



Article

Rethinking Power Efficiency for Next-Generation Processor-Free Sensing Devices

Yihang Song , Songfan Li, Chong Zhang , Shengyu Li and Li Lu *

School of Computer Science and Engineering, University of Electronic Science and Technology of China, Qingshuihe Campus, Chengdu 611731, China; songyihang@std.uestc.edu.cn (Y.S.); songfanli@std.uestc.edu.cn (S.L.); zhangchong@std.uestc.edu.cn (C.Z.); lishengyu@std.uestc.edu.cn (S.L.)

* Correspondence: luli2009@uestc.edu.cn

Abstract: The last decade has seen significant advances in power optimization for IoT sensors. The conventional wisdom considers that if we reduce the power consumption of each component (e.g., processor, radio) into μW -level of power, the IoT sensors could achieve overall ultra-low power consumption. However, we show that this conventional wisdom is overturned, as bus communication can take significant power for exchanging data between each component. In this paper, we analyze the power efficiency of bus communication and ask whether it is possible to reduce the power consumption for bus communication. We observe that existing bus architectures in mainstream IoT devices can be classified into either push-pull or open-drain architecture. push-pull only adapts to unidirectional communication, whereas open-drain inherently fits for bidirectional communication which benefits simplifying bus topology and reducing hardware costs. However, open-drain consumes more power than push-pull due to the high leakage current consumption while communicating on the bus. We present *Turbo*, a novel approach introducing low power to the open-drain based buses by reducing the leakage current created on the bus. We instantiate *Turbo* on I²C bus and evaluate it with commercial off-the-shelf (COTS) sensors. The results show a 76.9% improvement in power efficiency in I²C communication.



Citation: Song, Y.; Li, S.; Zhang, C.; Li, S.; Lu, L. Rethinking Power Efficiency for Next-Generation Processor-Free Sensing Devices. *Sensors* **2022**, *22*, 3074. <https://doi.org/10.3390/s22083074>

Academic Editor: Jinsong Han

Received: 13 March 2022

Accepted: 13 April 2022

Published: 16 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: backscatter sensing devices; power optimization; I²C; processor-free devices

1. Introduction

The last decade has seen significant advances in power optimization for IoT (Internet of Things) sensing devices. Specifically, the idea of passive communication eliminates the need for power-starving carrier generation in wireless communication, and conveys information by backscattering ambient signals, thereby reducing the power consumption of radio components down to microwatts (μW). Based on passive communication, energy budgets for transmitting data to the gateway for further processing, become cheaper than locally processing it at the sensing device. This fact fascinates a number of interesting researches to offload the processor, which is conventionally used to perform functions including sensor control, data processing and communication control at sensing devices, to the IoT gateway [1–3], giving rise to processor-free IoT sensing devices. This trend draws a compelling vision for us—given that various sensor ICs are triggered infrequently, and thus often operate at the μW -level of power [4] or even fully passive [5], the next-generation backscatter sensing devices are expected to have overall ultra-low power consumption and are able to be designed as passive and ubiquitous because all of the components are with μW -level of power. Besides, the processor-free device achieves a distinctive anonymity sensing method, which can prevent the unauthorized third party from revealing the identities of the communication parties [6], as there is no information related to the identity of the user, such as programs, wireless connection settings, and sensing logs.

Nevertheless, improving the power efficiency of such processor-free sensing devices is still challenging, because bus communication can take significant power for exchanging

data between each component. The bus communication represents a data transmission process executed on the digital bus (e.g., the I²C bus and the SPI bus) of the sensing devices. In traditional embedded systems, it is reasonable to suppose that bus communication consumes insignificant power since radios and processors account for the majority of power consumption, which drains at the level of milliwatts (mW). However, we observe that the power consumption of bus communication is increasingly becoming a predictable bottleneck for such processor-free sensing devices. For example, the advanced backscatter radio designs show power consumptions of about 15 μ W [7,8], whereas an I²C bus can eat power of 214 μ W when operates at the standard mode (Section 2.1 in detail), 14.2 \times of the power consumption of backscatter radios, as shown in Figure 1. Similarly, SMBus also consumes hundreds of μ Ws of power, as it requires similar pull-up resistors and supply voltages as I²C [9,10]. The power-consuming bus communication between the major components makes it challenging to further optimize the power consumption of processor-free sensing devices. In addition, the high power consumption of bus communication also constraints the number of sensors equipped on IoT end devices.

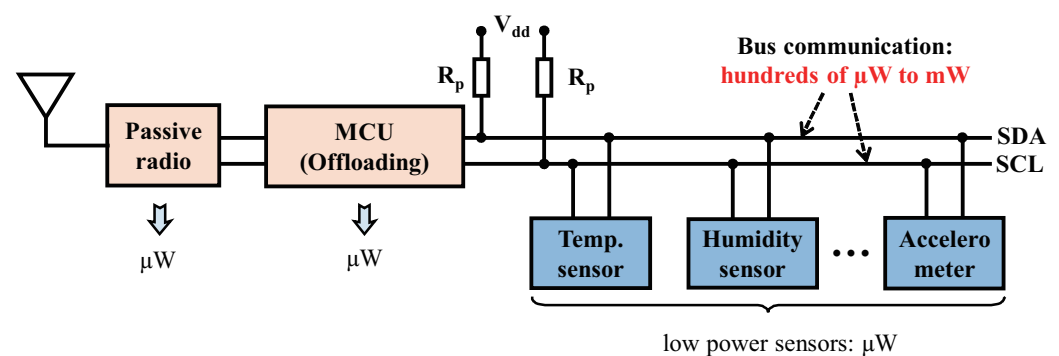


Figure 1. Although backscatter and offloading techniques could let components such as radio and the processor achieve ultra-low power, the bus communication to convey data among those components consumes significant power, which cannot be ignored in such backscatter devices.

In this paper, we rethink the power efficiency for future processor-free IoT sensing devices and ask whether it is possible to reduce the power consumption for bus communication. We observe that existing bus architectures in mainstream IoT devices can be classified into two categories (Section 2.1 in detail). (1) push-pull: although the push-pull architecture features relatively lower power consumption than the other, it only adapts to unidirectional communication, often complicating the bus topology and limiting the scalability of IoT sensing devices. (2) open-drain: the open-drain architecture inherently fits for bidirectional bus communication, thus having more flexibility in connection with sensors. For example, I²C is an open-drain-based bus that can easily connect a series of sensors via only two lines, which benefits simplifying topology in architecture and reducing hardware cost. However, the open-drain architecture has a significant leakage current consumption which leads to hundreds of μ Ws of power. As a conclusion, while there is no perfect bus architecture yet, if the power consumption problem of open-drain can be resolved, this architecture could become the most suitable choice for backscatter devices.

We advocate *Turbo*, a novel approach introducing low power to the open-drain based buses (e.g., I²C, SMBus) by reducing the leakage current created on pull-up resistors R_p during the bus communication. We instantiate *Turbo* on I²C bus to achieve a concrete illustration. To limit the current, the basic way relies on increasing R_p resistance, thereby limiting the current. The higher resistance, however, incurs worse signal distortion and may corrupt the communication. The I²C standard states a range of the resistance to limit the distortion, and thus restricts the maximum resistance. We show that the leakage consumption in many cases is significant yet, in spite of the selection of maximum resistance.

To break above barrier, theoretically, we can leverage a signal waveshaping module that reforms the waveform of distorted signals to be regular. In this way, a higher resistance that breaks the constraint of the maximum resistance value can be chosen to further lower the leakage consumption. The consequent signal distortion resulted from the high resistance is corrected by the waveshaping module before to be received by the target sensor IC. However, the waveshaping module introduces an extra challenge and prevents the I²C communication. The bus lines are specified to be bidirectional that transmits and receives signals on the same line. Oppositely, the desired waveshaper is an unidirectional function block and is only able to receive signals. Therefore, the waveshaper has to be placed on the Rx branch that is at inside of the chip. To implement the waveshaper, the design of the bus interfaces inside chips has to be modified for adding the waveshaper, which results in incompatible with existing chips.

We handle the challenge by our observation that the Rx data register in existing bus interfaces has a concealed function, in which the basic storage elements can be treated as the waveshaping module. Thus, the observation exempts us from changing the designs of existing sensors and allows us to achieve *Turbo* with the off-the-shelf sensors. The only change is the need of increasing the pull-up resistance on the I²C bus. While *Turbo* brings the dawn of low power I²C communication, two key questions still remain.

- First, the signal distortion becomes worse along with the increasing of resistance. Is the communication still robust with *Turbo*? At a certain resistor, the signal may not be effectively recovered by the waveshaping module so that I²C communication fails. How to decide a resistance selection range in *Turbo*?
- Second, while the power consumption decreases along with the increasing of resistance, we observe that the communication data rate is reduced as well. Why does the data rate decrease? What is the relationship between the power consumption and the data rate and how to formulate it? How to select an adequate resistor value to make a trade-off between the power consumption and the data rate?

We propose three technical solutions to address the above challenges.

- First, we determine the effective resistance selection range (Section 3.1) by investigations to analyze the signal distortions over different resistances.
- Second, we build a physical signal model to describe the relationship between the power consumption (Section 3.3) and the data rate (Section 3.2). The model is based on a heuristic method and consider physical signal features. We also find the reason of the data rate reduction, i.e., the higher resistance incurs the increase of required time for each bit data transmission.
- Third, we formulate an optimal resistance problem (Section 3.4) to seek the best solution to reduce the power consumption as much as possible, meanwhile sacrificing the data rate as less as possible. At the optimal resistance, the bit energy budget is minimized in I²C communication.

We conduct experiments to verify the feasibility of *Turbo* with off-the-shelf sensors. Then, we evaluate *Turbo* by both single-sensor and multi-sensors experiments, in which 1.6 megabytes of data are transmitted to each of the four sensors. The results show that *Turbo* inherits robust communication from the standard. *Turbo* achieves 0% bit error rate (BER) within the effective resistance selection range. Moreover, when the optimal resistance is selected, 76.9% of the bit energy budget is reduced on average while sacrificing 64.5% of the data rate. This demonstrate that although the bit rate is decreased, the total energy consumption to perform I²C communication is reduced.

The contributions of *Turbo* are in three folds.

- We first point out the problem of high power consumption of open-drain bus communication, which is widely used in low-power sensing devices, and we propose the basic idea of reducing leakage current of the open-drain architecture to reduce the energy overhead.

- We propose technical solutions to the challenges hindering the realization of our basic idea, including signal distortion, and the reduction of data rate.
- We verify the feasibility and evaluate the performance of *Turbo* with both hardware prototype and simulation.

2. Preliminary

2.1. Background of Bus Architecture

To better understand *Turbo*, first we introduce two types of output architectures that are commonly used in bus communication, then we illustrate the leakage current in *open drain*-based buses.

The *push-pull* architecture leverages a typical CMOS configuration, as is shown in Figure 2a. Each CMOS acts as a single-pole double-throw (SPDT) switch, which is driven by the transmitting data. When the logic level of data is high, the NMOS is turned on and connects the V_{out} to V_{dd} (logic high). When the logic level of data is low, the PMOS is turned on, and then the V_{out} is connected to ground and becomes logic low. However, two push-pull outputs cannot be connected together, since the current will flow freely from V_{dd} to ground if one output is high and the other is low. Thus, the *push-pull* architecture cannot achieve bidirectional communication.

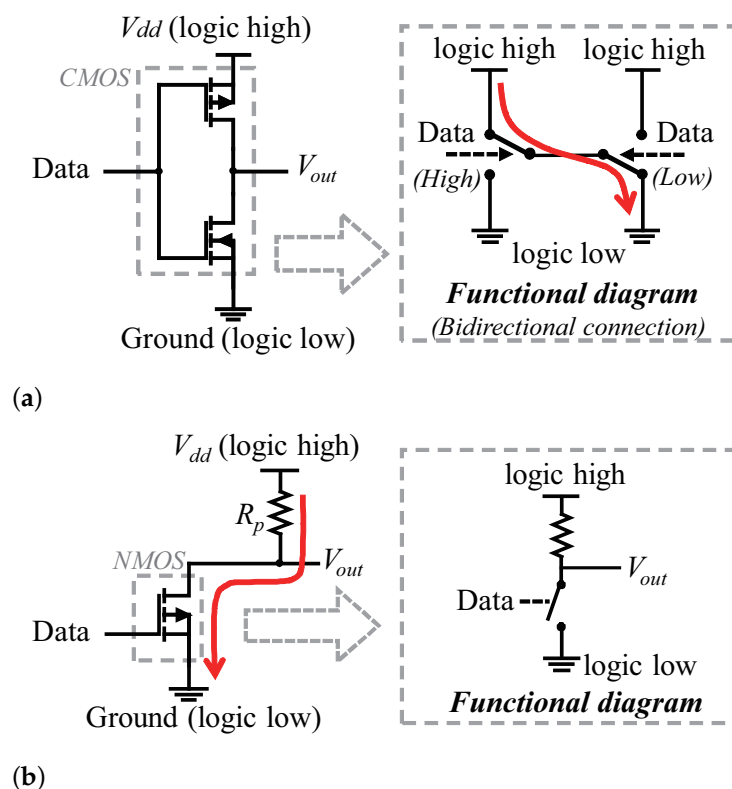


Figure 2. *Push-pull* and open-drain architectures. (a) The *push-pull* architecture. The CMOS act as a single-pole double-throw switch to connect the V_{out} to either V_{dd} or ground. When two *push-pull* outputs are connected, it is possible to cause a short circuit if one of the outputs is high and the other is low. (b) The open-drain architecture. The NMOS transistor acts as a switch. The leakage current (red solid arrow) occurs when the switch is closed and V_{out} is logic low.

The open-drain architecture consists of an NMOS transistor and a pull-up resistor R_p , as shown in Figure 2b. In open-drain, the transistor is driven by the data and works like a single-pole single-throw (SPST) switch. If the switch is open (disconnected), the output V_{out} gets pulled up to V_{dd} through the resistor R_p and begins to be logic high. If the switch is closed, the output is connected to the ground and becomes logic low, and due to the

existence of the pull-up resistor, the open-drain structure will not cause a short circuit when used for bidirectional communication.

However, the open-drain-based buses suffer from a leakage current problem, which increases their power budget of data transmission. Let us take I²C bus as an example to illustrate. I²C bus exploits the open-drain architecture in both SDA and SCL bus lines to realize bidirectional communication regardless of the master or slave roles. Nevertheless, the open-drain output brings the concern of notable leakage current when the output is logic low, as there is current which flows through the resistor R_p , and then the switch to ground. For example, a typical configuration ($R_p = 3.3 \text{ k}\Omega$ and $V_{dd} = 3.3 \text{ V}$) of I²C used on conventional embedded systems leads to a power consumption of more than 3 mW; further, on recent backscatter sensing devices such as WISP [11], the power consumption of I²C bus communication can be reduced to hundreds of μW . However, this is far from realizing μW -level overall power consumption on backscatter sensing devices. According to our experiment, the I²C bus communication on WISP ($R_p = 10 \text{ k}\Omega$ and $V_{dd} = 1.8 \text{ V}$) consumes 51.41 μJ to transmit 2 KB data with this configuration, yielding a power consumption of 214 μW , which are 14.2 \times , 42 \times and 7.8 \times of the power consumption of typical backscatter communication techniques [7,8], sensors [4] and simplified local processor functions [3], as shown in Figure 3.

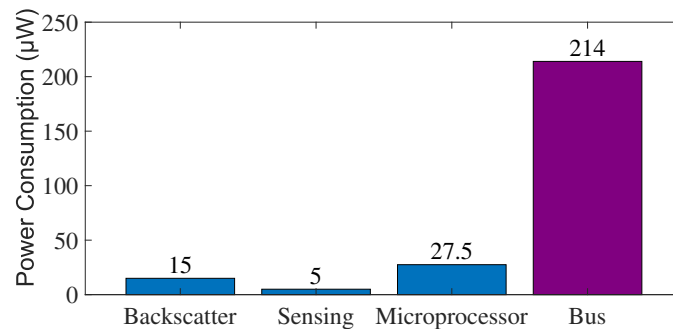


Figure 3. The power consumption comparison of typical components and functions on backscatter sensors, including the backscatter communication, sensors, processor functions (simplified using the R2B technique) and bus communication.

2.2. Challenges of Turbo

Challenge 1—Signal Distortion. To reduce the leakage current, the basic way is to increase the pull-up resistor R_p or lower the positive voltage V_{dd} . As this voltage is directly specified by the sensors connected to the bus and is typically fixed in practical use, the general purpose method is to increase R_p . Higher R_p , however, incurs worse distortion on signals. This is because, in practice, the bus lines and the adjacent circuit unavoidably construct an equivalent parasitic capacitor. The capacitor C_b and the resistor R_p constitute an RC circuit that makes the rise edge of square wave signals slow, and thus deforms the waveform of I²C voltage signals. To prevent the distortion, the I²C standard limits the maximum signal rising time $t_r \leq 1 \mu\text{s}$, which in other words, represents the maximum standard resistance $R_{p(max_std)}$. Empirically, the value of $R_{p(max_std)}$ is around 10 K Ω that often results in hundreds of μA leakage current and is still a huge overhead to the backscatter devices.

Solution. Our insight is that the distorted signals can be recovered by a waveshaping module that converts the slow rise edge to a fast edge, so that the signal waveform meets the time requirement. In this way, the selected R_p can exceed the maximum standard resistance $R_{p(max_std)}$ to further reduce the leakage current while the on-bus sensors can receive standard signal which is being recovered by the waveshaper. The waveshaping module can be implemented by a Schmitt trigger [12]. As shown in Figure 4, the Schmitt trigger contains two thresholds, V_{T+} and V_{T-} , respectively. When the input is higher than the upper threshold V_{T+} , the output of the trigger is $V_H = V_{dd}$. Oppositely, when the input is below the lower threshold V_{T-} , the output is $V_L = 0 \text{ V}$. The dual threshold feature

also improves the noise immunity from a single threshold trigger, so that the distorted waveform can be converted to the closely ideal waveform.

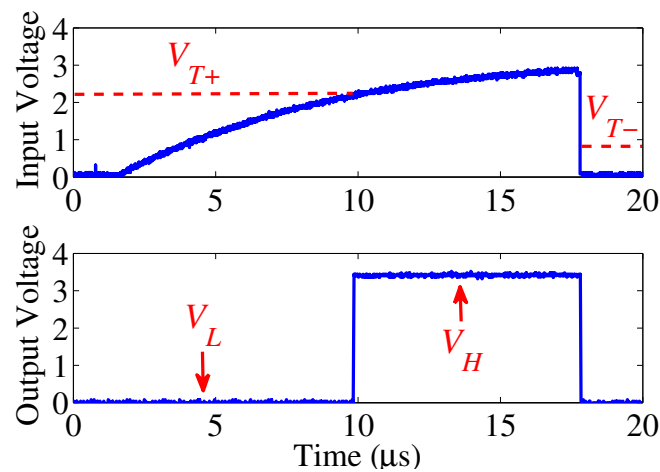


Figure 4. The input voltage is the distorted bus signal and the output is the standard I²C signal that the signal rising time is obviously less than 1 μs.

Challenge 2—Placement Problem. The waveshaping module, however, incurs a consequent problem that it will block the signal transmission on the bus. The I²C bus lines are bi-directional, where all chips receive and transmit signals via single bus line. The placement of the Schmitt trigger on the bus line will divide the line, so that the current from the source cannot pass through the trigger and the Tx circuit is ineffective, as described in Figure 5. The reason stems from the open-drain architecture, in which the output source is the voltage source V_{dd} at the pull-up resistor side, while the transmit (Tx) circuit actually controls whether collecting the voltage or releasing it (imagine that you are operating a water gate at the bottom of an impounding reservoir). Unfortunately, there are no designs for bi-directional triggers that allow current through and can be used in I²C communication to the best of our knowledge. Accordingly, the trigger has to be placed on the receive (Rx) branch. To do this, however, we have to modify the circuit inside the integrated circuit (IC), which incurs *Turbo* incompatible with existing ICs.

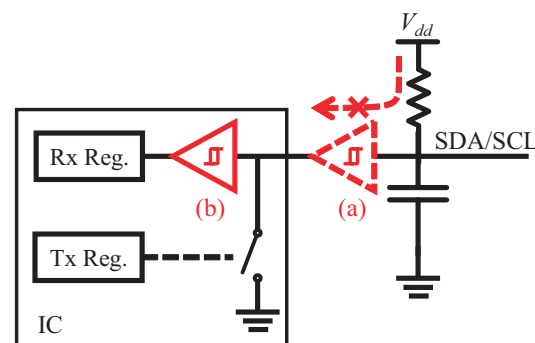


Figure 5. We expect that the waveshaping module is put at the place (a) outside the chip, but the module will block the current so that the chip cannot transmit data to the bus. The placement (b) on the receive branch is adequate, but it requires modifying the chip design.

Solution. We tackle the placement problem according to an observation that existing ICs contain a potential waveshaping function. The arriving signals are directly received by the Rx register inside the ICs. We observe that the register is provided with the function of a Schmitt trigger and has the ability to recognize correct data from distorted signals. While it looks like magic, we observe that the reason is scientific. The register consists of several flip-flops that are used as data storage elements, in which each flip-flop stores a single bit

of data [13], as shown in Figure 6. The kernel of a flip-flop is same to the trigger since they are both the bistable multivibrator [14]. Essentially, the multivibrator contains two NMOS transistors, in which the output of each transistor becomes the input of the other transistor. Each transistor acts as a NOT gate and the output of one transistor finally comes back as its input, which is a typical positive feedback scheme and is the fundamental idea of Schmitt triggers.

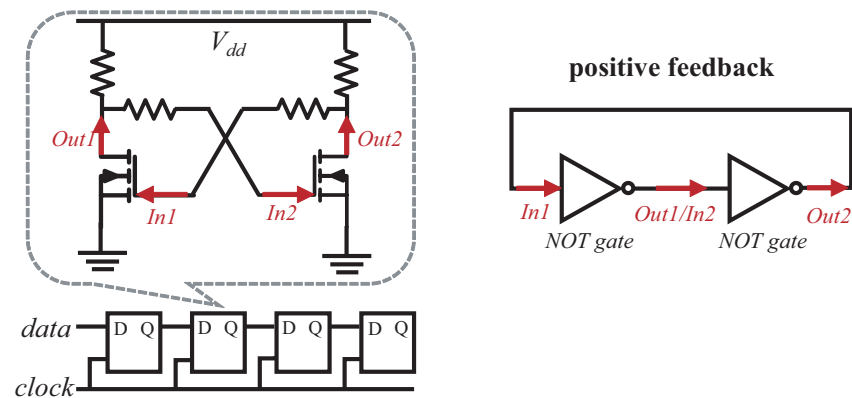


Figure 6. Rx register circuit diagram (4 bit). The kernel of a flip-flop (in the dashed box) is the same as a Schmitt trigger.

Although flip-flops have various types and different implementations [15], the kernel architectures are similar that contain the positive feedback, and hence have the functionality of waveshaping. However, the waveshaping performance of registers would be poorer than the Schmitt trigger because the circuit is optimized for data storage rather than the waveshaping. Fortunately, we observe that many existing ICs have already adopted Schmitt triggers at the input [16–18] for anti-noise applications. Note that their goal of using the trigger is different from *Turbo*. These ICs employ the trigger to improve the noise immunity that supports higher data rate in communication. Conversely, *Turbo* leverages the Schmitt trigger to correct the distorted signals in order to reduce power consumption that results in a lower data rate. In Section 4.3, we evaluate *Turbo* for both types of ICs (i.e., with or without Schmitt triggers).

3. Optimizing the Leakage Current

In this portion, we target the investigation of an effective resistance selection range and build a physical signal model to illustrate the relationship between the power consumption and the data rate over the pull-up resistance R_p .

To better understand the model, we use a heuristic way to illustrate the modeling. We conduct an experiment to observe the I²C communication at different R_p . We consider the experiment on an I²C bus with $C_b = 142$ pF and $V_{dd} = 3.3$ V. A processor which acts as the master sends pre-defined data via the bus to another processor which acts as the slave. Once the data has been sent, the slave checks the data correctness and lights a light-emitting diode (LED) for indicating whether the data transmissions are successfully completed. We repeat the experiment over different R_p and show the observed SCL waveform in Figure 7. We do not show SDA because the open-drain architectures of SDA and SCL are the same, but the SCL is more critical in that it defines whether I²C communication can be performed, while the SDA waveform only represents the transmitted data. The results introduce two key observations to us.

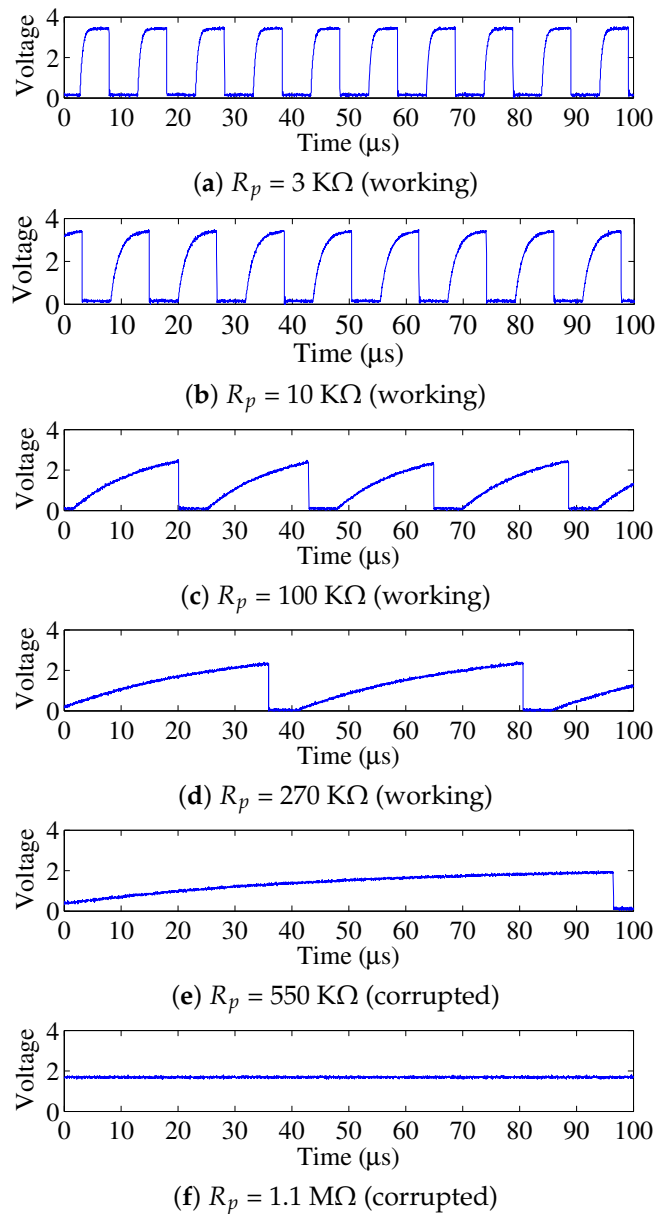


Figure 7. SCL waveform (before the waveshaper) over different pull-up resistances. I²C communication works in case of (a–d), while the communication is corrupted in case (e,f).

3.1. Resistance Selection Range

A selection range of R_p exists in *Turbo*. We observe that the maximum peak voltage V_{max} decreases along with the increase of R_p . For example, V_{max} is 3.3 V, 2.52 V and 2.36 V when the R_p is 10 k Ω , 100 k Ω and 270 k Ω , respectively, in Figure 7. The communication is corrupted when V_{max} is less than the logic high threshold V_{high} . For example, the V_{max} becomes 2 V and 1.72 V when the $R_p = 550 \text{ k}\Omega$ and 1.1 M Ω , respectively, while $V_{high} = 2.31 \text{ V}$. The reduction of V_{max} stems from the reason that the connected sensor ICs on the bus are equivalent to a load resistance R_{load} , as shown in Figure 8. Increasing R_p results in smaller V_{max} since higher R_p divides more voltages. Thus, the value of R_p is restricted by the condition of $V_{max} > V_{high} = 0.7V_{dd}$. With this restriction known, we can get, Equation (1).

$$R_{p(max_std)} < R_p < \frac{3}{7}R_{load} = \frac{3R_{PCB}}{7(1 + R_{PCB} \cdot \sum_{i=1}^n G_i)} \quad (1)$$

where $R_{p(max_std)}$ is the maximum resistance recommended by the I²C standard, $G_i = \frac{1}{R_i}$ refers to the equivalent conductance of the sensor IC i , R_{PCB} is the resistance of PCB, and n is the number of sensor ICs on the bus. According to Equation (1), we also observe that the maximum R_p decreases along with the increase of the number of connected chips.

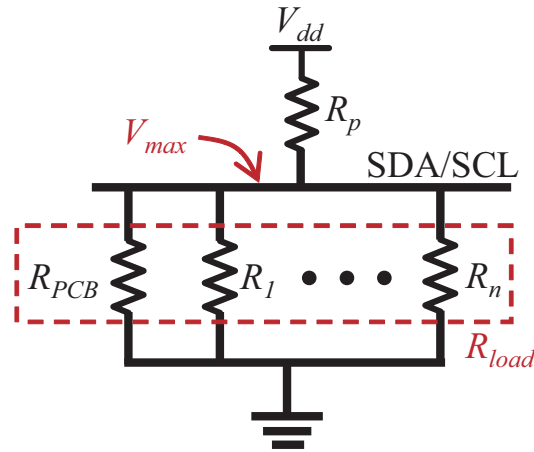


Figure 8. Equivalent circuit of the SDA/SCL line. R_i refers to the equivalent resistance of the i th IC ($i = 1$ to n) when the open-drain states the high impedance. R_{PCB} is related to the materials of making the PCB.

3.2. Data Rate Reduction

We observe that the communication data rate decreases as the duration to transmit each bit increases. For example, the bit rate is 100 Kbit/s, 84.7 Kbit/s, 43.7 Kbit/s and 22.4 Kbit/s, respectively, in Figure 7a–d. This happens because the PCB has equivalent capacitance C_b , and a higher R_p increases the RC value of bus lines and thus the voltage rises slowly. When the output is above the logic high threshold V_{high} and this clock period is passed, the output will be tripped to the logic low voltage after hitting V_{high} . Thus, the time to transmit one bit is formulated as Equation (2).

$$t_{bit} = \underbrace{\frac{1}{2f_{SCL}}}_{t_{low}} + \underbrace{R_p \cdot C_b \cdot \ln\left(\frac{V_{max}}{V_{max} - V_{high}}\right)}_{t_{charge}} + \Delta t \quad (2)$$

where f_{SCL} is the clock frequency on the SCL line, $R_p \cdot C_b \cdot \ln\left(\frac{V_{max}}{V_{max} - V_{high}}\right)$ is the time required for charging the RC circuit, which is composed of the equivalent capacitor of PCB and the pull-up resistor, Δt refers to the duration between the voltage reaching V_{high} and then becoming low. As we observe that Δt is typically very small compared to the clock period, for convenience, Δt can be negligible. Therefore, the maximum bit (data) rate is calculated as:

$$\begin{aligned} B_{rate} &\approx \frac{2f_{SCL}}{1 + 2f_{SCL} \cdot R_p \cdot C_b \cdot \ln\left(\frac{V_{max}}{V_{max} - V_{high}}\right)} \\ &= \frac{2f_{SCL}}{1 - 2f_{SCL} \cdot R_p \cdot C_b \cdot \ln\left(0.3 - 0.7 \cdot \frac{R_p}{R_{load}}\right)} \end{aligned} \quad (3)$$

3.3. Building Signal Model

Based on the two observations above, we build a physical signal model to explore the relationship between the power consumption and the bit rate. We first formulate the energy budget for single bit transmissions and then derive the relationship to the bit rate.

The single bit energy budget consists of the energy consumption on the two lines (SCL and SDA). First, the consumption on the SCL line comprises two portions. One is the

leakage current consumption (the t_{low} part in Figure 9), while the other is the switching dissipation (the t_{charge} part). The switching dissipation E_{spbit} refers to the energy stored in the bus capacitor C_b and further released to the ground. Second, the bit energy budget on the SDA line is related to the bit data. If the data is “bit-0”, the SDA line stays logic low during the bit period and the leakage current stays for a period of t_{bit} . If the data is “bit-1”, the line is charged to logic high and the switching power dissipation of E_{spbit} happens. Therefore, we describe the bit energy budget of “bit-0” in Equation (4).

$$\begin{aligned} E_{bit0} &= P_{leak} \cdot (t_{low} + t_{bit}) + E_{spbit} \\ &= \frac{V_{dd}^2}{R_p} \cdot \left(\frac{1}{2f_{SCL}} + t_{bit} \right) + \frac{1}{2} C_b V_{max}^2 \end{aligned} \quad (4)$$

where the P_{leak} is the power consumption caused by the leakage current. Accordingly, the bit energy budget of “bit-1” is formulated as Equation (5).

$$\begin{aligned} E_{bit1} &= P_{leak} \cdot t_{low} + 2E_{spbit} \\ &= \frac{V_{dd}^2}{2R_p f_{SCL}} + C_b V_{max}^2 \end{aligned} \quad (5)$$

Finally, we collect the bit energy budget to obtain the relationship between the bit rate and power consumption. The relationship is simply that the number of bits multiplied by the bit energy budget, as shown in Equation (6).

$$P_{diss} = B_{rate} \cdot [\alpha E_{bit0} + (1 - \alpha) E_{bit1}] \quad (6)$$

where we use the factor $\alpha \in [0, 1]$ to represent the proportion of “bit-0” in all the transmitted data.

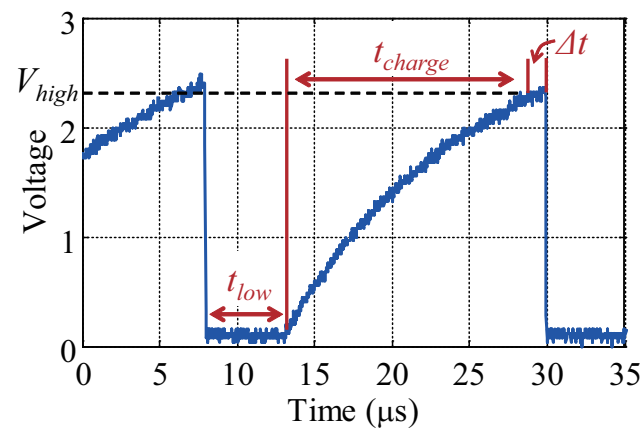


Figure 9. The time for each bit transmission.

3.4. Optimal Resistance Problem

In this portion, we formulate the optimal resistance problem. Solving this problem can help us to find the optimal pull-up resistance in practical applications so as to make a trade-off between the power consumption and the data rate.

From the results in Figure 7, the power consumption and the data rate both decrease over the increasing R_p . The optimal resistance problem specifies finding an optimal R_p value that obtains as much reduction as possible in the power consumption and as little sacrifice as possible to the data rate. We formulate the optimal resistance problem as follows.

$$\text{minimize } f_o(R_p) = \frac{P_{diss}(R_p)}{B_{rate}(R_p)} \quad (7)$$

subject to:

$$\frac{t_r}{\ln \frac{7}{3} \cdot C_b} < R_p < \frac{3}{7} R_{load} \quad (8)$$

Before solving this problem, we should conduct practical measurement to get the function of $P_{diss}(R_p)$ which depends on the fabrication of the practical PCB and the physical characteristics of the specific Sensor ICs on the PCB. The measured plot of $P_{diss}(R_p)$ is shown in the evaluation section (Figure 11e). By solving this problem, the optimal R_p can be selected to minimize the objective function $f_o(R_p)$. Since $f_o(R_p)$ is the rate of the power consumption to the bit rate, the optimal R_p represents the minimum average bit energy budget.

4. Implementation and Evaluation

4.1. Implementation

Thanks to the potential waveshaping function in existing chips, we can easily implement *Turbo* by changing the two pull-up resistors on the bus. Although some I²C master chips have internal pull-up resistors, to implement *Turbo*, we disable the internal resistors and use external resistors. The value of pull-up resistors can be determined by the solution of the optimal resistance problem. Next, the implementation of the signal model and the optimal resistance problem can be simply realized by Matlab codes. We utilize the `fminbnd` function [19] in Matlab Optimization Toolbox to solve the problem. The simple implementation benefits that *Turbo* can be rapidly implemented on existing devices, even if the PCBs have been made. Moreover, the firmware in the devices does not need any modification as *Turbo* is harmless to the I²C standard.

4.2. Evaluation Overview

Our evaluation includes two portions. First, we evaluate the quality of I²C communication with the implementation of *Turbo*. Further, we solve the optimal resistance problem and see how the optimal value is affected by various parameters.

We build two platforms for I²C experiments as shown in Figure 10. First, the mini-test board can be used for simple two-chips I²C experiments that provide the proof-of-concept of *Turbo*. Second, the multi-chips test board faces to the comprehensive evaluations on multi-chips cases. We can solder desired pull-up resistors and optional external bus capacitors to imitate communication circumstances in different devices. Note that the bus capacitance C_b is difficult to be decreased since it is determined by the nature of the bus lines. However, we can add C_b by connecting an external capacitor to the bus. In the multi-chips test board, we employ the digital resistor circuit that enables us adjusting the pull-up resistors digitally by the control of a processor. The digital resistor is mainly composed by the potentiometer, AD5262 [20]. In the evaluation, the I²C buses on the two platforms both work at $V_{dd} = 3.3$ V and $f_{SCL} = 100$ KHz.

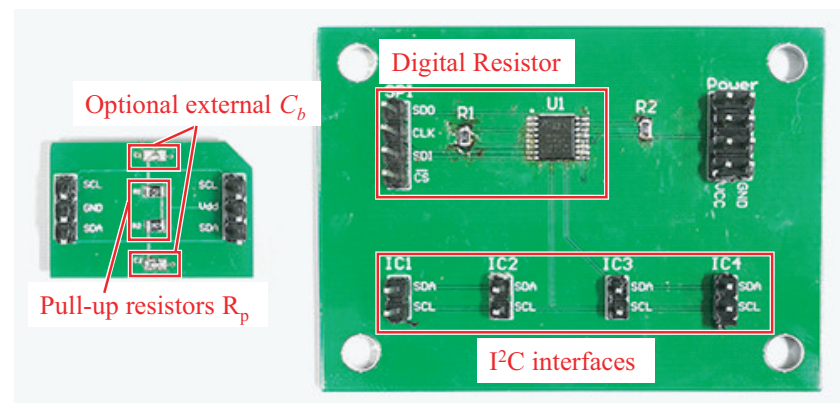


Figure 10. Experiment platforms. The left is a mini-test board, while the right is a multi-chips test board.

4.3. Quality of Communication

We conduct experiments to evaluate the *Turbo* performance, in terms of bit error rate (BER), bit energy budgets (E_{bit0} and E_{bit1}), bit rate (B_{rate}) and bus power dissipation P_{diss} . We consider four off-the-shelf ICs in two types, i.e., with or without Schmitt triggers inside the chip. For each IC, we evaluate the metrics at a range of the pull-up resistance R_p . The parameters of each IC are shown in Table 1. The maximum standard resistance $R_{p(max_std)}$ is calculated according to the standard via the factor C_b that can be obtained with an oscilloscope to observe the signal rising time. Considering *Turbo* in various devices that have different C_b , we add the external bus capacitance 50 pF, 100 pF and 150 pF to the tests for 24AA08, Si7013 and PCF8574, respectively.

Table 1. Experiment parameters in the evaluation and the comparison between the standard and *Turbo*. The symbol “●” represents the IC with the Schmitt trigger, while the symbol “○” indicates the IC without the Schmitt trigger.

Test IC	● Monza [16]	● 24AA08 [17]	○ PCF8574 [21]	○ Si7013 [22]	
C_b	63 pF	142 pF	206 pF	161 pF	
Standard	$R_{p(max_std)}$	18.7 K Ω	8.3 K Ω	5.7 K Ω	7.3 K Ω
	E_{bit0}	7 nJ	15.8 nJ	22.9 nJ	17.9 nJ
	E_{bit1}	3.6 nJ	8.1 nJ	11.8 nJ	9.2 nJ
	B_{rate}	90.9 Kbit/s	90.9 Kbit/s	90.9 Kbit/s	90.9 Kbit/s
Turbo	optimal R_p	200 K Ω	163.4 K Ω	148 K Ω	158.1 K Ω
	E_{bit0}	2.2 nJ	4.1 nJ	5.6 nJ	4.6 nJ
	E_{bit1}	0.7 nJ	1.4 nJ	1.9 nJ	1.5 nJ
	B_{rate}	52.7 Kbit/s	33.4 Kbit/s	17.6 Kbit/s	25.1 Kbit/s

4.3.1. Bit Error Rate

In the experiment, each IC is the slave role in the communication and is controlled by the master processor MSP430F2132 [18] working at 3.3 V supply voltage. The processor sends predefined data to each IC over hundred thousand times. The IC then executes established certain operations according to the data. For example, Monza will respond a fixed register value to the processor. Thus, the processor can evaluate the BER in communication. We repeat this process over different R_p . The results are shown in Figure 11a.

We observe that the BER of each IC polarizes (i.e., 0% or 100%). Specifically, the BER is zero when R_p is less than a threshold, while the BER jumps to 100% (i.e., the case where no data bits are received) once R_p is above the threshold. As we explained, the threshold is determined by the inside waveshaping module. When R_p increases, the maximum peak voltage decreases. If the voltage is below the threshold, the waveshaping module cannot recover the signal and consequently causes communication failures. The thresholds are different among the ICs. For the ICs with Schmitt triggers (e.g., Monza and 24AA08), the thresholds are close since the triggers are compliant to the standard including the voltage definitions. For the ICs without the trigger, the register is optimized to keep data, not for the waveshaping. Thus, the voltage thresholds may not be same with the standard definitions.

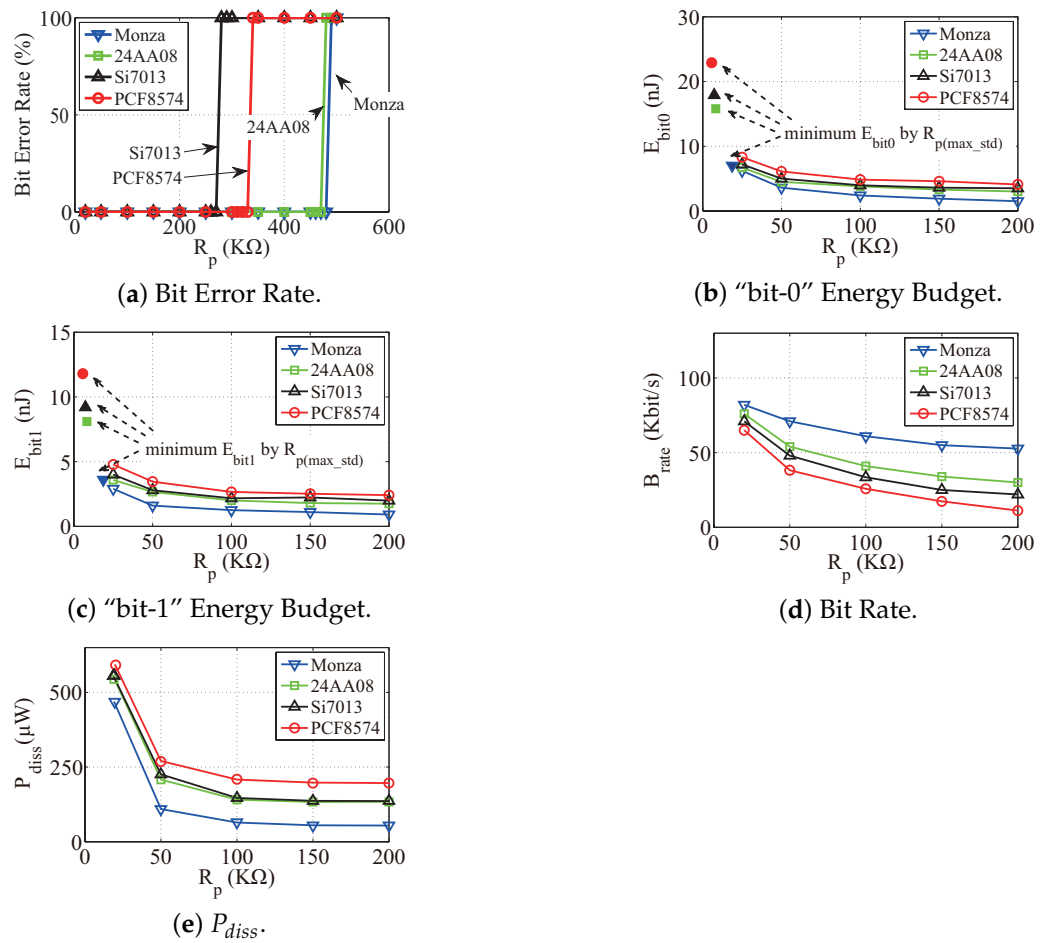


Figure 11. Experiment results of *Turbo*. The BER over different R_p is shown in (a). We use “100% BER” to illustrate the case in which there are no bits (no matter the bits are correct or not) that can be transmitted via bus lines since both the SDA and SCL lines fail to work. The bit energy budget over different R_p is shown in (b,c). We also mark the bit energy budget associated with the maximum standard resistance $R_{p(max_std)}$ as a comparison in (b,c). As the sink rate of energy budgets becomes slow along with R_p , we recommend an empirical value $R_p = 50$ KΩ as a general resistance selection for low power purpose. The maximum bit rates achieved with different R_p values are shown in (d), and the overall power dissipation of I²C bus over different R_p is shown in (e).

4.3.2. Bit Energy Budget

We evaluate E_{bit0} and E_{bit1} by experiments. We measure t_{bit} and V_{max} by an oscilloscope and calculate the energy budget according to Equations (4) and (5), respectively. We repeat the experiment at a range of R_p values. The results are shown in Figure 11b,c, respectively. From the results, we observe that E_{bit0} and E_{bit1} both decrease along with the increase of R_p because the leakage current is reduced. Further, larger C_b incurs higher E_{bit0} and E_{bit1} when R_p keeps constant. The reason stems from the fact that larger C_b implies more switching dissipation. Furthermore, E_{bit0} is higher than E_{bit1} in the same condition of R_p and C_b . This is because the leakage current is typically higher than the switching dissipation.

4.3.3. Bit Rate

This evaluation considers the bit rate B_{rate} associated with varying R_p . The experiment is conducted by measuring t_{bit} by an oscilloscope. Then, the bit rate can be calculated according to Equation (3). We show the results in Figure 11d. From the results, we observe that the bit rate has a negative relationship with both R_p and C_b . This is because

the equivalent RC circuit on the bus will delay the propagation of signals. Specifically, outputting a high level voltage can be treated as a process to charge the bus capacitor.

4.4. Optimal Resistance with Various Parameters

We can select the optimal R_p value by solving the optimal resistance problem in order to obtain the minimum bit energy budget. In this portion, we evaluate the optimal value over varying α (the proportion of “bit-0” in the test data), C_b , R_{load} and V_{dd} , respectively. The evaluation is conducted by simulations on the IC 24AA08 and the basic parameters are shown in Table 2. For each simulation, we only tune a single parameter and keep the others same. Figure 12 shows the simulation results with various parameters.

Table 2. Simulation parameters.

V_{dd}	f_{SCL}	C_b	R_{load}	α	Δt
3.3 V	100 KHz	142 pF	0.76 M Ω	0.5	0

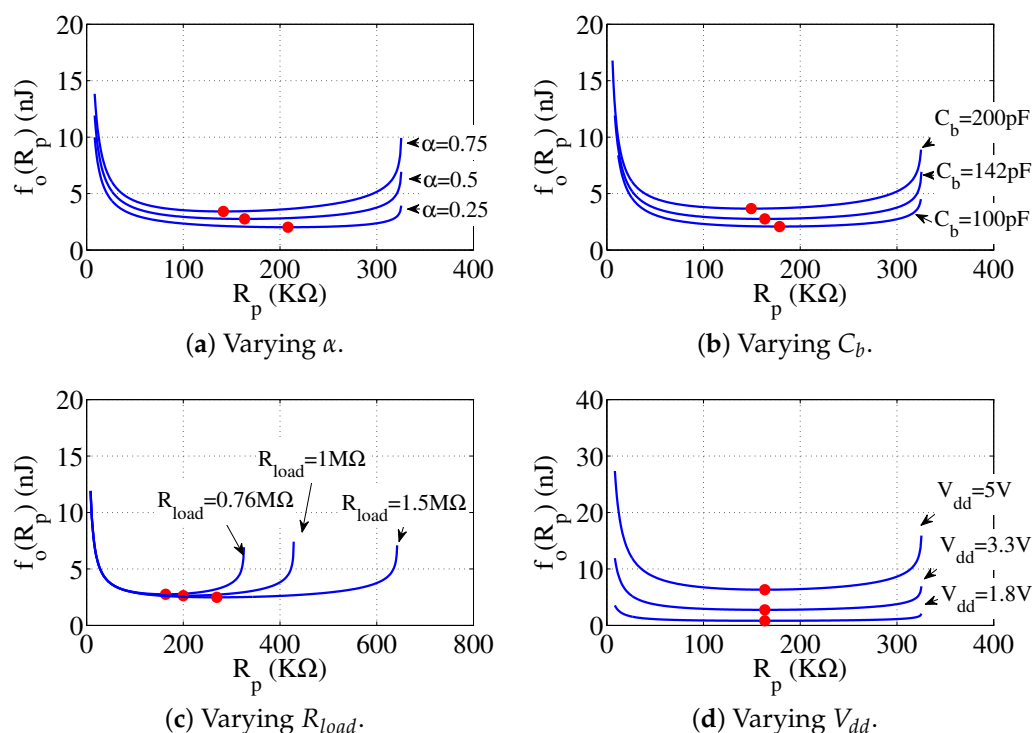


Figure 12. The simulation results of optimal resistance values over different parameters. Each optimal value is marked by a red point. $f_o(R_p)$ represents the average bit energy budget. In (a) and (b), the optimal R_p decreases along with the increase of α and C_b , respectively. In (c), R_{load} does not change the curve on the ordinate and just prolong the curve on the abscissa. In (d), higher V_{dd} will not impact the optimal R_p value and only introduces higher power consumption. As $f_o(R_p)$ is almost same when $R_p = 100$ K Ω to 250 K Ω in (a)–(d), we recommend an empirical value $R_p = 100$ K Ω as a general resistance selection.

From the results, we also observe that the value of $f_o(R_p)$ keeps almost same in a range of R_p , roughly 100 K Ω to 250 K Ω . Hence, we recommend an empirical value $R_p = 100$ K Ω since the bit energy budget is near to the minimum, while the data rate could be much higher than the minimum case. Taking 24AA08 as an example, the average bit energy budget is reduced to 24.3% compared to the maximum standard resistance $R_{p(max_std)}$, meanwhile only sacrificing 54.8% of the bit rate. As a comparison, the bit energy budget and the reduced bit rate are 23% and 63.3%, respectively, when the optimal R_p in Table 1

is selected. The recommended value would benefit engineers selecting a general pull-up resistor value in practical applications.

5. Discussion

Different I²C Speed Modes. The I²C standard supports several speed modes, including the standard mode (up to 100 Kbit/s), the fast mode (up to 400 Kbit/s) and the high-speed mode (up to 3.4 Mbit/s). We mainly discuss and evaluate *Turbo* in the standard mode. We believe that *Turbo* adapts to other modes because the standard specifies that the ICs of higher modes should incorporate the Schmitt triggers at their inputs and thus automatically gain the waveshaping ability.

Reduction of Clock Frequency. In digital circuits, reduction of clock frequency is a basic way to lower power consumption. However, this solution cannot work well because the leakage current is the major power consumption in I²C. Although the clock reduction introduces lower switching dissipation (dynamic power), the reduced clock frequency incurs more leakage consumption due to longer time to keep the bus at logic low that causes static leakage current. In future work, we will attempt to combine the clock reduction and the resistance increment to investigate a way to further reduce the power consumption in I²C communication.

Compatibility to Existing Chips. *Turbo* obeys the I²C standard definitions that the high voltage V_{high} refers to above 70% of V_{dd} and the low voltage V_{low} is below 30% of V_{dd} . In practice, however, we observe that some existing chips do not exactly follow the definitions. For example, the chip MSP430FR5969 [23] typically has $V_{high} = 65\% \cdot V_{dd}$ and $V_{low} = 35\% \cdot V_{dd}$. This is because each chip meets special considerations in the design. Specifically, I²C and another serial bus share a same I/O port on MSP430FR5969, in which the designers have to consider compatibility of the voltage definitions to both buses. The fact may introduce a negative impact that different chips may have different effects when *Turbo* is implemented. We believe that the effect is slight as the voltage definitions in existing chips would not be hugely different away from the I²C standard.

Dynamic Data Rate Demand. The device may require the I²C bus performing high bit rate transmissions in some cases and dynamically adopting low power I²C when the system available energy is weak. However, the maximum bit rate is affected by the value of R_p that is implemented by a resistor in hardware. In order to meet this case, the resistance should be dynamically controlled by software in the processor. To achieve this, the pull-up resistors should be digital resistors or digital potentiometers as shown in Figure 10.

6. Related Work

Passive Sensing Devices. This paper involves sensor devices. The most of research efforts on this topic concentrate on radio to mitigate the energy problem. Traditionally, the radio drains major power consumption (several mWs) in sensing devices, which is much higher than the power consumption regarding bus communication to operate sensors. In the past decade, backscatter communication dramatically reduces the power consumption of the radio down to μ Ws [7,8,24], and a series of efforts are proposed to optimize energy utilization efficiency [25–28], the energy cost of energy-level detection hardware [29], and even eliminate the power-consuming processor from sensing devices [1–3]. Thus, it is possible to make the devices passive with micro-energy harvesting. However, the inefficiency of energy harvesting causes that the energy is still scarce in passive devices [30,31]. However, for a long time, the weight of bus communication power consumption is overlooked. This paper at the first time points out this problem and takes steps to reduce the energy overhead in bus communication to access multiple sensors.

Multi-sensors Connections. In this paper, we advocate the I²C bus to connect multiple sensors. As we know, other general purpose digital buses like SPI can be also used to connect such sensors. However, the I²C bus has its unique advantage in terms of the simple topology on hardware connection, which towers over other buses in multi-sensors case. For example, the SPI bus requires at least $3 + n$ lines (n is the number of sensors), which

occupies additional processor ports that is a constrained resource in passive sensing devices. In addition, I³C [32] has been released as an evolution featuring lower power consumption than I²C. To implement I³C, the master processors have to support the specific functions. The existing processors employed in current passive sensing devices, however, do not support I³C. This means that I³C will not be implementable immediately to the existing devices. Therefore, we choose I²C and we believe that the proposed *Turbo* facilitates existing passive sensing devices with the simple topology and energy optimization simultaneously.

Power Optimization for I²C. Currently, the effective methods to optimize power consumption of I²C stem from the fact that the energy budget to transmit “bit-0” is higher than the energy for “bit-1” as the leakage current consumption is more obvious in “bit-0” transmission.

Disabling Pull-Up Resistors. The forum thread [33] discusses a question what happens if the pull-up resistors are removed. The omitted resistors are equivalent to $R_p = \infty \Omega$ that incurs zero leakage current. While the I²C standard partly allows the omission, it introduces two main problems. First, the approach is applied to a special case that in ultra fast mode, the data is transmitted in only one direction between two chips. Second, the approach is available only for a part of I²C-interfaced chips that the I/O circuit supports the push-pull output architecture. In this paper, the proposed *Turbo* considers the general cases that allow bidirectional communication among multi-chips. Further, the authors in [34] propose an approach using a software I²C implementation which disables pull-ups when the master sends “bit-0”. However, software I²C requires the processor staying active to simulate the protocol, which often incurs higher power consumption. Although this overhead can be mitigated by hardware implementation, it needs to modify the master and results in being not implementable to the existing devices.

Coding Techniques. Another attempt is to encode data transmissions with more “bit-1” than “bit-0”. The coding techniques can be classified into two categories. (1) The software coding [35] which is implemented by software in the ICs. This technique often is employed in data storage that the data can be freely encoded, but is restricted in many scenarios. For example, commands and sensing results in most of the sensors are not admitted to be encoded. In addition, the I²C clock on the SCL line cannot be encoded. (2) *The hardware coding* [36] leverages an encoder/decoder pair which is placed at the transmitter and receiver side, respectively. This technique has the merit of user transparency since the encoded data will be decoded before received, but mostly causes more energy consumption in I²C communication. This is because the original data has to be sent twice from the transmitter to the encoder and from the decoder to the receiver. Differing from *Turbo*, this technique is often used as relays for long distance communication but not for reducing the consumption, and it has higher hardware cost and poorer compatibility than *Turbo*.

Varying Pull-Up Resistors. The technical article [37] shows the signal waveforms associated with a range of R_p values by conducting several experiments. Further, the article analyzes the signal patterns over the increase of R_p . Compared with the experiments in this paper, there are three differences. First, the R_p range in the article is within the range specified by the standard. Second, the article does not discuss neither the over-sized R_p values nor the reasons of waveform effects. Third, the article only discusses the data rate over R_p , but does not take the power consumption into account. In this paper, we concern the I²C power consumption in low power devices. We investigate the waveform effects from not only the standard R_p range, but also the over-sized R_p values. Further, we interpret the essential reasons of the effects by our technical observations.

7. Conclusions

In this paper, we propose *Turbo* to reduce power consumption of I²C communication in order to achieve low power multi-sensors accesses in passive sensing devices while keeping the simple connections. The basic idea is leverage a waveshaping module for signal recovery. Then, we observe the potential waveshaping function in off-the-shelf sensor chips so that *Turbo* can be implemented directly by only changing the value of pull-up resistors.

Further, we determine the effective resistance selection range and show that *Turbo* achieves robust communication within the resistance selection range. We also propose the physical signal model for I²C bus to describe the relationship between the energy consumption and the data rate in *Turbo*. Finally, we formulate the optimal resistance problem as an optimization problem to seek the best pull-up resistor value. Our evaluation shows that when the optimal resistance is selected, the energy consumption is minimized to 23.1% of the standard value. We also recommend an empirical resistance 50 KΩ that achieves near minimum energy consumption in practical cases.

The future work of *Turbo* includes two directions. First, we will optimize the design of *Turbo*. For example, a better design could be IC independent that the waveshaping module should be outside the ICs. Second, we will further investigate a way to improve the accuracy of the signal model.

Author Contributions: Conceptualization, Y.S. and S.L. (Songfan Li); methodology, Y.S., L.L. and C.Z.; software, S.L. (Songfan Li) and S.L. (Sengyu Li); prototype, Y.S., S.L. (Songfan Li) and C.Z.; validation, Y.S., S.L. (Songfan Li) and C.Z.; investigation, S.L. (Sengyu Li); writing—original draft preparation, Y.S.; writing—review and editing, Y.S. and S.L. (Songfan Li); visualization, S.L. (Sengyu Li); supervision, L.L.; project administration, L.L.; funding acquisition, L.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research is funded by the National Natural Science Foundation of China under Grant 61872061.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the first author.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

IoT	Internet of Things
COTS	Commercial Off-the-shelf
BER	Bit Error Rate
SPDT	Single-Pole Double-Throw
SPST	Single-Pole Single-Throw
TX	Transmit
RX	Receive
IC	Integrated Circuit
LED	Light-Emitting Diode

References

1. Salajegheh, M.; Clark, S.S.; Ransford, B.; Fu, K.; Juels, A. CCCP: Secure Remote Storage for Computational RFIDs. In Proceedings of the USENIX Security Symposium, Montreal, QC, Canada, 10–14 August 2009; pp. 215–230.
2. Zhang, P.; Hu, P.; Pasikanti, V.; Ganesan, D. Ekhonet: High speed ultra low-power backscatter for next generation sensors. In Proceedings of the 20th Annual International Conference on Mobile Computing and Networking, Maui, HI, USA, 7–11 September 2014; pp. 557–568.
3. Li, S.; Zhang, C.; Song, Y.; Zheng, H.; Liu, L.; Lu, L.; Li, M. Internet-of-microchips: Direct radio-to-bus communication with SPI backscatter. In Proceedings of the 26th Annual International Conference on Mobile Computing and Networking, London, UK, 21–25 September 2020; pp. 1–14.
4. Devices, A. Analog Devices adxl362 Accelerometer Sensor. 2019. Available online: <https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL362.pdf> (accessed on 12 March 2022).

5. Varadharajan Idhiam, K.S.; Caswell, J.A.; Pozo, P.D.; Sabolsky, K.; Sierros, K.A.; Reynolds, D.S.; Sabolsky, E.M. All-Ceramic Passive Wireless Temperature Sensor Realized by Tin-Doped Indium Oxide (ITO) Electrodes for Harsh Environment Applications. *Sensors* **2022**, *22*, 2165, <https://doi.org/10.3390/s22062165>.
6. Li, F.; Wang, Y.; Ju, H.; Wang, Y.; Wang, Z.; Zhou, H. FarpScus: Fully Anonymous Routing Protocol with Self-Healing Capability in Unstable Sensor Networks. *Sensors* **2020**, *20*, 6683.
7. Kellogg, B.; Talla, V.; Gollakota, S.; Smith, J.R. Passive Wi-Fi: Bringing Low Power to Wi-Fi Transmissions. In Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation, NSDI'16, Santa Clara, CA, USA, 16–18 March 2016; pp. 151–164.
8. Kellogg, B.; Parks, A.; Gollakota, S.; Smith, J.R.; Wetherall, D. Wi-Fi Backscatter: Internet Connectivity for RF-Powered Devices. In Proceedings of the 2014 ACM Conference on SIGCOMM, SIGCOMM '14, Chicago, IL, USA, 17–22 August 2014; Association for Computing Machinery: New York, NY, USA, 2014; pp. 607–618, <https://doi.org/10.1145/2619239.2626319>.
9. Laboratories, S. Smbus Communication for Small Form Factor Device Families. 2013. Available online: <https://www.silabs.com/documents/public/application-notes/an141.pdf> (accessed on 12 March 2022).
10. Semiconductor, F. SMBus Quick Start Guide. 2013. Available online: <https://www.nxp.com.cn/docs/en/application-note/AN4471.pdf> (accessed on 12 March 2022).
11. Sample, A.P.; Yeager, D.J.; Powledge, P.S.; Mamishev, A.V.; Smith, J.R. Design of an RFID-based battery-free programmable sensing platform. *IEEE Trans. Instrum. Meas.* **2008**, *57*, 2608–2615.
12. Schmitt Trigger. 2018. Available online: https://en.wikipedia.org/wiki/Schmitt_trigger (accessed on 10 July 2018).
13. Higgins, R.J. *Electronics with Digital and Analog Integrated Circuits*; Prentice-Hall: Hoboken, NJ, USA, 1983.
14. Bistable Multivibrator. 2018. Available online: <https://en.wikipedia.org/wiki/Multivibrator> (accessed on 10 July 2018).
15. Wikipedia. Flip-Flop (Electronics). 2019. Available online: [https://en.wikipedia.org/wiki/Flip-flop_\(electronics\)](https://en.wikipedia.org/wiki/Flip-flop_(electronics)) (accessed on 10 May 2019).
16. Impinj. Monza X-2K Dura. 2016. Available online: <https://support.impinj.com/hc/en-us/articles/202756848-Monza-X-2K-Dura-Datasheet> (accessed on 12 March 2022).
17. Technology, M. 8K I2C Serial EEPROM. 2003. Available online: <http://ww1.microchip.com/downloads/en/DeviceDoc/21710c.pdf> (accessed on 12 March 2022).
18. Instruments, T. MSP430F2132 Datasheet. 2012. Available online: <http://www.ti.com.cn/cn/lit/ds/symlink/msp430f2132.pdf> (accessed on 12 March 2022).
19. MathWorks, I. Fminbnd Function Description. 2018. Available online: <https://www.mathworks.com/help/matlab/ref/fminbnd.html> (accessed on 25 December 2018).
20. Devices, A. 1-/2-Channel 15 V Digital Potentiometer. 2010. Available online: https://www.analog.com/media/en/technical-documentation/data-sheets/AD5260_5262.pdf (accessed on 12 March 2022).
21. Instruments, T. PCF8574 Remote 8-Bit I/O Expander for I2C Bus. 2015. Available online: <http://www.ti.com/lit/ds/symlink/pcf8574.pdf> (accessed on 12 March 2022).
22. Labs, S. Humidity and Temperature Sensor. 2016. Available online: <https://www.silabs.com/documents/public/data-sheets/Si7013-A20.pdf> (accessed on 12 March 2022).
23. Instruments, T. MSP430FR5969 Datasheet. 2018. Available online: <http://www.ti.com/lit/ds/symlink/msp430fr5969.pdf> (accessed on 12 March 2022).
24. Liu, V.; Parks, A.; Talla, V.; Gollakota, S.; Wetherall, D.; Smith, J.R. Ambient Backscatter: Wireless Communication out of Thin Air. In Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM, SIGCOMM '13, Hong Kong, China, 12–16 August 2013; Association for Computing Machinery: New York, NY, USA, 2013; pp. 39–50, <https://doi.org/10.1145/2486001.2486015>.
25. Shafik, R.; Yakovlev, A.; Das, S. Real-Power Computing. *IEEE Trans. Comput.* **2018**, *67*, 1445–1461, <https://doi.org/10.1109/TC.2018.2822697>.
26. Balsamo, D.; Das, A.; Weddell, A.S.; Brunelli, D.; Al-Hashimi, B.M.; Merrett, G.V.; Benini, L. Graceful performance modulation for power-neutral transient computing systems. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2016**, *35*, 738–749.
27. Balsamo, D.; Weddell, A.S.; Das, A.; Arreola, A.R.; Brunelli, D.; Al-Hashimi, B.M.; Merrett, G.V.; Benini, L. Hibernus++: A self-calibrating and adaptive system for transiently-powered embedded devices. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2016**, *35*, 1968–1980.
28. Kansal, A.; Hsu, J.; Zahedi, S.; Srivastava, M.B. Power management in energy harvesting sensor networks. *ACM Trans. Embed. Comput. Syst. TECS* **2007**, *6*, 32.
29. Li, S.; Lu, L.; Hussain, M.J.; Ye, Y.; Zhu, H. Sentinel: Breaking the Bottleneck of Energy Utilization Efficiency in RF-Powered Devices. *IEEE Internet Things J.* **2018**, *6*, 705–717.
30. Buettner, M.; Greenstein, B.; Wetherall, D. Dewdrop: An energy-aware runtime for computational RFID. In Proceedings of the Symposium on Networked, Systems Design and Implementation, USENIX NSDI, Boston, MA, USA, 30 March–1 April 2011; pp. 197–210.
31. Zhang, P.; Ganesan, D. Enabling Bit-by-Bit Backscatter Communication in Severe Energy Harvesting Environments. In Proceedings of the 11th USENIX Conference on Networked Systems Design and Implementation, NSDI'14, Seattle, WA, USA, 2–4 April 2014; USENIX Association: Berkeley, CA, USA, 2014; pp. 345–357.

32. Alliance, M. I³C Bus Specification. 2020. Available online: <https://www.mipi.org/specifications/i3c-sensor-specification> (accessed on 12 March 2022).
33. StackOverflow. What Happens If I Omit the Pullup Resistors on I2C Lines? 2014. Available online: <https://electronics.stackexchange.com/questions/102611/what-happens-if-i-omit-the-pullup-resistors-on-i2c-lines> (accessed on 20 July 2018).
34. Friesel, D.; Spinczyk, O. I2C considered wasteful: Saving energy with host-controlled pull-up resistors. In Proceedings of the 18th International Conference on Information Processing in Sensor Networks, Montreal, QC, Canada, 16–18 April 2019; pp. 315–316.
35. Jones, N. Lowering Power Consumption Tip 1—Avoid Zeros on the I2C Bus. 2009. Available online: <https://embeddedgurus.com/stack-overflow/2009/07/lowering-power-consumption-tip-1-avoid-zeros-on-the-i2c-bus/> (accessed on 20 July 2018).
36. Stan, M.R.; Burleson, W.P. Bus-invert coding for low-power I/O. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **1995**, *3*, 49–58.
37. circuits, D. Effects of Varying I2C Pull-Up Resistors. 2010. Available online: <http://dsscircuits.com/index.php/articles/47-effects-of-varying-i2c-pull-up-resistors> (accessed on 20 July 2018).