Wireless Identification and Sensing Platform Version 6.0

Rohan Menon*
Rohit Gujarathi*
rohanme@uw.edu
rgujju@uw.edu
University of Washington

Ali Saffari
saffaria@uw.edu
University of Washington

Joshua R. Smith
jrs@cs.washington.edu
University of Washington

ABSTRACT

Connected devices are becoming more ubiquitous, but powering them remains a challenge. The Wireless Identification and Sensing Platform (WISP) is a fully programmable device capable of energy harvesting and backscatter communication. It can accommodate a variety of sensing modalities and operate without batteries or a wired power supply, making it a suitable device for ubiquitous computing. A new version of WISP is presented. WISP-6.0 is designed to be low-power, modular, and enable dual energy harvesting from sources like a solar panel. Additionally, an upgraded cross-platform host application is built using the latest web technologies. Compared to its predecessor, WISP-5.1, WISP-6.0 consumes 13.62% and 6.29% less power in active accelerometer and active acknowledgment modes respectively. Furthermore, WISP-6.0 is better able to harvest RF energy collected from its antenna, with the greatest improvements at higher input powers.

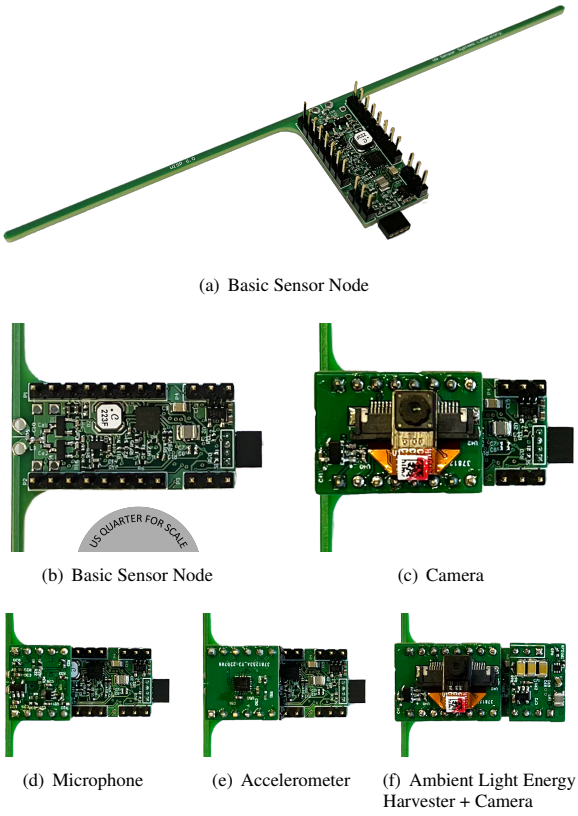
CCS CONCEPTS

• **Hardware → Wireless integrated network sensors**; **Sensor devices and platforms**.

KEYWORDS

WISP, sensor platforms, energy harvesting, wireless sensor networks, Internet of Things, battery-free camera, battery-free microphone, battery-free sensor

ACM Reference Format:
Rohan Menon, Rohit Gujarathi, Ali Saffari, and Joshua R. Smith. 2022. Wireless Identification and Sensing Platform Version 6.0. In *The 20th ACM Conference on Embedded Networked Sensor Systems (SenSys '22), November 6–9, 2022, Boston, MA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3560905.3568109



(a) Basic Sensor Node
(b) Basic Sensor Node
(c) Camera
(d) Microphone
(e) Accelerometer
(f) Ambient Light Energy Harvester + Camera

Figure 1: Prototype hardware.

1 INTRODUCTION

In recent decades, significant progress has been made toward ubiquitous computing, largely due to advancements in smaller and cheaper microcontrollers and sensors. However, traditional connected devices come with their own set of challenges. Namely, powering these wireless devices can be difficult. Wired sensors are limited in placement options and mobility. Battery-powered sensors are more mobile, but require maintenance to replace or recharge batteries. Additionally, the batteries themselves on these sensors can be relatively large, making embedding these devices difficult.

RF-powered sensors avoid these challenges by collecting their required energy wirelessly. They use the energy transmitted by a reader within range to power themselves and send back a response. The now widespread RFID standard uses collected RF energy to function without another power source. However, traditional RFID tags are not capable of sensing. RFID tags typically use a non-programmable IC to report a fixed ID when in the range of a reader. While this can be useful for inventory and tracking purposes, sensing requires a more flexible microcontroller.

WISP is an RFID-based RF-powered platform designed specifically for sensing [29]. WISP devices store energy harvested from an in-range RFID reader in a capacitor. When sufficient energy has

*Both authors contributed equally to this research.

ENSsys '22, November 6, 2022, Boston, MA, USA

ACM ISBN 978-1-4503-9886-2/22/11…$15.00
https://doi.org/10.1145/3560905.3568109

been collected, an onboard microcontroller takes a reading from a connected sensor. WISP devices use the Electronic Product Code (EPC) standard, a specification based on UHF RFID [17], to encode this sensor data and send it back to the reader. Since WISP was first developed, a variety of sensor types have been made, ranging from temperature sensors [28] to cameras [24].

Contributions. WISP-6.0 is a new iteration of the WISP project. Instead of focusing on the development of any individual sensor, we aim to consolidate the work of many previous WISP projects. In this paper, we contribute a modular platform for building battery-free sensors, a new RF frontend, dual-energy harvesting, and a companion host application for deploying sensors. Moreover, we open We begin by providing background on RFID communication and a brief history of the WISP project. Then we present our new system, including a new modular sensor design, and a companion host application. Finally, we evaluate our proposed system.

Platform availability. WISP-6.0 hardware, firmware, and desktop application files are available at:

https://github.com/wisp

2 RELATED WORK

We review the EPC RFID communication standard and prior work in developing WISP.

RFID Communication Primer: The EPCglobal Gen2 specification defines two types of devices in an RFID system [17]. An interrogator and a tag. An interrogator generates an RF carrier with DSB-ASK, SSB-ASK, or PR-ASK modulation and uses PIE encoding. A tag operates by harvesting this energy and communicates by backscattering the carrier wave using ASK or PSK modulation with FM0 baseband or Miller encoding. WISP currently supports FM0 encoding only.

An interrogator (RFID reader) typically interfaces with a host computer via ethernet or serial communication. Tags communicate with an interrogator through a sequence of exchanges known as an *Inventory Round*. An interrogator starts an inventory round in its surrounding area by sending a *Query* command. When a tag receives power from the RF carrier, it enters the *Ready* state. If a tag in the *Ready* state receives a *Query* command, it generates a random number, loads it into a slot counter, and transitions to the *Arbitrate* state if the number is non-zero or to the *Reply* state if the number is zero. The *Arbitrate* state is an anti-collision mechanism, designed to reduce the number of tags communicating at the same time. In the *Arbitrate* state, the tag decrements the number in the slot counter each time it receives the *QueryRep* command from the interrogator. Once the slot counter reaches zero, the tag transitions to the *Reply* state. The tag in *Reply* state backscatters an RN16, a 16-bit random number. If the interrogator sends a valid ACK with a valid RN16, the tag transitions to the *Acknowledged* state and can respond with various replies. One of which is an EPC (Electronic Product Code). WISP uses this response to communicate with the host, using the last 2 bytes to identify a WISP tag and the remaining 10 bytes to contain arbitrary sensor data.

An interrogator can further transmit a *Req_RN* command to request a handle for the tag. The handle can be used to perform read and write operations in a specific tags memory. When a tag receives *Req_RN* command, it replies with a handle and can transition to an *Open* or *Secured* state. If the tag has a non-zero password, it transitions to a *Secured* state, while if it does not have a password, it transitions to the *Open* state. An interrogator can perform various access commands when a tag is in these two states. If a tag receives a *Kill* command while in *Open* or *Secured* state, it transitions to the *Killed* state which permanently disables the tag. The WISP does not currently implement the *Open*, *Secured* or *Killed* states.

History of WISP: Wireless Identification and Sensing Platform (WISP) was designed and built by Smith et al. at Intel Research Seattle lab in 2006 [29], [28]. According to the authors, WISP was the first fully programmable computing system that can be powered wirelessly. It harvests energy from a long-range UHF (860-960 MHz) RFID reader, thus eliminating the need for batteries. The key novelty enabling WISP for ubiquitous computing is the fully configurable 16-bit microcontroller for performing arbitrary sensing and computation and the use of low bit-rate UHF backscatter communication to communicate an RFID reader, enabling a half-duplex link. This initial version of WISP operated at a data rate of 960bps. The platform was used to develop many low-power sensing devices capable of reporting temperature, humidity, ambient light, and position using an accelerometer.

Holleman et al. designed NeuralWISP, a wireless neural interface based on WISP [18]. NeuralWISP can be implanted due to it being wirelessly powered. Buettner et al. deployed a network of WISPs for daily activity recognition [15]. Zhao et al. built the NFC-WISP, which can communicate with an NFC-enabled smartphone [31]. It has a 2.7" e-ink display and a temperature sensor which were used to show a graph of the temperature of a milk carton for applications in a cold storage supply chain.

Naderiparizi et al. built WISPCam, which integrates a camera with WISP, enabling richer sensing applications [24]. Cameras consume more energy and generate more data than previously explored sensors. They overcame these challenges by creating a novel capacitor charge-storage model accounting for leakage and a scheme to capture and transmit a large payload, such as an image, over a backscatter link. In further work, WISPCam is demonstrated in machine vision applications like face detection, recognition and surveillance [22], [23], [25]. Real-time video streaming was not possible with this architecture due to power-hungry components like LNA and ADC. Further research redesigned the camera architecture to bypass these power-hungry components and fed the raw pixel voltages from the camera sensor directly to the backscatter communication [21].

More recently, Saffari et al. have designed a dual power harvesting camera system that combines energy from an RF and Solar harvester [26]. This allows for real-time video streaming with COTS parts at up to 13 fps. They further developed an occupancy detection system using a YOLOv5 model implemented on a Raspberry Pi4 board [27].

3 SENSOR DESIGN

Our goal is to develop a programmable RFID node that can accommodate different sensing modalities, communicate with a nearby base station (RFID reader acting as an interrogator) using the EPC Class 1 Gen 2 standard, and harvest energy from it. To enable this,

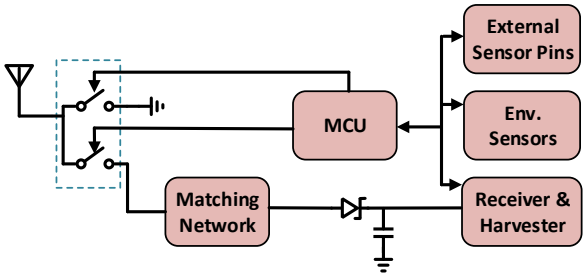


Figure 2: Block diagram of a basic WISP-6.0 board. Optional add-on sensors and harvesters can connect through the external sensor and harvesting pins.

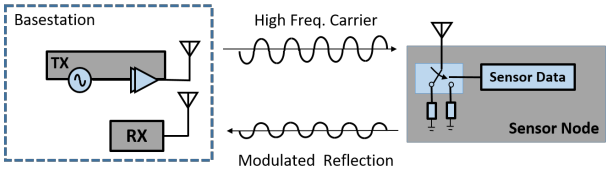


Figure 3: Backscatter communication system.

we use a low-power microcontroller with multiple wired serial communication protocols as the control unit. An RF harvester IC is used to collect energy from the base station and power the node. Additionally, an environmental sensor on the main WISP board demonstrates the functionality of the basic sensor node. Through the provided external pins, a user can add more sensors to the basic sensor node as needed. Fig. 2 shows the block diagram of our design.

Backscatter: Backscatter [20] is a low-power communication technology, wherein a directly-powered transmitter generates a high-frequency carrier signal, a sensor modulates the carrier by reflecting or absorbing the signal, and a receiver unit listens to the modulated packets and decodes them. Therefore, in a backscatter system, we delegate the high power components to the directly-powered base station and keep the sensor's power consumption low. Fig. 3 shows a backscatter communication system, where the base station composed of a transmitter and receiver unit generates the carrier and demodulates the reflected packets by the sensor node. An RF switch connected to the sensor's antenna is used to modulate the carrier.

Energy Harvesting: WISP is a self-powered platform wherein the device collects energy from the RFID reader to operate. Our harvester design rectifies the RF signal and stores the energy on a capacitor. When enough energy has been collected, it powers the system from the capacitor.

Down-Link Receiver: In our design, WISP remains in a deep-sleep mode until activated by the reader, therefore, the reader needs to be able to trigger WISP to wake remotely. We use a low-power receiver design to enable this link. Our MCU is in deep-sleep mode and consumes a small amount of power until the receiver receives the activation packet from the reader.

Modular Design: As mentioned, we aim to design a modular board that can accommodate many sensing modalities. In our design, external sensors are mounted on top of the basic board whose block diagram is shown in Fig. 2. We provide 16 IO pins on the board that can communicate with an arbitrary sensor through an SPI/UART/I2C connection.

Mount-on Boards: To verify our system works with different sensing modalities, we design multiple sensors as mount-on boards. These sensors are powered by the harvester, record data, communicate with MCU, and transfer the result to the reader via backscatter communication. Moreover, we design a solar harvester board that can be added to the basic sensor board to harvest energy from ambient light to supplement RF harvesting. In conventional RFID sensor design, sensors collect the entirety of their energy through RF harvesting. As a result, their range is limited by the minimum energy level the RF harvester IC can collect. However, ambient light energy harvesting is not range limited and can potentially improve the range of an RF-only sensor.

4 SENSOR IMPLEMENTATION

We implement WISP-6.0, consisting of basic sensor node units and mount-on boards on a 4-layer FR4 PCB.

4.1 Basic Sensor Node

Our basic sensor node is able to collect RF energy from a base station, receive commands from it, measure environmental data, and backscatter the results back to the base station. We use a TI MSP430FR5969 [8] microcontroller as the control unit. This MCU is ultra low-power while in sleep mode (0.3 μA), which is an essential requirement in developing battery-free sensors [19].

In sleep mode, our receiver is enabled and listens to the packets from the reader; Fig. 4 shows the receiver design where the input is the RF signal and it is shared with the harvester. The first stage of the receiver circuit rectifies the RF signal using an SMS7621 Skyworks Schottky diode [11] and a 10 pF capacitor. Next, we feed the rectified signal into an average calculator stage, where the signal itself is compared with its dynamic average. The output of the comparator is the decoded message from the RFID reader with a maximum bit rate of 640 Kbps. We use an Onsemi NCS2200 comparator with a quiescent supply current of 10 μA to decode the signal [9]. A simple pin read function is implemented on the MCU to read and store the comparator's output.

We equip our basic board with a low-power TI HDC2010 [5] humidity and temperature sensors to demonstrate an end-to-end battery-free RFID-based sensing platform without using any mount-on boards. The MCU queries and receives data from the sensor through an I2C connection. Once the data is read by the MCU, we use our RF switch (ADG902 from Analog Devices [1]) to backscatter data. As mentioned, backscatter data transfer in RFID communication happens when the sensor modulates the carrier by switching between two impedance levels (50 ohm and short). Fig. 3 shows that the antenna is connected to the RF switch whose state changes between two levels based on the data from the MCU. One state

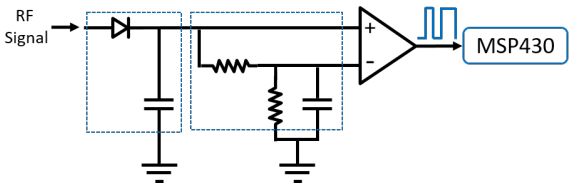


Figure 4: Low-power down-link receiver circuit diagram.

mostly absorbs the carrier, while the other one reflects the signal back.

Finally, we use a TI BQ25570 harvester IC [4] to collect energy from the reader and store it on a capacitor. This IC provides a 2 V (minimum voltage level is dictated by the MCU's FRAM [8]) supply to power up the board. Power hungry sensors require a larger capacitor to collect enough energy for the operation, while low-power sensors can work with storage capacitance as low as hundreds of μF. Similar to the receiver design, we need to rectify the RF signal before feeding it to the harvester IC. Fig. 1(a) and 1(b) show our basic sensor node prototype.

4.2 Mount-On Boards

Our basic board design provides 16 IO pins for interfacing with mount-on boards and sensors. Through these pins, we can power a sensor and communicate with it via an SPI/I2C/UART connection. Moreover, we provide 6 power management pins for any additional harvesters. If installed, an additional harvester works independently from the RF harvester of the basic sensor board and can collect energy from any source in its separate storage element. A simple switch controlled by the harvesters selects between the sources once they collect enough energy to power the sensor. We develop multiple mount-on boards such as cameras, microphones, accelerometers, and an ambient light energy harvester as sample boards. However, supported sensing modalities and harvesting sources are not limited to these boards and the user can design new sensors and harvesters.

Camera: We use a Himax HM01B0 [6] gray-scale image sensor to implement our camera mount-on board. Our MCU initializes the image sensor through an I2C connection in a QQVGA mode (160×120 pixels) with a pixel depth of 8 bits. We set the MCU in SPI peripheral mode to receive the image data from the sensor which is the SPI controller. In QQVGA mode, our image sensor only consumes 1.1 mW. Fig. 1(c) shows our camera board mounted on a basic sensor node board.

Microphone: A VM1010 Vesper [14] microphone and Analog Devices MAX4466 amplifier [7] are used to build our microphone board. The microphone sensor has two operating modes: 1- wake-on sound, and 2- normal. In the former, the microphone is inactive and waits for an acoustic event to trigger the normal mode. In this mode, the microphone consumes only 18 μW. Once the sensor triggers the normal mode, we enable the amplifier to amplify the raw analog audio data up to 100 times. Our MCU's ADC block samples the amplified audio signal at 8000 samples per second. Fig. 1(d) shows our microphone board mounted on the basic board.

Accelerometer: ADXL362 from Analog Devices [2] is the main IC on this mount-on sensor. The IC provides a 3-axis measurement and transfers the data to the MCU via an SPI connection. The IC only consumes 270 nA in the motion-activated wake-up mode which can be used to gate the normal operation with the current consumption of 1.8 μA. Fig. 1(e) shows our accelerometer board on the basic sensor.

Ambient Light Energy Harvester: We use the same IC as the RF harvester on the basic sensor node. Since the output of the panel is already a DC signal, A solar panel can directly connect to our harvester IC without a rectifier. This light harvester allows us to operate at distances where the RF harvester stops working. Note that in a typical RFID setup the range is limited by the RF harvester. By powering the sensors through other means we can potentially improve the operating range. Furthermore, adding this harvester improves the sensor's update rate because we harvest energy from two sources. In ambient light energy harvesting, we collect more energy as the light intensity increases or when we use larger panels. Fig. 1(f) show a basic sensor node equipped with the ambient light energy harvester and the camera board.

4.3 WISP-6.0 vs. WISP-5.1

As mentioned, the major improvement in this revision of the design is the modular approach, where our hardware can accommodate many sensors and enables other means of energy harvesting which ultimately will improve the operation range. Additionally, our software platform is not focused on any single type of sensor data and users can add widgets for new sensors or different visualizations of sensor data. In our hardware design, we take advantage of advancements in IC design and choose more power-efficient/powerful designs which were not available when previous versions were implemented. Table 1 summarizes the changes and reasoning behind our choices.

5 HOST APPLICATION

Data from WISP is transmitted by changing the EPC, a 12-byte value that can be read using a commercially available RFID reader [29]. However, significant processing is often required to convert these EPC values to meaningful sensor data. Due to the variety of WISP sensor types possible, we demonstrate a modular host application to process and visualize WISP EPC data in real-time.

The main challenge associated with such an application is the volume of incoming tag data, which can reach up to 330 tags per second with camera tags [24]. This mandates a separation between the processing of incoming tags and the user interface. This is accomplished with a web-based front-end using the React JS framework [16] that operates alongside a Python script that handles communication with the reader (Fig. 5). Sllurp [30], an open source Python package, is used to interface with a network-connected Impinj RFID reader. Tags read in an inventory round by the reader are parsed based on their sensor type. For most tags, an entire sensor reading is contained within a single EPC. However, for camera tags, a single image is spread across many thousands of EPCs. In this case, pixels from individual tags are accumulated until a complete image can be formed.

This formatted data is then sent to the user interface. Here, widgets can display this information in various sensor-specific ways. For example, an accelerometer widget visualizes an accelerometer WISP

Table 1: Minor modifications to WISP-6.0 hardware and expected improvements by comparing data sheets or cited publications.

IC	WISP-5.1	WISP-6.0	Improvement
RF Harvester	S-882Z24 [10]	BQ25770 [4]	Max. input voltage increases from 3 V to 5.1 V.
RF Switch	BF1101WR [3]	ADG902 [1]	Insertion loss reduces from 6.9 dB to 0.8 dB.
Voltage Regulator	TPS780xx [12]	TPS7A02xx [13]	Quiescent current reduces from 500 nA to 25 nA.
Image Sensor	OVM7690 [24]	HM01B0 [19]	Capturing energy reduces from 8.51 mJ to 6.05 mJ.

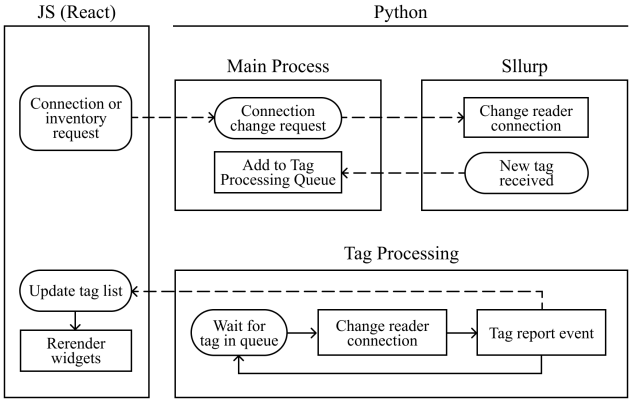


Figure 5: Host application block diagram. Dashed arrows indicate event-based signaling between processes.

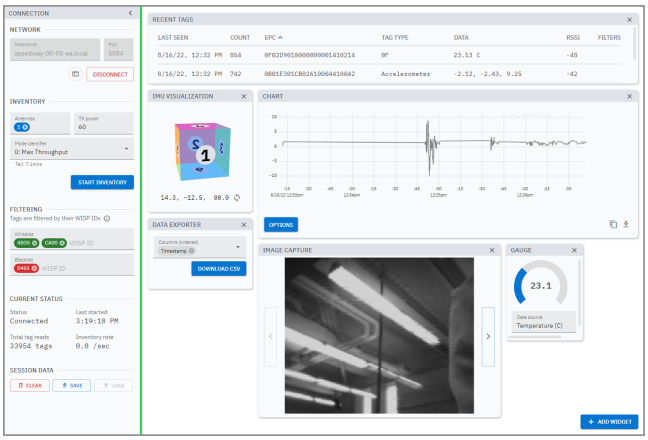


Figure 6: The host application GUI supports a customizable dashboard where widgets display information from incoming tags.

tag as a cube oriented in 3D space (Fig. 6). Widgets can also serve as general-purpose utilities, such as graphing and data export widgets

6 EVALUATION

To demonstrate the performance of our system, we evaluate the host application and device independently. The host application is evaluated based on its WISP tag processing rate. The WISP-6.0 device is evaluated based on its power consumption in various modes.

6.1 Host Application

We evaluate the host application by measuring the average amount of time required for the application to process the 12-byte EPC packet from a WISP tag. As different sensors require significantly different levels of processing, we test the application with three WISP sensors: (1) A camera tag, which requires the application to accumulate pixels over the image's transmit period and perform image inpainting to remove occlusions; (2) An accelerometer tag, which requires simple algebra to compute acceleration values; (3) An acknowledgment tag, which requires no sensor data processing and acts as a baseline. Our results are listed in Table 2.

Table 2: The average processing time of the application for three WISP sensors.

WISP Sensor Type	Processing Time
Camera	825.3 μs
Accelerometer	76.4 μs
Acknowledgment	24.3 μs

We find that the host application has a relatively small overhead for processing WISP EPC packets. Accelerometer and acknowledgment EPC packets are not likely to limit the system's performance. In contrast, camera EPC packets show an average processing time of 0.83 ms. The inverse (1200 packets per second) is the absolute maximum rate of camera EPC packets. As individual camera tags can reach a rate of 330 packets per second [24], we are limited by the application to 3-4 simultaneous camera readings.

6.2 Power Consumption

We measure the power consumption of the WISP-6.0 and WISP-5.1 in listening mode, where the MCU is in deep-sleep mode and the down-link receiver's comparator is active. Next, we program WISP-5.1 and WISP-6.0 boards to continuously send acknowledgment packets to the RFID reader and we measure the average power consumption. Finally, we repeat the experiments for the accelerometer tag. Table 3 lists the power consumption in these modes.

In comparison with its predecessor (WISP-5.1), WISP-6.0 consumes less power in active states for both accelerometer and acknowledgment tags. In listening mode, WISP-6.0 performs slightly worse, consuming 4.2% more power than the previous version.

Table 3: Average power consumption in listening mode and for active accelerometer and active acknowledgment tags.

Measurement	WISP-5.1	WISP-6.0
Listening Mode (μW)	21.32	22.22
Active Accelerometer (μW)	459.20	396.66
Active Acknowledgment (μW)	895.80	839.42

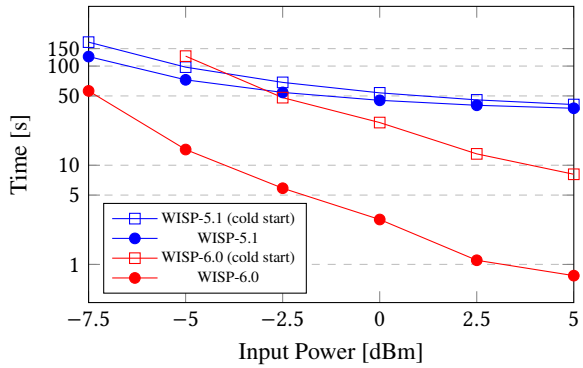


Figure 7: Time required to capture 500μJ of energy.

6.3 Energy Harvesting Rate

To compare the energy harvesting rates of WISP-5.1 and WISP-6.0, we measure the time required for each to harvest 500μJ of energy in two scenarios, (1) from cold start, i.e. when no charge is on the capacitor, and (2) after a cold start , i.e. in between successive energy harvests.

To measure this energy harvesting, we place a constant load on the regulated output of each WISP device. The voltage across this load is measured and the power dissipation of the load is measured over time at a variety of input power levels to calculate the total accumulated energy.

Figure 7 shows the time required for the WISP devices to accumulate 500μJ of energy. WISP-6.0 can accumulate energy in a shorter period of time across most input powers, but is unable to cold start below an input power of -7dBm. This is likely due to the minimum input voltage requirements of the BQ25770 energy harvesting chip on WISP-6.0.

7 CONCLUSION

Ubiquitous computing has seen significant progress, with devices becoming smaller, cheaper, and increasingly common. However, powering these connected devices remains a challenge. RF-powered devices offer a potential solution to creating battery-less sensor networks. With its modular design, we believe the WISP framework addresses many of the challenges associated with developing such a sensor, providing a starting point to develop any number of wireless, battery-less sensor models. The companion host application can make it easier to test and deploy WISP devices in the field.

Future Work: Reducing the physical size of WISP-6.0 is an area for further research. We imagine that, just as standard RFID devices have become significantly smaller since the technology was introduced, it's possible to miniaturize RF-powered sensors. The PCB dipole antenna contributes significantly to its overall dimensions. Further experiments can be performed to find a suitable replacement antenna that provides sufficient gain and satisfies size constraints.

Additionally, the performance of the host application can be improved to allow for a higher rate of WISP tag readings making it possible to capture more simultaneous data from camera and microphone sensors.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their helpful feedback. This work was funded by NSF award CNS-1823148.

REFERENCES

[1] 2022. *ADG902 datasheet*. https://www.analog.com/en/products/adg902.html
[2] 2022. *ADXL362 datasheet*. https://www.analog.com/en/products/adxl362.html#product-overview
[3] 2022. *BF1101WR datasheet*. https://www.nxp.com/part/BF1101WR#/
[4] 2022. *BQ25570 datasheet*. https://www.ti.com/product/BQ25570
[5] 2022. *HDC2010 datasheet*. https://www.ti.com/product/HDC2010/part-details/HDC2010YPAR
[6] 2022. *HM01B0 datasheet*. https://www.himax.com.tw/products/cmos-image-sensor/always-on-vision-sensors/hm01b0
[7] 2022. *MAX4466 datasheet*. https://www.maximintegrated.com/en/products/analog/audio/MAX4466.html
[8] 2022. *MSP430FR5969 datasheet*. https://www.ti.com/product/MSP430FR5969
[9] 2022. *NCS2200 datasheet*. https://www.onsemi.com/products/signal-conditioning-control/amplifiers-comparators/comparators/ncs2200
[10] 2022. *S-882Z24 datasheet*. https://www.digikey.com/en/products/detail/ablic-inc/S-882Z24-M5T1G/1662214
[11] 2022. *SMS7621 datasheet*. https://www.skyworksinc.com/Products/Diodes/SMS7621-Series
[12] 2022. *TPS780xx datasheet*. https://www.ti.com/product/TPS780/part-details/TPS780330220DDCR
[13] 2022. *TPS7A02xx datasheet*. https://www.ti.com/product/TPS7A02
[14] 2022. *VM1010 datasheet*. https://vespermems.com/products/vm1010
[15] Michael Buettner, Richa Prasad, Matthai Philipose, and David Wetherall. 2009. Recognizing daily activities with RFID-based sensors. In *Proceedings of the 11th international conference on Ubiquitous computing*. ACM. https://doi.org/10.1145/1620545.1620553
[16] Facebook. 2022. React JS. https://github.com/facebook/react.
[17] GS1. [n.d.]. EPC Radio-Frequency Identity Protocols Generation-2 UHF RFID Standard. https://www.gs1.org/sites/default/files/docs/epc/gs1-epc-gen2v2-uhf-airinterface_i21_r_2018-09-04.pdf.
[18] Jeremy Holleman, Dan Yeager, Richa Prasad, Joshua R. Smith, and Brian Otis. 2008. NeuralWISP: An energy-harvesting wireless neural interface with 1-m range. In *2008 IEEE Biomedical Circuits and Systems Conference*. IEEE. https://doi.org/10.1109/biocas.2008.4696868
[19] Mohamad Katanbaf, Ali Saffari, and Joshua R. Smith. 2021. MultiScatter: Multistatic Backscatter Networking for Battery-Free Sensors. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems* (Coimbra, Portugal) *(SenSys '21)*. Association for Computing Machinery, New York, NY, USA, 69–83. https://doi.org/10.1145/3485730.3485939
[20] Vincent Liu, Aaron Parks, Vamsi Talla, Shyamnath Gollakota, David Wetherall, and Joshua R. Smith. [n.d.]. Ambient Backscatter: Wireless Communication out of Thin Air *(SIGCOMM '13)*.
[21] Saman Naderiparizi, Mehrdad Hessar, Vamsi Talla, Shyamnath Gollakota, and Joshua R Smith. 2018. Towards Battery-Free HD Video Streaming. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*. USENIX Association, Renton, WA, 233–247. https://www.usenix.org/conference/nsdi18/presentation/naderiparizi
[22] Saman Naderiparizi, Zerina Kapetanovic, and Joshua R. Smith. 2016. Battery-Free Connected Machine Vision with WISPCam. *GetMobile: Mobile Comp. and Comm.* 20, 1 (July 2016), 10–13. https://doi.org/10.1145/2972413.2972417
[23] Saman Naderiparizi, Zerina Kapetanovic, and Joshua R Smith. 2016. Wispcam: An rf-powered smart camera for machine vision applications. In *Proceedings of the 4th International Workshop on Energy Harvesting and Energy-Neutral Sensing Systems*. ACM, 19–22.
[24] Saman Naderiparizi, Aaron N. Parks, Zerina Kapetanovic, Benjamin Ransford, and Joshua R. Smith. 2015. WISPCam: A battery-free RFID camera. In *2015*

IEEE International Conference on RFID (RFID). IEEE. https://doi.org/10.1109/rfid.2015.7113088

[25] Saman Naderiparizi, Yi Zhao, James Youngquist, Alanson P Sample, and Joshua R Smith. 2015. Self-localizing battery-free cameras. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 445–449.

[26] Ali Saffari, Mehrdad Hessar, Saman Naderiparizi, and Joshua R. Smith. 2019. Battery-Free Wireless Video Streaming Camera System. In *2019 IEEE International Conference on RFID (RFID)*. IEEE. https://doi.org/10.1109/rfid.2019.8719264

[27] Ali Saffari, Sin Yong Tan, Mohamad Katanbaf, Homagni Saha, Joshua R. Smith, and Soumik Sarkar. 2021. Battery-Free Camera Occupancy Detection System. In *Proceedings of the 5th International Workshop on Embedded and Mobile Deep Learning*. ACM. https://doi.org/10.1145/3469116.3470013

[28] A.P. Sample, D.J. Yeager, P.S. Powledge, A.V. Mamishev, and J.R. Smith. 2008. Design of an RFID-Based Battery-Free Programmable Sensing Platform. *IEEE Transactions on Instrumentation and Measurement* 57, 11 (nov 2008), 2608–2615. https://doi.org/10.1109/tim.2008.925019

[29] Joshua R. Smith, Alanson P. Sample, Pauline S. Powledge, Sumit Roy, and Alexander Mamishev. 2006. A Wirelessly-Powered Platform for Sensing and Computation. In *UbiComp 2006: Ubiquitous Computing*. Springer Berlin Heidelberg. https://doi.org/10.1007/11853565_29

[30] Florent Viard. 2021. Sllurp 2. https://github.com/fviard/sllurp.

[31] Yi Zhao, Joshua R. Smith, and Alanson Sample. 2015. NFC-WISP: A sensing and computationally enhanced near-field RFID platform. In *2015 IEEE International Conference on RFID (RFID)*. IEEE. https://doi.org/10.1109/rfid.2015.7113089