

Physical Computing im Informatikunterricht

Mareen Przybylla, Ralf Romeike

Institut für Informatik / Didaktik der Informatik

Universität Potsdam

August-Bebel-Str. 89

14482 Potsdam

przybyll@cs.uni-potsdam.de

romeike@cs.uni-potsdam.de

Abstract: Die Steuerung physischer Systeme hat in Form von Robotik bereits einen festen Platz in der informatischen Bildung eingenommen. So wie interaktive Computersysteme nun eine zunehmende Bedeutung in der Gesellschaft erlangen, ermöglicht die Entwicklung einfach programmierbarer Mikrocontroller eine neue Form kreativer Gestaltung von Informatiksystemen, die als Physical Computing bezeichnet wird und die Erstellung interaktiver Objekte und Installationen zum Ziel hat. Im Beitrag werden Ansätze des Physical Computing im Informatikunterricht diskutiert, Anforderungen an Hardware und Unterrichtsgestaltung formuliert und erste praktische Erfahrungen mit der prototypischen Gestaltungs-, Programmier- und Lernumgebung *My Interactive Garden* dargestellt.

1 Einleitung

Innerhalb der letzten Jahrzehnte erlangten interaktive Computersysteme eine immer größer werdende Bedeutung in der Gesellschaft. Sie sind mittlerweile derart in den Alltag aller Menschen integriert, dass sie vielerorts als intelligente Helfer das Leben erleichtern und bereichern. Dieser Trend spiegelt sich auch in der informatischen Bildung wider und hat – bisher vor allem in Form von Robotik – Einzug in die Schulen erhalten. Für Schüler stellt es offenbar ein besonderes Erfolgserlebnis dar, echte physische Objekte steuern können, was in der Lerntheorie des Konstruktivismus reflektiert wird [PH91]. Mit der Idee des Physical Computing, dem künstlerischen Entwickeln interaktiver Objekte, eröffnen sich auch für den Informatikunterricht neue Ansätze konstruktivistischen Lernens. Im vorliegenden Artikel erfolgt eine Diskussion verschiedener Trends und Möglichkeiten zur Programmierung physischer Objekte im Informatikunterricht. Die von den Autoren entwickelte Gestaltungs-, Programmier- und Lernumgebung „My Interactive Garden“ hat zum Ziel, einen attraktiven Zugang zum Physical Computing in der Schule zu ermöglichen, sowie dessen Stärken und Schwächen zu untersuchen. Nach der Diskussion von Physical Computing als Zugang zur Informatik werden *My Interactive Garden* sowie erste Erfahrungen im Unterrichtseinsatz vorgestellt.

2 Interaktive Systeme

Nachdem lange die Entwicklung von Software den gestalterorientierten Teil im Informatikunterricht bestimmte, wird mit Mindstorms, PicoBoards, PicoCrickets u. Ä. zunehmend die virtuelle mit der realen Welt in Verbindung gebracht, da der haptische Zugang als wichtig erachtet wird. Im Sinne des konstruktionistischen Lernansatzes erschien es wichtig, Kindern nicht nur das Erschaffen virtueller Objekte zu ermöglichen, sondern programmierbarer Systeme, mit denen in der realen Welt interagiert werden kann. Dies führte zur Entwicklung von Lego Mindstorms. In zahlreichen Unterrichtserfahrungen wird beschrieben, wie mit LEGO Mindstorms meist Fahrzeugroboter und industrielle Anwendungen, wie z. B. Hochregallager, nachgebaut werden (z. B. in [Wa05], [We08]). Kritisiert werden Mindstorms' beschränkte Möglichkeiten zur kreativen Entfaltung, was sich u. a. in einer mangelnden Beteiligung von Mädchen äußert (vgl. [Re07]).

Die technischen Entwicklungen der letzten Jahre haben dazu geführt, dass Computer nicht mehr nur als Maschinen gesehen werden, die Anweisungen erhalten und umsetzen, sondern als interaktive *eingebettete* Medien. Wie auch Roboter kennzeichnen sich eingebettete Systeme dadurch, dass Hard- und Software in ein Gesamtsystem integriert sind. Eingebettete Systeme erfüllen immer einen speziellen Zweck und unterscheiden sich von Robotern vor allem dadurch, dass sie Maschinen oder Geräte nicht auf den industriellen Bereich beschränken und auch den mechanischen Aspekt im Bereich der Akteure nicht betonen.

Um Schülern die Gelegenheit zu bieten, bekannte eingebettete Systeme ihrer täglichen Umgebung genauer zu verstehen, bietet es sich an, diese als Unterrichtsgegenstand zu thematisieren. Oft ist Schülern gar nicht bewusst, wo Informatik in ihrem Alltag überall eine Rolle spielt: beispielsweise in Waschmaschinen, im Auto oder Flugzeug, in der Medizin, in Digitalkameras, Navigationsgeräten, DVD-Playern oder Klimaanlage. Dies geht damit einher, dass Informatik von Schülern häufig als ein Unterrichtsfach wahrgenommen wird, welches abstrakte und realitätsferne Themen behandelt und daher nur für eine spezielle Klientel informatisch talentierter Schüler geeignet ist [Kn11]. Ein konkretes Beispiel zur Thematisierung eines eingebetteten Systems liefern Pelz und Arnhold [PA13] mit der Steuerung einer Modell-Waschmaschine. Einen Unterrichtsversuch mit der Implementierung eingebetteter Systeme in einem Puppenhaus beschreibt Strecker [St11]. Bei beiden Projekten wird der Vorteil der haptischen Erfahrbarkeit als positiv bewertet. Während sich diese Versuche vor allem mit der Nachahmung existierender eingebetteter Systeme beschäftigen, betont *Physical Computing* eine stärkere Einbeziehung von Aspekten aus Kunst und Design und eröffnet damit ein größeres Spektrum kreativer Möglichkeiten.

3 Physical Computing

„Physical Computing uses electronics to prototype new materials for designers and artists. It involves the design of interactive objects that can communicate with humans using sensors and actuators controlled by a behavior implemented as software running inside a microcontroller (a small computer on a single chip)“ [Ba11].

Physical Computing hat in den letzten Jahren verstärkt an Bedeutung gewonnen. Vor allem Künstler und Designer verwenden programmierbare Hardware, um interaktive physische Systeme zu kreieren, welche die analoge Welt wahrnehmen und auf sie reagieren. Physical Computing kann als kreative Tätigkeit verstanden werden, bei der interaktive Objekte und Installationen unter Zuhilfenahme von Bastel-, Kunst- oder Designmaterial hergestellt werden. Physical Computing überträgt die traditionell virtuellen kreativen Möglichkeiten der Informatik durch Einbeziehung von Aspekten aus Kunst und Design in die reale Welt und kann das Fach vielfältiger und damit attraktiver erscheinen lassen. Im Informatikunterricht wird Physical Computing bisher mit Sensor- und Aktorenboards, mit vorgefertigten Sensoren und Aktoren, sowie mit Mikrocontrollerboards durchgeführt.

3.1 Interaktive Objekte mit Sensor- und Aktorenboards

Sensorboards sind einfache Leiterplatten, die mit verschiedenen Sensoren ausgestattet sind und mit einem Computer verbunden werden. So gibt es beispielsweise das PicoBoard, welches mit Klang- und Lichtsensoren, Widerstandssensorik, einem Schalter und einem Schieberegler versehen ist und mit Scratch interagiert [PIC10]. Dadurch ist es möglich, in Scratch real existierende Werte der Umgebung als Eingaben zu benutzen. Nachteilig für die Verwendung in interaktiven Objekten ist jedoch, dass Sensorboards keine Ausgänge besitzen und daher ausschließlich als Eingabegeräte einsetzbar sind. Das SenseBoard der Open University [RS10] behebt diesen Mangel: es verfügt über zusätzliche Ausgänge. Eigens für ihre Studenten wurde als Open-Source-Projekt hierfür die auf Scratch basierende Software „Sense“ entwickelt. Somit konnte Ubiquitous Computing – und damit die Entwicklung eingebetteter Systeme –im universitären Bereich auch für Anfänger auf dem Gebiet zugänglich gemacht werden [Ma12]. Aus dem schulischen Bereich gibt es bisher jedoch keine Erfahrungsberichte. Aufgrund der Einfachheit der Handhabung rechnen wir mit einer erfolgreichen Verbreitung auch im Schuleinsatz, sobald das SenseBoard oder vergleichbare Produkte erhältlich sind.

3.2 Interaktive Objekte mit vorgefertigten Sensoren und Aktoren

Die Trennung von Aktoren und Sensoren von einem Board ermöglicht eine bessere Realisierung interaktiver Objekte im Sinne des Physical Computing. PicoCricket [PIC08] ermöglicht die Nutzung vorgefertigter Sensoren und Aktoren und einer „Drag&Drop“-Programmierungsumgebung bereits im Grundschulbereich. Der Ansatz von PicoCricket geht damit wesentlich über das der Robotik-Systeme hinaus: Es sollen Erfinderbaukästen bereitgestellt werden, die Technologie und Kunst vereinen und somit die Kreativität der Benutzer gezielt ansprechen. Dies entspricht genau der Idee des Physical Computing. Es können Lampen, Motoren und Sensoren ergänzt und somit verschiedenartigste interaktive Projekte realisiert werden, die den persönlichen Interessen der Schüler entsprechen. Die Schüler müssen nur noch die einzelnen Bauteile zusammenstecken und dann in der simpel gehaltenen Umgebung programmieren, um ihre ersten eigenen interaktive Objekte zu erstellen. Von positiven Erfahrungen bei der Erstellung interaktiver Objekte mit PicoCricket, insbesondere zur Interessengewinnung und kreativem handlungsorientierten Lernen, berichten u. a. [Gu10, MKL10, Ru08].

Erste Erfahrungen im Informatikunterricht beschreiben Robotik mit PicoCrickets als “spannenden, interessanten und kreativen Zugang zur Informatik” [RR11].¹

3.3 Interaktive Objekte mit Mikrocontrollerboards

Die vielfältigsten Möglichkeiten ergeben sich aus der Nutzung von Mikrocontrollerboards, wie z. B. Arduino [Ba11], welche auch den Ursprung des Physical Computing darstellen: Eine sehr große Anzahl und Bandbreite an Sensoren und Aktoren kann genutzt werden. Baumann [Bau11] berichtet über den Einsatz von Arduino im Informatikunterricht, in welchem Schüler u. a. eine Ampelsteuerung nachbauen. Verwendet werden ein Arduino-Mikrocontroller und handelsübliche Bauteile. Programmiert wird in der zugehörigen Arduino-Programmierungsumgebung in einer an C/C++ angelehnten Programmiersprache. Als nachteilig ergibt sich die Notwendigkeit der Arbeit mit Steckbrettern und losen Litzen, so dass die entstehenden Projekte mitunter schnell unübersichtlich werden oder ein nur sehr abstraktes Modell des zu realisierenden Gegenstandes darstellen (Abb. 1). Hinzu kommt, dass die feinen Drähte leicht brechen oder aus den Buchsen rutschen können, was zur Folge hat, dass häufig Kontakte geprüft werden müssen. Speziell bei komplexeren Aufbauten kann dies einige Zeit in Anspruch nehmen, von der informatischen Ideen ablenken und schnell frustrierend wirken. Größere Experimente verlangen zudem einiges elektrotechnisches Knowhow.

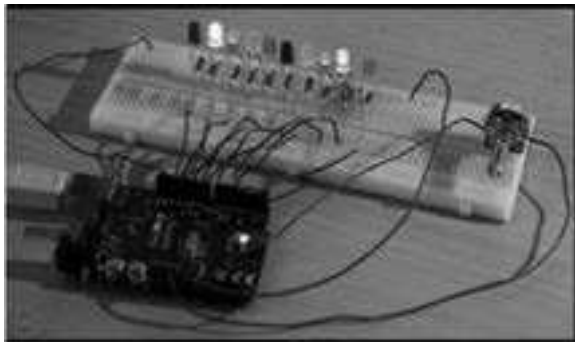


Abbildung 1: Arbeit mit Mikrocontrollerboard, Steckbrett und Litzen

4 Physical Computing als *informatisches Töpfern*

Physical Computing ermöglicht, dass im Informatikunterricht konkrete, physische Produkte der realen Welt entstehen, die der Vorstellungskraft der Lernenden entstammen. Konstruktionistisches Lernen wird damit auf eine haptisch erfahrbare Ebene gehoben und somit konkretisiert. Vahrenhold [Va12] ist der Ansicht, dass dem Informatikunterricht ein „Going to Paris“-Effekt fehlt. Während Schülern und Eltern als Ziel des Französischunterrichts die Möglichkeit, sich auf einer Parisreise verständigen zu können,

¹ Der Vertrieb von PicoCrickets wurde allerdings 2012 eingestellt.

offensichtlich ist, fehlt eine solche Zielvorstellung für den Informatikunterricht, jenseits eines besseren Umgangs mit dem Computer. Unter dem Aspekt des *informatischen Töpfers* bezeichnen wir es als ein Ergebnis des Informatikunterrichts, analog zum Herstellen einer Vase im Töpferkurs, ein digitales, interaktives Artefakt herzustellen, welches im konstruktionistischen Sinn untersucht, herumgezeigt und bewundert werden kann. Aus diesem Verständnis heraus erhält Informatikunterricht einen persönlichen Bezug.

4.1 Kompetenzzuwachs beim Physical Computing

Die von der GI empfohlenen Bildungsstandards Informatik für die Sekundarstufe I [GI08] beinhalten Formulierungen von informatikbezogenen Kompetenzen, die jeder Schüler im Laufe seiner schulischen Informatikausbildung der Sekundarstufe I erwerben soll. Die folgende Übersicht zeigt exemplarisch, welche Kompetenzen in ausgewählten Inhalts- und Prozessbereichen besonders gut beim Physical Computing erworben werden können.

Inhaltsbereiche:

- 1) **Information und Daten:** Der Zusammenhang zwischen Daten und Information wird implizit deutlich. Die Schüler wandeln Daten, die sie von Sensoren erhalten, in sinnvolle Informationen um. Sie interpretieren analoge und digitale Eingabewerte. Bei der Verarbeitung von Daten verwenden die Schüler arithmetische und logische Operationen. Sie sind außerdem in der Lage, Informationen zu Daten zu kodieren, welche an die Aktoren ausgegeben werden können, so dass das gewünschte Verhalten eintritt.
- 2) **Algorithmen:** Beim Entwurf ihrer interaktiven Objekte und Installationen formulieren die Schüler selbstständig Algorithmen, um zuvor gestellte Ziele zu verwirklichen. Die entworfenen Algorithmen werden unter Nutzung der durch Sensoren gewonnenen Daten und der durch Aktoren realisierten Aktionen von den Schülern getestet und gegebenenfalls modifiziert.
- 3) **Informatiksysteme:** Ein Verständnis von (eingebetteten) interaktiven Informatiksystemen wird durch die Arbeit mit dem Physical Computing-Baukasten erworben. Die Schüler erstellen mit diesem Werkzeug selbstständig neue Informatiksysteme.

Prozessbereiche:

- 1) **Modellieren und Implementieren:** Die Schüler implementieren in ihrer Ganzheit zu betrachtende Informatiksysteme bestehend aus Hard- und Software und beurteilen ihre Entwürfe kritisch. Sie nehmen bewusst Veränderungen an ihren Implementationen und an der Hardware vor und beobachten die dabei entstehenden Auswirkungen.
- 2) **Begründen und Bewerten:** Bei der Konzeption ihrer Projekte nutzen die Schüler ihr informatisches Wissen, um komplexe Problemstellungen zu entwerfen. Sie stellen Vermutungen über geeignete Lösungswege an und treffen begründete Entscheidungen zur Modellierung informatischer Sachverhalte.

5 Physical Computing mit „My Interactive Garden“

Der Einsatz von Physical Computing im Informatikunterricht wirft Fragen hinsichtlich einzusetzender Hardware, Programmierumgebung und des Unterrichtskontexts auf. My Interactive Garden (MyIG) [PR12] stellt eine prototypische Gestaltungs-, Programmier- und Lernumgebung dar, mit welcher Potential und Grenzen des Physical Computing im Informatikunterricht untersucht werden sollen. Im Folgenden werden wesentliche Designentscheidungen sowie erste Unterrichtserfahrungen diskutiert.

MyIG versucht eine Brücke zwischen den oben charakterisierten Mikrocontrollerboards und vorgefertigten Sensoren und Aktoren zu schlagen. Hierfür ist es notwendig, die Hardware entsprechend zu vereinfachen, welche unter Berücksichtigung der folgenden Prinzipien realisiert wurde:

- 1) **Einfache Handhabung:** Die Schüler erhalten vorgefertigte Sensoren und Aktoren mit einfachen Steckverbindungen, die kein Hantieren mit losen Litzen oder gar Löten erfordern, sondern sofortiges Experimentieren ermöglichen.
- 2) **Flexibilität und Erweiterbarkeit:** Die Komponenten sollten eine flexible Gestaltung interaktiver Objekte ermöglichen. Bei Bedarf soll eine unkomplizierte Erweiterung um zusätzliche Komponenten möglich sein.
- 3) **Black-/Whiteboxen:** Komponenten, die für das Verständnis des Sensors oder Aktors irrelevant sind, müssen nicht sichtbar sein, sollen aber bei Bedarf sichtbar werden können.
- 4) **Betonung informatischer Grundsätze:** Grundlegende informatische Prinzipien sollen deutlich werden, z. B. das EVA-Prinzip durch optische Trennung der Anschlüsse für Ein- und Ausgabekomponenten (Sensoren/Aktoren).

MyIG stellt auf der Basis von Arduino einen Baukasten zur Verfügung, der neben vorgefertigten Sensoren und Aktoren auch ein Shield enthält, welches ein einfaches Anstecken der Komponenten ermöglicht (Abb. 2). Zur Programmierung können damit verschiedene Programmiersprachen genutzt werden, wie z. B. eine Variante von Scratch (S4A) (Abb. 3) oder Arduino-C. Hinzu kommen beliebige zusätzliche Materialien, z. B. ein Bastelset.

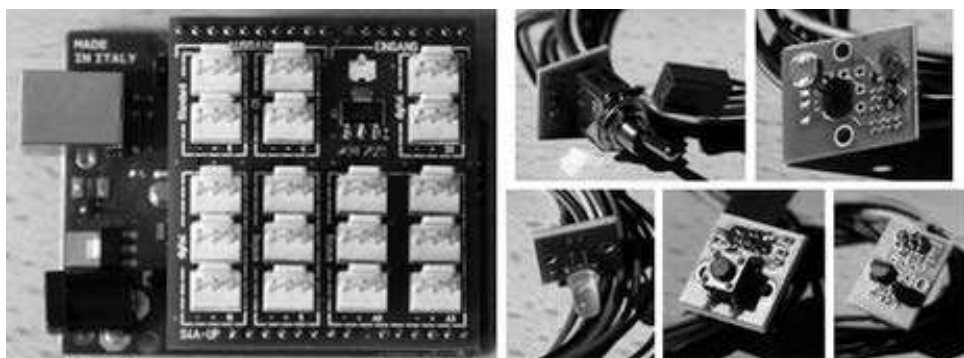


Abbildung 2: Arduino mit aufgestecktem MyIG-Shield und vorgefertigte Sensoren und Aktoren

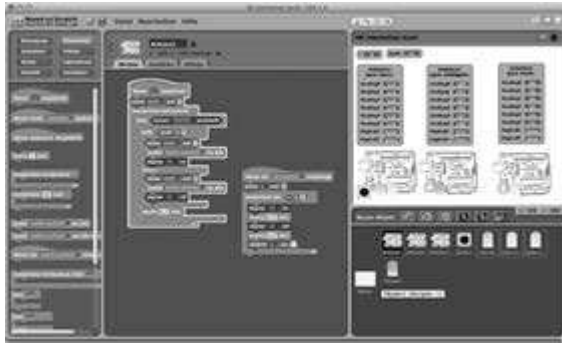


Abbildung 3: Programmierumgebung S4A

Ziel des Lernens mit MyIG ist die kollaborative Erstellung einer Ausstellung interaktiver Objekte, wie sie in einem futuristischen interaktiven Garten zu finden sein können – ein Rahmen, der vielfältige Projekte ermöglicht und die Kreativität der Schüler anregen soll. Folgende Strategien werden in Anlehnung an [Ru08] empfohlen:

- Konzentration auf Themen statt Aufgaben
- Kombination von Kunst und informatischem Modellieren
- Ermutigung zum Erzählen von Geschichten
- Organisation von Ausstellungen statt Wettbewerben

Erprobung

In einem ersten Schulversuch wurde der Einsatz des MyIG-Baukastens in einer neunten Klasse erprobt. Möglichkeiten und Grenzen von Physical Computing wurden hinsichtlich der folgenden Aspekte untersucht: (1) Wahrnehmung eingebetteter Informatiksysteme, (2) Akzeptanz „informatischen Töpfers“, (3) Ausgewogenheit informatischer und ergänzender Basteltätigkeiten, (4) Mehrwert des Physical Computing.



Abbildung 4: Beispielprojekte „Interaktive Blume“ und „Lagerfeuer“

Zur Motivation und Darstellung der Möglichkeiten von Physical Computing, wurden zu Beginn des Schulversuchs zwei Beispielprojekte vorgestellt (Abb. 4), gefolgt von einer Experimentierphase, in der die Schüler eigenständig alle Komponenten des Baukastens ausprobieren konnten. Im Anschluss daran erstellten die Schüler einen Projektplan für ihre ersten interaktiven Objekte zum Thema „My Interactive Garden“. Während der Arbeitsphase arbeiten sie nach ihren selbst erstellten Zeitplänen. In regelmäßigen Evaluationsphasen wurden Fortschritte reflektiert, mögliche Probleme und Lösungs-

ansätze diskutiert und fachliche Erkenntnisse genutzt, um den Zeitplan an die sich neu ergebenden Umstände anzupassen.

(1) Zu Beginn der Erprobung nannten die Schüler auf die Frage, wo Informatik überall eine Rolle spielt, überwiegend Beispiele, die sich unter dem Begriffen „Standardsoftware“, „Computerspiele“ und „html“ subsumieren lassen. Zwei Schüler nannten außerdem LEGO-Mindstorms-Roboter und Smartphones. Eingebettete Systeme des Alltags hingegen sind ihnen als typische Produkte der Informatik nicht eingefallen. Im Anschluss an die Experimentierphase war schon ein erster Wandel dieser Informatikwahrnehmung zu beobachten. Während sich die Schüler darüber austauschten, was sie über die Funktionsweise der Sensoren und Aktoren herausgefunden haben, waren häufig Vergleiche wie: „Das ist wie beim Auto, da piept es auch unterschiedlich schnell, je nachdem wie weit ein Hindernis entfernt ist.“ zu vernehmen.

(2) Mittels eines Fragebogens wurden die teilnehmenden Schüler vorab zu ihrer Wahrnehmung des Informatikunterrichts befragt. Dabei stellte sich heraus, dass sie einstimmig nicht das Gefühl hatten, im Informatikunterricht gemeinsam an etwas Größerem arbeiten zu können. Gleichzeitig antworteten einige Schüler positiv auf die Frage, ob sie Produkte des Informatikunterrichts bereits einmal anderen vorgeführt haben. Dies erlaubt die Schlussfolgerung, dass durchaus ein Interesse daran besteht, im Informatikunterricht vorzeigbare Produkte zu fertigen. Im Projektverlauf erschuf nun jeder Schüler sein eigenes interaktives Objekt, welches dann am Ende in die gemeinsame Ausstellung des interaktiven Gartens integriert wurde.

(3) Es besteht die Gefahr, dass beim Physical Computing das Basteln in den Vordergrund und die informatischen Inhalte in den Hintergrund rücken. Unsere Vermutung ist jedoch, dass umfangreiches Basteln auch umfangreiche Programmierung implizieren kann, da das Gesamtwerk an Komplexität gewinnt. Interessanterweise ist in der Lerngruppe den meisten Schülern die Programmierung wichtiger als das Basteln. Die Schüler versuchen, in ihren Projekten so viele Sensoren und Aktoren unterzubringen wie möglich und umfangreiche Programme zu erstellen. Die äußere Hülle des interaktiven Objekts erscheint derzeit noch eher nebensächlich zu sein, die Schüler sind eher von der Technik, also den Sensoren und Aktoren, fasziniert als vom Bastelmaterial. Gleichzeitig konnten wir aber auch feststellen, dass sie sich darauf freuen, ihren Erfindungen die passende Form zu geben. Sie haben zahlreiche kreative Ideen - nur ist dieser Aspekt nicht prioritär in der Reihenfolge der auszuführenden Arbeitsschritte in der Projektplanung.

(4) Im Rahmen der Physical-Computing-Projekte konnten relevante Fragestellungen an konkreten Problemen untersucht werden, die sich sonst im Informatikunterricht oft nur abstrakt bearbeiten lassen: Was ist der Unterschied zwischen analog und digital? Welche Komponenten gehören an die Eingänge, welche an die Ausgänge? Größtenteils konnten sich die Schüler ihre Fragen selbst oder gegenseitig beantworten. Interessant war zu beobachten, dass sie sich nicht lange damit aufhielten, die einzelnen Sensoren und Aktoren nacheinander zu testen, sondern sie schnell in umfangreichere Gebilde eingebunden haben. So entstand beispielsweise bei einem Schüler schon in der Experimentierphase eine komplexe Ampelschaltung mit durch Helligkeitssensor

gesteuerte Tag- und Nachtmodi, Bedarfsschaltung und Piepton für blinde Fußgänger. Ein anderer Schüler entwarf ein komplexes Gebilde aus Taster, Kippschalter, blinkenden Lampen, Summer und Servo. Alle Schüler sind im Laufe der Experimentierphase mit dem Problem der Mehrfachausführung konfrontiert gewesen, haben die „forever“-Schleife und bedingte Schleifen kennengelernt und sich mit Alternativen befasst. Arithmetische Operationen, Botschaften und Variablen wurden ebenfalls bereits verwendet – das alles geschah automatisch, intuitiv und interessengesteuert und ohne aktives Eingreifen durch die Lehrperson. Auffällig war auch, dass die Schüler offensichtlich den experimentellen Zugang bevorzugten. Sie haben erst dann nach Hilfe gefragt, wenn sie allein tatsächlich nicht mehr weiter kamen. Dies war jedoch nur selten der Fall.

6 Diskussion und Ausblick

Informatikunterricht steht seit jeher vor der Herausforderung, zeitstabile Ideen und Konzepte zu vermitteln, diese aber an aktuellen Themen und unter Zuhilfenahme moderner Werkzeuge zu motivieren. Mit Physical Computing eröffnet sich für den Informatikunterricht ein Anwendungsgebiet, welches so wie Mindstorms die Ideen von Logo erfolgreich in die reale Welt übertragen hat, das Potential der kreativen Gestaltung von Scratch in Produkte der realen Welt transferiert. Betrachtet man den Motivationswert eines selbstkreierten Produkts, wie z. B. durch Töpfern, wird schnell deutlich, dass dieses sich auch für den Informatikunterricht nutzen lässt. Die Verbindung von Informatik mit Kunst und Design hat das Potential, auch weniger computeraffine Schüler anzusprechen.

Unsere Erfahrungen zeigen, dass Schüler Spaß haben am Physical Computing. Dies äußerten sie verbal, zeigten es aber auch beispielsweise dadurch, dass sie am Ende des Unterrichts nicht nach Hause gehen wollten und stattdessen noch weiter mit den Bauteilen experimentierten. Die Schüler wünschten sich außerdem Baukästen für zu Hause, da sie sonst nur einmal pro Woche Gelegenheit haben, an ihren interaktiven Objekten zu arbeiten.

Während es sich bei MyIG um einen lokal hergestellten Prototypen handelt, befinden sich zur Zeit verschiedene vergleichbare Systeme in der Entwicklung, die in absehbarer Zeit für den Einsatz im Unterricht erhältlich sein sollen (z. B. [TK13], [Go10]). Die ersten Erfahrungen mit MyIG zeigen, dass Physical Computing im Informatikunterricht einen spannenden und vielversprechenden Ansatz darstellt. Wir gehen davon aus, dass in wenigen Jahren die Erstellung interaktiver Objekte einen festen Platz in der informatischen Bildung eingenommen haben wird, ähnlich wie Mindstorms-Robotik heute. Dieser Beitrag kann für den Einsatz der Systeme eine erste Orientierung geben.

Literaturverzeichnis

- [Ba11] Banzi, M.: Getting Started with Arduino. O'Reilly Media / Make, Sebastopol, CA, 2011.
- [Bau11] Baumann, R.: Eingebettete Systeme verstehen. In: LOG IN, H. 171, LOG IN Verlag GmbH, 2011; S. 33-45.

- [GI08] Gesellschaft für Informatik (GI) e.V.: *Grundsätze und Standards für die Informatik in der Schule*. In: LOG IN, H. 150/151, LOG IN Verlag GmbH, 2008; Beilage.
- [Go10] Gomba, D.: Visual Programming Arduino: modkit and the others. 2010. <http://arduino.cc/blog/2010/10/05/visual-programming-arduino-modkit-and-the-others/> (07.03.2013)
- [Gu10] Guzdial, M.: Dancing and singing humans, even more than robots. 2010. <http://computinged.wordpress.com/2010/12/31/> (07.03.2013).
- [Kn11] Knobelsdorf, M.: Biographische Lern- und Bildungsprozesse im Handlungskontext der Computernutzung. Dissertation, Berlin, 2011.
- [Ma12] Petre, M.: Computing Is Not a Spectator Sport: Rethinking How We Introduce Our Discipline to Students. In: GI (Hrsg.): Praxisband zur GI-Fachtagung „HDI 2012 – Informatik für eine nachhaltige Zukunft“. Universitätsverlag Potsdam, 2012. S. 155-159.
- [MKL10] Marcu, G.; Kaufman, S. J.; et al: Design and evaluation of a computer science and engineering course for middle school girls. Proc. SIGCSE 2010, Milwaukee, ACM, 2010.
- [PA13] Pelz, L.; Arnhold, W.: Die Waschmaschine - Embedded Computing im Alltag. 2013. Workshop auf der 12. Tagung der GI-Fachgruppe „Informatik-Bildung in Berlin/BRB“.
- [PH91] Papert, S., Harel, I. Situating Constructionism. In: Papert, S.; Harel, I. (Hrsg.), Constructionism. Ablex Publishing Corporation. Norwood, 1991.
- [PIC08] The Playful Invention Company (PICO): PicoCricket – Invention kit that integrates art and technology. 2008. <http://www.picocricket.com/whatisit.html> (06.03.2013)
- [PIC10] The Playful Invention Company (PICO): Picoboard – Sensor Board that works with MIT's Scratch. 2010. <http://www.picocricket.com/picoboard.html> (06.03.2013)
- [PR12] Przybylla, M.; Romeike, R.: My Interactive Garden – A Constructionist Approach to Creative Learning with Interactive Installations in Computing Education. In: Kynigos, C.; Clayson, J. E.; Yiannoutsou, N.: Constructionism: Theory, Practice and Impact. Proceedings of Constructionism 2012. Athen. S. 395-404.
- [Re07] Resnick, M.: Sowing the Seeds for a More Creative Society. Proc. Learning & Leading with Technology, International Society for Technology in Education (ISTE), 2007.
- [RR11] Romeike, R.; Reichert, D.: PicoCrickets als Zugang zur Informatik in der Grundschule. In: Thomas, M. (Hrsg.): Informatik in Bildung und Beruf (14. GI Fachtagung Informatik und Schule - INFOS 2011). Köllen, Bonn 2011, S. 177-186.
- [RS10] Richards, M.; Smith, N.: Teaching UbiComp with Sense. In: Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries (NordiCHI '10), ACM, New York, 2010; S. 765-768.
- [Ru08] Rusk, N., Resnick, M., Berg, R., & Pezalla-Granlund, M.: New Pathways into Robotics: Strategies for Broadening Participation, In: Journal of Science Education and Technology, 17 (1), 2008; S. 59-69.
- [St09] Strecker, K.: Informatik für Alle – Wie viel Programmierung braucht der Mensch? Dissertation, Göttingen, 2009.
- [St11] Strecker, K.: Wie viel Programmierkompetenz braucht der Mensch? In: LOG IN, H. 169/170, LOG IN Verlag GmbH, 2011; S. 40-47.
- [TK13] TinkerKit: TinkerKit! 2013. <http://www.tinkerkit.com/> (07.03.2013)
- [Va12] Vahrenhold, Jan: On the Importance of Being Earnest: Challenges in Computer Science Education. Proceedings of WiPSCe 2012, ACM, New York, 2012.
- [Wa05] Wagner, O.: LEGO Roboter im Informatikunterricht - Eine Untersuchung zum Einsatz des LEGO-Mindstorms-Systems zur Steigerung des Kooperationsvermögens im Informatikunterricht eines Grundkurses (12. Jahrgang, 2. Lernjahr) der Otto-Nagel-Oberschule (Gymnasium). Schriftliche Prüfungsarbeit im Rahmen der zweiten Staatsprüfung für das Amt des Studienrates, Berlin, 2005.
- [We08] Weber, M.: Vermittlung von informatischen Grundkonzepten der Realschulbildung anhand einer robotergesteuerten Lagerverwaltung. Universität Erlangen-Nürnberg, Erlangen, 2008.