Computational Thinking and Technology Toys

by

Veronica Lin, Computer Science 2015

Thesis

Presented to the Program in Computer Science in Partial Fulfillment of the Requirements for Honors in the degree of

Computer Science

Wellesley College
May 2015
Acknowledgements

This research project is the culmination of my experience at Wellesley College. It has taken me a while to discover my passion for the intersection of technology and education, but this work allowed me to dive in deeper than I could have imagined. I would like to express my sincere gratitude to all of the individuals who made this accomplishment possible.

To my advisor, Orit Shaer, for recognizing the potential in me four years ago and for your guidance, patience, and support throughout my academic career at Wellesley. There are so many opportunities that I benefited from because of your dedicated mentorship and advice. Thank you for caring so much about my work and my interests, and also for introducing me to the wonderful world of Human Computer Interaction.

To my committee members: Sohie Lee, Eni Mustafaraj, and Noah Rubin, for your insight and assistance. Thank you for taking genuine interest in my work and providing thoughtful feedback that has helped me think more broadly and more deeply.

To Robbin Chapman, Amanda Sullivan, Tracy Gleason, Becky Geer, and the Wellesley College Child Study Center for sharing your experiences and expertise in teaching, child psychology, and evaluating with children. Your work has informed my research, and I am grateful for the time you so generously gave me.

To all of the child participants and parents, without whom this research would not be possible. A special thank you to the teachers and students at The Rashi School, who allowed me to assist during robotics activities and helped me discover my passion for technology education.

To the elementary and middle school students in Wellesley who I have taught programming classes to during recent months for giving me the opportunity to rediscover computer science through your youthful eyes. It has been both refreshing and inspiring, and I have high hopes for all of you.

To Kayla Jang, Vivien Chen, and all of the members of Robogals Wellesley for eagerly volunteering their time and efforts to support my research. And to the Robogals executive board, thank you and congratulations. It has been an incredible first year, and I am honored to have worked with you to inspire more girls to pursue STEM.

To my lab mates, especially Johanna Okerlund, for all of your help, encouragement, companionship, and advice.
To my friends, especially Veronica, Rocky, Kathy, Nicole, Maya, and Molly, for inserting breaks into my busy schedule and for always putting a smile on my face.

To Suzanne for giving me the confidence and inspiration to take on new challenges. Thank you for pushing me, supporting me, and being there for me through it all. I could not ask for a better best friend.

To my family, whose unconditional love and support have sustained me over the past two decades. Mom and Dad, you have provided me with opportunities that have made me the woman I am today, and I cannot express enough my gratitude for your hard work, dedication, and guidance. Denis, without you, I may have never discovered Computer Science. You knew me better than I knew myself, and I am so grateful to have such a talented, compassionate big brother. And Cody, you will always be my baby brother, no matter how tall you grow. Thank you for always believing in me.
Computational thinking is an increasingly popular topic for computer science educators, and the human computer interaction community has suggested the potential of Tangible User Interfaces to support children’s learning. This research aims to study how commercially available tangible technology toys, such as littleBits and KIBO, can promote the development of computational thinking for children in early elementary school. Evaluation included user studies with children in three different formal and informal educational settings. I investigated how each setting affects engagement, complexity, and collaboration. Findings demonstrate that TUIs support learning of computational thinking skills for young children in various settings, and can be used to further the discussion of how computational thinking is taught.
# Table of Contents

List of Tables .................................................................................................................. viii
List of Figures .................................................................................................................. ix

**INTRODUCTION** ....................................................................................................... 1

**BACKGROUND** ........................................................................................................ 2
Computational Thinking .................................................................................................. 2  
  Overview ...................................................................................................................... 2  
  Frameworks ................................................................................................................ 2  
  Curriculum Domains ................................................................................................. 4  
Related Work .................................................................................................................. 5  
  Tangible User Interface Design ................................................................................. 5  
  Comparing Technology Toys ..................................................................................... 7  
  Evaluating with Children .......................................................................................... 10  
  Indicators for Learning ............................................................................................. 13  
  Curriculum Models for Teaching Computational Thinking ...................................... 14  
Toy Design .................................................................................................................... 16  
  Products ...................................................................................................................... 17  
    Overview .................................................................................................................. 17  
    littleBits .................................................................................................................. 18  
    KIBO ....................................................................................................................... 19  
    LEGO WeDo ......................................................................................................... 21  
    Scratch .................................................................................................................... 22  

**EVALUATION** ......................................................................................................... 25  
Methodology .................................................................................................................. 25  
  Preparation ................................................................................................................ 25  
  Experimental Design ............................................................................................... 25  
  Basic Protocol .......................................................................................................... 28  
Data Analysis Methodology ......................................................................................... 28  
  Data Analysis Tool .................................................................................................. 28  
  Talk Categories ....................................................................................................... 30
List of Tables

Table 1: Brennan & Resnick’s computational thinking framework ........................................3
Table 2: Description of each setting and their characteristics .........................................25
Table 3: Coding scheme for classifying talk categories ......................................................31
Table 4: Coding scheme for analyzing physical gestures and facial expressions ..............32
List of Figures

Figure 1: littleBits Base Kit ........................................................................................................19
Figure 2: KIBO construction kit .................................................................................................20
Figure 3: LEGO WeDo software .................................................................................................22
Figure 4: Sample LEGO WeDo project ......................................................................................22
Figure 5: Scratch editor .............................................................................................................23
Figure 6: Sample Scratch project ...............................................................................................24
Figure 7: ATLAS.ti screenshot of association editor .................................................................29
Figure 8: ATLAS.ti screenshot of video coding .........................................................................29
Figure 9: ATLAS.ti screenshot of transcription coding ...............................................................30
Figure 10: Lab setting ..................................................................................................................35
Figure 11: Distribution of talk categories per gender – lab setting, littleBits ............................36
Figure 12: Distribution of talk categories per gender – lab setting, KIBO .................................36
Figure 13: Distribution of talk categories per toy – lab setting ..................................................37
Figure 14: Engagement components per gender – lab setting, littleBits .................................38
Figure 15: Engagement components per gender – lab setting, KIBO ........................................38
Figure 16: Comparison of engagement components per toy – lab setting .................................39
Figure 17: Complexity score components per gender – lab setting, littleBits .........................40
Figure 18: Sample littleBits projects created by lab participants .............................................40
Figure 19: Collaboration components per gender – lab setting, littleBits .................................41
Figure 20: Collaboration components per gender – lab setting, KIBO .........................................41
Figure 21: Comparison of collaboration components per toy – lab setting ...............................42
Figure 22: Enjoyment scores overall – lab setting .......................................................................43
Figure 23: Enjoyment scores per gender – lab setting ...............................................................44
Figure 24: Enjoyment scores per grade level – lab setting .........................................................44
Figure 25: Workshop setting ......................................................................................................48
Figure 26: Distribution of talk categories per grade level – workshop setting ............................48
Figure 27: Engagement components per grade level – workshop setting ..................................49
Figure 28: Complexity score components per grade level – workshop setting ..........................50
Figure 29: Sample littleBits projects created by workshop participants .....................................50
Figure 30: Collaboration components per grade level – workshop setting ...............................51
Figure 31: Classroom setting ...........................................................................................................54
Figure 32: Distribution of talk categories per gender – classroom setting ..................................55
Figure 33: Engagement components per gender – classroom setting ............................................55
Figure 34: Complexity score components per gender – classroom setting ..................................56
Figure 35: Sample littleBits projects created by classroom participants .....................................56
Figure 36: Collaboration components per gender – classroom setting ...........................................57
Figure 37: Distribution of talk categories per setting – all participants .......................................60
Figure 38: Engagement components per setting – all participants ................................................61
Figure 39: Complexity score components per setting – all participants .......................................62
Figure 40: Collaboration components per setting .........................................................................63
Figure 41: Ten Weeks of Tech: Cover & introduction .................................................................74
Figure 42: Ten Weeks of Tech: Sample week .............................................................................75
Figure 43: Ten Weeks of Tech: Sample week (cont.) .................................................................76
Figure 44: Got STEAM: Cover & structure of challenges ..........................................................78
Figure 45: Got STEAM: Sample challenges & tips ......................................................................79
Figure 46: Smileyometer ...............................................................................................................91
INTRODUCTION

Following Jeannette Wing’s 2006 article [42] about the importance and application of computational thinking, the computer science education community has increasingly focused on the embedding of computational thinking concepts into K-12 curriculum to help our youth develop computational thinking skills [4, 5, 24, 25]. Challenging questions arise about what computational thinking encompasses [5], how computational thinking concepts can be made accessible to educators [4, 24], how the development of computational thinking in children can be assessed [10, 25, 41], and what tools can be used to teach computational thinking skills [6]. Research efforts have focused mostly on the first three topics, but there are few studies that explore strategies and tools for teaching computational thinking in early elementary school.

Meanwhile, the human computer interaction and interaction design communities have suggested the potential of Tangible User Interfaces (TUIs) to support children’s learning [3, 20, 27, 31, 44, 46]. Researchers note that TUIs promote active, hands-on engagement; offer opportunities for collaboration between users; and allow for exploration and reflection [3, 50, 51]. Recent studies present TUIs that promote the development of computational thinking skills [15, 28, 38, 50], but most evaluations are conducted with early prototypes rather than with commercially available tools accessible to the general public.

This research project aimed to study how commercially available tangible technology toys such as littleBits and KIBO can help promote the development of computational thinking for children in early elementary school. After conducting a literature review on computational thinking, Tangible User Interface and toy design, technology toys, evaluating with children, indicators for learning, and curriculum models and domains for computational thinking, we evaluated littleBits in three different formal and informal educational settings to investigate how each setting affects engagement, complexity, and collaboration. In addition, we studied littleBits and KIBO to explore how particular characteristics in toy design contribute to the learning of computational thinking. We also used teacher interviews and two curriculum development projects to explore the potential for teaching computational thinking in classrooms and at home. These findings demonstrate the capability of TUIs to support the learning of computational thinking skills for young children in various settings, and can be used to further the discussion of how computational thinking can be taught.
BACKGROUND

Computational Thinking

OVERVIEW

Computational thinking refers to the set of thinking skills, practices, and approaches that are essential for solving complex problems. It had long been seen as a skill relevant only for computer scientists and engineers; however, when Jeannette Wing introduced the concept in 2006 as a “fundamental skill for everyone,” computer scientists and educators alike engaged in discussions about the meaning of computational thinking and how it can be incorporated into K-12 curriculum [48]. Researchers have demonstrated that computational methods and models are universally applicable and can benefit professionals of every discipline [5, 24, 48]. Experts are eager to increase student exposure to computational thinking as early as pre-kindergarten, which has led to the development of several computational thinking frameworks and the discussion of various curriculum domains in which computational thinking can be taught [5, 10, 24, 25, 41]. However, research efforts have focused primarily on exploring strategies and tools for teaching computational thinking in middle and high school, offering little support for younger children.

FRAMEWORKS

In exploring how computational thinking can be integrated into youth education, researchers have focused on developing frameworks to understand and assess computational thinking. Although most educators and computer scientists agree that computational thinking is related to problem solving, there are discrepancies about what it actually involves. Wing’s 2006 article stated that computational thinking involves concepts such as problem decomposition, data representation, and modeling, as well as ideas more specific to computer science, such as binary search, recursion, and parallelization [48]. The broad range of this definition makes it challenging for educators, curriculum developers, and toy designers to create opportunities for our youth learn computational thinking. In creating computational thinking frameworks like the ones described below, researchers have attempted to make accessible the ideas and concepts of computational thinking by providing narrow definitions with clear examples.
Brennan and Resnick present a computational thinking framework for studying and assessing the development of computational thinking [10]. Their framework consists of three key dimensions: computational concepts, computational practices, and computational perspectives. Computational concepts, the concepts that users engage with as they program, include sequencing, loops, parallelism, conditionals, operators, and data. Computational practices, the practices that users develop while thinking and learning, include testing and debugging, being incremental and iterative, reusing and remixing, and abstracting and modularizing. Computational perspectives, the perspectives that users form about the world around them and themselves, include expressing, connecting, and questioning. All of these concepts, practices, and perspectives, defined in more detail in Table 1, are fundamental skills that are employed by expert programmers. Brennan and Resnick demonstrate how each skill is supported by Scratch, a programming environment created to promote learning through design-based activities.

Table 1: Concepts, practices, and perspectives of Brennan & Resnick’s computational thinking framework.

<table>
<thead>
<tr>
<th>term</th>
<th>category</th>
<th>definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>sequences</td>
<td>concepts</td>
<td>tasks are expressed as a series of individual steps or instructions</td>
</tr>
<tr>
<td>loops</td>
<td>concepts</td>
<td>run the same sequence multiple times</td>
</tr>
<tr>
<td>parallelism</td>
<td>concepts</td>
<td>sequences of instructions happening at the same time</td>
</tr>
<tr>
<td>events</td>
<td>concepts</td>
<td>one thing causing another thing to happen</td>
</tr>
<tr>
<td>conditionals</td>
<td>concepts</td>
<td>ability to make decisions based on certain conditions</td>
</tr>
<tr>
<td>operators</td>
<td>concepts</td>
<td>provide support for performing numeric/string manipulations</td>
</tr>
<tr>
<td>data</td>
<td>concepts</td>
<td>storing, retrieving, updating values</td>
</tr>
<tr>
<td>being incremental &amp; iterative</td>
<td>practices</td>
<td>using cycles of imagining and building</td>
</tr>
<tr>
<td>testing &amp; debugging</td>
<td>practices</td>
<td>strategies for dealing with/anticipating problems</td>
</tr>
<tr>
<td>reusing and remixing</td>
<td>practices</td>
<td>building on existing work, understanding ownership/authorship</td>
</tr>
</tbody>
</table>
Seiter and Foreman also introduce a framework for understanding and assessing computational thinking in the primary grades, the Progression of Early Computational Thinking (PECT) Model [41]. Their model consists of three components: computational thinking concepts, design pattern variables, and evidence variables. Again applied to Scratch, the computational thinking concepts include procedures and algorithms, parallelization, abstraction, problem decomposition, and data representation. Design pattern variables are proficiencies specific to coding patterns in Scratch, such as animation, conversation, score-keeping, and user interaction. Evidence variables, which help measure levels of sophistication, are based on computing categories in Scratch, such as looks, sound, motion, conditionals, coordination, and operators. The PECT Model combines the design pattern and evidence variables in order to assess students’ proficiency of computational thinking concepts. Seiter and Foreman provide a detailed rubric with three levels: basic, developing, and proficient. They conduct a pilot study with 150 Scratch projects to demonstrate the PECT Model’s ability to detect differences in computational thinking among students of various ages.

**Curriculum Domains**

It is only in recent years that educators and computer scientists have advocated for the widespread education of computational thinking. Researchers have demonstrated the range of applicability by incorporating computational thinking in a variety of curriculum domains, including robotics, reading, math, and even social studies.

Lee et al. promote the use of robotics to teach computational thinking [24]. They present three different domains in which computational thinking takes place: modeling and simulation, robotics, and game design and development. They argue that computational thinking, which they
define as the use of abstraction, automation, and analysis in problem solving, can easily be embedded within activities that empower learners to imagine, create, play, and reflect.

In addition, Barr et al. provides a structured model for embedding core computational thinking concepts and capabilities in activities across multiple disciplines [5]. Concepts and capabilities, such as abstraction, problem decomposition, parallelization, and data analysis, are applicable in subjects from math and science to social studies and language arts.

Lu and Fletcher illustrate the ways in which computational thinking aligns with primary and secondary curricula in topics such as reading, social studies, and math [25]. They present the idea of a “computational thinking language” in order to teach the vocabularies and symbols associated with computational processes. By presenting several examples, Lu and Fletcher provide a model that helps students act as computers to become “more knowledgeable, skilled, and effective.”

Related Work

Tangible User Interface Design

The human computer interaction and interaction design communities have suggested the potential of Tangible User Interfaces (TUIs) to support children’s learning. The term TUIs refers to user interface technologies that link the digital and physical worlds [42]. Researchers have suggested that TUIs promote active, hands-on engagement; offer opportunities for collaboration between users; and allow for exploration and reflection. Here I describe several studies that demonstrate the benefits of TUIs, particularly for child users.

According to Xu, TUIs are more intuitive, support trial-and-error activity, allow more than one user, and offer users an alternative form of interaction and control of the environment [51]. These benefits, combined with the fact that TUIs do not require any screen time and yet are equally engaging, make tangible toys quite appealing as learning tools. Marshall echoes these claims, noting that using physical materials in a learning task may change the nature of the knowledge gained [27]. In addition, Marshall states that tangible interfaces may lead to increased engagement, reflection, and collaboration for children. The familiar interaction style of tangible interfaces also makes TUIs more accessible to young children and people with learning disabilities, leading to a lower threshold for participation.
O’Malley and Fraser suggest that tangible systems encourage discovery and participation [31]. They note that tangibles have the potential to provide children with innovative ways to play and learn, and the ability for TUIs to capitalize on users’ familiarity with the physical world increases the users’ ability to explore, manipulate, and reflect.

Xie et al. investigate how physical, graphical, and tangible interfaces impact the enjoyment and engagement of children while playing with puzzles [50]. Physical puzzles are traditional jigsaw puzzles; graphical puzzles are manipulated with a mouse or a touchpad on traditional computers; and tangible puzzles involve direct manipulation on multi-touch tabletops. Xie et al. compared quantitative measures of enjoyment and engagement across implementation styles, and found that children performed better when they could physically manipulate objects. They were also quicker to complete puzzles with the physical and tangible user interfaces.

Horn et al. also present a study comparing the use of a tangible user interface and a traditional graphical user interface with an interactive robotics museum exhibit [19]. Their results show that while users found both interfaces easy to use, the tangible interface attracted more visitors and encouraged more group participation. In addition, Horn et al. observed that girls were significantly more likely to use the exhibit with the tangible interface than the graphical interface. This finding has the potential to advance current efforts to promote girls in STEM.

Towards Utopia is a tabletop environment designed to facilitate children ages 7-10 in learning about sustainable development [3]. Antle et al. implemented the software tool with constructivist learning theories in mind and conducted user studies to validate the effectiveness of their design. The evaluation consisted of a pre-test and a post-test, and results showed that students significantly increased their scores after using Towards Utopia.

Zuckerman et al. introduce “Montessori-inspired Manipulatives,” technology enhanced building blocks that enable children to physically explore abstract concepts [52]. The SystemBlocks that are presented and evaluated demonstrate that TUls are capable of promoting group interaction and discussion, and also create opportunities for engaging children in learning.

Wyeth presents a framework that focuses on how tangible technologies offer enjoyable experiences for users through their support of gross and fine motor interactions [49]. The analysis, based on a mapping from characteristics in TUI design to skill and challenge in
interaction to flow-like enjoyment, provides design guidelines for implementing effective tangible technologies.

Ultimately, researchers in the human computer interaction and interaction design communities have demonstrated the potential of TUIs to support children’s learning. TUIs provide benefits that are especially helpful for young children. Recent studies also provide guidelines for TUI design, which inform our discussion about technology toys.

**Comparing Technology Toys**

There is a wide variety of technology toys developed in the past decade to accommodate the learning of computational thinking, including color-coded electronic modules, programmable robots, board games, and iPad applications. Together, they represent many different modes of interaction: some are strictly visual, characterized by traditional interfaces (computer screen, keyboard, and mouse); some are strictly tangible, requiring no screen time at all; and some are hybrids, incorporating both visual and tangible aspects.

**Visual Toys**

Technology toys that are strictly visual include different types of desktop- and tablet-based software applications. They are generally easier to develop and access, since there is no manufacturing necessary. One of the most popular tools associated with learning computational thinking is Scratch, originally developed by the MIT Media Lab. Scratch is a multimedia programming interface that was designed to make programming accessible to children and adults of all ages. Although most popular with children between the ages of eight and sixteen, Scratch has attracted a diverse set of users who have built projects from video games to interactive birthday cards. A similar application with a narrower focus is Frog Pond, which was studied by Horn et al. Frog Pond, a learning environment designed to introduce natural selection to elementary and middle school students, allows learners to use a blocks-based programming interface to control frogs inhabiting a lily pond [20]. It has been implemented as an application on laptop computers, tablets, and multi-touch tabletops, and was evaluated in a museum setting with users between the ages of 9 and 16, who showed effective engagement through programming.
In addition, a number of iPad apps are available for free download in Apple and Android mobile app stores, including ScratchJr, LightBot, Hopscotch, and Tynker. ScratchJr, a version of Scratch for children aged 5-7, allows kids to program their own interactive stories and games using icon-based blocks instead of text-based blocks. ScratchJr is a graphical programming language that addresses the developmental and learning needs of children in kindergarten to second grade. LightBot uses a similar layout with drag-and-drop icon-based blocks. Appropriate for children ages 4+, the LightBot application presents a series of puzzles that involve programming logic. The Hopscotch app, designed for children ages 9-11, also lets learners drag and drop blocks of code to create their own programs. It supports the creation of games, animations, and stories, and allows users to shake, tilt, or shout in order to control characters. Tynker, another iPad app for children ages 9-11, uses coding puzzles and game building to teach programming concepts and skills.

Tangible Toys

In recent years, an increasing number of tangible technologies have been developed to promote the learning of computational thinking. Most are studied exclusively in an academic context, while others have thrived in the commercial market. Horn et al. presented Tern, a tangible computer language that allows users to create physical computer programs using interlocking wooden blocks [19]. Each block represents an action for the robot to perform. Tern evolved into Cherp, which Bers et al. present as a hybrid tangible and graphical computer language that utilizes pictorial icons to help young children create programs to control their robots [6]. Cherp later evolved into KIBO, which is discussed in the next section. Topobo, a construction toy with kinetic memory, is another popular tangible user interface for children [28]. It contains passive (static) and active (robotic) pieces that can be snapped together, and the final creation can record and playback physical motion. Escape Machine and Electronic Blocks are two other tangible technologies. Weller et al. present Escape Machine, a game that is played through manipulating a tangible state machine built with Posey, a computationally enhanced construction kit [41]. Wyeth and Purchase present Electronic Blocks, tangible programming blocks with electronic circuits inside them [49]. The blocks can be stacked and arranged to form structures that interact with the physical world. While these technologies demonstrate how
tangible user interfaces can engage children in computational thinking during child’s play, they are not entirely accessible to the general public.

The commercial market contains just a few tangible technology toys, including littleBits, KIBO, and GoldieBlox, but they all demonstrate a strong presence in the media. littleBits has gained significant traction with children and adults ages 8+. As a construction kit with a variety of color-coded electronic modules, littleBits are a novel and exciting toy that can be used everywhere from home to school. Users can create a variety of circuit combinations by magnetically connecting inputs and output pieces, and can add crafts and other materials to build smart objects with a purpose. Another tangible technology that has been popular in the commercial market is KIBO, a robot kit for children ages 4-7. It allows children to build programs using wooden blocks that closely resemble Cherp. Each block represents a command that the robot can perform, such as “move forward,” “spin,” or “sing.” The user can then scan the program using the robot’s scanner, and the robot will execute the program. GoldieBlox, named the “Engineering Toy for Girls,” was developed to appeal to girls ages 5-9. It combines reading and building to help users build with a purpose; equipped with construction pieces like axles, pulleys, and gears, learners are tasked with creating machines in order to solve problems for characters in the books. The design features of GoldieBlox, including the colors of the construction pieces, story themes, and characters, all contribute to the ultimate goal of helping girls learn engineering concepts.

Board games have also been developed to teach programming concepts to younger children. Robot Turtles is one such board game that is available at many retail and toy stores. It teaches the fundamentals of programming for children ages 3-8; players are tasked with getting their turtle to the treasure using different command cards, such as move forward and turn right. The low cost of these technology-less tools makes them more accessible to everyone, which is increasingly important in a world where social inequality is so prevalent.

**Hybrid Toys**

Finally, many toys fall under the category of hybrids, which reflect a combination of visual and tangible modes of interaction. Some tools, such as TanPro-Kit, Makey Makey, and LEGO WeDo, allow users to manipulate tangible objects and view the results on a traditional
visual-based display. TanPro-Kit, also developed by Wang et al., is a set of visual programming blocks with an LED pad that presents visual animations and audible feedback according to the arrangement of the blocks [46]. Unlike TanPro-Kit, Makey, Makey is now commercially available. Originally designed in the MIT Media Lab, the Makey, Makey invention kit is now sold by JoyLabz to help children and adults ages 8+ explore art and engineering. Makey, Makey allows users to turn everyday objects into touchpad inputs. LEGO WeDo, sold by LEGO Education and made for children ages 7+, is a construction set and a visual blocks programming interface that enables students to build and program simple LEGO models. LEGO WeDo is incredibly popular because of its familiar LEGO platform, and is most commonly used in schools and workshop settings.

Other hybrid technology toys include programmable robots, which have visual input interfaces and tangible output. Robo-Blocks, presented by Sipitakiat et al., is a programming interface that allows students to create a program by connecting physical command blocks and wirelessly control the motion of a floor robot [44]. Sphero is a commercially available programmable sphere for children ages 8+. As a spherical robot that can be controlled from a smartphone or tablet application, the Sphero can change colors, move every direction, and even jump. Similarly, Bo and Yana, designed for children ages 5+, consists of a robot and a visual programming interface that helps kids learn to code by controlling robots and creating stories and animations.

**Evaluating with Children**

Several works from academia and industry describe the process of usability testing with children, providing many insights into experiment design, evaluation methods for children, and tricks and challenges associated with assessing children’s technologies. The key contributions of these papers were important considerations used in developing and executing the study protocol.

Many researchers present work investigating and comparing different evaluation methods and experimental designs for conducting user studies with children. Donker and Reitsma, in a study with 70 children in grades K-1, investigated the effect of experience on usability testing [13]. They employed a talk aloud method in their studies, where children were prompted to talk about what they were doing. Their findings indicate that behavioral observations and voluntary talk aloud is suitable for usability testing with children.
Rounding et al. presents a few evaluative techniques that they used with children [37]. Their findings indicate that a combination of hands-on collaborative activities performed in pairs, followed by whole-group discussion, is most effective. They suggest that paired activities should be hands-on and allow the children to produce tangible results that they can share with others. In addition, by pairing children with others of similar genders and age, comfort and collaboration is increased. Rounding et al. also employed a stations approach, in which they had children rotate through several usability testing activities at different stations of 25 minutes or less. Both methods required the presence of additional adult facilitators to keep children on task. Moreover, Rounding et al. discusses testing with children “in the wild.” They mention the benefits of doing so, such as being able to observe how the environment facilitates learning and how different teaching styles affect technology education, but also note the challenges in timing and security that prevent many researchers from using this approach.

Edwards and Benedyk assessed three usability evaluation methods with 6-8 year old children within a school setting using an interactive, multimedia platform [14]. Active intervention involves the tester probing the participant’s understanding of the concepts; peer tutoring involves a child tutor spending some time with a product and then teaching a friend to complete tasks; cross-age tutoring is similar but the child tutor is older than the friend. These techniques are alternatives to traditional think-aloud methods when user testing with younger children. Edwards and Benedyk also mention retrospection, in which participants comment on their thought processes after completing the tasks, and co-discovery, where two participants work together to solve the tasks, as other techniques for usability testing with children. They analyzed the verbal utterances of all participants during each session, and found that active interaction elicited the largest number of comments.

Read et al. describes methods for measuring enjoyment, endurability, and engagement with children aged between 5 and 10 [33]. The first method, called the “funometer,” is a vertical scale with an unhappy face at the bottom and a happy face at the top. It resembles a thermometer, and children are instructed to color in the bar based on how much fun they had. The second tool, the “smileyometer,” has been used in several user studies with children. Based on a 1-5 Likert scale, it uses five pictorial representations from an unhappy face to a happy face. The third method that Read et al. presents is the again-again table, which is used to measure endurability. The table contains a list of the activities in the first column, and then three columns for yes,
maybe, and no. Users are asked to indicate whether or not they would like to do the activity again. Finally, to measure engagement, Read et al. encourages researchers to analyze video footage using positive and negative instantiations. These instantiations include smiles, laughter, concentration signs (fingers in mouth, tongue out), excitable bouncing, and positive vocalization for positive engagement, and frowns, signs of boredom (ear playing, fiddling), shrugs, and negative vocal instantiation for disengagement.

In addition, several researchers offered guidelines and suggestions for conducting user studies with children in general. Edwards and Benedyk discussed several issues with child participants, including issues of intervention to encourage and motivate children, children’s large individual differences in capability and personality, and the variety of contexts in which they use multimedia products [14]. They suggest giving child participants a few minutes to explore the product before presenting them with formal tasks.

Hanna et al. presents many guidelines for usability testing with children [17]. In addition to providing general characteristics of particular age ranges, they discuss beneficial adaptations when using child participants. They suggest setting up the lab in a child-friendly manner, and give examples of how to do so. They also recommend establishing a relationship with children by engaging in small talk and asking about their birthdays, favorite subjects, or favorite games. Gaining trust with parents is also important, and Hanna et al. propose giving a lab tour prior to the study. Moreover, they suggest motivating children by emphasizing the importance of their role and provide example scripts to do so. They also suggest redirecting child users’ questions with questions, offering generic positive feedback, and switching the order of tasks so that the same tasks are not performed when children are likely tired. Hanna et al. recommend observing physical signs of engagement and disengagement to gauge child participants’ levels of enjoyment. Smiles, laughter, and leaning forward are all signs of engagement, while frowns, sighs, yawns, and turning away from the computer indicate disengagement. According to Hanna et al., these observations are more reliable than verbal comments, as children are eager to please adults.

These researchers’ suggestions and guidelines were taken into account when designing the study protocols and conducting the user studies. We discuss our rationale in detail in the “Methodology” section.
INDICATORS FOR LEARNING

Taken together, the dimensions of engagement, complexity, and collaboration could serve as indicators of learning. Research has proven that each dimension is an integral aspect of children’s playful learning experiences, and the presence of each can suggest the occurrence of learning for children.

Many studies suggest that engagement and learning are closely related. Carini et al. examines the association between student engagement and traditional measures of academic performance [11]. In their study of over 1,000 college students, many measures of student engagement were linked positively with learning outcomes such as grades. Corno and Mandinach also describe the ways in which cognitive engagement affects learning and motivation in the classroom [12]. They investigate how classroom instruction can be used to develop learners who are engaged and thereby self-regulated. Their findings suggest that student engagement is linked with learning.

In addition, several studies in the human computer interaction space measure engagement as a dimension of learning. Read et al. and Hanna et al. both present methods for measuring engagement with children during usability studies, demonstrating that engagement is a dimension of interest [17, 33]. They suggest observing physical signs of engagement and disengagement to quantitatively measure the engagement of child participants. Xie et al. also investigates engagement as a measure of comparison for different implementations of jigsaw puzzles [50]. They use the amount of on-task activity time and the number of starts and completions of the puzzle as indicators of engagement. In addition, Schneider et al. presents a study of Phylo-Genie as a tabletop interface that supports learning [39]. They propose that the physical aspect of tangible interfaces foster engagement and exploration among children. Ultimately, engagement requires cognitive effort and deep processing of new information, which reflects the presence of learning.

Software tools are most effective with low-threshold and high-ceiling, which means that increasing complexity in children’s projects and programs could reflect the high-ceiling of a technology. Myers notes that having a high ceiling is an important goal for future user interface software tools because of the increasing diversity of user interfaces [29]. Tools must provide not only easy entry, but also a smooth path to increased power. This will make tools more engaging and more effective, allowing more learning to occur.
Research also demonstrates the relationship between collaboration and learning. Kirschner et al. compares the effectiveness of individual learning environments with collaborative learning environments [22]. They find that with complex tasks, collaborative learning reduces the mental task load for users and allows them to process additional information.

Many studies use collaboration as a dimension for learning when evaluating technology applications. Shaer et al. presented G-nome Surfer 2.0 and conducted user studies to demonstrate how the tabletop interface supports collaborative learning [43]. They compared a set of quantitative measures and qualitative indicators for two conditions: the tabletop interface and a traditional mouse graphical user interface. Effective collaboration was demonstrated by users who actively communicated with each other to demonstrate shared effort. In addition, task-related reflection induces more meaningful learning. Shaer et al. used a collection of indicators including participant attitudes, level of participation, nature of discussion, collaboration styles, and problem strategies in order to measure the effectiveness of collaborative learning with the G-nome Surfer 2.0 project. Xie et al. also demonstrates the importance of collaboration in children’s learning [50]. By communicating with each other and imitating one another, children are able to acquire new knowledge and develop a fundamental ability to collaborate with others.

In reviewing the literature, we determined that the dimensions of engagement, complexity, and collaboration are closely linked with learning. In the “Methodology” section, we discuss the verbal and physical indicators we use to measure these dimensions of learning.

**Curriculum Models for Teaching Computational Thinking**

Computational thinking curriculum has been developed for a wide range of audiences, from pre-K children to college students. A review of the existing curriculum models provided important insights about how computational thinking is currently taught and learned, both in formal and informal learning environments.

Bers et al. presents a curriculum model to engage kindergarten children in computational thinking and problem-solving [6]. Their 20-hour TangibleK Robotics Program uses the CHERP programming language and LEGO construction kits. The curriculum was tested in three kindergarten classrooms with a total of 53 children. Their findings indicate that although the TangibleK curriculum was engaging and developmentally appropriate for kindergarten students,
a modified curriculum with additional time for exploration is necessary to reinforce students’
learning and understanding.

Rogers and Portsmore present curriculum for teaching engineering in elementary school
using LEGO hardware and ROBOLAB software [36]. They demonstrate how the tools can be
used to teach math, science, and engineering, and discuss some of the lessons they have learned.
Incorporating robotics technology in the classroom is associated with many difficulties in
product troubleshooting and general teaching support. In addition, they observe a clear gender
difference during robotics activities: while girls are willing to work in teams and enjoy focusing
on designing before building, boys prefer working individually and tend to skip the design phase
altogether. They warn these gender differences make it difficult for teachers to engage all
students.

Grover et al. presents a six-week middle school curriculum, Foundations for Advancing
Computational Thinking (FACT) [16]. The curriculum, executed in Scratch and designed for
Israeli students, included computational thinking topics such as loops, variables, booleans, and
conditionals. Grover et al. assessed the curriculum and student learning in two six-week studies
through prior experience surveys, pre-tests, post-tests, quizzes, graded assignments, and student
interviews. Students showed improvements in learning outcomes, demonstrating potential for the
six-week module to teach computational thinking to middle school students using Scratch.

In 1999, Beer et al. developed a robotics course for college students to practice problem
solving skills [7]. They identified challenges that engineering and science students face when
transitioning to professional endeavors, and used the problem of building an autonomous robot to
engage students in real-world problem solving, multidisciplinary teamwork, and creative and
critical thinking.

In addition, Lee et al. propose a three-stage progression for engaging youth in
computational thinking, called “use-modify-create” [24]. Based on the idea of scaffolding, this
learning progression allows students to develop their own project by working off of others’
creations, transforming difficult activities into appropriate, incrementally challenging
experiences.

Barr et al. also suggest teamwork with explicit use of decomposition, abstraction,
negotiation, and consensus building as a model for creating a classroom culture that is most
conducive to computational thinking [5]. On a broader scale, they encourage K-12 administrators
to prepare teachers by providing more opportunities for professional development and more accessibility to open-source tools and networks.

These works helped us understand how computational thinking is currently taught and learned in a variety of formal and informal educational settings. Many of them, however, focus on older children, and there are few studies that describe computational thinking curriculum for children in early elementary school.

TOY DESIGN

To investigate the design factors that affect engagement, complexity, and collaboration in computational thinking toys, it was useful to review literature related to the design of tangible toys and the evaluation of non-technology toys. Antle’s framework drew connections between learning theory and the design of TUIs, which outlined the aspects of TUI design that impact learning. Other research, focused on the categorization of toys, provided a variety of dimensions that are commonly used in discussing and evaluating toys. These works informed the analysis of the results as well as the discussion about implications for toy design.

Antle and Wise present the Tangible Learning Design Framework to help guide the design of tangible user interfaces [1]. The framework describes aspects of TUI design that are important to consider in learning contexts: physical objects, digital objects, actions on objects, informational relations, and learning activities. 17 principles are provided to help designers make theory-informed design choices and to better explain how and why TUIs can be designed to support learning.

Miller conducted a study in 1987 where undergraduate students rated 50 toys on 12 dimensions including manipulability, symbolic play, creativity, sociability, competition, handling, nurturance, constructiveness, aggressiveness, attractiveness, appropriateness for boys, and appropriateness for girls [28]. The results demonstrated that toys considered appropriate for girls differed on many dimensions from toys considered appropriate for boys. Blakemore and Centers built on Miller’s study two decades later, investigating the characteristics of 126 toys [9]. They conducted a two-part study to determine whether the toys were suitable for boys, girls, or both, and analyzed how the toys performed on a 5-point scale for 26 dimensions, including many of those used by Miller. Additional dimensions of interest included: artistic, expensive, educational, exciting, moves on its own, encourages cooperation, sustains attention, develops
physical skills, provides an actual response to child’s input, and develops cognitive or intellectual ability. Their findings demonstrated that girls’ toys were associated with appearance, attractiveness, and nurturance. Meanwhile, highly masculine toys were rated higher on the dimension of violence and competitiveness. In addition, boys’ toys provided more feedback, involved construction, and exhibited movement that requires visual tracking. There was no evidence that girls’ toys are more creative than boys’ toys or that boys’ toys are more likely to encourage social play than are girls’ toys. These results contrast with the findings of Miller’s 1987 study.

The literature surrounding the design of tangible toys and the evaluation of non-technology toys is helpful in understanding the complexity of Tangible User Interface design and toy design. Antle’s framework breaks down TUI design into individual aspects that impact learning, while the studies by Miller and Blakemore provide dimensions that can be used to discuss and evaluate toys. We formed our own set of dimensions for technology toy design, which are further discussed in the “Implications for Design” section.

Products

OVERVIEW

In the previous section, we reviewed the technology toys in the commercial market and academic space, which demonstrates the wide range of learning technologies for children in early elementary school. littleBits and KIBO, two commercially available, tangible technologies, are the primary focus of this research because there is little existing literature about either of them, despite a strong presence in media. While both toys have the potential to help users develop computational thinking concepts and skills, they exhibit many differences in physical appearance, size, and interaction style. These differences are critical in investigating what design factors affect engagement, complexity, and collaboration in computational thinking toys.

littleBits and KIBO were studied in each of the lab sessions, and littleBits were also studied in the classroom and workshop settings. However, other technologies were also used to promote learning in the classroom and workshop settings. Although no quantitative data was collected for either of them (LEGO WeDo in the classroom and Scratch in the workshops), it is
important to note that those participants had exposure to computational thinking technologies. All four of these technology tools are described in detail below.

**LITTLEBITS**

littleBits are magnetic modules that snap together to create sophisticated circuits. Invented and sold by littleBits Electronics, the toys are building blocks that aim to “make everyone into an inventor” (littlebits.cc). littleBits are recommended for children and adults over the age of 8; however, a pilot study with a 5-year old female demonstrated that littleBits can be both appropriate and beneficial for younger children. Each module has a unique function, which is color-coded into four categories: power (blue), input (pink), output (green), and wire (orange). Although littleBits were developed with the goal of making prototyping and electronics more accessible to non-engineers, they show potential for teaching computational thinking concepts. In addition to supporting all of the computational practices and perspectives outlined by Brennan and Resnick, littleBits also help users understand sequences, parallelism, and events, key concepts of computational thinking.

While the most basic littleBits kit contains 10 littleBits modules, there are over 50 modules in the littleBits library, shown in Appendix A.1. The blue power modules provide power for the circuit; to reduce syntax errors, other modules can only connect from the right side. The pink input modules include different sensors, buttons, and switches, which can control the green output modules. Through lights, motors, and speakers, littleBits can produce visual, physical, and audible output. Finally, the orange wire modules include basic extension pieces, branches, and logic gates. The circuit works sequentially from left to right, so that pink input pieces only affect green pieces that come after, or to the right, of them.

Although not explicitly stated by littleBits Electronics, the littleBits modules have the potential to teach users the computational thinking concepts of sequences, parallelism, and events. Each circuit is like a program, and each module represents an individual instruction. In designing and building circuits, users must think about the order of the modules, thus learning about sequences. Parallelism is expressed through the orange branch modules, which allow multiple mini-sequences to happen at the same time. The input-output model of the green and pink littleBits modules demonstrates the concept of events, that one thing causes another thing to
happen. For instance, moving the slider from left to right makes the light brighter, while moving the slider from right to left makes the light dimmer.

Figure 1: littleBits modules of the littleBits Base Kit, which includes the power module, four input modules, four output modules, and one wire module.

KIBO

KIBO is an interactive robot that can be programmed using wooden blocks. KIBO was developed in an academic setting and is now sold commercially by KinderLab Robotics. Unlike littleBits, KIBO has an explicit mission of helping young children learn programming skills and concepts. The wooden KIBO blocks, each with a unique instruction such as “sing” or “forward,” fit together to create a program. In building a program and executing it with the robot, users learn computational thinking concepts such as sequences, loops, and conditionals.

A KIBO kit includes wooden programming blocks and a robot. The blocks, each with its own label, icon, and barcode, are shown in Appendix A.2. Except for the “begin” and “end” blocks, which are designed differently to prevent users from putting blocks before the “begin” or after the “end,” each block has a hole on the left side and a peg on the right side. This design reduces syntax errors, helping users build programs that the robot can actually read and execute. After creating a program using the wooden blocks, the user scans the blocks sequentially to allow the robot to “read” the program. For each block, the KIBO robot will either produce a high-frequency beep to signal a successful scan or a low-frequency sound to alert the user of a syntax error. After scanning the entire program, the user can press the button on the robot’s main body to “run” or execute the program that the robot has most recently scanned. The robot then
comes alive, lighting up, making sounds, and moving forwards, backwards, and in a circle. The repeat blocks enable the use of loops in a program, so that the same action or actions can be repeated a specific number of times. In addition, special versions of the kit include sensor pieces that fit on top of the robot, which allow users to create programs that sense input and act accordingly using conditional if-then blocks. KIBO has the following sensors: a light sensor, which can detect if the surrounding environment is extremely bright or dark; a distance sensor, which can see how near or far KIBO is from other objects; and a sound sensor, which can hear sounds.

Unlike littleBits, KIBO was developed with the goal of teaching young children aged 4-7 fundamental programming concepts. Tasks for the robot are expressed as a series of individual instructions, which are represented by single blocks. The KIBO robot, like a compiler, executes each instruction in order; users learn about sequences while building, executing, and debugging their KIBO programs. In addition, while the “repeat” block helps users understand loops, the “if” block teaches them about conditionals. These blocks allow children to not only create more complex programs for their robots, but also learn about fundamental concepts of computational thinking.

Figure 2: KIBO, including the interactive robot and the wooden programming blocks. Once the blocks are assembled to create a program, the robot’s scanner can be used to scan each block’s barcode. The triangle button in the center of the KIBO robot can be clicked to execute a loaded program.
LEGO WeDo

LEGO WeDo, made for children ages 7+, is a construction set and a visual blocks programming interface that enables students to build and program simple LEGO models. The hardware consists of typical LEGO bricks and a few motorized components. Each kit contains a USB hub to connect the hardware to the software on the computer; two motors that interface with axles; a motion sensor that detects objects; and a title sensor that detects six different positions. These bricks enable students to build a variety of models, including robotic animals, playgrounds, airplanes, and more.

The WeDo software uses a simple, colorful drag and drop interface that helps students learn programming concepts. The programming icons include: start, which must be placed at the beginning of every program; turn motor (left or right), which turns the motors on; stop, which marks the end of a program; motor power, which allows users to adjust the speed of the motors; motor timer, which allows users to specify the amount of time the motor should be on for; motor off, which turns the motors off; and wait, which tells the computer to wait a specific amount of time before reading the next instruction in the program. In addition, the LEGO WeDo software supports more advanced programming concepts by including a loop block to repeat a certain block of code and various input blocks (such as motion sensor, tilt sensor, numeric input, and random numeric input) to make dynamic interaction possible.

Unlike littleBits and KIBO, LEGO WeDo are geared towards teaching STEM concepts in the classroom. Built by LEGO Education, the WeDo kit is praised as “the best robotics system for elementary school students.” However, for students whose schools cannot afford to purchase WeDo kits, there are few alternatives for learning computational thinking in the classroom.
Figure 3: screenshot of the LEGO WeDo software. Blocks from the blocks palette (bottom) can be dragged to the workspace in the center to create programs.

Figure 4: a sample LEGO WeDo project, built with basic LEGO bricks, a motion sensor, and a WeDo motor.

**Scratch**

Scratch is a multimedia programming interface that was developed to make programming accessible to children and adults of all ages. Although most popular with children between the ages of 8-16, Scratch has attracted a diverse set of users who have built projects from video games to interactive birthday cards. The creators of Scratch, driven by the lack of digital fluency in today’s youth, aimed to create a platform that could foster creativity and learning [34].

22
Scratch, available for free as a desktop or online editor, has a child-friendly user interface. It is composed of three main components: the stage area, where users can view the results of their programs (i.e. animations and simulations); the blocks palette, where blocks are sorted by categories of motion, looks, sound, pen, data, events, control, sensing, and operators; and the workspace, where users can drag and drop blocks to build their programs. In addition to providing a large library of characters and sounds, Scratch also allows users to design and record their own, which has contributed to the great variety of projects. The Scratch blocks support the development of many computational thinking concepts, including sequences, loops, conditionals, events, operators, parallelism, and data.

Scratch is currently used in many different settings, including schools, museums, community centers, and homes. One of the most unique aspects of Scratch is its online community, which allows users to not only share their own projects, but also favorite, comment on, or work off of others’ projects. The sharing feature encourages collaboration between Scratch users and has created a community of motivated learners who encourage and support each other. While Scratch has succeeded in many ways, it is only available on traditional graphical user interfaces and may not appeal to users with kinesthetic learning styles.

Figure 5: screenshot of the online Scratch editor. Stage area is in the top left corner, blocks palette is in the middle, and workspace is on the right.
Figure 6: a sample Scratch project that uses an under-the-sea backdrop, several sprites, and many programming blocks.
EVALUATION

Methodology

PREPARATION

Prior to designing the studies, we completed an online course on Responsible Conduct of Research for Social, Behavioral, and Education Sciences to ensure compliance with standard ethics when conducting research with human subjects. In addition, we consulted the Wellesley College Child Study Center and professors from the Wellesley College Psychology Department for advice regarding research studies with children. Although both of these resources typically focus on children of a younger age, their advice and insights provided helpful guidelines when designing the experiment. After forming an initial study protocol, we conducted two pilot studies. We then refined the protocol and appropriately modified it for each setting.

EXPERIMENTAL DESIGN

To investigate how tangible toys like littleBits and KIBO promote engagement, complexity, and collaborative learning for young children, we conducted studies in three different settings: 1) lab; 2) workshop; and 3) classroom (see Table 2). For the lab setting, a within-subjects experiment was used to compare the dimensions of learning (engagement, enjoyment, complexity, and interactivity) for each treatment (littleBits and KIBO). Studying three different settings helped us understand the differences between them.

Table 2: Description of each setting and their characteristics.

<table>
<thead>
<tr>
<th>setting</th>
<th>lab</th>
<th>workshop</th>
<th>classroom</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>child friendly laboratory space at Wellesley College</td>
<td>college classrooms at Wellesley College</td>
<td>kindergarten classroom at an elementary school in Dedham, MA</td>
</tr>
<tr>
<td>toys studied</td>
<td>littleBits, KIBO</td>
<td>littleBits</td>
<td>littleBits</td>
</tr>
<tr>
<td>analysis comparisons</td>
<td>gender (F vs M), littleBits vs. KIBO</td>
<td>age group (K-1 vs 2-3)</td>
<td>gender (F vs M)</td>
</tr>
<tr>
<td></td>
<td>other toys interacted with (no data)</td>
<td>Scratch, ScratchJr.</td>
<td>LEGO WeDo</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>--------------------------------------</td>
<td>---------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>length of session</td>
<td>3 hours (1 hour littleBits, 1 hour KIBO, 1 hour total including introduction, breaks, and post-task interviews)</td>
<td>1.5 hours (40 minutes littleBits, 40 minutes Scratch/ScratchJr, 10 minutes including introduction and reflections)</td>
<td>1 hour (1 hour littleBits)</td>
</tr>
<tr>
<td>number of sessions conducted</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>timing of session(s)</td>
<td>morning during non-school days</td>
<td>afternoon during non-school days</td>
<td>afternoon during a school day</td>
</tr>
<tr>
<td>total number of participants (n)</td>
<td>18</td>
<td>15</td>
<td>7</td>
</tr>
<tr>
<td>number of participants by gender</td>
<td>8 female 10 male</td>
<td>15 female 0 male</td>
<td>4 female 3 male</td>
</tr>
<tr>
<td>number of participants by age</td>
<td>8 in grades K-1 (3F, 5M) 10 in grades 2-3 (5F, 5M)</td>
<td>4 in grades K-1 (4F) 11 in grades 2-3 (4F)</td>
<td>7 in grade K (4F, 3M)</td>
</tr>
<tr>
<td>number of adults present per session</td>
<td>2 (1 facilitator, 1 volunteer)</td>
<td>5-8 (1 facilitator, 4-7 college student volunteers)</td>
<td>3 (1 facilitator, 2 teachers)</td>
</tr>
<tr>
<td>noise level</td>
<td>quiet</td>
<td>quiet</td>
<td>noisy</td>
</tr>
<tr>
<td>familiarity level with space and other participants</td>
<td>medium</td>
<td>low</td>
<td>high</td>
</tr>
<tr>
<td>data collected</td>
<td>pictures, videos, transcriptions, pre-task questionnaire (parents), post-task interview (users)</td>
<td>pictures, videos, transcriptions</td>
<td>pictures, videos, transcriptions, teacher interviews</td>
</tr>
<tr>
<td>full protocol</td>
<td>Appendix B.5a</td>
<td>Appendix B.5b</td>
<td>Appendix B.5c</td>
</tr>
</tbody>
</table>

We chose to conduct our studies in these three settings for several reasons; together, they cover both formal and informal learning environments for children in early elementary school, allowing us to study the differences in learning indicators across educational settings. We used
the lab setting, which we had the most control over, to study both littleBits and KIBO. The workshop setting provided a way to investigate the learning of computational thinking in a single-gender environment, and the classroom setting allowed us to study the use of technology in schools. Each environment included different affordances and challenges, and conducting user studies in each setting offered a unique opportunity to explore best practices for teaching computational thinking to young children.

The lab setting was the least constrained environment, which enabled us to collect additional data. While all parents and older participants of all settings completed the consent and assent forms (Appendix B.2), only parents of lab participants completed the pre-questionnaire containing basic demographic information and information about the child’s technology usage at home and in school (Appendix B.3). In addition, only lab studies allowed enough time to conduct post-task interviews with the child participants to assess self-reported enjoyment and other thoughts. We were also only able to assess KIBO in the lab sessions, as the number of KIBO kits available was not sufficient for the workshop and classroom studies, in which more participants were present. The medium level of familiarity for participants in the lab setting can be explained by the recruitment strategy used here; many participants were children of Wellesley College faculty members and were already familiar with other participants.

The protocol for the workshop setting differed from the original study protocol because of the increased number of child participants and presence of college student volunteers. In addition, since the Robogals workshop was part of a separate outreach initiative to teach girls about STEM, the workshop protocol was developed with a different focus in mind (i.e. inspiring more girls to pursue STEM fields by increasing workshop participants’ exposure to a variety of interesting technologies). Participants were recruited from the Boston-Wellesley area, and there were many participants who did not know the other participants.

The classroom setting was the most constrained environment. We adjusted the study protocol to accommodate the space and time that the kindergarten classroom had available. Because this user study occurred off campus, only one facilitator was present to carry out the protocol. The classroom’s two teachers were present but remained uninvolved during the study. Other non-participant students were present and engaged in other non-related activities around the classroom; although there was no direct interference with the study participants, it contributed to a relatively high noise level in the classroom.
**Basic Protocol**

Despite differences between settings, the same basic protocol remained the same for the littleBits part of the study. For the littleBits, the protocol was as follows: After a quick icebreaker activity, the facilitator explained littleBits and demonstrated how they work. Scaffolding was utilized by introducing the modules in order of difficulty. First, participants were instructed to build circuits with only the blue power and green output modules. Then, the pink input pieces were introduced, and finally, the orange wire pieces as well. Each time, participants were given a few minutes to build circuits so that they could discover the functionality of each module. After all the pieces were introduced and played with, the facilitator prompted participants to build a project using the littleBits and crafts materials, which included scissors, tape, markers, construction paper, pom-poms, confetti strips, and popsicle sticks. “Robotic animals” were given as examples, and the facilitator also asked participants to share what they made at the end.

For KIBO in the lab setting only, a similar protocol was used. Blocks were also introduced in a scaffolded manner. Participants first built programs with the basic, motion, and sound blocks only, and then the facilitator introduced one-by-one the wait for clap blocks, the repeat blocks, and the if blocks. The two pilot studies demonstrated that final projects with KIBO were less feasible in the given time, as both pilot users saw KIBO as a robot vehicle already and struggled to see it as something they could create and decorate. Participants spent the remainder of the time building and executing programs using all of the available blocks. Full protocols for each study are detailed in Appendix B.5.

**Data Analysis Methodology**

**Data Analysis Tool**

Following the user studies, all sessions were transcribed and analyzed. We used ATLAS.ti, a quantitative data analysis software, to facilitate this process ([http://atlasti.com/](http://atlasti.com/)). While the analysis was still time consuming, ATLAS.ti provided an organized way to manage all video and transcription documents. In addition, its support of document coding allowed the easy addition of codes for thorough video and dialogue analysis. Screenshots of the software are shown in Figures 7-9.
Figure 7: Screenshot of ATLAS.ti association editor; allows transcription of video dialogue to be associated with video anchors.

Figure 8: Screenshot of ATLAS.ti, coding the video using the video coding scheme.
Figure 9: Screenshot of ATLAS.ti, coding the transcription document using dialogue coding scheme.

Talk Categories

For analysis, detailed coding schemes were developed iteratively to analyze the dialogue as well as the physical gestures and expressions of participants. Most codes are related to learning, engagement, and collaboration; a few are unrelated to these dimensions of learning but are interesting to look at. Complexity was measured only for littleBits projects (KIBO analysis does not include complexity). The dialogue coding scheme classified talk into thirteen general categories, which are described in detail in Table 3. Inter-coder agreement based on 32% of the littleBits data was excellent with 93% agreement for dialogue coding. We did not assess inter-coder agreement for KIBO.
Table 3: Coding scheme for classifying talk categories.

<table>
<thead>
<tr>
<th>Category</th>
<th>Definition</th>
<th>Examples from Transcripts</th>
</tr>
</thead>
</table>
| Problem Solving   | Planning, relevant questions or facts           | B6: “Well I have a good idea. How about we put it ['wait for clap' block] at the beginning?” (KIBO)  
|                   |                                                 | G1: “I think you have to connect that to power first.” (littleBits)                       |
| Computational Thinking | Demonstrating CT concepts (i.e. sequencing, parallelism, etc.) | B7: “No, you have to start this way!” *points at the begin block and then the next few blocks* (KIBO)  
|                   |                                                 | G2: “No, you have to put the button first!” (littleBits)                                 |
| Technology        | Referring to how the technology works           | B10: “Is this how many batteries it needs?” (KIBO)                                        
|                   |                                                 | B10: “Did this just waste more battery for nothing?” (littleBits)                         |
| Magic             | Referring to “magical” concepts                 | R1: “How does it know when you’re eating candy?” (littleBits)                             
|                   |                                                 | G4: “It just senses.”                                                                    |
|                   |                                                 | G1: “It’s magic!”                                                                       |
|                   |                                                 | G4: “It senses and magic.” (littleBits)                                                   |
| Frustration       | Expressing task-related frustration             | B6: “I can’t do it!” (KIBO)                                                              |
|                   |                                                 | G4: “This isn’t working!”                                                                |
| Excitement        | Expressing task-related excitement              | G8: “Now let’s do this one!” (KIBO)                                                       |
|                   |                                                 | B2: “Look at what I made!”                                                               |
| Confusion         | Expressing task-related confusion               | G8: “Robot, spin! … huh?” (KIBO)                                                          |
|                   |                                                 | B5: “Why isn’t this lighting up?” (littleBits)                                           |
| Coordination      | Turn-taking, sharing the objects                | B6: “Can I have this one?” (KIBO)                                                         |
|                   |                                                 | G2: “Put that one there.”                                                                |
| Seeking help      | Expressing a task-related desire for help       | G8: “Can you scan it for me?” (KIBO)                                                      |
| Reflection        | Referring to previous tasks or occurrences      | G8: “But last time that didn’t work!” (KIBO)                                              |
|                   |                                                 | B5: “Oh, remember? Connect that one.” (littleBits)                                        |
| Application       | Comparing the technology to other things in the world | B7: “Hey! It’s like a grocery scanner!” (KIBO)                                            |
|                   |                                                 | G2: “The tail is actually like a little fan. Ahh. That feels good.” (littleBits)        |
| Brief Response    | Short responses to questions/suggestions        | B3: “Okay.”                                                                            |
| Disengagement     | Non-task related                               | G9: “Do you know when the last workshop was?”                                             |
Physical Gestures and Facial Expressions

Nine codes were also developed to analyze facial expressions (i.e. smiling, laughing) and physical gestures (i.e. both arms up like in a victory pose, taking and giving objects). These codes are described in Table 4. Inter-coder agreement based on 32% of the littleBits data was excellent with 90% agreement for video coding.

Table 4: Coding scheme for analyzing physical gestures and facial expressions.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Indicator of</th>
</tr>
</thead>
<tbody>
<tr>
<td>smile</td>
<td>Smiling</td>
<td>engagement (+)</td>
</tr>
<tr>
<td>laugh</td>
<td>Laughing or giggling</td>
<td>engagement (+)</td>
</tr>
<tr>
<td>armsUp</td>
<td>Raising both hands, as in a victory pose</td>
<td>engagement (+)</td>
</tr>
<tr>
<td>reach</td>
<td>Reaching out to grab something from a community pile</td>
<td></td>
</tr>
<tr>
<td>taking</td>
<td>Taking something from an area which clearly belongs to another participant</td>
<td>collaboration (-)</td>
</tr>
<tr>
<td>giving</td>
<td>One participant is giving or handing over something</td>
<td>collaboration (+)</td>
</tr>
<tr>
<td>leanFor</td>
<td>Leaning forwards (because of engagement, not because reaching for a toy)</td>
<td></td>
</tr>
<tr>
<td>turnAway</td>
<td>Head or body turned away from the task at hand/the technology toys</td>
<td>engagement (-)</td>
</tr>
<tr>
<td>imitating</td>
<td>Making sounds or gestures that imitate the technology toy (related to task)</td>
<td>engagement (+)</td>
</tr>
</tbody>
</table>

Measuring the Dimensions of Learning

We used the codes for physical gestures, facial expressions, and dialogue to measure learning, engagement, and collaboration.

**Learning** was reflected by the nature of discussion. In particular, utterances demonstrating problem solving, computational thinking, reflection, and application, suggest instances of learning.
**Engagement** was measured using physical and verbal indicators. Positive indicators associated with engagement include physical (smiling, laughing, raising both arms up like a victory pose, and imitating the technology) and verbal (talk classified as excitement) cues. Indicators associated with disengagement include physical (turning away from task at hand) and verbal (utterances of frustration, confusion, and disengagement) cues as well.

**Collaborative learning** was analyzed in a similar way. Positive collaboration was indicated by physical (giving something to another participant) and verbal (talk classified as coordination) cues, while the lack of collaboration was measured by physical (taking something from another child) cues only.

**Complexity** was assessed by assigning each littleBits project a complexity score using the following formula: (1.5 x C) + LB + S, where C is the crafts score (weight of 1 for simple materials like paper and markers, and weight of 2 for all other materials); LB is the littleBits score (weight of 1 for blue modules, 2 for green and orange modules, and 3 for pink modules which are most computationally complex); and S is the storytelling score (represented by the number of words used in the presentation of the project). The crafts score is given an extra weight of 1.5 because the addition of crafts materials better reflects complex thinking. All scores were adjusted based on the average score for all participants.

Considering the differences between settings, we did not conduct a rigorous statistical test. Quantitative indicators were calculated to give a general sense of the data, and are described in the section for each of the three educational settings.

**Settings**

We conducted studies in three different settings that varied by time, environment, and number of participants. The littleBits protocol was carried out in each setting to allow for the comparison of how learning with tangible technology toys is impacted by aspects of the environment. The lab sessions also included a study of KIBO, for which quantitative results are presented. Workshop participants engaged with other technologies such as Scratch, and classroom participants also used LEGO WeDo; qualitative observations are presented for both. Finally, the classroom research also included several teacher interviews to provide more context for understanding the integration of STEM and robotics activities into K-12 education.
LAB

Introduction

As the most controllable environment, the lab setting presented many opportunities for researching computational thinking toys. We conducted studies in our Human Computer Interaction Lab to investigate how littleBits and KIBO promote learning in an informal education setting. The characteristics of the lab setting closely resemble that of an afterschool program or summer camp, in which children spend several hours learning and playing together. These user studies help us explore how children learn computational thinking in such settings.

Participants

To analyze the potential of littleBits and KIBO in informal education settings, a within-subjects study was conducted in lab settings to compare various dimensions of learning for each toy. The sample consisted of 18 children (10 male and 8 female). All participants were in grades K-3, with 8 children in grades K-1 (5 male and 3 female) and 10 children in grades 2-3 (5 male and 5 female).

All participants were recruited by email and reside within the Wellesley area. Many were children of Wellesley College faculty members, and so some participants were already familiar with other participants. Prior to the user study sessions, parents were asked to complete a pre-task questionnaire online, which solicited information about their children’s exposure to technology. Daily interaction with technology was limited for all participants in the lab setting: more than half of the parents indicated that their child spent between half an hour and an hour interacting with technology per day, and the rest of the parents estimated less than half an hour per day. 2 of the 18 children had previously worked with littleBits, and 1 child had played with KIBO before.

Procedure

Lab sessions were conducted in a quiet laboratory space on non-school days. Participants were brought in for three hours total to play with littleBits and KIBO. One facilitator was present during all sessions, and an additional volunteer was present to provide additional help with data collection and extra supervision. The lab space is a unique environment with colorful walls, modern lighting, and chairs with wheels; this layout made for a child-friendly environment.
At the beginning of the lab sessions, the facilitator gave parents and participants a tour of the space. Once the parents departed, the participants engaged in an icebreaker activity and were notified of the agenda. They were then split into two groups based on grade level. Each group worked with either littleBits or KIBO first, and then switched after an hour. Participants worked in groups of 2-3. Within each activity, blocks and modules were introduced in a scaffolded manner; halfway through the activities, participants were given 5 minute breaks to draw, go to the bathroom, or walk around the lab space. Snacks were provided between activities. Following the activities, participants were interviewed one at a time by the facilitator about their enjoyment. The full protocol for the lab sessions, including the questions asked in the post-task interview, is described in Appendix B.4.

![Lab setting image](image)

Figure 10: Lab setting.

**Analysis**

**Learning.** For littleBits, the nature of discussion was comparable for both genders, except for utterances related to excitement and coordination. Male participants displayed more excitement through their discussion, while female participants had more talk categorized as coordination. However, all participants engaged in discussion demonstrating problem solving, computational thinking, reflection, and application, suggesting the presence of learning in the lab setting with littleBits.
For KIBO, lab participants also had utterances related to problem solving and computational thinking; however, the amount of reflection and application utterances is noticeably lower. With KIBO, females expressed more excitement while males engaged in more discussion about coordination.
Users expressed significantly more frustration with KIBO than with littleBits. Several children made comments such as “The scanning is too hard” and “Scanning is very hard, I always thought it was easy.” The difficulty of scanning the blocks also prompted users to ask for more help with KIBO, whereas there were no utterances for help with littleBits. However, KIBO still had a comparable amount of utterances related to excitement, which suggests that it still has much to offer despite being frustrating for users. Surprisingly, the frustration of scanning also contributed to additional excitement with KIBO; many users made utterances of excitement when they were able to scan a sequence of blocks successfully. Utterances of excitement with KIBO were also common when participants built programs that were either “long” or “annoying;” comments such as “make it sing forever!” elicited several enthusiastic responses.

On the dimension of learning, littleBits seemed to produce more utterances related to problem solving, reflection, and application, while KIBO fared better with computational thinking. littleBits, as modules that can be used to construct a variety of different objects, are more open-ended and can better support problem solving, reflection, and application. KIBO, on the other hand, consists of blocks that are more focused on computational thinking concepts and are less applicable to the real world.

![Figure 13: Distribution of talk categories per toy for all lab participants.](image)

Figure 13: Distribution of talk categories per toy for all lab participants.
**Engagement.** While all lab participants were somewhat engaged while playing with littleBits, male participants exhibited more verbal indicators of positive engagement, as measured by the amount of talk classified as excitement. Female participants appeared less engaged with a higher percentage of physical indicators associated with negative engagement. They were more likely than male participants to turn away from the task at hand.

![Figure 14: Engagement components per gender for all lab participants while using littleBits.](image)

With KIBO, male participants in the lab setting actually demonstrated more negative engagement than positive engagement. Levels of positive engagement were slightly less for male participants relative to female participants, while levels of negative engagement were similar among genders.

![Figure 15: Engagement components per gender for all lab participants while using KIBO.](image)
Figure 16 shows the comparison of engagement components for littleBits and KIBO. While both toys produced similar results for positive engagement, KIBO saw higher levels of disengagement with users in the lab. KIBO had a greater amount of both physical and verbal indicators associated with negative engagement, resulting in a fairly low amount of total engagement.

![Comparison of engagement components for littleBits and KIBO.](image)

Figure 16: Comparison of engagement components for littleBits and KIBO.

**Complexity.** All participants in the lab setting were able to demonstrate some level of complexity using littleBits, as shown in Figure 17. Female participants achieved a higher average craft score, while male participants scored better with the littleBits modules. The storytelling score was similar for participants of both genders. Figure 18 shows examples of littleBits projects created in the lab setting.
Figure 17: Complexity score components per gender for all lab participants.

Figure 18: Sample littleBits projects created by lab participants. Left: “cat” created by two girls (K, 1st); middle: “the kitty project” created by one girl (3rd); right: “the helicopter” created by one girl (3rd) and one boy (3rd).

**Collaboration.** With littleBits, female participants demonstrated more verbal indicators of positive collaboration than male participants. They also exhibited more indicators of negative collaboration, but achieved higher overall collaboration scores. We observed that female participants were more likely to have utterances of coordination and also more likely to grab objects from other participants. Male participants exhibited less indicators of both positive and negative collaboration.
KIBO produced favorable results with collaboration between lab participants. Female and male participants engaged in talk classified as coordination, and some physical collaboration occurred with male participants. Collaboration indicators were comparable for both genders, with participants exhibiting quite a bit of coordination talk. We rarely observed participants grabbing blocks from other users, which may be explained by the larger size of the KIBO pieces.

Figure 19: Collaboration components per gender for all lab participants while using littleBits.

Figure 20: Collaboration components per gender for all lab participants while using KIBO.
Figure 21 shows the collaboration components for both littleBits and KIBO in the lab setting. For all components, participants demonstrated similar levels of positive and negative collaboration with littleBits and KIBO. This suggests that littleBits and KIBO both support collaborative learning for child participants in the lab setting.

Figure 21: Comparison of collaboration components for littleBits and KIBO.

**Enjoyment.** In the lab setting, we also measured enjoyment for the child participants. During the post-task interview (see Appendix B.4), participants were presented with the children’s Likert scale [33] and asked three questions, how they felt: 1) during the entire session; 2) while playing with KIBO; and 3) while playing with littleBits. Participants were asked to point at the face they felt most reflected their feelings, and these quantitative measures (1 corresponding to extremely unhappy, 5 corresponding to extremely happy) are reported in Figure 22 for all participants in the lab setting. All participants appeared to enjoy themselves during the sessions, and KIBO received a slightly lower enjoyment score than littleBits. The post-task interviews also revealed some qualitative answers, eliciting comments from participants such as: “I didn’t like KIBO because it was hard to scan” or “My favorite part was littleBits because you can keep making stuff.” The nature of the comments reflected the quantitative enjoyment scores; many participants said that KIBO was their least favorite activity of the session, while only one participant said they did not enjoy littleBits. When asked about their favorite part of the session,
many children referred to their littleBits projects. There were also several participants that cited “everything” as their favorite part.

![Graph showing enjoyment scores](image)

**Figure 22**: Enjoyment scores for lab participants overall and while using KIBO and littleBits.

Figures 23 and 24 show enjoyment scores for each toy by grade level and gender. For overall, KIBO, and littleBits, female participants self-reported higher enjoyment scores. It is unclear, however, whether these findings indicate that the female participants actually had a more enjoyable time playing with the toys or whether they are the result of gender differences (i.e. perhaps boys are less likely to express their enjoyment). Because of the ambiguity, the engagement components we previously reported are a more reliable measure of each participant’s experience.

Enjoyment scores by grade level show mixed results. Older participants enjoyed KIBO more, while younger participants enjoyed littleBits more. This is unexpected, as KIBO was designed for a younger audience (ages 4-7) and littleBits was designed for a slightly older audience (ages 8+). These findings suggest that both KIBO and littleBits can be appropriate learning toys for children in early elementary school.
Discussion

The results of the lab setting demonstrate that both littleBits and KIBO provide children with additional opportunities to engage in computational thinking and play. Both elicited a high amount of discussion surrounding problem solving, computational thinking, excitement, and coordination. However, participants using KIBO had less utterances categorized as reflection and more utterances categorized as frustration, seeking help, and disengagement. This result may be
explained by the fact that littleBits has been in the commercial market longer than KIBO has; KIBO is still somewhat prototype-like, and the technical issues may have caused the extra frustration and confusion. This also suggests that maybe users need more support and guidance when using KIBO in order to have effective learning experiences. The engagement scores also reflected this finding. Participants were more likely to be engaged when using littleBits than when using KIBO, although positive engagement scores were similar for both toys. This suggests that while users exhibited more disengagement with KIBO, KIBO still elicited a comparable amount of indicators for positive engagement.

Complexity was only measured for littleBits; complexity scores demonstrate that all lab participants were able to create complex projects with lengthy stories. Female participants were more likely to include crafts materials in their littleBits projects than male participants were, but male participants used more complex littleBits modules to build their projects. The collaboration results were comparable for both toys, and enjoyment scores were slightly higher for littleBits than for KIBO. These results are further analyzed in the “Implications for Design” section.

The user study sessions in the lab has several limitations that require future work. First, it is necessary to conduct longitudinal studies with both toys in order to assess long-term learning and more closely resemble an afterschool program or summer camp, both of which are recurring events. Second, it might be useful to include additional dimensions of learning, such as creativity. We only measured complexity for littleBits, but comparing complexity for both toys would be helpful. In addition, future work should consider measuring learning itself by conducting a pre-test and post-test and assessing the difference of scores. While many participants used dialogue demonstrating computational thinking, this does not imply that they understand and can effectively apply the concepts. Finally, although all participants had limited technology interaction, they come from similar socioeconomic backgrounds. Future studies should investigate the use of technology toys with a more diverse sample.

Summary

The results demonstrate that all participants in the lab setting were able to engage in learning while using littleBits and KIBO. Their nature of discussion indicates that both toys elicited utterances in reflection, coordination, problem solving, and excitement. Participants were engaged while playing with both toys; however, children were more disengaged while using
KIBO. This result is consistent with the nature of discussion, which demonstrated a higher rate of utterances categorized as frustration and confusion. This finding, in addition to the many comments along the lines of “scanning is so hard,” suggests that children need toys that are reliable and easy to use. littleBits and KIBO provided opportunities for children to collaborate, and littleBits allowed children to create complex projects. Overall, these results support the idea that littleBits and KIBO can engage children in play and allow them to learn computational thinking skills.

WORKSHOP

Introduction

Researchers have suggested that the integration of STEM into K-12 schools requires a significant amount of effort for school teachers. This insight, coupled with the increasing importance of teaching computational thinking, has led to the development of many initiatives and programs to help our youth develop computational thinking skills. Programs like Girls Who Code, Techbridge, and various tech camps offer opportunities for children to learn with others. In addition to helping children learn and make new friends, these programs also make technology tools accessible to a more diverse population. Last year, we helped start a Robogals chapter at Wellesley College to expose young girls in the local area to robotics and technology. We collected data at two of the Robogals workshops to study the learning potential of littleBits and similar technology tools in a workshop setting.

Participants

To analyze the potential of littleBits in informal education settings, data was collected during the littleBits session of Robogals workshops. The sample consisted of 15 children in Robogals workshops (15 female). All participants were in grades K-3, with 4 children in grades K-1 and 11 children in grades 2-3.

Robogals is an international organization that aims to increase the number of talented young women in the field of engineering; Wellesley College started its chapter in 2014, and this research utilized its first two workshops in December 2014 and February 2015. The workshops were open to girls in grades K-5, and there were approximately 15 participants at each workshop,
most residing within the Wellesley-Boston area. Data was collected for all participants; however, due to the low quality of some of the video-recordings, data is only reported here for 15 of the participants that were in grades K-3.

**Procedure**

Both workshops were conducted in quiet college classrooms on weekends. Each workshop was an hour and a half long, including an introduction, two 40-minute activities, and reflections. Several college student volunteers were present at each workshop to provide additional support. All volunteers were trained by the facilitator during a one-hour training session that occurred one week prior to each workshop. They were given short demos of littleBits and Scratch, and spent approximately twenty minutes playing with each technology in pairs. The facilitator for all user study sessions also serves as the training manager for Robogals Wellesley, and was solely responsible for designing and executing the training sessions and workshops.

The workshops were designed to introduce young girls to the technologies and to give them the opportunity to build something they felt proud of. Each workshop began with a brief introduction to robotics and an icebreaker activity for all participants and volunteers. Then participants were split into two groups for 40-minute activities. One group used Scratch and ScratchJr. first, where participants used the child-friendly programming platform to build simple projects. The other rotation was the littleBits activity, which followed the same basic protocol used in the lab and classroom studies. It is important to note that some participants played with Scratch first, while others played with littleBits first. Participants worked in groups of two to four within the littleBits activity. At the end of the workshop, all participants were asked to reflect on their experience at the workshop and share what they had learned. The detailed protocol for the workshop setting is described in Appendix B.5b. Figure 25 shows the layout of the workshop setting, with four workshop participants and two college student volunteers actively engaging with them.
Analysis

Learning. The nature of discussion shows that workshop participants engaged in quite a bit of problem solving and reflection. While older participants had more utterances related to problem solving and technology, they were generally mellower than younger participants, displaying less talk categorized as excitement and coordination. Participants had relatively fewer utterances related to computational thinking and application.

Figure 26: Distribution of talk categories per school grade for all workshop participants.
**Engagement.** Younger participants at the workshops appeared significantly more engaged. They showed more verbal and physical indicators associated with positive engagement, while older participants displayed more physical indicators associated with negative engagement, as characterized by turning away from the task at hand. These discrepancies may be explained by differences in mindset and behavior at certain ages; the younger girls may be less afraid to express their engagement, not caring about what others think yet, while the older girls may have a tendency to hide their true emotions for fear of being judged. The younger participants also displayed few indicators of disengagement, suggesting that perhaps, the littleBits are more appealing for girls in grades K-1 than girls in grades 2-3.

![Engagement components per school grade for all workshop participants.](image)

**Complexity.** Complexity scores were quite similar for both age groups in the workshop setting. All participants showed a good balance of incorporating crafts and littleBits modules in their projects, and older participants demonstrated an increased complexity by telling slightly longer stories about their projects. This aligns well with what one would expect; as children grow older and expand their vocabularies, they should be able to tell longer and more detailed stories.
Figure 28: Complexity score components per school grade for all workshop participants.

Figure 29: Sample littleBits projects created by workshop participants. Left: “Chocolate Chip the Dog” created by two girls (K, K); middle: “the rattlesnake” created by two girls (3rd, 3rd); right: “the dog” created by four girls (2nd, 3rd, 3rd, 3rd).

**Collaboration.** All workshop participants engaged in collaborative learning while using littleBits. The results demonstrate that younger participants showed more verbal indicators associated with positive collaboration. None of the workshop participants showed any evidence of negative collaboration, which was measured by the occurrences of participants taking objects from other participants. This can be explained by both gender and the environment of the workshop settings; the workshop participants worked with others they had not met before, and were perhaps more likely to exhibit polite behavior (i.e. asking for objects instead of just grabbing them) because of it.
Discussion

These findings suggest that littleBits promoted learning, engagement, complexity, and collaboration in the workshop setting. Participants engaged in discussion related to problem solving and reflection, and demonstrated many physical and verbal indicators associated with engagement. They also used littleBits to build projects with some level of complexity, and were able to tell stories to accompany them. Moreover, all workshop participants engaged in collaborative learning with other users, demonstrating that littleBits, when used in this particular environment, can provide opportunities for girls to collaborate with others.

This study shows that littleBits can help girls learn and collaborate in a workshop setting; however, it has some limitations that require future work. Although the volunteers received training from the facilitator, their input and guidance for workshop participants varied. Volunteers also represented a range of personalities and engagement during the workshop, which may have altered the number of utterances for some participants. Future studies might give stricter guidelines for the type and frequency of feedback that volunteers should provide. In addition, since Robogals Wellesley was developed only last year, all workshop participants were first-time participants. However, in many technology-related workshops and programs, children are participating on a weekly or monthly basis. It would be necessary to also study how these learning indicators change during participants’ second or third workshop experiences.
Summary

These findings support the idea that littleBits can help girls in early elementary school engage in play and learn computational thinking skills. Workshop participants engaged in discussion related to problem solving and reflection, and also showed indicators demonstrating that their engagement and ability to collaborate while using littleBits. In addition, the littleBits allowed participants to build projects that incorporated a balance of crafts materials and littleBits modules. Ultimately, these results demonstrate that littleBits can be used at workshops to help young girls develop computational thinking skills and engage in collaborative learning.

CLASSROOM

Introduction

In the last decade, schools have already begun exploring using computational thinking and robotics activities in the classroom. Some schools have integrated technologies such as Scratch, LEGO WeDo, and even littleBits into their curriculum. Formal education settings present an opportunity to help children learn computational thinking skills that they can apply in other areas, including math, science, and even language arts. Classrooms are natural environments for learners to collaborate with others, and afford the ability to incorporate computational thinking into topics that may appeal to students with diverse interests and learning styles. We conducted a study with seven children in a kindergarten classroom to study the potential of littleBits as a learning tool for computational thinking in early elementary school, and also interviewed seven teachers to better understand teaching computational thinking in the classroom.

Participants

To investigate the potential of technology toys like littleBits in formal education settings, a within-subjects study was conducted in a kindergarten classroom at an elementary school in Dedham, MA. The sample consisted of 7 children total (3 male and 4 female, all K). The facilitator had worked with the particular kindergarten classroom two months prior to the study as a robotics expert during their LEGO WeDo activity, so students were already familiar with the facilitator.
All students had previous exposure to technology through the LEGO WeDo projects, but their level of expertise with technology varied. In the LEGO WeDo activity, students were each given a LEGO WeDo kit and a laptop computer. During six hour-long sessions, the children were introduced to robots and the engineering design process, and then asked to build any creation of their choice that utilized one motor piece. Some children built car-like projects, while others mounted the motors on top and attached gears and other LEGO pieces to build spinning dreidels and “shavers.” Many students also incorporated sound and text on the software platform of LEGO WeDo.

For the second part of the classroom setting study, seven teachers were interviewed (1 male and 6 females). The male is a 4th grade lead teacher; three females are kindergarten lead teachers; and three females are kindergarten assistant teachers. All interview participants teach at the same elementary school in Dedham, MA.

**Procedure**

The classroom study was one-hour long and held during the afternoon of a school day. It occurred during their “choice” time, when students can select from a variety of activities to pursue in the classroom. The facilitator introduced littleBits to all students, and those who were interested participated in the study. Other non-participant classmates were present but engaging in unrelated activities. This made the classroom setting quite loud relative to the workshop and lab settings. There were also two teachers present in the classroom, although they did not intervene during the study. Students worked at their child-sized tables and chairs. Only one facilitator was present, and all participants worked in pairs, with the exception of one child who worked alone.

Since the child participants were already familiar with the facilitator, this protocol did not include an icebreaker activity. Students proceeded with building littleBits circuits, with the type of modules introduced one at a time. Following the exploration time, students were tasked with building littleBits projects together and presenting their creations at the end. A detailed study protocol can be found in Appendix B.5c.
In addition to studying how the elementary school students interacted with littleBits in the classroom setting, several teachers were interviewed about robotics education. Each interview lasted approximately 10 minutes and consisted of 8 questions regarding their experience teaching robotics in the classroom, comfort and preparation level of doing so, challenges they face with incorporating robotics, and overall impressions of robotics activities (Appendix B.6). The answers gave many insights into the value of robotics and computational thinking in the classroom, and the type of support and materials that are necessary to successfully integrate STEM into K-12 education.

**Analysis**

**Learning.** The nature of discussion of the classroom participants suggests a high amount of problem solving, computational thinking, reflection, and application for both genders. Female users appeared to engage in more computational thinking, while male users showed a higher amount of application. All students showed utterances of reflection, demonstrating that littleBits facilitated learning in the classroom setting.
Engagement. We observe comparable patterns for engagement with both genders in the classroom. Male participants exhibited slightly more verbal indicators of positive engagement, often making comments such as “Whoa!” and “Look at this!” Meanwhile, female participants displayed more physical indicators of negative engagement, as they appeared to be more easily distracted by other non-participant classmates in the area. Overall, all children appeared engaged while playing with littleBits, although they were subject to many distractions in the classroom.
**Complexity.** Complexity was better expressed by females than males in the classroom setting. The male participants integrated less craft materials into their projects, while female participants were eager to decorate their littleBits modules. Females also outperformed males in the technical complexity of their projects as well as the storytelling component. In general, while all participants were able to create unique projects using littleBits, female users produced more complex creations.

![Figure 34: Complexity score components per gender for all classroom participants.](image)

Figure 34: Complexity score components per gender for all classroom participants.

![Figure 35: Sample littleBits projects created by classroom participants. Left: “bridge” created by two girls (K, K); right: “massager” created by one boy (K).](image)

Figure 35: Sample littleBits projects created by classroom participants. Left: “bridge” created by two girls (K, K); right: “massager” created by one boy (K).

**Collaboration.** Discrepancies between genders were most obvious with collaboration. Male participants exhibited no verbal indicators associated with collaboration, while female
participants engaged in quite a bit of coordination talk. The levels of physical indicators associated with positive and negative collaboration were comparable for both genders.

![Image: Collaboration components per gender for all classroom participants.]

**Figure 36: Collaboration components per gender for all classroom participants.**

**Teacher Interviews.** We conducted teacher interviews at the elementary school in order to better understand their experiences of integrating robotics into their classroom curriculum. Several themes emerged from the results of these interviews: teachers believed that 1) robotics activities had many benefits for the students, and 2) robotics presented many challenges that prevent more activities from occurring in the classroom. 6 of the 7 teachers who were interviewed were female, kindergarten teachers; 1 was a male, fourth grade teacher. All teachers had three or four years of experience doing robotics in their classroom.

Teachers cited several benefits that robotics activities provide for the students. In addition to exposing students to the engineering design process, robotics activities also help students the importance of making mistakes. According to teachers, the hands-on aspect of robotics also helps students develop fine motor skills and is “more fun” relative to other class activities. The exposure to technology is also beneficial, and many teachers were excited by the additional opportunities for students to be engaged and creative. During robotics activities, students demonstrated more independence as they practiced trial-and-error problem solving techniques.

Despite having conducted robotics activities for 3+ years, few teachers felt fully prepared to lead robotics activities. Some challenges teachers mentioned include not having enough adults present for support, not having enough time throughout the school year to do robotics activities
more than once, the integration of activities into curriculum requiring too much time and effort, and not being entirely familiar with the robotics materials (i.e. the hardware and the software). When asked about potential solutions to increase their comfort and preparation level, many teachers suggested having “more hands on deck” by bringing in expert volunteers and trained parents. Only two teachers wanted teacher time to play with the materials, and one teacher called for a guidebook to clarify technical issues and help establish project themes.

Discussion

Overall, the results demonstrate that littleBits promoted learning, engagement, complexity, and collaboration in the kindergarten classroom. Participants engaged in discussion related to problem solving, computational thinking, reflection, and application, suggesting the presence of learning. In addition, despite a loud environment, they appeared engaged with the activity. The littleBits also allowed students to create projects that demonstrated complexity, more so for female users than for male users. All participants were able to collaborate with each other using littleBits, with female students demonstrating more verbal collaboration.

While this study demonstrates the possibility of using littleBits in the classroom, it has several limitations that require future work. The results of this study may have been impacted by the loud environment of the non-participant classmates. In an ideal classroom setting, all students would be engaging in the same activity, which would reduce the likelihood of outside distractions and noise. Future studies should also investigate the potential of littleBits as a learning tool within other curriculum topics; instead of prompting students to build any creation, it might be useful to provide more structure for a littleBits activity in order to relate it to current curriculum. For instance, teachers might ask students to use littleBits to build something within a certain theme (i.e. “things you find around your house”) or with certain constraints (i.e. assigning prices for each module type and encouraging students to think about the purpose of each module they incorporate). This may expand the potential for littleBits to be used multiple times within the same academic year.

The teacher interviews provided helpful insight into what teachers want and need. All teachers mentioned several benefits of doing robotics in their classroom, and many expressed that the lessons students learn through robotics are applicable in other domains. Their overall
impressions demonstrated that doing robotics activities presented additional challenges, the biggest one being a lack of support. Even for teachers who felt fully prepared to teach robotics activities, they still wanted more experts in the classroom. These results demonstrate that integrating robotics activities into K-12 education requires more support, and future research should investigate how this support can best be provided.

**Summary**

This study demonstrates the potential for littleBits to be used in the classroom as a learning tool that promotes learning, engagement, complexity, and collaboration. Classroom participants engaged in discussion related to problem solving, computational thinking, and reflection, and remained engaged for the duration of the activity. In addition, students were able to use the littleBits and crafts materials to create different projects. The littleBits also provided classroom participants with an opportunity to collaborate with others. Future studies should investigate the potential for littleBits to be used within specific constraints or themes. The teacher interviews suggest that teachers would appreciate additional support when integrating robotics into K-12 education.

**Conclusions**

**Comparison of Results**

The evaluation of littleBits in three formal and informal environments allows for an investigation of how each setting differentially affects engagement, complexity, and collaboration. Studies in each setting varied by time, environment, number of participants, and physical layout of the space, all of which may either create affordances or challenges for teaching computational thinking.

**Learning.** The learning of computational thinking using littleBits was measured using the nature of discussion for all participants. Utterances demonstrating problem solving, computational thinking, reflection, and application suggest the presence of learning. Figure 37 shows the proportional distribution of talk categories per settings. These results suggest that while users in the lab and workshop settings tend to engage in more reflection, problem solving, excitement, and coordination than users in the classroom, classroom participants showed the
least amount of verbal disengagement, despite being in the loudest environment. Classroom participants were also more likely to express task-related confusion, perhaps because of their heightened sense of comfort in a supportive and familiar environment. Overall, these results demonstrate that littleBits allowed all participants, regardless of setting, to engage in computational practices and perspectives.

![Figure 37: Distribution of talk categories per setting for all participants.](image)

**Engagement.** Engagement was measured using physical and verbal indicators. Figure 38 shows the engagement results by setting. The classroom setting showed the lowest level of total engagement. Perhaps classroom participants were less physically engaged because they had already spent several hours in school by the time the study was conducted, whereas the lab and workshop studies took place on non-school days. Alternatively, the high levels of positive engagement expressed both verbally and physically by children in the lab and workshop settings may reflect heightened engagement because of the opportunity to make new friends. Overall, all users appeared engaged while playing with littleBits.
**Complexity.** Complexity was assessed by assigning each littleBits project a complexity score. The score is represented by the formula $(1.5 \times C) + LB + S$, all components are calculated as follows and adjusted relative to the average: $C$ is the crafts score (weight of 1 for simple materials like paper and markers, and weight of 2 for all other materials); $LB$ is the littleBits score (weight of 1 for blue modules, 2 for green and orange modules, and 3 for pink modules which are most computationally complex); and $S$ is the storytelling score (represented by the number of words used in the presentation of the project). Figure 39 shows the results of complexity score components for each setting. Complexity was best expressed in the lab setting. All projects were fairly similar in the technical and crafts aspects; the main difference was that the stories of lab participants were lengthier and more detailed. The low complexity scores for workshop users can likely be attributed to the availability of other activities (eagerness to move on leads to shorter, simpler stories) and the presence of more adult volunteers (increases shyness).
**Collaboration.** Finally, to understand how littleBits promotes collaboration, video recordings were analyzed to identify verbal and physical indicators associated with collaborative learning. Figure 40 shows collaboration components by settings. Collaboration was most common for participants in the workshop and classroom settings. This may be explained by age and gender differences; workshop participants were all female, while classroom participants were all in kindergarten. Perhaps users are more likely to collaborate with others like themselves. The workshop users exhibited no behavior associated with negative collaboration. The low familiarity level of workshop participants may have led them to be more polite, as is reflected by the high level of verbal indicators associated with collaboration. In other words, instead of taking objects from other users, they were more likely to ask other users to give them pieces. In addition, the classroom participants worked with users they were already familiar with, and thus had little need to verbally communicate with other users. Moreover, those users may be accustomed to working independently in the classroom and have a more difficult time transitioning to collaborating with other users.
IMPLICATIONS FOR EDUCATORS

Conducting studies in three learning environments allowed us to observe differences of how users learn with littleBits across educational settings. Our findings indicate that littleBits promotes learning; however, the dimensions of learning are expressed at varying levels for each setting. This suggests that the child’s environment impacts their learning while using computational thinking technology toys. Here, we provide recommendations to help educators better support the development of computational thinking skills for children in early elementary school.

1. **Physical space**: each setting facilitates a different learning experience. In general, the space should be welcoming and inviting; the familiarity of the space may influence children’s style of collaboration and willingness to express confusion.

Participants in the lab exhibited the most excitement, which may be accounted for by the novelty and modern aesthetics of the space. The room contained one glass wall and three other walls that were covered with whiteboard material, while the chairs were all on wheels. This latter aspect of the environment led to some disengagement, as some
children enjoyed rolling around in their chairs. Workshop participants were in a similar space during the workshops, and the environment presented no obvious challenges or affordances. In the classroom, children were seated in child-sized chairs at child-sized tables. Their familiarity with the space may have led to the increased amount of utterances of confusion, as participants perhaps felt comfortable enough to acknowledge their confusion. We also consider whether the familiarity may have impacted the amount of verbal collaboration that occurred. For instance, since the classroom participants were paired with classmates that they interacted with on a daily basis, perhaps they did not feel the need to politely ask for objects, which was far more common in the workshop setting. However, the amount of verbal collaboration may also be related to the age of participants. Ultimately, the learning of computational thinking can be effectively facilitated in various environments. All settings allowed children to engage in learning, complexity, and collaboration, and other factors are more likely to have impacted these indicators than the physical space itself.

2. **Other learners**: learning best occurs when participants consist of many similar-aged children. For girls, it is also beneficial to conduct activities in single-sex settings.

The workshop setting demonstrated the highest levels of collaboration and engagement. This supports existing research [36] that claims that girls tend to learn technology-related topics more effectively when learning with other girls. It is important to continue initiatives such as Robogals, Girls Who Code, and Techbridge, which seek to advance women in technology by educating girls in single-sex settings. These environments help girls feel more comfortable and confident, which enhances their ability to learn and improves their attitudes towards technology fields. In addition, the heightened level of disengagement in the classroom setting suggests that the presence of other non-participant children may decrease the quality of learning because of the added distraction. All children present in the area of the activity should be actively engaged in the activity in order to minimize distractions that detract from learning. Moreover, many participants in the lab expressed excitement when sharing their creations or discoveries with nearby participants. The presence of similarly-aged children can contribute to more learning and
more excitement [36]. Future work should study how children learn computational thinking in settings of fewer children or individually.

3. *Adults present:* too many adults may hinder children’s expression of creativity and complexity, but the presence of adult support also contributes to learning by increasing the amount of reflection. Adult involvement seems to be more useful in the classroom than in other settings.

The workshop setting had considerably more adults present, which may have resulted in the lower complexity scores of the workshop participants. These low scores were primarily due to the shorter length of the stories describing their creations. Participants may have been unwilling to tell more complex stories because the greater number of volunteers present may have caused additional anxiety or shyness for some children. However, the workshop participants also demonstrated the highest level of engagement, perhaps because the lower student-to-volunteer ratio allowed volunteers to provide more support to the participants. Workshop participants had more utterances of reflection, which may be explained by the involvement of volunteers, who prompted participants to explain their thoughts and actions. The teacher interviews from the classroom setting also provided insight into adult involvement with computational thinking activities. Many teachers expressed that having expert volunteers and “more hands on deck” would help address the main challenges of conducting robotics activities in the classroom. A few teachers even suggested training parents with the robotics materials so that adults could help facilitate such activities. Future studies might explore these ideas to investigate how having trained parents or additional expert volunteers in the classroom would affect the students’ learning experiences.

4. *Time:* the optimal duration of a computational thinking activity in a group setting depends on the age of participants and the setting in which it is conducted, but seems to be around an hour and a half. Participants of longer studies exhibited more disengagement towards the end, while participants of shorter studies demonstrated lower complexity levels. If
possible, activities should also be held on non-school days when children are most able to engage in learning.

As expected, results demonstrate that users experienced the highest complexity levels as well as relatively low engagement levels in the lab setting, which had the longest duration. Lab participants were able to tell more complex stories than participants in the classroom and workshop settings, suggesting a greater capacity for complexity when given additional time to build and think through a project. However, the computational thinking activity should be no longer than an hour and a half, as most children started to show signs of disengagement after that. It might be useful to explore recurring sessions or day-long sessions with several unplugged activities or breaks in between. In addition, children exhibited the highest levels of engagement in the workshop and lab settings, which were both held on non-school days. This may be affected by the presence of other non-participant children in the classroom setting; however, the classroom participants demonstrated the least amount of verbal disengagement. Their lack of discussion about coordination, reflection, excitement, and application may be an indicator that they were tired from a full day in school. If possible, workshops and activities to help children learn computational thinking skills should be conducted on non-school days so that children can better engage in learning and collaboration.

**IMPLICATIONS FOR DESIGN**

Conducting lab user studies with both littleBits and KIBO allowed us to compare the potential for each toy to help children learn computational thinking. Our results demonstrate that both toys prompted user discussion related to computational thinking, problem solving, reflection, application, excitement, and coordination. littleBits and KIBO are technology toys that can engage lower elementary school children in computational thinking during play; however, the differences in learning indicators across toys suggest that while some characteristics of the toys enhance the child’s learning experience, other characteristics might cause a confusion and occasional frustration for users. Here, we synthesize our results and observations from the user studies to provide suggestions for the design of technology toys.
1. **Combining physical and digitally-enhanced objects:** digitally-enhanced objects appeared to elicit more smiles and laughs for all users, and thus led to higher levels of excitement and engagement. The presence of digitally-enhanced objects can improve children’s engagement.

   littleBits consist of all digitally-enhanced objects, while the KIBO set is made up of one digitally-enhanced object (the KIBO robot) and several physical objects (the wooden blocks). Child participants were noticeably more engaged when playing with the digital aspects of both toys. With KIBO, participants demonstrated more indicators of excitement when running their programs; many users engaged in a pattern of building one program and executing it several times. With littleBits, participants remained engaged and often commented on the outputs (i.e. “Look at the lights!”) as well as the inputs (i.e. “Look what this does!”). This issue also relates to active and passive tokens. Users interacted with active tokens more often than with passive tokens, which suggests that the immediacy of the feedback impacts child excitement and engagement.

2. **Ease of manipulation:** despite differences in ease of manipulation, both littleBits and KIBO produce similar results with collaboration and excitement. Toys should be either easy to manipulate or have token and constraints as cues to guide user interaction with the system.

   Despite the tokens and constraints utilized by both littleBits and KIBO, participants experienced more difficulty with connecting littleBits modules [45]. The littleBits pieces were relatively small and had to be aligned a certain direction in order for the magnetism to work. Each piece has an X etched into it on both ends to help the user make the connections, but few participants noticed the icons. They were eager to build circuits and instead exhibited a pattern of rotating the pieces in different ways until a successful connection was made. littleBits supported this trial-and-error activity, and users were quickly able to recognize if the pieces were properly put together because of the haptic feedback that the magnetic aspect provided. The magnetic connections were a constraint
employed by littleBits to prevent users from building bad circuits. KIBO, on the other hand, was easier to manipulate. The blocks were larger in size, and the peg-style connections made it easy for participants to build programs. KIBO employed a more obvious token and constraint system, and it was clear to users that blocks had to be oriented so that pegs were aligned with holes. With these differences between littleBits and KIBO, we would expect to see more frustration with littleBits. However, the results demonstrate that participants using littleBits experienced less frustration and instead engaged in more problem solving to figure out how to put modules together. In addition, results demonstrate that both toys allow children to engage effectively in collaboration. littleBits and KIBO support collaboration by including multiple points of entry, meaning that any user can easily join the activity by adding blocks or modules. Ultimately, as long as manipulation is well-supported, the toy can successfully engage children and provide an opportunity for collaboration.

3. Error diagnosis and recovery: users with littleBits and KIBO demonstrated a need for better error-handling methods.

While all errors were handled with the help of the facilitator during user studies, it is important that the physical design of the toys directly supports error diagnosis and recovery. Different types of errors for KIBO included the robot not responding to a button press, users forgetting to re-scan a program after adding or changing blocks, and incorrect programs (such as having a repeat block with no end repeat block). In most cases, participants were stumped by the errors and sought help from the facilitator. For littleBits, participants encountered issues with connecting the modules and placing modules in the wrong order (such as putting pink modules at the end of the sequence, in which case the user would not be able to tell what the pink module does). Participants easily resolved the first error through trial-and-error problem solving, and they often did not recognize the second case as an error and simply ignored it. Error diagnosis and recovery is one of the ten usability heuristics for user interface design, and is especially critical for interfaces for children [30]. littleBits and KIBO both take some measures to minimize user error; for example, both starting objects (the power module and the begin
block) are designed in a way that prevents other objects from being placed before them. However, when errors do occur (i.e. for littleBits, when pink modules are placed at the end of the sequence; for KIBO, when if or repeat blocks are used without the end-if or end-repeat blocks), users are rarely aware of them. littleBits programs still work with errors, but learning opportunities are missed when child users are not prompted to think about what the program is doing. The misplacement of modules are simply ignored. With KIBO, although the KIBO robot makes an error sound when scanning a misplaced block, the sound is not highly distinguishable from the other “beep” and “sing” sounds made by the robot. Few users responded to error sounds, and most children kept scanning but sought help when it “didn’t work.” In addition, while KIBO had measures in place to help users recognize errors, users expressed frustration with KIBO’s error recovery process. Many children built long programs and did not realize that there was an error until their attempt to run the program failed. Not knowing where in the program the KIBO robot had scanned successfully up to, children had to spend additional time scanning the entire program from the start. By providing users with more feedback, KIBO would perhaps be a less frustrating experience.

4. *Attractiveness*: while users did not comment on the general appearance of the toys, they demonstrated a high level of excitement when the toys lit up, made sounds, or moved. These features contribute to the general attractiveness of the toys and can enhance the user’s experience.

It is difficult to assess how attractiveness of a toy impacts the user’s learning experience. Although the studies demonstrated that both toys can serve as appropriate learning toys for young children, they were originally designed for different age groups. littleBits, partially because the small modules present a choking hazard for extremely young children, is aimed at anyone ages 8+. KIBO, on the other hand, is for children ages 4-7. This may account for the differences in size and appearance; KIBO’s larger wooden blocks resemble the toys of preschoolers, while the littleBits modules are smaller and more brightly colored. Both are gender neutral and include pieces that can light up and

69
make sounds. These features, which add to the attractiveness of the toys, elicited many smiles and laughs from participants.

5. *Time and space multiplexing:* both littleBits and KIBO employ a space-multiplex input style, but neither take full advantage of its capabilities. Future work should study the differences in learning indicators for toys with space-multiplexed input and toys with time-multiplexed input, as well as toys consisting of devices whose size and shape reflects their functionality.

Both littleBits and KIBO have input devices that can be classified as space-multiplexed input. Fitzmaurice and Buxton describe space-multiplexed input and time-multiplexed input in their evaluation of “Graspable User Interfaces” [15]. With space-multiplexed input, each function is manipulated by a single control that occupies its own space. With time-multiplexed input, one device can control different functions at different points in time. littleBits utilizes space-multiplexed input because each input module occupies its own space and performs one specific function (i.e. the pulse module causes blinking). Similarly, each KIBO block corresponds to one command (i.e. the forward block makes the KIBO robot move forward). The space-multiplex input style affords the capability of using different shapes and sizes in order to increase functionality and decrease complexity; however, neither littleBits nor KIBO capitalize on this. Both toys use different colors to represent different types of modules or commands (i.e. pink for input with littleBits, and blue for motion with KIBO), but color is rather arbitrary to the user, and perhaps the shapes and sizes could be altered to more closely reflect each object’s functionality. It might interesting to explore a version of littleBits and KIBO that uses time-multiplexed input instead of space-multiplexed input. The toys would then require less pieces, which could potentially decrease their cost. Future work should explore the differences in learning indicators for toys with space-multiplexed input and toys with time-multiplexed input.

6. *Relevance to real world:* users were more engaged in the long-term when playing with toys that naturally supported connections to the real world.
littleBits exhibited a higher relevance to the real world, allowing users to either build creations that closely resembled objects and items in everyday life or projects that could be used with a real purpose (i.e. a timer). The projects that users created using littleBits demonstrated varying levels of complexity. KIBO, on the other hand, was less effective in engaging users in real-world concepts and ideas. Children experienced more disengagement while using KIBO, much of it occurring towards the end of the KIBO activities when all blocks had been introduced. Many of the participants became less engaged after they had discovered a few interesting things that they could make KIBO do, such as beeping forever. KIBO would perhaps be more engaging if it were taught using methods that allowed children to make more connections between the KIBO programs and the real world. Its physical design, however, does not naturally support this, as many children referred to the KIBO robot as a “car” but no children actually compared the behavior of KIBO with the behavior of a car. This, in addition to the lower rate of utterances categorized as application, demonstrate that few participants were able to make connections between KIBO and the real world. Rogers and Portsmore also note the importance of real-world relevance [36]; their observations indicate that girls, in particular, are more engaged when the task at hand has a more meaningful purpose.

**LIMITATIONS**

This study has several limitations that require future work. The study did not assess individual learning over time or participants’ ability to apply their learning. Additional studies of longitudinal use are necessary to measure long-term learning and to account for the novelty factor. We measured learning based on dimensions of engagement, complexity, and complexity. While these are all facilitators for learning, they may not be an accurate measurement of learning, especially in relation to the development of computational thinking skills. Future studies should investigate the potential of littleBits and similar toys to directly impact the obtainment of learning goals.

In addition, most participants were from the local Wellesley area, which reflects relatively higher socioeconomic backgrounds and more educated households. Thus, this sample may not be an accurate representation of population as a whole. It would be important to
continue the study on a larger and more diverse sample. Future studies should also consider investigating the potential of other computational thinking toys that employ different styles of interaction. Both KIBO and littleBits are strictly tangible toys, and it would be interesting to study the effects of adding a visual component.
CURRICULUM DEVELOPMENT

Overview

In order to further explore the intersection of technology and education and how computational thinking can be taught in various settings, I pursued two projects during the Fall 2014 semester - final projects for the courses EDUC/CS 322 (Digital Technologies in Learning Communities) and EDUC 215 (Understanding and Improving Schools). The CS322 “Ten Weeks of Tech” project targeted younger children and their families, while the EDUC 215 “Got STEAM?” project was designed for elementary school teachers.

Ten Weeks of Tech

“Ten Weeks of Tech” was inspired by the idea that play is a fundamental aspect of life. Play contributes to neurological growth and development, play is described as “one important way that children build complex, responsive, socially adept and cognitively flexible brains” [18]. However, these benefits can only come from play in the three-dimensional world, which is becoming increasingly less common in our technology-driven society [18]. In addition, research has demonstrated the importance of technological fluency in today’s world; while digital technologies such as Scratch and Hour of Code promote computational thinking for young children, tangible technologies offer additional benefits, such as supporting trial-and-error activity and fostering collaboration between users [51]. The “Ten Weeks of Tech” program builds off of these ideas, utilizing littleBits to help families teach computational thinking to young learners. Developed using a design-thinking process, the 81-page program combines several components, including mini-stories about a fictional character, portfolio pages for reflection and preservation of projects, and easy-to-read instructions and hints. The current prototype, which is available at http://cs.wellesley.edu/~vlin/TenWeeksOfTech.pdf, targets users in grades 2-3 and their parents. Figures 41 through 43 show sample pages from the “Ten Weeks of Tech” program.
Figure 41: Top left: cover page of the Ten Weeks of Tech handbook; top right: introduction page to guide learners and their families; bottom left: sample pages from week 1 to introduce users to the different littleBits modules.
Figure 42: Top left: reflection sheet from week 1 to allow learners to draw and write about their experiences; top right & bottom: pages from week 2 that includes an introduction and materials needed, a fictional story, and a planning page.
Figure 43: Top left: instruction page for week 2, including written instructions and a circuit; top right: success page for week 2 to accompany the fictional story; bottom: two reflection pages that allow learners to preserve their creations and reflect about their experience.
Got STEAM?

Unlike the “Ten Weeks of Tech” program, which focuses on engaging children in computational thinking during play, the “Got STEAM?” project promotes the development of computational thinking for young children in formal learning environments. Education researchers suggest that the types of teaching and learning used to teach computational thinking, such as robotics and other problem-solving activities that encourage creativity and exploration, can improve education as a whole [7, 8, 23, 36, 40]. In recent years, there has been a growing effort to teach STEM (science, technology, engineering, and math) concepts in schools; however, researchers note that while curriculum materials are abundant, teachers lack the preparation, support, and knowledge to comfortably execute the lessons [21, 35]. In addition, experts call for the addition of art to the STEM discussion in order to advance innovation in our society [26].
The “Got STEAM?” program, delivered in the form of a teacher’s guide, includes over 25 pages of tips, activities, guidelines, and planning sheets to help elementary school teachers feel better prepared to teach STEAM (science, technology, engineering, arts, and math) concepts in the classroom. It provides teachers with a framework that significantly lowers the threshold of knowledge they must acquire in order to successfully execute the activities, and utilizes low-cost materials to account for the budget restrictions that most teachers face. The current prototype, which is available at http://cs.wellesley.edu/~vlin/GotSteam.pdf, is most appropriate for teachers of grades 2+. Figures 44 and 45 show sample pages from the “Got STEAM?” program.
Figure 44: Top left: cover page for the Got STEAM? handbook; top right & bottom: pages to detail the structure of each STEAM challenge activity – each activity should include a team building exercise, an intro and planning phase, and two attempts with sharing and presentations.
STEAM CHALLENGES
The following activities are provided for you to use – each page includes a title, intro, materials required, and extra section. The EXTRA gives an option for making the challenge a little more complicated. Feel free to vary the activities to fit your own comfort level and classroom – just remember to let the students plan, build, and share as a team. You’re encouraged to tie this into content standards as much as you can – for example, when they’re done building, have each team count and tally the number of each type of material used. Or have them write sentences or stories about what they built or the lab in general, asking them to use vocabulary words of the week. Have fun – they’ll come up with amazing things!

1.

how high can you go? – building towers
intro:
As a class, we must figure out how to build the highest tower. Your team can use only the materials in front of you to build the tallest, standing tower that you can. You may cut, tape, and bend things, but you won’t get extras if you change your mind later. Good luck!

materials required:
- marshmallows & toothpicks
OR
- white printer paper and construction paper, tape & scissors

EXTRA (to assign costs to each material - give a profit amount per height, and see who has the most profit (tallest & cheapest tower)

4.

beds and beyond – design your own bedroom
intro:
As a class, we must figure out how to design the best bedroom possible. It can have anything that you want and can be any shape or size, but no more than 5 things in the bedroom. You may use any materials that you can think of, including what’s provided. Good luck!

materials required:
- craft materials - paper, popsicle sticks, tape, scissors, felt, string, feathers, pom-poms, tissue paper, etc.

EXTRA (to give an additional constraint such as placing a cost on the materials and limiting their “spending”

TIPS & TRICKS
Here are some tips and tricks to make it easier!

HANDLING FRUSTRATION
Some students are likely to feel frustrated at one point or another. You won’t need to attend to the whole class while they’re building, so it might be easier to speak to upset students one-on-one. But it’s also a good idea to have a “frustrated chair” where students can move to if they are feeling upset (I’ve seen it done with chairs that go up and down, and it does wonders for calming students down!). Give them a few minutes, and ask them to rejoin their team when they’re ready.

HANDLING QUESTIONS
If you can help it, try not to answer students’ questions directly. Ask them to consult their teams, and then ask them to just try it to see if it works! There aren’t any right or wrong answers, so their answers are just as good as yours, as long as they can explain why.

MANAGING CHAOS
The classroom will be chaotic during these sessions, but that’s not a bad thing. As long as everyone seems to be working towards some sort of solution, you’re in good shape. If you need a way to keep the noise down, there are mobile apps (http://noisypop.com/) that can help you monitor the noise.
CONCLUSION AND FUTURE WORK

This research presents a study of Tangible User Interfaces to support the learning of computational thinking skills for young children in various settings. It investigates how different settings and technology toys impact engagement, complexity, and collaboration for children in early elementary school.

User studies were conducted with forty children in three different informal and formal educational settings: 1) lab settings; 2) workshop settings; and 3) classroom settings. Our findings suggest that both littleBits and KIBO provide children in lower elementary school with opportunities for reflection, problem solving, and application of computational thinking concepts. In addition, the results demonstrate that although learning is expressed in each setting, the child’s environment impact learning while using computational thinking toys. Moreover, a comparison of littleBits and KIBO in the lab setting indicates the affordances and challenges of each toy. These findings provide implications for educators and toy designers, which contributes to better supporting the development of computational thinking skills for children in early elementary school.

The addition of teacher interviews and two curriculum development projects allowed us to further explore technology education in K-12 classrooms and in family homes. These aspects of this research helped us develop empathy for the educators and parents that we seek to help, and deepened our understanding for teaching computational thinking to young children.

Future work for this research includes additional studies with a more diverse sample. Funded by the Laura W. Bush Traveling Fellowship and the Wellesley Serves! Grant, I will be traveling to South Africa for ten weeks in June 2015 - August 2015 to work with ORT SA CAPE, an educational NGO in Cape Town. ORT SA CAPE seeks to advance education in South Africa by offering after-school programs for impoverished children to explore reading and robotics, providing teacher training with a focus on STEM fields and early childhood development, and conducting research to evaluate education programs and projects. My project, as an extension of this research, includes hands-on work with the students and teachers at workshops and training programs, as well as research-based work to help identify areas of improvement for ORT SA CAPE. I will also be responsible for starting a littleBits Global
Chapter in Cape Town, and am eager to apply my research and expertise in ways that will positively impact even more children.

This research was conducted in the hope that future educators, computer scientists, toy designers, and parents will further contribute to the discussion concerning computational thinking and the development of computational thinking skills for young children. Future generations of youth stand to benefit from these efforts, as computational thinking becomes increasingly relevant in our society.
APPENDIX A: TECHNOLOGY TOYS

A.1 littleBits

The following table describes all littleBits modules used in this research (see [http://littlebits.cc/shop?filter=Bits](http://littlebits.cc/shop?filter=Bits) for full library of modules).

<table>
<thead>
<tr>
<th>module</th>
<th>function</th>
<th>description</th>
<th>image</th>
</tr>
</thead>
<tbody>
<tr>
<td>power</td>
<td>power</td>
<td>uses a 9 volt battery to supply electricity - must be connected to the circuit</td>
<td><img src="image" alt="Power Module" /></td>
</tr>
<tr>
<td>button</td>
<td>input</td>
<td>a round button - push it to turn the following modules on, and release it to turn them off</td>
<td><img src="image" alt="Button Module" /></td>
</tr>
<tr>
<td>dimmer</td>
<td>input</td>
<td>operates like a knob - turn it clockwise to send more signal to the following Bits, and counter clockwise for less</td>
<td><img src="image" alt="Dimmer Module" /></td>
</tr>
<tr>
<td>sound trigger</td>
<td>input</td>
<td>listens to the noise level - sends an ON signal if the noise gets over a threshold, which can be adjusted with the screwdriver</td>
<td><img src="image" alt="Sound Trigger Module" /></td>
</tr>
<tr>
<td>pressure sensor</td>
<td>input</td>
<td>a touch-activated module - the more pressure that is applied, the more signal it sends to the following Bits</td>
<td><img src="image" alt="Pressure Sensor Module" /></td>
</tr>
<tr>
<td>pulse</td>
<td>input</td>
<td>like an electronic heartbeat - sends out a stream of ON signals to make the following modules “blink;” speed of the pulse can be adjusted by using the screwdriver</td>
<td><img src="image" alt="Pulse Module" /></td>
</tr>
<tr>
<td>Device</td>
<td>Type</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Slide dimmer</td>
<td>input</td>
<td>like a light dimmer, the lever can be moved from one end to the other to adjust the intensity of following Bits</td>
<td></td>
</tr>
<tr>
<td>Light wire</td>
<td>output</td>
<td>four feet of the wire glows a soft blue color when activated; used for wearable products</td>
<td></td>
</tr>
<tr>
<td>Servo</td>
<td>output</td>
<td>a controllable motor that moves back and forth - has 2 modes, one where the input determines the position of the arm (“turn” mode) and one where the input controls the speed of the servo (“swing” mode)</td>
<td></td>
</tr>
<tr>
<td>Bargraph</td>
<td>output</td>
<td>contains 5 LEDs in different colors that light up to reflect the amount of signal received by the module</td>
<td></td>
</tr>
<tr>
<td>LED</td>
<td>output</td>
<td>a simple LED that lights up with a green color</td>
<td></td>
</tr>
<tr>
<td>RGB LED</td>
<td>output</td>
<td>an LED whose color can be adjusted - use the screwdriver to change how much red, green, or blue is shown</td>
<td></td>
</tr>
<tr>
<td>Long LED</td>
<td>output</td>
<td>contains a white LED at the end of the cable</td>
<td></td>
</tr>
<tr>
<td>Vibration Motor</td>
<td>output</td>
<td>a module that vibrates and buzzes when activated</td>
<td></td>
</tr>
<tr>
<td>block</td>
<td>function</td>
<td>description</td>
<td>image</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>bright led</td>
<td>output</td>
<td>an LED with a lot of bright white light when activated</td>
<td></td>
</tr>
<tr>
<td>dc motor</td>
<td>output</td>
<td>rotates the shaft when activated; contains a switch to set the direction of rotation</td>
<td></td>
</tr>
<tr>
<td>wire</td>
<td>wire</td>
<td>like an extension cord, allows for the physical separation of modules</td>
<td></td>
</tr>
<tr>
<td>fork</td>
<td>wire</td>
<td>allows for the connection of one Bit to three different Bits (in close proximity); used to perform several actions in parallel</td>
<td></td>
</tr>
<tr>
<td>branch</td>
<td>wire</td>
<td>allows for the connection of one Bit to three other Bits (in different directions); used to perform several actions in parallel</td>
<td></td>
</tr>
</tbody>
</table>

### A.2 KIBO


<table>
<thead>
<tr>
<th>block</th>
<th>function</th>
<th>description</th>
<th>image</th>
</tr>
</thead>
<tbody>
<tr>
<td>begin</td>
<td>basic</td>
<td>denotes the start of a program; must be used at the beginning</td>
<td></td>
</tr>
<tr>
<td></td>
<td>basic</td>
<td>motion</td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>---------</td>
<td>-----------------------------</td>
<td>---</td>
</tr>
<tr>
<td>end</td>
<td>denotes the end of a program; must be used at the end</td>
<td></td>
<td></td>
</tr>
<tr>
<td>spin</td>
<td>spin makes the KIBO robot turn in a circle</td>
<td></td>
<td></td>
</tr>
<tr>
<td>forward, backward, shake, turn right, turn left</td>
<td>motion makes the KIBO robot move forwards, backwards, shake and wiggle, turn right, and turn left</td>
<td></td>
<td></td>
</tr>
<tr>
<td>white light on</td>
<td>light turns on the KIBO robot’s white light</td>
<td></td>
<td></td>
</tr>
<tr>
<td>beep</td>
<td>sound makes the KIBO robot beep once</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sing</td>
<td>sound makes the KIBO robot play a pre-loaded tune</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Logic Block</td>
<td>Description</td>
<td>Image</td>
<td></td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>repeat &amp; end repeat</td>
<td><strong>logic</strong> repeats the blocks between the “repeat” and the “end repeat” blocks; can be used with the parameters: # of times, forever, until near, until far, until light, until dark</td>
<td><img src="image1" alt="Image" /></td>
<td></td>
</tr>
<tr>
<td>if &amp; end if</td>
<td><strong>logic</strong> only uses the blocks between the “if” and “end if” blocks if the given condition (near, far, light, dark) is true</td>
<td><img src="image2" alt="Image" /></td>
<td></td>
</tr>
<tr>
<td>wait for clap</td>
<td><strong>logic</strong> uses the sound sensor and tells the KIBO robot to wait until it hears a noise</td>
<td><img src="image3" alt="Image" /></td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX B: USER STUDY MATERIALS

B.1 Introduction & Summary

The following introduction and summary was sent to parents to recruit their children as participants for the lab study sessions.

Hi ________,

I hope this email finds you well.

I am currently pursuing an Honors Thesis in Computer Science with Professor Orit Shaer about innovative toys and computational thinking, and we would like to invite your child to participate in the research study. We are interested in investigating the use of innovative technologies to promote computational thinking skills during early childhood. Computational thinking involves defining, understanding, and solving problems, and while existing research has demonstrated the practicality of computational thinking, experts are still exploring how to integrate it into K-12 curriculum. We are focused on two specific toys, KIBO and littleBits.

We are looking for children in grades K-3 to take part in the study, which includes:

- An introduction to the technology toy
- Free play with the technology toy
- A culminating project with the technology toy
- An interview to assess enjoyment and understanding

If you are interested in having your child participate in the research study, please take a few minutes to complete this pre-questionnaire. We will contact you soon with more details.

Please do not hesitate to call or email us if you have any questions or concerns.

Taking part in research is voluntary, and you may choose not to take part. If you decide not to take part in this study, your decision will have no effect on any relationship you may or may not have with Wellesley College.

Thank you for your time.

Best,

Veronica Lin
Wellesley College | Class of 2015

Orit Shaer
Wellesley College | Assistant Professor
B.2 Pre-Task Questionnaire

For the lab study sessions, parents of child participants were asked to provide basic demographic information and information concerning their child’s technology usage by completing the following pre-task questionnaire.

Toys & Computational Thinking

For my thesis, I’m studying a variety of technology-linked toys and their abilities to promote computational thinking skills in younger children (ages 5 - 10). I am hoping to conduct user studies with child participants to measure performance gain, engagement, vocabulary, creativity, and complexity. To see the official research consent form (must be signed in person), please go here: 
http://cs.wellesley.edu/~vlin/Toys-ResearchConsentForms.pdf

If you would like your child to be considered as a participant, please complete this survey to the best of your ability. If you have any comments, concerns, or questions, please contact me by email at vlin@wellesley.edu. Thanks in advance!

* Required

What is your name? *

What is your email? *

What is your child’s name? *
First and last, please!

How old is your child? *

- 5 years old
- 6 years old
- 7 years old
- 8 years old
- Other: 

What grade is your child currently in? *
- Kindergarten
- First Grade
- Second Grade
- Third Grade

Has your child played with littleBits or KIBO before? *
- littleBits only
- KIBO only
- Both littleBits and KIBO
- Neither littleBits nor KIBO
- Not sure
- Other: __________________________

What type of technology does your child actively interact with on a day-to-day basis? *
Please check all that apply.
- Smart phones (iPhone, Android, Windows Phone)
- Tablets (iPad, MS Surface, Nexus, etc.)
- Laptops (MacBook, Windows PC, etc.)
- Desktop computers
- E-readers (Nook, Kindle, etc.)
- Video games
- None
- Other: __________________________

How much time does your child spend interacting with technology per day? *
Please estimate to the best of your ability. (For my purposes, technology does NOT include TV)
- Less than half an hour
- Between half an hour and one hour
- Between one and two hours
- Between two and three hours
- More than three hours
B.3 Post-Task Interview

The post-task interview was conducted during the last hour of lab study sessions. Participants were interviewed individually, and each interview lasted approximately 5 minutes.

1. Children were shown a picture of the “Smileyometer,” a children’s version of the Likert scale – see Figure 46 [33] and asked to point to a face for three different questions:
a. How did you feel during your entire time here?
b. How did you feel while playing with KIBO, the robot with the wooden blocks?
c. How did you feel while playing with littleBits, the colorful pieces that could connect to each other?

Figure 46: Smileyometer used to assess children’s enjoyment

2. What was your favorite part about this whole morning?
3. What was your least favorite part about this whole morning? What did you not like?
4. What did you learn today?
5. What did you build with the littleBits? What pieces and crafts materials did you use?

B.4 Consent and Assent Forms

The following consent and assent forms were presented to participants in all settings prior to their study sessions. Parents were required to read and sign the consent forms, and children ages 7 and older were required to read and sign the assent forms.

Parent/Guardian Research Consent Form

Introduction:
For my senior honors thesis, I am researching the use of innovative technologies to promote computational thinking skills during early childhood. These technologies include tangible toys such as littleBits and KIBO, which are specifically designed for young children in order to connect creativity and STEM concepts.

Participants:
Your child will be one of many children in the study; however, all research participants will be closely monitored to ensure cooperation and respect for each other. Participants will be working in groups of 2 or 3 with similar aged children.
Timing:
This study will occur on weekends or holidays, days on which children do not have school. The duration of the study will vary from 1 to 3 hours long, allowing time for snacks, bathroom trips, and appropriate programming including relaxing breaks between activities.

Location:
This study will take place in the Wellesley College Human Computer Interaction Lab, located in E125 of the Wellesley College Science Center. It is an inviting environment, as it contains colorful furniture and innovative technologies such as multi-touch surface tables and gesture recognition devices. All technologies are safe for children. There is also a nearby bathroom and water fountain for participants to use. Your child will be supervised by researchers during all times of the study.

Parental Instructions:
The study will be most effective if your child can focus on the activities; for this reason, you will be asked to drop your child off and then return at a set time.

Study Procedures:
As the primary researcher for this study, I will observe children learning about and playing with different technology toys. The lesson will include an introduction to the toy, some free play, and a culminating project. Your child may be asked to talk to a researcher about his/her experience with the technology toys. The researcher will ask your child about the project he/she is working on, how much he/she enjoyed the activity, and what his/her most and least favorite parts were. Some children may be interviewed by a researcher and asked to participate in short tasks and games. Pending your consent, your child may be photographed, videotaped, or audio-taped as part of observations. All children will be helped by researchers during this study.

Risks:
Participating in this study involves minimal risks.

Benefits:
Your child may benefit from this experience by engaging in learning and self-reflection about his/her learning experience. His or her participation in this study may help us improve curriculum and technology used in early elementary school in the future.
Voluntary Participation:
You or your child may choose not to participate in the study about technology toys. There will be no penalty if you decide not to allow your child to participate in the study; it will not affect any relationship you may have with Wellesley College.

Payments & Compensations:
You will not be paid or compensated for participating in this research, and there is no cost for participating in this educational technology session.

Privacy & Confidentiality:
The research team (consisting of Veronica Lin, student researcher, and Orit Shaer, thesis advisor) will keep all of the information collected about you and your child strictly confidential, as required by law.

Completion and Withdrawal:
You have the right to remove your child from the research study at any time without negative consequences. You also have the right to request that any or all of your child’s information be withdrawn. To remove your child, please call or e-mail Veronica Lin at Wellesley College.

Rights and Welfare:
If you have any questions about your rights in this research, concerns, suggestions, or complaints that are not being addressed by the researcher or research-related harm, please contact: Nancy L Marshall, Chair, Wellesley College IRB, at 781.283.2551 or nmarshall@wellesley.edu [106 Central Street, Wellesley, MA 02481].

Contact Information:
If you have questions or concerns about this research, please contact Veronica Lin or Orit Shaer by phone or email listed below.

Thesis Advisor:  
Orit Shaer  
oshauer@wellesley.edu

Student Researcher:  
Veronica Lin  
vlin@wellesley.edu
Statement of Consent [PARENT]

When you sign this document, you are agreeing to have your child take part in this research study. You will see 2 places where signatures are requested. The first is for you to sign on behalf of your child, and the second is for your child to sign only if he/she is 7 years or older. If you have any questions or there is something you do not understand, please do not hesitate to ask. You will receive a copy of this consent document.

Child’s Name ____________________________________________________________

Child’s Date of Birth ______________________________________________________

I agree to allow my child to participate in this research study.
☐ Yes ☐ No

I agree to allow my child to be photographed, videotaped, and/or audiotaped for data collection purposes.
☐ Yes ☐ No

I agree to allow photographs, videotapes, and/or audiotapes of my child to be used for educational and research purposes (e.g. poster talks, conference presentations, thesis report, etc.). No names or identifying information will be included with the child’s image or audio.
☐ Yes ☐ No

______________________________________________ _____________________
Name of Parent or Legal Guardian (Please Print) Date

________________________________________________________
Signature of Parent or Legal Guardian
Statement of Consent [STUDENT]

I would like to participate in a study on technology toys and how they help me learn. The toys are lots of small pieces that I can put together to build and create different things. During this study, I will play with the toys and then build a project. I will then show what I learned by doing some puzzles and explaining what I did.

I understand that this is a safe experiment and that I can stop anytime I want to. I can also take a break if I want to.

Everything I do during this study will be kept private and no one will know what I did in the study except for the researchers. They will not tell anyone else what I did or did not do, including other children. I may be audiotaped and/or videotaped in order to allow researchers to take notes on my responses.

I can ask questions about the procedure now or after the study. If I decide to stop the experiment, I will just tell the person in charge that I want to stop.

______________________________  __________________
My first name                                         Today’s date

______________________________
My signature
B.5 Study Protocols

User studies in each setting differed in many characteristics, including physical space, time, and participants. We adjusted the protocols for each setting to accommodate the various constraints – these protocols are described below.

**B.5A Lab Setting**

The following table describes the full protocol for the lab user study sessions.

<table>
<thead>
<tr>
<th>Activity Name</th>
<th>Activity Details</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introductions &amp; Icebreaker Activity</td>
<td>Introduced myself and the plan for the session. Asked each participant to say their name, their grade, and their favorite ice cream flavor.</td>
<td>5 min</td>
</tr>
<tr>
<td>KIBO Part 1: Intro &amp; Basic Blocks</td>
<td>Introduced how KIBO works. Started off by showing the KIBO robot and the red scanner, and then the barcodes on the basic blocks. Explained begin and end blocks, and demonstrated using a simple 4-block program to make the KIBO begin, shake, forward, and end.</td>
<td>3 min</td>
</tr>
<tr>
<td>KIBO Part 2: Free Play</td>
<td>Participants were given time to build their own programs using the basic blocks that had been introduced: begin, end, forward, backward, shake, spin, turn right, turn left, beep, and sing.</td>
<td>5 min</td>
</tr>
<tr>
<td>KIBO Part 3: Repeat Blocks</td>
<td>Introduced the repeat blocks by asking participants what repeat means. Explained that KIBO only repeats the blocks that are between the repeat and end repeat blocks. Demonstrated using a simple program that repeated forwards and sing 3 times.</td>
<td>3 min</td>
</tr>
<tr>
<td>KIBO Part 4: Free Play</td>
<td>Participants were again given time to build their own programs, this time including the repeat blocks.</td>
<td>5 min</td>
</tr>
<tr>
<td>Break</td>
<td>During break time, participants were encouraged to use the bathroom, eat snacks, and draw.</td>
<td>5 min</td>
</tr>
<tr>
<td>KIBO Part 5: Sensors &amp; If Blocks</td>
<td>Introduced the sensors on the top of the KIBO robot as well as the if and end if blocks. Demonstrated using a program where spin was activated by the distance sensor.</td>
<td>3 min</td>
</tr>
<tr>
<td>Part</td>
<td>Activity</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>KIBO 6</td>
<td>Free Play</td>
<td>Participants used all blocks to create their own programs.</td>
</tr>
<tr>
<td>KIBO 7</td>
<td>Microwave Robot</td>
<td>As a group, the facilitator led the participants in building a program to resemble a microwave. Participants were asked how microwaves work, and the facilitator wrote their commands on the board. Participants then built and executed their “microwave robot” which turned its light on, spun 5 times, turned its light off, and beeped.</td>
</tr>
<tr>
<td>KIBO 8</td>
<td>Free Play</td>
<td>Participants were again given time to use all blocks to build their own programs.</td>
</tr>
<tr>
<td>Break</td>
<td></td>
<td>During break time, participants were encouraged to use the bathroom, eat snacks, and draw.</td>
</tr>
<tr>
<td>LittleBits 1</td>
<td>Intro &amp; Power and Output Modules</td>
<td>Introduced how littleBits work by building a simple circuit with 1 blue and 2 green modules. Provided participants with a variety of blue and green modules.</td>
</tr>
<tr>
<td>LittleBits 2</td>
<td>Free Play</td>
<td>Participants were given time to explore all of the green pieces. At the end of free play time, they were asked to explain what the green modules do.</td>
</tr>
<tr>
<td>LittleBits 3</td>
<td>Pink Modules</td>
<td>Introduced pink modules by introducing human senses and using light switches as an example. Demonstrated by building a circuit with 1 blue, 1 pink, and 1 green module. Provided participants with a variety of all three kinds of modules.</td>
</tr>
<tr>
<td>LittleBits 3</td>
<td>Free Play</td>
<td>Participants were again given time to explore what all of the pink modules do, and asked to explain them at the end.</td>
</tr>
<tr>
<td>Break</td>
<td></td>
<td>During break time, participants were encouraged to use the bathroom, eat snacks, and draw.</td>
</tr>
<tr>
<td>LittleBits 4</td>
<td>Introduce Project</td>
<td>Asked participants to build a project in groups of two or three. Provided “robotic animal” as an example, and gave them crafts materials (including scissors, tape, markers, construction paper, pom-poms, confetti strips, and popsicle sticks). The facilitator also asked them to share their projects at the end.</td>
</tr>
<tr>
<td>Activity Name</td>
<td>Activity Details</td>
<td>Duration</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Introductions &amp; Icebreaker Activity</td>
<td>Introduced all volunteers and the plan for the workshop. Asked each participant to say their name, their grade, and their favorite color. Went through presentation slides to introduce participants to Computer Science, robotics, and Robogals. Split the group into two based on age.</td>
<td>7 min</td>
</tr>
<tr>
<td>Scratch &amp; ScratchJr.</td>
<td>Participants worked with Scratch or ScratchJr. depending on their age. More volunteers were present in this room; the higher volunteer-to-participant ratio meant that the experiences within the activity were more varied and exploratory. Participants walked through the Scratch “Getting Started” tutorial to create a dancing cat, and then were</td>
<td>40 min</td>
</tr>
</tbody>
</table>

**B.5c WORKSHOP SETTING**

The following table describes the full protocol for the workshop user study sessions. Half of the participants used littleBits first, while the other half used Scratch/ScratchJr. first.
encouraged to create their own animations and projects once they finished.

<table>
<thead>
<tr>
<th>Break/Transition</th>
<th>Participants switched groups, and were encouraged to use the bathroom during the quick break.</th>
<th>3 min</th>
</tr>
</thead>
<tbody>
<tr>
<td>littleBits Part 1: Intro &amp; Power and Output Modules</td>
<td>Introduced how littleBits work by building a simple circuit with 1 blue and 2 green modules. Provided participants with a variety of blue and green modules.</td>
<td>1 min</td>
</tr>
<tr>
<td>littleBits Part 2: Free Play</td>
<td>Participants were given time to explore the green pieces. At the end of free play time, they were asked to explain what the green modules do.</td>
<td>4 min</td>
</tr>
<tr>
<td>littleBits Part 3: Pink Modules</td>
<td>Introduced pink modules by introducing human senses and using light switches as an example. Demonstrated by building a circuit with 1 blue, 1 pink, and 1 green module. Provided participants with a variety of all three kinds of modules.</td>
<td>1 min</td>
</tr>
<tr>
<td>littleBits Part 3: Free Play</td>
<td>Participants were again given time to explore what all of the pink modules do, and asked to explain them at the end.</td>
<td>4 min</td>
</tr>
<tr>
<td>littleBits Part 4: Introduce Project</td>
<td>Asked participants to build a project in groups of two or three. Provided “robotic animal” as an example, and gave them crafts materials (including scissors, tape, markers, construction paper, pom-poms, confetti strips, and popsicle sticks). The facilitator also asked them to share their projects at the end.</td>
<td>2 min</td>
</tr>
<tr>
<td>littleBits Part 5: Build Project</td>
<td>Participants were given time to build their littleBits projects. Help was given more freely by the college student volunteers.</td>
<td>23 min</td>
</tr>
<tr>
<td>littleBits Part 6: Present Project</td>
<td>Each group was asked to share their project. The facilitator asked them to share: 1) what they built, and 2) how they built it.</td>
<td>5 min</td>
</tr>
<tr>
<td>Reflections &amp; Closing</td>
<td>All participants were brought back together at the end, and each participant was asked to share their name and one thing they learned at the workshop. Participants were thanked for coming, and then were picked up by their parents.</td>
<td>5 min</td>
</tr>
<tr>
<td>TOTAL</td>
<td>~ 1.5 hrs</td>
<td></td>
</tr>
</tbody>
</table>
### B.5c Classroom Setting

The following table describes the full protocol for the classroom user study session.

<table>
<thead>
<tr>
<th>Activity Name</th>
<th>Activity Details</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>littleBits Part 1: Intro &amp; Power and Output Modules</td>
<td>Introduced how littleBits work by building a simple circuit with 1 blue and 2 green modules. Provided participants with a variety of blue and green modules.</td>
<td>3 min</td>
</tr>
<tr>
<td>littleBits Part 2: Free Play</td>
<td>Participants were given time to explore all of the green pieces. At the end of free play time, they were asked to explain what the green modules do.</td>
<td>5 min</td>
</tr>
<tr>
<td>littleBits Part 3: Pink Modules</td>
<td>Introduced pink modules by introducing human senses and using light switches as an example. Demonstrated by building a circuit with 1 blue, 1 pink, and 1 green module. Provided participants with a variety of all three kinds of modules.</td>
<td>3 min</td>
</tr>
<tr>
<td>littleBits Part 3: Free Play</td>
<td>Participants were again given time to explore what all of the pink modules do, and asked to explain them at the end.</td>
<td>5 min</td>
</tr>
<tr>
<td>littleBits Part 4: Introduce Project</td>
<td>Asked participants to build a project in groups of two or three. Provided “robotic animal” as an example, and gave them crafts materials (including scissors, tape, markers, construction paper, pom-poms, confetti strips, and popsicle sticks). The facilitator also asked them to share their projects at the end.</td>
<td>3 min</td>
</tr>
<tr>
<td>littleBits Part 5: Build Project</td>
<td>Participants were given time to build their littleBits projects. Help was given as needed, but questions were often answered with other questions.</td>
<td>25 min</td>
</tr>
<tr>
<td>littleBits Part 6: Present Project</td>
<td>As participants finished their projects, each group was asked to share their project. The facilitator asked them to share: 1) what they built, and 2) how they built it.</td>
<td>5 min</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>~ 50 min</td>
</tr>
</tbody>
</table>
B.6 Teacher Interviews

In the classroom setting, several teachers were also interviewed about robotics education. Each interview lasted approximately 10 minutes, and consisted of all of the following questions:

1. How many years of experience do you have with these robotics materials?
2. In your opinion, are there benefits of doing robotics in your classroom? If so, what are they?
3. Could you tell me about what kinds of robotics activities you conducted in your classroom either this year or last year?
4. What were your overall impressions of these robotics sessions?
5. How comfortable do you feel with the robotics materials and computers, on a scale of 1-10? (1 = extremely uncomfortable, 10 = extremely comfortable)
6. How prepared do you feel to carry out robotics activities, on a scale of 1-10? (1 = extremely unprepared, 10 = extremely prepared)
7. What are the biggest challenges you face in incorporating robotics?
8. What could help you feel more prepared and more comfortable?
REFERENCES


