UBIQUITIOUS COMPUTING DEVICES IN THE TRAINING OF TEACHER-TRAINERS

Neil Smith, The Open University
Helen Caldwell, University of Northampton
United Kingdom

Abstract

In September 2014, the computing curriculum in English schools changed to one with a much greater emphasis on computer science. However, 66% of existing ICT teachers are non-specialist and require significant continuing professional development (CPD) to deliver this new curriculum. One initiative to provide this is the Computing At School (CAS) Master Teacher programme. This paper describes some physical computing projects that were used in training a cohort of Master Teachers, preparing them to deliver both improved lessons in classrooms and CPD tailored for the requirements of their peers.

Introduction

In September 2014, the computing curriculum in English schools changed to one with a much greater emphasis on computer science, compared to the previous curriculum mainly based on ICT and digital literacy (Department for Education, 2013). Of UK computing school teachers, 66% are non-specialists in computer science who do not have the subject-specific skills or experience to deliver the new curriculum as effectively as they would like (Furber, 2012). One initiative to address this deficit in skill is the Computing At School (CAS) Master Teacher programme (Computing At School, 2015), where technically competent in-service teachers provide professionally relevant professional development to their peers within and across schools.

This paper describes our experience with the CAS Master Teaching training programme, where we train the master teachers. The training involves a combination of technical education and training the teachers how to deliver effective continuing professional development (CPD) to adult learners. We employed a blended learning approach, combining face-to-face sessions, online tutorial support, and guided review of example practice by the teachers in delivering CPD.

We use various physical computing devices in the training to illuminate key computer science principles, as well as show the trainee master teachers how these devices can be used for CPD delivery and directly in classrooms with children. As well as the use of a variety of robots and physical computing projects, we also use the SenseBoard (see below), a novel device developed by the Open University for supporting ubiquitous computing and internet-of-things applications.

Teaching Approach

Training for master teachers is a two-stage process. Level 1 training is focussed on developing subject knowledge expertise (SKE) in the teachers, to ensure they have sufficient mastery of their subject. Level 2 training focusses on how to develop and deliver CPD for other teachers. We were providing Level 1 training.

Our cohort consisted of ten in-service teachers, split between five primary school teachers and five secondary school teachers. (Primary school covers ages 5 to 11; secondary school covers ages 11-18.) The training provision suggested by Computing At School suggested a blended learning approach. All teachers received five days of face-to-face training plus ten hours of online tutorial support; the secondary school teachers received an additional five days of face-to-face training to cover the additional subject knowledge expertise requirements. The online tutorial support was delivered through a combination of video conferencing using Google Hangouts and email. The face-to-face teaching time was supplemented by the trainees undertaking various activities between the contact periods, where the trainees had to perform various tasks such as preparing sample CPD material and reflecting on what they had learnt and how that could be used to improve their practice.

The face-to-face teaching was supplemented by the use of online tools where trainees could develop and share resources created during the training. These resources included learning resources taken from various places online and resources created by the trainees both during the sessions and elsewhere.

Ubiquitous and Physical Computing Devices

Physical computing has recently been seen as a first step in getting novices and children engaged with computer science (e.g., Buechley, Eisenberg, Catchen, & Crockett, 2008, Lau, Ngai, Chan, & Cheung, 2009, Richards, Petre, & Bandara, 2012, Richards & Smith 2010). The use of physical computing devices has several benefits over a software-only approach to education. The physical device offers a tangible focus of attention for the learner, the use of a playful and interactive device can reduce feelings of insecurity in learners, and the physical device can offer immediate and obvious feedback on progress.

As part of the teacher training, we have demonstrated a variety of physical computing devices and outlined their different pedagogic applications for both teaching children and for delivering CPD to teachers (specialist and non-specialist alike). In the remainder of this section, we outline the devices demonstrated. In the next section, we outline how these devices have been used in schools, both by the trainee master teachers and others.

SenseBoards

The SenseBoard (Figure 1) is a tethered device based around the Arduino microcontroller. It was developed for novice computer science students working at a distance in the UK's Open University, as part of the module *My Digital Life* (Richards et al., 2012). We wanted a way of introducing our starting students, many of whom have never before studied any form of computer science, to computing in a gentle and immediate way. The use of physical computing was a natural way to introduce the creative and practical aspects of computer science (Richards & Smith 2010).

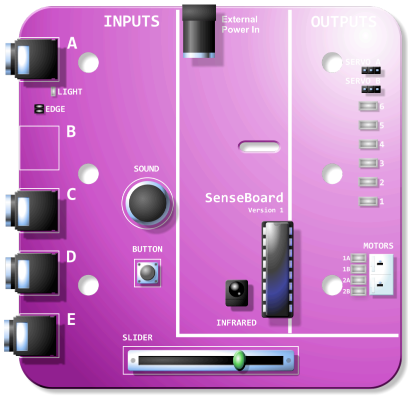

Figure 1. A schematic view of a SenseBoard.

The SenseBoard was designed as a ubiquitous computing lab-in-a-box for teaching undergraduate students at a distance. The SenseBoard's robust construction and ease of use make it suitable for use in classrooms. The accompanying Sense programming language (based on Scratch) allows even young children easily to develop internet-aware physical computing devices.

The Open University teaching context of solely distance learning leads to some interesting constraints on how a physical computing environment can be delivered. One constraint is that the SenseBoard kit supplied to the students should be self-contained; there is no facility for students to pop into a lab and collect additional equipment for specific projects. Another major constraint is that support and troubleshooting of the devices is difficult and often slow. If the device does not work, students, working at a distance, will often spend considerable time attempting to fix the problem before contacting tutors or other support personnel. These contacts are likely to be by telephone (or similar) or email. This is a rather different context from a traditional university, where students use physical computing devices in labs where skilled staff and technicians are available to step in and swiftly resolve trivial, but show-stopping, configuration or hardware errors. The device must be sufficiently reliable to have a very low manufacturing defect rate, survive postage to the student, and continue to function after over a year in the untidy environment of a family home, with all the attendant possible insults from pets and small children that are present in a home environment. Finally, the kit must be cheap enough that it need not be returned to the Open University on completion of a student's studies, as past experience has shown that the cost of receiving and refurbishing the kits is prohibitive.

The SenseBoard has, as the name implies, a number of sensors mounted on the board: a slider, a non-latching push button, a microphone, an infra-red sensor that detects signals from remote controls, and four 3.5mm sockets for plugging in additional sensors. The SenseBoard kit comes with a tilt sensor, a temperature sensor, and a light sensor; students can easily make or connect additional ones, such as pressure sensors and rheostats. The board also has some outputs: mounted on the board is a bank of seven LEDs in various colours, and the kit also comes with an IR LED on a long lead and a stepper motor, both of which can be plugged in to the board. There is scope to connect

DC motors and servomotors, but we do not supply these in the kit. This range of sensors and outputs means the SenseBoard is a flexible physical computing device capable of many uses. Everything that plugs in to the SenseBoard does so with simple non-reversible sockets that do not require the insertion of leads into small holes in a breadboard or the manipulation of small individual components. The SenseBoard connects to a host computer via a USB cable. To keep things simple, the SenseBoard is not capable of autonomous operation and must be controlled by a host computer.

Given the context of students starting computer science studies at a distance, the supplied programming environment also required that students be able to get started easily with the programming environment without being held back by trivial syntax errors common in novice programmers working with traditional textual programming languages for the first time. Therefore, we developed our own programming environment, Sense, based on the popular Scratch graphical, block-based, programming environment (Maloney, Resnick, Rusk, Silverman, & Eastmond, 2010). Sense was based on Scratch 1.3 and extended to make it suitable for undergraduate study. Main extensions were introduction of list variables, inclusion of blocks to control and read the SenseBoard and to read and write text files, and addition of blocks for network communication. Various other changes were made, including addition of more data manipulation blocks and numerous user interface changes.

The network communication blocks allow Sense to read arbitrary content from the Web, but also have dedicated support for reading RSS feeds, exposing the content of the feed as a list-like structure. We also allow Sense to write data to a dedicated server run by the Open University; this data is made available as both a simple web page and as an RSS feed, suitable for reading by Sense. This feature allows for individual students to write data such as logs, for students to view each others' data for collaboration on group projects, and for near real-time communication between students, either for chat, distributed presence, or simple game controls. (The extension of lists was folded back into the main Scratch 1.4 and hence Scratch 2, developed by MIT.)

While the most often used interface for the SenseBoard is Sense, we have also developed a Python library for driving the SenseBoard (Smith & Smith 2015).

Raspberry Pi

The Raspberry Pi (Upton & Halfacree, 2014) is a small, low-cost, single board microcomputer developed to give children a first experience of *unpackaged* computing, separate from the world of apps and ready-made software. To enhance this experience, the Raspberry Pi comes with a set of general-purpose input/output (GPIO) pins that allow it to interface with electronics and microcontrollers. Amateurs and educators have developed a great many projects, and Raspberry Pi (2015) has a curated collection of some simple projects, which illustrate the range of projects that have been completed with the Raspberry Pi.

One such set of projects demonstrated is the Miniband collection of homemade musical instruments (Smith, 2014), consisting of a drum machine, maracas, and a keyboard (see Figure 2 for a schematic of the keyboard). These projects are supplied with instructions for children to develop the instruments themselves, from components and writing the simple Python code required to drive the instrument. Each instrument takes about 90 minutes to complete, with children working in pairs.

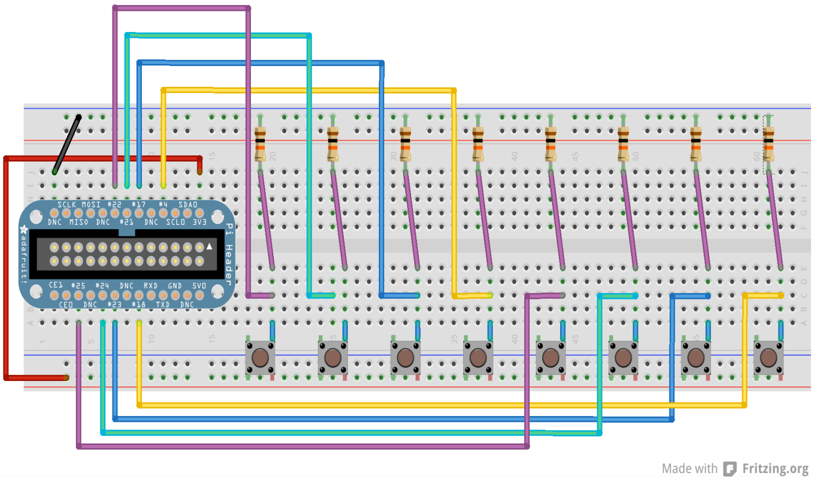

Figure 2. A schematic of wiring for a Raspberry Pi powered electronic keyboard

Junkbots

As well as physical computing devices, we also introduced the trainee teachers to a variety of robots for use in schools. One innovative class of devices is the Junkbot (Turner, 2013; Turner 2015; Turner & Tetley, 2015), which is a general family of small robots made from junk parts, such as drink cans, old DC electric motors, and marker pens (Figure 3). The robots are easily made by children and controlled using ScratchGPIO on the Raspberry Pi.

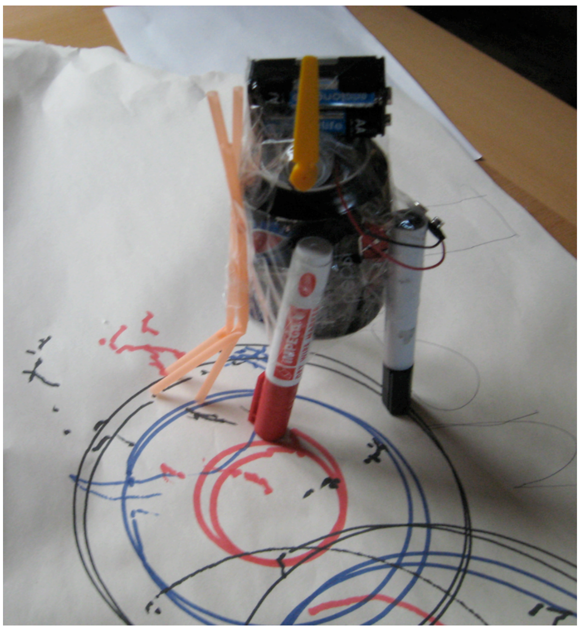

Figure 3. A Junkbot.

Reception by Master Teachers

We demonstrated these devices during some face-to-face sessions with all the master teacher trainees and invited them to experiment with the devices while we were able to provide supervision and advice. We supplied all trainees with at least one SenseBoard. Many already had several Raspberry Pis in their schools. We continued to provide support for trainees via videoconferencing as they developed their own activities using these devices.

Master teacher trainees' reception was positive for all devices presented to them, and they have already taken steps to incorporate the use of these devices into their practice, for both classroom teaching and CPD. Several have developed novel projects using these devices, such as additional musical instruments (including a theremin-like device using a rheostat connected to a Raspberry Pi) and an ambient weather display using the SenseBoard.

School Use

The authors have used most of these devices in a school environment, so we are sure of the effectiveness of them in providing an engaging context for studying computer science. They have also been used by third parties in both one-off workshops and for extended use in classroom settings.

Workshops

The constraints on the SenseBoard design, coincidentally, mean that the device is well suited for use in schools with children of all ages. We have successfully used the SenseBoards in several workshops for schoolchildren, notably the Digital Summer Camp day in 2013, where we introduced several hundred secondary school children to Sense and the SenseBoard through the development of a simple game, Sense Shooter, that used the SenseBoard as a simple game controller for playing a game of target shooting (see Figures 4 and 5). The slider was used to control the gun's aim point, and the microphone was used to detect a loud shout of "Bang!" by the child to fire the gun. Following an instruction sheet, the children had about an hour to build the game from scratch, though using the image and sound templates we provided. The simple and immediate nature of the controller was a great success in the workshop, as was the physical nature of the game controls.

Figure 4. Sense Shooter playing area.

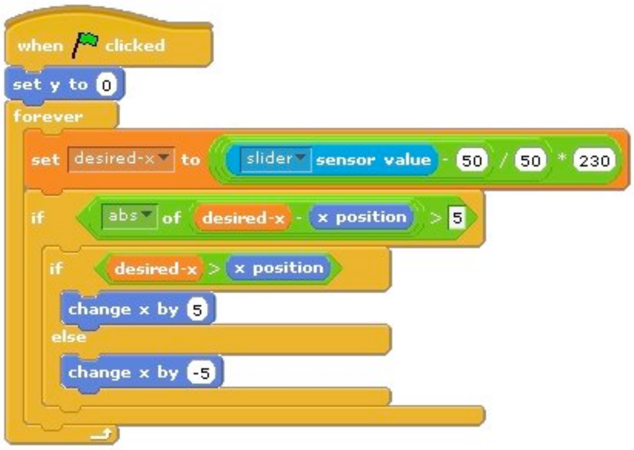

Figure 5. A sample script from the Sense Shooter program, showing the new sensor value block.

Sense has also been used in more extended workshops, where groups of children work together to develop a novel physical computing device for a particular use. Examples of projects that have come from these workshops include a visual fire alarm for the deaf that used the heat and sound sensors to detect a fire, and a sound-activated disco light (using an additional laser pointer bouncing off a home-made glitter ball, moved by the motor when a loud noise is sensed.

Classroom

The SenseBoard has also been used in the classroom in several contexts. Several secondary schools have had sets of SenseBoards for classroom use for several years and are using them across the STEM curriculum as well as in computer science and ICT lessons. They are using the SenseBoards for many of the uses outlined above, including using the sensors for data logging during science experiments, for later analysis.

Another use of SenseBoards in classrooms was as part of the Distance project (Kortuem, Bandara, Smith, Richards, & Petre, 2013), a distributed Internet of Things project across a number of UK schools. Different schools combined data from weather stations, indoor and outdoor air quality sensors, SenseBoards, and online sources of information (such as weather forecasts and traffic reports) to understand the environment within and around the schools. The schools also shared data with each other (via a central data hub) to collaborate and compare results.

It is still early days for use of these devices by the master teacher trainees in their practice, as they have only just completed their training. However, one master teacher has already successfully used the SenseBoard in classrooms and during open evenings for parents.

Continuing Professional Development

The use of these physical computing devices is novel in CPD for adults. Several of the master teachers have developed CPD resources using these devices, with many of the activities based on those included in the *My Digital Life* resources. These have included using the junkbots to illustrate the concept of algorithms to non-specialist teachers, and using multiple scripts in Sense as part of training on multi-threaded programming.

Conclusion

Physical computing devices are a powerful and useful way to entice learners into the study of computer science, and to illustrate and explain some deep concepts in the field. This paper has shown a variety of physical computing projects that are suitable for school children of all ages and which are generally quick to develop in a classroom or workshop environment. We have also illustrated some additional uses of these devices over longer-term periods, such as using the SenseBoard as an environmental condition logger, with results passed to a central server for later consolidation and analysis.

The master teacher trainees have enthusiastically taken up these devices. They have already developed several projects for various devices and are currently deploying them in their classrooms and in CPD.

References

Buechley, L., Eisenberg, M., Catchen, J., & Crockett, A. (2008). The LilyPad Arduino: Using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proc. SIGCHI conference on Human factors in computing systems (CHI).*

Computing At School. (2015). Master teacher training programme. Retrieved from http://www.computingatschool.org.uk/index.php?id=noe-master-teachers

Department for Education. (2013). National Curriculum in England: Computing Programmes of Study.

Furber, S. (2012). *Shut down or restart: The way forward for computing in UK schools*. The Royal Society

Kortuem, G., Bandara, A., Smith, N., Richards, M., & Petre, M. (2013). Educating the Internet-of-Things generation. *Computer*, *46*(2), 53–61.

Lau, W. W., Ngai, G., Chan, S. C., & Cheung, J. C. (2009). Learning programming through fashion and design: A pilot summer course in wearable computing for middle school students. In *ACM SIGCSE Bulletin, 41(*1), 504-508.

Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). The Scratch programming language and environment. *ACM Transactions on Computing Education (TOCE)*, *10*(4), 16.

Raspberry Pi. (2015). Sample projects, from http://www.raspberrypi.org/resources/make/ [Accessed 28/3/2015]

Richards, M., Petre, M., & Bandara, A. (2012). Starting with Ubicomp: Using the SenseBoard to introduce computing. In: *43rd ACM Technical Symposium on Computer Science Education*.

Richards, M., & Smith, N. (2010). Teaching UbiComp with Sense. In: *Proceedings of the 6th Nordic Conference on Human-Computer Interaction (NordiCHI)*.

Smith, N. (2014). Raspberry Pi miniband. Retrieved from http://community.computingatschool.org.uk/resources/1811

Smith, N., & Smith, A. (2015). PySense library. Retrieved from https://github.com/NeilNjae/Sense-Library
Turner, S. (2013). Junkbots. *Higher Education Academy Science Technology Engineering and Mathematics (HEA STEM) Annual Learning and Teaching Conference 2013: Where Practice and Pedagogy Meet*.
Turner, S. (2015). Junkbots. Retrieved from http://junkbots.blogspot.co.uk/
Turner, S., & Tetley, H. (2015). Raspberry Pi Junkbot. Retrieved from http://raspberrypijunkbot.weebly.com/
Upton, E., & HalfacreeB, G. (2014). *Raspberry Pi user guide*. John Wiley & Sons.

Author Details

Neil Smith
n.smith@open.ac.uk

Helen Caldwell
helen.caldwell@northampton.ac.uk

Back to Contents page