Making Progress: Barriers to Success in End-User Developers' Physical Prototyping

Tracey Booth
Centre for Human-Computer Interaction Design,
School of Mathematics, Computer Science and Engineering
City University London, UK
tracey.booth.1@city.ac.uk

***Abstract*— Previous research has investigated difficulties faced by novice and end-user programmers and suggested how these could be solved. However, there has been scant research investigating barriers faced by end-user developers (EUDs) constructing and programming physical prototypes, and how these could be overcome. I report the results of an empirical study designed to uncover barriers for EUDs in physical prototyping, with a view to design support for overcoming them.**

I. Introduction

As part of the current Maker Movement, a growing number of end-user developers (EUDs) - artists, designers, researchers, and hobbyists - are creating interactive physical prototypes, to support their personal and work interests. Physical prototyping platforms like Arduino aim to make developing microcontroller-based devices and systems much easier than previously possible, however, the challenges faced by an EUD can still be considerable: they must learn and apply both programming and electronics concepts, and also develop some understanding of the relationships between the virtual (programming) and physical (circuit) aspects in order to solve problems that arise. However, to date, there has not been much research into the barriers faced by EUDs as they develop physical prototypes, in order to offer better support to overcome them.

The overarching aim of my PhD research is to investigate the barriers adult EUDs face when creating physical prototypes, to determine the mental models [1] they hold and construct as they learn to develop physical prototypes, and to investigate possible ways to support EUDs to develop physical prototypes more effectively and efficiently. I will achieve this through a number of empirical studies, the first of which is described later in this paper.

II. Background and Related Work

Previous research has investigated difficulties faced by both novice programmers and end-user programmers ([2], [3], and [4]). Ko et al. identified six learning barriers: Design, Selection, Coordination, Use, Understanding, and Information - knowledge breakdowns that can lead to further barriers [3]. In my previous work, I have shown that learning barriers occur in *programming* physical prototypes [5]. However, so far there is scant research in investigating barriers for EUDs when constructing physical prototypes which combines elements of both programming and engineering.

There have been numerous approaches suggested for helping programmers. One strand has focused on developing tools that make programming easier for novices [6] or offer tool-based support to programmers when they get stuck [4], while other work has focused on teaching and educating programmers. However, we do not understand enough yet about physical prototyping to design appropriate support for EUDs.

III. Preliminary Study

I conducted an empirical study designed to uncover what and where barriers occur for EUDs when developing physical prototypes, and EUDs' mental models of the concepts involved in constructing and programming physical prototypes.

A. *Study Set-up*

Twenty EUDs (8 female, 12 male, mean age 31.8) with limited experience of using Arduino took part in the study. Prior to the main task, participants completed surveys gathering background, programming and physical prototyping experience, demographic information, and self-efficacy. For the main task, participants were asked to create a physical prototype from scratch to show an increase in temperature through a series of lit-up LEDs (a simple "love-o-meter" often used as a tutorial project for Arduino). Participants were given equipment and the standard Arduino IDE. They were allowed to use help content and examples built into the IDE and search online for other resources, enabling us to observe how they tried to overcome barriers. Participants were asked to think aloud as they undertook the task. They were given 45 minutes to complete this task.

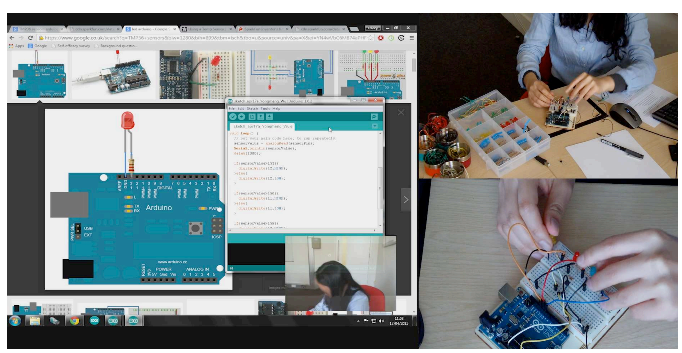

Fig. 1. Split screen, multi-viewpoint video of physical prototyping task

The task was followed by a semi-structured interview to elicit participants' mental models of the physical prototyping concepts involved in the task. The session was video-recorded for later analysis (Fig. 1).

Video transcripts were coded for barriers encountered by participants. For this coding scheme, a barrier was defined as an impediment to progress, either slowing progress, or even, in the case of an insurmountable barrier, resulting in not completing the task. Barriers were coded on evidence of knowledge gaps or breakdowns, including errors in judgment or action, evidence of inadequate/incomplete knowledge, or uncertainty. In short, on evidence of any obstacle to progress.

B. *Results*

Analysis of the study data is still in progress but results so far show the magnitude and location of where barriers occurred (Fig. 2). Every participant was in some way impeded in their progress; on average, 54.6 barriers were found per participant over the 45 minutes they worked on the task (SD: 22.12, total: 1092). Most barriers occurred in Programming (mean: 29.35, SD: 14.7, total: 587), followed by Circuit barriers (mean: 15.6 SD: 8.46, total: 312), while on average 8.25 barriers per participant were found in coordinating Program and Circuit, and understanding their relationship to ouput (SD: 14.7, total: 165). Only few barriers stemmed from using the IDE tool (mean: 1.45 SD: 2.42, total barriers: 21).

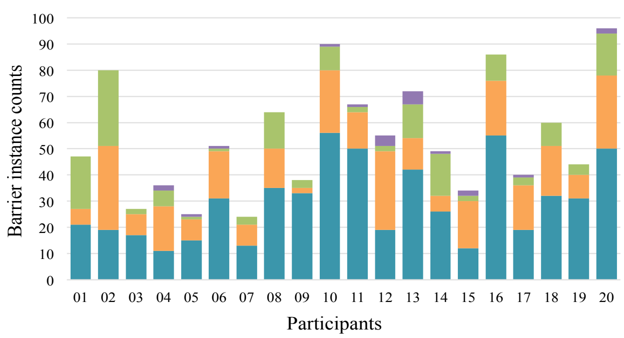

Fig. 2. Participants' barrier instances and locations, in: Program (blue), Circuit (orange), Program-Circuit Coordination (green), IDE (purple)

The number of barriers encountered also explains the low task completion. Of the 20 participants, only 6 succeeded in completing the task. By far the main cause of failure (10 participants) was error in circuit construction, although participants did not always realize this. Three participants failed to complete the task due to programming error, however, they all managed to construct the circuit correctly. The remaining participant encountered an insurmountable barrier in the IDE: unable to find out how to view output in the IDE, he was unable to determine what conditions to set in his program.

While some barriers were easily navigated and resolved, others led to task failure. Some barriers occurred early on and participants spent most of the remainder of the session trying unsuccessfully to overcome them. In several cases, incorrect diagnosis of error location meant that participants modified their program when the cause of error lay in their circuit (or vice versa); this sometimes introduced further error.

We analyzed task completion in conjunction with several measures from the background questionnaires. We found no relationships between task completion and self-efficacy, training, self-rated expertise or employment, however we will analyze the data in greater depth as we proceed.

C. *Remaining analyses*

My next step involves analyzing the barriers as to what kinds of barriers they are. For this I will draw on Ko's learning barriers, Reason's error classification [7], and inductive coding.

I will also analyze the post-task interview and task observation videos to gain insight into participants' mental models of the physical prototyping concepts involved in the task. Once complete, this will enable me to determine whether there are relationships between the types of barriers participants experience and their mental models.

I also plan to investigate whether there are any relationships between participants' backgrounds (including self-efficacy), the mental models they hold, and the barriers they encounter.

IV. Future Work

The remainder of my PhD research will be directed by the findings from the above study. Identifying the most prevalent barriers affecting EUDs will pinpoint where EUDs encounter most difficulty in physical prototyping, while relationships discovered between those barriers and the mental models EUDs hold of the concepts involved should provide a foundation upon which to design support. My final step will be to evaluate the support mechanisms for EUDs I have designed. Together my findings should be of use to those designing new physical prototyping systems or enhancements to existing ones, and may also be of interest to those researching physical prototyping with other groups, for example, young people, or EUDs in other domains.

References

[1] D. Norman, 'Some observations on mental models', in Mental Models, D. Gentner and A. L. Stevens, Eds. Hillsdale, NJ, USA: Lawrence Erlbaum Associates, Inc., 1983, pp. 7–14.

[2] J.F. Pane and B.A. Myers, "Usability issues in the design of novice programming systems," Carnegie Mellon University, School of Computer Science Technical Report CMU-CS-96-132, Pittsburgh, PA, August 1996, 85 pages.

[3] A. J. Ko, B. A. Myers, and H. H. Aung, 'Six learning barriers in end-user programming systems', in Proceedings of the 2004 IEEE Symposium on Visual Languages & Human Centric Computing, Washington, DC, USA, 2004, pp. 199–206.

[4] J. Cao, S. D. Fleming, M. Burnett, and C. Scaffidi, 'Idea Garden: Situated support for problem solving by end-user programmers', Interacting with Computers, May 2014.

[5] T. Booth and S. Stumpf, 'End-user experiences of visual and textual programming environments for Arduino', in End-User Development, Y. Dittrich, M. Burnett, A. Mørch, and D. Redmiles, Eds. Springer Berlin Heidelberg, 2013, pp. 25–39.

[6] C. Kelleher and R. Pausch, 'Lessons learned from designing a programming system to support middle school girls creating animated stories', in IEEE Symposium on Visual Languages and Human-Centric Computing, 2006, pp. 165–172.

[7] J. Reason, Human Error. Cambridge England ; New York: Cambridge University Press, 1990.