Warp: A Hardware Platform for Efficient Multi-Modal Sensing with Adaptive Approximation

Phillip Stanley-Marbell
University of Cambridge

Martin Rinard
MIT

We present *Warp*, the first open hardware platform designed explicitly to support research in approximate computing. Warp incorporates 21 sensors, computation, and circuit-level facilities designed explicitly to enable approximate computing research, in a 3.6 cm×3.3 cm×0.5 cm area. Warp uses these facilities to support a wide range of precision and accuracy versus power and performance tradeoffs.

Sensor integrated circuits are critical components of many hardware platforms, from augmented reality and wearable health monitors, to drones. Sensors convert physical signals such as temperature, vibration, rotation, and so on, into signals which are then digitized and used in computations. Because sensor circuits are often constrained by the physics of the phenomena they are designed to measure, sensors often do not benefit from the scaling of semiconductor technology that has enabled dramatic reduction in power dissipation of digital logic. As a result, sensors today constitute an important component of the power dissipation in many energy-constrained platforms. Such energy-constrained platforms are a promising next frontier for application of techniques from approximate computing[15].

The power dissipated by sensors depends on their electrical configuration (e.g., supply voltage) as well as on their software configuration (e.g., number of bits per sample for sensors with digital interfaces). These configuration parameters also affect the precision and accuracy of samples produced by sensors. System designers can capitalize on this observation to trade energy efficiency and performance for precision and accuracy. These tradeoffs have been investigated, primarily for computation as opposed to sensors, by several research efforts in the last decade[1, 3, 5, 7, 8, 9, 10, 12, 16, 17, 18]. Despite this interest in efficiency versus precision and accuracy tradeoffs, no common open hardware platforms for research evaluation exist today.

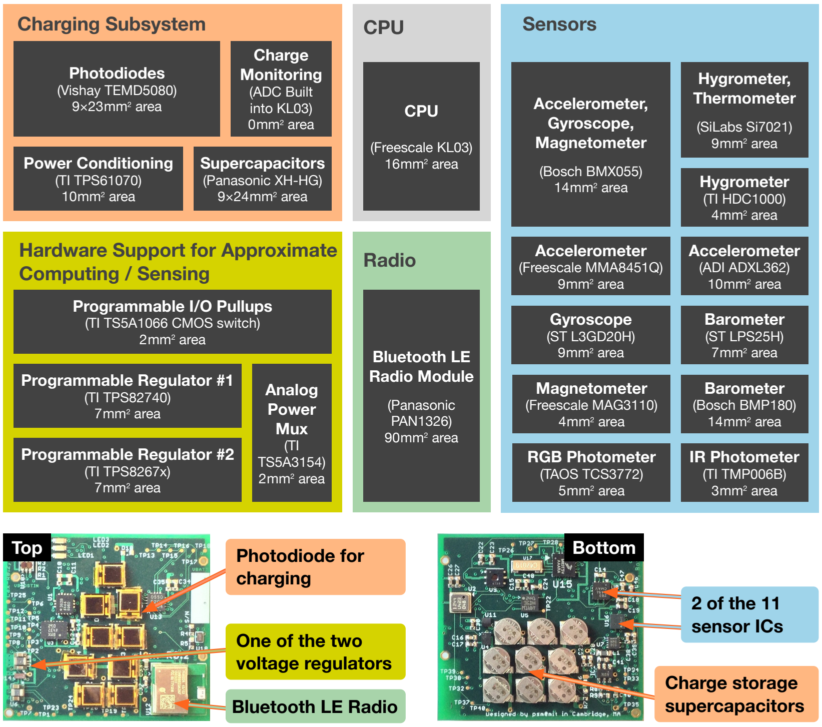

Charging Subsystem
Photodiodes (Vishay TEMD5080) 9×23mm^2 area
Charge Monitoring (ADC Built into KL03) 0mm^2 area
Power Conditioning (TI TPS61070) 10mm^2 area
Supercapacitors (Panasonic XH-HG) 9×24mm^2 area
CPU
CPU (Freescale KL03) 16mm^2 area
Sensors
Accelerometer, Gyroscope, Magnetometer (Bosch BMX055) 14mm^2 area
Hygrometer, Thermometer (SiLabs Si7021) 9mm^2 area
Hygrometer (TI HDC1000) 4mm^2 area
Hardware Support for Approximate Computing / Sensing
Programmable I/O Pullups (TI TS5A1066 CMOS switch) 2mm^2 area
Programmable Regulator #1 (TI TPS82740) 7mm^2 area
Programmable Regulator #2 (TI TPS8267x) 7mm^2 area
Analog Power Mux (TI TS5A3154) 2mm^2 area
Radio
Bluetooth LE Radio Module (Panasonic PAN1326) 90mm^2 area
Accelerometer (Freescale MMA8451Q) 9mm^2 area
Accelerometer (ADI ADXL362) 10mm^2 area
Gyroscope (ST L3GD20H) 9mm^2 area
Barometer (ST LPS25H) 7mm^2 area
Magnetometer (Freescale MAG3110) 4mm^2 area
Barometer (Bosch BMP180) 14mm^2 area
RGB Photometer (TAOS TCS3772) 5mm^2 area
IR Photometer (TI TMP006B) 3mm^2 area
Top
Photodiode for charging
One of the two voltage regulators
Bluetooth LE Radio
Bottom
2 of the 11 sensor ICs
Charge storage supercapacitors

Figure 1. The Warp hardware platform contains integrated circuits which together provide 21 sensors across eight sensing modalities (further detailed in Table 1). Warp combines this diversity with circuit support to enable approximate computing tradeoffs between precision, accuracy, performance, and energy-efficiency.

This article introduces Warp, an open hardware platform for evaluating hardware and software techniques that trade precision, accuracy, and reliability for improved efficiency in energy-constrained systems. We have made the hardware design files and our basic firmware available on GitHub[11]. Other researchers can use the hardware designs to easily recreate the Warp hardware using the manufacturing instructions we provide. Because we provide the complete hardware and firmware design source, researchers can also extend Warp as they see fit. Warp fills an unmet need for research evaluation hardware, and the measurements from platforms such as Warp could serve as valuable error models for research on algorithmic, programming language, and system software techniques for approximate computing. Figure 1 shows the system components of Warp. Warp's design provides facilities for trading sensor precision for energy usage, trading sensor accuracy for energy usage and performance, and trading sensor access reliability for energy and performance

Although Warp contains a photovoltaic subsystem for charging and a supercapacitor array for charge storage, Warp is neither targeted at energy-scavenged systems nor at intermittent computing systems. When fully charged, Warp’s supercapacitors can power the processor for over an hour. The hardware facilities for approximation which we implement in Warp are therefore complementary to research on intermittent computing[4].

WARP: AN APPROXIMATE COMPUTING PLATFORM

We designed Warp to provide a greater range of energy versus correctness tradeoffs than is available using commercial off-the-shelf hardware. We named the platform “Warp” because it provides flexibility for warping sensor values for the benefit of efficiency. Warp achieves flexibility by integrating sensors that have a broad range of hardware-implemented precisions and accuracies. Table 1 lists the sensors, their operating voltage ranges, and their output precision, accuracy, and noise characteristics.

Table 1. Operating voltage ranges, precision, accuracy, and noise properties of the sensors in Warp. Many sensor integrated circuits include temperature sensors, hence the abundance.

Sensor	Supply Voltage Range (V)	Accuracy Range (Noise Measure)	Interface Precision Range (bits/sample)
MMA8451Q accelerometer	1.95 – 3.6	99 – 126 μg/$\sqrt{\text{Hz}}$	8 or 14
BMX055 accelerometer	2.4 – 3.6	150 μg/$\sqrt{\text{Hz}}$	8 or 12
ADXL362 accelerometer	1.6 – 3.5	175 – 550 μg/$\sqrt{\text{Hz}}$	4, 8, or 12
L3GD20H gyroscope	2.2 – 3.6	0.011 °/s/$\sqrt{\text{Hz}}$	8 or 16
BMX055 gyroscope	2.4 – 3.6	0.014 °/s/$\sqrt{\text{Hz}}$	8 or 16
MAG3110 magnetometer	1.95 – 3.6	0.25 – 0.4 μT	8 or 16
BMX055 magnetometer	2.4 – 3.6	0.3 – 1.4 μT	8 or 13 (x-, y-), 15 (z-)
SI7021 hygrometer	1.9 – 3.6	±2% accuracy ±0.025–0.2% precision	8, 10, 11, or 12
HDC1000 hygrometer	3.0 – 5.0	±4% accuracy ±0.1% precision	14
LPS25H barometer	1.7 – 3.6	0.01 – 0.03 hPa	8, 16, or 24
BMP180 barometer	1.6 – 3.6	0.03 – 0.06 hPa	8, 16, or 19
HDC1000 thermometer	3.0 – 5.0	±0.2°C	14
SI7021 thermometer	1.9 – 3.6	±0.3°C	11, 12, 13, or 14
ADXL362 thermometer	1.6 – 3.5	±0.5°C	4 or 12
TMP006B thermometer	2.2	±1°C	8 or 14
BMP180 thermometer	1.6 – 3.6	±1°C	8 or 16
MAG3110 thermometer	1.95 – 3.6	$\geq \pm 1°$	8
L3GD20H thermometer	2.2 – 3.6	$\geq \pm 1°$	8
LPS25H thermometer	1.7 – 3.6	±2°C	8 or 16
BMX055 thermometer	2.4 – 3.6	±2°C	8
TCS3772 photometer	2.7 – 3.3	14%–35% Irradiance Responsivity	8 or 16 per R/G/B/clear

The sensors in Warp cover eight sensing modalities: (1) temperature, (2) acceleration in three axes, (3) angular velocity in three axes, (4) magnetic flux density in three axes (often used as a digital compass), (5) humidity, (6) pressure (for measuring, e.g., atmospheric pressure or elevation), (7) infrared radiation, and (8) color (a red-green-blue-clear sensor with filters for 615 nm, 525 nm, and 465 nm light). For each of the first six modalities, Warp contains at least two different state-of-the-art sensor integrated circuits (ICs) from different manufacturers, each of which represents a different point in the tradeoff space between precision, accuracy, power dissipation, and performance. For example, for atmospheric pressure, Warp contains an LPS25H IC which can provide up to 24 bits precision per sample, and a BMP180 IC which is limited to 19 bits precision per sample. These two ICs also have differing power dissipation and noise properties, providing software with a tradeoff between power dissipation, accuracy, and precision.

Warp uses this diversity of sensors to allow approximate computing researchers to explore precision and accuracy versus energy efficiency tradeoffs. Warp complements this inter-sensor flexibility with new hardware facilities for sensor accuracy and sensor communication reliability versus energy efficiency tradeoffs. Figure 2 shows a simplified schematic of the system, highlighting hardware support for flexible sensor precision, flexible sensor accuracy, and flexible sensor reliability, all designed to support research in approximate computing.

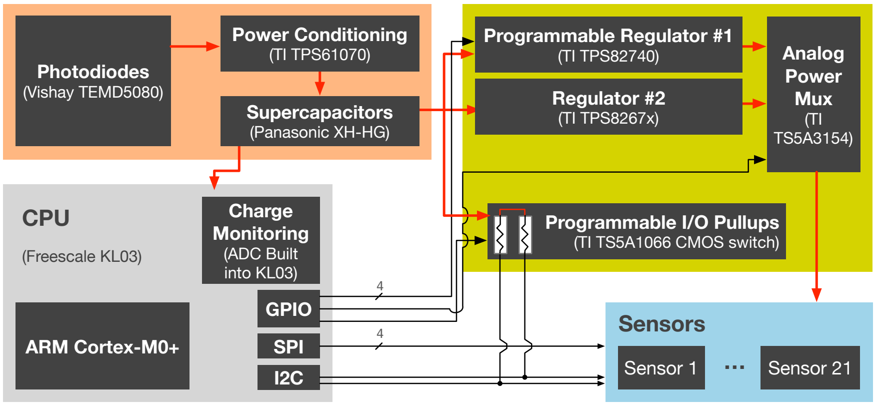


Figure 2. The processor controls the sensor operation voltage using one dynamically-programmable voltage regulator paired with a second design-time-configurable voltage regulator to trade sensor accuracy for power dissipation; sensor precision is controlled completely in software by configuration commands for each sensor as well as by choosing between sensors for a given physical signal. The processor controls I/O reliability versus power dissipation tradeoff using the programmable I/O pull-up switch.

Comparing Warp to related platforms from the domains of sensor networks and intermittent computing

Today, despite a growing body of research on techniques to trade errors for efficiency (approximate computing), there is no hardware platform that allows researchers to explore the many techniques proposed to trade errors for correctness. Warp is the first hardware platform we know of explicitly designed to support approximate computing research. Warp however exists in the context of existing research on low-power hardware platforms, including prior work such as Sunflower[14], Flicker[6], WISP[13], and contemporary work such as Capybara[4]. These prior and contemporary platforms largely address the needs of researchers in wireless sensor networks, energy scavenging, and intermittent computing. Warp addresses the needs of researchers in approximate computing. Warp might nevertheless be a useful platform in these related research areas: With 21 integrated sensors in its 3.6 cm×3.3 cm area, Warp is a third the area of Capybara while containing more than twice the number of sensors. Warp is smaller than all the aforementioned platforms except Sunflower (but Sunflower contains only four sensors). By making our complete design files and firmware publicly available[11], our intention for Warp is to provide a foundation on which researchers in approximate computing can build more sophisticated systems.

Sensor precision tradeoff facilities in Warp

By including multiple hardware implementations of sensors for the same sensing modality, each of which achieves a different energy efficiency versus precision and accuracy tradeoff (Table 1), Warp allows its users to evaluate techniques that trade precision and accuracy for efficiency. For example, for acceleration, Warp provides hardware support for sampling at 4, 8, 12, or 14 bits precision, and to do so with a range of measurement noise, by selecting amongst three different accelerometer implementations which have different energy efficiencies.

Sensor accuracy tradeoff facilities in Warp

In addition to the achieving accuracy versus energy efficiency tradeoffs by allowing software to choose between sensors (Table 1, third column), Warp implements the Lax[16] sensor hardware approximation technique using two miniature voltage regulators, each occupying less than 7 mm^2 in circuit board area.

One of the two voltage regulators is software-controllable to set the supply voltage of the system's sensors to one of eight voltage levels: either 1.8 V to 2.5 V, or 2.6 V to 3.3 V, in steps of 0.1 V. The choice between these two voltage ranges, which are implemented by two different regulators with identical printed circuit board footprint, is fixed at the point at which the board is assembled. The second voltage regulator, which is also fixed at design time, can have an output voltage of one of 1.05 V, 1.1 V, 1.2 V, 1.225 V, 1.26 V, 1.5 V, 1.6 V, 1.8 V, 1.86 V, 1.95 V, or 2.1 V. The outputs of these two regulators are fed into a software-controlled analog switch, allowing software to dynamically select between the two voltage regulators (programmable output and fixed output) at runtime. Figure 2 shows a simplified schematic of the software-controlled sensor power supply which is part of Warp's hardware support for approximate computing.

Warp's sensor supply voltage changes have a typical hardware latency of 315 µs due to the output voltage switching latency of the voltage regulators and the switching time of the analog switch. This low latency makes it feasible to implement sensor energy efficiency versus accuracy by voltage control at fine temporal granularities.

Sensor I/O reliability tradeoff facilities in Warp

Warp implements a hardware facility to allow software control of the pull-up resistors which are mandatory for the I2C serial communication standard used by most sensor integrated circuits. Disabling board-level I/O pull-ups leaves the I2C signals with only the microcontroller's on-chip pull-ups. This removes the main source of power dissipation for open-drain interfaces such as I2C, but reduces the reliability of communication. For example, for an I2C interface operating at an I/O supply voltage of 2.5V, the average power dissipated in the typical 4.7 kΩ pull-up resistor is 1.3 mW, more than the power dissipation of most sensors in Warp.

Implementation miniaturization

We optimized the implementation of Warp for size, to achieve a form factor of 3.6 cm×3.3 cm×0.5 cm that is small enough for use in user studies (e.g., as a wearable platform). To achieve this level of integration, we implemented Warp using a 10-layer printed circuit board process with a board thickness of 62 mils (1.6 mm). Fully populated with components, the Warp prototype is only ~5 mm thin. Researchers using our open hardware design as a starting point can populate the system with a subset of the sensor ICs listed in Table 1 and with a choice of different voltage regulators (both fixed and software-controlled).

EVALUATION

We highlight Warp's facilities to trade sensor access speed for average power dissipation for seven of the sensors in Warp below. Such tradeoffs are valuable for systems that are power-limited: Because energy stores such as coin cell batteries as well as supercapacitors have non-negligible internal resistance, lower power dissipation can reduce supply voltage droop. As a result, being able to trade performance for power can make the difference between a system that works and one which does not, even when it leads to larger overall energy usage. We then demonstrate the tradeoffs between power dissipation and sensor accuracy that Warp's programmable sensor supply voltage enables.

Performance versus power tradeoff results

We use a Keysight B2962A source-measure unit (SMU) for power measurements. The B2962A provides current sourcing precision of 10 fA, voltage sourcing precision of 100 nV, current measurement precision of 10 nA, and voltage measurement precision of 200 mV. These current and voltage measurement specifications enable us to measure power dissipation to a resolution of better than 1 μW.

Figure 3(a) shows a representative example of how the power dissipation for accessing a sensor (the BMX055 gyroscope) varies with I/O speed. For the BMX055 gyroscope, power dissipation increases by over 0.2 mW as the speed at which the sensor is accessed is increased from 1 kb/s to 64 kb/s. Even though power dissipation increases with I/O speed, Figure 3(b) shows that the energy per bit for I/O decreases with I/O speed.

Figure 3(c) and Figure 3(d) show similar trends in I/O power and energy per bit for seven of Warp's sensors and shows how power dissipation varies by 0.2 mW – 0.3 mW as a function of I/O speed. The magnitude of this change in I/O power dissipation is greater than the power dissipation of many of the sensors in the platform, motivating the need for precise and approximate techniques for improving I/O power efficiency.

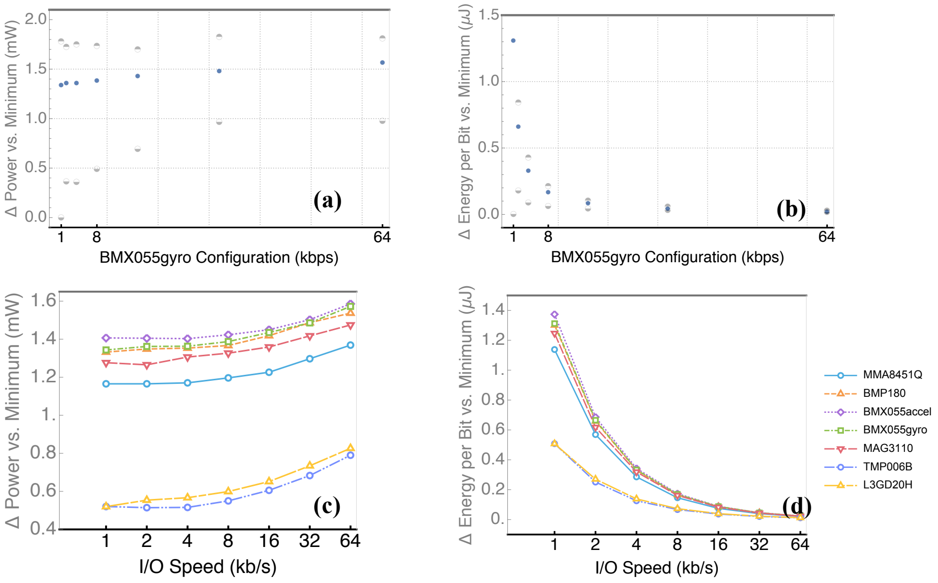
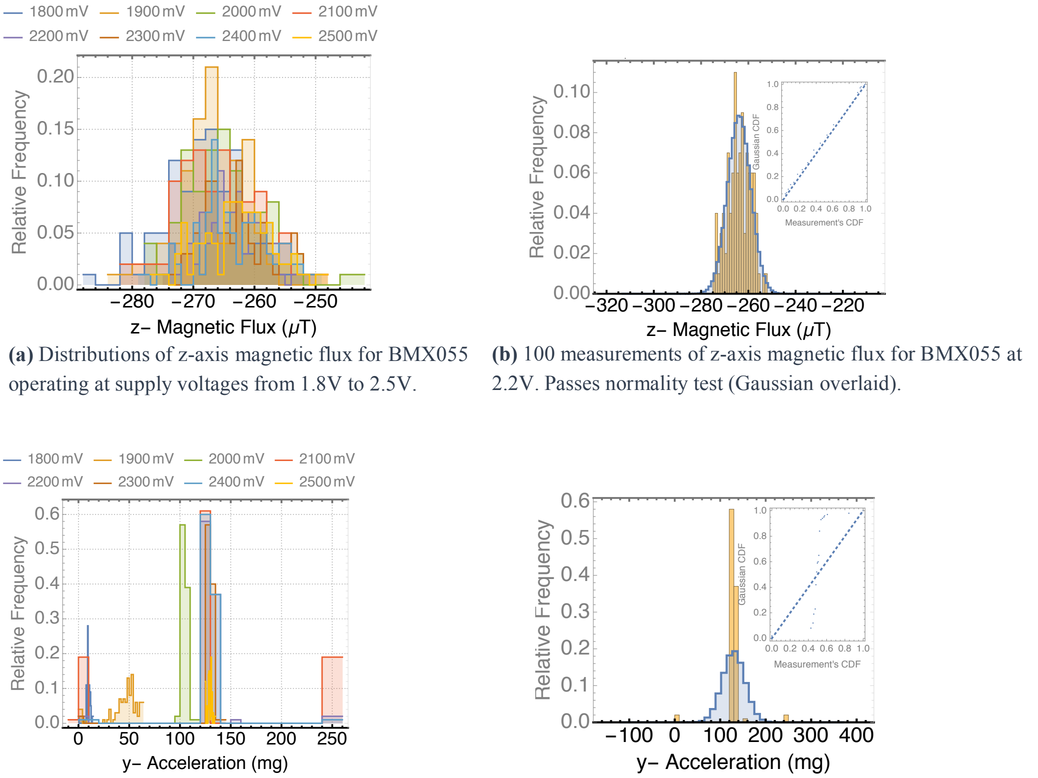


Figure 3. Warp enables tradeoffs between I/O power dissipation, energy per bit, and I/O data transfer speeds.



(a) Distributions of z-axis magnetic flux for BMX055 operating at supply voltages from 1.8V to 2.5V.

(b) 100 measurements of z-axis magnetic flux for BMX055 at 2.2V. Passes normality test (Gaussian overlaid).

(c) Distributions of y-axis acceleration for ADXL362 operating at supply voltages from 1.8V to 2.5V.

(d) 100 measurements of y-axis acceleration for ADXL362 at 2.2V. Fails normality test (Gaussian overlaid).

Figure 4. Distributions of sensor readings differ across sensor modalities and across integrated circuit implementations and vary with supply voltage.

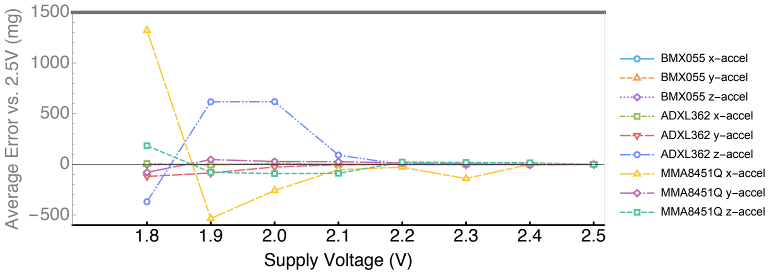


Figure 5. Acceleration inaccuracy (difference in value versus value when supply voltage is at the nominal 2.5 V). The nine data series in the plots are acceleration readings across three axes (*x*, *y*, and *z*) of the three accelerometers in Warp.

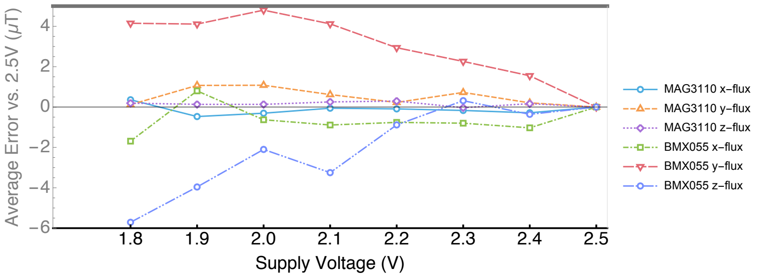


Figure 6. Magnetic flux inaccuracy (difference in value versus value when supply voltage is at the nominal 2.5 V). The six data series in the plots are angular rate readings across three axes (*x*, *y*, and *z*) of the two magnetometers in Warp.

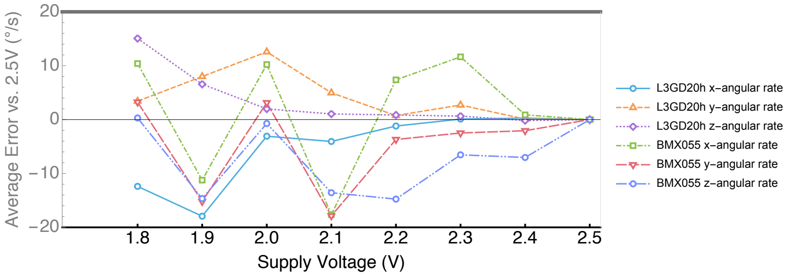


Figure 7. Angular rotation rate inaccuracy (difference in value versus value when supply voltage is at the nominal 2.5 V). The six data series in the plots are angular rate readings across three axes (*x*, *y*, and *z*) of the two gyroscopes in Warp.

Sensor accuracy versus voltage tradeoff results

We evaluate the tradeoff between accuracy of sensor data and supply voltage by operating the three different accelerometers and the two different gyroscopes in Warp over a range of supply voltages. For each of the three axes of these five sensors (15 signal dimensions in total), we operate the sensors at one of eight supply voltages uniformly spaced between 1.8V and 2.5V, a total of 120 measurement configurations. We use the Warp platform's on-board programmable voltage regulator subsystem to control these sensor supply voltages.

Because sensor accuracy typically increases with increasing supply voltage, we use this highest voltage as our reference for correctness. In each of the 120 measurement configurations, we compare the average of 100 sensor signal measurements at each of the eight supply voltage settings to an average of 100 sensor measurements when the sensor is operating under identical conditions but at a nominal supply voltage of 2.5V.

Figure 4 shows examples of the distributions of values from two of the 15 signal dimensions we studied. Figure 4(a) shows the distributions of z-axis magnetic flux values returned by the BMX055 magnetometer, in a fixed orientation, as we change the supply voltage of the sensor from 1.8 V to 2.5 V. Figure 4(b) shows the distribution of one of these eight distributions (sensor values measured at 2.5 V). We overlay a histogram of random variates drawn from a Gaussian distribution with the same mean and variance to provide a visual indicator of the distance of the measured variation from a Gaussian distribution. We also test for normality numerically: The null hypothesis that the data is distributed according to the Gaussian with the same mean and variance as the sample is not rejected at the 5% level based on the Cramer-von Mises test.

Noise distributions and error models often play a role in techniques for approximate computing. In the absence of quantitative measurements such as those in Figure 4(b), researchers today have no choice but to make assumptions about noise distributions. Typical assumptions include uniform distributions in space and normal (Gaussian distributions) over repeated measurements. As we show below, such assumptions may be invalid.

Figure 4(c) shows the distributions of y-axis acceleration sensor values obtained from the ADXL362 accelerometer in a fixed orientation, as a function of sensor supply voltage. The distributions in Figure 4(c) show significantly greater separation than those in Figure 4(a) and are distinctly non-Gaussian, as the overlay of the Gaussian with the same mean and variance in Figure 4(d) shows. The null hypothesis that the data is Gaussian with the same mean and variance as the sample is rejected at the 5% level based on the Cramer-von Mises test.

Figure 5, Figure 6, and Figure 7 show that the accelerometers and magnetometers in Warp provide a useable tradeoff between supply voltage (and hence power dissipation) and accuracy with respect to the output at a reference operating voltage (2.5 V in our measurements). The benefit from going from 2.5 V supply down to 1.8 V supply is an 11.8% reduction in dynamic power dissipation

The gyroscopes provide less distinct trend in improving accuracy from higher supply voltage operation. We attribute this observation to the higher variance in the output of the gyros. In our measurements, both the BMX055 and the L3GD20H gyroscopes have high coefficients of variation of over 115%, indicating that the value of the standard deviation across the 100 samples in each measurement set was even larger than the value of the mean.

CONCLUSIONS

Data from embedded sensing systems form the foundation for applications ranging from wearable health monitors to infrastructure monitoring and augmented reality. In many of these sensor-driven systems, energy is severely constrained and techniques to improve energy efficiency or to trade energy efficiency for some other system metric are valuable. Platforms such as Warp provide a foundation for research into employing techniques from approximate computing in low-power embedded systems, and complements existing research platforms targeted at precise execution on RF- scavenged energy[2] or intermittent computing[4].

Warp enables approximate computing research by integrating 21 sensors that reside in a large range of precision, accuracy, and power dissipation tradeoff points, and Warp augments this with custom hardware in the form of programmable I/O pullups and dynamically reconfigurable sensor supply voltages to enable additional efficiency versus accuracy tradeoffs.

By making the design and firmware for Warp publicly available, we hope to foster new experimentation in approximate computing research, and to provide new possibilities for calibrating techniques developed across the system stack, with measurements from real hardware systems.

ACKNOWLEDGEMENTS

This research is supported by an Alan Turing Institute award TU/B/000096 under EPSRC grant EP/N510129/1, by Royal Society grant RG170136, and by EPSRC grants EP/P001246/1 and EP/R022534/1.

REFERENCES

1. J. Bornholt, T. Mytkowicz, and K. S. McKinley. Uncertain⟨T⟩: A First- order Type for Uncertain Data. In *Proceedings of the 19th International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS ’14, pages 51–66, 2014. ACM.
2. M. Buettner, R. Prasad, A. Sample, D. Yeager, B. Greenstein, J. R. Smith, and D. Wetherall. Rfid sensor networks with the intel wisp. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 393–394. ACM, 2008.
3. M. Carbin, S. Misailovic, and M. C. Rinard. Verifying quantitative reliability for programs that execute on unreliable hardware. In *Proceedings of the 2013 ACM SIGPLAN International Conference on Object Oriented Programming Systems Languages & Applications*, OOPSLA ’13, pages 33–52, 2013. ACM.
4. A. Colin, E. Ruppel, and B. Lucia. A reconfigurable energy storage architecture for energy-harvesting devices. In *Proceedings of the Twenty- Third International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS ’18, pages 767–781, 2018. ACM.
5. H. Esmaeilzadeh, A. Sampson, L. Ceze, and D. Burger. Architecture support for disciplined approximate programming. In *Proceedings of the Seventeenth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS XVII, pages 301–312, 2012. ACM.
6. J. Hester and J. Sorber. Flicker: Rapid prototyping for the batteryless internet-of-things. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*, SenSys ’17, pages 19:1–19:13, 2017. ACM.
7. H. Hoffmann, S. Sidiroglou, M. Carbin, S. Misailovic, A. Agarwal, and M. Rinard. Dynamic knobs for responsive power-aware computing. In *Proceedings of the Sixteenth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS XVI, pages 199–212, 2011. ACM.
8. Y. Kim, S. Behroozi, V. Raghunathan, and A. Raghunathan. Axserbus: A quality-configurable approximate serial bus for energy-efficient sensing. In *Low Power Electronics and Design (ISLPED, 2017 IEEE/ACM International Symposium on*, pages 1–6. IEEE, 2017.
9. S. Lee, L. K. John, and A. Gerstlauer. High-level synthesis of approximate hardware under joint precision and voltage scaling. In *Proceedings of the Conference on Design, Automation & Test in Europe*, pages 187– 192. European Design and Automation Association, 2017.
10. A. Lingamneni, K. K. Muntimadugu, C. Enz, R. M. Karp, K. V. Palem, and C. Piguet. Algorithmic methodologies for ultra-efficient inexact architectures for sustaining technology scaling. In *Proceedings of the 9th Conference on Computing Frontiers*, CF ’12, pages 3–12, 2012. ACM.
11. P. Stanley-Marbell. *Warp* hardware designs and baseline firmware, https://github.com/physical-computation/Warp, 2018.
12. A. Sampson, W. Dietl, E. Fortuna, D. Gnanapragasam, L. Ceze, and D. Grossman. Enerj: Approximate data types for safe and general low-power computation. In *Proceedings of the 32Nd ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI ’11, pages 164–174, 2011. ACM.
13. J. R. Smith, A. P. Sample, P. S. Powledge, S. Roy, and A. Mamishev. A wirelessly-powered platform for sensing and computation. In International Conference on Ubiquitous Computing, pages 495–506. Springer, 2006.
14. P. Stanley-Marbell and D. Marculescu. An 0.9x1.2", low power, energy- harvesting system with custom multi-channel communication interface. In DATE ’07: Design Automation and Test in Europe, pages 15 – 20, 2007.

15. P. Stanley-Marbell and M. Rinard. Approximating outside the processor. In *Workshop on Approximate Computing Across the System Stack (WAX'15)*, 2015.
16. P. Stanley-Marbell and M. Rinard. Lax: Driver interfaces for approximate sensor device access. In *15th Workshop on Hot Topics in Operating Systems (HotOS XV)*, Kartause Ittingen, Switzerland, May 2015. USENIX Association.
17. P. Stanley-Marbell and M. Rinard. Reducing serial i/o power in error- tolerant applications by efficient lossy encoding. In *Proceedings of the 53rd Annual Design Automation Conference*, DAC '16, pages 62:1–62:6, 2016. ACM.
18. Stanley-Marbell, Phillip, and Diana Marculescu. "A programming model and language implementation for concurrent failure-prone hardware." In *Proceedings of the 2nd Workshop on Programming Models for Ubiquitous Parallelism, PMUP'06*. 2006.

ABOUT THE AUTHORS

Phillip Stanley-Marbell is an assistant professor in the Department of Engineering at the University of Cambridge. His research focus is on exploiting an understanding of properties of the physical world and the physiology of human perception to make computing systems more efficient. Prior to joining Cambridge, he was a researcher at MIT, from 2014 to 2017. He received his Ph.D. from CMU in 2007, was a postdoc at TU Eindhoven until 2008, was a permanent Research Staff Member at IBM Research—Zurich from 2008 to 2012, and then an engineer at Apple until 2014. Prior to completing his Ph.D., he held positions at Bell-Labs Research (1995, 1996), Lucent Technologies and Philips (1999), and NEC Research Labs (2005). He is a senior member of the IEEE. Contact him at phillip.stanley-marbell@eng.cam.ac.uk.

Martin Rinard is a Professor in the MIT Department of Electrical Engineering and Computer Science and a member of the MIT Computer Science and Artificial Intelligence Laboratory at the Massachusetts Institute of Technology. His research interests have included programming languages, computer security, program analysis, program verification, software engineering, and distributed and parallel computing. Prominent results have included automatic techniques that enable applications to survive otherwise fatal errors and security attacks and techniques that trade off accuracy of end-to-end results in return for increased performance and resilience. He holds a PhD in Computer Science from Stanford University. He is an ACM Fellow and has received many awards including an Alfred P. Sloan Research Fellowship and Distinguished and Best Paper awards from a variety of publication venues. Contact him at rinard@csail.mit.edu.