‘Maker’ within Constraints: Exploratory Study of Young Learners using Arduino at a High School in India

Sowmya Somanath[1], Lora Oehlberg [1], Janette Hughes[2], Ehud Sharlin[1], Mario Costa Sousa[1]
[1]University of Calgary, [2]University of Ontario Institute of Technology
[1]Calgary, [2] Oshawa, Canada,
[1]{ssomanat,lora.oehlberg,ehud,smcosta}@ucalgary.ca, [2]{janette.hughes}@uoit.ca

ABSTRACT
Do-it-yourself (DIY) inspired activities have gained popularity as a means of creative expression and self-directed learning. However, DIY culture is difficult to implement in places with limited technology infrastructure and traditional learning cultures. Our goal is to understand how learners in such a setting react to DIY activities. We present observations from a physical computing workshop with 12 students (13-15 years old) conducted at a high school in India. We observed unique challenges for these students when tackling DIY activities: a high monetary and psychological cost to exploration, limited independent learning resources, difficulties with finding intellectual courage and assumed technical language proficiency. Our participants, however, overcome some of these challenges by adopting their own local strategies: resilience, nonverbal and verbal learning techniques, and creating documentation and fallback circuit versions. Based on our findings, we discuss a set of lessons learned about makerspaces in a context with socio-technical challenges.

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

Author Keywords
DIY; Maker Culture; India; Young Learners; Physical Computing; HCI4D

INTRODUCTION
Maker movement-inspired activities in schools (e.g. [26, 27, 37]), and extracurricular outreach (e.g. [32, 36]) has increased and broadened participation in do-it-yourself (DIY) activities among young learners. A general emphasis has been placed on the idea that every child can become an innovator [18, 23, 34]. However, not all children have the resources or support they need to innovate [4]. Contextual challenges [33, 41, 52] or situational traits of unique learner groups [31, 33, 55] hinder young learners from participating in the maker movement.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CHI 2017, May 06-11, 2017, Denver, CO, USA.
Copyright © 2017 ACM ISBN/978-1-4503-4655-9/17/05...$15.00.
DOI: http://dx.doi.org/10.1145/3025453.3025849

The overarching goal of our research is to explore what happens when there are systematic infrastructural and cultural limitations that inhibit maker culture from taking hold.

An impoverished school in India is a prime example of a context with both economic and rigid education culture constraints that challenge several assumptions of maker culture – for example, easy access to technology, abundant independent learning resources and the intellectual ability of a student to independently select and solve problems [18, 37]. Within these constraints, we explore how young learners in India respond to the innovation and self-directed learning fostered by DIY. Introducing maker culture into formal education in India (e.g. teacher preparation and curriculum design) is a multi-faceted challenge and beyond the scope of this study.

In this paper, we present an observational study of a physical DIY workshop conducted at the Kar school (a pseudonym), a high school in peri-urban Bengaluru, India. Twelve (6 girls, 6 boys) grade 8 students (13-15 years old) participated in our three-day workshop and used the Arduino to prototype beginner level project ideas. We adopted a similar study methodology to what has been previously explored in other DIY workshops (e.g. [11, 31]). We observed, engaged in the participant projects, and conducted informal interviews with the participants. The results of our observations and informal interviews from both during and after the workshop indicate that students at the Kar school face monetary and psychological cost to exploration, have limited independent learning resources, struggle to find the necessary intellectual courage to explore and have technical barriers to engage in DIY physical computing activities. However, students are resilient, adopt traditional learning techniques and make do with the means available to them to overcome some of the challenges.

The primary contributions are: (i) identification of challenges to practicing DIY activities caused by infrastructural and resource limitations and cultural constraints, specifically those prevalent in India; And (ii) observed students’ strategies to overcome some of the identified challenges. Based on our observations we discuss a set of lessons learned about makerspaces within technical, infrastructural and social constraints.

IDENTIFYING LIMITATIONS IN THE CONTEXT OF INDIA
The Kar school is a prime example of two forms of limitations prevalent in India, more broadly: 1) infrastructure and educational resource limitations and 2) cultural resistance to non-conforming DIY activities.

Infrastructure and Educational Resource Limitations

DIY activities often assume that technology infrastructure (e.g. computers) and educational resources (e.g. documentation, access to books, instructional videos etc.) are readily available. However, an impoverished school in India may not meet this implied requirement. Several articles highlight that rural schools, and sometimes public and private schools in urban cities, can have very poor or sometimes non-existent educational inputs, teaching material or facilities [15, 28, 42, 45]. For example, limited access to computers (particularly in rural schools) is not uncommon. Pawar et al. [45] pointed out that it is common for several students (sometimes up to ten) to share the same computer. In some schools, a single PC is used as a solo teaching aid; an entire class (30-40 students) crowds around the same computer, ultimately causing the students to lose interest and shift their attention to other things [45]. Similarly, students have limited access to educational resources and thus lack the necessary exposure, confidence and knowledge to participate in self-directed DIY activities. For example, Krishna Kumar's [30] article about "textbook culture" in schools in India notes that resources other than the textbook are not available in the majority of the schools, and where non-textbook resources are available they are seldom used. Teachers fear damaging such resources, and the poor chances of repair or replacement discourage the teacher from using them, in turn limiting students' opportunities to interact with them [30]. These infrastructure and learning resource limitations challenge many of the assumptions of traditional maker culture – that people have the resources they need to independently learn how to create things.

Cultural Resistance to Non-Conforming DIY Activities

Beyond infrastructure and educational resource constraints, India has a rigid education culture that includes non-negotiable curricula, a top-down learning approach, limited interaction with teachers and teacher-centric teaching models. This dominant educational culture actively discourages non-conformist behavior, including innovation and the freedom to explore subjects independently. For example, most schools in India follow a "textbook culture", wherein the textbook is the main source of knowledge for both the teacher and the students [30]. Per this pedagogical approach, the teacher must ensure that students can answer questions based on the textbook without consulting the text during examinations. This examination-driven structure and "textbook culture" encourages rote learning and gaining surface level knowledge, as opposed to deeper analytical or critical knowledge perspectives [15, 30]. Mitra et al. [40] identified the teaching method employed in the majority of schools in India as teacher-centric. A single teacher is in charge of the entire class, and students are not allowed to interact or consult with each other during class time. Students are required to only perform individual learning and complete individual assessments. Mukerjee [41] observed that such teacher-centric, rote-learning is less effective because often students' understanding is limited, distorted or all together wrong. This educational culture is in direct conflict with the student-driven, self-motivated, and discovery-based principles suggested by DIY.

Limitations at the Kar School

We found both infrastructure and resource constraints as well as traditional educational culture in effect at our study site. We conducted a preliminary observation of one of Kar School's several computer science laboratory sessions. We observed an entire 30 minutes' computer science laboratory session of 15 grade 7 students. This observation was conducted prior to the three-day workshop.

For the laboratory session, a computer science teacher instructed the class in Excel. Although the purpose of the laboratory session was to provide students with hands-on training, the predominant discourse was a one-sided teacher-driven theory lecture. The teacher instructed students to take notes and draw screenshots of Excel menu options, as read from a textbook by the teacher (a classic example of "textbook culture" [30]). During the lecture, students appeared disinterested and easily distracted. During the entire session, no students accessed any of the computers. Further conversations with the teacher revealed that because the lab computers were maintained and updated by an individual outside of the school facility, often the teachers were afraid to let students access computers for fear that the students might damage the computers or disturb the installed software. Within these observed resource limitations and the rigid educational culture constraints at the Kar school we wanted to explore how students react to DIY.

RELATED LITERATURE

Similar research exploring ideas of constructionist and project-based learning has been done with impoverished communities in the learning science and education fields [4, 8, 12, 41, 46, 50, 52]. Several researchers' have also explored 'making' with 'at-risk' youth [31, 44, 55, 57], whose participation in the maker movement is hindered by their situational traits. Below we briefly discuss a few of these research works.

Impoverished Communities

Mukherjee [41] explored hands on learning in a school in India and posited that a constructivist approach to learning could benefit students trained in âĂIJtextbook cultureâĂİ. Hands on activities such as BRiCS (build robots create science) was hypothesized as a means to improve practical knowledge and understanding of educational concepts for students who are used to rote-learning [41]. Sipitakiat et al. [51, 52] explored the use of GoGo board, a small autonomous computer with sensing and control abilities, in an economically challenged context (Brazil). From the ethnographic studies conducted using the GoGo board, Sipitakiat et al. [52] found that cost constraints and limited availability of hardware are potential challenges for promoting physical computing activities in impoverished communities of Brazil. The authors suggested locally manufacturing the microcontroller board to reduce costs, and to re-use found and existing materials such as broken electronics to encourage exploration of readily available technology (e.g. clocks and radios). Lin and Shaer conducted a case study to explore how technology toys can promote computational thinking for young children in Cape Town [33]. They explored the use of littleBits [6] with elementary school children from diverse socioeconomic backgrounds. They found the main challenges for South African children to be peer pressure, low self-esteem and unequal treatment from teachers. In contrast to privileged students, children of impoverished communities were observed to exhibit important differences: had a lower frequency of communication and primarily relied on non-verbal communication, affecting the social aspects of making; had less of a gender divide for DIY-based activities; and thanks to littleBits [6], had a new opportunity to develop fine motor skills and practice basic language skills (e.g, using prepositions to describe their circuit).

Unique Learner Groups

Situational traits of ‘at-risk’ youth hinder their participation in the maker movement; these students tend to quickly give-up, are unwilling to experiment and communicate, and demonstrate less engagement and lack of motivation. Kuznetsov et al. [31] introduced e-textile activities as therapy and for mentoring of ‘at-risk’ students. They found that the e-textile workshop sessions inspired their participants, who tended to be uninterested and uncooperative in educational activities, to complete interactive projects and engage with workshop volunteers as mentors and peers. Stager [57] introduced the concept of Constructionist Learning Laboratory (CLL) to engage youth in prison facilities. The design goal of CLL was to create an environment that mimics the principles of constructionism [44], wherein, youth engaged with a wide range of low and high-tech materials (e.g. LEGO, Arduino) to build physical artifacts. It was observed that CLL students engaged in learning-by-making and students who were thought to be incapable of learning proved quite capable and even enrolled in college courses while in the CLL. Somanath et al. [55] explored the use of a variety of computational platforms (e.g. Arduino, Makey-Makey and LilyPads) in DIY activities to engage ‘at-risk’ learners. They observed that in contrast to instruction-based starter kit activities, engaging their participants in open-ended design activities improved participants’ self-assessed experiences with circuitry and programming. Based on their observations, they discussed that creative tasks, access to a variety of tools and making the projects relevant to the students could improve students’ engagement in DIY activities.

In this paper we explore how students faced with *both* economic and cultural limitations react to DIY activities. Students in India are a unique learner group whose rigid education culture and resource constraints hinder their participation in the maker movement. The situational constraints for students at the Kar school actively discourage innovation and independent exploration. In light of our goal, India presents a unique opportunity to gain insights about ‘making’ within technical, infrastructural, and social challenges.

STUDY METHOD

Our research team hosted a three-day ‘make-a-thon’ style [54] physical computing workshop at the Kar School, a private high school in peri-urban Bengaluru, India. Each day, the workshop lasted three hours and students engaged in building simple projects using Arduino and other electronics (sensors, actuators and components). To understand what typical classroom interactions look like at the Kar school we also did a preliminary observation of a computer science laboratory session, as already discussed.

Figure 1. Workshop site: Kar school computer science laboratory.

The Kar school is a low-fee charging institution and hence, affordable to students from low social economic status (SES) backgrounds. We conducted the workshop during school hours (9:00 am - 4:00 pm) at the school’s only computer lab facility (Figure 1). There are two reasons why we could not conduct the study as a longer after school program. First, the computer lab facility and the school were shut down after school hours. Second, few students (especially girls) were willing to participate in the workshop after school hours due to security concerns, or lack of parental permission. Electricity availability at the school was also limited and unscheduled outages were a common occurrence during the workshop. Due to unscheduled outages, workshop times had to be flexible. The school’s faculty had no prior knowledge of Arduino, or about the maker movement.

Participants

A group of 12 students (6 girls and 6 boys) ages 13-15 years (grade 8) participated in our three-day workshop. The choice of grade 8 students was opportunistic. The school principal selected the 12 participants based on the following criteria. First, because students would miss a total of nine hours of regular class time, the principal wanted to select students who could cope with this break from in-class learning; hence, he chose students who performed academically well among the grade 8 students. Second, we wanted to ensure that the same students could attend all three days of the workshop. As a result, the principal chose students who were also regular attendees at school, ensuring that there was a high probability that all participants would attend the entire workshop series.

From our pre-questionnaire we gathered that all our participants belong to the low SES strata. The participants’ parents’ occupations can be classified as low-paying jobs that require minimal or no prior education (e.g. building painter, janitor, barber etc.).

All but one participant had experience with computers since grade 5; one participant began using a computer during grade 6. Two of the 12 participants owned a personal computer. The most common use of the computers (as specified on the questionnaire) in the school was for using programs such as MS Access and MS Excel (current curriculum). Beyond computers, students had previously interacted with mobile phones that belonged to other family members and two participants personally owned mobile phones. Participants used them for playing games, calling friends and watching videos. There was no explicit mention of using Internet on the phone. When asked about their prior exposure to programming, programming languages and programmable electronics, 11 out of 12 participants listed English as a programming language. Participants' perception of programming was also quite different from the traditional definition: our participants' equated computer programming to using installed programs on a computer. No participant reported any prior knowledge of electronics and/or programmable electronics.

Setup

To observe how young learners at the Kar school would engage with DIY physical computing activities, we provided the resources for this study: 6 Arduino microcontrollers (one Uno R3 and five Leonardo) and a range of electronic components, sensors and actuators (switches, push buttons, resistors, phototransistors, light sensitive resistors, piezo buzzers, servos, vibration sensors, mini speakers, carbon monoxide sensors, temperature sensors, LEDs and force sensitive resistors). The total cost of the electronics purchased was ∼$400. Most of the electronics used in the study are accessible within urban India, and can be purchased online and are shipped internationally. We also used Arduino's in our workshop because of their affordance to build a wide variety of projects [55] and rich documentation. During the workshop all the electronics were kept on a central table for participants to freely access. At the end of the workshop, the entire package and additional resources (a copy of the Arduino programming notebook [19] and SparkFun Inventor's Guide [56]) were donated to the school for future use by students.

A total of six computers were used during the study to create a 2:1 student-to-computer ratio. Four out of six computers belonged to the school; the researchers provided two additional laptops. Kar school had a total of ten computers, however, six of the computers were not working at the time of the study. Because multiple users cannot simultaneously build circuits using an Arduino, we used the 2:1 student-to-computer ratio. An increased ratio would more likely cause students to crowd around the single Arduino and a computer, resulting in one student becoming the dominant circuit builder and programmer. Other students would become passive onlookers and perhaps ultimately disengage from the activity (similar to single computer use scenario described by Pawar et al. [45]).

We installed the Arduino IDE (line programming) and Ardublocks [3] (graphical programming) on all computers. As the Kar school had no Internet connection, none of the computers used for the study had Internet access. This setback prevented students from accessing online learning resources. To mitigate the lack of Internet access, the researchers provided a word document with basic sample code for each of the electronic components on all the computers. Because the provided sample code could be directly copy pasted into the Arduino IDE as opposed to ArduBlocks, where the participant would have to assemble graphical programming blocks, participants ended up using the Arduino IDE both during and after the workshop.

Workshop

The goal of our three-day workshop was to position the participants as investigators with agency and in turn, to observe how the challenges of the context shaped their DIY experience. The study encouraged discovery-based collaborative learning [2] wherein, the participants worked in small groups of two or three and helped their peers to debug code and circuit connections. Throughout the workshop period the researcher adopted an inquiry-based learning approach [1], guiding students by posing questions or problems rather than presenting solutions without much invested effort. The researcher was also available for help; however, because only one researcher was managing the study, their help was also a scarce resource. Throughout the remaining paper we reference the researcher as R, participant as P and group as G.

Figure 2. Workshop: Participants working on their projects.

Workshop Day 1. On day one the researcher briefly introduced the participants to basic electronics: what is an electric circuit, what is a breadboard and how to build a circuit. The researcher drew a simple circuit diagram on the blackboard to aid the explanation. The researcher demonstrated a practical example of a circuit using an LED and a coin cell battery, showing how to turn on the LED by pressing the LED legs to the positive and negative side of the battery. To familiarize our participants with the Arduino, the researcher walked the participants through a step-by-step LED blinking exercise. Pairs of participants replicated and extended the exercise by connecting multiple LEDs. Participants used the Arduino sample code *Blink* to program the LED. The researcher demonstrated the use of both Arduino IDE and the Ardublocks IDE. After the LEDs were successfully blinking, the researcher gave participants time to continue exploring the circuit connections. After the circuit exploration phase, each group presented to their peers detailing their experimentation process. The researcher used the presentation sessions to probe students' understanding of hardware and software, and to learn how they thought the LEDs blinked. Following these discussions, it became easier for the researcher to clarify and formalize more technical concepts such as serial connections and functionality of a microcontroller. Example of a question asked by the researcher during the discussion is as follows:

R: *How did the LEDs turn on and off?*
P7 : *The circuit is continuously going. The current is passing from one bulb [LED] to another, so it is glowing and then shut down, then again it is starting.*

The researcher employed a similar inquiry-based approach to break down parts of the code. Participants were asked by the researcher to share their understanding of keywords such as *setup*, *loop* and *delay*. Based on participants' definitions, the researcher explained the corresponding functions in the code. To provoke the participants to think further, the researcher posed logical questions. For example:

R: *How do you think we can make the LED stop blinking?*
P7: *By removing delay?*

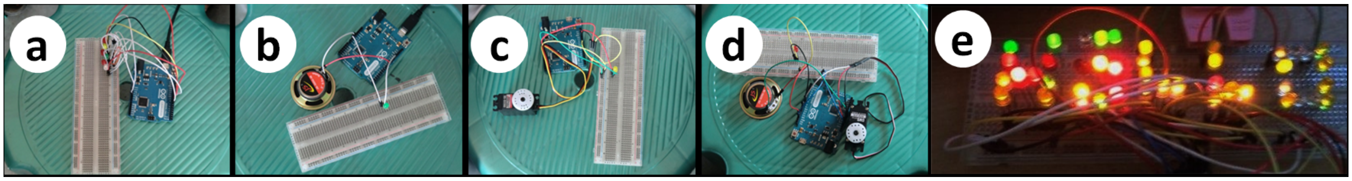

Figure 3. Projects demonstrated at the end of the workshop: (a) LED calculator, (b) Sound and Light, (c) Servo Controlled LED and (d) Servo + LED + Speaker; Project demonstrated at the science fair: (e) "Hello World".

At the end of the session the researcher briefly explained the functionality of the remaining components. In preparation for the 'make-a-thon', the researcher asked the participants to think of simple project ideas for day two. To inspire the participants, the researcher orally discussed examples of projects that participants could build.

Workshop Day 2. On the second day, each of the groups presented their project ideas. Presentations were semi-structured asking students to identify the project they wanted to build and briefly list the initial set of hardware they would need to use. The researcher asked questions during the presentations to better understand the group's goal for the chosen project. Participants spent the remaining workshop time building circuits and programming (Figure 2). The researcher encouraged the participants to collaborate and ask peers for help before approaching the researcher. The researcher would eventually help to move them along. The researcher asked the groups to make notes of their working process and presentations were conducted at the end of the session summarizing their tasks and challenges for the day. Two groups (G3 and G4) discontinued working on their projects during day two and joined other groups.

Workshop Day 3. On the final day, four groups completed their chosen projects. The groups did a final presentation of their projects (Figure 3a-d). The workshop closed with an open-ended discussion with the participants. The discussion covered the following topics: participation experience, participants' view of how they benefited from attending this workshop, what they found challenging, and their thoughts on future possibilities for building other physical interactive prototypes.

Beyond the workshop: Science Fair

Six weeks after the workshop we were informed by the school principal that the school was conducting a science fair. Two workshop participants from G1 demonstrated two electronics projects that they had built at the science fair. The first project demonstrated by G1 was a reconstructed version of their workshop project, a LED calculator (Figure 3a). The second project was a new and independent exploration by G1 (no researcher help) entitled *"Hello World"* (Figure 3e). The student built an array of blinking LEDs arranged to spell "Hello" as seen in Figure 3e. A researcher conducted an unstructured interview during the school science fair to gain insights into the design process and challenges faced by the group.

Data Collection and Analysis

Data sources for this study included the following:

1. **Pre-questionnaire** – asked personal demographic information and a few questions regarding prior technology and programming experience.
2. **Presentation videos** – at the end of each workshop day we video recorded all participant presentations as they summarized their work, detailing the tasks they had accomplished and how they resolved project related issues. In addition, on day two of the workshop we video recorded all the participants as they presented their project ideas.
3. **Individual group videos** – we used the two laptop web cameras to capture conversations and working processes of two groups.
4. **Written notes from students** – at regular intervals ($\sim$2 times per session) each group was asked to write notes about problems they were addressing and a list of any unresolved problems.
5. **Closing discussion video** – at the end of the workshop, we video recorded an informal discussion of the participants' experiences. All workshop participants were part of this discussion.
6. **Informal interview at science fair** - we audio recorded an informal interview with G1 at the school science fair to learn more about their progress after the workshop.

Majority of the collected data was in English. However, parts of the laptop videos were spoken in the local state language (Kannada) and were translated by the researcher (native speaker). Our qualitative analysis methodology is inspired by Walny et al.'s approach [61]. One author (analysis lead) did several passes through the transcribed video and interview data, and identified the sequence of activities performed by each group captured as workflow diagrams (Figure 4). Looking across all group's workflows, the analysis lead created a set of common activity labels. These labels were discussed and revised among three co-authors to arrive at the final set of labels:

A *identify project*: corresponds to the project proposed by the group;
B *identify material*: lists the components chosen by the group for their project;
C *identify behaviour*: describes the expected behaviour of the project as explained by the participants;
D *implement*: summarizes the main steps taken to implement the project;
E *project demonstration*: name of the final project demonstrated.

Using the activity labels, we traced each student groups workshop journey, from project identification to project demonstration. Guided by the workflow diagrams, we returned to our

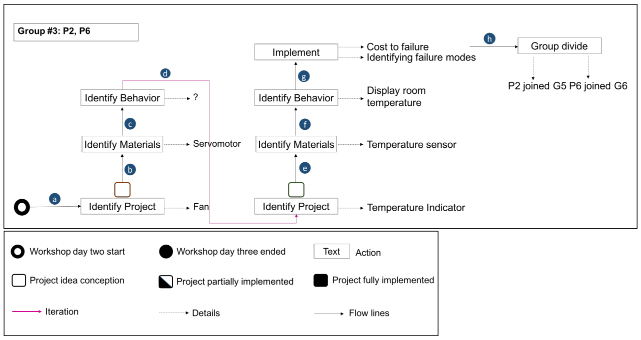

Figure 4. Example workflow diagram (Group 3)

transcribed video and interview data, and student notes data to identify challenges and student strategies.

CHALLENGES TO PRACTISING DIY-BASED ACTIVITIES

Within the known resource limitations and cultural constraints prevalent at the Kar school, we observed monetary, resource and situational challenges for students to engage in DIY physical computing activities.

C1: Monetary and Psychological Cost to Exploration

Exploration and tinkering is common to DIY activities [22, 37]. Failure is common, necessary, and fruitful. During exploration and tinkering, learners may fail to accomplish the desired results, may damage the hardware components or may decide to not use a purchased component. However, in a context like the Kar school, there is a high monetary and psychological cost associated with trial and error.

In our study we observed two instances of fear. G3 wanted to build a project using the Arduino temperature sensor. However, G3's attempts at incorporating the temperature sensor failed. The researcher upon debugging G3's circuit connection found that the temperature sensor was connected incorrectly and was damaged (the program constantly displayed values in the range of 200°C). After realizing they had broken the sensor, the participants could not be motivated to continue their project and did not want to work with another temperature sensor. To keep G3 involved in the workshop the researcher suggested that the members of G3 join other groups whose projects interested them.

This fear is even apparent when coding software, even though there is very low likelihood of causing irreversible damage through failure. For example, upon G2's circuit building success, the researcher advised G2 to explore the sample program code by modifying the program variables. During this process, the participants would only modify the variable as suggested by the researcher and were reluctant to modify the variables' values on their own. Possible reasons for this hesitation is that G2 may have been concerned that they could not go back to the original code, they did not understand the code or that they did not know what could be modified and how it could be modified. However, irrespective of the researcher encouragement and assurance that they could not "damage" code, participants of G2 did not engage in much free-form code exploration. Similar hesitation with modifying code was also observed in the other groups.

C2: Limited Independent Learning Resources

In a typical makerspace (a communal space for practising hands on learning), learners have access to several learning resources that help introduce young learners to new educational and computational technologies. These resources may include online learning resources, specialized after school programs, and qualified mentors (e.g. [9, 20, 37]). However, in a setting like the Kar school, the number of available learning resources is heavily restricted. There was no Internet access available at the Kar school, limiting students from accessing any online documentation. Students did not have any textbooks or guides for physical computing available as a reference. Additionally, teachers at the school were not aware of programmable electronics, and could not mentor or guide the students. The limited availability of independent learning resources challenges several assumptions of how a student can successfully participate in DIY-based activities – How does a novice learner become aware of the idea of DIY? How do novices learn about physical computing technologies such as Arduino, Makey-Makey and littleBits? How do they understand materials and behaviours? Below we illustrate instances of participants having difficulty with: (a) finding sources of inspiration; (b) understanding how to use and work with technology; and, (c) increasing project complexity due to limited independent learning resources.

Use of independent learning resources such as, online videos and pictures is a common way of inspiring ideas [24] and has been used in prior studies (e.g. [55]). However, students in our workshop had difficulty coming up with ideas because they did not have exposure to outside learning resources. Although proposing new ideas for prototyping is a challenge for any novice learner and has been previously observed (e.g. [31]), in our study site it was further emphasized due to limited learning resources. During the workshop 4 out of 6 groups (G1, G3, G5 and G6) proposed to build a fan using a servomotor. Originally the project was proposed by G1, followed by G3, G5 and G6 proposing the same idea. Not having access to resources that could inspire the participants (e.g. showcase of online examples), groups ended up proposing the same ideas. We observed this stagnant ideation again when G5 completed the fan project, but was unable to suggest a new project idea to explore. The researcher had to suggest a new project ("Servo Controlled LED") to allow them to continue exploring.

Understanding hardware and their corresponding behaviour requires some initial documentation input, for example, datasheets or books that explain simple components. Although participants were given a brief introduction to each hardware component being used in the study on day one of the workshop, they had no learning aid that they could use for support. Consulting the researcher was their only source for clarification and learning. Because of this, two groups were seen proposing wrong project ideas. For example, G2 first proposed to build a project titled "Voice Recording", listing FSR and mini speaker as the required electronic components for this project. The identified behavior was to use the speaker to record the FSR input. However, G2 had no understanding of the distinction between an input and an output device resulting in a flawed behavior identification. The participants were not aware that speaker was an output device and could not be used for recording purposes. Clarification had to be provided by the researcher. Similary, G3 attempted to use a temperature sensor, but wrongly interpreted that 200°C was the correct output for a temperature sensor. The researcher had to re-explain what a temperature sensor was and what the expected output value ranges would be.

Limited resources also make it difficult to increase project complexity or scaling up an idea. Participants had been exposed to connecting one LED to their circuit during day one of the workshop. Scaling from one LED connection to multiple LEDs (required for the LED calculator project) was challenging for all groups. For example, G1 initially connected all the LEDs to the same pin of the Arduino and had no control over individual LED states. Without a reference for concepts like serial and parallel circuit connections, participants were constrained to either solve by trial and error, or ask the researcher. Because our participants were novices, the researcher had to guide them to keep the group moving ahead with their project. Scaling up was also an observed issue for programming. Participants had template code to make one LED blink, however, to change the states of the individual LEDs they had to modify the code accordingly. This was found to be a challenging task as participants did not fully understand how to modify the line code (*"...we were having problems in the codes which we were not knowing"* [P7]).

C3: Finding intellectual courage

Unlike students familiar with an interactive collaborative learning context [7], students schooled in rigid educational settings like our study site focus on following the teacher instead of independently exploring an educational concept. Rigid educational paradigms like "textbook culture" expect students to memorize textbook content and follow exactly what the teacher proposes (teacher-centric teaching) [30, 39, 41]. At the Kar school we observed that students sometimes lacked the necessary intellectual courage to freely explore and learn.

An instance of students' unable to find their intellectual courage was observed on day one of the workshop. After the LED demonstration, participants were given time and were encouraged to explore the circuit. While a majority of the groups explored connecting multiple LEDs to their circuit, G3 was an exception. G3 did not modify their one LED circuit connection. When asked if they wanted to connect more LEDs to their circuit, participants of G3 said *"no"*.

While finding the necessary intellectual courage is a limitation among all groups, particularly G3 who refused to follow others in experimentation, G6 demonstrated unusual intellectual courage for the group. From the laptop videos of G6 we observed that P9 was rather experimental in his approach - *"wait I am doing something, even I don't know what I am doing"* [P9]. Unstructured exploration was characteristic of G6 throughout the workshop - G6 continued to add electronics to their circuit with no explicit goal. Even though they were faced with frustration in the process and the participants felt like they should have done a project similar to others (*"we should have also taken LED project"* [P9]), they strived to keep pushing ahead (*"Don't try to hurry, let's keep trying"* [P12]). This was interesting because although they had no set goal, when G6 found the intellectual courage to explore, they discovered several aspects of circuit building and programming in the process of unstructured exploration.

C4: Assumed Language Proficiency

To identify and solve programming errors, students need to understand the syntax of the programming language, and interpret error messages. However, these error messages assume that students have the necessary proficiency with technical English. In places like the Kar school, this assumed language proficiency is yet another barrier for students. This poses a fundamental challenge to practicing DIY-based activities as students who are dealing with unfamiliar English vocabulary will have even more difficulty comprehending the underlying technical concepts behind an error message.

An instance of assumed language proficiency was observed in the video data of G6. Upon uploading the modified template code to the Arduino board, the IDE notified the participants of "precautions" (a compiler notification). However, in order to troubleshoot and isolate the debugger messages the participants have to first understand them. G6 did not understand the meaning of the word precautions (*"let's ask her [the researcher] what is precaution"* [P12]), causing a fundamental block in their progress. Although English is the language of instruction at our study site, this observation shows how technology may prevent people from accessing it when they do not have enough English language proficiency, and much less proficiency in technical language or concepts.

FINDING SUCCESS: STUDENT STRATEGIES

In the previous section we identified some of the challenges that a context like our study site poses for practicing DIY-based activities. In this section we present strategies that students adopted to overcome some of the above challenges. Motivated by Smyth et al.'s [53] argument about needs assessment, we acknowledge that although the above identified challenges are potential roadblocks to practicing DIY inspired physical computing, the identified "needs" or "challenges" may not be as strongly felt by the 'maker' as perceived by an outsider. Below we discuss four strategies that our participants adopted as a way to overcome *some* of the challenges (C1-C4).

S1: Resilience

Resilience, the ability to creatively cope with challenges has been discussed in the context of economically challenged settings (e.g. [43, 52]) and makerspaces (e.g. [49, 58]). *Jugaad*, 'to make do', has been discussed in the context of India as an innovative and improvised solution to resource constraints [47, 48]. Within our study context, being resilient was yet again an emergent strategy. Participants creatively coped with both material unavailability (C1) as well as limited independent learning resources (C2) by being resilient.

During the workshop G2 wanted to use the force sensing resistor (FSR) and a speaker to build a project that would generate audio based on FSR input ("FSR Sound"). However, they could not conveniently include the FSR into their prototype because they did not have a soldering iron. Once the group realized they needed – but did not have – a soldering iron, G2 reconsidered their options and decided to alter the scope of their project. This time they chose to use an LED and proposed altering the LED state based on the audio tone (“Sound Light”). The unavailability of the tool forced the participants to rethink the possibilities of what could be explored and come up with new ideas.

G1 demonstrated resilience at several levels during their after workshop experience, while building the “hello world" project (Figure 3e). First, G1 did not have a resistor that they required for their project. Instead, to keep moving forward with the project goal, P1 used their common sense to use a metal wire as a substitute for a resistor. Although the metal wire may not be the exact solution to a missing resistor, the spirit to keep trying is essential to making. Second, G1 initially built a series circuit for the “Hello world” LED display, but soon realized they did not have enough wires and had to reconsider their circuit building strategy. To overcome this challenge, G1 referenced the textbook that was provided to the school as part of the after-workshop package and found a solution to reduce the number of wires required. Lastly, P1 had forgotten how to code his circuit; instead of being deterred, he involved his friend to get programming help.

S2: Nonverbal and Verbal Learning Techniques

Nonverbal communication is a social learning technique observed in young children [62]. Lin and Shaer [33] observed that children in their workshop in South Africa, also primarily used gestures to communicate with each other. In our study, we observed imitation as a form of nonverbal communication. Within the “textbook culture", rote-learning, and teacher-centric pedagogy practiced in schools in India [30, 39, 41], imitation is an implicit learning technique. Students are trained to memorize textbook content and reproduce the same during examinations. Also, because of the teacher-centric teaching style, students are trained to follow. Therefore, over a period of time, students become accustomed to imitate or copy. While much of “textbook culture” is counterproductive to DIY culture, students adopted this learned skill of imitation as a strategy to overcome the limited learning resources challenge (C2).

On day one of the workshop the researcher guided the participants using an introduction activity that involved building a simple LED circuit. Participants’ were given time (30 minutes) to explore the electrical circuit before the workshop was continued further. During this period, G1 took the lead and connected multiple LEDs to their circuit by trial and error. Following this, four other teams (with the exception of G3) imitated them and connected multiple LEDs. This was interesting because they were not following the instructor, instead, they were following their peers. Another instance of imitation was observed during day two and three of the workshop where participants were seen visiting work spaces of other workshop group. Seeing G2 use a mini speaker in their project, G6 was also inspired and decided to include a mini speaker in their circuit. However, it is important to note that while imitation is a successful strategy to overcome the limited resources challenge, imitation as a strategy can fail if there is a problem at the source level. For example, by imitating G1’s initial project proposal, four out of six groups (G1, G3, G5 and G6) proposed the “Fan” project. Because G1 was not sure about what they wanted to build for the “Fan” project, all other groups also drew a blank as they did not know what the expected behavior of the “Fan” project should be.

In addition to nonverbal social learning techniques, asking peers or friends for help is a common verbal learning technique employed by students [10]. Although students in India are mostly used to individual learning and are discouraged from consulting or interacting with their peers within a classroom setting [39], within our workshop setting, asking peers for help was an emergent and encouraged strategy. With help from peers and occasionally from the researcher, students managed to build projects and stay involved in the physical computing workshop, as well as after the workshop (where there was no researcher available for help). When P6 from G2 joined G6 (due to temperature sensor burn out), participants of G6 were encouraging of their new group member and got her up to speed by explaining their circuit connections (*“red line is power, black line is ground, and white line is analog. Analog is here, it’s analog 0”* [P12]). They were also ready to trust their new member and willing to delegate tasks to her (*“you connect the circuit”* [P9]). We also observed that the group members would often help each other with connecting components and would think together when they were unsure. One member of the group was found to be the dominant circuit builder, while others took on the role of observers and advisers (*“connect to the positive, what you are doing is wrong”* [P12]). Similarly, for programming, since G6 was connecting multiple electronic components to their circuit, merging and debugging code was a challenging task for them. They were observed to be collaboratively dissecting code piece-by-piece to aid understanding. Seeking help of other group members was seen to be a prominent activity of G6. They were observed to validate and correct their own circuit connections by checking those of other groups (*“wait let me think, G1 has done the same”* [P12]). Similarly programming also involved inter-group collaborative effort. Participants were observed to be: (a) clarifying code logic (*“To turn off, turn the code to LOW”* [P7]), (b) clarifying syntax (*“LOW has to be in capital letters”* [P4]) and (c) suggesting possible applications for each other’s projects - *“They can create a calculator (LED based) and the speaker can tell the result”* [P7].

S3: Documentation and Fallbacks

The utility of taking notes as a way to help learning is a common practice in schools in general [25]. In our preliminary observation, students took notes as the teacher dictated various aspects of computer software. In our workshop note taking was an emergent strategy students employed to overcome limited learning resources and to create their own documentations and reference points.

Based on our analysis of the collected written notes from all the groups and the post-workshop questionnaire responses, participants documented their current experience by jotting down personal pointers to assist them in their future DIY attempts. Two examples of personal notes are: (1) *“a) Red* *line is power; b) Blue line is ground; and c) Every circuit has to end in ground line"* [P3]. (2) *"While building the electrical circuit we have to keep the pin codes in mind. After connecting the wires, we have to keep the pin codes in the mind and type in the computer"* [P11].

Unlike the elaborate notes dictated by the teacher from the textbook, the student notes from the workshop are similar in nature to logbooks in engineering and science – i.e. an informal document used for personal record keeping, to serve as a reminder of work-in-progress, recording actions and other people's input [13, 38].

As part of development of understanding (i.e. a broad view of understanding as both a process and an outcome [22]), note taking practices were also extended to physical circuits. Participants were observed to have a fallback circuit version they could go back to when their circuit connections were not working. A common strategy that was observed across all the working groups when their circuit connections did not work was to trace back to a simple one LED circuit, rebuild it, test it, and if found to be working, then they would attempt to reconstruct their current circuit. Having a last working version to fallback on (analogous to software versions) served both as reassurance to continue attempting to build their circuits, as well as, served as a technical base to build upon.

LESSONS LEARNED

Given the general growth in interest in makerspaces and HCI4D, in this section we discuss lessons learned as researchers running a 'maker' workshop, based on the technical, infrastructural, and social challenges observed at the Kar School.

Engaging with the School

We faced two primary challenges when engaging with the school. The first challenge was to gain permission to run a maker workshop in a school in India. We approached several public and private schools in peri-urban areas of Bengaluru, prior to running the workshop at the Kar school. Due to the non-negotiable curricula and the examination-driven structure followed by all the schools [29], convincing the school principal to allow students to participate in an activity that is outside their curriculum was a challenge. School principals were reluctant to let their students miss scheduled in-class learning to learn skills that were not being tested in the examinations. There were several ways we addressed such concerns. First, we had to convince the principal that learning new skills (programming, electronics) could help students in their future education and employment. Second, we had to limit the workshop hours, so that students did not miss several in-class learning hours. Third, we also learned that the school principal needed to be in control of participant recruitment. As already discussed, the principal selected students who performed academically well, as he considered them capable of coping with the break from in-class learning. The biased recruiting strategies can be a challenge for studies which aim to measure students' improved technology literacy.

The second challenge was our limited opportunities for continued observations. After the workshop, we donated the electronics to the school to allow students to continue exploring. One student team built a project for the school science fair after the workshop. However, from an informal interview conducted after the workshop, we gathered that students did not have free access to the electronics – the school principal had securely locked away the electronics to avoid damage and distraction during examinations. As already discussed, the poor chance of repair or replacement, discourages teachers from giving students opportunities for direct interaction [30]. In addition, practising skills outside of the curricula is not favoured. Because students were limited from further hands-on exploration, we as researchers had fewer opportunities for in-the- wild observations.

Supporting Students

We learned two lessons related to supporting students in India-like contexts: (a) peer-support to scaffold self-directed learning is essential, and (b) allow learners to create self-assistance structures.

In impoverished contexts, where there are limited independent learning resources and limited mentor support, we learned that peer-support to scaffold self-directed learning is essential. Learners trained in rigid educational contexts are required to work individually and are not allowed to interact with other students during class [40]. Also, the young learners trained in rigid educational structures are accustomed to teacher-driven learning [40, 41]. However, a fundamental requirement for DIY is self-direction. Alternative to the traditional, students learn self-direction if they are placed in an environment where they are somewhat forced to take ownership over the learning process (as demonstrated by the local student strategies resilience and learning techniques). Peer-support can benefit from technology. For example, creating social support structures – such as online communities of novice learners, a project showcase in school where students can speak about their experiences and student-organized workshops where peers can teach each other – is one way to leverage resilience and non-verbal and verbal learning strategies demonstrated by students [35]. Additionally, creating technology that enables more process-focused rather than end product-focused documentation [59] may help self-directed experimentation.

To overcome the independent learning resources challenge, learners create self-assisting learning resources to help themselves (S3). For example, learners developed informal documentation and circuit fallback plans to assist themselves in the circuit building process and, more generally, problem solving. Technology can help learners to create self-assistance structures that can be referred to when needed. For example, in our workshop students created logbook-like documentation. More widely accessible digital documentation can be created for instance using mobile devices (e.g. [17, 5]) and multimedia-based collaborative note taking tools (e.g. [16, 21]). Such approaches may help young learners to create textual and multimedia reference points for both programming and circuit building (e.g. capturing images of the circuit, capturing videos of circuit building, annotating images, creating list of important pointers) in the absence of rich independent learning sources.

Researcher as a Mentor

In sharp contrast to the teacher-centric and top down schooling environment at Kar-like schools, during the workshop the researcher assumed the role of a mentor. On the first day of the workshop, the mentorship role was guided by the theory of ZPD (zone of proximal development) [60]. The theory of ZPD argues that children will not learn much if they were left to discover everything on their own. The theory suggests that in the beginning the learner should be guided and assisted to help attain the necessary minimal skills. To allow students to ease into the DIY process and to encourage novice participation, methods should be employed that start simple and slowly move towards open-ended activities. For example, starting with a simple guided linear activity (e.g. making an LED blink as used in our study), will provide students an entry point to get a glimpse of what is possible. Following structured exercises, students can be given more freedom to explore possibilities and self-directed ideas. If the introduction activity is simple to follow learners are encouraged by the initial success of completing the activity, easing the transition from the linear activity to open-ended exploration. For example, after the guided single LED blinking activity, 5 out of 6 groups experimented with connecting several more LEDs to their circuit.

We also learned that an inquiry-based approach [1] to mentoring was helpful (especially during the 'make-a-thon'). Students at the Kar school are used to imitation. As already discussed, imitation can be harmful if the students' understanding is limited or all together wrong [41]. However, by guiding students' by asking questions, one can steer the students' towards understanding the problems and exploring alternative course of action. For example, G3 imitated G1 and proposed to build a "Fan". However, upon being asked about the expected behaviour of this project by the researcher, it was found that G3 was unaware of what the "Fan" project entailed.

Maker Tools

From our study we learned that choice of electronics is affected by four variables: availability, transferability of skills, learning curve and learning goals. Conventional DIY electronics like Arduino are the more viable microprocessor option for learners who may need to work with off-the-shelf components, for availability or cost reasons. Arduino-like platforms are also more accessible (the design is open-source). In a spectrum spanning transferability of skills and learning curve, Arduino-like platforms fall on one end of the spectrum [14]. While Arduino has a steep learning curve, it affords learning of traditional electronics skills (e.g., breadboards, wires, components) which are highly transferable compared to self-contained educational platforms (e.g. littleBits or Makey-Makey). In addition, although Arduinos have steep learning curve, they have a "high-ceiling", i.e. with gradual increase in learner's technical competence, the learning is extensible to explore more complex projects [55].

We also learned that there exists a similar trade-off for choice of programming environment: ease of access versus ability to remix code. In our study we introduced participants to both visual programming environment, ArduBlocks, and the more traditional line programming, Arduino IDE. However, we learned that students gravitated towards using an approach that facilitated easy copy-paste and remixing of code (also observed by Kafai et al. [26]). While visual programming is easy to approach, most starter kit books and online resources include non-visual programming sample code, making line coding a more viable option (especially in Kar-like contexts where textbooks are the main source of information).

LIMITATIONS

We conducted a short observational study of students using a single DIY platform, Arduino, in a high school in India. Our study represents one of many possible resource-constrained settings. Our design recommendations may also apply to other communities of learners with similar constraints. Economic constraints and rigid education culture exist in other developing country contexts. For example, Sipitakiat et al. [52] observed economic challenges in Brazil similar to those observed in India which inhibit DIY from taking hold. Lin and Shaer [33] noted that students from impoverished communities in South Africa experience reduced teaching quality, and found that littleBits can help improve students computational thinking. We encourage future researchers to examine how our findings apply to other learner groups with similar constraints.

CONCLUSION AND FUTURE WORK

This paper has contributed to an understanding of how students in an impoverished and rigid education culture context (here, a high school in India) would react to the possibility to practice DIY-based activities. Our study at the Kar school shows that young learners face both resource and psychological limitations for engaging in DIY physical computing activities. However, within the constraints, students were observed to be persistent and strived to overcome some of the challenges by creating local strategies. Based on the results, we discussed a set of lessons learned and challenges faced to inform future researchers exploring makerspaces in contexts with technical, infrastructural and social challenges.

We have several directions for future work. One direction is to realize our technology design recommendations and study the effects on engaging India-like learners via our prototype systems. Second, is to study the effects of our technology design with two different unique learner groups (e.g. India-like learners and 'at-risk' youth) to compare and contrast how technology for DIY should be designed for young learners who are less ready to get involved in DIY. Third, it would be interesting to learn if there is a real improvement in STEM learning when India-like learners are engaged in DIY-based activities in comparison to traditional learning activities.

ACKNOWLEDGMENTS

We would like to thank the Kar school for granting us permission to run our study. We would also like to thank all our participants for taking part in the workshop. This research was supported by the AITF scholarship.

REFERENCES

1. Kathryn Alesandrini and Linda Larson. 2002. Teachers bridge to constructivism. *The Clearing House* 75, 3 (2002), 118–121.
2. WS Anthony. 1973. Learning to discover rules by discovery. *Journal of Educational Psychology* 64, 3 (1973), 325.
3. Ardublock. 2016. Ardublock: a Graphical Programming Language for Arduino. http://blog.ardublock.com/. (2016). [Online; accessed 2016].
4. Angela Calabrese Barton, Edna Tan, and Day Greenberg. 2016. The Makerspace Movement: Sites of Possibilities for Equitable Opportunities to Engage Underrepresented Youth in STEM. *Teachers College Record* (2016).
5. Aaron Bauer and Kenneth Koedinger. 2006. Evaluating the effect of technology on note-taking and learning. In *CHI'06 Extended Abstracts on Human Factors in Computing Systems*. ACM, 520–525.
6. Ayah Bdeir. 2009. Electronics as material: littleBits. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*. ACM, 397–400.
7. Helen Beetham and Rhona Sharpe. 2013. *Rethinking pedagogy for a digital age: Designing for 21st century learning*. routledge.
8. Paulo Blikstein. 2008. Travels in Troy with Freire. *Social justice education for teachers: Paulo Freire and the possible dream* (2008), 205–244.
9. Paulo Blikstein. 2013. Digital fabrication and 'making' in education: The democratization of invention. *FabLabs: Of machines, makers and inventors* (2013), 1–21.
10. Kenneth A Bruffee. 1984. Collaborative learning and the "Conversation of Mankind". *College English* 46, 7 (1984), 635–652.
11. Leah Buechley, Mike Eisenberg, Jaime Catchen, and Ali Crockett. 2008. The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 423–432.
12. David Cavallo, Paulo Blikstein, Arnan Sipitakiat, Anindita Basu, Alexandra Camargo, Roseli de Deus Lopes, and Alice Cavallo. 2004. The city that we want: generative themes, constructionist technologies and school/social change. In *Advanced Learning Technologies, 2004. Proceedings. IEEE International Conference on*. IEEE, 1034–1038.
13. Karin Knorr Cetina. 2009. *Epistemic cultures: How the sciences make knowledge*. Harvard University Press.
14. Joshua Chan, Tarun Pondicherry, and Paulo Blikstein. 2013. LightUp: an augmented, learning platform for electronics. In *Proceedings of the 12th International Conference on Interaction Design and Children*. ACM, 491–494.
15. Gretchen Rhines Cheney, Betsy Brown Ruzzi, and Karthik Muralidharan. 2005. A profile of the Indian education system. *Prepared for the New Commission on the Skills of the American Workforce* (2005).
16. Patrick Chiu, Ashutosh Kapuskar, Sarah Reitmeier, and Lynn Wilcox. 1999. NoteLook: Taking notes in meetings with digital video and ink. In *Proceedings of the seventh ACM international conference on Multimedia (Part 1)*. ACM, 149–158.
17. Liwei Dai, Wayne G Lutters, and Carlie Bower. 2005. Why use memo for all?: restructuring mobile applications to support informal note taking. In *CHI'05 extended abstracts on Human factors in computing systems*. ACM, 1320–1323.
18. Dale Dougherty. 2013. The maker mindset. *Design, make, play: Growing the next generation of STEM innovators* (2013), 7–11.
19. Brian W Evans. 2007. Arduino Programming Notebook. *Publeshed: First Edition August* (2007).
20. Neil Gershenfeld. 2005. Fab: Personal fabrication, fab labs, and the factory in your computer. (2005).
21. Werner Geyer, Heather Richter, Ludwin Fuchs, Tom Frauenhofer, Shahrokh Daijavad, and Steven Poltrock. 2001. A team collaboration space supporting capture and access of virtual meetings. In *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*. ACM, 188–196.
22. Joshua P Gutwill, Nina Hido, and Lisa Sindorf. 2015. Research to Practice: Observing Learning in Tinkering Activities. *Curator: The Museum Journal* 58, 2 (2015), 151–168.
23. Erica Rosenfeld Halverson and Kimberly Sheridan. 2014. The maker movement in education. *Harvard Educational Review* 84, 4 (2014), 495–504.
24. Scarlett R Herring, Chia-Chen Chang, Jesse Krantzler, and Brian P Bailey. 2009. Getting inspired!: understanding how and why examples are used in creative design practice. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 87–96.
25. Michael JA Howe. 1974. The utility of taking notes as an aid to learning. *Educational Research* 16, 3 (1974), 222–227.
26. Yasmin B Kafai, Eunkyoung Lee, Kristin Searle, Deborah Fields, Eliot Kaplan, and Debora Lui. 2014. A crafts-oriented approach to computing in high school: Introducing computational concepts, practices, and perspectives with electronic textiles. *ACM TOCE* 14, 1 (2014), 1.

27. Yasmin B Kafai, Kristin Searle, Eliot Kaplan, Deborah Fields, Eunkyoung Lee, and Debora Lui. 2013. Cupcake cushions, scooby doo shirts, and soft boomboxes: e-textiles in high school to promote computational concepts, practices, and perceptions. In *Proceeding of the 44th ACM technical symposium on Computer science education*. ACM, 311–316.

28. Geeta Kingdon. 1996. The Quality and Efficiency of Private and Public Education: A Case-Study of Urban India. *Oxford Bulletin of Economics and Statistics* 58, 1 (1996), 57–82.

29. Geeta Gandhi Kingdon. 2007. The progress of school education in India. *Oxford Review of Economic Policy* 23, 2 (2007), 168–195.

30. Krishna Kumar. 1988. Origins of India's "textbook Culture". *Comparative Education Review* 32, 4 (1988), 452–464.

31. Stacey Kuznetsov, Laura C Trutoiu, Casey Kute, Iris Howley, Eric Paulos, and Dan Siewiorek. 2011. Breaking boundaries: strategies for mentoring through textile computing workshops. In *Proc. of the SIGCHI*. ACM, 2957–2966.

32. Ladies learning Code. 2016. Digital Literacy for Women and Youth. http://ladieslearningcode.com/program/girls-learning-code/. (2016). [Online; accessed 2016].

33. Veronica Lin and Orit Shaer. 2016. Beyond the Lab: Using Technology Toys to Engage South African Youth in Computational Thinking. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 655–661.

34. Silvia Lindtner, Shaowen Bardzell, and Jeffrey Bardzell. 2016. Reconstituting the utopian vision of making: HCI after technosolutionism. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1390–1402.

35. Emily Lovell and Leah Buechley. 2011. LilyPond: an online community for sharing e-textile projects. In *Proceedings of the 8th ACM conference on Creativity and cognition*. ACM, 365–366.

36. Make:. 2016. Make: Education. http://makezine.com/tag/education/. (2016). [Online; accessed 2016].

37. Sylvia Libow Martinez. 2013. Invent To Learn: Making, Tinkering, And Engineering In The Classroom Author: Sylvia Libow Martinez, Gary S. Stager, Publis. (2013).

38. Hamish McAlpine, Ben J Hicks, Greg Huet, and Steve J Culley. 2006. An investigation into the use and content of the engineer's logbook. *Design Studies* 27, 4 (2006), 481–504.

39. Sugata Mitra. 2005. Self organising systems for mass computer literacy: Findings from the 'hole in the wall' experiments. *International Journal of Development Issues* 4, 1 (2005), 71–81.

40. Sugata Mitra, Ritu Dangwal, Shiffon Chatterjee, Swati Jha, Ravinder S Bisht, and Preeti Kapur. 2005. Acquisition of computing literacy on shared public computers: Children and the" hole in the wall". *Australasian Journal of Educational Technology* 21, 3 (2005), 407.

41. A Mukherjee. 2002. Build Robots Create Science–A Constructivist Education Initiative for Indian Schools. *Proceedings of Development by Design, Bangalore, India* (2002).

42. Karthik Muralidharan and Michael Kremer. 2006. Public and private schools in rural India. *Harvard University, Department of Economics, Cambridge, MA* (2006).

43. Dayo Olopade. 2014. *The bright continent: Breaking rules and making change in modern Africa*. Houghton Mifflin Harcourt.

44. Seymour Papert and Idit Harel. 1991. Situating constructionism. *Constructionism* 36 (1991), 1–11.

45. Udai Singh Pawar, Joyojeet Pal, and Kentaro Toyama. 2006. Multiple mice for computers in education in developing countries. In *Information and Communication Technologies and Development, 2006. ICTD'06. International Conference on*. IEEE, 64–71.

46. Leïla Perié, Livio Riboli-Sasco, and Claire Ribrault. 2014. Straight into conflict zones, scientific research empowers the minds. *LISTENING AND EMPOWERING* (2014), 69.

47. Nimmi Rangaswamy and Melissa Densmore. 2013. Understanding Jugaad: ICTD and the tensions of appropriation, innovation and utility. In *Proceedings of the Sixth International Conference on Information and Communications Technologies and Development: Notes-Volume 2*. ACM, 120–123.

48. Nimmi Rangaswamy and Nithya Sambasivan. 2011. Cutting Chai, Jugaad, and Here Pheri: towards UbiComp for a global community. *Personal and Ubiquitous Computing* 15, 6 (2011), 553–564.

49. Kimberly M Sheridan and Abigail Konopasky. 2016. Designing for Resourcefulness. *Makeology: Makerspaces as Learning Environments* 1 (2016), 30.

50. Arnan Sipitakiat. 2001. *Digital technology for conviviality: making the most of students' energy and imagination in learning environments*. Ph.D. Dissertation. Massachusetts Institute of Technology.

51. Arnan Sipitakiat and Paulo Blikstein. 2010. Think globally, build locally: a technological platform for low-cost, open-source, locally-assembled programmable bricks for education. In *Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction*. ACM, 231–232.

52. Arnan Sipitakiat, Paulo Blikstein, and David P Cavallo. 2004. GoGo board: augmenting programmable bricks for economically challenged audiences. In *Proceedings of the 6th international conference on Learning sciences*. International Society of the Learning Sciences, 481–488.

53. Thomas N Smyth, Satish Kumar, Indrani Medhi, and Kentaro Toyama. 2010. Where there's a will there's a way: mobile media sharing in urban india. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 753–762.

54. Sowmya Somanath, Janette Hughes, Ehud Sharlin, and Mario Costa Sousa. 2015. The Concept of DIY as an Education Model for Indian Public Schools. In *CHI'15 Extended Abstracts*. ACM.

55. Sowmya Somanath, Laura Morrison, Janette Hughes, Ehud Sharlin, and Mario Costa Sousa. 2016. Engaging 'At-Risk' Students Through Maker Culture Activities. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '16)*. ACM, 150–158.

56. SparkFun. 2016. SparkFun Inventor's Kit Guidebook - V3. https://www.sparkfun.com/products/retired/11976/. (2016). [Online; accessed 2016].

57. Gary S Stager. 2013. Papert's prison fab lab: implications for the maker movement and education design. In *Proceedings of the 12th International Conference on Interaction Design and Children*. ACM, 487–490.

58. Joshua G Tanenbaum, Amanda M Williams, Audrey Desjardins, and Karen Tanenbaum. 2013. Democratizing technology: pleasure, utility and expressiveness in DIY and maker practice. In *Proc. of the SIGCHI*. ACM, 2603–2612.

59. Tiffany Tseng and Mitchel Resnick. 2014. Product versus process: representing and appropriating DIY projects online. In *Proceedings of the 2014 conference on Designing interactive systems*. ACM, 425–428.

60. L Vygotsky. 1987. Zone of proximal development. *Mind in society: The development of higher psychological processes* (1987), 52–91.

61. Jagoda Walny, Jonathan Haber, Marian Dörk, Jonathan Sillito, and Sheelagh Carpendale. 2011. Follow that sketch: Lifecycles of diagrams and sketches in software development. In *Visualizing Software for Understanding and Analysis (VISSOFT), 2011 6th IEEE International Workshop on*. IEEE, 1–8.

62. Stephen C Want and Paul L Harris. 2002. How do children ape? Applying concepts from the study of non-human primates to the developmental study of 'imitation' in children. *Developmental Science* 5, 1 (2002), 1–14.