A Brief Survey of Distributed Computational Toys

Eric Schweikardt
Computational Design Lab
Carnegie Mellon University
tza@cmu.edu

Mark D Gross
Computational Design Lab
Carnegie Mellon University
mdgross@cmu.edu

Abstract

Distributed Computational Toys are physical artifacts that function based on the coordination of more than one computing device. Often, these toys take the form of a microcontroller network embedded in a children's construction kit. We present a survey of Distributed Computational Toys. Although most of the toys we surveyed were built in the last five years and exist only as research projects, they build on the rich history of Constructivism, Constructionism and Kinesthetic Learning. Projects are tagged according to their structure, status, and intended functionality: Construction Kits, Physical Programming, and Cellular Automata.

1. Introduction

A long history of research points to the conclusion that children learn effectively by designing and building things. Whether it's a LEGO spaceship, a couch-cushion fort or a piece of software, the acts of building an artifact and receiving feedback on it are valuable in many ways.

Early work by Piaget [1] argued that information is not transferred directly from an educator to a child, but that concepts must actually be constructed in each child's mind. This idea, known as *Constructivism*, is not to say that each student must individually discover that the earth rotates around the sun, but that she must go through the process of learning about gravity and planetary orbits before she can truly know and understand the earth's rotation. In this view, a teacher's role is not solely to disseminate factual information to students, but to create an environment that facilitates a child's internal constructs.

Whereas constructivism holds that any idea or concept needs to be built to be understood, Seymour Papert's term *Constructionism* describes how building a real, physical artifact can be an especially effective learning process [2]. As with constructivist knowledge, by building a toy spaceship or a virtual city, a child is exposed to all of the parts and processes that the spaceship or city is based on. In contrast to constructivism, however, the final *artifact* (the spaceship as opposed to the *concept* of heliocentricity) exists outside the mind and can be viewed, critiqued, and improved by others.

Constructivist and constructionist ideas both apply in an educational setting, but it has also been shown that significant learning and development can occur during play. Vygotsky [3] argues that play provides a key transition between the reactive actions of a baby and the the abstract thinking of an adult. Because of its inherent freedom and self-direction, play provides an ideal situation in which to construct both artifacts and concepts.

Constructivism, constructionism and the value of play lead naturally to the creation of building toys. Froebel's nineteenth-century building block sets are often cited by designers as being hugely influential [4], and have been succeeded by a large variety of commercially available construction toys including Erector Sets, Meccano, Fischertechnik, Tinker Toys and arguably the most successful, LEGO.

These toys serve as tools to scaffold the process of children learning to design. Many skills are embedded in the act of design: synthesis, analysis, reflection and problem characterization are but a few. Based on our own background, we believe that the act of design is the most valuable educational process available, and that both learning *to* design and learning *by* designing are fundamental lifelong pursuits. This is also a familiar theme in the growing literature on constructivist learning. [5, 6]

Recently, toys for design have evolved beyond passive building sets to include computation and robotics. The LEGO Mindstorms kit, which added a small computer and several sensors and motors to the standard selection of plastic bricks, allowed kids to program their constructions and build mobile robots.

Today, the idea of decentralized computing, in which work is done by several communicating computers instead of just one, is finding its way into toys. Advances in computation, networking, and wireless communication have caused significant attention to be focused on distributed computing, and popular awareness of the idea has increased, whether due to the internet, sensor networks, cellular phones, computer gaming, or smart home networks.

2. Distributed Computational Toys

Distributed parallel processes are a powerful way to understand and model certain processes in the physical world. In this view, individual actors make decisions that affect only their local neighbors, but the combined effect can be observed globally. Conway's Game of Life is perhaps the best known example, though Burks and von Neumann described this form of computation over fifty years ago. [7] More recently Wolfram's *A New Kind of Science* outlined an ambitious agenda for distributed cellular automata as a way of thinking about the world. [8]

To date the distributed computation approach has been used to model various "emergent" phenomena including animal flocking behavior, traffic patters and economic markets.

Resnick [9] argues that decentralized systems are an important tool for comprehending how the world works, and his StarLogo version of the Logo programming language encourages users to experiment with local rules that govern thousands of parallel turtles and the global patterns that emerge.

Most, if not all, explorations in distributed computation have been restricted to the computer screen. Users formulate rules for local behavior of pieces, then populate a simulated world with a pattern of pieces, and watch the screen as the rules play out in global effects.

We believe that the time is ripe for a new class of "distributed computational toys", (DCTs) which bring experimentation with distributed parallel processes into the physical world. We propose the following definition:

> *Distributed Computational Toys operate based on the inclusion of computation in more than one part of the toy.*

A building set in which every piece contains a microcontroller is an ideal example. With every part able to communicate with every other, an assembly of a distributed computational construction kit could modify its actions based on how the user had assembled the pieces.

2.1. What our survey excludes

In focusing our survey on toys that have some of the characteristics of distributed computation, we exclude several categories of electronic toys and toys that are computer-controlled. One such category is toys that operate with centralized computational control. This excludes most conventional electronic toys like Simon or Electronic Battleship. Notably, it also excludes building sets like LEGO Mindstorms that have one central microcontroller or computer that governs the toy's behavior.

We distiguish between computational toys and games played on a personal computer. Simulations and PC-based games, though hugely popular, are also not covered in this survey. Collaborative games played on users' cellular phones are based on concepts of distributed computation but we omit them due to their on-screen nature.

Several interesting projects operate within a framework containing static, passive components and a camera-enabled host application that uses machine-vision techniques to act based on the construction. Of note in this category is Quetzal [10], a tangible programming system in which young users program Mindstorms robots by assembling plastic components and then photographing the assembly.

Also excluded from our survey are toys that require a separate graphical user interface. For example, a Playstation console, which requires a connection to a host computer and display, is not a toy itself, but an interface for a toy. Such devices can be classified as Tangible Interfaces, an active research area in its own right. Tangible interfaces focus on the use of physical artifacts as tools to interface with the virtual world, instead of the tangible world of the artifact.

The ability to sense a user's physical construction and transmit the construction's geometry to a host computer is a specific application for a tangible interface. *Computational Building Blocks* are an example; each block can self-identify and the construction can determine the assembly. In this sense the blocks are a distributed system; however their only purpose is to self-describe to a host computer. [11]

2.2. Learning Objectives

All the toys surveyed here can instill curiousity about decentralized systems and data flow between modules. Even toys that are "black-boxed" to hide connectors and communication hardware can encourage users to experiment with their operation and capabilities.

These projects have more to offer than just a primer in distributed computing. Many of the toys attempt to scaffold learning in an orthogonal domain such as audio sequencing (Blockjam), responsibility (Tamagotchi) or robotics (roBlocks). Although we find it plausible that many of these toys offer significant learning benefits, it is not our intention to evaluate the effectiveness of any of these projects as individual educational tools.

We believe that the most important educational benefit of DCTs is to foster a general, logical way of looking at problems that Wing [12] has termed *computational thinking.* Construction kits allow children to separate complicated tasks into manageable modules, and the nature of building toys encourages builders to think about multiple layers of abstraction simultaneously. When building a robotic dog from a construction kit, the user is keeping in mind abstract ideas of the dog's behavior, spatial constraints on the placement of pieces, and engineering concerns such as power, balance, and data flow. This concurrent management of several streams of thought is an immensely valuable skill.

3. An Ontology

Figure 1 shows a simple ontology of the DCTs that we survey here. Each row names a construction toy or project. The first two columns indicate whether the toy is a commercial product or research project. The next two columns indicate whether the pieces of the kit are homogeneous or heterogeneous. Finally, the last three columns label the toys with one of three descriptors: *Construction Kits, Physical Programming* and *Cellular Automata*.

Construction Kits, many inspired by LEGO, focus on the physical morphology of an assembled collection of pieces.

Physical Programming toys are generally intended to instruct novice users in basic programming concepts, often by snapping blocks together instead of dragging boxes in a visual programming language or writing code in a text based language

Cellular Automata toys are the most transparent DCTs, encouraging users to think about how local rules can have global effects.

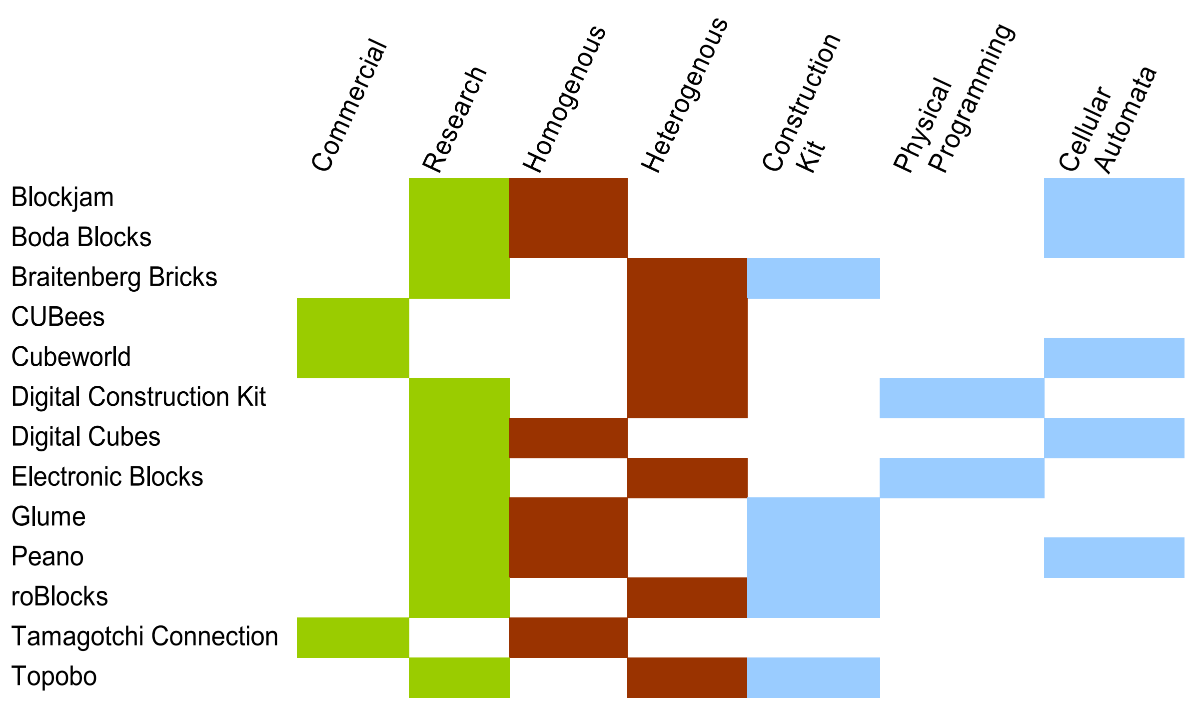

Figure 1. An Ontology of Distributed Computational Toys

4. Modular Robotics

DCTs have much in common with (and sometimes are) modular robotic systems. Both must address problems of control, communication and inter-module connectivity. Although modular robotics is still a nascent field, it has already received significant funding attention due to potential applications in military, surveillance and space applications. New technology for DCTs will likely 'trickle down' from research in modular robotics. We suspect that this is why we know of no mobile DCTs aside from Topobo and our own roBlocks project. Locomotion, coordination, and multi-robot communication are still active research areas and will continue to influence the design of DCTs.

A common distinction in modular robotic systems is between homogeneous systems, in which all modules are the same, and heterogeneous systems, which contain different kinds of pieces. In general, toys (and modular robots) made of heterogenous components are more flexible and powerful, as each module can contain a different function or actuator. A homogeneous robotic construction kit, on the other hand, would need to contain every type of sensor and actuator in every piece, which is costly and impractical. However, an advantage of homogeneous toys is that coordinating the behavior of the modules can be more straightforward, because they all operate on the same set of rules.

5. A Survey of Projects

Figure 2. Cube World

Cube World

One of only three commercial projects included here, Cube World is a system of plastic cubes with LCD displays, motion sensors and magnetic connectors [13]. The cubes are stacked in a plane, and low-resolution stick figures displayed on each cube interact according to how the cubes are arranged or shaken. Each cube's virtual character is pre-programmed with a personality (e.g., handyman or baseball player, and the set acts as a view into an apartment building where the various characters interact. Each display is limited, but as blocks are connected, characters jump between cubes and carry out amusing scripted skits. The toys are expensive, at around US$30 for a pack of two.

Figure 3. Blockjam

Blockjam

A highly polished music composition tool from Sony's Interaction Lab, Blockjam consists of a homogeneous set of square modules that connect in a 2D grid via hidden magnets. The modules include a button and a scroll wheel, and exhibit different behaviors and functions even though the hardware is identical. A user creates and modifies audio sequences by rearranging blocks, and can design the flow of a looped sequence by adding blocks or changing their values via the scroll wheel. Users can create multiple simultaneous sequences and watch an icon representing the beat travel through the assembly on small LCD screens embedded within the blocks. Although the system requires a host computer, its function is only to synthesize audio, a task which could easily be carried out by the blocks themselves.

CUBees

In direct contrast to Blockjam, CUBees are a commercial DCT that play back different combinations of pre-recorded audio depending on how they're stacked [14]. There are several distinct CUBees, each styled as a different animal, with a plastic flap that opens and closes when they "sing." When the user stacks the cubic cartoon animals, they play back one of three children's tunes, with the animal on the top of the pyramid singing lead and the others singing backup. At first glance, CUBees would seem to be merely a strange, contrived, instantiation of a DCT. We include them because children may gain insight into the concepts of synchronization and hierarchy by rearranging the cubes and listening to the output.

Digital Cubes

Digital Cubes are a unique set of four simple 8cm cubes with a matrix of cubic LEDs on their top face [15]. Depending on the adjacency of the blocks, legible or illegible text is scrolled across the connected displays. A project by artist Simon Schiessl, the Digital Cubes obscure their infrared connectors and microprocessors behind a simple exterior, but encourage users to rearrange the modules in hopes of coherent emergent behavior.

Figure 4. Tamagotchi Connection (Image provided by Bandai America Inc.)

Tamagotchi Connection

In contrast to the other toys surveyed here, the Tamagotchi Connection [16] operates by distributing computation among several users. Tamagotchi, which translates loosely to "lovable egg" is a small plastic device worn on a keychain. It displays the status of a virtual pet, which is improved by daily interaction with its user. The Tamagotchi Connection, added to the popular series of toys in 2004, interacts with other users' toys via infrared and may spontaneously marry or have a child with another connected pet. Although the educational benefits of a Tamagotchi are questionable, chidren may take away ideas of both responsibility and computer networking.

Peano

The Peano system [17] is comprised of one inch cubic modules containing a single LED, microcontroller and touch sensor. Each module has two connectors – on some blocks, the connectors lie on opposite faces, on others, the connectors are adjacent, to form a right angle when snapped together with neighboring cubes. As cubes can be connected to a maximum of two neighbors, any emerging structure is inherently one-dimensional, similar to a space-filling curve, first described by Giuseppe Peano in 1890. Peano cubes can be programmed to exhibit light patterns using two different desktop tools to either display a prewritten animation or execute a program based on their sequence, orientation, touch or network topology.

Figure 5. roBlocks – a heterogeneous DCT kit for experimenting with robotics.

roBlocks

Our own roBlocks [18] are 40mm ABS plastic cubes of several varieties with hermaphroditic magnetic connectors. The blocks are categorized into Sensor, Actuator, Logic and Operator blocks, and by snapping them together, users create both a physical robot and the distributed system which controls it. Snapping a light sensor block to a belt-drive block creates a simple Braitenberg Vehicle [19] that moves according to the amount of light in the environment. The system encourages more complex constructions with specialized blocks that block the signal flow, invert sensor values, or allow the user to manually set thresholds, for instance. An ad-hoc network of blocks is created when the modules are snapped together, but because each roBlock is locally controlled by its internal microcontroller, a roBlocks construction can illustrate emergent behavior of simple, connected modules. Advanced users have the option of reprogramming individual modules via a host computer interface.

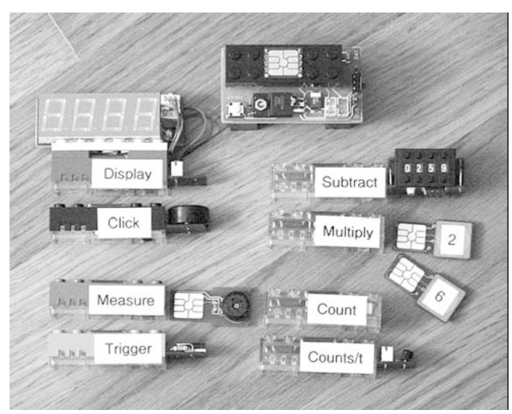

Figure 6. Digital Construction Set

Digital Construction Set

The Digital Construction Set [20] is a tangible programming environment built from modified Lego bricks. Each brick represents a function and contains a microcontroller and a card slot into which a user can insert different labeled cards representing parameters. Connectors on the top and bottom of bricks enable the user to create a linear stack representing a program. To combat the restriction of merely executing a linear sequence of instructions, the Digital Construction Set operates on the idea of *streams*, where control flow is explicitly from bottom to top, and each brick represents an operator such as "click," connected to a digital sensor, and "count" in the brick above to tally the number of clicks. A "display" brick can be added anywhere in the system to show the state at any point and aid in debugging. The simple syntax and limited functionality of the Digital Construction Set builds on Papert's concept of a *Microworld* [21], in which a constrained, simulated environment can assist in illuminating certain concepts (like data flow) to young users.

Smart Tiles and Boda Blocks

Cellular automata constitute a model of computation in which discrete, homogeneous modules change state on regular time steps according to the state of their neighbors. A common example is John Conway's *Game of Life* [22], in which regular squares on a grid turn black or white at each interval depending on the number of their neighbors that are black. *Boda Blocks* [23] and *Smart Tiles* [24] are construction kits created to encourage children to experiment with cellular automata. A set of networked blocks and connectors, Smart Tiles are small blocks containing PIC microprocessors that can be arranged on a grid and programmed with local cellular automata rules. Each tile indicates its state with a colored LED and can be switched on or off by striking it gently with a mallet. Boda Blocks are a three dimensional cellualar automata kit. Each Boda block is cast from urethane resin and includes an LED and a PIC to communicate with its neighbors and with the global block, which synchronizes the blocks. The design of the kit allows users to create 3D cellular automata and watch the emergent behavior.

Figure 7. Electronic Blocks

Electronic Blocks

Wyeth and Wyeth's Electronic Blocks [25] are a series of modified Duplo bricks that encourage young children (three to eight years old) to create simple robots by stacking. A limited ontology of sensor, logic and action blocks allows kids to create constructions that illuminate when pressed or drive forward in the presence of light. Although each block only contains simple circuitry to pass an electrical signal from top to bottom, a construction can be thought of as a basic distributed system, representing communication between each piece in the analog network.

Figure 8. Topobo

Topobo

A unique example of physical programming by demonstration, Topobo [26] consists of digitally controlled motors which can be attached to a variety of passive pieces using LEGO connectors. Users twist their construction to program behavior into each motor; the assembly then replays the user's actions. Users may gain insight into the concepts of relative motion and inverse kinematics through playing with the toy. It is interesting to note that coordination of multiple, distributed motors proved complex, and a Queen brick was added to the kit to provide centralized control. Additional computational "backpacks" can be added to an assembly to modify how messages are passed through the assembly.

Figure 9. Glume

Glume

In stark contrast to the rigidity of other computational toys is Glume [27], a prototype construction kit built from sophisticated electronics, soft silicone and hair gel. Glume modules are homogeneous and resemble malleable hand-sized versions of a piece of the game Jacks. Each module communicates with its neighbors capacitively and includes six RGB LEDs. Although intended as a tangible interface to a solid modeling program, the light output suggests that the Glume modules may be interesting to users as a standalone construction kit.

Braitenberg Bricks

There has been a rich history of computational toy development at MIT's Media Laboratory, and the LEGO-based toys developed there have spanned several iterations, from Programmable Bricks [28] to Crickets [28] and the Handyboard [29].

Although many of these projects rely on a single computer coordinating the machinations of a LEGO construction, the Braitenberg Creatures [30] built with Electronic Bricks are a notable exception. The Electronic bricks, regular LEGO bricks outfitted with sensors, actuators and wires for inter-brick communication, were assembled in several configurations based on the Vehicles proposed by Braitenberg [19]. Following Braitenberg's theme, the robots were named according to traits they seemed to exhibit as personality, perhaps suggesting to very young users that complicated human behavior may be explained as a large collection of simple rules.

6. The Future

The two largest current technological problems in modular robotics are power distribution and the mechanics of inter-module connectors. The scarcity of commercial DCTs reflects this, but we believe that more instances of this kind of toy will appear as ideas and engineering advances trickle down to the toy market.

As engineering solutions to the power and connectivity problem appear, we will see more computational construction kits both in the research laboratory and the market. The act of design is highly effective as an educational tool, and different kits scaffold different types of learning in domains such as electronics, computer programming, physics and structural engineering. The designed pieces of a particular construction kit can serve to enhance users' creativity. The noted late graphic designer Paul Rand speaks to the benefits of a constrained system as something "without which fruitful and creative work is extremely difficult." [31]

As microcontrollers and components decrease in cost, we will see more (and more realistic) physical simulations of real-world systems along the lines of Cube World. While StarLogo [9] encourages children to create screen-based simulations of parallel, distributed control, we imagine a set of tiny, mobile robots that could be programmed to work like a flock of birds or an ant colony.

Unfortunately, the trend in commercial toys may not be to increase the curiosity of its users. Although many of these toys are advanced computational objects, their design often serves to obscure the technology with which they are built. Hiding magnets and communication connectors reduces the diagnostic transparency of a system and encourages a user to think of it as a magic black box instead of a distributed calculating machine. Describing Cube World, one retailer advertised that "The idea is simple, just don't ask us how they get it to work." [32]

Ultimately, distributed computational toys provide more than just a context in which to learn about a target domain such as music, robotics or multi-family living arrangements. Their very nature affords insight into their operation – inter-module communication, power and data flow, networking and computation. These concepts are aspects of *computational thinking* and are valuable tools for learning, exploring and thinking about the world. We believe that the operation and constraints of future DCTs should be made diagnostically transparent in their design.

Acknowledgements

This research was supported in part by the National Science Foundation under Grant ITR-0326054. The views and findings contained in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

7. References

[1] J. Piaget, *The Child's Construction of Reality*. London: Routledge and Kegan Paul, 1955.

[2] S. Papert, "Situating Constructionism," in *Constructionism*, I. Harel and S. Papert, Eds. Norwood, NJ: Ablex Publishing Company, 1991, pp. 1-11.

[3] L. Vygotsky, "Play and its Role in the Mental Development of the Child," in *Play: Its role in development and evolution*, J. Bruner, A. Jolly, and K. Sylva, Eds. New York: Penguin Books, 1933.

[4] N. Brosterman, *Inventing Kindergarten*. New York: Abrams, 1997.

[5] M. Guzdial, J. Kolodner, C. Hmelo, H. Narayanan, D. Carlson, N. Rappin, R. Hübscher, J. Turns, and W. Newstetter, "Computer support for learning through complex problem solving," *Communications of the ACM*, vol. 39, pp. 43 - 45, 1996.

[6] Y. B. Kafai, C. C. Ching, and S. Marshall, "Children as designers of educational multimedia software," *Computers & Education*, vol. 29, pp. 117-126, 1997.

[7] A. W. Burks, "Von Neumann's Self-Reproducing Automata," in *Essays on Cellular Automata*, A. W. Burks, Ed. Champaign, IL: Univ. of Illinois Press, 1970, pp. 3-74.

[8] S. Wolfram, "A New Kind of Science," *http://www.wolframscience.com/*.

[9] M. Resnick, "Turtles, Termites, and Traffic Jams." Cambridge, MA: MIT Press, 1994.

[10] M. S. Horn and R. J. K. Jacob, "Tangible programming in the classroom: a practical approach " in *CHI '06 extended abstracts on Human factors in computing systems* Montreal, Quebec, Canada: ACM Press, 2006.

[11] D. Anderson, J. L. Frankel, J. Marks, A. Agarwala, P. Beardsley, J. Hodgins, D. Leigh, K. Ryall, E. Sullivan, and J. S. Yedidia, "Tangible Interaction + Graphical Interpretation: A New Approach to 3D Modeling " presented at International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH), 2000.

[12] J. M. Wing, "Computational Thinking," *Communications of the ACM*, vol. 49, pp. 33-35, 2006.

[13] Radica Games, "http://www.radicagames.com," Accessed 15 Oct 2006.

[14] Takara USA, "http://www.takara-usa.com/toys/cubee/," Accessed 15 Oct 2006.

[15] S. Schießl, "Digital Cubes," http://www.aec.at/en/archives/center_projekt_ausgabe.asp?iProjectID=10931, Ed., 2002.

[16] Bandai America Inc., "http://www.bandai.com/," Accessed 02 Oct 2006.

[17] K. B. Heaton, "Physical Pixels," in *Media Arts and Sciences*, vol. Master of Science in Media Arts and Sciences. Cambridge, MA: Massachusetts Institute of Technology, 2000.

[18] E. Schweikardt and M. D. Gross, "roBlocks: A Robotic Construction Kit for Mathematics and Science Education," in *International Conference on Multimodal Interaction*. Banff, Alberta, Canada, 2006.

[19] V. Braitenberg, *Vehicles: Experiments in Synthetic Psychology*. Cambridge, MA: MIT Press, 1984.

[20] T. S. McNerney, "From turtles to Tangible Programming Bricks: explorations in physical language design," *Personal and Ubiquitous Computing*, vol. 8, pp. 326-337, 2004.

[21] S. Papert, *Mindstorms: Children, Computers, and Powerful Ideas*, 2nd ed. New York: Basic Books, 1999.

[22] M. Gardner, "Mathematical Games: The fantastic combinations of John Conway's new solitaire game "life"," in *Scientific American*, 1970, pp. 120-123.

[23] M. Eisenberg, L. Buechley, and N. Elumeze, "Computation and Construction Kits: Toward the Next Generation of Tangible Building Media for Children," presented at Proceedings of Cognition and Exploratory Learning in the Digital Age (CELDA), Lisbon, Portugal, 2004.

[24] N. Elumeze and M. Eisenberg, "SmartTiles: Mobility and Wireless Programmability in Children's Construction and Crafts," presented at IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE 2005), Tokushima, Japan, 2005.

[25] P. Wyeth and G. Wyeth, "Electronic Blocks: Tangible Programming Elements for Preschoolers," presented at Eighth IFIP TC13 Conference on Human-Computer Interaction (Interact 2001), 2001.

[26] H. S. Raffle, A. Parkes, and H. Ishii, "Topobo: A Constructive Assembly System with Kinetic Memory," presented at Human Factors in Computing (CHI) '04, 2004.

[27] A. Parkes, V. LeClerc, and H. Ishii, "Glume: exploring materiality in a soft augmented modular modeling system," in *CHI '06 extended abstracts on Human factors in computing systems*. Montreal, Quebec, Canada: ACM Press, 2006, pp. 1211-1216.

[28] M. Resnick, F. Martin, R. Berg, R. Borovoy, V. Colella, K. Kramer, and B. Silverman, "Digital manipulatives: new toys to think with," presented at SIGCHI conference on Human factors in computing systems, Los Angeles, CA, 1998.

[29] F. Martin, *Robotic Explorations: A Hands-On Introduction to Engineering*. Upper Saddle River, New Jersey: Prentice Hall, 2001.

[30] D. W. Hogg, F. Martin, and M. Resnick, "Braitenberg Creatures," Epistemology and Learning Group, MIT Media Laboratory June 5,1991 1991.

[31] P. Rand, "Design and the Play Instinct," in *Education of Vision, Vision & Value Series*, G. Kepes, Ed. New York: George Braziller, 1965, pp. 159.

[32] Kohl's, "http://www.kohls.com/," Accessed 14 Oct 2006.