Time Keeping Ability of Crystal-Free Radios

Osama Khan, David C. Burnett, Filip Maksimovic, Brad Wheeler, Sahar Mesri, Arvind Sundararajan, Bob L. Zhou, Ali M. Niknejad, *Fellow, IEEE*, and Kristofer S. J. Pister

***Abstract*—The Internet-of-Things (IoT) promises one trillion wireless sensors in the next 10 to 15 years. To enable this scale, we present the experimental results on the time-keeping ability of crystal-free radios to lower the cost of the wireless sensors. We propose a network referenced frequency lock loop and demonstrate a one sigma mean frequency accuracy of 47 p/min, post calibration. We also extend this concept to calibrate the on-chip RF local oscillator by exploiting the network time synchronization for frequency calibration. This timing accuracy is sufficient to create a scalable wireless mesh network up to 10 hops deep with a 1 ms guard time interval. The impact of time synchronization on the average power consumption of the wireless sensor node is negligible and especially true for environments with mobility and high data traffic. We envision these emerging microsystems to be embedded into everyday objects and discuss the tradeoff of mobility with the average power consumption.**

***Index Terms*—Crystal-free radio, dynamic networks, Internet of Things (IoT), time synchronization, wireless sensor network.**

I. Introduction

THE semiconductor technology is undoubtedly one of the greatest invention of the 20th Century if not the greatest; when humans learned how to convert sand into computing tools. The 70 years of transistor evolution has fueled the semiconductor industry and resulted in an exponential growth. This exponential growth can be seen from the evolution of computing as shown in Fig. 1, also known as Bell's law [1]. With every new computing class, we reduce the price by a factor of 10, lower the active power consumption by a factor of 100, and that increases the scale by a factor of 10. If we follow this trend, it is clear that we are entering into an era where computing, sensing, and communication is essentially becoming disposable. This will enable the next level of scale, which is billions or trillions of these microsystems in the next decade or so [2], [3].

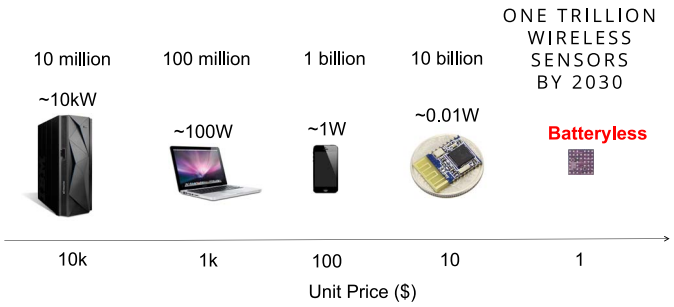


Fig. 1. Evolution of computing.

This trend in computing opens up many new opportunities as well as system design challenges. Limited battery life and cost are the major bottlenecks in the proliferation of these devices [4], [5]. Therefore, energy harvesting from ambient environment is an attractive alternative as a power source, but it has to be economically viable to be commercially successful at scale. The economy of scale also requires that these emerging microsystems need to be low-cost (<$1). Therefore, full system-integration on a single piece of silicon is needed to further reduce the cost.

Monolithic integration also reduces the profile of these microsystems, a much desirable property. We envision achieving these goals will enable the vision of dynamic networks. We define dynamic networks that are self-aware of the network objectives and self-maintaining. We expect these broad terms to correspond to different levels of capability as the underlying fundamental technologies advances. For example, in a wireless sensor network self-maintaining could be as simple as, if a wireless sensor runs low on battery, it places an online order of fresh batteries that are delivered to the facility without any human intervention. Alternatively it could be as sophisticated as a drone or a robot working symbiotically with the network, replaces the battery or perhaps deploy a new wireless sensor node without any human intervention.

In this paper, in Section II, we present how this paper is related to the previously published literature. In Section III, we present a system solution for a crystal-free operation toward reducing the cost of the wireless sensors and define the scope of this paper. In Section IV, we present implementation details and the measured silicon results of the time-keeping ability of crystal free radio. In Section V, we envision how these emerging wireless sensors can form a dynamic network and

Manuscript received August 3, 2018; accepted August 26, 2018. Date of publication September 10, 2018; date of current version May 8, 2019. This work was supported in part by the DARPA Vanishing Programmable Resources Program, in part by the National Science Foundation Graduate Research Fellowship under Grant 1106400, and in part by the National Science Foundation SBIR Phase I Award No. 1646925. *(Corresponding author: Osama Khan.)*

O. Khan is with the Berkeley Sensor and Actuator Center, University of California at Berkeley, Berkeley, CA 94720 USA, and also with MonoLets, Inc., Berkeley, CA 94704 USA (e-mail: oukhan@berkeley.edu).

D. C. Burnett, F. Maksimovic, B. Wheeler, S. Mesri, A. Sundararajan, B. L. Zhou, and K. S. J. Pister are with the Berkeley Sensor and Actuator Center, University of California at Berkeley, Berkeley, CA 94720 USA (e-mail: db@berkeley.edu; fil@berkeley.edu; brad.wheeler; smesri@berkeley.edu; arvinds@berkeley.edu; bob.linchuan@berkeley.edu; ksjp@berkeley.edu).

A. M. Niknejad is with the Berkeley Wireless Research Center, University of California at Berkeley, Berkeley, CA 94720 USA (e-mail: niknejad@berkeley.edu).

Digital Object Identifier 10.1109/JIOT.2018.2869143

discuss their associated design challenges. Finally, we present the conclusion and future work in Section VI.

II. Related Work

Time synchronization for the wireless sensor network has been studied extensively [6]–[8]. Most of the previous work has focused on achieving time synchronization among a group of wireless sensor nodes using a crystal (XTAL)-based clock source [9]. An off-the-shelf commercial product LTC5800 exploiting crystals in a wireless sensor network, reports single digit microseconds of synchronization error for a node 5 hops deep in an 8 °C/min temperature ramp environment [10].

In this paper, we extend these concepts to a crystal-free wireless sensor node to achieve time synchronization and exploit the network wide time synchronization to also achieve frequency calibration of all on-chip free running oscillators including the RF local oscillator (LO). To the best of our knowledge this has not been demonstrated before.

In the reference broadcast synchronization (RBS) a reference message is broadcasted [6]. The receivers record their local time when receiving the reference broadcast and exchange the recorded time with each other [8]. The main advantage of RBS is that it eliminates transmitter-side nondeterminism. The disadvantage of this approach is that additional message exchange is necessary to communicate the local time-stamps between the nodes [8].

The Timing-sync Protocol for Sensor Networks (TPSNs) algorithm first creates a spanning tree of the network and then performs pairwise synchronization along the edges [7]. Each node gets synchronized by exchanging two synchronization messages with its reference node one level higher in the hierarchy [8]. The shortcoming of TPSN is that it does not estimate the clock drift of nodes, which limits its accuracy, and does not handle dynamic topology changes [8].

We find this paper to be similar to the Flooding Time Synchronization Protocol (FTSP) and Time Synchronized Mesh Protocol (TSMP) [8], [11]. The FTSP synchronizes the time of a sender to possibly multiple receivers utilizing a single radio message time-stamped at both the sender and the receiver sides [8]. The FTSP has been shown to perform better than both RBS and TPSN [8]. The TSMP uses regular updates to maintain a shared sense of time in multihop WSNs. TSMP propagates time from a single time master, the access point (AP), to all other devices. Devices with direct AP links use each transaction to synch their clocks to the AP. Similarly, when these "first hop" devices talk to their children, the child adopts the parent's clock [11]. Time synchronized channel hopping (TSCH) is adopted from the TSMP and is discussed further in Section V.

Since our focus is on the crystal-free wireless sensor nodes, we expect the most dominant source of timing error will be from the on-chip free running oscillator itself. Therefore, we can ignore some of the timing error sources as highlighted in [8] as their relative impact is negligible or the timing error presented in this paper can be improved by considering all the uncertainties in a radio message delivery as presented in [8], but it is not the focus of this paper.

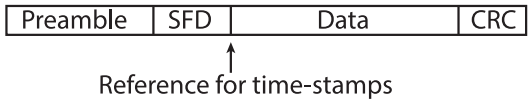


Fig. 2. IEEE 802.15.4 RF packet format.

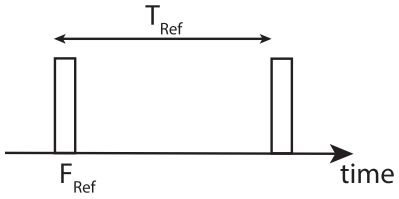


Fig. 3. Periodic network beacons used as a timing reference.

III. Time-Frequency Synchronization

We presented a system solution using off-the-shelf components as a proof of concept to provide a precise frequency reference for a crystal-free radio operation [12]. In our previous work, we demonstrated post-calibration one sigma frequency accuracy of 70 p/min. This approach relies on receiving periodic RF packets that serves as a timing reference for the crystal-free wireless sensor node. We employ MAC layer time-stamping of the RF packets to provide a precise time-stamp, removing timing uncertainties caused by the higher layer protocol stack.

The IEEE 802.15.4 standards' compliant RF packets are used in our experimental setup and the reference for time-stamping is arbitrarily chosen to be the end of the start-of-frame delimiter in an RF packet as shown in Fig. 2.

Fig. 3 shows the periodic network beacons being transmitted at a predetermined fixed RF frequency/frequencies (F_{Ref}) and a timing interval (T_{Ref}). Both of these parameters are parametrizable and can also be dynamically updated to cater different environments and use-case requirements.

In order to calibrate all the on-chip free running oscillators on a crystal-free wireless sensor node, the wireless node needs to be able to first receive the network beacons. On cold start, this can be addressed by executing a blind network beacons search algorithm on the wireless sensor node by sweeping the RF LO across all the possible RF channels. Once a network beacon is found the wireless sensor node can then execute the time-frequency synchronization algorithm.

Note, the timing recovery module on the receiver should be designed to recover the clock from the received RF packet for the clock timing recovery. The crystal free radio does not have any impact on the frame synchronization and therefore it is handled as in a traditional crystal-based radio.

In case of loss of RF packets, the wireless sensor node loses the timing reference. The wireless sensor node can then rely on the calibrated free-running frequency references to track time and known good frequencies, until the time it starts receiving the network beacons again. In case the network beacons are lost for a sufficiently long time the wireless sensor node reinitiate the cold-start process of finding the network beacons.

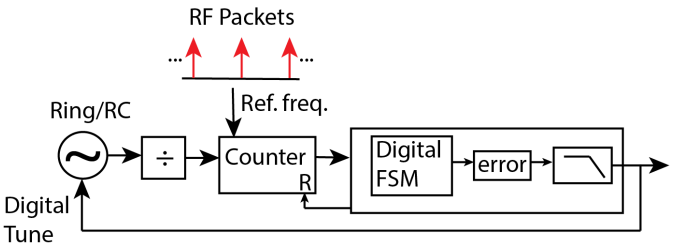


Fig. 4. Network referenced FLL.

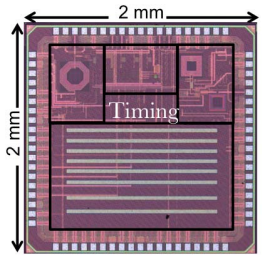


Fig. 5. Die photograph of the test chip in 65-nm CMOS.

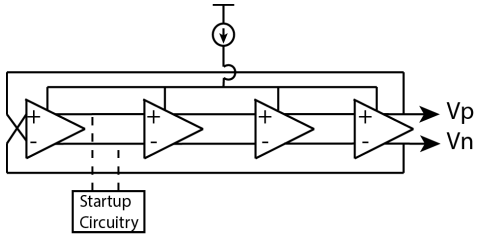


Fig. 6. Differential 4-stage 25-MHz ring oscillator.

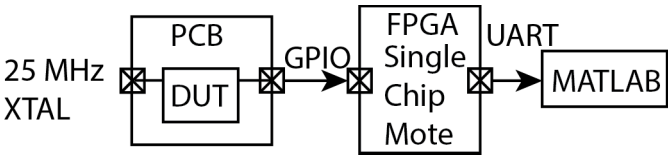


Fig. 7. Measurement setup for measuring timing accuracy.

In this paper, we will assume that the network beacons are already found by the wireless sensor node and the timing recovery has already been performed. We intend to address this issue in detail in our future work.

IV. Crystal-Free Timing Accuracy

The notion of time plays a critical role in any dynamic system. The dynamic network, depending on the level of autonomy, can not only sense the environment but can also actuate and control the physical space that is being monitored. For any control algorithm, having an accurate sense of time is crucial for the edge devices. In a typical microsystem, time is measured by counting a precise frequency reference generated using a very high-quality off-chip resonator, typically a crystal (XTAL) or a micro electro mechanical system (MEMS) device that adds to the overcall cost of these devices at scale. The cost of an XTAL is almost equal to the cost of the 2×2 mm^2 silicon and at high volumes ($>$100 M units) represents a significant fraction of the overall system cost.

We intend to design the edge devices without any external resonator (XTAL/MEMS). The proposed idea is conceptually illustrated in Fig. 4. The network referenced frequency locked loop (FLL) comprises a relaxation oscillator as a digitally controlled oscillator (DCO) the output of which is divided down to a lower frequency to be used as a clock source for an on-chip counter. One node in the network serves as a timeserver with accurate clock source as compared to the crystal-free devices and periodically sends RF packets at regular time intervals. The periodic RF packets or network pings/beacons serve as a timing reference for the FLL. When an RF packet is received the digital state machine stores the counter value and resets the counter. Since the RF packets arrives at a predetermined timing interval, by receiving two consecutive packets the crystal-free devices can compute the error on their local clock reference. The computed error can then be averaged over several consecutive measurements and can be used to digitally calibrate the on-chip clock reference as shown in Fig. 4.

We taped-out a test chip in 65-nm CMOS, see Fig. 5, to further validate and characterize the proposed idea in silicon. The section labeled timing in Fig. 5 generates all the on-chip clocks for the system-on-chip (SoC) from the calibrated ring oscillator reference. The timing section implements the ring oscillator, divider, and the counter on chip. The digital finite state machine (FSM), error computation and filtering are implemented in the software. The ring oscillator is implemented as a four-stage differential structure in 65-nm CMOS and designed to operate at 25-MHz reference frequency as shown in Fig. 6. It occupies 0.025 mm^2 silicon area with a startup circuitry that was included to guarantee oscillation from cold-start.

The designed SoC has a 32-bit ARM M0 microprocessor, 128 kB of SRAM, and an IEEE 802.15.4 transceiver. The performance of which is not discussed in this paper.

In our implementation of Fig. 4, the 25-MHz ring oscillator consumes 7 μW of power and the divider is configured to divide by 50 to generate a 500 kHz clock source for the on-chip counter. No attempt is made to design the absolute best on-chip clock reference with superior supply and temperature sensitivity. In fact, the proposed calibration scheme can also calibrate out the frequency changes due to on-chip supply and temperature variations. The 25-MHz frequency reference was chosen to satisfy different clocking requirements on the SoC. Any other lower clock frequency reference can be used for time-keeping depending on the application. In the published literature, less than 1%/V supply voltage sensitivity and less than 1% temperature sensitivity has been demonstrated over a wide temperature range for on-chip relaxation/ring-based oscillators [13]–[17].

A. Measured Results

In order to measure the timing accuracy of the free-running on-chip 25-MHz reference we setup an experiment as shown in Fig. 7. A *C*-code was written and executed on the device under test (DUT) to toggle a general purpose input output (GPIO) pin

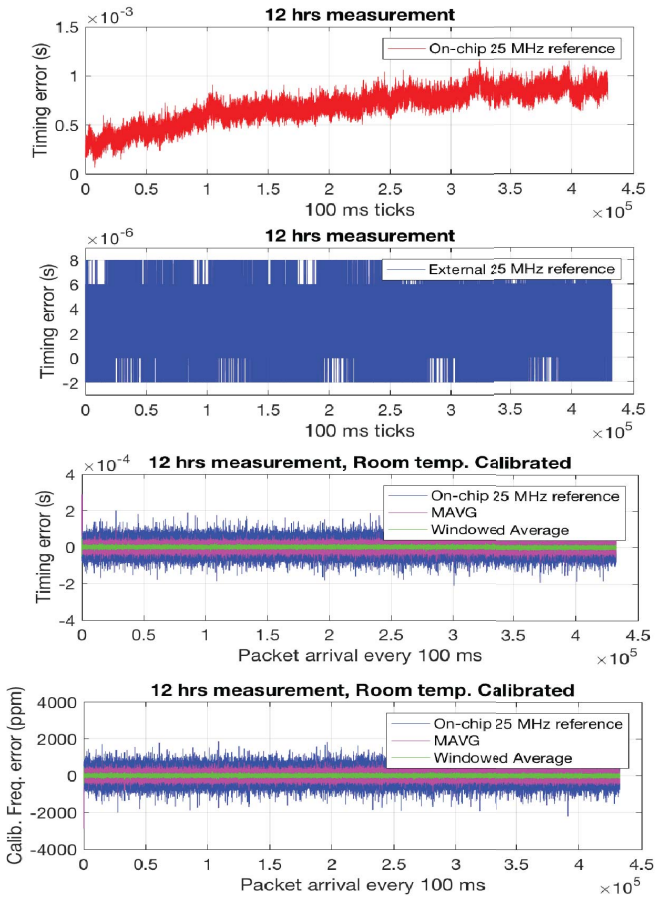


Fig. 8. Timing accuracy for the 100 ms time interval. The top plot shows the timing error due to the free-running 25-MHz reference. The second plot from the top shows the timing error due to an external 25-MHz crystal frequency reference. The third and fourth plot from the top shows the measured calibrated timing and frequency error, respectively.

every 100 ms using the on-chip 25-MHz reference as a clock source. For comparison, the DUT can also be configured to use an external off chip 25-MHz XTAL clock. The GPIO pin was connected to a field programmable gate array (FPGA) that was used to measure the timing interval and the measurement data was transmitted over the universal asynchronous receiver transmitter (UART) to be processed in MATLAB. The FPGA implements the digital system of the DUT named as single chip mote to facilitate testing [18].

The top plot in Fig. 8 shows the measured timing accuracy over 12 h at room temperature using the 25-MHz on-chip reference and the second plot from the top shows the same data using an external 25-MHz XTAL derived clock. The x-axis is the number of 100 ms ticks and the y-axis is the measured timing error. The timing accuracy is 3 orders of magnitude better with the XTAL clock as expected. The observed timing error using the on-chip 25-MHz reference is due to the random noise and deterministic environmental effects caused by the overnight temperature variation.

In order to verify the feasibility of the network referenced FLL, a second experiment was setup as shown in Fig. 9. The FPGA is configured to send periodic IEEE 802.15.4 standard compliant RF packets using clock and data wires directly

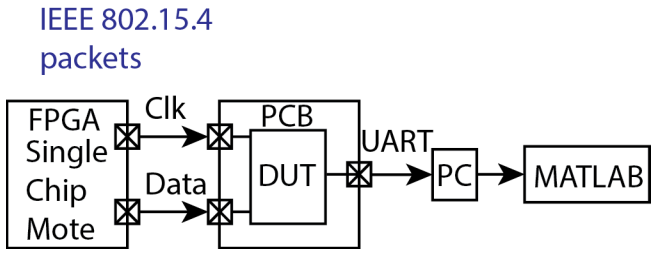


Fig. 9. Measurement setup for network referenced FLL.

connected to the test printed circuit board hosting the DUT. This emulates the timeserver sending periodic RF packets. A *C*-code was written and executed on the DUT, that puts the device into continuous receive mode. When an RF packet is received the FSM captures the timestamp of the event. The timestamp data is then transmitted over the UART to a personal computer. A Python script is used to log the data and stores it into a file to be processed in MATLAB. Although the clock and data in Fig. 9 are delivered to the DUT using wires, this does not impact the conclusion derived from the measured results. The RF channel delay spread can be on the order of 10–50 ns, whereas the achieved timing accuracy is in μs. Therefore, the achieved calibrated frequency accuracy is not limited by the RF channel delay spread.

By receiving two consecutive RF packets the local clock error can be estimated and is shown in the third plot from the top (blue plot) in Fig. 8. The calibrated reference has a measured standard deviation of 30.3 μs, which is mostly dominated by the white noise process. This curve corresponds to the output of the error module in the conceptual illustration of Fig. 4. Since the noise character is mostly white, averaging can further improve the mean frequency estimate of the on-chip frequency reference or reduce the variance of the timing error. Two low-pass FIR filters have been explored. First, the cumulative moving average (MAVG) computes the mean of all the past samples up to the current sample. Second, the windowed average computes the mean of the past M samples. For the windowed average we arbitrarily choose $M = 10$ (with 100 ms packet arrival this corresponds to computing the average of the past 1 s time interval). The output of the two low-pass FIR filters is also plotted in Fig. 8. The standard deviation reduced to 14.2 μs for the MAVG filter and 4.7 μs for the windowed average filter.

The reason that windowed average performs better than the cumulative MAVG is over long time intervals nonstationary noise sources (nonwhite character) can become dominant and cannot be reduced by averaging. The optimum filter duration for the windowed average can be found by measuring the Allan deviation (ADEV) of the on-chip relaxation oscillator and corresponds to the averaging time interval over which the ADEV is minimum.

The measured frequency of the on-chip relaxation oscillator is shown in p/min units in bottom plot of Fig. 8. The measured p/min error after on-chip calibration is 303 p/min, 142 p/min for the MAVG and 47 p/min for the windowed average. Note that this achieved frequency accuracy corresponds

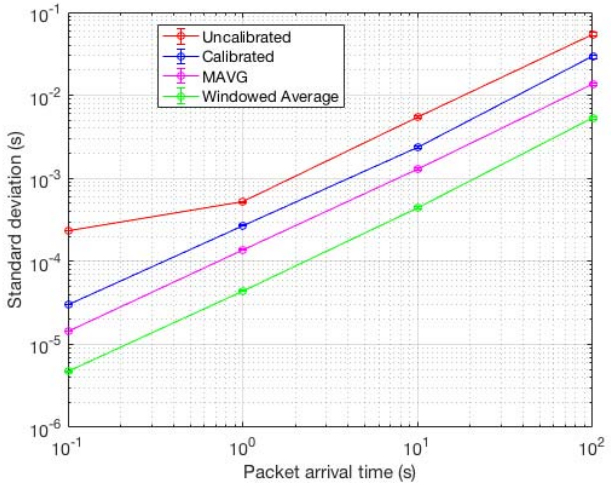


Fig. 10. Measured timing error as a function of packet arrival time.

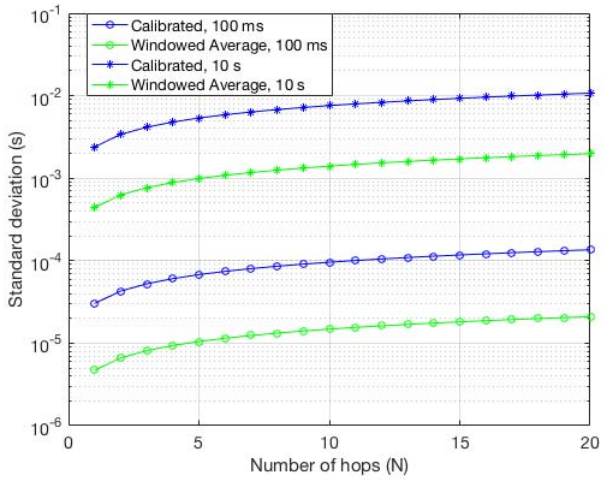


Fig. 11. Computed timing error for a multihop network.

to the mean frequency estimate of the on-chip frequency reference and does not correspond to its timing jitter performance which can be an order of magnitude higher and trades directly with power. This must be carefully evaluated for satisfying system jitter specifications of the on-chip clocks.

The computed error estimate can be used to tune the on-chip DCO using a feedback loop as shown conceptually in Fig. 4 or timing calibration can be applied digitally using a feedforward lookup table approach.

B. Impact of Synchronization Interval and Hop Count

Since the on-chip frequency reference is free running during the network pings or the periodic RF packets, it is expected that the timing error will increase as the RF packet arrival time increases. Fig. 10 shows the measured timing error as a function of the RF packet arrival time. This graph shows the clock's free-running time keeping ability as a function of the synchronization interval. The clock time-keeping ability over a time-interval τ is given by the following equation [19]:

$$x_p(\tau) = k\tau\sigma_y(\tau) \tag{1}$$

where k depends on the noise character dominant over the measurement time interval τ and $\sigma_y(\tau)$ is the ADEV of the reference clock. Due to the nonstationary nature of the noise-sources, it becomes very difficult to come up with an accurate analytical expression for the time-keeping ability of a given clock source. Therefore, we have resorted to the empirical results in Fig. 10.

At 1 s time interval, the standard deviation for the windowed average is roughly about 45 μs. This means if the edge devices sleep for 1 s then the six sigma timing uncertainty is about 270 μs, therefore a 1 ms guard time will be more than sufficient to maintain synchronization.

A 1 ms guard time and assuming 1 ms of radio packet duration, would correspond to 0.2% radio duty cycle for every 1 s synchronization interval. If the active radio power is 1 mW then this corresponds to an average power consumption of just 2 μW which is sufficient to operate from harvested energy without any battery even from indoor lighting conditions at 300 lux [20].

It is interesting to note that a linear relationship is observed between the guard time (proportional to the standard deviation) and the synchronization interval (packet arrival time) in the log–log plot of Fig. 10 over the measured timescales dominated by the white noise. The duty cycle is limited by the fraction of the time the radio has to stay on for a given sleep interval to maintain synchronization. Increasing the synchronization interval, say by a factor of 10, also increases the guard time by the same factor and therefore does not help to lower the radio duty cycle and to reduce the average power. A sublinear relationship would have helped to reduce the average radio power by further increasing the synchronization interval.

An important advantage of using a low-Q on-chip relaxation oscillator as a timing and frequency reference as compared to an off-chip high Q resonator (XTAL/MEMS) is that the system can be duty cycled very quickly, a much desirable property. The startup time of a crystal oscillator can easily take upto 1 ms or longer [14]. This directly saves energy and reduces the average power of the crystal-free devices.

Since the timing error is dominated by a white noise process over the measured synchronization intervals, therefore if the crystal-free devices without any external resonator (XTAL/MEMS) themselves acts as timeservers then the timing error will grow as $\sqrt{N}$, where N is the hop count in a multihop mesh network as shown in Fig. 11. For a 1 s synchronization interval and a 10-hop network, the six sigma timing accuracy would be 853.8 μs sufficient for a 1 ms guard time interval.

C. In-Field Calibration

The 47 p/min achieved frequency accuracy on the network calibrated on-chip frequency reference allows us to use it as a reference to frequency calibrate the RF LO in the field. The RF frequency can be divided down, measured using an on-chip counter and calibrated using an FLL as shown in Fig. 12. Alternatively, instead of using the network calibrated relaxation oscillator, the periodic RF packets can also be used as

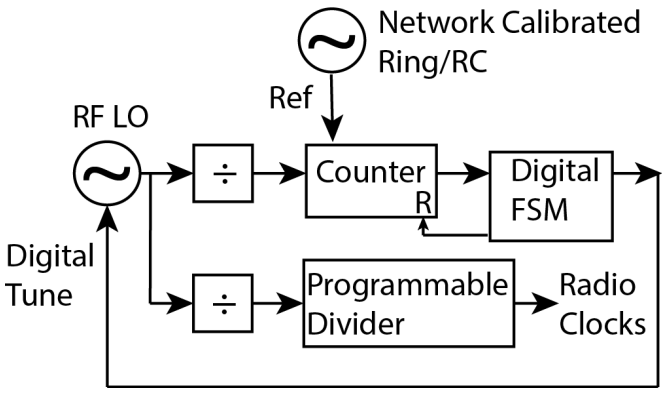


Fig. 12. RF oscillator frequency calibration using FLL.

a precise timing reference on-chip to calibrate the RF LO similarly as shown in Fig. 4.

The calibrated RF LO or the network calibrated relaxation oscillator can be used to generate all the radio relevant clocks, e.g., symbol clock (2-MHz for the IEEE 802.15.4 and 1-MHz for the BLE), ADC clock, discrete time signal processing, and the digital baseband processor clock. However, since the RF LO is a resonant oscillator, it has superior jitter performance and therefore is a preferred source for generating all the radio clocks as shown in Fig. 12.

D. Power Tradeoff of the Proposed Algorithm

A typical wireless sensor node uses at least one XTAL oscillator. It is either used as a real time clock (RTC) or as a high frequency reference for the wireless communication. In either case, the off-chip XTAL resonator adds to the bill-of-material cost and increases the profile of the wireless sensors. The state-of-the-art RTC consumes power in the nanowatts [21]. However, the high-frequency XTAL oscillator for the wireless communication can easily consume upto 100 μW of active power [22]. The proposed frequency calibration algorithm therefore reduces the power, cost, and profile of the wireless sensor nodes. Although the achieved calibrated on-chip frequency accuracy is not equal to an XTAL frequency reference but it is good enough to support scalable multihop wireless mesh networks using narrow band radios. The proposed algorithm requires a periodic RF traffic, which is not a concern in a dynamic network where the inherent network data traffic provides frequency synchronization. In networks where there is not enough data traffic, the cost of frequency synchronization is negligible. Assuming a 1 ms RF packet sent every 10 s and 1 mW of active radio power, this would correspond to 100 nW of average power due to frequency calibration. The leakage power on the modern SoCs can easily exceed this number so the proposed frequency calibration algorithm does not have a significant impact on the battery life.

V. Dynamic Network

When the wireless connectivity finds itself embedded into everyday objects, the amount of data generated is also expected to explode. Just to put numbers into context, the

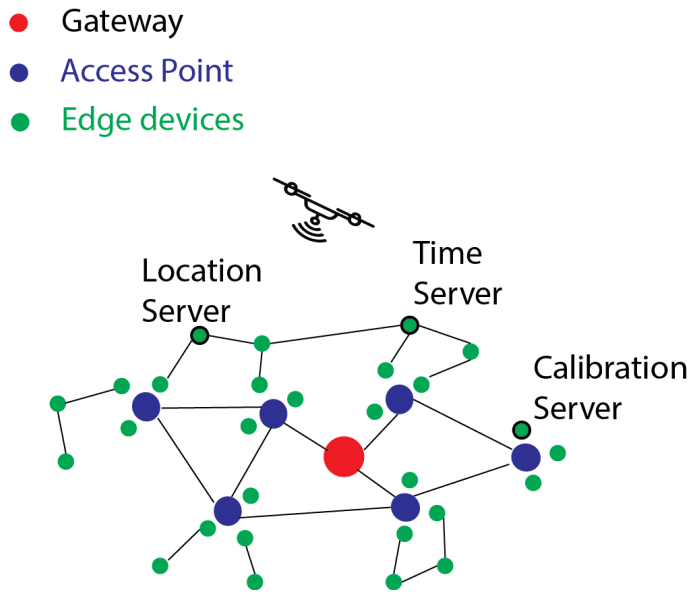


Fig. 13. Network topology.

Visa, Inc. handles on average about 2000 transactions per second, which is the data generated by interaction with just one smart object. To handle the increase data generated by human and machine-to-machine interaction, the edge nodes are expected to process the data locally due to energy efficiency, wireless capacity, and latency reasons. The wireless network is expected to provide real-time data visibility of physical spaces and sends relevant contextual alarms to the cloud infrastructure.

A. Network Topology

An example network topology that can achieve these objectives is shown in Fig. 13. The gateway serves as a data sink and provides the cloud connectivity. Since the edge devices are expected to operate from sources with limited energy capacity, e.g., printed batteries or batteryless operation from harvested energy; they are expected to be less robust to interference. As the robustness to interference requires high linearity in the analog/RF front-end and is typically traded with power or high quality off-chip passives that adds to the cost of these devices. Both tradeoffs are not desirable for the edge devices. Therefore, we expect the edge devices to communicate with APs. These APs are robust to interference at the cost of increased power consumption, can form wireless mesh network with neighbors and consume data locally from the edge devices to increase wireless network capacity. The edge devices can also form a wireless mesh network with their neighbors as shown in Fig. 13. We also envision different wireless nodes providing different network services, e.g., some wireless nodes will be location servers for indoor positioning, timer servers for frequency calibration, calibration servers for the sensors, etc.

The network shown in Fig. 13 is assumed to have a shared sense of time and a global schedule for communication. A schedule coordinates all of the communication in a TSCH network [23]. The time is divided into timeslots and in each

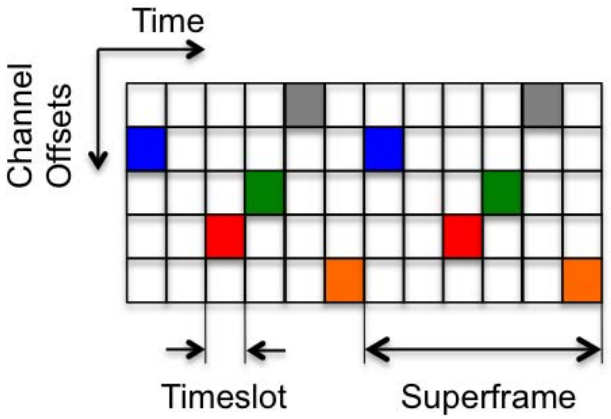


Fig. 14. Example communication schedule.

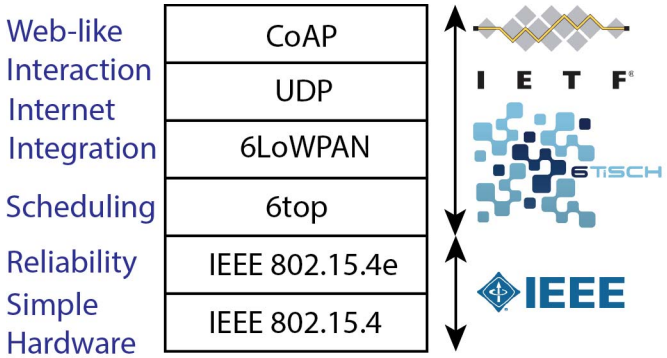


Fig. 15. Network stack.

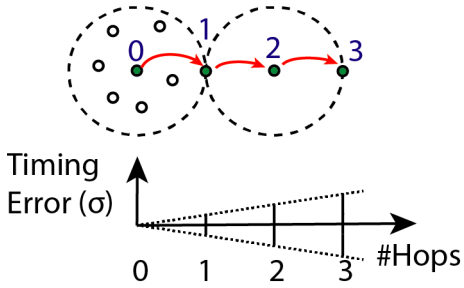


Fig. 16. Multihop time servers and their relative timing error.

timeslot a node can: transmit, receive, or sleep. A timeslot is long enough for a radio to send a packet and receive an acknowledgment. Therefore, both radio transmission and reception events can happen within a timeslot per device. For each transmit or receive slot, the schedule also indicates the neighbor to communicate with, and a channel offset to communicate on [23]. To increase throughput of the network more timeslots needs to be scheduled but at the cost of increasing the average power consumption of the nodes. A collection of consecutive time slots form a super frame that repeats indefinitely till the lifetime of the network. An example schedule is shown in Fig. 14. In this example six timeslots form a superframe that repeats indefinitely. Each row represents a different channel offset; multiple communications can happen in the network at the same time, but on a different channel offset without interference [23].

Since the edge devices (disposables) have finite lifetime (lasting few months up to several years), the devices are expected to join and leave the network dynamically. The network needs to allocate sufficient resources to discover new devices and dynamically adjust network topology to maintain a certain quality of service. One possibility is to use the time synchronization periodic RF packets to sample the RF space for device/network discovery as well.

In our implementation when an edge device wakes up from cold start, its local clock can be off by as much as ±5000 p/min as shown in Fig. 8. That means over a 1 s time interval the edge device's clock will accumulate about ±5 ms of timing error. If the periodic RF packets are sent every 1 s with a slot duration of 10 ms and if the edge device wakes up every 1 s for 10 ms, then it is guaranteed to find the network in l00 s.

This will correspond to 1% radio duty cycle during network discovery. If this latency is unacceptable and if sufficient energy is available either from the battery or the harvester then the device can listen continuously up to a second to find the network pings.

The networking stack is all open-source and standards' compliant as shown in Fig. 15 [23]. The IEEE 802.15.4 PHY layer allows implementing ultralow power and low cost wireless sensors. The IEEE 802.15.4e defines the MAC layer, 6 top provides a communication schedule, 6LoWPAN allows IPv6 integration and UDP and CoAP provides Web like interaction. Since the edge devices supports IPv6 packets they can be easily integrated with a cloud infrastructure to collect real time data of physical spaces.

B. Dynamic Schedule

A dynamic network would require a dynamic communication schedule (in contrast to a static communication schedule discussed in the previous section) to accommodate changes in the network topology due to mobility, finite node lifetime, data traffic, and interference. To achieve this goal in an energy efficient manner at scale, is an active area of research [24]. One possibility is to assume every node in a network has an independent schedule and the nodes negotiate with their neighbors to find a common communication channel (time slot and frequency offset). This would require the nodes to share their schedule with their neighbors. If we assume every node to have a fixed superframe length say 1 s with 10 ms timeslots, then each node can accommodate 1600 users with 16 RF channels for the IEEE 802.15.4 PHY. When a node wakes up, it starts scanning for the network beacons. If it does not find any network beacon it can then start broadcasting its own beacon to find neighbors. The network beacons can be broadcast at a predetermined set of frequencies and time interval.

When a device discovers a network through beacons then it negotiates a common communication schedule. There could be many edge cases that needs to be addressed before this could enable a completely decentralized dynamic schedule. We intend to address this in our future work. An example topology is shown in Fig. 16, where Node 0 is broadcasting network beacons and simultaneously serving as

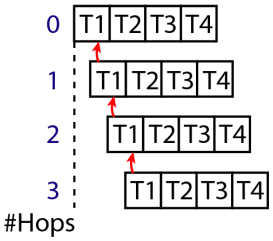


Fig. 17. Relative timing error between communication schedules as a function of hops.

a time server. Node 1 is receiving timing synchronization from Node 0 and serving as a time server for Node 2 which in turn is a time server for Node 3. The timing error grows as the number of hop count increases as discussed in the previous section and also shown in Fig. 16.

This timing error between the nodes will result in a relative shift in the communication schedule as shown in Fig. 17. If we assume that the Node 0 has the most accurate clock in the network then the timing error between Node 3 and Node 0 could be larger than a time slot (10 ms) or several time slots. But the communication between Node 3 and Node 0 can still happen as the relative timing error between one hop neighbors is bounded to be less than a guard time interval. This can extend the communication distance to arbitrary hops as long as the nodes' schedules are in time-sync with their immediate neighbors. However, applications that needs a precise time-stamp for sensor data or for control applications might dictate an upper limit on the absolute timing error that can be tolerated on the edge devices and in turn limit the number of hops.

In a dynamic network we expect the edge devices to be mobile. The mobility of wireless nodes in a network requires dynamic network topology. The rate of mobility (speed) has direct impact on the network average power consumption, as the RF space needs to be sampled more frequently to adapt the network topology. This presents a challenging resource optimization problem in energy constraint environments, which is an open area of research [25]. In order to discover a mobile device, we assume at least four communication events happen and the mobile device should cover a distance no more than a communicate range of a single device which we assume 20 m in our case. The four events are as follows: 1) the mobile device discovers the network and requests to join the network; 2) the network authenticates the mobile device and responds with a schedule; 3) the mobile device finds a common communication channel and respond to the network; and 4) the network acknowledges and accepts the new schedule. All these events can happen on the network beacon channel. We calculated the synchronization interval needed for different mobility rates and its impact on the average power consumption as shown in Fig. 18. A 10 m/s mobility corresponds to an average power consumption of 2 μW (assuming 1 mW of active radio power and 1 ms RF packet duration) for a synchronization interval (beacon rate) of 500 ms. For reference, a brisk walk is about 1.7 m/s and the sprint runners are about 12.2 m/s [26]. We acknowledge that the scalability of this approach in different RF traffic environments need to be addressed in the future work.

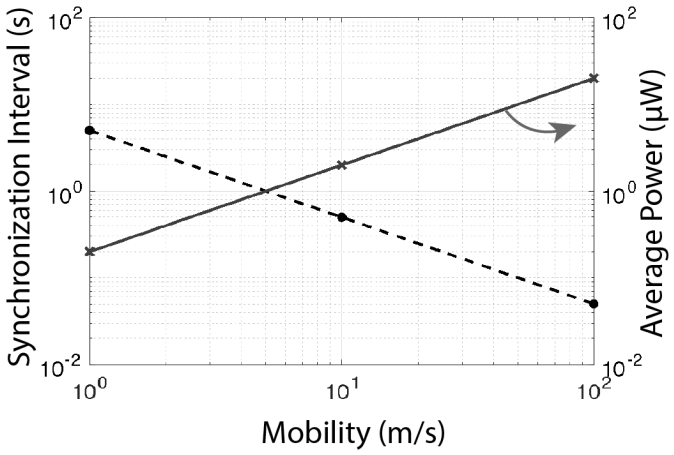


Fig. 18. Impact of mobility on communication overhead.

C. Network Programming

In the example network of Fig. 13, the firmware updates can be pushed to the network over the air or nodes can be programmed optically [27], using LED light infrastructure or a drone can fly by pushing the firmware updates using either of the two. Good software engineering design principles need to be incorporated, to make sure that the firmware updates do not constrict the network capacity and or have a heavy power penalty. This way the dynamic network can incorporate new personalities in an energy efficient manner.

VI. Conclusion

We presented measured silicon data of our latest crystal-free wireless sensor nodes and their time keeping ability. Introduced the notion of networked referenced FLL and studied the timing error of crystal free nodes in a multihop network. The standards' compliant networking stack enables seamless integration of these edge devices with existing networking and cloud technologies. Therefore, it is envisioned that as the compute capacity of the edge devices increases, more machine intelligence would be pushed toward the edge nodes. This will enable networks that manifest intelligence by adapting to their local environments and maintain their network objectives. We also discussed some preliminary ideas to enable the vision of dynamic networks. The future work requires to investigate the feasibility, scalability, and robustness of dynamic schedule in different RF environments. Additionally, we also intend to combine the time-frequency synchronization algorithm presented in this paper with the network beacons acquisition from cold-start.

Acknowledgment

The authors would like to thank the students, faculty, and sponsors of the Berkeley Wireless Research Center.

REFERENCES

[1] G. Bell, "Bell's Law for the birth and death of computer classes: A theory of the computer's evolution," *IEEE Solid-State Circuits Soc. Newslett.*, vol. 13, no. 4, pp. 8–19, 2008.

[2] Y. Lee, G. Chen, S. Hanson, D. Sylvester, and D. Blaauw, "Ultra-low power circuit techniques for a new class of sub-mm^3 sensor nodes," in *Proc. IEEE Custom Integr. Circuits Conf.*, 2010, pp. 1–8.

[3] Q. Ju and Y. Zhang, "Clustered data collection for Internet of Batteryless Things," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 2275–2285, Dec. 2017.

[4] Y. Lee *et al.*, "A modular 1mm^3 die-stacked sensing platform with low power I^{2}C inter-die communication and multi-modal energy harvesting," *IEEE J. Solid-State Circuits*, vol. 48, no. 1, pp. 229–243, Jan. 2013.

[5] G. Kim *et al.*, "A millimeter-scale wireless imaging system with continuous motion detection and energy harvesting," in *IEEE VLSI Circuits Dig. Tech. Papers*, Honolulu, HI, USA, 2014, pp. 1–2.

[6] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," *Oper. Syst. Design Implement.*, vol. 36, pp. 147–163, Dec. 2002.

[7] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *Proc. Embedded Netw. Sensor Syst.*, Nov. 2003, pp. 138–149.

[8] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, "The flooding time synchronization protocol," in *Proc. Embedded Netw. Sensor Syst.*, Nov. 2004, pp. 39–49.

[9] T. Chang, T. Watteyne, K. Pister, and Q. Wang, "Adaptive synchronization in multi-hop TSCH networks," *Comput. Netw.*, vol. 76, pp. 165–176, Jan. 2015.

[10] Dust Networks. *SmartMesh*. Accessed: Sep. 3, 2018. [Online]. Available: http://www.analog.com/media/en/technical-documentation/data-sheets/5800ipmfa.pdf

[11] K. S. J. Pister and L. Doherty, "TSMP: Time synchronized mesh protocol," in *Proc. IASTED Int. Symp. Distrib. Sensor Netw.*, Orlando, FL, USA, 2008, pp. 391–398.

[12] O. Khan *et al.*, "Frequency reference for crystal free radio," in *Proc. IEEE Int. Freq. Control Symp. (IFCS)*, New Orleans, LA, USA, 2016, pp. 1–2.

[13] A. Paidimarri, D. Griffith, A. Wang, A. P. Chandrakasan, and G. Burra, "A 120nW 18.5kHz RC oscillator with comparator offset cancellation for ±0.25% temperature stability," in *Int. Solid-State Circuits Conf. Dig. Tech. Papers*, San Francisco, CA, USA, 2013, pp. 184–185.

[14] D. Griffith, J. Murdock, P. T. Røine, and T. Murphy, "5.9 A 37μW dual-mode crystal oscillator for single-crystal radios," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, San Francisco, CA, USA, 2015, pp. 1–3.

[15] Y. Tokunaga, S. Sakiyama, and A. Matsumoto, and S. Dosho, "An on-chip CMOS relaxation oscillator with voltage averaging feedback," *IEEE J. Solid-State Circuits*, vol. 45, no. 6, pp. 1150–1158, Jun. 2010.

[16] K. Sundaresan, P. E. Allen, and F. Ayazi, "Process and temperature compensation in a 7-MHz CMOS clock oscillator," *IEEE J. Solid-State Circuits*, vol. 41, no. 2, pp. 433–442, Feb. 2006.

[17] J. Lee and S. Cho, "A 10MHz 80μW 67 p/min/°C CMOS reference clock oscillator with a temperature compensated feedback loop in 0.18μm CMOS," in *Proc. IEEE Symp. VLSI Circuits*, 2009, pp. 226–227.

[18] S. Mesri, "Design and user guide for the single chip mote digital system," Elect. Eng. Comput. Sci. Dept., Univ. California at Berkeley, Berkeley, CA, USA, Rep. UCB/EECS-2016-71, May 2016.

[19] D. W. Allan, "Conversion of frequency stability measures from the time-domain to the frequency-domain, vice-versa and power-law spectral densities," Jan. 26, 2012. [Online]. Available: http://www.allanstime.com/Publications/DWA/Conversion_from_Allan_variance_to_Spectral_Densities.pdf

[20] B. P. Lechêne *et al.*, "Organic solar cells and fully printed supercapacitors optimized for indoor light energy harvesting," *Nano Energy*, vol. 26, pp. 631–640, Aug. 2016.

[21] N. Shafiee, S. Tewari, B. Calhoun, and A. Shrivastava, "Infrastructure circuits for lifetime improvement of ultra-low power IoT devices," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 9, pp. 2598–2610, Sep. 2017.

[22] M. Ding *et al.*, "5.3 A 95μW 24MHz digitally controlled crystal oscillator for IoT applications with 36nJ start-up energy and >13× start-up time reduction using a fully-autonomous dynamically-adjusted load," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, San Francisco, CA, USA, 2017, pp. 90–91.

[23] D. Dujovne, T. Watteyne, X. Vilajosana, and P. Thubert, "6TiSCH: Deterministic IP-enabled Industrial Internet (of Things)," *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 36–41, Dec. 2014.

[24] S. Duquennoy, B. A. Nahas, O. Landsiedel, and T. Watteyne, "Orchestra: Robust mesh networks through autonomously scheduled TSCH," in *Proc. 13th ACM Conf. Embedded Netw. Sensor Syst.*, New York, NY, USA, 2015, pp. 337–350.

[25] A. Mehta, B. Kerkez, S. D. Glaser, and K. S. J. Pister, "TDMA-based dual-mode communication for mobile wireless sensor networks," *Sensors*, vol. 12, no. 12, pp. 16194–16210, 2012.

[26] Wikipedia. *Speed*. Accessed: Feb. 23, 2018. [Online]. Available: https://en.wikipedia.org/wiki/Speed

[27] G. Kim *et al.*, "A 695 pW standby power optical wake-up receiver for wireless sensor nodes," in *Proc. IEEE Custom Integr. Circuits Conf.*, San Jose, CA, USA, 2012, pp. 1–4.

Osama Khan received the M.S. and Ph.D. degrees from the University of Michigan, Ann Arbor, MI, USA, in 2010 and 2014, respectively.

In 2009, he was with the RF Division, Qualcomm, San Diego, CA, USA. In 2014, he was with Psikick Inc., Santa Clara, CA, USA, an ultralow-power wireless startup, where he developed a low-power crystal oscillator and a short-range radio front-end. He was a Post-Doctoral Research Scholar with the Ubiquitous SWARM Lab, University of California at Berkeley (UC Berkeley), Berkeley, CA, USA, from 2014 to 2017, where he also taught and designed the new course "SoC for the IoT." He is currently a Visiting Research Scholar with the Ubiquitous SWARM Lab, UC Berkeley, and the founder of a stealth startup. He holds four patents and has authored or co-authored several relevant publications. His current research interests include battery independent, robust, adaptive microsystems and their applications.

David C. Burnett received the B.S. and M.S. degrees in electrical engineering (focusing on circuits and embedded systems) from the University of Washington, Seattle, WA, USA, and is currently pursuing the Ph.D. degree in electrical engineering and computer sciences at the University of California at Berkeley, Berkeley, CA, USA, with Prof. K. S. J. Pister.

He was a Technical Staff Member with Sandia National Laboratories, Livermore, CA, USA, a Visiting Lecturer with the Da Nang University of Technology, Da Nang, Vietnam, and a Robotics Engineer with the McMurdo Station, McMurdo Sound, Antarctica. His current research interests include wireless embedded systems and mixed-signal integrated circuits.

Dr. Burnett also served on ACM SIGGRAPH conference committees, serving as a Submissions Juror and responsible for special technical projects and data networks.

Filip Maksimovic received the B.S. degree in electrical engineering and in aerospace engineering from the University of Colorado at Boulder, Boulder, CO, USA. He is currently pursuing the graduation degree at the Berkeley Wireless Research Center, University of California at Berkeley (UC Berkeley), Berkeley, CA, USA, and the Berkeley Sensor and Actuator Center, UC Berkeley.

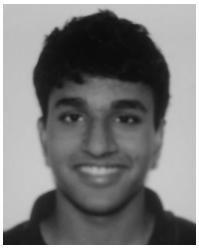
His current research interests include low-power RF and mm-wave wireless transceivers.

Brad Wheeler received the B.S. degree in electrical engineering from the University of Missouri, Columbia, MO, USA. He is currently pursuing the graduation degree at the Berkeley Sensor and Actuator Center, University of California at Berkeley, Berkeley, CA, USA.

His current research interest includes low power circuit design for crystal-free transceivers.

Sahar Mesri received the B.S. and M.S. degrees in electrical engineering and computer sciences (focusing on digital integrated circuit designs for energy-efficient wireless sensor solutions) from the University of California at Berkeley, Berkeley, CA, USA.

She is currently a Digital Design Engineer with Qualcomm, where she performs research on LDO regulator control and chip-level integration for power management ICs on mobile devices.



Arvind Sundararajan is currently pursuing the B.S. degree in electrical engineering and computer sciences at the University of California at Berkeley, Berkeley, CA, USA.

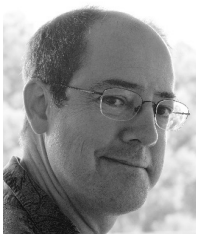
Bob L. Zhou received the B.S. degree in electrical engineering and computer sciences from the University of California at Berkeley, Berkeley, CA, USA, where he is currently pursuing the Ph.D. degree with a concentration in analog integrated circuit design, advised by Prof. E. Alon.

His current research interests include energy-efficient signal processing and high-speed link design for VLSI.

Ali M. Niknejad (S'93–M'00–SM'10–F'13) received the B.S.E.E. degree from the University of California at Los Angeles, Los Angeles, CA, USA, in 1994, and the master's and Ph.D. degrees in electrical engineering from the University of California at Berkeley (UC Berkeley), Berkeley, CA, USA, in 1997 and 2000, respectively.

He is currently a Professor with the Electrical Engineering and Computer Science Department, UC Berkeley, and the Faculty Director of the Berkeley Wireless Research Center. He is a co-founder of HMicro and an inventor of REACH technology, which have the potential to deliver robust wireless solutions to the healthcare industry, and a co-founder of RF Pixels, a 5G technology startup. His current research interests include wireless and broadband communications and biomedical imaging and sensors, integrated circuit technology (analog, RF, mixed-signal, and millimeter-wave), device physics and compact modeling, and applied electromagnetics.

Dr. Niknejad was a recipient of the 2017 IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS Darlington Best Paper Award, the 2017 Most Frequently Cited Paper Award from 2010 to 2016 of the Symposium on Very Large-Scale Integration Circuits, the CICC 2015 Best Invited Paper Award, and the 2012 ASEE Frederick Emmons Terman Award for his textbook on electromagnetics and RF integrated circuits. He was a co-recipient of the 2013 Jack Kilby Award for Outstanding Student Paper for his research on an efficient quadrature digital spatial modulator at 60 GHz, the 2010 Jack Kilby Award for Outstanding Student Paper for his research on a 90-GHz pulser with 30 GHz of bandwidth for medical imaging, and the Outstanding Technology Directions Paper of ISSCC 2004 for co-developing a modeling approach for devices up to 65 GHz.



Kristofer S. J. Pister received the B.A. degree in applied physics from the University of California at San Diego, La Jolla, CA, USA, in 1986, and the M.S. and Ph.D. degrees in electrical engineering from the University of California at Berkeley (UC Berkeley), Berkeley, CA, USA, in 1989 and 1992, respectively.

From 1992 to 1997 he was an Assistant Professor of electrical engineering with the University of California at Los Angeles, Los Angeles, CA, USA, where he helped to develop the graduate MEMS curriculum and coined the phrase "smart dust." Since 1996, he has been a Professor of electrical engineering and computer sciences with UC Berkeley. In 2003 and 2004, he was on leave from UC Berkeley as the CEO and then a CTO with Dust Networks, Hayward, CA, USA, a company he founded to commercialize wireless sensor networks. He participated in the creation of several wireless sensor networking standards, including Wireless HART (IEC62591), IEEE 802.15.4e, ISA100.11A, and 6TiSCH minimal. He has participated in many government science and technology programs, including DARPA ISAT and the Defense Science Study Group, and is currently a member of JASON. His current research interests include MEMS, micro robotics, and low-power circuits.