BINARY BLASTER ACTIVITY

Essential Question/Summary

Students will learn how to convert values from binary to decimal by implementing a program on the Ozobot. They will use Evo specific blocks (lights and sounds) to complete this activity.

Information

- Instructors should know how to use the blocks from level 4.
- The program from the Timer Mini Lesson is incorporated into the Binary Blaster program. It may be useful to have students first download the Timer program (or complete the Timer program) and then they can build on that for this activity.
- Please see this guide for help on uploading OzoBlockly code onto Evo http://files.ozobot.com/stem-education/ozoblockly-evo-getting-started.pdf

Prerequisites

Students will need to be familiar with loops and logic statements. They should also understand the concept of a variable and know how exponents work. It is recommended to complete the Evo mini lessons before starting this lesson.

Grouping

Students should be working in pairs, but each student will write their own program. It is possible for students to work in groups of four, where each student will write their own program, but they can share the Ozobot by uploading their programs one at a time.

Materials

- OzoBlockly.com editor on a computer or tablet
- Ozobot Evo, one per student or pair/group
- Printout of the *Programming Instructions Handout* which can be found on the final page
- Pencils and scratch paper

Age/Grade Level

Grades 6 through 12

OzoBlockly Programming Topics

Loops, math, logic, sounds, variables, lights, time

OzoBlockly Mode

OzoBlockly Evo mode must be used. Most blocks used will come from level 4.

Duration

Approximately one or two 60 minute sessions.

Topics

- Computer Science (Loops, variables, logic)
- Math (binary, exponents)

Academic Standards

CCSS.MATH.CONTENT.4.OA.A.3 Solve multistep word problems with whole numbers and having whole-number answers using the four operations, including problems in which remainders must be interpreted
CCSS.MATH.CONTENT.4.OA.C.5 Generate and analyze patterns
CCSS.MATH.PRACTICE.MP1 Make sense of problems and persevere in solving them.
CCSS.MATH.PRACTICE.MP2 Reason abstractly and quantitatively
CCSS.MATH.PRACTICE.MP4 Model with mathematics
CCSS.MATH.PRACTICE.MP5 Use appropriate tools strategically
CCSS.MATH.PRACTICE.MP6 Attend to precision
ISTE 1.c Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways
ISTE 4.c Develop, test and refine prototypes as a part of a cyclical design process
ISTE 5.c Break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem solving
ISTE 6.a Choose the appropriate platforms and tools for meeting the desired objectives of their creation or communication
ISTE 7.a Use digital tools to connect with learners from a variety of backgrounds

Vocabulary

- ***Binary:*** a system of numerical notation that is base 2 rather than base 10
- ***Bit:*** short for binary digit. Ask students what they think it stands for before giving the answer. It is the smallest unit of data in a computer, and can hold either 0 or 1 (almost like a switch with off/on).
- ***Byte:*** 8 bits. Unit of storage that can hold a single character. Most computer storage is referenced in terms of bytes.
- ***Nibble:*** half of a byte. Ask students to guess what half of a byte is called before revealing that it is a nibble.

- ***Decimal:*** base 10 number system, the most common number system
- ***Signed:*** signed number representations are required to encode negative numbers in binary number systems. For 8 bits, values range from -128 to 127. Eight bits can allow you to encode a total of 256 numbers including zero, so for signed numbers half of the values are negative and half are positive (zero being considered positive).
- ***Unsigned:*** positive representation of numbers. For 8 bits, values range from 0 to 255. This is because 8 bits can allow you to encode a total of 256 numbers including zero.
- ***Hard-code:*** fix data in the program in such a way that they cannot be altered without modifying the program

Overview

Students will *hard-code* a 7-bit binary number by programming their Ozobot to speak a series of zeroes and ones. They should not use more bits because the range of values that they can program in OzoBlockly are 0 to 127. Do not have students use negative numbers (this would require 8 bits and the knowledge of how to represent negative numbers in binary).

Students will use their Ozobot as a timer. The front lights will light up one at a time to show how many seconds have passed. At 20 seconds (or less, if the students find it too easy) the Ozobot with give a warning (the street light animation) and then will say the hard coded value out loud. Students will swap Ozobots with a partner and must calculate and say the decimal number before their partner's Evo does.

Related Activities

The Evo Basic Trainings should be completed before beginning this lesson.

Questions?

Please contact us at ozoEdu@ozobot.com

LESSON/ACTIVITY PLAN

Students should understand how to calculate decimal numbers from binary by the end of this lesson. The code will build off of the Timer program, and the coding portion of the activity just requires students to program Ozobot to say a series of zeroes and ones. The majority of the time should be spent learning how binary works and then having students see how many numbers they can convert before their time is up!

Part 1: Group Activity

As a class, go through examples of decimal and binary numbers:

Think of the number 2016. Each digit has 10 possible options, 0 through 9, so we say that **dec**imal is base **10**. For each place, we multiply the coefficient by the base (10) to the power of the place that it is in. We start from zero, so for the ones place it will be the value times 10^0. 2016 can be expanded to look like this:

2016 base 10 = **2**(10^3) + **0**(10^2) + **1**(10^1) + **6**(10^0) = 2000 + 0 + 10 + 6

Binary is a series of 0s and 1s. Here is an example of a binary number:

10101010

Each place has two options, a 0 or a 1. Since we have 2 options instead of 10, we say that binary is base 2. Here is the previous value expanded.

10101010 base 2 =
1$\times 2^7$ + **0**$\times 2^6$ + **1**$\times 2^5$ + **0**$\times 2^4$ + **1**$\times 2^3$ + **0**$\times 2^2$ + **1**$\times 2^1$ + **0**$\times 2^0$ =
128 + 0 + 32 + 0 + 8 + 0 + 2 + 0 =
170

So all of the possible options for a binary number is 2 to the power of bits in the binary number (2^8 is 256 different possible numbers —> so the values 0-255 can be represented within 8 bits). This is one byte because the number has 8 bits.

Optional Class Game:

Put your students into two even groups of 8 (if you have more or less students, you

can put students into groups of 4, 5, 6, or 7). Have them line up shoulder to shoulder, facing the opposing team[s]. Each student will represent a binary digit place. All students start standing, as if they are all ones. The person to the farthest left represents the largest place and the person furthest to the right represents the smallest place.

The instructor will call out a decimal number and students must then calculate the binary number. If a student represents a one then they stand up and if they represent a zero then they stay crouched. Whichever team comes up with the correct solution first gets a point. The game can be best out of 5 or 10 and will take about 20-30 minutes to explain and play.

Here are the ranges of decimal values that can be converted to binary based on the number of students in each group:
Four Students: 0 - 15
Five Students: 0 - 31
Six Students: 0 - 63
Seven Students: 0 – 127
Eight Students: 0 – 255

Part 2: Partner Programming Activity

Task:

Have students split into pairs. Each student will write their own program (using the included timerProgram.ozocode file), and then will have their partner convert the binary value that Ozobot says to decimal before Ozobot says the answer.

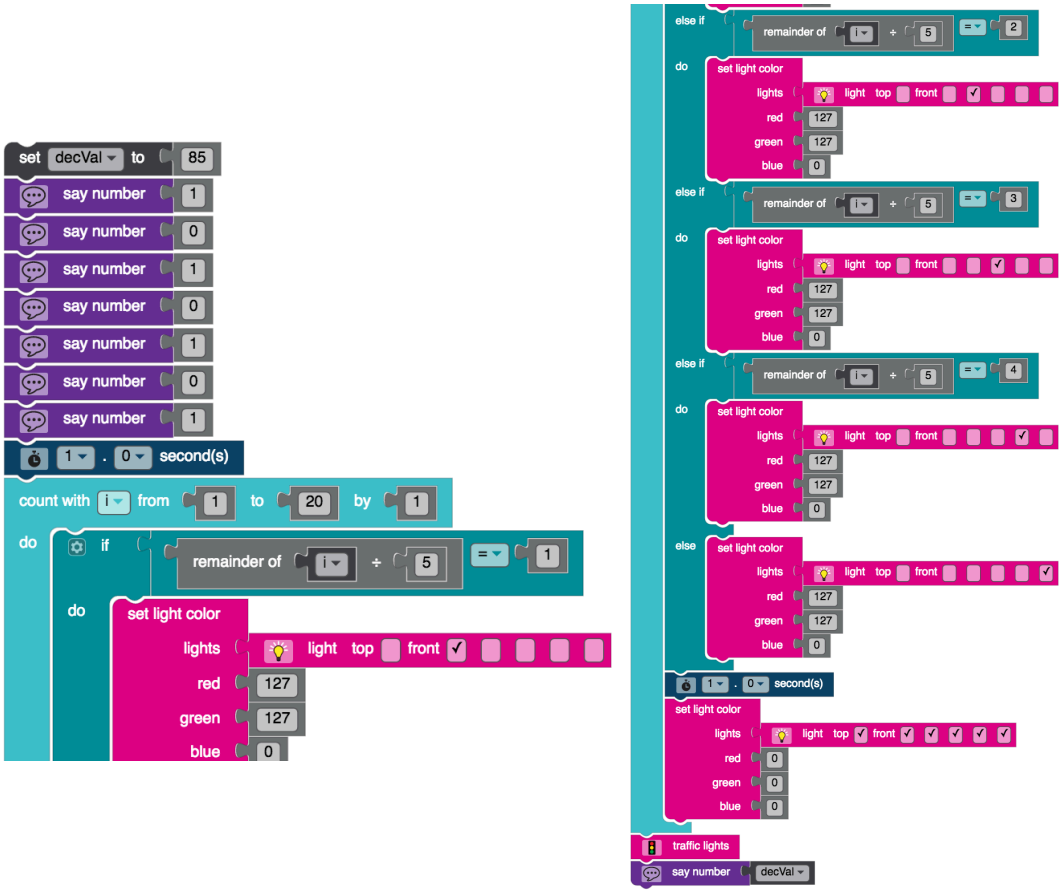
Instructions:

1. First, the students should create a variable called decVal and should store the decimal number in the variable.
2. Students will hard code a series of 0s and 1s (a total of 7 values) that the Ozobot will say out loud using the "say number" block.
3. The Ozobot will wait one second and then will be utilized as a timer, illustrating seconds passing, the same as in the Timer program (attach new code from steps one and two to the top of the Timer program).
4. The instructions for the timer program can be found in the link below.
5. *However*, instead of lighting up with the rainbow animation when the time is up (after either 15 or 20 seconds, depending on how fast the students are) the Ozobot should play the street light animation. This will show the students that they are out of time!
6. Once the animation ends the Ozobot will say the decimal number out loud. If

the student does not say the number before Ozobot then they do not get a point!

7. After students hardcode the values into their program, they will load the program onto the Ozobot Evo and will run their program for their partner. The partner must calculate the decimal value before Ozobot says it out loud. Make the game best out of 5.

Here is an example of what the code could look like:



Instructions for the Timer Program can be found here:
http://portal.ozobot.com/lessons/detail/ebtl2-timer

Notes:

- There is an optional worksheet that will help students convert the binary to decimal, found below
- Otherwise, no handouts are necessary, unless the instructor would like each student to have a copy of the activity instructions.

Optional Explanation of Real World Application:

This explanation is good for AP students who program in languages that are typed.

Data Types and their Sizes:
Char- 1 byte
Integer- 4 bytes
Float-8 decimal places- 4 bytes
Double-15 decimal places- 8 bytes
Bool- 1 byte

- Ask students if they can calculate how many different values can be stored in one byte (2^8 = 256). Then ask if they can calculate the largest possible integer value that can be stored in 4 bytes (4294967295 unsigned, 2147483647 signed). 2^32 will give you one more, but that is how many values can be stored. Since that includes zero, we subtract one from this value to get the largest integer that can be stored in that number of bytes. The maximum signed value is 2^32 divided by two and then one is subtracted. This is because half of the values that can be stored are reserved for the negative values.
- Example: At one point YouTube ran into issues with their video view count because some videos surpassed the maximum (signed) number that could fit into 32 bits (which is explained above). See if students can figure out what positive value was surpassed that made YouTube switch to a 64-bit counter (same as the signed value above).

Binary Blaster Activity

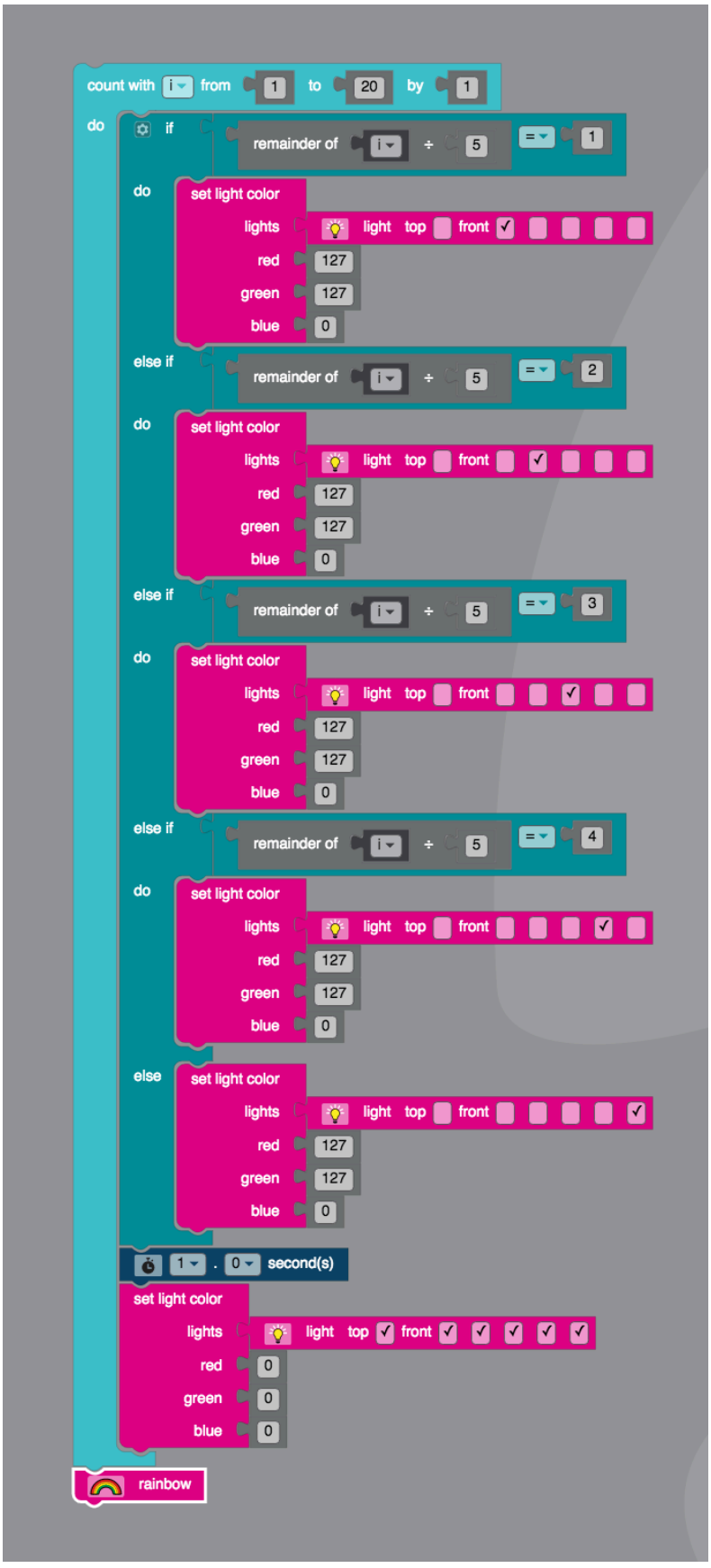
Partner Programming Activity

Task:

Find a partner or group to work with. Each student should write their own program (using the included timerProgram.ozocode file), and then have your partner (or the rest of the group) convert the binary value Ozobot says out loud to decimal before Ozobot says the answer.

Instructions:

1. First, create a variable called decVal and store the decimal number in the variable.
2. Hard code a series of 0s and 1s (a total of 7 values) that the Ozobot will say out loud using the "say number" block. The first number said will be the leading value (i.e. saying the number 2 first in 2016)
3. The Ozobot will wait one second and then will be utilized as a timer, illustrating seconds passing, the same as in the Timer program (attach new code from steps one and two to the top of the Timer program. Image of Timer program is shown below).
4. *However*, instead of lighting up with the rainbow animation when the time is up, the Ozobot should play the street light animation. This will show you that you are out of time!
5. Once the animation ends the Ozobot should say the decimal number out loud. If your partner does not say the number before Ozobot does, then they do not get a point!
6. After you hardcode the values into your program, you will load the program onto the Ozobot Evo and run the program for your partner. Your partner must calculate the decimal value before Ozobot says it out loud. The game should be best out of 5 (each player runs their program five times with different binary numbers hardcoded into the program).


count with i from 1 to 20 by 1
do
if remainder of i ÷ 5 = 1
do
set light color
lights light top front
red 127
green 127
blue 0
else if remainder of i ÷ 5 = 2
do
set light color
lights light top front
red 127
green 127
blue 0
else if remainder of i ÷ 5 = 3
do
set light color
lights light top front
red 127
green 127
blue 0
else if remainder of i ÷ 5 = 4
do
set light color
lights light top front
red 127
green 127
blue 0
else
set light color
lights light top front
red 127
green 127
blue 0
1 . 0 second(s)
set light color
lights light top front
red 0
green 0
blue 0
rainbow

Binary Blaster Game

Place	6	5	4	3	2	1	0
Binary digits	_____	_____	_____	_____	_____	_____	_____
*	*	*	*	*	*	*	*
2 to the place	64	32	16	8	4	2	1
=	=	=	=	=	=	=	=
Add these final values							

Round 1 Decimal Value: _____

Place	6	5	4	3	2	1	0
Binary digits	_____	_____	_____	_____	_____	_____	_____

Round 2 Decimal Value: _____

Place	6	5	4	3	2	1	0
Binary digits	_____	_____	_____	_____	_____	_____	_____

Round 3 Decimal Value: _____

Place	6	5	4	3	2	1	0
Binary digits	_____	_____	_____	_____	_____	_____	_____

Round 4 Decimal Value: _____

Place	6	5	4	3	2	1	0
Binary digits	_____	_____	_____	_____	_____	_____	_____

Round 5 Decimal Value: _____