform agile maneuvers along all three rotational axes, the pitch, roll, and yaw calculated by the control algorithm must be translated into corresponding PFM pulse drive signals to actuate the wings.

Fig. 4(a) shows the interactions between these workloads and the sequence of processing. The real-time requirements are also highlighted. Image processing occurs at 100 frame/s and its speed is limited by both the computing capacity of the BrainSoC, which has to run the image processing algorithms in real time, and the maximum I/O bandwidth of the vision sensor, which sends out the image at a maximum rate of 1 pixel/μs. The maximum image resolution supported by the vision sensor is 128×128, and we vary the resolution settings from 16×16 to 64×64 in our experiment. Previous work [23] indicated that limited test results can be obtained at resolution as low as 1×32 in artificial laboratory environment with high color contrasts. The IMU is polled by the BrainSoC
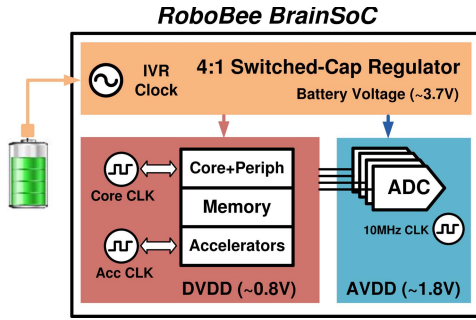
Fig. 5. System diagram of the self-sustained BrainSoC.

at 2000 read/s via its inter-integrated circuit (I2C) interface. The body control feedback loop runs at a loop frequency of 1500 Hz to ensure stable and yet responsive closed-loop flight control. Finally, the actuator control accelerator is activated every 1 ms with an updated set of pitch, roll, and yaw parameters to generate the desirable PFM pulses using a high-precision 10-MHz oscillator as the timing base. All of the real-time speed parameters listed in Fig. 4 have been identified during the initial flight tests.

D. BrainSoC

The focus of this paper is the design of the 3-mg BrainSoC, which is the central controller of the multi-chip system optimized for the RoboBee [3] and indispensable for achieving the ultimate autonomous flight experiment. To minimize weight and yet be self-contained, the SoC integrates a number of peripheral support circuits to obviate external components other than a battery. As shown in Fig. 5, a fully integrated 4:1 switched-capacitor (SC) regulator (IVR) steps the 3.7-V battery voltage down to an analog supply voltage of approximately 1.8 V and a digital supply voltage of approximately 0.8 V, resulting in three distinct voltage domains. On-chip oscillators generate multiple clock signals for general-purpose cores, hardware accelerators, peripherals, and the IVR. The 10-MHz relaxation oscillator [24] provides an accurate time base for the actuator controller to set the wing-flapping frequency, and it can also serve as the ADC's sampling clock. Both the 10-MHz clock and the ADC subsystem operate off the 1.8-V analog supply.

Once system integration allows us to fit the necessary functional components within the weight budget, the next step is to fulfill the computational requirement of autonomous flight under a fixed battery capacity. To evaluate the computational demand of autonomous flight on the core, we constructed a synthetic autonomous flight workload that represents a flight experiment by combining all of the computations discussed in Section II-C. If we assume that all computation is performed by a general-purpose core, such as ARM Cortex-M0, Fig. 4(b) presents the aggregate number of cycles needed to run the synthetic flight workload as a percentage of M0 CPU cycles at 200-MHz clock frequency. Image processing and body control clearly dominate the computation. The Cortex M0 core maxes out for an image size of 32×32 pixels. Finer image resolution, higher frame rates, or higher control loop update rates, all of which can improve flight performance, are not possible with M0 as the sole computational core.

Given the nontrivial computational requirements for autonomous flight, our design exploration concluded that a custom SoC with dedicated hardware accelerators provides a viable solution. After carefully analyzing the characteristics of different workloads in need of speedup, we decided to adopt two distinct strategies to design the hardware accelerators: monolithic and composable. Given their fixed and well-defined algorithms, convolution filter, image interpolation optical flow (IIOF), and actuator control accelerators were designed as dedicated monolithic accelerators that are capable of executing the entirety of the associated workloads. In contrast, the body control algorithm easily decomposes into simpler atomic arithmetic functions, such as dot product, matrix multiply, and finite impulse response filters, which can be implemented as a suite of atomic accelerator units in the DSP engine. Since the parameters and the controller structure used in the algorithm require constant adjustments and tweaking, its design is best left flexible. Therefore, we let the M0 orchestrate at the higher level, but then offload atomic arithmetic functions to the DSP engines. The Siskiyou Peak (SKP) core is added as a backup plan in case the accelerators fail to perform. It also provides additional data points for power and performance characterization in comparison between general-purpose cores and accelerators.

Anticipating the need to interface with a multitude of external sensors in a pico-robotic system, we have integrated versatile I/O serial protocol controllers on a peripheral bus, including I2C, serial peripheral interface, and general purpose input/output, as well as utility blocks, such as a programmable four-channel timer and an interrupt controller supporting up to 64 vectored interrupt sources.

III. CIRCUIT IMPLEMENTATION

In this section, we delve into the circuit-level implementation details of the three main parts in the BrainSoC prototype chip: the IVR, the digital computation block, and the clock generators. In each of these blocks, design choices were made to minimize weight, reduce power consumption, and improve system performance and/or reliability.

A. Switched-Capacitor Integrated Voltage Regulator

A DC–DC voltage regulator is necessary to convert the high battery voltage (3.7 V) down and deliver energy to the digital computate blocks in the SoC, and we chose an SC-based converter topology [4] in our design. SC converters are well suited for our application, since they only require capacitors and MOS transistors, thus obviating the need for off-chip inductor that consumes weight and area. SC converters typically operate alternately in two phases [25]: in one phase, energy drawn from the input charges the flying capacitor up and flows to the load; in the other phase, energy stored on the capacitor during the previous phase flows to the load.

Fig. 6 shows the system block diagram of the SC-IVR. The design cascades two 2-to-1 SC stages to achieve a conversion ratio of 4-to-1. The first stage connects directly to the battery and converts the 3.7-V battery voltage down to a 1.8-V intermediate voltage (V_{INT}). To handle the 1.8-V swing, this stage uses thick-oxide transistors available in the process.
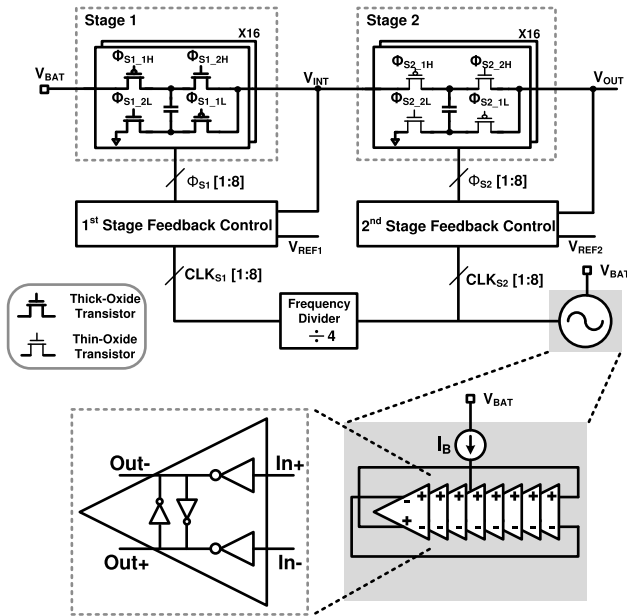
Fig. 6.   System diagram of the two-stage SC-IVR.

Fig. 7.   System architecture of BrainSoC.

The second stage converts the intermediate 1.8 down to 0.8 V for the final output (V_{OUT}) using thin-oxide transistors. Each stage includes identical, but separate feedback control loops.

Cascading two 2-to-1 SC stages offers two main advantages in our design. First, the intermediate output voltage of the first stage V_{INT} at around 1.8 V can work as the supply for the external vision sensor and IMU for RoboBee, and both V_{INT} and V_{OUT} can serve as stacked supplies for the switch drivers in each stage without additional voltage rails. Second, the two-stage topology offers an opportunity to optimize each stage separately. The topologies of the two SC stages are identical, but use different transistor types and sizing. Each SC stage implements a multi-phase topology to reduce voltage ripple. Sixteen modules operate off both edges of eight interleaved clock phases. A multi-phase differential voltage controlled oscillator (VCO) generates the clock edges and operates directly off of the battery to guarantee proper start-up operation. The bias current (I_B) of the VCO is generated by a supply independent biasing circuit [26] in order to reduce the VCO's frequency sensitivity to battery voltage. To ensure a balanced number of modules in operation, pairs of modules operate 180° out-of-phase off of one shared clock phase.

B. Hardware-Accelerated Digital Computation

As briefly described in Section II-D, digital computation in the BrainSoC is performed by a heterogeneous architecture including both general-purpose cores and hardware accelerators. In the digital domain (Fig. 7), a 32-bit ARM Cortex-M0 core handles general computing needs and is master of the advanced micro-controller bus architecture (AMBA) high-speed bus that connects to various memories and the AMBA peripheral bus. On-chip SRAM memories are organized in banks to save power from reduced memory I/O peripherial circuitry. Four single-port memories, 16 kB each, store instruction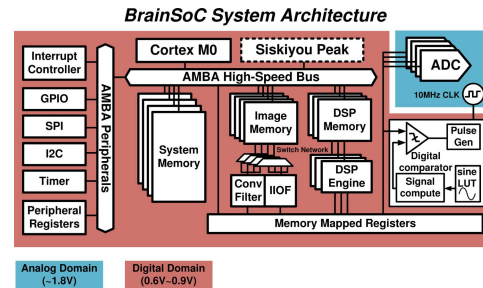s and general data structures managed by the core, while dual-port memories provide direct access to both the core and the accelerators for special-purpose data, such as the image from the vision sensor and the waveform lookup table for the actuator controller. While data transfer between the core and the accelerators is enabled by the dual-port memories, control signals for coordination from M0 to the accelerators are passed via memory-mapped registers on the main system bus.

To comprehensively compare the power/energy performance between general-purpose core and hardware accelerator, an Intel SKP processor was incorporated in the BrainSoC as a gated shadow core on the AMBA main bus. The SKP is a 32-bit core optimized for minimal area, power, and configurability and supports a subset of the x86 instruction set and system software model. We implemented it as a five-stage, single-issue, integer pipeline configuration. M0 can transfer over full control of the system to the SKP including the master control of the main bus, allowing us to compare power and performance using different micro-processors.

The overall architecture for the accelerator subsystem is shown in Fig. 7. Eight dual-port memories are dedicated to hardware accelerators: four 8-kB SRAMs are used as image memories to store the 8-bit pixel values, allowing access to the core and the convolution and IIOF accelerators; three 2-kB SRAMs, each directly connecting to a DSP engine, are used as scratch pad for basic arithmetic function acceleration; and one 2-kB SRAM works with the actuator control accelerator to store a lookup table. Since the image processing algorithms require large bandwidth and frequent data movement, we partition the image memories to interleave convolution and optical flow computations. As fresh pixel data come in from the vision sensor, it is put into an unused partition of one of the image memory banks. Taking advantage of the image processing pipeline, the filled image memory partition is then passed to the convolution filter, and the filtered results are written to a different partition, which becomes the next input to the IIOF accelerator. A programmable switch network implemented as sets of multiplexers is configured at run time to connect the active image memory bank to the corresponding accelerator, be it convolution filter or IIOF, so that the execution of image processing can be pipelined to achieve better performance.

The accelerator designs emulate fixed point with 32-bit integer operation by tracking the decimal point explicitly in software to save power and area. Configurability is built into the accelerators for flexibility. The convolution filter can be programmed via memory mapped registers: the size of the

convolution filter can be configured from 4×4 to 64×64; the filter constants can be reprogrammed on the fly; the filter window can be selected between 3×3 and 1×1; parts of the image can be discarded; support is built-in to conduct vertical or horizontal convolutions on the image data. The IIOF accelerator is configurable to return a 2-D vector, a vector field, or a set of vectors averaged over the images. It implements Lucas–Kanade algorithm [27] for computing optical flow and it is flexible to only operate on part of the image and has both 1-D and 2-D operation modes. The outputs of the image accelerators (convolution and IIOF), as well as those of the DSP engines, can be accessed by the core during subsequent execution of the body control algorithm. The image processing results are used in the outer loop to control position and latitude, and filter results of the IMU are used in the inner loop to control the attitude of the robot for upright stability [28].

The convolution filter, the IIOF accelerator, and the DSP engine were developed using the Xilinx Vivado High-Level Synthesis (HLS) tool [29]. Vivado HLS has built-in support to interface with the AMBA protocols and memory mapped registers by using an external finite-state machine controlled by the general purpose core. It is our experience that the performance difference between register-transfer level (RTL) generated by HLS versus hand-coded RTL is relatively small, so we are able to generate high-performing efficient accelerator designs by appropriately tuning the HLS directives from a high-level representation. On the other hand, the actuator control accelerator is generated from hand-coded RTL. The actuator control accelerator implements the digital feedback loop for both stages of the power electronics shown in Fig. 3. It operates off of the 10-MHz relaxation oscillator, and consists of a sinusoid compute block, a digital comparator, and a pulse generator block. At 100-kHz intervals, the sinusoid compute block calculates a new point on the sinusoid for the PEU to track, and this new point serves as a digital reference for the comparator. Then, the comparison result between this reference and the ADC output updates the pulse frequency of the outputs from the pulse generator block. This accelerator frees the processor from frequent timer interrupts to compute each point on the sinusoid at fixed time intervals, to bring the ADC outputs into the digital comparator, or to explicitly manage I/O pins for interfacing with the power electronics.

C. Digitally Controlled Clock Generator

In order to accommodate the stringent weight budget, we again avoided the use of any external components and exclusively relied on on-chip clock generators for the BrainSoC. In addition to the IVR clock and the 10-MHz clock for precise actuation control of the wing flapping [24], the system clocks required by the general-purpose computing core, the memory, the bus, the peripheral controllers, and the hardware accelerators are fully generated on-chip. We implemented two identical digitally controlled ring oscillators (DCROs) to enable flexible independent accelerator clock frequency be different from the core clock frequency.
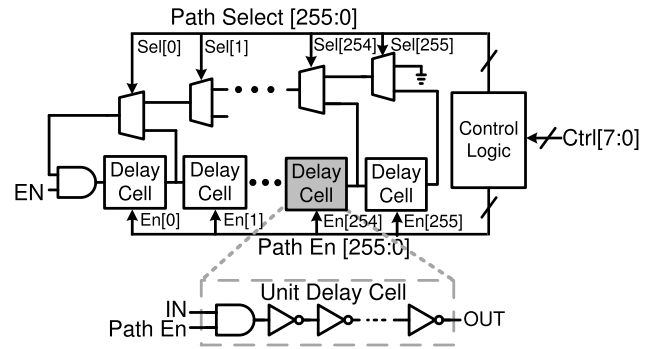
Digitally- Controlled Ring Oscillator (DCRO)

Fig. 8. System diagram of the DCRO.

We applied an adaptive clocking scheme for the system, since the DCRO is known to experience intrinsic sensitivity to the supply voltage. Advantages of higher performance and more robust supply noise tolerance of the adaptive clocking scheme have been explored in a previous prototype chip [30], and have proved to be beneficial for systems with integrated voltage regulation where static voltage ripples from switching regulators present large voltage margin requirement for a fixed-frequency clocking scheme [31]. We intentionally built this capability to seperate the accelerator clock from the core clock, because hardware accelerators offer tremendous performance improvements and thus provide opportunities to reduce power by scaling down its clock frequency. Each DCRO, shown in Fig. 8, implements a variable-length ring oscillator comprised of unit delay cells, programmable via an 8-bit control code. The control code sets the number of delay cells in the oscillator loop, with each unit adding approximately 125-ps delay.

To enable dynamic frequency scaling, the DCRO is designed to allow dynamic reconfiguration at run time via memory-mapped registers exposed to user code. Special care has been taken to ensure glitch-free operation, especially at the high-to-low frequency transition, when unknown values in the delay line could inadvertently inject high-frequency signals into the low-frequency delay loop, using the technique of augmenting the delay cell with an explicit enable signal [32]. The default system operating frequency is configured during initial post-fabrication testing of the BrainSoC so that the digital logic runs at its maximum operating frequency without incurring timing violation across the prescribed operating voltage range of the output voltage of the IVR by setting the DCRO to a nominal frequency of 220 MHz at 0.8 V.

IV. SYSTEM EVALUATION

The BrainSoC chip and a custom PEU make up the multi-chip system [3] designed to work together with an optical-flow vision sensor and an IMU sensor in the final RoboBee autonomous flight experiments. In this paper, we specifically focus on the BrainSoC chip to evaluate its performance in the context of potential PAV applications. The full assembly of a complete multi-chip system to be mounted on the RoboBee body and the ultimate experimental demonstration of RoboBee's autonomous flight are beyond the scope of this paper and require additional on-going work. Therefore, the results we present in this section is based on measurements
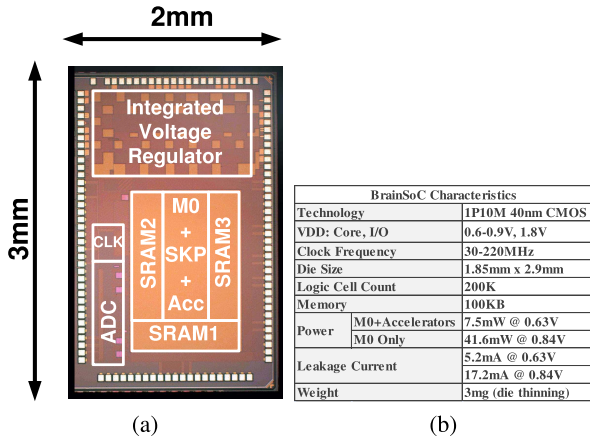
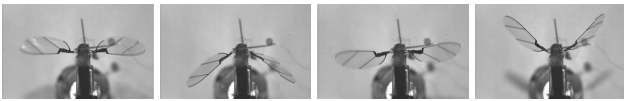Fig. 9.   (a) BrainSoC die photograph. (b) BrainSoC chip summary.

BrainSoC Characteristics		
Technology		1P10M 40nm CMOS
VDD: Core, I/O		0.6-0.9V, 1.8V
Clock Frequency		30-220MHz
Die Size		1.85mm x 2.9mm
Logic Cell Count		200K
Memory		100KB
Power	M0+Accelerators	7.5mW @ 0.63V
	M0 Only	41.6mW @ 0.84V
Leakage Current		5.2mA @ 0.63V
		17.2mA @ 0.84V
Weight		3mg (die thinning)

Fig. 10.   Slow motion video capture snapshots of RoboBee's right wing flapping.

Fig. 11.   Open-loop operation across V_{BAT} with 160-MHz switching frequency.

from a test board designed for debugging and characterization, and not intended for on-board flight.

The BrainSoC was fabricated in Taiwan Semiconductor Manufacturing Company's 40-nm CMOS technology using its standard digital process. The die photograph of the 2 mm × 3 mm chip is shown in Fig. 9(a). The table in Fig. 9(b) provides a summary of the chip features and characteristics.

A. System-Level Functionality

To verify the basic functions, we performed an open-loop wing flapping experiment, where the RoboBee is driven by the PEU to flap its wings with open-loop commands from the BrainSoC. Fig. 10 shows video capture snapshots of the RoboBee wing flapping, where the left wing was kept stationary and the right wing was driven by the PEU. It verifies the coordination between the PEU, the ADC, the actuator control accelerator, and M0 on the BrainSoC. Although M0 alone is sufficient for this type of open-loop wing control, autonomous flight requires closed-loop operation with compute-intensive image processing and feedback control that are not fully captured here.

B. Characterization of IVR

The IVR in the BrainSoC has been exhaustively tested in two modes: open- and closed-loop operation. In open-loop, the output voltage and output power can be tuned by changing the IVR switching frequency, f_{SW}. The first stage switching frequency is set to be 1/4 of that of the second stage. In the closed loop, an internal ring oscillator that clocks the IVR runs at its maximum frequency and the feedback loop implements single-bound control [33] to adjust the effective switching frequencies at both stages to regulate the output.

Fig. 11 shows how the output voltage changes with the load current at different V_{BAT} values when the converter
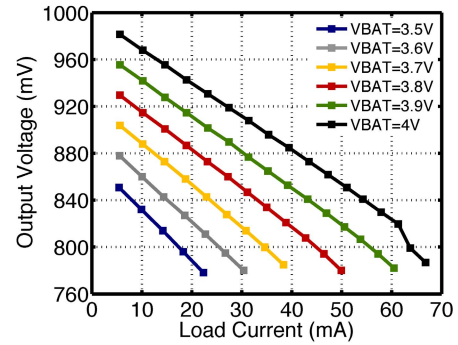
is operating in the open-loop mode at a peak switching frequency of 160 MHz. Output voltage decreases as load current increases because of the non-zero equivalent output resistance of the converter. The higher the V_{BAT} value is, the higher load current that the converter can deliver at a certain output voltage, because at every switching cycle, more charge can be transferred from input to output during the charge redistribution process at larger V_{BAT}.

Fig. 12 summarizes conversion efficiency versus output voltage at different V_{BAT} values. First, conversion efficiency is higher for open-loop operation, due to lower voltage ripple overhead. Second, conversion efficiency peaks at higher outputs when V_{BAT} is higher, since charge redistribution loss and switching loss are both related to the conversion ratio, V_{OUT}/V_{BAT}.

Fig. 13 presents the IVR's measured response to 47-mA output load transients using an on-chip load generator circuit with a rise and fall time of roughly 100 ps. As seen in Fig. 13(a), when the IVR runs in open loop with maximum f_{SW}, a 3–50-mA load step causes V_{OUT} to drop by 155 mV. When running in closed loop with the nominal output voltage set to 750 mV, however, the control loop reacts quickly and the voltage droop caused by the load current step is much smaller. The zoomed-in scope capture in Fig. 13 shows that the ∼60-mV droop is mostly due to the larger steady-state voltage ripple caused by higher output power.

C. Characterization of Hardware Accelerators

Next, we are interested in fully characterizing the power and energy efficiency of the hardware accelerators. Leakage current is quite significant in our design because of the high performance process chosen for fabrication. Although the BrainSoC is able to meet its power budget despite the leakage, we think further improvement is possible by moving to a low-leakage process and employing power gating techniques. We are unable to accurately measure the leakage contribution from different computational blocks, because they share the same power domain. A full treatment of leakage evaluation is left for a future version of the SoC that implements separate power-gated voltage domains. Instead, in this paper, we focus on evaluating the dynamic power/energy efficiency. By comparing the ratio of dynamic energy between different designs, which is independent of the process leakage, we are
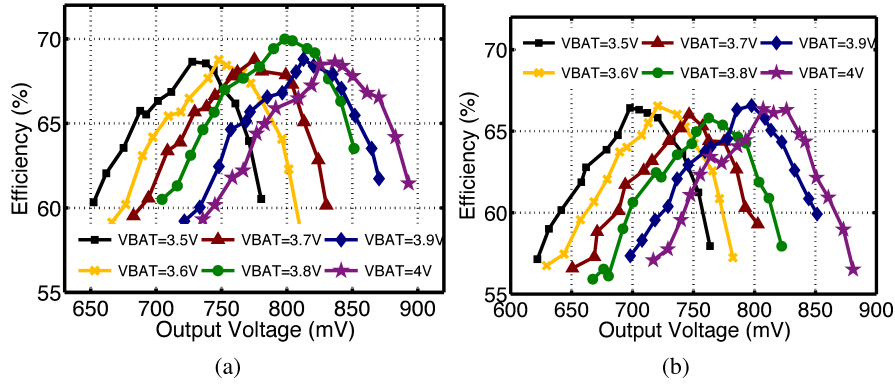
Fig. 12. Measured efficiency at different V_{BAT} and V_{OUT} values during (a) open-loop operation and (b) closed-loop operation.
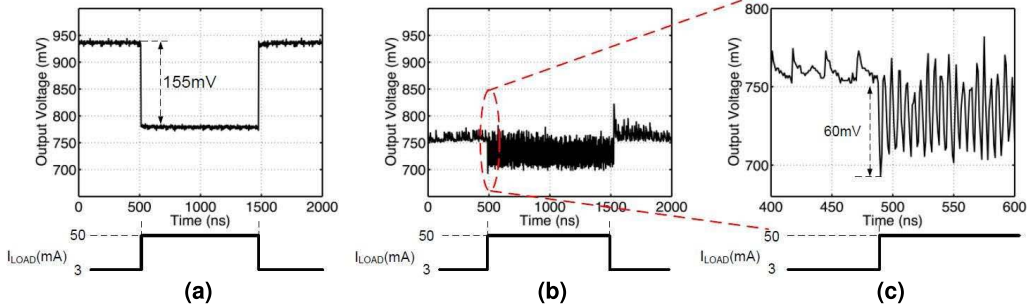

Fig. 13. IVR transient responses in (a) open-loop operation with maximum switching frequency, (b) closed-loop operation, and (c) zoomed-in transition region of (b).

able to derive an orthogonal performance measure to evaluate the digital implementation. Therefore, we subtract leakage from the total power by sweeping the operating frequency and extrapolating the leakage power that is insensitive to operating frequency.

Performance of the general-purpose cores is assessed by running micro-benchmarks that are constructed by emulating the same fixed-function computations carried out by the hardware accelerators. Due to the simplicity of the M0's in-order three-stage pipeline architecture, we find that its dynamic power is not a strong function of the workload, hence the overlapping linear lines in Fig. 14 showing the same amount of power consumed by M0 while running DSP functions, convolution filter, and IIOF. The same is true for the in-order SKP core.

The use of a separate accelerator clock and block-level clock gating allows us to independently control the operating frequency of each accelerator. Comparing the dynamic power in Fig. 14, we can see accelerators consume less dynamic power across the board than M0 executing the same functions in software. Simpler DSP functions have steeper power reduction of 6.6×, while complex image processing algorithms achieve 2.9× and 2.7× power reduction, respectively, for convolution and IIOF. Being a more complex design, the SKP consumes 1.6× higher dynamic power across the micro-benchmarks, as compared with M0.

However, dynamic power reduction does not fully capture the benefits of incorporating hardware accelerators, because significant computational speedup is possible through hardware acceleration. To account for both the reduction in power
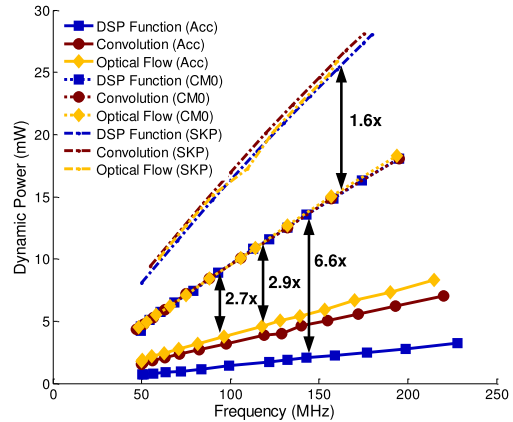

Fig. 14. Dynamic power of running different workloads in Cortex M0 (CM0) and accelerators (Acc) as frequency scales.

and execution time, we plot the dynamic energy for comparison between the software-only method of running the workload on M0 (or SKP) and versus hardware acceleration. As shown in Fig. 15, the atomic arithmetic functions in the DSP engine yield 10× energy improvements, whereas the monolithic implementation style of the convolution filter and IIOF accelerators offers over two orders of magnitude improvement, as shown in Fig. 16. This is true for comparisons with both M0 and the SKP. These measured results unequivocally confirm the computation efficiency advantages of monolithic hardware accelerators. Differences do emerge among the workloads, when comparing the improvement between the two general-purpose cores. For example, due to the existence of an optimized hardware multiplier, the SKP outspeeds our
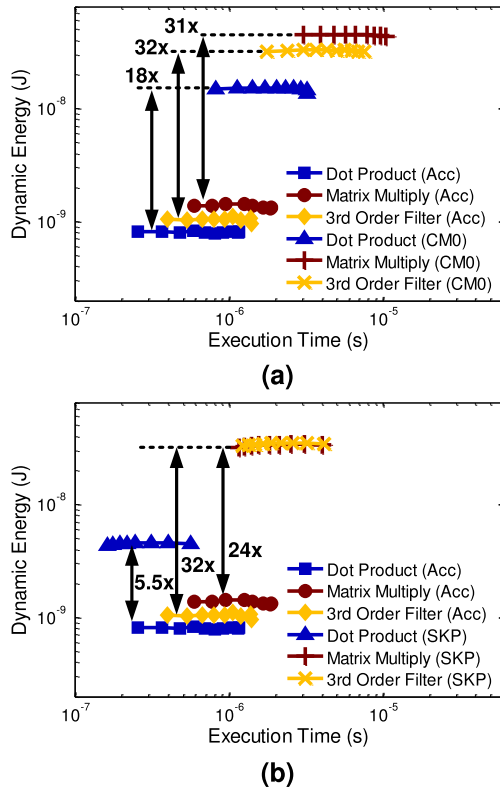
Fig. 15. Dynamic energy of different DSP workloads compared between (a) Cortex M0 (CM0) versus hardware accelerator (Acc) and (b) SKP versus hardware accelerator (Acc).
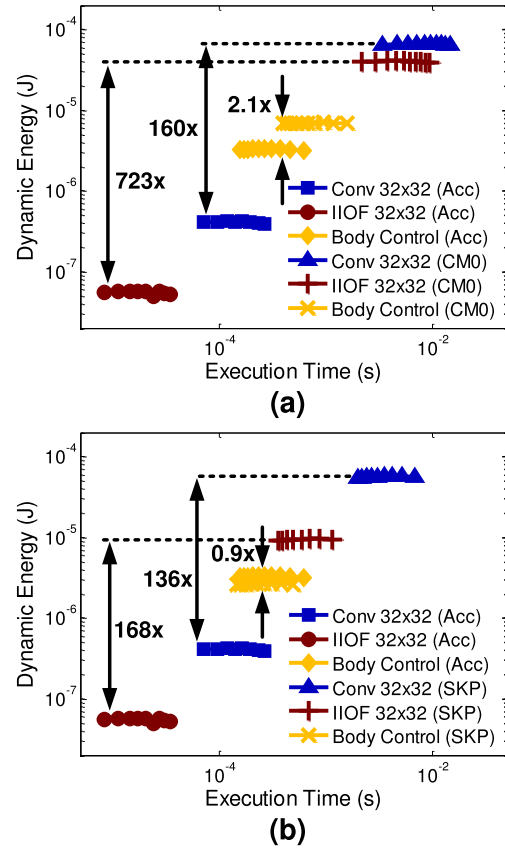

Fig. 16. Dynamic energy of convolution filter (Conv), IIOF, and body control algorithm compared between (a) Cortex M0 (CM0) versus hardware accelerator (Acc) and (b) SKP versus hardware accelerator (Acc).

HLS-generated DSP engine on performing dot-product operations in Fig. 15(b).

Compared with the monolithic accelerators, the improvement from accelerating the body control algorithm is more modest and nuanced. Since it is difficult to implement monolithically due to its ephemeral nature, we chose to accelerate the body control algorithm by composing the atomic functions in the DSP engine. This method, while gaining more than $2\times$ energy improvement over M0, does not compete well against the SKP, as indicated in Fig. 16. It suggests that a composable approach may not fully exploit the energy efficiency gain of specialization when the overhead is considered. Such overhead is caused by the additional data movement operations to offload the computation from M0 to the DSP engine, which can only access its own DSP memory space. Figs. 17 and 18 show how the energy consumption scales with varying pixel resolutions for the two imaging processing algorithms.

In summary, our power and energy characterization of hardware accelerators concludes that the monolithic image processing accelerators in the BrainSoC are necessary to achieve the computational efficiency required by the RobeBee, whereas the composable accelerators may be outperformed by a general-purpose core with a dedicated hardware multiplier, such as the SKP.

D. System-Level Performance and Power

The synthetic autonomous flight workload as shown in Fig. 4 is used to characterize the system performance and power. The typical image resolution is assumed to be 32×32.

All characterization performed in this section assume that a single system clock frequency is shared by all active computational blocks excluding the actuator control accelerator, which uses a derived clock from the 10-MHz relaxation oscillator.

First, we utilize Shmoo chart to illustrate the BrainSoC's maximum operating frequency across different supply voltages and to demonstrate the beneficial effect of adaptive clocking. In the Shmoo test, we only check for functional correctness by running the entire synthetic flight workload sequentially on M0, which does not intend to meet the real-time image processing requirements at the same time. Fig. 19 compares the Shmoo charts of the BrainSoC with and without adaptive clocking. Each Shmoo chart is generated by running the synthetic workload at the specified supply voltage and operating frequency condition repeatedly 20 times and recording the number of successful executions of the entire workload. The status (success or failure) of every execution is determined by observing the existence of the correct external I/O signaling and probing the indicator values saved in the internal memory. In Fig. 19, the color map represents the number of successful executions out of the total 20 runs. It clearly shows that adaptive clocking offers higher operating frequency at any given supply voltage.

Next, executing the synthetic workload with and without accelerators highlights the advantages of hardware acceleration. In these measurements, evocation of all the tasks involved in the synthetic flight workload is achieved by setting

TABLE II
LIST OF ALL CLOCK SOURCES USED ON THE BRAINSOC CHIP

| Name	List of Clock Sources			
	Description			
	On-Chip?	# of Control Bits	Freq. Range (MHz)	Features
Core Clock	Yes	8	25~700	Fanout-of-4 delay tracking synthesizable DCRO
Acc. Clock	Yes	8	25~700	Fanout-of-4 delay tracking synthesizable DCRO
10MHz Clock	Yes	6	8~16.7	Low-power self-calibrated relaxation oscillator with high precision
IVR Clock	Yes	4	90~240	Fully-differential ring oscillator with good power supply rejection ratio
Bypass Core Clock	No	A clock input pin that allows bypassing the internal core clock with external clock		
Bypass Serial Clock	No	A clock input pin that allows calibration signal for the 10MHz clock or bypassing it with external clock		
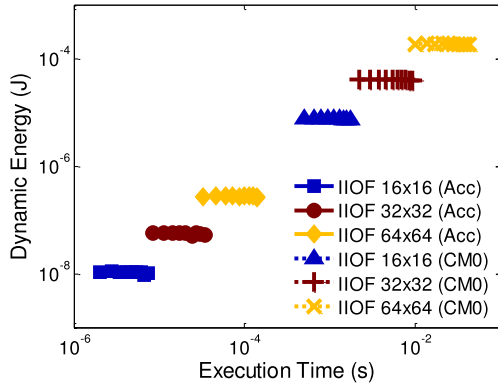
(a)
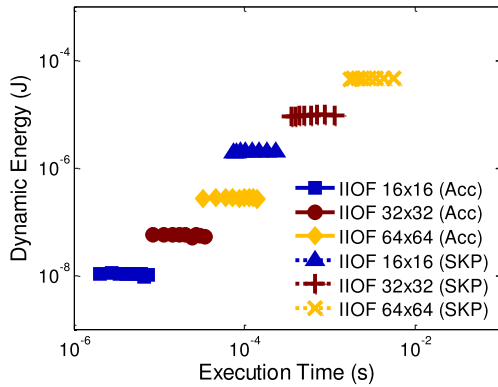
(b)

Fig. 17. Dynamic energy of IIOF at varying pixel resolutions compared between (a) Cortex M0 (CM0) versus hardware accelerator (Acc) and (b) SKP versus hardware accelerator (Acc).
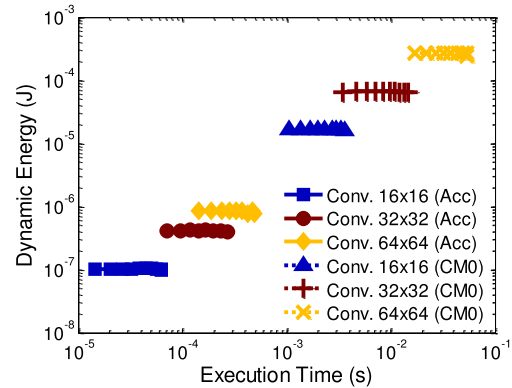
(a)
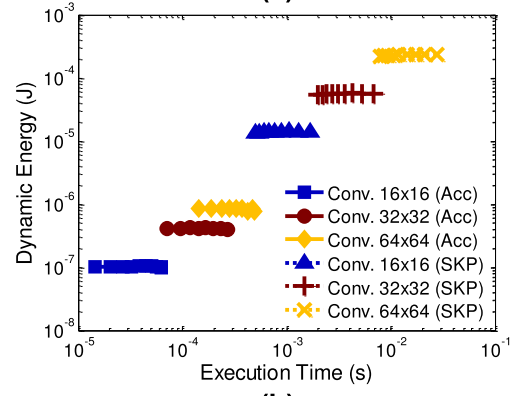
(b)

Fig. 18. Dynamic energy of convolution filter (Conv) at varying pixel resolutions compared between (a) Cortex M0 (CM0) versus hardware accelerator (Acc) and (b) SKP versus hardware accelerator (Acc).

up the appropriate timer-triggered interrupts according to the real-time requirements listed in Fig. 4. We fix the frame rate at 100 frame/s, the image size at 32×32, the IMU access rate at 2000 read/s, and the update rate at the actuator controller at 1000 calculation/s. Then, we sweep the feedback loop frequency of the body control algorithm. In the M0-only case, these tasks have to be time-interleaved, because M0 is the sole central computing resources; while in the accelerator-assisted case, multiple computational blocks are running in parallel. Not surprisingly, higher system clock frequency is needed to sustain faster control loop feedback. Fig. 20 shows the system clock frequency versus control loop frequency. To achieve the 1500-Hz minimum loop frequency limit for RoboBee flight stability, an M0-only system would have to run at 190 MHz, whereas a modest 60-MHz system clock is sufficient for an
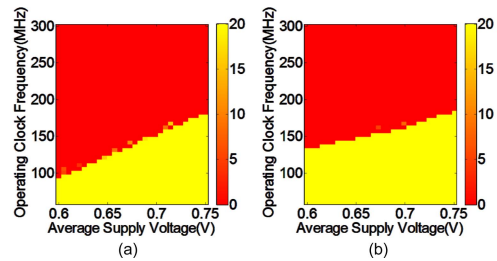

(a)

(b)

Fig. 19. Shmoo charts of BrainSoC under different clocking configurations. (a) IVR closed-loop supply with fixed external clock. (b) IVR closed-loop supply with internal adaptive clock.

accelerator-assisted system, as the most intensive workload can now be offload to the hardware accelerators, relieving the general-purpose core of the heavy duty computations.

TABLE III
LIST OF DIFFERENT CLOCK DOMAINS USED ON THE BRAINSOC CHIP

| List of Clock Domains	
Blocks in the Domain	**Clock Source to Choose From**
Conv. acc., IIOF acc., all 4 image memory banks	Core Clock, Acc. Clock or Bypass Core Clock
DSP engine 1 and DSP memory bank 1	Core Clock, Acc. Clock or Bypass Core Clock
DSP engine 2 and DSP memory bank 2	Core Clock, Acc. Clock or Bypass Core Clock
DSP engine 3 and DSP memory bank 3	Core Clock, Acc. Clock or Bypass Core Clock
Actuator control accelerator and its memory	Derived from 10MHz Clock or Bypass Serial Clock
M0, SKP, system memory, AMBA bus and peripherals	Core Clock or Bypass Core Clock
ADC	10MHz Clock or Bypass Serial Clock
IVR	IVR Clock
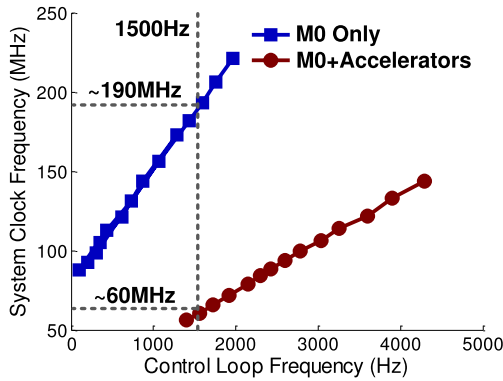
Fig. 20.   Minimum system clock frequency required at different body control loop frequencies.
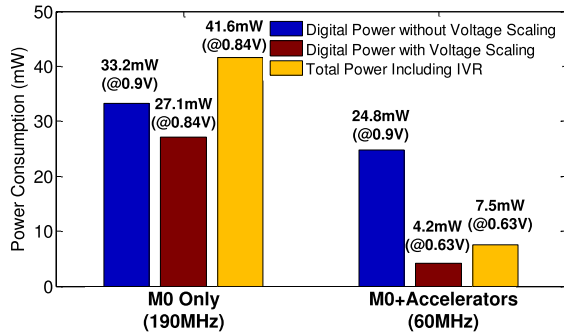

Fig. 21.   BrainSoC power consumption under different configurations.

Finally, Fig. 21 shows the resulting power consumption of the BrainSoC under different voltage scaling scenarios. Reducing clock frequency down to 60 MHz with accelerators allows more aggressive voltage scaling, bringing the BrainSoC's digital power from 24.8 mW at 0.84 V to 4.2 mW at 0.63 V. In addition to the quadratic reduction in dynamic power, leakage current reduces from 17.2 mA at 0.84 V down to 5.2 mA at 0.63 V. Using the IVR to power the SoC increases the overall power due to IVR losses and we also have to account for more degradation in the IVR conversion efficiency when regulating at lower voltage levels. However, this higher power cost is offset by the reduction in weight enabled by the IVR, and we are able to bring down the BrainSoC's total power consumption to a fraction of the 100-mW power budget. The approximately 15-mg weight reduction enabled by the IVR according to Table I can translate to 2250-mJ additional battery capacity, which, in turn, boosts up the power budget

by 41.4 mW for a target flight time of 54.4 s. This more than accounts for the poorer IVR conversion efficiency (60%∼70%) as compared with that of off-the-shelf VRM [17] (90%) when the IVR load is less than 100 mW as set by the target RoboBee power budget.

V. CONCLUSION

A fully integrated BrainSoC that embeds on-chip voltage regulation, clock generation, and analog-to-digital conversion has been designed, fabricated, tested, and evaluated. This BrainSoC is part of a multi-chip system optimized for autonomous insect-scale PAVs. It has been demonstrated to perform open-loop wing flapping control, coordinated with the PEU. A combination of different design techniques has been incorporated in the BrainSoC to meet the stringent weight and power budget and satisfy the real-time demand of autonomous flight experiment, including hardware acceleration for higher performance and better energy efficiency, adaptive clocking for improved supply noise resilience, and voltage frequency scaling and clock gating for lower power consumption. We believe that the SoC approach of building a highly compact and capable pico-robotic computing platform demonstrated in this paper has the potential to be applied to a wide range of diverse weight/size constraint embedded applications and paves the ways toward future transformative advancement in robotic technology.

APPENDIX A
CLOCK DOMAINS IN BRAINSOC

The BrainSoC contains several different integrated clock generators and a number of separate clock domains. To clarify the clock generation and distribution scheme used in the BrainSoC, we summarize the clock sources in Table II and the clock domains in Table III. Each clock domain can select its clock source either statically during initial scan chain configuration or dynamically during run-time code execution. Synchronization when crossing the clock domain boundaries is handled in two ways.

Since the image memory and the DSP memory sit between the core/bus clock domain and the respective accelerator clock domains, we leverage the asynchronous operation capability of these dual-port memory IPs to handle cross-domain data accessing. This takes care of the high-bandwidth data movement between the core and the accelerators, and we carefully

TABLE IV
SUMMARY OF PERFORMANCE COMPARISON BETWEEN BRAINSoC COMPONENTS AND PREVIOUS WORK

Performance Comparison			
Component	Reference/Figure of Merits (FoM)	Baseline for Comparison	BrainSoC
IVR	Reference	[34]	[4]
	Peak Efficiency (%)	74.3	70
	Conversion Ratio	3:1	4:1
	Output Droop (mV)	76	60
10MHz Clock	Reference	[35]	[24]
	Frequency (MHz)	12.8	10
	Power (μW)	98.4	20
	RMS Jitter (ps)	90	8.7
	Variation (%) / VDD (V)	0.14%/(1.1~1.5)	1.1%/(3~4.1)
DCRO	Reference	[32]	This Work
	Frequency Range (MHz)	24.1~321	25~700
	Frequency Levels	16	256
Convolution	Dynamic Power Efficiency (op/s/mW)	9.4G(M0), 5.6G(SKP)	27.8G
	Total Execution Energy (nJ)	660162(M0), 559482(SKP)	410.8
IIOF	Dynamic Power Efficiency (op/s/mW)	9.4G(M0), 5.6G(SKP)	25.6G
	Total Execution Energy (nJ)	403260(M0), 94063(SKP)	55.7
Matrix Multiply	Dynamic Power Efficiency (op/s/mW)	9.6G(M0), 5.7(SKP)	60.4G
	Total Execution Energy (nJ)	44.9(M0), 33.6(SKP)	1.4
ADC	Reference	[36]	[37]
	Resolution	10b	8b
	Area (mm2)	0.0065	0.00986
	DNL (LSB)	0.44	0.48
	INL (LSB)	0.45	0.5
	Sample Rate (Sample/s)	200k	1M
	ENOB (bits)	8.95	7.29

manage in software to avoid accessing the same memory address from both ports of a memory bank at the same time.

The other type of clock domain crossing happens through the interfaces of the memory mapped registers. In this case, we use a synchronizer circuit to resample the signal by its destination clock.

APPENDIX B
BLOCK-LEVEL PERFORMANCE COMPARISON

Due to the diverse blocks integrated on the BrainSoC, a direct side-by-side performance comparison of the entire SoC with previous work delivering the same functionality is challenging. Instead, Table IV summarizes the block-level performance compared with the existing published work [32], [34]–[36]. Since the BrainSoC incorporates multiple custom IP blocks that have been developed in-house, the performance reported here includes results that have been published in our prior work [4], [24], as well future work in preparation for publication [37]. The detailed implementation of some of these sub-blocks is outside of the scope of this paper.

REFERENCES

[1] K. Y. Ma, P. Chirarattananon, S. B. Fuller, and R. J. Wood, "Controlled flight of a biologically inspired, insect-scale robot," *Science*, vol. 340, no. 6132, pp. 603–607, 2013.

[2] R. J. Wood *et al.*, "Progress on 'pico'air vehicles," *Int. J. Robot. Res.*, vol. 31, no. 11, pp. 1292–1302, Sep. 2012.

[3] X. Zhang *et al.*, "A multi-chip system optimized for insect-scale flapping-wing robots," in *Proc. VLSI Circuit Symp.*, 2015, pp. 152–153.

[4] T. Tong, X. Zhang, W. Kim, D. Brooks, and G.-Y. Wei, "A fully integrated battery-connected switched-capacitor 4:1 voltage regulator with 70% peak efficiency using bottom-plate charge recycling," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Sep. 2013, pp. 1–4.

[5] X. Zhang, T. Tong, D. Brooks, and G.-Y. Wei, "Supply-noise resilient adaptive clocking for battery-powered aerial microrobotic system-on-chip in 40 nm CMOS," in *Proc. Custom Integr. Circuits Conf. (CICC)*, Sep. 2013, pp. 1–4.

[6] Y.-T. Liao, H. Yao, A. Lingley, B. Parviz, and B. P. Otis, "A 3-μW CMOS glucose sensor for wireless contact-lens tear glucose monitoring," *IEEE J. Solid-State Circuits*, vol. 47, no. 1, pp. 335–344, Jan. 2012.

[7] M. Monge *et al.*, "A fully intraocular 0.0169 mm²/pixel 512-channel self-calibrating epiretinal prosthesis in 65 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Sep. 2013, pp. 296–297.

[8] M. Fojtik *et al.*, "A millimeter-scale energy-autonomous sensor system with stacked battery and solar cells," *IEEE J. Solid-State Circuits*, vol. 48, no. 3, pp. 801–813, Mar. 2013.

[9] Y.-P. Chen *et al.*, "An injectable 64 nW ECG mixed-signal SoC in 65 nm for arrhythmia monitoring," *IEEE J. Solid-State Circuits*, vol. 50, no. 1, pp. 375–390, Jan. 2015.

[10] M. Shoaran *et al.*, "A 16-channel 1.1 mm² implantable seizure control SOC with sub-μw/channel consumption and closed-loop stimulation in 0.18 μm CMOS," in *Proc. IEEE Symp. VLSI Circuits (VLSI-Circuits)*, Jun. 2016, pp. 1–2, doi: 10.1109/VLSIC.2016.7573557.

[11] *Nano Hummingbird*, accessed on May 25, 2017. [Online]. Available: http://www.avinc.com/nano/

[12] M. Karpelson, G.-Y. Wei, and R. J. Wood, "A review of actuation and power electronics options for flapping-wing robotic insects," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Oct. 2008, pp. 779–786.

[13] M. Karpelson, G.-Y. Wei, and R. J. Wood, "Milligram-scale high-voltage power electronics for piezoelectric microrobots," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Sep. 2009, pp. 2217–2224.

[14] *Aanalog Devices, AD7450 Data Sheet*, accessed on May 25, 2017. [Online]. Available: http://www.analog.com/en/products/analog-to-digital-converters/ad-converters/ad7450.html#product-overview

[15] *NXP, Kinetis Km34 Sub-Family Data Sheet*, accessed on May 25, 2017. [Online]. Available: http://www.nxp.com/assets/documents/data/en/data-sheets/KM34P144M75SF0.pdf

[16] *Linear Technology, LTC6900 Data Sheet*, accessed on May 25, 2017. [Online]. Available: http://www.linear.com/product/LTC6900

[17] *Linear Technology, LTC3620 Data Sheet*, accessed on May 25, 2017. [Online]. Available: http://www.linear.com/product/LTC3620

[18] *Invensense, MPU-6500 Data Sheet*, accessed on May 25, 2017. [Online]. Available: https://www.invensense.com/products/motion-tracking/6-axis/mpu-6500/

[19] *Centeye Vision Chips*, accessed on May 25, 2017. [Online]. Available: http://www.centeye.com/technology/vision-chips/

[20] *Wikipedia, Energy Density*, accessed on May 25, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Energy_density

[21] M. Lok, X. Zhang, E. F. Helbling, R. Wood, D. Brooks, and G.-Y. Wei, "A power electronics unit to drive piezoelectric actuators for flying microrobots," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Jun. 2015, pp. 1–4.

[22] M. Lok, D. Brooks, R. Wood, and G.-Y. Wei, "Design and analysis of an integrated driver for piezoelectric actuators," in *Proc. IEEE Energy Convers. Congr. Expo. (ECCE)*, Apr. 2013, pp. 2684–2691.

[23] P.-E. J. Duhamel, N. O. Pérez-Arancibia, G. L. Barrows, and R. J. Wood, "Biologically inspired optical-flow sensing for altitude control of flapping-wing microrobots," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 2, pp. 556–568, Feb. 2013.

[24] X. Zhang, D. Brooks, and G.-Y. Wei, "A $20\,\mu$W 10 MHz relaxation oscillator with adaptive bias and fast self-calibration in 40 nm CMOS for micro-aerial robotics application," in *Proc. IEEE Asian Solid-State Circuits Conf. (ASSCC)*, Oct. 2013, pp. 1–4.

[25] H.-P. Le, S. R. Sanders, and E. Alon, "Design techniques for fully integrated switched-capacitor DC-DC converters," *IEEE J. Solid-State Circuits*, vol. 46, no. 9, pp. 2120–2131, Sep. 2011.

[26] B. Razavi, "Design of analog CMOS integrated circuits," in *Bandgap References*. New York, NY, USA: McGraw-Hill, 2001, ch. 11.

[27] *Opencv Optical Flow Tutorial*, accessed on May 25, 2017. [Online]. Available: http://docs.opencv.org/3.2.0/d7/d8b/tutorial_py_lucas_kanade.html

[28] S. B. Fuller, E. F. Helbling, P. Chirarattananon, and R. J. Wood, "Using a MEMS gyroscope to stabilize the attitude of a fly-sized hovering robot," in *Proc. Int. Micro Air Veh. Conf. Competition (IMAV)*, Delft, The Netherlands, Aug. 2014.

[29] *Vivado High-Level Synthesis*, accessed on May 25, 2017. [Online]. Available: http://www.xilinx.com/products/design-tools/vivado/integration/esl-design.html

[30] X. Zhang, T. Tong, D. Brooks, and G.-Y. Wei, "Evaluating adaptive clocking for supply-noice resilience in battery-powered aerial microrobotic system-on-chip," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 8, pp. 2309–2317, Aug. 2014.

[31] B. Zimmer *et al.*, "A RISC-V vector processor with tightly-integrated switched-capacitor DC-DC converters in 28 nm FDSOI," in *Proc. Symp. VLSI Circuits (VLSI Circuits)*, 2015, pp. C316–C317.

[32] J. Perez-Puigdemont, F. Moll, and A. Calomarde, "All-digital simple clock synthesis through a glitch-free variable-length ring oscillator," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 61, no. 2, pp. 90–94, Feb. 2014.

[33] T. M. Van Breussegem and M. S. Steyaert, "Monolithic capacitive DC-DC converter with single boundary–multiphase control and voltage domain stacking in 90 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 46, no. 7, pp. 1715–1727, Jul. 2011.

[34] H.-P. Le, J. Crossley, S. R. Sanders, and E. Alon, "A sub-ns response fully integrated battery-connected switched-capacitor voltage regulator delivering 0.19 W/mm^2 at 73% efficiency," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, Sep. 2013, pp. 372–373.

[35] Y. Cao, P. Leroux, W. De Cock, and M. Steyaert, "A 63,000 Q-factor relaxation oscillator with switched-capacitor integrated error feedback," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, Feb. 2013, pp. 186–187.

[36] H. Y. Tai, Y. S. Hu, H. W. Chen, and H. S. Chen, "11.2 A 0.85 fJ/conversion-step 10 b 200 kS/s subranging SAR ADC in 40 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, Feb. 2014, pp. 196–197.

[37] S. Chaput *et al.*, "An area efficient 8b single-ended ADC with extended input voltage range," in preparation.

Xuan Zhang (S'08–M'15) received the B.Eng. degree in electrical engineering from Tsinghua University, Beijing, China, in 2006, and the M.S. and Ph.D. degrees in electrical and computer engineering from Cornell University, Ithaca, NY, USA, in 2009 and 2012, respectively.

She was a Post-Doctoral Fellow in computer science with the John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA. She is currently an Assistant Professor with the Preston M. Green Department of Electrical and Systems Engineering, Washington University in St. Louis, St. Louis, MO, USA. She works across the fields of VLSI, computer architecture, and robotics. Her current research interests include adaptive learning hardware for autonomous systems, hardware/software co-design of efficient power delivery and distribution, and ubiquitous self-powered Internet-of-Things devices.

Mario Lok (S'08) received the B.Sc. degree in engineering physics from The University of British Columbia, Vancouver, BC, Canada, in 2008 and the M.S. degree in electrical engineering from The University of Texas at Austin, Austin, TX, USA, in 2010. He is currently pursuing the Ph.D. degree with Harvard University, Cambridge, MA, USA, with a focus on miniature power electronics design on micro-air vehicle.

His current research interests include power integrated circuits, switch-mode power electronics, robotics, and highly integrated electrical mechanical system.
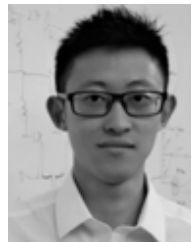
Tao Tong (S'10) received the B.E. degree from Tsinghua University, Beijing, China, the M.S. degree from Oregon State University, Corvallis, OR, USA, and the Ph.D. degree from Harvard University, Cambridge, MA, USA.

He was with MediaTek Wireless Inc., Woburn, MA, USA, and Lion Semiconductor Inc., San Francisco, CA, USA, where he was involved in designing analog-to-digital converters and fully integrated dc–dc converters for mobile applications. His current research interests include integrated voltage regulators and their applications in energy efficient computing systems.
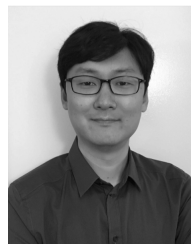
Sae Kyu Lee (S'10) received the B.S. degree in electrical engineering from Seoul National University, Seoul, South Korea, in 2006, the M.S. degree in electrical and computer engineering from The University of Texas at Austin, Austin, TX, USA, in 2008, and the Ph.D. degree from Harvard University, Cambridge, MA, USA, in 2016.

He was with Intel Corporation, Austin, TX, USA, and Advanced Micro Devices, Inc., where he was involved in mobile microprocessor designs. He is currently a Post-Doctoral Fellow with Harvard University. His current research interests include variety of topics from VLSI design for efficient on-chip power delivery solutions to building energy-efficient hardware accelerators for machine learning applications.

Brandon Reagen (S'14) received the bachelor's degree in computer systems engineering and applied mathematics from the University of Massachusetts Amherst, Amherst, MA, USA, in 2012 and the M.S. degree in computer science from Harvard University, Cambridge, MA, USA, in 2014, where he is currently pursuing the Ph.D. degree.

His current research interests include the fields of computer architecture, VLSI, and machine learning with specific interest in designing extremely efficient hardware to enable ubiquitous deployment of machine learning models across all compute platforms.

Simon Chaput (S'14) received the B.Eng. and M.A.Sc. degrees in electrical engineering from the Université de Sherbrooke, Sherbrooke, QC, Canada, in 2011 and 2013, respectively. He is currently pursuing the Ph.D. degree with Harvard University, Cambridge, MA, USA.

From 2011 to 2013, he was with Teledyne DALSA, Waterloo, ON, Canada, where he was involved in the design of high-voltage power management integrated circuits (PMICs). His current research interests include the design of high-voltage driver ICs for novel application and the design of PMICs.

Pierre-Emile J. Duhamel (S'11) received the A.B. degree in computer science, and chemistry and physics from the Harvard College, Cambridge, MA, USA, in 2009, and the M.S. degree in computer science from the John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA, in 2014, where he is currently pursuing the Ph.D. degree in computer science.

He was with the Harvard Microrobotics Laboratory, John A. Paulson School of Engineering and Applied Sciences, Harvard University. His current research interests include computer vision, computer hardware, and robotics.

Mr. Duhamel received the National Science Foundation Graduate Research Fellowship in 2010.

Robert J. Wood (M'01–SM'16) received the M.S. and Ph.D. degrees from the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA, USA, in 2001 and 2004, respectively.

He is currently the Charles River Professor of Engineering and Applied Sciences with the John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA, a Founding Core Faculty Member with the Wyss Institute for Biologically Inspired Engineering, Cambridge, MA, USA, Harvard University, and a National Geographic Explorer. His current research interests include microrobotics, soft and wearable robots, and bioinspired robotics.

David Brooks (F'16) received the B.S. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, and the M.A. and Ph.D. degrees in electrical engineering from Princeton University, Princeton, NJ, USA.

He was a Research Staff Member with the IBM T. J. Watson Research Center, Yorktown Heights, NY, USA. He is currently the Haley Family Professor of Computer Science with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA. His current research interests include resilient and power-efficient computer hardware and software design for high-performance and embedded systems.

Dr. Brooks has received several honors and awards, including the ACM Maurice Wilkes Award, the ISCA Influential Paper Award, the NSF CAREER award, the IBM Faculty Partnership Award, and the DARPA Young Faculty Award.

Gu-Yeon Wei (M'00) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA, in 1994, 1997, and 2001, respectively.

He is currently a Gordon McKay Professor of Electrical Engineering and Computer Science with the John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA. His current research interests include multiple layers of a computing system: digital and mixed-signal integrated circuits, computer architecture, and design tools for efficient hardware, and identifying synergistic opportunities across these layers to develop energy-efficient solutions for a broad range of applications from flapping-wing microrobots to warehouse scale computing to deep learning on edge devices.