

A Binary Dance Workshop

Andrea Hawksley
Communications Design Group
SAP Labs, San Francisco, CA, USA
hawksley@gmail.com

Abstract

Binary operations and arithmetic are fundamental to how computers work, but are not always taught to students. This paper describes a mathematical dance workshop that can be used to visually display and kinesthetically teach the basics of binary including both basic binary logic as well as binary counting and addition.

1 Introduction

Computers are a pervasive part of our world today. Understanding computer fundamentals, such as binary, has thus become as important as any other basic knowledge taught in school. However, these fundamentals are not necessarily even taught to students. Indeed, mathematics in bases other than 10 are entirely omitted from the Common Core State Standards [5].

In this paper, I describe an interactive workshop wherein the participants first learn and dance basic binary logic operations such as AND, OR, and XOR, before progressing to binary counting and addition. The workshop attempts to teach binary in a way that is particularly representative of a computer. Specifically, the state of the binary dance is stored by a group of people that are functioning as “registers”, and it is possible to add more bits than there are registers causing “overflow” behavior.

2 Previous Work

Mathematical dance is an area that has been fairly well explored. Watson identifies spatial exploration, rhythm, structure, and symbolisation as characteristics in common between dance and mathematics [13]. Schaffer and Stern have choreographed and danced a number of pieces that incorporate mathematical concepts, as well as describing related workshops [11]. Richeson has described the mathematics of the braids formed by maypole dancing [9].

The use of mathematical dances as pedagogical tools is also not new. Ward-Penny gives an overview of various cross-curricular teaching methods for teaching mathematics through dance and movement [12]. Schaffer, Stern, and Kim have published an entire book of mathematical dance exercises in 2001 which includes a bibliography of mathematical dance resources [6]. Hawksley, Conway, and Naylor and Hart have also described movement-based workshops covering mathematical concepts [4, 7, 1, 3].

There are several examples of computer-related concepts being portrayed with dance. Algo-rhythmics has made a series of folk dance videos illustrating sorting algorithms [10]. Specific to binary dance, Hart has made a video of a ‘binary hand dance’, where binary numbers are counted using the fingers to music [2]. Regan describes having students stand up or sit down to represent ones and zeros as part of a larger non-movement related class on binary numbers [8].

3 Workshop Description

The binary dance workshop consists of two major parts. In the first part, the participants work in small groups to become familiar with basic binary logic operations such as NOT and OR, with a special focus on XOR. In the second part, the participants work together in a larger group to perform more complicated binary arithmetic, first simply counting up binary numbers, then adding arbitrary binary numbers as indicated by a drummed ‘instruction tape’.

3.1 Binary Logic Operations

Participants break into groups of 3 for the first part of the exercise. In each group, one person performs the function of a register and holds the current state or value that has been calculated. At the end of an operation, if this person has a partner, the calculated value is 1, if the register does not have a partner, the calculated value is 0. The other people can be thought of as each representing bits.

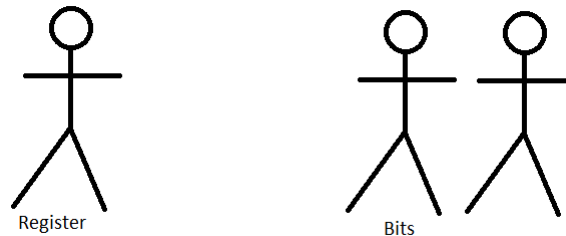


Figure 1: *One person represents a register, the other people represent bits.*

Each register may be pre-set with a state of either 1, where the register has a partner bit, or 0, where the register is alone.

The groups begin by learning about the NOT operation. After the register is set, the leader of the workshop announces a ‘calculate’ step. If a register starts out without a partner, it acquires a partner at this step. Otherwise, it drops its partner. The NOT operation effectively flips its state.

Simple rules for partner changing can be used to define all of the basic bitwise operations. Most of these operations involve 2 bits. One of them will be set on the register already, the other one will be indicated as a 1 or a 0 on the announced ‘calculate’ step based on whether or not a drum is hit on the calculate step.

An AND operation returns 1 if and only if both inputs are 1. This can be danced using the following rules:

1. If the register does not have a partner, he or she always refuses a partner
2. If the register does have a partner, he or she always lets the partner go
3. If a bit has been indicated by a drum beat, the dancer representing that bit takes the register as a partner unless refused

An OR operation returns 1 if either input is 1. This is fairly simple to dance - the register always accepts new partners (up to a maximum of one partner) and never lets partners go.

The most important operation for our binary dance workshop is the XOR, or exclusive or, operation. The XOR operation is fundamental to boolean logic, but is often the least intuitive operation for people to understand. XOR is true only if the two inputs differ, its truth table can be seen in Table 1.

	0	1
0	0	1
1	1	0

Table 1: Truth table for XOR

XOR is important for this workshop because the XOR operation is the building block from which the binary adder in section 3.2 is built. Indeed, the XOR operation as practiced here can be thought of as single bit addition (mod 2). XOR can be represented with the following rules:

1. If the register does not have a partner, he or she always accepts a partner
2. If the register has a partner, and a bit is not indicated, the register keeps the partner
3. If the register has a partner, and a bit is indicated, the new bit ‘steals’ the registers partner, leaving the register without a partner

Figure 2 shows the interactions between dancers as they perform the XOR operation. Participants should spend some time practicing the XOR operation before continuing on to the next part of the workshop.

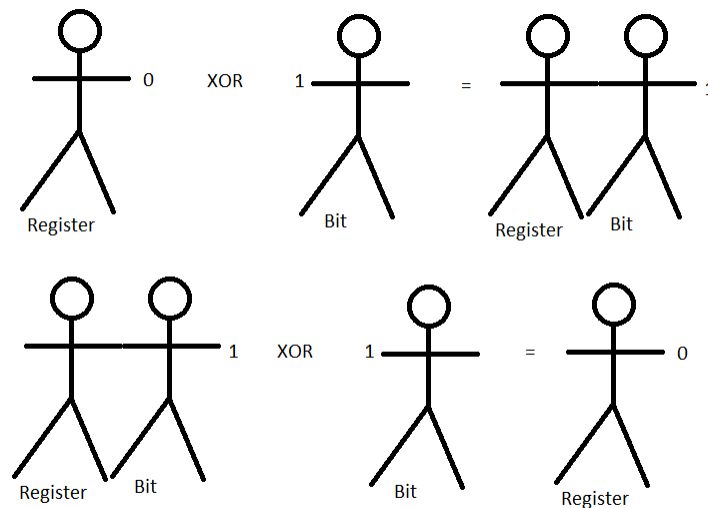


Figure 2: The dances resulting from $0 \text{ XOR } 1$ and $1 \text{ XOR } 1$ are shown above. In $0 \text{ XOR } 1$, the register gains a partner bit. In $1 \text{ XOR } 1$, the new bit steals the register’s partner bit, leaving the register empty.

3.2 Binary Addition

Once participants seem comfortable with the XOR operation, the workshop should continue on to the next phase. In this phase, we will use more dancers to create a larger register. In general, a register of 4 or 5 people seems to be a particularly manageable size. The right-most member of the register represents the 1 bit. The person next to them is the 2 bit, and so on. Participants may find it helpful to use nametags to label each member of the register with the bit that it represents.

3.2.1 Counting

The multi-register portion of the workshop begins by simply counting, or adding 1 repeatedly, in binary. At each ‘calculate’ step, a single bit is added to the right-most register. The addition operation is very similar to the XOR operation, except that when the new bit steals a register’s current partner, rather than completely running off, it instead simply moves the partner to the register one to the left, as in figure 3. If that register is full, the process repeats until either the next register is not full or there is no next register. If there is no next register, we face a not unusual computer science problem - an overflow problem. Participants may be surprised to discover that for a 4-bit register $1111(15) + 1 = 0$. Participants are effectively performing arithmetic mod 2^n , where n is the number of registers.

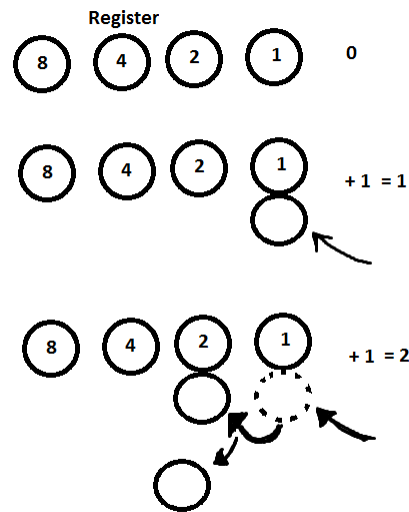


Figure 3: This diagram shows how dancers interact when adding 1 to an empty register twice. Here the members of the registers are labeled

As bits are added to the register, participants may find it an interesting exercise to count along “1, 10, 11, 100, 101...” to get a sense of how binary numbers grow. Participants should also try to calculate how large any given one of those numbers is. This can be done simply by adding the values of each member of the register with a partner as in figure 4.

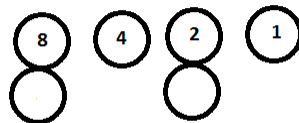


Figure 4: The ‘8’ and ‘2’ registers have partners, so the number being represented (1010) is equivalent to $8 + 2$ or 10 in binary.

3.2.2 Addition of Arbitrary Numbers

Finally, the workshop progresses to the addition of arbitrary numbers. By this point, participants should be comfortable with the basic dance movements involved from XOR and counting. We now introduce the

ability to add bits to registers other than the right-most register. For example, to add 4, a bit would be added to the third register from the right. Arbitrary binary numbers can be generated by flipping a coin.

There are two reasonable ways to add bits at arbitrary location. One option is to have the desired numbers pre-planned, and to have all of the bits join and add simultaneously on the ‘calculate’ step. This is nice because the numbers are being added in a single step.

The other option has the nice property that the music itself can be used to indicate the number. Dancers representing bits begin in a line in front of the right-most register. On each calculate step, a drumbeat indicates that a dancer (bit) should be added to the register. All the bit-dancers then shift to the left until they hit the left-most register. Thus, a 4-bit number can be added in four calculate steps.

With practice, a skilled dance group should be able to complete a single add and shift-left step per bar of music. They would then be able to truly dance the addition of any arbitrary sequence of 1’s (drumbeats) and 0’s given by an audience.

4 Conclusions

In this paper we describe a workshop that explicitly visualizes and adds physicality to the operations of binary arithmetic. Participants take turns representing registers and bits, and interact following simple rules to ‘dance out’ the solutions to binary logic, binary counting, and binary addition problems. This workshop focuses on building up to basic binary arithmetic. Future work in this area might include developing dances that solve other binary arithmetic problems, or teach other fundamental computer science concepts.

5 Acknowledgements

The author would like to thank all the members of the Octahedral Group for participating in and offering comments on this workshop. She would especially like to thank Sai and Karl Schaffer for their specific thoughts on workshop modifications.

References

- [1] John Conway. The power of mathematics. <http://thewe.net/math/conway.pdf> (as of Feb. 27, 2012).
- [2] Vi Hart. Binary hand dance, 2011. <http://www.youtube.com/watch?v=0CYZTg3jahU>.
- [3] Deanna Haunsberger and Stephen Kennedy, editors. *The Edge of the Universe: Celebrating 10 years of Math Horizons*. The Mathematical Association of America, Inc., 2006.
- [4] Andrea Hawksley. Exploring braids through dance: the “waves of tory” problem. In Douglas McKenna Robert Bosch and Reza Sarhangi, editors, *Proceedings of Bridges 2012: Mathematics, Music, Art, Architecture, Culture*, pages 613–618. Tessellations Publishing, 2012.
- [5] Common Core State Standards Initiative. *Common Core State Standards*. 2012. <http://www.corestandards.org/>.
- [6] Scott Kim Karl Schaffer, Erik Stern. *Math Dance with Dr. Schaffer and Mr. Stern*. Lulu, 2001.
- [7] Mike Naylor and Vi Hart. Human geometry workshop. In Reza Sarhangi and Carlo H. Séquin, editors, *Proceedings of Bridges 2011: Mathematics, Music, Art, Architecture, Culture*, pages 649–656. Tessellations Publishing, 2011.
- [8] Rick Regan. *How I Taught Third Graders Binary Numbers*. 2011. <http://www.exploringbinary.com/how-i-taught-third-graders-binary-numbers/>.

- [9] Dave Richeson. The maypole braid group, 2009. <http://divisbyzero.com/2009/05/04/the-maypole-braid-group/> (as of Feb. 27, 2012).
- [10] Algo rhythms team. Algo-rhythmics, 2014. <http://algo-rhythmics.ms.sapientia.ro/>.
- [11] Karl Schaffer and Erik Stern. *Dr. Schaffer and Mr. Stern Dance Ensemble*. 2014. <http://www.mathdance.org/>.
- [12] Robert Ward-Penny. *Cross-Curricular Teaching and Learning in the Secondary School: Mathematics*. Routledge, 2011.
- [13] Anne Watson. Dance and mathematics: engaging senses in learning. *Australian Senior Mathematics Journal*, 19, 2005.