RF-Powered, Backscatter-Based Cameras

Saman Naderiparizi[1], Zerina Kapetanovic[2], Joshua R. Smith[3]
[1]Electrical Engineering, University of Washington, Seattle, USA, samannp@uw.edu
[2]Electrical Engineering, University of Washington, Seattle, USA, zerinak@uw.edu
[3]Computer Science and Engineering and Electrical Engineering, University of Washington, Seattle, USA, jrs@cs.washington.edu

***Abstract*—RF-powered devices equipped with general-purpose microcontrollers face energy limitation constraints for performing arbitrarily complex sensing and computation tasks. While richer capabilities such as image capture and processing would enable many new RF-powered use-cases, this energy limitation narrows the application space. Enabling richer sensing tasks has two main challenges: efficiently retaining and utilizing harvested energy, and storing and communicating large quantity of sensor data. This paper reviews the WISPCam design, an RF-powered programmable camera. Taking into account both of the mentioned challenges, WISPCam integrates an off-the-shelf VGA camera which is a rich sensor example energy and data wise. The paper also presents an ultra-low power scheme that is able to provide periodic updates on charge state of the device before enough energy has been accumulated for the camera to capture an image. This paper presents a novel data storage and bi-directional communication scheme that enables reliable transfer of complete images to an RFID reader application even when packets are lost or the device runs out of energy.**

I. Introduction

Recently, reserchers have shown the feasibility of wireless sensor nodes that can harvest their required energy from propagating radio waves. The first form of battery-free sensing systems targeted simple applications, such as sensing temperature or light intensity [1], more popular phenomena like neural signals [2], and sensing specific motions [3]. In recent years, RFID based comercial sensing systems have become available, targeting a wide range of applications from strain monitoring to pick-to-light systems, and even tire pressure monitoring for aircrafts [4], [5]. In addition to RFID, powering some sensors with WiFi signals have been demonstrated recently [6]. In our recent work, we built and presented the *WISPCam* [7], what we believe to be the world's first completely wireless and battery-free camera.

In comparison to other sensors, cameras are rich in information content, however they come with their own tradeoffs. Looking back a previous battery-free work, focused on low-power and low-datarate sensors, cameras are significantly more power hungry and require higher data rates. For instance, a temperature sensor will burn power in the oder of microJoules to provide a few bytes of data, whereas a camera will consume tens of milli-Joules of energy to capture an image worth of kilobytes of data [7].

Recently the capabilities of *WISPCam* have been extended from image-capture only devices to platforms that can be used for richer and more smart scenarios. A network of battery-free cameras that can localize themselves [8] and off-loading computationally demanding tasks to the RFID reader to enable face detection/recognition tasks [9] are two examples of this advancements.

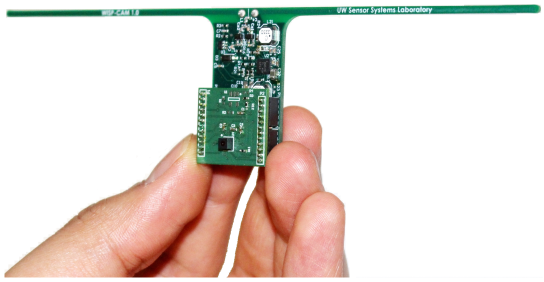


Fig. 1: The latest *WISPCam* prototype

In this paper we review the high-level design aspects of the *WISPCam* and present two of our latest accomplishments in regards to advancing battery-free cameras:

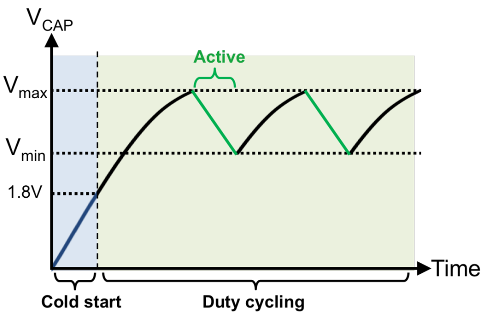
- Leveraging two-way communication capabilities of EPC (Electronic Product Code) Generation 2 Class 1 protocol to build an infrastructure to retrieve missing pixels during image transmit.
- Implementing ultra-low power circuitry on *WISPCam* to provide periodic stored energy status report to the RFID reader with negligible energy draw overhead from the *WISPCam*.

II. WISPCam Design

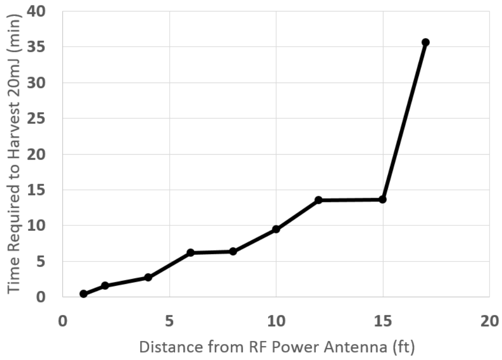
The *WISPCam* can be broken down into four key components, shown in Figure 3, which includes a power harvester to store enough energy on a supercapacitor, a receive demodulator to decode incoming data, a microcontroller to handle firmware and interfacing with components, and most importantly and off-the-shelf VGA camera.

Similar to any energy scavenging device, using a voltage regulator, the power harvester on the *WISPCam* gathers energy on its charge reservoir, in this case a supercapacitor, until it reaches and upper threshold voltage V_{max}. During the charging period, the supply voltage to the entire system is disabled to preserve energy. When the V_{max} threshold is reached, the *WISPCam* wakes up and remains active until the charge reservoir discharges to a V_{min} threshold. This duty cycling process is shown in Figure 2a. It is important to note that V_{max} and V_{min} can be calculated from the equation below in accordance to the load energy requirement E_{load}:

$$E_{load} = \frac{1}{2}C(V_{max}^2 - V_{min}^2) \tag{1}$$



(a) Inherent duty-cycling



(b) Update rate vs distance

Fig. 2: 2b*WISPCam* inter frame time versus distance from the RFID reader and 2aduty-cycling nature of the energy scavenging devices

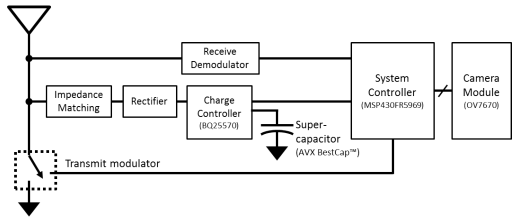
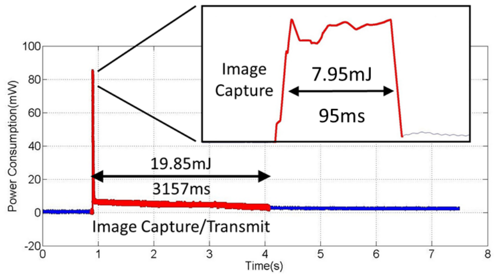


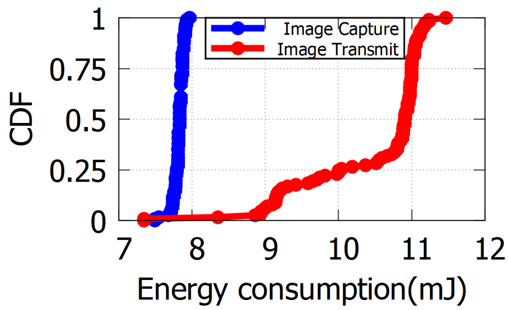
Fig. 3: High-level block diagram of *WISPCam* design

The duty-cycling frequency, which is the *WISPCam*'s update rate, dependent on the available power. To elaborate, it depends on the distance of the *WISPCam* from the RFID reader. As the *WISPCam* gets further away from the reader, it receives less power, thus its inter-frame time will increase. Figure 2b shows the time required to harvest a fixed amount of energy (20mJ in our case) versus distance between the *WISPCam* and RFID reader.

Figure 4a illustrates the power consumption of the *WISPCam* from waking up until the end of image transmission. The *WISPCam* will begin by capturing and initially storing an image on the on-board low-power non-volatile FRAM memory, during this process the *WISPCam* consumes a significant amount of power, as high as 80mW. Since we are more concerned about the consumed energy instead of power, it is very important to keep this period as short as possible. Then *WISPCam* will detach its camera from the power supply to avoid any possible extra energy leakage. Then the *WISPCam* starts transmitting image data while burning less than 5mW of power.



(a) *WISPCam* power profile



(b) Energy variability of image capture/transmit

Fig. 4: 4a Power consumption of WISPCam when doing a full round of image capture and transmit. 4b CDF of energy consumption for image capture and transmit, showing significant energy variability for image transmit phase

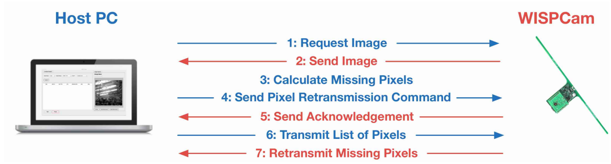


Fig. 5: Protocol overview to perform pixel retransmission with the *WISPCam*

Figure 4a shows a sample power consumption of the *WISPCam*, meaning the *WISPCam* may have varied total energy consumption from one image capture and transmission to the other. We collected the energy consumption data for over 100 image captures and transmissions to generate the CDF (Cumulative Distribution Function), shown in Figure 4b. It is obvious that the image transmission phase has significant variability due to the huge variation in the wireless communication channel. Since the total energy provided to the *WISPCam* at each round is fixed (set by the voltage regulator), it is very possible for the *WISPCam* to run out of power before finishing one complete round of image capture and transmission. To overcome this problem, we leverage the on-board nonvolatile memory, allowing the *WISPCam* to pick up from where it left off each time it gains power again.

III. Missing Pixels Re-transmission

A hindrance that comes with the *WISPCam* is excessive packet loss, which poses an issue for applications like surveillance or face detection due to distorted images. Building on top of Wisent platform [10] and leveraging EPC Generation 2 Class 1 downstream capability, we overcome this challenge by implementing a simple protocol to requests the retransmission of missing pixels from the *WISPCam*. Figure 5, demonstrates a high level view of the protocol. The process is started by a user, who sends a command to the *WISPCam* to transmit an image. Once the image is received, the user decides if retransmission is required, if so, a command is sent to the *WISPCam* and the missing pixels are transmitted. The *WISPCam* will receive a list of missing pixels and retransmit to update the image. This process is continued until the user is satisfied with the image quality.

An example of pixel retransmission is shown in Figure 6, where you can see the original image, and three rounds of pixel retransmission, where it is visible that the image is improving with each round. The initial image had 240 words of missing data and after the first retransmission this decreased to 120 words. The second retransmission brought down the data loss to 30 words and the final retransmission recovered the image fully.

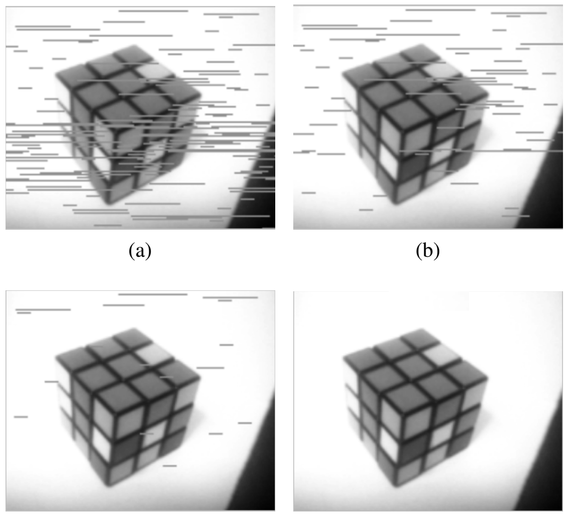
We were able to implement half-duplex communication with the *WISPCam* by leveraging the EPC Generation 2 protocol. To elaborate, we used SLLURP, a low-level reader protocol, that allows us to send write commands to the *WISPCam* and the *WISPCam* would acknowledge by modifying its EPC. This protocol does come with the limitation of only being able write 150 words of data, which is limiting when requesting the retransmission of pixels. To work around this, we came up with a simple idea to packetize the missing pixels. Instead of sending each pixel value, we alternatively find the first missing pixel and count how many consecutive missing pixels come after, allowing one word to represent a bulk of data. This concept is then continued throughout the search until all missing pixels have been identified.

IV. Interactive WISPCam

The two-way communication between *WISPCam* and the RFID reader is discussed in previous section. Taking advantage of this feature without awareness of the available energy on the *WISPCam* is almost impossible. Imagine a scenario where the RFID reader wants to command *WISPCam* to perform an operation. The failure or success of this operation is highly dependent on either *WISPCam* has enough energy to perform the task or not. So providing RFID reader with some feedback about available energy on the *WISPCam* is an important capability. Here we talk about how to implement a periodic ultra-low power wake up mechanism for *WISPCam* to provide the reader with some status information.

A. Ultra-low Power Energy Level Feedback

A trivial way to wake-up the *WISPCam* periodically is to put the microcontroller in deep sleep mode with the timer enabled. In this mode the microcontroller will burn about 1μW of continuous power [11]. Then microcontroller can be woken up after arbitrary time intervals. Additionally, a voltage regulator also needs to be active all the time since microcontrollers desire almost constant voltage to operate properly and the lowest power regulators will burn about



(a) (b) (c) (d)

Fig. 6: The *WISPCam* captures an image and sends it to the host PC shown in in Figure 6a. The image was distorted so the host PC requests a set of missing pixels and the *WISPCam* transmits back as shown in Figure 6b. Pixels are requested two more times to get a clearer images, as shown in Figures 6c and 6d.

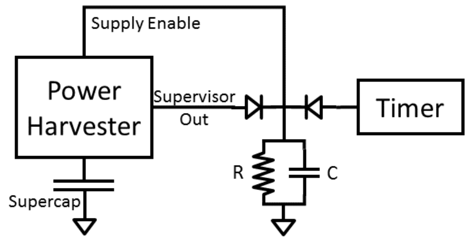


Fig. 7: Ultra-low power periodic wake up circuitry

1.5μW at a 3V supply voltage [12]. So using this method our system will burn about 2.5μW of continuous power. This few extra micro-watts of power will limit the minimum operating sensitivity of the system. Assume $P_{harvest}$ to be the harvested power of the *WISPCam* and P_{static} to be the static power draw due to the wake up mechanism, then the harvesting efficiency of the *WISPCam* will be degraded by a factor of η_{wakeup}:

$$\eta_{wakeup} = \frac{P_{harvest} - P_{static}}{P_{harvest}} \quad (2)$$

Based on Eq 2 if $P_{static} << P_{harvest}$ then $\eta_{wakeup} \approx 1$, consequently the total efficiency degradation is negligible. However, in energy scavenging systems since we are more concerned about the performance of the power harvester at its sensitivity (this is something that defines the operation distance range), $P_{harvest}$ will be in the order of few micro-watts when the *WISPCam* is far from the RFID reader (about 10 meters). So the total efficiency can be significantly degraded due to the 2.5μW of P_{static}.

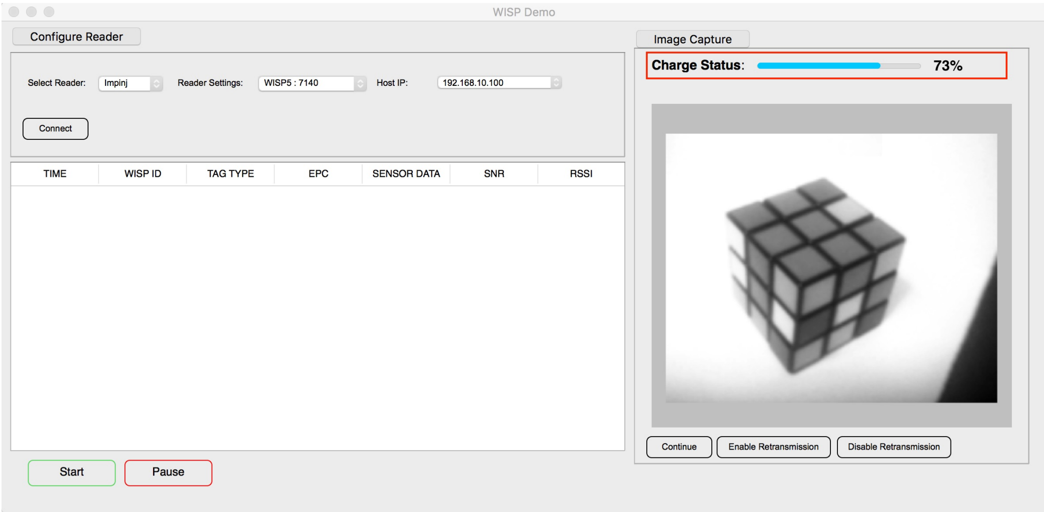


Fig. 8: Software interface developed to interact with the *WISPCam*

What we suggest instead, is to use an ultra-low power external timer that can wake up the microcontroller periodically while burning nano-watts of power. The implemented circuit is shown in Figure 7. *Supervisor Out* is the voltage supervisor output that activates the supply voltage to the rest of the circuit when the supercapacitor voltage reaches V_{max}. The two diodes are implementing a passive **OR** logic and will activate the supply voltage to the rest of the *WISPCam* when either the timer or voltage supervisor are triggered. We use *TPL5000* [13] nano-power timer from Texas Instruments, which will burn less than 100nW, thus the total P_{static} will be about 100nW instead of 2.5μW. The time constant $\frac{1}{RC}$ of the **RC** sub-circuit dictates how long after the timer output is low will the supply remain activated. This is to ensure that the microcontroller is given just enough time to perform one harvested voltage sampling and transmit its data back to the RFID reader. We leverage this data to display a status update on the software interface to notify the user of how much charge the *WISPCam* currently has. This implementation is shown in Figure 8, where the charge status update is highlighted in red.

V. Conclusion

This paper presented an overview of *WISPCam* design, which is a bettery-free programmable wireless camera. It also presented two extensions built on top of *WISPCam*. The first one, talks about implementing a protocol leveraging two-way communication capabilities of the RFID reader and *WISPCam* to retrieve pixels lost during image transmit, this will help enabling some smart inputs from the reader gets transmitted to the *WISPCam*. The second extension was to implement an ultra-low power wake up circuitry to update the reader with *WISPCam* charge status periodically while *WISPCam* is being charged. This circuit is based on an external nano-power timer circuit and a passive **OR** mechanism.

VI. Acknowledgements

This work is funded by Intel Science and Technology Center for Pervasive Computing (ISTC-PC) and NSF award number CNS 1305072.

References

[1] A. Sample and J. Smith, "Experimental results with two wireless power transfer systems," in *Radio and Wireless Symposium, 2009. RWS '09. IEEE*, pp. 16–18, 2009.

[2] D. Yeager, J. Holleman, R. Prasad, J. Smith, and B. Otis, "Neuralwisp: A wirelessly powered neural interface with 1-m range," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 3, pp. 379–387, Dec. 2009.

[3] M. Buettner, R. Prasad, M. Philipose, and D. Wetherall, "Recognizing daily activities with RFID-based sensors," in *UbiComp*, Sept. 2009.

[4] "Farsens battery-free sensor solutions." http://www.farsens.com/en/battery-free-sensor-solutions. Accessed December 2014.

[5] "Tire pressure and brake temperature systems – smartstem." http://www.craneae.com/Products/Sensing/SmartStem.aspx. Accessed December 2014.

[6] V. Talla, B. Kellogg, B. Ransford, S. Naderiparizi, S. Gollakota, and J. R. Smith, "Powering the next billion devices with wi-fi," *CoRR*, vol. abs/1505.06815, 2015.

[7] S. Naderiparizi, A. N. Parks, Z. Kapetanovic, B. Ransford, and J. R. Smith, "Wispcam: A battery-free rfid camera," in *2015 IEEE International Conference on RFID (RFID)*, pp. 166–173, April 2015.

[8] S. Naderiparizi, Y. Zhao, J. Youngquist, A. P. Sample, and J. R. Smith, "Self-localizing battery-free cameras," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '15, (New York, NY, USA), pp. 445–449, ACM, 2015.

[9] S. Naderiparizi, Z. Kapetanovic, and J. R. Smith, "Battery-free connected machine vision with wispcam," *GetMobile: Mobile Comp. and Comm.*, vol. 20, pp. 10–13, July 2016.

[10] J. Tan, P. Paweczak, A. Parks, and J. R. Smith, "Wisent: Robust downstream communication and storage for computational rfids," in *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9, April 2016.

[11] "MSP430FR5969 datasheet." http://www.ti.com/lit/ds/symlink/msp430fr5969.pdf. Accessed October 2016.

[12] "TPS780 datasheet." http://www.ti.com/lit/ds/symlink/tps780.pdf. Accessed October 2016.

[13] "TPL5000 datasheet." http://www.ti.com/lit/ds/symlink/tpl5000.pdf. Accessed October 2016.