



# Enabling Bit-by-Bit Backscatter Communication in Severe Energy Harvesting Environments

Pengyu Zhang and Deepak Ganesan, *University of Massachusetts Amherst*

<https://www.usenix.org/conference/nsdi14/technical-sessions/presentation/zhang>

This paper is included in the Proceedings of the  
11th USENIX Symposium on Networked Systems  
Design and Implementation (NSDI '14).

April 2–4, 2014 • Seattle, WA, USA

ISBN 978-1-931971-09-6

Open access to the Proceedings of the  
11th USENIX Symposium on  
Networked Systems Design and  
Implementation (NSDI '14)  
is sponsored by USENIX

# Enabling Bit-by-Bit Backscatter Communication in Severe Energy Harvesting Environments

Pengyu Zhang, Deepak Ganesan  
{pyzhang, dganesan}@cs.umass.edu  
University of Massachusetts Amherst

## Abstract

Micro-powered wireless sensors present new challenges due to the severe harvesting conditions under which they need to operate and their tiny energy reservoirs. However, existing low-power network stacks make a slew of design choices that limit the ability to scale down to such environments. We address these issues with QuarkNet, a backscatter-based network stack that is designed to enable continuous communication even if there is only enough harvested energy to transmit a few bits at a time while simultaneously optimizing throughput across a network of micro-powered devices. We design and implement QuarkNet on a software radio based RFID reader and the UMass Moo platform, and show that QuarkNet increases the communication distance by  $3.5\times$  over Dewdrop,  $9\times$  over Buzz, and is within 96% of the upper bound of achievable range. QuarkNet also improves the communication throughput by  $10.5\times$  over EPC Gen 2,  $5.8\times$  over Dewdrop, and  $3.3\times$  over Flit for tag-to-reader communication and by  $1.5\times$  over EPC Gen 2 for reader-to-tag communication.

## 1 Introduction

The idea of networks of perpetual self-powered sensing, communication and actuation devices that can fly in swarms, swim through the bloodstream, and navigate through pipes and debris has propelled the imagination of science fiction writers for decades, but reality is finally catching up. While practical instantiations of self-powered devices have largely been limited to RFID tags, a new generation of micro-powered devices promises to go beyond simple identification towards computation, sensing, and actuation. Among the key technology trends enabling this vision are advances in micro-harvesters that scavenge energy from light, electro-magnetic waves, vibrations, temperature, and other sources [7]. Such micro-harvesters enable platforms to cut their reliance on stored energy in batteries, thereby enabling true miniaturization and perpetual operation [24, 25].

While micro-powered devices present an exciting opportunity, they present tremendous challenges due to the amount of energy they harvest and the sizes of their energy reservoirs. The amount of harvested power using a micro-energy harvester is of the order of *nano*Watts to  $\mu$ Watts, which is three to six orders of magnitude lower than the average power draw of a Mote. At first glance, this seems to suggest that if we wait long enough, the device can trickle charge to accumulate sufficient energy to operate similar to a battery-powered device. But there are three problems. First, long delays before performing useful work are often unacceptable, particularly for continuous sensing and communication. Second, the voltage from the incoming energy source is often low, therefore accumulating energy into an energy reservoir requires boosting voltage which is wasteful compared to incoming energy (imagine pumping water up a hill to store for future use). Third, micro-powered platforms often have small energy reservoirs to reduce form-factor. For example, the Intel WISP [5] and Michigan Micro Mote ( $M^3$ ) [15] have energy reservoirs that are 4 – 6 orders of magnitude smaller than a coin cell respectively.

The dual limitations of low harvesting rates and tiny energy reservoirs have profound implications on the design of a network stack for micro-powered devices. Every communication task needs to be small enough to fit within the available energy in the reservoir. Enabling communication despite such minuscule energy budgets is akin to working on a micro-sculpture — optimizations at the granularity of individual instructions, bits, on-off transitions, and analog-to-digital conversions are needed. To compound matters, small short-term variations in harvesting conditions that typically would be smoothed out by a larger energy reservoir begin to impact system operation, and can cause an order of magnitude variation in available energy for a task.

These challenges are not addressed by existing protocols such as EPC Gen 2. RFID tags operate solely on continuous harvested power without buffering energy,

therefore EPC Gen 2 assumes a regime where the tag either has enough power to operate continuously, or not at all. In contrast, micro-powered devices can buffer energy, thereby enabling operation in regimes where there is insufficient power to operate continuously, but enough power to operate intermittently.

Recent systems such as MementOS [18] and Dewdrop [8] tackle this problem in different ways. Both these systems use backscatter similar to RFIDs, but the challenge is fitting the communication stack within the energy budget. MementOS introduces checkpoints within computation tasks such that it can recover from outages and continue execution. Dewdrop continually adapts task execution to harvesting conditions such that the efficiency of execution is optimized. To evaluate the ability of these systems to scale down, we consider two harvesting conditions — strong light (2000 lux) and natural indoor light (200 lux), both of which should, in principle, provide enough energy to operate a micro-powered sensor. But while both MementOS and Dewdrop operate under strong light, they are inoperable under natural light.

The inability of current systems to scale-down illustrates the central challenge in designing a network stack for micro-powered devices. A wireless network stack involves a variety of tasks that are simply too large to fit into the extreme energy constraints of this regime. Even the core primitive of a network stack — packet transfer — can involve hundreds of instructions and bits. In this work we ask the following question — what are the general principles that we, as systems designers, should use to enable these micro-powered platforms to communicate continuously despite trickles of energy, tiny energy reservoirs, and dynamic harvesting conditions?

We present QuarkNet, a network stack that embodies a simple but powerful abstraction — by fragmenting a backscatter network stack into its smallest atomic units, we can enable the system to scale down to resource-impooverished regimes. The fundamental building block of QuarkNet is the ability to dynamically fragment a larger packet transfer into  $\mu$ frames that can be as small as a single bit under severe energy constraints, and as large as the whole packet when sufficient energy is available. On top of this abstraction, we design a variety of innovative techniques to handle dynamic frames that can be abruptly terminated in low energy settings, maximize throughput by tracking harvesting dynamics in a low-overhead manner, interleave  $\mu$ frames across nodes to maximize throughput despite different harvesting rates, and minimize overhead across the entire stack.

Our results on a USRP reader and Moo nodes show that:

- The maximum communication distance achieved by QuarkNet is 21 feet, 3.5 $\times$  longer than Dewdrop and 4.2 $\times$  longer than EPC ID transfer. QuarkNet

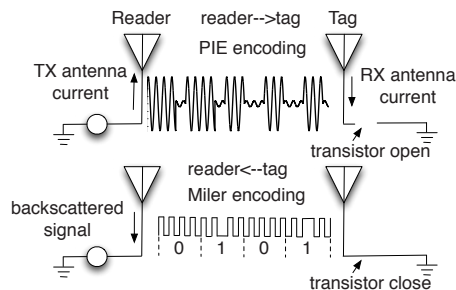


Figure 1: Backscatter signaling at PHY.

achieves close to the maximum achievable range, beyond which decoding even a single bit fails.

- The minimum illuminance required for QuarkNet to operate is 150 lux, which is 13 $\times$  lower than the 2000 lux requirement of 12 byte EPC ID transfer. This suggests that  $\mu$ frame can operate when a device is powered by natural indoor illuminance, dramatically increasing utility of micro-powered devices for practical deployments.
- The throughput of QuarkNet for node to reader transfer is 18 kbps, 10.5 $\times$  higher than EPC Gen 2, 5.8 $\times$  higher than Dewdrop, and 3.3 $\times$  higher than Flit. For reader to node transfer, we obtain throughput of 1.5 kbps, 2 $\times$  higher than a battery-assisted device which uses the EPC Gen 2 write command.
- When ten nodes transmit simultaneously to a reader, we achieve a throughput of 16.5 kbps as a result of variability-aware scheduling and interleaving of  $\mu$ frames, which is 5.4 $\times$  higher than the throughput when devices are inventoried individually. Flit and EPC Gen 2 obtain zero throughput in this case.

## 2 Case for $\mu$ frames

A backscatter radio is designed to both provide power to a passive device as well as to enable communication. As shown in Figure 1, the reader provides a carrier wave, which can be reflected by a passive device back to the reader with its own information bits. This makes backscatter a considerably more energy-efficient communication mechanism compared to active radios, and ideally suited to the constraints of micro-powered devices. The Intel WISP [5] and UMass Moo [27] are examples of backscatter-enabled sensor platforms.

Despite the energy benefits of backscatter radios, existing network stacks achieve only short communication range and low throughput. We make an empiric argument these limitations are, in part, due to the design of the network stack. To do this, we compare the range and throughput of existing network stacks versus achievable performance. Our experiment uses a UMass Moo [27]

Table 1: EPC Gen 2 vs Achievable Performance.

	Range(ft)	Throughput(kbps)	SNR(dB)
Gen 2	3.6±0.8	3.6±0.3	9.6±1
Optimal	18.6±3.3	21.7±3.7	6.9±0.9

and a USRP reader [9]. Since combining multiple micro-power sources can enable higher performance, broader operating conditions, and enable wider range of applications, we augment the Moo with a small solar panel [11, 7, 10]. We vary the distance from the reader by small steps, and at each step, we vary RF power from 17dBm to 26dBm, while not changing the light levels (normal indoor light).

To measure the achievable range, we look at the raw backscattered signal at the reader, and find the distance at which the reader is unable to decode even a single bit. This would be the edge of the communication range for our hardware platform.

Measuring the maximum achievable throughput is harder since it is influenced by several system parameters including voltage at the energy reservoir when communication starts, the length of each transmission unit, and control overheads associated with the protocol. We brute-force search across all possible voltages and packet lengths to find the setting that results in the maximum number of transistor flips at the node. We then convert the transistor flips to a maximum number of bits transmitted using the default Miller-4 encoding scheme, and assume zero control overhead for each packet, which gives us an estimate of the maximum throughput.

Table 1 shows the range and throughput while executing the EPC Gen 2 stack (used in Mementos [18], Dewdrop [8], and Blink [30]) versus achievable limits. We see that the achievable range is 18.6 feet, which is over 5× longer than the communication range of EPC Gen 2. Surprisingly, we find that EPC Gen 2 ceases to operate even when its SNR is 9.6dB, 1.4× higher than the optimal case. Similarly, we see that the achievable throughput is 21.7 kbps, whereas EPC Gen 2 achieves barely 1.7 kbps, an order of magnitude difference.

We now investigate the fundamental factors underlying this performance gap, and outline the core challenges that need to be addressed to bridge the gap.

**Challenge 1: Variable energy per transmission** A key challenge in designing a backscatter network stack is handling variability in the amount of energy accumulated in the energy reservoir. To understand the reasons, let us look at how micro-powered devices work. As shown in Figure 2, micro-powered devices operate in a sequence of charge-discharge cycles since there is too little energy to continually operate the device. The device sleeps for a short period during which it harvests energy and charges a small energy reservoir, and then wakes up and transmits

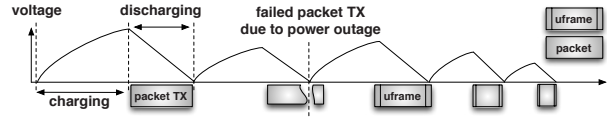


Figure 2: Energy harvesting systems.

a packet during which the reservoir discharges.

There are several reasons why it is difficult to anticipate how much energy will be available in each discharge cycle. First, if harvesting conditions are too low, it is often too expensive to push more energy into a reservoir due to the inefficiencies of stepping up the voltage. As a result, the maximum amount of energy that can be accumulated depends on current harvesting conditions. Second, RF energy harvested by a node depends on how much energy is output by the reader. When a reader is doing nothing, the RF output power is roughly constant. However when a reader is communicating, this RF carrier wave is being modulated which changes the amount of harvested energy. In a multi-node network, the reader is communicating with different nodes, therefore harvesting rates continually vary at each node. Third, even if the node were to wait until it has a certain amount of energy prior to communication, this requires measurement of energy levels using analog-to-digital conversions (ADC). Each ADC operation consumes 327 uJ on the Moo platform [27], which is equal to the energy budget for transferring 27 bits of data. Such overhead is far too substantial on a micro-powered platform.

While choosing a smaller transmission unit might seem like a straightforward solution to this problem, this over-simplifies the design challenge. As the distance between the node and reader increases to the limit of the achievable range in Table 1, the number of bits that can be successfully transmitted reduces. Thus, we need to use frames that may be as small as one or a few bits in size when the energy levels are low, which requires a network stack that can scale down to unprecedented levels. But such scale down often comes at the expense of throughput, which suffers due to the overheads associated with each transmission, including preambles, headers, and hardware transition overheads. To simultaneously optimize throughput, it is important to transmit as large a transmission as is possible given available energy. Thus, the problem faced by a node is that it needs to scale down its transmission unit to the bare minimum under poor harvesting conditions, while scaling up to improve throughput when the conditions allow.

**Challenge 2: Variable harvesting rate**

The energy harvesting rate has significant impact on the communication throughput, since higher harvesting rate means that more energy can be used for data transfer.

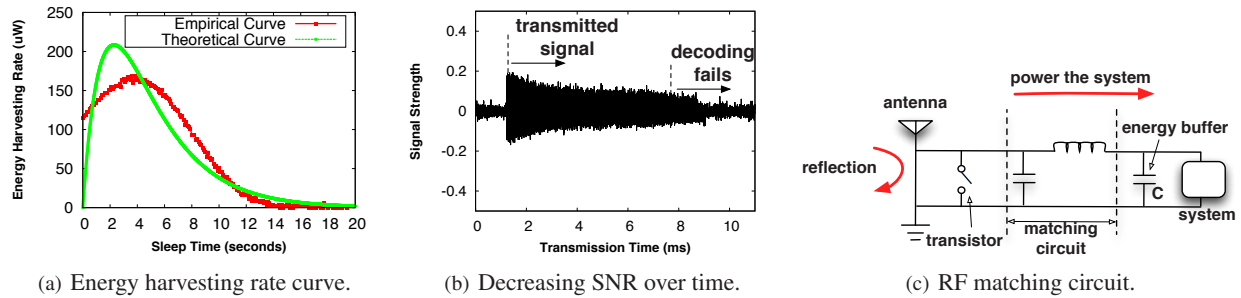


Figure 3: Factors that impact communication throughput.

While energy harvesting rate might seem like a characteristic of the harvesting source, system parameters have a surprisingly high impact. Figure 3(a) shows the empirically measured harvesting rate as we vary the amount of time for which the node replenishes energy between two transmissions. The results are counter-intuitive — while one might expect more energy to be harvested over time, the harvesting rate drops to zero for longer sleep durations.

This observation can be explained analytically by looking at how capacitors buffer energy. The charging process of a capacitor follows its charging equation  $V = V_{max}(1 - e^{-t_s/\tau})$ , where  $t_s$  is the sleep time,  $\tau$  is the RC circuit time constant, and  $V_{max}$  is the maximum voltage to which the capacitor can be charged under the current harvesting conditions. Its energy harvesting rate follows the equation:  $H = C \times V_{max}^2 \times \tau^{-1}(1 - e^{-t_s/\tau})e^{-t_s/\tau}$ . When the harvesting conditions are constant (i.e.  $V_{max}$  and  $\tau$  are fixed),  $H$  is a concave function of  $t_s$ , which is shown both analytically and empirically in Figure 3(a). When harvesting conditions change, both  $V_{max}$  and  $\tau$  change, therefore the maximum operating point changes as well. Thus, to optimize throughput, it is important to adapt to current harvesting conditions, and continually track the maximum harvesting point.

One factor that should not be overlooked is keeping the overhead of adaptation low. Most methods to track the charging rate of batteries and capacitors use analog-to-digital conversions to obtain the voltage at the energy reservoir. This overhead is minuscule for most platforms, but a significant part of the harvested energy in our case. Thus, it is important to minimize such overheads while adapting to harvesting conditions.

**Challenge 3: Time-decaying SNR** A peculiar aspect of backscatter communication is that the signal to noise ratio (SNR) of the received signal at the reader degrades steadily as the size of the transmission unit increases. Figure 3(b) shows that the signal strength of a node response decreases gradually from 0.18 at 1.5ms to 0.05 at 8ms during the transmission process. While decoding the initial part of the transmission is straightforward due

to high SNR, it becomes much more challenging after about 8ms since the SNR is too low for reliable decoding, resulting in packet losses.

In order to understand why this happens, let us look at how a backscatter radio works. A backscatter radio provides power to a passive device and enables communication. The reader provides a carrier wave, which can be reflected by a passive device back to the reader with its own information bits. The modulation is achieved by toggling the state of the transistor of a backscatter device shown in Figure 3(c). Since the same RF power source is shared by different system components, some fraction of the incoming power is used to operate the micro-powered device while the rest is reflected back to the reader for communication. The exact fraction depends on the state of the energy reservoir  $C$  and the state of the matching circuit, which is designed to charge the energy reservoir  $C$  when the voltage is low. Therefore, when the transmission begins,  $C$  is fully charged, the antenna resistance is mismatched with the resistance of other hardware components of the system. As a result, most of the incoming power will be reflected back to the reader, which receives a strong signal that can be easily decoded. As the transfer progresses,  $C$  slowly discharges, and the antenna resistance matches the resistance of the system load. Therefore, most of the incoming power is harvested to operate the system, and less RF power is reflected. This leads to decreased backscatter signal strength at the reader, and consequently, packet losses. Thus, to ensure that packets are received successfully, the tag needs to adapt the size of each packet such that the SNR at the tail of the packet is higher than the minimum decoding requirement.

**Challenge 4: Energy-induced reader to node losses**

While time-decaying SNR only presents a problem when a node communicates with a reader, reader to node communication presents other challenges. The central issue is that the energy level on the receiving node might dip below the low watermark at any point during the reception, at which point the node has to shut off its RF circuit and go to sleep to recharge. The reader, however, does not know that the node has gone to sleep, and only

realizes this fact after a timeout.

While such losses can be attributed to small energy harvesting variations at longer ranges, we observed to our surprise that such losses occur even when a tag is placed relatively close to the reader — 40% losses at 2 ft. The reason for this behavior is that data transfer from the reader to tag comes at the expense of RF power being transmitted to the tag. Since the reader is actively transmitting to the tag, the carrier wave from the reader to tag is intermittent, causing substantial variations in RF energy harvesting and consequently variations in energy levels at the tag.

The energy dynamics at the tag makes it difficult to use reader-side estimation to identify the best transmission unit to communicate with a tag. In addition, explicitly providing information to the reader about the current energy level has considerable overhead while not being robust to dynamics. Thus, the challenge we face is that the reader needs to have a way of knowing the instantaneous energy state at the tag, and detecting its shut-off point without using cumbersome protocol-level mechanisms to enable this information exchange.

### 3 Fragmenting packets into $\mu$ frames

At the heart of QuarkNet is a simple hypothesis — by breaking down packet transmission into its smallest atomic units, which we refer to as  $\mu$ frames, we can enable the system to scale down to severely limited harvesting regimes. We address the challenges in enabling such extreme fragmentation both for node-to-reader and reader-to-node communication.

#### 3.1 Fragmentation at bit boundaries

The first question we ask is: what are the practical considerations that determine how we can dynamically fragment a logical transmission unit (packet) into  $\mu$ frames? Ideally, we would want to insert fragment boundaries at arbitrary positions within a packet so that we can make  $\mu$ frames as small or large as needed, however, this makes decoding extremely error-prone.

To understand where to place fragmentation boundaries, we need to give some more detail about how backscatter modulation works. Figure 4 shows a sequence of backscatter pulses that compose bits in a packet. Backscatter modulation uses On-Off-Keying (OOK), therefore each bit is composed of a sequence of on and off pulses. As can be seen, the template for a '0' pulse and '1' pulse differ only slightly in the phase information of the pulses within the bit.

The key observation is that placing boundaries at certain points in a packet can be done without disrupting the phase information required for decoding, whereas other

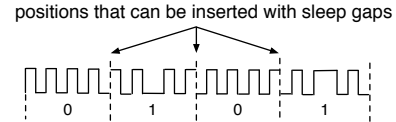


Figure 4: Sleep gaps can be inserted into backscatter pulses at various position (lines with dots).

boundaries would disrupt decoding. For example, suppose that a fragment boundary is inserted between two adjacent bits, the phase information of each bit is maintained, thereby not impacting the ability to match the template to the bit. On the other hand, suppose that a fragment boundary is inserted within a single bit, the phase information within the bit is disrupted, thereby causing a mismatch at the decoder between received bit pulses and its template.

This leads us to a general principle for fragmenting a packet into  $\mu$ frames —  $\mu$ frame boundaries can be inserted between bits but not within a bit. The ability to fragment at any bit boundary gives us the requisite combination of fine-grained fragmentation as well as low decoding error.

#### 3.2 Tuning inter- $\mu$ frame gap

We now have a method for fine-grained fragmentation of larger packets, but how do we use this to dynamically fragment packets? How do we decide the length of each  $\mu$ frame and the sleep gap between  $\mu$ frames where the node replenishes energy?

We first answer this question for node-to-reader communication. In this case, we need to address two of the challenges discussed in §2: a) how to optimize throughput by operating at the optimal harvesting rate, and b) how to ensure the tail of each  $\mu$ frame transmitted from a node has sufficiently high SNR to be decoded at the reader.

**Gradient descent algorithm** As can be seen in Figure 3(a), the harvesting rate curve is a concave function of the gap between  $\mu$ frames (under constant harvesting conditions). A fast and effective method for converging to the optimum of a concave function is to use gradient descent [2]. The gradient descent algorithm works as follows: first, we start with an initial guess about the optimal sleep gap. Second, we compute the gradient at this point, and look for the direction of the positive gradient. Third, we take a step along the direction of the positive gradient with step size proportional to the gradient. We repeat this process until convergence (i.e. step is smaller than a threshold). The algorithm takes large steps when the gradient is steep (i.e. point is far from optimal), and small as the gradient reduces (i.e. point is near optimal).

What if the harvesting conditions change and the curve itself shifts to create a new optimal harvesting point? Our gradient descent-based sleep gap adaptation algorithm operates continually — once it converges to the optimal, it periodically probes the gradient at the current optimal, and moves along the positive gradient if the optimal harvesting rate changes. In this manner, the algorithm seamlessly adapts to such dynamics.

**Handling time-varying SNR** We need to add another constraint to the gradient descent algorithm — the SNR at the tail of the frame should be higher than the decoding threshold at the reader, otherwise the frame cannot be decoded. This constraint is easy to add since it simply translates to a bound on the maximum length of the inter- $\mu$ frame gap. Since the length of the gap directly impacts the length of the  $\mu$ frame, capping the inter- $\mu$ frame gap ensures that the length of each  $\mu$ frame is lower than the decoding threshold. The only change to the gradient descent algorithm is that a step cannot exceed the maximum inter- $\mu$ frame as determined by the SNR constraint.

**Duty-cycling the radio** One important aspect of the inter- $\mu$ frame gap is that we shut off the node's RF circuit for this length of time. In a multi-node environment, the reader is constantly talking to other nodes, so leaving the RF circuit on results in substantial reception overhead since backscatter is a broadcast-based protocol, and wakes up every node that has its radio circuit turned on. To avoid these costs, we turn off the RF circuit during the recharge cycle. Once the node has slept for the intended duration, it switches on its RF circuit. One side-effect of our decision to turn off the RF circuit during gaps is that the reader now has to be more careful to avoid transmitting to a node or scheduling a node for transmission while it is inactive. We return to this question in §4.2.

### 3.3 Remote interrupts

We now turn to  $\mu$ frame adaptation for communication from a reader to a node. As described in §2, the key challenge is that the reader cannot detect when a node's energy level drops below a low watermark, and it should stop transmitting. Similarly, once a node has gone to sleep, a reader does not know when it will wake up for the next  $\mu$ frame. Given these constraints, how can we enable reader-to-node communication?

**Estimating  $\mu$ frame length** Our idea is to use a remote interruption mechanism, where a node issues an in-band interrupt during reader transmission, and informs the reader that it has reached a low-energy state. This remote interrupt is generated by toggling its transistor while receiving the current frame. In other words, the remote interrupt is a signal that is overlaid on the same

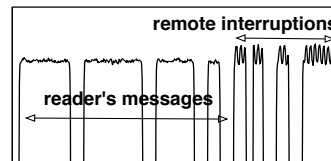


Figure 5: In-band remote interruptions from nodes.

time-slot and frequency signal as the message from the reader to node.

How can the reader decode an in-band interrupt from the node? The key insight is that the reader modulates the carrier by toggling the carrier wave whereas the node communicates back to the reader by changing the amplitude of the backscattered signal. In other words, both can occur simultaneously! Thus, when the reader is sending an ON pulse, the amplitude of the backscattered signal that it receives depends on whether the state of the transistor at the node is ON or OFF — the amplitude is higher when the node's transistor is ON and lower when it is OFF. When the carrier is OFF at the reader, then the state of the node's transistor does not matter since there is no backscattered signal. The reader can detect the remote interrupt by looking for a large signal variance in the carrier wave when the reader has the carrier wave turned on.

Figure 5 shows an example signal where toggling the transistor causes a large variance on the carrier wave, which is monitored by a reader and can be identified by tracking the signal variance within a reader pulse. However, the signal variance is detected only when the carrier wave is on. As shown in the figure, a reader cannot observe the large signal variance when the carrier wave is off. Fortunately, the carrier wave is on for 50% of the time when the reader transmits 0s and 75% for 1s. Thus, as long as a remote interrupt is longer than 50% of the length of a '0' bit from a reader, it can reliably detect the interrupt and pause its transfer.

Finally, an auxiliary benefit of the remote interrupt is that it acts as an inexpensive  $\mu$ frame ACK from the node, which obviates the need for more explicit protocol-level mechanisms and reduces our overhead.

One limitation of our current design is that it is not robust to noise spikes in the frequency band. Such spikes can occur because of multiple readers transmitting to nodes since backscatter is a broadcast medium and reader-to-node communication has to be serialized. Robustness against external interference could be improved by making the remote interrupt longer and encoding the signal, but we do not do this in our current implementation.

**Estimating inter- $\mu$ frame gap** We now have a way for the node to interrupt a reader when it needs to replenish

energy, but how long should the reader wait before initiating the next  $\mu$ frame transfer? Clearly, this duration should be at least as long as the inter- $\mu$ frame gap that the node is using, otherwise the reader might be trying to communicate to a node that has its RF circuit turned off. We address this by using a simple probing-based approach at the reader — for each  $\mu$ frame gap that the reader selects, it knows whether the frame was received or not by checking the presence of a remote interrupt. If no remote interrupt is received, the reader knows the node does not receive the frame properly. The reader continually adjusts the gap to minimize missed frames at the node.

## 4 QuarkNet for multi-node networks

So far, we have focused on communication between a single node and reader. We now turn to the case where there are several nodes in the vicinity of a reader. The key difference between a single node and multi-node setting is that in the former, the reader stays idle during times when the node is asleep to replenish energy, whereas in the latter, these inter- $\mu$ frame intervals present an opportunity to schedule another node's  $\mu$ frame transfer, thereby ensuring that throughput is maximized.

### 4.1 Design Options

Before launching into the details of our design, let's step back and look at the design options. Co-ordination mechanisms for backscatter networks are more restrictive than typical active radio-based networks for two reasons: a) nodes cannot overhear each other's transfer, hence carrier sense-based approaches are infeasible, and b) the stringent resource constraints of nodes render approaches that require complex coding and synchronization infeasible. As a result, existing proposals have focused on two classes of techniques — EPC Gen 2 and variants which use a sequence of random-access slots, and rateless transfer where nodes transfer concurrently, and the reader simultaneously and successively decodes all transmissions.

While the deficiencies of EPC Gen 2 for severely energy constrained regimes have been detailed earlier in this paper, other alternatives and enhancements are surprisingly poor in dealing with this regime as well. In particular, consider two prominent recent techniques — Flit [12] and Buzz [23]. Our earlier work, Flit, re-purposes EPC Gen 2 slots for bulk transfer, thereby amortizing overhead, but it assumes that nodes are able to sustain a long stream of transfer, which we realized was not the case in severe harvesting conditions. Buzz uses rateless codes, but in-order to get these codes to work, it has to use synchronous single-bit slots across nodes. Each

single-bit slot incurs substantial overhead due to slot indicators, and turning on and off the radio, which dramatically impacts performance. Given that existing approaches are not well-suited to our nodes, the question is what protocol to use for co-ordinating nodes.

### 4.2 Variability-aware node scheduling

Our scheduler is designed to interleave  $\mu$ frames from different nodes, thereby fully utilizing the inter- $\mu$ frame gaps. The reader divides time into variable-sized  $\mu$ slots, during which it explicitly schedules a single node to transmit its  $\mu$ frame. The length of each  $\mu$ slot depends on the size of the  $\mu$ frame — a node-to-reader  $\mu$ frame terminates when the node reaches its low watermark energy level and the reader ACK is received, and a reader-to-node  $\mu$ frame terminates when the node issues a remote interrupt. In both cases, there is a maximum bound on the  $\mu$ frame size to deal with nodes that have plentiful energy.

While the  $\mu$ slot mechanism appears relatively straightforward, the main challenge is handling the fact that nodes turn off their RF circuit when they are asleep. As a result, if a node is scheduled too early by the reader, then it may not be awake to utilize the slot, but if it is scheduled too late, then it is not operating at its maximal harvesting rate.

To handle this, we use a token-based scheduler to deal with the stochastic nature of harvesting conditions, while optimizing throughput. For each node, the scheduler maintains a running estimate of the gap between  $\mu$ slots assigned to a specific node, and whether the  $\mu$ slot resulted in a successful transfer. It uses the estimate to select the inter- $\mu$ frame gap that ensures a high likelihood of obtaining a node response.

The reader's estimate of the inter- $\mu$ frame gap is used as input to a token bucket scheduler, which assigns tokens to nodes at a rate inversely proportional to its inter- $\mu$ frame gap. Once a node has accumulated sufficient tokens, it is likely to have woken up after sleep, therefore the reader places the node into a ready queue since it is ready to be scheduled. The ready nodes can be scheduled based on a suitable metric — for example, the highest throughput node may be selected from the queue to maximize throughput, or the node that has received least slots may be selected for fairness.

## 5 Implementation

In this section, we describe key implementation details not covered in earlier sections. We use the USRP reader and UMass Moo for our instantiation of QuarkNet. The source code of QuarkNet is available at [3].



## 5.1 Platforms

**USRP Reader** QuarkNet is built based on the USRP software radio reader developed by Buettner [9] with a ANT-NA-2CO antenna [4]. We modify the signal processing pipeline to enable variable sized  $\mu$ frame decoding, harvesting-aware tag scheduling, and detection of in-band remote interrupts. The RFX900 USRP RF daughterboard on our platform is only able to transmit 200mW of power, which is  $5\times$  smaller than the 1W of power issued by a commercial reader. Therefore, we attach a  $3\text{cm}\times 3\text{cm}$  solar panel to each Moo to increase the amount of harvested energy. The use of hybrid power (RF + ambient) is known to increase range from a reader, which enhances the regimes where backscatter can be used [11].

**Backscatter node** The UMass Moo is a passive computational RFID that operates in the  $902\text{MHz} \sim 928\text{MHz}$  band. Perhaps the most challenging aspect of our implementation is debugging under extreme low energy conditions. Traditional methods for debugging embedded systems, such as using JTAG, supply power to the node and change its behavior. Instead, we instrument the Moo to toggle GPIO pins at key points during its execution, and a logic analyzer to record the toggle events. In many cases, however, it is difficult to insert sufficient instrumentation to have visibility while still working with tiny energy harvesting levels. Thus, intuition and experience is particularly important in designing systems for these regimes.

## 5.2 Trimming Overheads

One important aspect of our system is careful measurement and tuning of all overheads, which impacts our ability to scale-down to severe harvesting conditions.

**Radio transition overhead:** An important source of overhead is transition times for turning on or off the radio. Fortunately, since hardware timers are responsible for generating the pulses on the backscatter radio, sleep gaps can be inserted by clearing the hardware timers and turning the micro-controller into its low power mode. These operations are inexpensive energy-wise, and consume roughly the same amount of energy as a data frame of size 3 bits. Note that this observation does not hold for more complex active radios — for example, a WiFi radio takes 79.1ms to be on, and 238.1ms to be turned off [13], which is five orders of magnitude higher than the corresponding numbers for a backscatter radio.

**Pilot tone:** Each backscatter frame can potentially include a pilot tone in addition to the payload. A pilot tone is used when a tag changes its baud rate [19]. We focus on a minimalist protocol that uses a fixed baud rate,

therefore we remove the pilot tone. The total overhead per  $\mu$ frame is 6 bits of preamble, in contrast to the 22 bits overhead of EPC Gen 2 (and variants such as Flit [12]).

**Probing energy state:** As mentioned earlier, analog-to-digital conversions are expensive, and should be avoided while tracking the maximum energy harvesting rate. Our key insight is that rather than measure the voltage on the node, we can leverage the existing low watermark threshold detector that is already present on such nodes. Such a detector is common on harvesting-based sensor platforms for two reasons: a) the platform needs to know when to save state and go to sleep to avoid an outage, and b) the platform needs to know when to wake up after sleep to continue operation. Thus, QuarkNet gets an interrupt both when the voltage crosses above the threshold, as well as when it drops below the threshold, and uses this information as a one-bit proxy for the actual voltage. The voltage threshold is chosen to be 2V which is slightly higher than 1.8V, the minimum voltage required for operating a micro controller. This information is input to a sleep time tracker, which determines how long to wait after crossing the threshold in the upward direction before initiating transfer. Our approach is  $100\times$  less expensive energy-wise than an ADC conversion.

## 5.3 Protocols and Algorithms

While we do not describe the complete protocol in the interest of space, more details as well as pseudocode for our algorithms can be found in our technical report [28].

## 6 Evaluation

The evaluation consists of three parts: 1) demonstrating the range and throughput benefit of  $\mu$ frame transmission, 2) benchmarking the performance of our reader-to-node communication, and 3) evaluating the benefit of interleaving  $\mu$ frames from multiple nodes.

### 6.1 Benefit of $\mu$ frames

In this section, we validate our claim that the ability to breakdown packets into  $\mu$ frames that can be as small as a single bit can allow us to operate under lower energy conditions and achieve higher operating range. To focus on the effect of the choice of frame size, we strip off overheads (slot indicators, handshakes, etc) for all protocols that we compare.

**Minimum operating conditions** We look at two harvesters — RF and solar — and ask what is the minimum power requirements for different approaches. We find

that the minimum illuminance required for a 1 bit  $\mu$ frame is 150 lux, which is  $13\times$  lower than the 2000 lux budget of 12 byte packet transmission (the same packet size used by EPC Gen 2, Dewdrop, Flit, etc). We choose 12 byte packet size for EPC Gen 2-based protocols because the 12 byte EPC identifier needs to be transmitted in a singulation phase prior to executing Read or Write commands. Thus, this packet is the bottleneck for operation. To translate from lux to the typical energy available from indoor energy sources, we measure the natural indoor illuminance in 30 positions in an office room. We find that 92% the measured illuminance value is between 150 lux and 1000 lux. This suggests that  $\mu$ frames can operate in most of natural indoor illuminance conditions while a canonical 12 byte transfer scheme can almost never operate under natural indoor light.

The minimum RF power required for a 1 bit  $\mu$ frame is 13dBm, which is  $20\times$  smaller than the 26dBm budget of a 12 byte packet transmission that is the minimum needed for EPC Gen 2 and its variants to operate. Both experiments illustrate the benefits of using tiny  $\mu$ frames.

**Increased operational range** Our second claim is that we can improve operational range by using  $\mu$ frames. Figure 6 shows the maximum range that is achieved by QuarkNet with 1 bit  $\mu$ frames, EPC Gen 2 with fixed 12 byte packets, Dewdrop with fixed 12 byte packets, Buzz with two slot choices, and a battery-assisted node which represents the best-case scenario. We adjust the RF power of the USRP RFID reader from 17dBm to 25.7dBm, which represents the range of RF power that can be generated by the USRP RFX900 daughterboard.

The results show that the communication range of QuarkNet is longer than other schemes across all RF power levels. At the lowest power level (17.5dBm),  $\mu$ frames do not improve range since the node is not able to decode the reader signal beyond 5ft. But as the RF level increases, the operational range increases dramatically, and is about  $4\times$  longer than EPC Gen 2 at the highest power. In fact, the performance of 1 bit  $\mu$ frame transfer while using harvested energy almost matches the performance of a battery-assisted node, which shows that we are able to reach the ceiling of operational range despite operating on micro-power.

Figure 6 also shows that Buzz [23] performs poorly compared to other schemes. This can be attributed to the fact that each one-bit slot in Buzz has substantial overhead — the reader sends a pulse, followed by one bit from the node, random number generation for deciding whether to transfer in the next slot, and a recharge period. Thus, while Buzz has high range in some settings, the overhead is too high to scale gracefully.

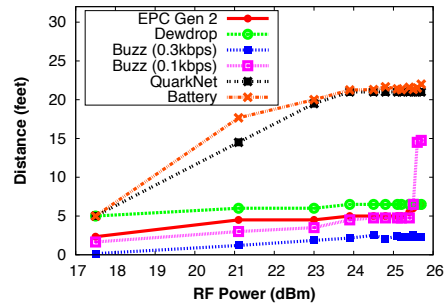


Figure 6: The maximum range achieved by EPC Gen 2, Dewdrop, Buzz, QuarkNet, and a battery assisted node. QuarkNet operates at ranges close to the battery assisted node.

## 6.2 Benefits of $\mu$ frame adaptation

We now turn to the benefits of adapting the inter- $\mu$ frame gap to maximize throughput.

**Convergence of gradient descent** How well does the gradient-descent algorithm learn the optimal harvesting rate? Figure 7 shows the results for a node placed in three RF+light harvesting combinations that include short and medium range, and low and medium light. In all cases, we see convergence to close to the optimal point — the best inter- $\mu$ frame gap ranges from 0ms for 350lux at 1 foot since there is enough power to continuously operate the node, 4ms when the node is moved to 6ft, to 12ms when the light conditions dip further. In all cases, our tracking algorithm converges in very few steps ( $\leq 4$ ).

**Throughput benefits** We now know that QuarkNet picks close to the optimal harvesting rate, but what are the benefits in terms of throughput? To understand this, we place a node 3 feet from a reader, vary RF power from 17dBm to 26dBm in small steps of 0.3dBm, and inventory the node 2000 times for each scheme. Figure 10(a) shows the throughput achieved by EPC Gen 2, Dewdrop, Flit, and QuarkNet. We find that the throughput achieved by QuarkNet is higher than EPC Gen 2, Dewdrop and Flit across all RF power levels. The average communication throughput of QuarkNet is 18kbps,  $10.5\times$  higher than EPC Gen 2,  $5.8\times$  higher than Dewdrop, and  $3.3\times$  higher than Flit. While the figure does not show Buzz's throughput, note that Figure 6 already showed that this number is low since the per-slot overhead dominates. The lowest slot size we achieved in our implementation of Buzz is 3ms, which means about 0.3kbps throughput.

The previous experiments were done by varying the RF power level. To be sure that these results translate to the case where nodes are placed at different locations in front of a reader, we measure the throughput achieved by EPC Gen 2, Dewdrop, Flit, and QuarkNet at 30 different randomly chosen locations between 2 to 13 ft in front of a reader. Figure 8 shows that the throughput achieved by

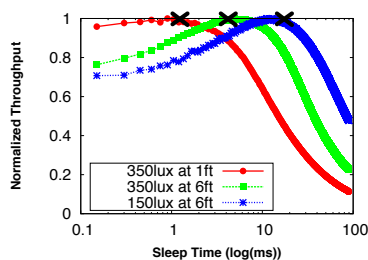


Figure 7: Throughput achieved for different sleep times (inter- $\mu$ frame gaps). The sleep time chosen by QuarkNet is within 98% of the optimal.

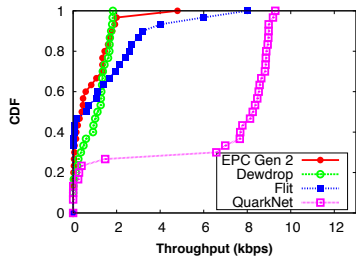


Figure 8: Throughput achieved by EPC Gen 2, Dewdrop, Flit, and QuarkNet across 30 locations. QuarkNet has at least  $4.4\times$  higher throughput than other schemes.

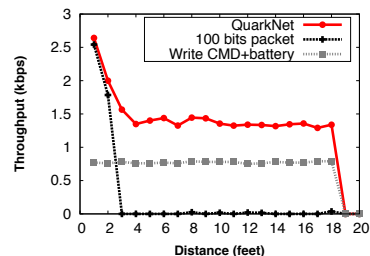
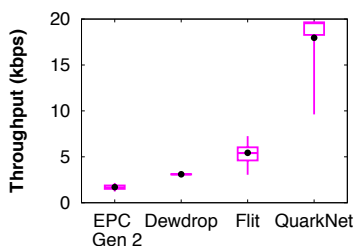
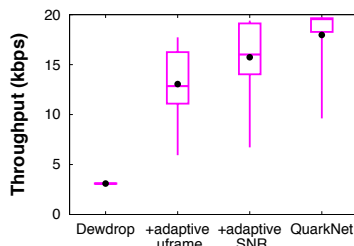


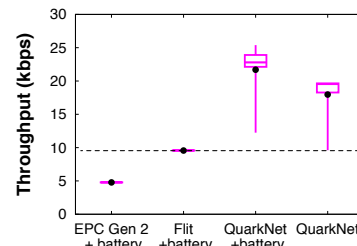
Figure 9: Throughput of reader-to-node communication. QuarkNet has  $2\times$  higher throughput than battery-assisted EPC Gen 2 Writes.



(a) QuarkNet vs micro-powered nodes.



(b) QuarkNet vs Dewdrop.



(c) QuarkNet vs battery assisted nodes.

Figure 10: For micro-powered devices, QuarkNet improves throughput by at least  $3.3\times$  over all other schemes, and even performs better than battery assisted nodes. The benefit comes from reducing overhead, and adapting  $\mu$ frame sizes to energy and SNR.

QuarkNet is higher than the other three schemes across all locations. The average throughput of QuarkNet is  $7.8\times$  higher than EPC Gen 2,  $6.4\times$  higher than Dewdrop, and  $4.4\times$  higher than Flit. In particular, QuarkNet continues to operate in many locations where other schemes cease to operate.

**Breaking down the benefits** QuarkNet has a variety of optimizations including reduced overheads, variable-sized  $\mu$ frames, and SNR adaptation. To understand the contributions of these techniques to throughput, we start with the default implementation of Dewdrop, and add one optimization at a time: a) Dewdrop + adaptive frame, which includes variable-length  $\mu$ frames, and b) Dewdrop + SNR adaptation which includes the SNR adaptation. Figure 10(b) shows the throughput achieved by the three variants of Dewdrop vs QuarkNet. Clearly, each of the optimizations plays a major role in the throughput improvements observed by QuarkNet. The average communication throughput of  $\mu$ frame is 18kbps,  $5.79\times$  higher than Dewdrop,  $1.37\times$  higher than Dewdrop with adaptive  $\mu$ frames, and  $1.14\times$  higher than the case when SNR adaptation is included. In the final step, we replace Dewdrop's adaptation algorithm with our version that eliminates ADC conversions to get QuarkNet.

**QuarkNet vs battery-assisted alternatives** Another interesting question is how QuarkNet performs when

compared to battery-assisted versions of the other protocols (excluding Dewdrop + battery, which is identical to EPC Gen 2 + battery). Some protocols, such as Flit [12], improve in performance when there is more energy since there is more opportunity for bulk transfer. Would these outperform QuarkNet in battery-assisted scenarios? Figure 10(c) shows that throughput achieved by QuarkNet is consistently better. The average throughput of QuarkNet is 18kbps,  $3.75\times$  higher than EPC Gen 2 + battery, and  $1.87\times$  higher than Flit + battery. This result shows the benefit of reducing per-frame overheads in QuarkNet.

### 6.3 Reader-to-node communication

We now turn to an evaluation of reader-to-node communication. We begin by looking at the effectiveness of remote interrupts. We find that remote interrupts are extremely reliable — the reader detects remote interrupts with 100% accuracy across all distances where the node can communicate with the reader, and detection rate directly drops to 0% at roughly 19 – 20 feet where the node cannot detect the signal sent by the reader. While the accuracy will degrade under external interference, we plan to extend remote interruption to include encoded bits to improve robustness.

Next, we look at the throughput of reader-to-node

communication when a node is placed at different distances from the reader. Figure 9 shows that the throughput achieved by QuarkNet is always higher than fixed 100 bit transfer across all distances. (We chose 100 bits instead of 12 bytes because of the slower baud rate of the reader-to-node link, as a result of which 12 byte transfer ceases to operate even when the node is deployed 1 feet from the reader.) The throughput of QuarkNet is higher than even a battery-assisted EPC Gen 2 node. This shows that the benefit of variable sized  $\mu$ frames is substantial even for reader-to-node communication.

One trend in the graph that requires a bit more explanation is the fact that throughput decreases rapidly when the node is close to the reader (less than 4 feet), and plateaus until about 18 ft after which it quickly drops to zero. This is because RF-harvesting only works until 4ft (because of the limitations of the USRP reader), and beyond this distance, indoor light harvesting plays the dominant role.

## 6.4 Evaluating the QuarkNet MAC layer

We now turn to the evaluation of our MAC layer that includes all components of the protocol including various co-ordination overheads, frame interleaving, and scheduling. Figure 11 shows the communication throughput when we deploy 10 nodes in front of the reader and adjust the RF power from 17dBm to 26dBm. We use a throughput-maximizing scheduling policy in this experiment. For each RF power level, we plot the averaged throughput across the ten nodes and the confidence interval when they are scheduled in an interleaved manner and when they are inventoried individually. The throughput achieved by other MAC layer designs — EPC Gen 2 and Flit — are close to zero, so we do not plot them.

We find that even at the lowest RF power level, almost all nodes get to transmit data to the reader, and the average throughput steadily increases with higher RF power. In addition, the throughput achieved by interleaving the 10 nodes is 5.4 $\times$  higher than the throughput when those 10 nodes are inventoried individually. These results show that our algorithm scales well across a wide dynamic range of harvesting conditions, and uses gaps between  $\mu$ frames efficiently.

## 6.5 Microbenchmarks

Table 2 shows the overhead incurred by different components of QuarkNet. The biggest system overhead is the switch from inactive mode to transmission mode (47.5 us), to configure several registers associated with transmission, such as the hardware timer register and data register. The overhead of the entire  $\mu$ frame size and inter-

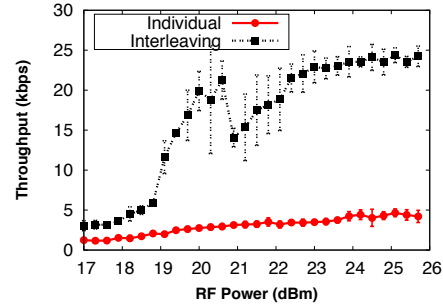


Figure 11: Throughput of 10 nodes is 5.4 $\times$  higher when interleaved than when individually inventoried.

Table 2: Overhead of  $\mu$ frame transmission.

System overhead (us)		$\mu$ frame overhead (us)	
TX to inactive	9.9	interrupt config	10.58
inactive to TX	47.5	handle interrupt	9.3
RX to TX	4.08	$\mu$ frame adaptation	24.3
sleep to wakeup	9.83	voltage detection	3

$\mu$ frame gap adaptation algorithm (47.2us), is comparable to the total system overhead, and 10 $\times$  smaller than the cost of an ADC conversion. Overall, the results show that our performance tuning measures have substantial benefits — the sum total of these overheads is smaller than the cost of transmitting 7 bits.

## 7 Discussion

**Interoperability with other PHY mechanisms** While our work does not explicitly address co-existence of QuarkNet with other physical layer and upper layer mechanisms, many of these can be easily layered above the methods described in this paper. For example, rate adaptation is widely used to adapt to wireless channel conditions, thereby maximizing communication throughput. This method operates at the bit-level, where each bit is composed of several symbols. Such an approach can be layered above QuarkNet, with gaps introduced between bits. Similarly, error correction codes or other encoding mechanisms that reduce bit error rate can be implemented above QuarkNet.

**QuarkNets role with evolving technology** As micro-harvesters continue to improve in efficiency, one question is whether QuarkNet will continue to remain relevant. We argue that QuarkNet’s relevance will increase for two reasons. First, the maximum harvesting rates are fundamentally limited by the physics of the harvesting source and form-factor. For example, RF energy harvesting is limited by the antenna size and the amount power issued by antennas, solar energy harvesting is limited by the panel size and the intensity of illuminance, and thermal energy harvesting is limited by the surface area and

the temperature differential. Even if micro-harvesters become extremely efficient (say upwards of 80%), there is still a small amount of energy available, and systems optimizations similar to QuarkNet are critical to using the energy in an efficient manner. Second, trends in nano-electronics and low-power embedded systems are resulting in sensing and computing platforms that consume only tens of micro-watts of power [1]. These trends will make it possible to design many more micro-power based applications such as implantables and on-body sensors, enhancing the relevance of QuarkNet.

**Fragmenting other tasks** While our focus in this paper is on fragmenting the network stack, the abstraction of task fragmentation presented by QuarkNet can be potentially used for breaking down other components of a task such as sensing and computation into smaller atomic units. In our position paper [29], we presented preliminary results that demonstrated the ability to fragment an image sensing task such that the entire sensor can operate with a 3cm×3cm solar panel under natural indoor illuminance. However, many questions remain to fully enable such fragmentation, requiring a combination of architectural modifications to the sensing and computing blocks to facilitate fine-grained fragmentation, systems techniques similar to QuarkNet that can take advantage of the fragmentation capability, as well as data processing techniques to enable useful applications over a layer that dynamically fragments sensing tasks.

## 8 Related Work

We have already discussed Dewdrop, Flit, Buzz, and EPC Gen 2, so we focus on other approaches.

**Computational RFIDs (CRFIDs)** There has been increasing emphasis on CRFIDs in recent years given its potential for battery-less perpetual sensing. Ambient Backscatter [16] uses the backscatter of FM signals for short-range communication between tags. This is a severely energy limited platform, and could leverage QuarkNet when harvested energy is low. BLINK [30] is a bit-rate and channel adaptation protocol to maximize communication throughput, which can also leverage QuarkNet for performance. [20] introduces a power-optimized waveform which is a new type of multiple-tone carrier and modulation scheme that is designed to improve the read range and power efficiency of charge pump-based passive RFIDs. [21] presents a system architecture for backscatter communication which reaches 100m communication distance at the cost of slow bit rate (10 bits per second). Such techniques are complementary to QuarkNet — each bit transmitted at slow bit rate can be fragmented into several segments where the information within each bit is still preserved. Also of note

is MementOS [18], which uses non-volatile flash storage for checkpoints within a task such that it can continue execution after an outage. Flash checkpointing is useful for outage tolerance but is more than the cost of transmitting an entire EPC Gen 2 packet, hence it has limited utility in our case.

**EPC Gen 2 optimizations** Much of the work on backscatter communication is specific to EPC Gen 2 tags, for example, better tag density estimation [22], better search protocols to reduce inventorying time [14], better tag collision avoidance [17], more accurate tag identification [26], better recovery from tag collisions [6], and more efficient bit-rate adaptation [30]. None of these tackle the problem of maximizing range and throughput from RFID-scale sensors, which have the ability to offload sensing data back to a reader.

EPC Gen 2 supports tag user memory operations in addition to simple EPC queries including the Read and Write command, however they are second-class citizens in the protocol since the main goal is to inventory tags. As a result, both are inefficient primitives for data transfer from tag to reader or vice-versa. In our experiments, we found that the Read and Write commands simply do not work at all under low energy conditions.

## 9 Conclusion

In this paper, we present a powerful network stack, QuarkNet, that can enable systems to seamlessly scale down to severe harvesting conditions as well as substantial harvesting dynamics. At the core, our approach deconstructs every packet into  $\mu$ frames, handles dynamics with variable-sized  $\mu$ frames, and maximizes throughput via low-cost adaptation algorithms and interleaving of  $\mu$ frames. Results show that QuarkNet provides substantial benefits in pushing the limits of micro-powered devices, and allow them to perform useful work under more extreme environments than previously imagined possible. Our network stack tolerates such conditions, thus makes it valuable to a wide range of emerging micro-powered embedded systems and applications.

## Acknowledgements

This research was partially funded by NSF grants CNS-1218586, CNS-1217606, and CNS-1239341. We thank our shepherd Shyam Gollakota for providing guidance in preparing our final draft and the anonymous reviewers for their insightful comments. We also thank Michael Buettner for extensive and insightful comments, Jeremy Gummeson for help in Flit implementation, and Jue Wang for assistance with Buzz.

## References

- [1] Advanced Self-Powered Systems of Integrated Sensors and Technologies. <http://assist.ncsu.edu>.
- [2] Gradient descent. [http://en.wikipedia.org/wiki/Gradient\\_descent](http://en.wikipedia.org/wiki/Gradient_descent).
- [3] Source code of QuarkNet. <https://github.com/pengyuzhang/QuarkNet>.
- [4] ThingMagic Bistatic Antenna. <http://buyrfid.com/>.
- [5] WISP: Wireless Identification and Sensing Platform. <http://seattle.intel-research.net/wisp/>.
- [6] C. Angerer, R. Langwieser, and M. Rupp. Rfid reader receivers for physical layer collision recovery. *IEEE Transactions on Communications*, 2010.
- [7] S. Bandyopadhyay and A. Chandrakasan. Platform architecture for solar, thermal and vibration energy combining with mppt and single inductor. In *VLSI Circuits (VLSIC), 2011 Symposium on*, pages 238–239. IEEE, 2011.
- [8] M. Buettner, B. Greenstein, and D. Wetherall. Dewdrop: an energy-aware runtime for computational rfid. In *USENIX NSDI*, 2011.
- [9] M. Buettner and D. Wetherall. A software radio-based uhf rfid reader for phy/mac experimentation. In *RFID (RFID), 2011 IEEE International Conference on*, pages 134–141. IEEE, 2011.
- [10] M. Gorlatova, P. Kinget, I. Kymissis, D. Rubenstein, X. Wang, and G. Zussman. Challenge: ultra-low-power energy-harvesting active networked tags (enhants). In *Proceedings of the 15th annual international conference on Mobile computing and networking*, pages 253–260. ACM, 2009.
- [11] J. Gummesson, S. S. Clark, K. Fu, and D. Ganesan. On the limits of effective hybrid micro-energy harvesting on mobile crfid sensors. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, pages 195–208. ACM, 2010.
- [12] J. Gummesson, P. Zhang, and D. Ganesan. Flit: A bulk transmission protocol for rfid-scale sensors. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 71–84. ACM, 2012.
- [13] J. Huang, F. Qian, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck. A close examination of performance and power characteristics of 4g lte networks. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 225–238. ACM, 2012.
- [14] C. Law, K. Lee, and K. Siu. Efficient memoryless protocol for tag identification. In *Proceedings of the 4th international workshop on Discrete algorithms and methods for mobile computing and communications*, pages 75–84. ACM, 2000.
- [15] Y. Lee, G. Kim, S. Bang, Y. Kim, I. Lee, P. Dutta, D. Sylvester, and D. Blaauw. A modular 1mm<sup>2</sup> 3 $\times$ 3 die-stacked sensing platform with optical communication and multi-modal energy harvesting. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, pages 402–404. IEEE, 2012.
- [16] V. Liu, A. Parks, V. Talla, S. Gollakota, D. Wetherall, and J. R. Smith. Ambient backscatter: wireless communication out of thin air. In *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*, pages 39–50. ACM, 2013.
- [17] V. Namboodiri and L. Gao. Energy-aware tag anticollision protocols for rfid systems. *Mobile Computing, IEEE Transactions on*, 9(1):44–59, 2010.
- [18] B. Ransford, J. Sorber, and K. Fu. Mementos: System support for long-running computation on rfid-scale devices. *ACM SIGPLAN Notices*, 47(4):159–170, 2012.
- [19] K. E. Sundstrom, S. A. Cooper, A. Sarajedini, A. Esterberg, T. E. Humes, and C. J. Diorio. Rfid reader systems detecting pilot tone, Jan. 8 2013. US Patent 8,350,665.
- [20] M. S. Trotter, J. D. Griffin, and G. D. Durgin. Power-optimized waveforms for improving the range and reliability of rfid systems. In *RFID, 2009 IEEE International Conference on*, pages 80–87. IEEE, 2009.
- [21] G. Vannucci, A. Bletsas, and D. Leigh. A software-defined radio system for backscatter sensor networks. *Wireless Communications, IEEE Transactions on*, 7(6):2170–2179, 2008.
- [22] H. Vogt. Efficient object identification with passive rfid tags. *Pervasive Computing*, pages 98–113, 2002.
- [23] J. Wang, H. Hassanieh, D. Katabi, and P. Indyk. Efficient and reliable low-power backscatter networks. *ACM SIGCOMM Computer Communication Review*, 42(4):61–72, 2012.
- [24] A. Yakovlev, S. Kim, and A. Poon. Implantable biomedical devices: wireless powering and communication. *Communications Magazine, IEEE*, 50(4):152–159, 2012.
- [25] D. Yeager, J. Holleman, R. Prasad, J. Smith, and B. Otis. Neuralwisp: A wirelessly powered neural interface with 1-m range. *Biomedical Circuits and Systems, IEEE Transactions on*, 3(6):379–387, 2009.
- [26] D. Zanetti et al. Physical-layer identification of uhf rfid tags. In *ACM MobiCom*, 2010.
- [27] H. Zhang, J. Gummesson, B. Ransford, and K. Fu. Moo: A batteryless computational rfid and sensing platform. Technical report, Tech. Rep. UM-CS-2011-020, UMass Amherst Department of Computer Science, 2011.
- [28] P. Zhang and D. Ganesan. Enabling bit-by-bit backscatter communication in severe energy harvesting environments. Technical report, School of Computer Science, UMass Amherst. URL: <http://cs.umass.edu/~pyzhang/papers/QuarkNet-tech-report.pdf>, 2014.
- [29] P. Zhang, D. Ganesan, and B. Lu. Quarkos: Pushing the operating limits of micro-powered sensors. In *Proceedings of the 14th USENIX conference on Hot Topics in Operating Systems*, pages 7–7. USENIX Association, 2013.
- [30] P. Zhang, J. Gummesson, and D. Ganesan. Blink: A high throughput link layer for backscatter communication. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 99–112. ACM, 2012.