# *$CO_2CoDe$*: Towards Carbon-Aware Hardware/Software Co-Design for Intermittently-Powered Embedded Systems

Phillip Raffeck
FAU Erlangen-Nürnberg
Erlangen, Germany

Sven Posner
FAU Erlangen-Nürnberg
Erlangen, Germany

Peter Wägemann
FAU Erlangen-Nürnberg
Erlangen, Germany

## ABSTRACT

The upcoming battery-free Internet of Things comes with a major benefit: These systems operate energy-self–sufficiently during runtime by harvesting energy from the environment. However, the promise of thereby achieving sustainability neglects the fact that manufacturing such systems requires substantial carbon resources, eventually embodied into these systems' cradle-to-gate footprint. When developing these hard-/software systems, designers currently have no possibility to make carbon-aware decisions and assess their cross-cutting consequences throughout the system's stack (i.e., hardware, operating system, scheduler, application).

To address these problems, we present *$CO_2CoDe$*, an approach to carbon-aware co-design of embedded systems. In this paper, we exemplify the necessity of carbon-aware co-design by means of the energy-storage (e.g., capacitor) selections in intermittently-powered embedded systems. System designers have a choice of various storage types manufactured from different materials, which influence not only the environmental impact but also operational characteristics (e.g., internal resistances of capacitors). These differences in operational parameters have cross-cutting, system-wide ramifications: For example, specific types of storage have a smaller/higher carbon footprint while likewise making runtime scheduling decisions harder/easier. Our evaluations on a real-world intermittently-powered system with several capacitor types and scheduling approaches validate our optimization objective of co-designing for carbon awareness with the constraints of meeting energy budgets.

## CCS CONCEPTS

• **Hardware → Impact on the environment**; Sensor applications and deployments; Power estimation and optimization; • **Computer systems organization → Embedded and cyber-physical systems**.

## KEYWORDS

intermittent systems, carbon-aware systems, HW/SW co-design

**ACM Reference Format:**
Phillip Raffeck, Sven Posner, and Peter Wägemann. 2024. *$CO_2CoDe$*: Towards Carbon-Aware Hardware/Software Co-Design for Intermittently-Powered Embedded Systems. In *Proceedings of 3rd Workshop on Sustainable Computer Systems (HotCarbon'24).* ACM, New York, NY, USA, 7 pages.


*HotCarbon'24, July 9, 2024, Santa Cruz, CA*


## 1 INTRODUCTION

*Overshooting Resource Budgets.* Systems research has made substantial progress over the last few decades with the precise budgeting of resources at runtime: Regarding energy demand, the upcoming Battery-Free Internet of Things [1] handles energy budgets in the range of picojoules per bit as part of backscatter communication. Regarding timeliness, communication schemes, such as White Rabbit [23], achieve sub-nanosecond data transmission. Zooming out from these computing systems to a global scale, humanity is unfortunately not yet able to meet given resource budgets that stem from the earth's regenerative biocapacity [4, 22, 36]. That is, humanity significantly overshoots given planetary carbon budgets.

*Impact of Computer Systems & Their Embodied Carbon.* Zooming back in on computer systems, these systems also play a non-negligible role in global warming [15]. Their overall carbon footprint is subdivided into *embodied carbon*, which is required for the overall production and recycling of these systems, and *operational carbon*, which originates from the required resources during the systems' runtime. Even in operation-heavy systems like Bitcoin miners, the embodied carbon can make up to 30 % of the total carbon footprint [10]. Several researchers motivate the need to design, implement, and operate computer systems with a focus on the footprint associated with each step of their life cycle [6, 25, 41].

*Energy-Harvesting Systems & Intermittent Operations.* The scope of this paper targets embedded systems that harvest their required energy from the environment, for example, through solar cells, piezoelectric generators, or electromagnetic radiation. While relying on ambient energy, the systems' operational carbon is negligible. Consequently, we focus on their embodied carbon footprint. A major challenge of these systems is the fact that they face intermittent execution due to the missing stable power supply. Because of this intermittent execution, these systems require checkpointing approaches to make progress with the energy available at runtime.

*Problem of Cross-Cutting Constraints.* While techniques exist to assess the carbon footprint, within the scope of life-cycle analysis (LCA), on the hardware level (i.e., chips, R/L/C components, printed circuit boards) [10, 26, 27, 41], we argue that the problem of accurately describing cross-cutting design considerations for developing embedded systems remains unsolved. That is, system designers lack abstractions and tooling support to make carbon-aware decisions during hardware design. Looking at this problem from the reverse perspective, system-software designers miss abstractions for developing code with awareness of the underlying hardware's carbon impact, which misses optimization potential.

*Carbon-Aware Co-Design & $CO_2CoDe$'s Contributions.* In this paper, we examine the cross-cutting design considerations of intermittently powered systems. The central hardware-related research object is the energy storage (i.e., capacitor) of these systems. We illustrate that the used capacitor type (i.e., ceramic, tantalum, aluminum, super capacitors) has implications on the entire system stack (hardware, operating system, scheduler, application). In turn, these types have different environmental footprints, offering carbon-aware optimization potential. To solve these problems, we introduce the *$CO_2CoDe$* approach to tackle carbon-aware hardware/software (HW/SW) co-design for future embedded systems. The goal of *$CO_2CoDe$* is to solve multi-objective decisions within the design space of carbon awareness and power-aware scheduling. In summary, this paper makes the following three contributions:

(1) *Embodied Carbon Awareness (Hardware ↦ Software)*: We examine alternative capacitor types, assess their carbon footprint, and compare their suitability for scheduling tasks in intermittent systems.
(2) *Power- & Energy-Aware Scheduling (Software ↦ Hardware)*: Based on cross-cutting design constraints, we show alternative scheduling approaches to execute tasks under energy budgets on carbon-aware hardware designs.
(3) *Open-Source Prototype & Evaluation*: Our open-source hardware/software prototype allows us to compare different capacitor types along with their implications on the scheduling. Our evaluations show that whole-system awareness enable carbon-aware design decisions.

## 2 PROBLEM STATEMENT

*Whole-System Perspective.* Figure 1 illustrates *$CO_2CoDe$*'s notion of a system stack: From top to bottom, the system executes tasks that use power-consuming devices, such as transmitters or sensors. The runtime schedules the application's tasks according to a resource-aware policy, that is, whether to first execute `Task 1` or `Task 2`. (The evaluation later in Section 4 extends this running example.) Since our goal are intermittently-powered systems, we employ a checkpointing scheme in order to store the system's state prior to a power failure. The checkpoint management is located at the operating-system layer. The OS, in turn, makes use of the hardware, which is equipped with non-volatile memory for checkpointing, state-of-charge assessment, and the energy storage. *$CO_2CoDe$*'s prototype PCB features a capacitor-selection mechanism illustrated on the bottom layer inspired by an existing design [8]. The hardware platform currently uses two types of energy storage, namely, the ceramic-capacitor and supercapacitor type. For evaluation purposes, these types can be configured (by software) and are outlined in the figure with *HW Variant 1 & 2*. The variants' constraints for the whole stack are examined as follows.

*Equivalent Series Resistance & Load-Dependent Voltage Drop.* Unfortunately, the systems' energy storage is not an ideal capacitor $C$. Instead, each capacitor has a non-negligible internal equivalent series resistance (ESR), which is depicted with $R_{ESR}$ in Figure 1. Ruppel et al. [33] emphasize the relevance of accounting for the ESR in intermittent systems. However, to the best of our knowledge, the carbon-related aspect and its implications on the system stack have not yet been addressed. Specifically, different types of

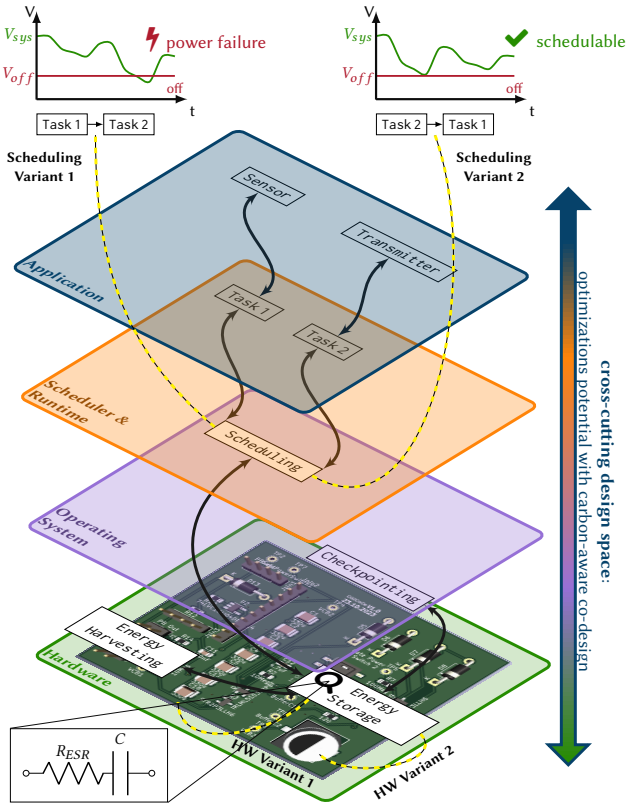


**Figure 1: The (carbon-aware) selection of the energy storage during design time influences the scheduling of tasks.**

capacitors (based on different materials) have different ESRs and, likewise, different embodied carbon footprints. The problem of the ESR rules out scheduling tasks in intermittent systems only based on their *energy* demand (i.e., power over time): In Figure 1, `Task 1` and `Task 2` each have a distinct energy demand. Since the system directly runs from the capacitor with the voltage $V_{sys}$, it faces a power failure when $V_{sys}$ drops below $V_{off}$ (i.e., 2.5 V in our prototype). An ESR-agnostic and only *energy-centric* scheduling policy would assume that the order of scheduling these tasks is irrelevant. However, the ESR causes a voltage drop due to the required *power* of the ESR $P_{sys} = I_{sys}^2 \cdot R_{ESR}$ when the respective task demands the current $I_{sys}$. $V_{sys}$ again increases when the current drawn is reduced. The difference in $V_{sys}$ before and after the task's execution shows the task's energy demand.

*Leakage Resistance.* In addition to the ESR, non-ideal capacitors also have a parallel resistance component. Imperfections due to the materials used or due to the construction of the capacitor manifest as leakage current. Over time, this leakage current causes the capacitor to discharge itself and lose some of its retained energy (i.e., converting it to heat). This self-discharge affects the checkpointing mechanism of the operating system as it relies on accurate state-of-charge assessment. Consequently, capacitor choice potentially complicates forward progress assurances of the whole system.

*Scheduling Variants.* Figure 1 exemplifies two scheduling variants: Task 2 has a higher drawn current than Task 1 and, consequently, a higher ESR-related voltage drop. In the first variant (top-left), Task 1 executes first, and Task 2 follows. Regarding this case, Task 2 faces a power failure ϟ because the ESR-related voltage drop causes $V_{sys}$ to fall below $V_{off}$. In the second variant, Task 2 executes first. Task 1 can be executed subsequently without power failure. Thus, the system is schedulable ✔. Consequently, not only the energy of tasks but also the power is of relevance, which is material-dependent and propagates throughout the entire system stack. In turn, the material dependence as well as the capacity of the energy storage is directly connected to its carbon footprint.

## 3 THE $CO_2CODE$ APPROACH

*Multi-Objective Optimization Problem.* We tackle the outlined problems with the carbon-aware co-design approach $CO_2CoDe$. In $CO_2CoDe$, we formulate a multiple-objective optimization problem within our constrained design space. The related questions and main objectives are twofold:
(#1) How to minimize the *embodied carbon* of the designed system?
(#2) How to maximize the available energy with runtime *scheduling*?

These objectives are intertwined and potentially include contradictory sub-objectives, calling for a hard-/software *co-design* to find a Pareto-optimal solution. Starting from the functional requirements of the application layer, it is the system designers' task to lay out the hard- and software layers to adhere to these requirements in an optimized way. We envision $CO_2CoDe$ to provide expressive abstractions and serve as a holistic tool facilitating that task.

*(#1) Minimizing Carbon Footprint.* We argue that the primary design goal from a non-functional perspective is minimizing the carbon footprint. Given humanity's resource consumption, it is every system designer's responsibility to increase the system's sustainability. Existing LCA tools and their associated databases [5, 12, 16, 17, 28, 35] help to select hardware designs with the lowest carbon footprint (for IoT systems, we refer to a comprehensive overview [25]). However, the use of these LCA tools alone is insufficient as they lack understanding of the effects choices on the hardware level have on the rest of the system stack. Choosing the components with the lowest carbon footprint is inadequate without consideration of other functional design goals. In our case study, we show at the example of capacitors that $CO_2CoDe$'s holistic view is required to make carbon-aware design choices.

*(#2) Maximizing Available Energy.* The primary design goal from a functional perspective in intermittent systems is maximizing the available energy. Due to the dependency on energy harvesting and unreliable harvesting conditions, energy has to be treated as a scarce resource. Wasting energy potentially leads to a lack of energy to complete task execution, threatening the progress of the system as a whole. Naturally, the total capacity of the energy storage has to be sufficiently large to make meaningful progress. Oversizing the energy storage, however, is no solution as capacitors with a larger capacity need more time to charge to the target voltage compared to capacity with a smaller capacity. Higher charging times, in turn, delay execution and, again, threaten the progress of the system. This determines capacity constraints for the capacitor selection.

One aspect of maximizing the energy is minimizing the ESR, as ESR diminishes the amount of useable energy at runtime through heat dissipation and the ESR-related voltage drop. Schedulers must aim to keep the voltage drop above the system's operational threshold ($V_{off}$) to guarantee safe operation, otherwise the system faces a power failure. ESR consideration represents one conflict in system design as it is intertwined with carbon-footprint minimization and energy maximization. Figure 2 illustrates how capacitor selection is an example of a Pareto-optimal design choice. The figure gives an overview of the relationship between ESR and volume for the 500 capacitors with the highest capacity available at DigiKey and actively produced, for each of the shown capacitor types. The values are scaled to a capacity of 1 mF. Capacitors without available data on the ESR are skipped. Notably, no ESR data for ceramic capacitors was available in the data set. However they have low ESR values by design[1]. The black crosses and line mark capacitors on the Pareto-optimal frontier. None of the niobium capacitors lies on the Pareto-optimal frontier. The aluminum-polymer capacitors mark the lower end of the ESR spectrum but require a higher volume. The supercapacitors, on the other hand, show their high energy density but come with the drawback of high ESR values.

A second aspect of maximizing the energy is deriving schedules that make optimal use of the available energy. Analysis approaches enable to statically derive (worst-case) estimates for the energy consumption and power demand of tasks [7, 19, 32, 37, 38]. Similarly, knowledge of the ESR allows the derivation of safe voltage bounds via static analysis [33]. Expressive device models, including the maximum current demand or maximum power demand of device operations [31], enable the static derivation of ESR-related voltage drops from the source code. We propose the notion of ESR-aware scheduling to increase the usable energy by scheduling tasks with high ESR-related voltage drops at higher levels of $V_{sys}$ to mitigate the effects of the voltage drop and avoid crossing the threshold of $V_{off}$. Reconsidering the example given in Figure 1, $CO_2CoDe$ schedules Task 2 first (Variant 2). Independently of the actual ESR values, it always holds that a higher current drawn leads to higher voltage drops. Statically derived estimates on the power demand (and, with that, current demand) of tasks allow the creation of static schedules prioritizing high-current tasks as long as other constraints (e.g., precedence constraints) allow. By scheduling high-current tasks earlier, the effects of voltage drops are diminished. Knowledge about the ESR enables calculating the voltage drop from the static estimates and allows runtime decisions on whether a task can safely be dispatched at the current state of charge. For this, the scheduler has to be aware of the chosen capacitors, linking the question of schedulability to the hardware design.

Another aspect is the leakage resistance of capacitors. If reliable state-of-charge assessment is not possible, continuous checkpointing mechanisms with potentially complex undo or redo operations are required. In turn, this increases the checkpoint-restoration overhead, leading to a higher energy consumption. The choice of capacitor here, too, affects both the power- and energy-aware scheduling.

*Holistic Abstractions.* The current lack of fitting abstractions for cross-cutting whole-system constraints necessitates a fragmentary design approach relying on different abstractions and tools. In the

[1] See, for example, Murata's overview: https://ds.murata.co.jp/simsurfing/mlcc.html

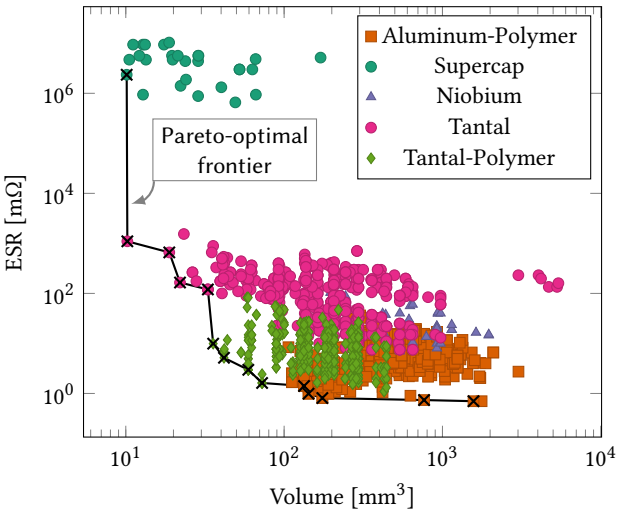


**Figure 2: The ESR of capacitor types over their volume, scaled to a capacity of** 1 mF**.**

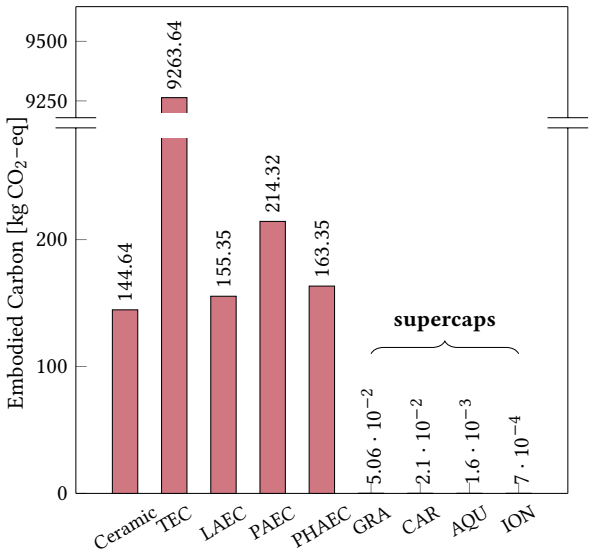


**Figure 3: Comparison of carbon footprint measured in** kg $CO_2$−eq **between capacitor types normalized to 1 Farad.**

following, we present a case study using our prototype platform, underlining the need for a holistic hardware/software co-design approach like *$CO_2$CoDe*. We do not consider leakage resistance in the case study, as the focus is on ESR. We envision *$CO_2$CoDe* to relieve system designers of the burden of manually combining different techniques to solve the multiple-objective optimization problem of carbon-aware system design.

## 4 EVALUATION

*Evaluation (#1): Embodied Carbon.* We assume in this scenario that the target processor and PCB are predetermined, leaving us with the carbon-aware optimization potential of the energy storage. To assess the sustainability of the capacitors used in our prototype system, we compare their embodied carbon in kg $CO_2$−eq. In our current approach, we rely on LCA results for various capacitor types available in the literature: Specifically, we have a look at multi-layer ceramic and tantalum-electrolyte (TEC) capacitors [34], aluminum electrolyte capacitors (liquid (LAEC), polymer (PAEC), and polymer hybrid (PHAEC)) [40], graphene-based (GRA) and activated-carbon-based (CAR) supercapacitors [9], and supercapacitors with an aqueous electrolyte (AQU) and an ionic electrolyte (ION) [21]. As our systems harvest their own energy during operation, we focus on the cradle-to-gate view on the carbon footprint, which includes the emissions of obtaining the raw material and producing the final component.

A challenge with values obtained from different literature works is the differences in the functional unit based on which the embodied emissions are calculated. In our case, we face a wide range from 1 F [9], 5 F [21], 1 000 000 150 µF capacitors [40], and 1 kg of 1 µF capacitors [34]. For comparison, we scale the obtained values to 1 F, and the values are shown in Figure 3.

*Result (#1): Embodied Carbon.* The embodied carbon of tantalum-electrolyte capacitors is significantly larger than that of all the other materials. Ceramic capacitors perform slightly better than all three variants of the aluminum-electrolyte capacitors. With values in the ranges of few g $CO_2$−eq (≈ 0.7 g to 50 g), all variants of supercapacitors offer comparably low embodied carbon. The effect stems mainly from their comparably high *energy density*. The main conclusions for *$CO_2$CoDe*'s co-design from these observations is that the low amount of embodied carbon of the supercapacitors comes at the price of their high ESR. Thus, for our prototype platform, we choose not a single capacitor type but a *combination* in order to balance the drawbacks of the materials: Combining ceramic and supercapacitors provides a mixture of low ESR and low embodied carbon. Ceramic-only design is not possible due to space constraints on the PCB; it would require around 150 capacitors. With *$CO_2$CoDe*'s combination, we balance the trade-off between system-level functionality and carbon awareness.

*Evaluation Platform.* Our custom evaluation board consists of multiple PCBs. A mother board features an ESP32-C3 [14], which comes with integrated sensors (e.g., a temperature sensor) and a combined transceiver for Bluetooth Low Energy (BLE) and Wi-Fi. Additionally, the board provides an external LoRa transceiver connected via SPI. The daughter board depicted in Figure 4 provides two capacitor banks, which buffer the energy from the power source and supply the mother board. One capacitor bank comprises seven 330 µF ceramic capacitors with a combined capacity of 2.31 mF, the other comprises a single supercapacitor with a capacity of 47 mF.

We use a combination of two capacitor types to achieve a reduced embodied carbon footprint while compensating the drawbacks of the chosen capacitor types. Using only a supercapacitor would come with a high ESR value, decreasing the available energy budget. Using only ceramic capacitors, on the other hand, provides very little capacity, impeding the execution of energy-intensive tasks. Using transistor circuits as software-controlled switches allows the configuration of the used capacitor banks at runtime to select either one of the capacitor banks or both combined.

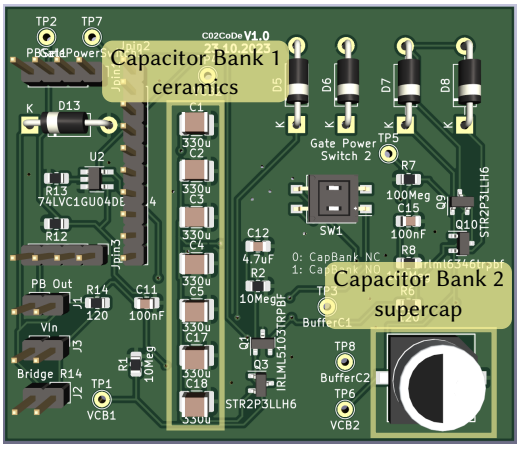


**Figure 4: Custom PCB with two capacitor banks selectable by both software-controlled and hardware switches.**

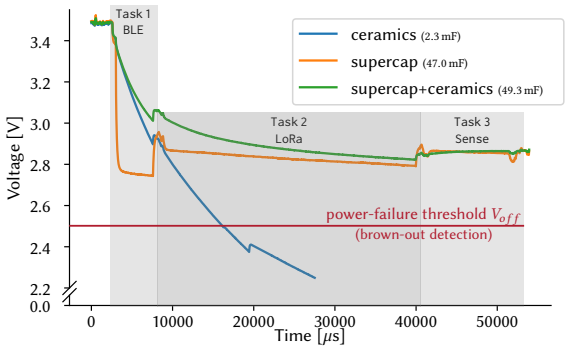


**Figure 5: Voltage trace of the three example tasks with three possible capacitor configurations.**

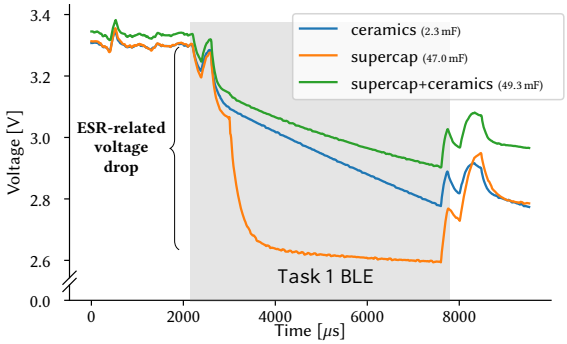


**Figure 6: Voltage trace of BLE transmission (Task 1) with three configurations, highlighting the ESR-related voltage drop.**

We use the reconfiguration mechanism mainly for evaluation purposes to compare workload behavior under different capacitor types. Nonetheless, reconfiguration allows the system to adapt to the current harvesting conditions at runtime. When the voltage is sufficiently high to tolerate all ESR-related effects, for example, only the supercapacitor can be used, letting the ceramics capacitors fully recharge in the meantime.

For this case study, we assume a fixed PCB size and processor with a given carbon footprint. For the evaluation, we only consider the carbon footprint of the used capacitors, ignoring the impact of the selection circuit. Either capacitor type alone comes with drawbacks that jeopardize system functionality. Thus, we argue that the hybrid solution using two capacitor types is beneficial for intermittent computing with devices, although it decreases sustainability due to the higher resource use.

*Evaluation (#2): Scheduling.* Extending the example from Section 2, we now consider three tasks that use different devices: temperature sensing, BLE transmission, and LoRa transmission. The sensing continuously reads the temperature sensor for 10 ms and computes a rolling mean, slightly increasing the current consumption of the system. Due to shortcomings in the BLE module's hardware design (i.e., BLE initialization requires a power-wasting initialization of an entire Wi-Fi stack), the BLE transmission is simulated as 5 ms of constant load via 48 Ω resistance. This energy behavior closely resembles our measurements for actual BLE workloads. The LoRA task transmits data for around 32 ms with varying current consumption but a peak power demand lower than the BLE task. We capture the voltage provided by the capacitors during execution with a JouleScope JS220 [20] measurement unit.

Figure 5 shows a voltage trace over one execution of all three tasks with each of the three possible capacitor-bank selections: ceramic capacitors only (ceramics), supercapacitor only (supercap), and both types of capacitors (supercap+ceramics). The scheduling order follows the descending order of peak power consumption, which translates to the order from highest to lowest ESR-related voltage drop. The difference in available energy is visible, as the ceramics configuration does not have sufficient capacity to execute all tasks. Also noticeable is the difference in the ESR-related voltage drop between supercap and supercap+ceramics, as the addition of the ceramic capacitors significantly reduces the voltage drop.

Figure 6 provides an isolated look at the voltage trace of the execution of only the first task (BLE, `Task 1`). The differences in the ESR-related voltage drop are even more pronounced in this close-up look. For the variant with both capacitor types, the voltage drop is significantly reduced, and the voltage level after execution remains higher. With the smaller voltage drop, the supercap+ceramics variant provides a greater voltage range that is usable for execution. The higher voltage level after the execution implies a lower energy-consumption overhead of the task execution as less energy is lost as heat dissipation due to the ESR.

*Result (#2): Scheduling.* The main result from the observations of Figures 5/6 is that the selection of used materials enables the system to mitigate the effects of the ESR and the ESR-related voltage drop. A co-design approach enables us to derive an optimized scheduling order that makes the most of the available energy and minimizes the ESR while likewise minimizing the embodied carbon footprint.

## 5 RELATED WORK

*System-Level Impact Analysis.* Several existing approaches for determining embodied carbon (in kg $CO_2$–eq) apply their impact analysis on the system level [2, 10, 11, 24, 29, 30, 39]. With *$CO_2$CoDe*, we strive to build upon such analyses to enable the application of carbon-aware co-design of software and hardware. This way, *$CO_2$CoDe*'s goal is to give an impact assessment upfront while also considering the cross-layer effects on functional constraints.

*Architecture-Level Impact Analysis.* Gupta et al. proposed the *ACT* tool [18] with the goal of modeling the embodied carbon footprint on the architectural level of processors. In contrast to their work, with *$CO_2$CoDe*, we target the system-level layer of embedded devices based on the environmental footprint of existing components. We consider the combination of both architecture- and system-level carbon-aware design a promising direction for future work. Having application-level requirements of embedded systems and mapping them in carbon-aware designs to architecture level very likely yields further carbon optimizations. FOCAL [13] presents a parameterized carbon model to design sustainable systems, aiming to make the impacts of design decisions on the embodied and operational carbon footprint explicit. Both FOCAL and *$CO_2$CoDe* thrive to enable the design of systems optimized for their function while simultaneously incurring a minimal carbon footprint. While FOCAL classifies generic design choices according to their carbon footprint, *$CO_2$CoDe* advocates for abstractions that allow application-specific analyses and tailored design decisions.

*Power- & Energy-Aware Scheduling.* Ruppel et al. identified the problematic effects of ESR-related voltage drops on the safety of program schedules [33]. Their proposed Culpeo approach also addresses the issue at the hard-/software interfaces, but unlike *$CO_2$CoDe* has no carbon awareness. The Capybara platform [8] features switchable capacitor banks to compensate the trade-off between capacity and charging time, which is another example of design-time decisions which come with runtime trade-offs. The Pudu approach [32] addresses the influence of devices' energy consumption on the rest of the system, but supports no notion of cross-layer dependencies.

*Comparability of Electronic Designs.* Zhang et al. recently introduced the *DeltaLCA* tool for comparing electronic designs [41]. This work is especially relevant for our endeavors on *$CO_2$CoDe* as it enables the comparison of two different electronic designs. *DeltaLCA* employs domain-specific heuristics to judge the sustainability of hardware designs, including biodegradable materials [3]. However, with its hardware-oriented view, *DeltaLCA* has no notion of how design decisions affect the system software and vice versa. In contrast, with *$CO_2$CoDe*, we explore interdependencies on the hard-/software levels to enable the choice between carbon-aware variants in the design space.

## 6 OUTLOOK & CONCLUSION

In this paper, we made the case that the energy-storage selection in intermittently-powered embedded systems propagates throughout the whole system stack. Selecting storage with the lowest carbon footprint has cross-cutting impacts on the software, which, in turn, can make scheduling decisions that account for the underlying hardware. We presented a case study for capacitor selection, illustrating these impacts in a real-world example using a hybrid solution with two capacitor types, which offers better schedulability but decreased sustainability. Due to the current lack of proper abstractions, it involved a fragmentary approach with manual interoperation of unrelated abstractions and tools.

Based on the findings for our current *$CO_2$CoDe* prototype, our directions for future work are twofold: Firstly, we strive to identify more design decisions that propagate through the system-software stack (e.g., leakage parameters of energy storage), which marks the material- or hardware-related direction. Secondly, we target more expressive software abstractions in order to make carbon-aware scheduling decisions, outlining the software-related direction. Based on these explorations, we envision a hardware/software co-design framework that eventually allows us to make more system-wide, carbon-conscious decisions during the design phase.

This paper acts as a step towards comprehensive system-level hardware/software co-design for embedded systems. With our work on *$CO_2$CoDe*, we want to contribute to more sustainable designs of carbon-aware and battery-free systems. We envision that tooling for carbon-aware hardware/software co-design allows us to bridge the gap between system-level functionality and carbon-awareness to achieve the multi-disciplinary goal of more sustainable systems that are provably carbon-minimal under application constraints.

## ACKNOWLEDGMENTS

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project number 502615015, ResPECT; project number 502947440, Watwa.

*$CO_2$CoDe's hard- & software are publicly available:*
https://gitos.rrze.fau.de/co2code

## REFERENCES

[1] Saad Ahmed, Bashima Islam, Kasim Sinan Yildirim, Marco Zimmerling, Przemysław Pawełczak, Muhammad Hamad Alizai, Brandon Lucia, Luca Mottola, Jacob Sorber, and Josiah Hester. 2024. The Internet of Batteryless Things. *Commun. ACM* 67, 3 (Feb. 2024), 64–73. https://doi.org/10.1145/3624718

[2] Anders S. G. Andrae, Mengjun Xia, Jianli Zhang, and Xiaoming Tang. 2016. Practical Eco-Design and Eco-Innovation of Consumer Electronics–the Case of Mobile Phones. *MPDI Challenges* 7, 1 (2016). https://doi.org/10.3390/challe7010003

[3] Vicente Arroyos, Maria L K Viitaniemi, Nicholas Keehn, Vaidehi Oruganti, Winston Saunders, Karin Strauss, Vikram Iyer, and Bichlien H Nguyen. 2022. A Tale of Two Mice: Sustainable Electronics Design and Prototyping. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI EA '22)*. Article 263, 10 pages. https://doi.org/10.1145/3491101.3519823

[4] Linus Blomqvist, Barry W. Brook, Erle C. Ellis, Peter M. Kareiva, Ted Nordhaus, and Michael Shellenberger. 2013. Does the Shoe Fit? Real versus Imagined Ecological Footprints. *PLOS Biology* 11, 11 (Nov. 2013), 1–6. https://doi.org/10.1371/journal.pbio.1001700

[5] Brightway Developers. 2024. *Brightway LCA Software Framework.* https://brightway.dev/

[6] Tingyu Cheng, Gregory D. Abowd, Hyunjoo Oh, and Josiah Hester. 2023. Transient Internet of Things: Redesigning the Lifetime of Electronics for a More Sustainable Networked Environment. In *Proceedings of the 2nd Workshop on Sustainable Computer Systems (HotCarbon '23)*. 18:1–18:8. https://doi.org/10.1145/3604930.3605723

[7] Hari Cherupalli, Henry Duwe, Weidong Ye, Rakesh Kumar, and John Sartori. 2017. Determining Application-Specific Peak Power and Energy Requirements for Ultra-Low-Power Processors. *ACM Transactions on Computer Systems (ACM TOCS)* 35, 3 (2017), 9:1–9:33. https://doi.org/10.1145/3148052

[8] Alexei Colin, Emily Ruppel, and Brandon Lucia. 2018. A Reconfigurable energy Storage Architecture for energy-harvesting Devices. In *Proceedings of the 23rd International Conference on Architectural Support for Programming Languages*

*and Operating Systems (ASPLOS '18)*. 767–781. https://doi.org/10.1145/3173162.3173210

[9] Matteo Cossutta, Viliam Vretenar, Teresa A. Centeno, Peter Kotrusz, Jon McKechnie, and Stephen J. Pickering. 2020. A comparative life cycle assessment of graphene and activated carbon in a supercapacitor application. *Journal of Cleaner Production* 242 (Jan. 2020). https://doi.org/10.1016/j.jclepro.2019.118468

[10] Ludmila Courtillat-Piazza, Thibault Pirson, Louis Golard, and David Bol. 2024. A Cradle-to-Gate Life Cycle Analysis of Bitcoin Mining Equipment Using Sphera LCA and ecoinvent Databases. *arXiv preprint arXiv:2401.17512* (2024). https://doi.org/10.48550/arXiv.2401.17512

[11] Lucie Domingo, Damien Evrard, Fabrice Mathieux, and Guillaume Moenne-Loccoz. 2011. Synergico: a new "Design for Energy Efficiency" Method enhancing the Design of more environmentally friendly Electr(on)ic Equipments. In *Glocalized Solutions for Sustainability in Manufacturing*. Springer, 148–153. https://doi.org/10.1007/978-3-642-19692-8_26

[12] ecoinvent. 2024. *Life Cycle Assessment - ecoinvent.* https://ecoinvent.org/life-cycle-assessment/

[13] Lieven Eeckhout. 2024. FOCAL: A First-Order Carbon Model to Assess Processor Sustainability. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS '24)*, Vol. 2. 401–415. https://doi.org/10.1145/3620665.3640415

[14] Espressif Systems 2023. *ESP32-C3 Series Datasheet.* Espressif Systems. https://www.espressif.com/sites/default/files/documentation/esp32-c3_datasheet_en.pdf

[15] Charlotte Freitag, Mike Berners-Lee, Kelly Widdicks, Bran Knowles, Gordon S Blair, and Adrian Friday. 2021. The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations. *Patterns* 2, 9 (Aug. 2021). https://doi.org/10.1016/j.patter.2021.100340

[16] Sphera (GaBi). 2024. *Life Cycle Assessment Software.* https://sphera.com/life-cycle-assessment-lca-software/

[17] GreenDelta GmbH. 2022. *openLCA: The source for LCA data sets.* https://nexus.openlca.org/databases

[18] Udit Gupta, Mariam Elgamal, Gage Hills, Gu-Yeon Wei, Hsien-Hsin S. Lee, David Brooks, and Carole-Jean Wu. 2022. ACT: designing sustainable computer systems with an architectural carbon modeling tool. In *Proceedings of the 49th Annual International Symposium on Computer Architecture (ISCA '22)*. 784–799. https://doi.org/10.1145/3470496.3527408

[19] Ramkumar Jayaseelan, Tulika Mitra, and Xianfeng Li. 2006. Estimating the Worst-Case Energy Consumption of Embedded Software. In *Proceedings of the 12th Real-Time and Embedded Technology and Applications Symposium (RTAS '06)*. 81–90. https://doi.org/10.1109/RTAS.2006.17

[20] Jetperch LLC. 2022. *Joulescope JS220 User's Guide Precision DC Energy Analyzer.* https://download.joulescope.com/products/JS220/JS220-K000/users_guide/ Revision 1.3.

[21] A. Kamal Kamali, Edis Glogic, Nilanka M. Keppetipola, Guido Sonnemann, Thierry Toupance, and Ludmila Cojocaru. 2023. Prospective Life Cycle Assessment of Two Supercapacitor Architectures. *ACS Sustainable Chemistry & Engineering* 11, 44 (Oct. 2023), 15898–15909. https://doi.org/10.1021/acssuschemeng.3c04007

[22] David Lin, Leopold Wambersie, and Mathis Wackernagel. 2023. *Estimating the Date of Earth Overshoot Day 2023.* Technical Report. Global Footprint Network. https://www.overshootday.org/content/uploads/2023/06/Earth-Overshoot-Day-2023-Nowcast-Report.pdf

[23] Maciej Lipiński, Tomasz Włostowski, Javier Serrano, and Pablo Alvarez. 2011. White rabbit: A PTP application for robust sub-nanosecond synchronization. In *Proceedings of the IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS '11)*. 25–30. https://doi.org/10.1109/ISPCS.2011.6070148

[24] Pol Maistriaux, Thibault Pirson, Maxime Schramme, Jérôme Louveaux, and David Bol. 2022. Modeling the Carbon Footprint of Battery-Powered IoT Sensor Nodes for Environmental-Monitoring Applications. In *Proceedings of the 12th International Conference on the Internet of Things (IoT '22)*. 9–16. https://doi.org/10.1145/3567445.3567448

[25] Thibault Pirson. 2023. *Environmental perspectives and limits for the Internet of Things in the Anthropocene: going beyond systematic Life-Cycle Assessment.* Ph. D. Dissertation. UCL-Université Catholique de Louvain.

[26] Thibault Pirson and David Bol. 2021. Assessing the embodied carbon footprint of IoT edge devices with a bottom-up life-cycle approach. *Journal of Cleaner Production* 322 (Nov. 2021). https://doi.org/10.1016/j.jclepro.2021.128966

[27] Thibault Pirson, Thibault P. Delhaye, Alex G. Pip, Grégoire Le Brun, Jean-Pierre Raskin, and David Bol. 2023. The Environmental Footprint of IC Production: Review, Analysis, and Lessons From Historical Trends. *IEEE Transactions on Semiconductor Manufacturing* 36, 1 (Feb. 2023), 56–67. https://doi.org/10.1109/TSM.2022.3228311

[28] Pré Sustainability. 2024. *SimaPro LCA software for informed changemakers.* https://simapro.com/

[29] Marina Proske, Christian Clemm, and Nikolai Richter. 2016. Life cycle assessment of the Fairphone 2. *Berlin: Fraunhofer IZM* (2016).

[30] Marina Proske, David Sánchez, Christian Clemm, and Sarah-Jane Baur. 2020. Life cycle assessment of the Fairphone 3. *Berlin: Fraunhofer IZM* (2020).

[31] Phillip Raffeck, Johannes Maier, and Peter Wägemann. 2024. WoCA: Avoiding Intermittent Execution in Embedded Systems by Worst-Case Analyses with Device States. In *Proceedings of the 25th ACM SIGPLAN/SIGBED International Conference on Languages, Compilers, and Tools for Embedded Systems (LCTES '24)*. 83–94. https://doi.org/10.1145/3652032.3657569

[32] Emily Ruppel. 2022. *Peripheral and Power Management in Batteryless, Energy-Harvesting Systems.* Ph. D. Dissertation. Carnegie Mellon University.

[33] Emily Ruppel, Milijana Surbatovich, Harsh Desai, Kiwan Maeng, and Brandon Lucia. 2022. An Architectural Charge Management Interface for Energy-Harvesting Systems. In *Proceedings of the 55th IEEE/ACM International Symposium on Microarchitecture (MICRO '22)*. 318–335. https://doi.org/10.1109/MICRO56248.2022.00034

[34] Lucy Smith, Taofeeq Ibn-Mohammed, SC Lenny Koh, and Ian M Reaney. 2018. Life cycle assessment and environmental profile evaluations of high volumetric efficiency capacitors. *Applied Energy* 220 (June 2018), 496–513. https://doi.org/10.1016/j.apenergy.2018.03.067

[35] Umberto. 2024. *Life Cycle Assessment.* https://www.ifu.com/umberto/lca-software/

[36] Mathis Wackernagel, David Lin, Mikel Evans, Laurel Hanscom, and Peter Raven. 2019. Defying the Footprint Oracle: Implications of Country Resource Trends. *Sustainability* 11, 7 (April 2019). https://doi.org/10.3390/su11072164

[37] Peter Wägemann, Christian Dietrich, Tobias Distler, Peter Ulbrich, and Wolfgang Schröder-Preikschat. 2018. Whole-System Worst-Case Energy-Consumption Analysis for Energy-Constrained Real-Time Systems. In *Proceedings of the 30th Euromicro Conference on Real-Time Systems (ECRTS '18)*, Vol. 106. 24:1–24:25. https://doi.org/10.4230/LIPIcs.ECRTS.2018.24

[38] Simon Wegener, Kris K. Nikov, Jose Nunez-Yanez, and Kerstin Eder. 2023. EnergyAnalyzer: Using Static WCET Analysis Techniques to Estimate the Energy Consumption of Embedded Applications. In *Proceedings of the 21th International Workshop on Worst-Case Execution Time Analysis (WCET '23)*. 9:1–9:14. https://doi.org/10.4230/OASIcs.WCET.2023.9

[39] Eric D. Williams, Robert U. Ayres, and Miriam Heller. 2002. The 1.7 kilogram microchip: Energy and material use in the production of semiconductor devices. *Environmental Science & Technology* 36, 24 (Oct. 2002), 5504–5510. https://doi.org/10.1021/es025643o

[40] Cheng Zhang, Yu Zheng, Junfeng Jing, Yun Liu, and Haihong Huang. 2022. A comparative LCA study on aluminum electrolytic capacitors: From liquid-state electrolyte, solid-state polymer to their hybrid. *Journal of Cleaner Production* 375 (Nov. 2022). https://doi.org/10.1016/j.jclepro.2022.134044

[41] Zhihan Zhang, Felix Hähnlein, Yuxuan Mei, Zachary Englhardt, Shwetak Patel, Adriana Schulz, and Vikram Iyer. 2024. DeltaLCA: Comparative Life-Cycle Assessment for Electronics Design. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)* 8, 1, Article 29 (March 2024), 29 pages. https://doi.org/10.1145/3643561