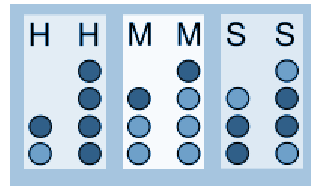
Give Binary a Try!



Provided by TryEngineering - www.tryengineering.org

Lesson Focus

Lesson focuses on how binary codes function and binary applications for computer engineers. The lesson offers students an activity to learn to download software and read online binary clock, and advanced students an opportunity to build one from a kit.

Lesson Synopsis

The Give Binary a Try! lesson explores how binary codes work, how it is applied by computer engineers to computers and other electronic equipment including clocks. Students learn how to use the code, read binary clocks, and advanced students can build their own binary clock from a kit.

Age Levels

8-18.

Objectives

- Learn about binary code and its applications in computing.
- Learn about downloading, running, and managing software applications.
- Learn about wiring, and manufacturing of a simple electronic device.
- Learn how engineering teams approach project work.
- Learn about teamwork and working in groups.

Anticipated Learner Outcomes

As a result of this activity, students should develop an understanding of:

- binary code
- electronic product design
- problem solving
- teamwork

Lesson Activities

Students learn about binary code, and how it is used in computers and other equipment. Students explore a simple use of binary code as a clock. More advanced student assemble a working binary clock.

Resources/Materials

- Teacher Resource Documents (attached)
- Student Worksheets (attached)
- Student Resource Sheets (attached)

Alignment to Curriculum Frameworks

See attached curriculum alignment sheet.

Internet Connections

- TryEngineering (www.tryengineering.org)
- Building a Binary Clock (http://gimpfaq.org/tutorials/binclock)
- Free Binary Clock for Computer Desktop (www.sb-software.com/binaryclock/)
- Another Free Binary Clock for Computer Desktop (www.goldsofts.com/soft/321/37385/Scotts_Binary_Clock.html)
- Electronics USA Binary Clock Kit (http://electronicsusa.com/bc10.html)
- Alltronics Binary Clock Kit (www.alltronics.com/cgi-bin/item/23K006/KC/Binary-Clock-Kit)
- ITEA Standards for Technological Literacy: Content for the Study of Technology (www.iteaconnect.org/TAA)
- National Science Education Standards (www.nsta.org/publications/nses.aspx)

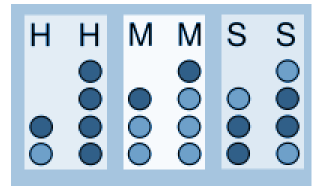
Recommended Reading

- Code: The Hidden Language of Computer Hardware and Software by Charles Petzold (ISBN: 0735611319)
- How Computers Work by Ron White and Timothy Edward Downs (ISBN: 0789736136)

Optional Writing Activity

- Write a paragraph about the history of binary code in computer use.

Give Binary a Try!



For Teachers:
Teacher Resources

◆ **Lesson Goal**

The Give Binary a Try lesson explores how binary code works, how it is applied by computer engineers to computers and other electronic equipment including clocks. Students learn how to use the code, read binary clocks, and advanced students can build their own binary clock from parts or a kit.

◆ **Lesson Objectives**

- Learn about binary code and its applications in computing.
- Learn about wiring, and manufacturing of a simple electronic device.
- Learn how engineering teams approach project work.
- Learn about teamwork and working in groups.

◆ **Materials**

- Student Resource Sheets and Worksheets
- Internet Access (for Binary software downloads)
- Optional: one set of materials for each group of students (Kits cost about $48)

◆ **Procedure**

1. Show students the various Student Reference Sheets. These may be read in class, or provided as reading material for the prior night's homework.
2. Have students complete student worksheet to learn how to read a binary clock.
3. Download binary clock software and set up on classroom or library PC computer so students download and install software and can keep up their binary code skills. This may be assigned or suggested as an extension project for home; software is windows-based.
   - Free Binary Clock for Computer Desktop (www.sb-software.com/binaryclock)
   - Another Free Binary Clock for Windows-based Computer Desktop (www.goldsofts.com/soft/321/37385/Scotts_Binary_Clock.html)
   - MAC Free Binary Clock (http://mac.softpedia.com/get/Dashboard-Widgets/Calculate-Convert/Binary-Clock.shtml)

◆ **Time Needed**

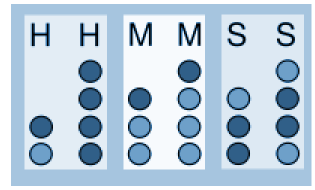
One to two 45 minute sessions

◆ **Advanced Options**

1. For more advanced students, divide students into groups of 2-3 students, providing a set of materials per group. Have them work as a team to construct a working electronic binary clock from a kit. (Note that all kits are about $50)
   - Electronics USA Item BC10 (http://electronicsusa.com/bc10.html)
   - Gibson Tech Ed Item G-540 (www.gssteched.com/G-540.html)
   - Alltronics - Item 23K006 (www.alltronics.com/cgi-bin/item/23K006/KC/Binary-Clock-Kit)
2. Each student group then evaluates the designs developed by other teams, and completes an evaluation/reflection worksheet.

Note: A true binary clock indicating minutes of the hour would display values from 0 to 59, or 000000 to 111011. But this would be more difficult to read since adding values 32 + 16 + 8 + 2 + 1 = 59 is not as easy as 8 + 0 + 0 + 1 = 9. So, these kits work along with the downloadable versions referenced elsewhere in this lesson.

Give Binary a Try!



For Teachers:
Teacher Resources (continued)
Student Worksheet: What Time is it? - SOLUTION

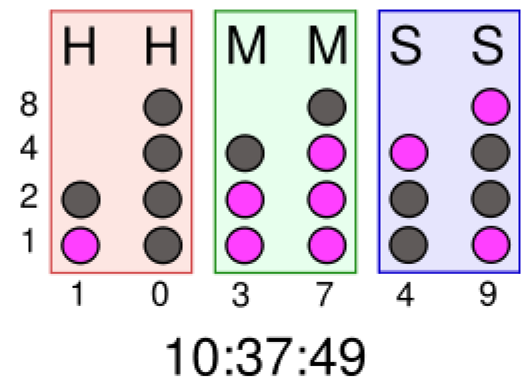
A fun and easy way to learn how binary coding works is to learn how to tell time using the binary system. This worksheet will help you learn the code and how it can be read using a digital binary clock.

◆ **What Time is It?**

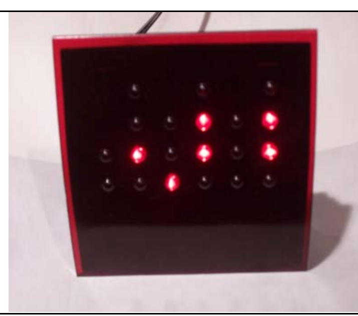
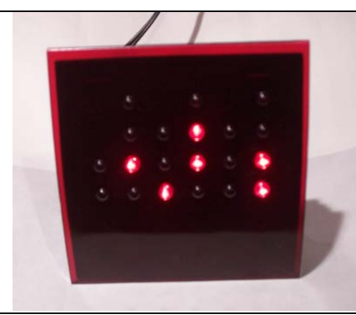
The following clock is set up in an array with numbers represented in the following structure:

Hours		Minutes		Seconds	
	8		8		8
	4	4	4	4	4
2	2	2	2	2	2
1	1	1	1	1	1

Based on which light are illuminated at any given time, you can determine the hour, minute, and seconds. In the illustration below, the time is 10 hours, 37 minutes and 49 seconds.



What time do the following binary clocks say?

		
Time: **Correct Answer: 02:15:39**	Time: **Correct Answer: 02:16:06**	Time: **Correct Answer: 02:16:03**

◆ Binary Bytes and Computer Applications

The binary numeral system (base 2 numerals), or bin for short, represents numeric values using two symbols, typically 0 (off) and 1 (on). Because of its straightforward implementation in electronic circuitry, the binary system is used internally by virtually all modern computers. And, computers can be found in just about every product used in today's society - from cars, to phones, to refrigerators -- and also in most manufacturing processes.

In almost all modern computers, each memory cell is set up to store binary numbers in groups of eight bits (called a byte). Each byte is able to represent 256 different numbers; either from 0 to 255 or -128 to +127. To store larger numbers, several consecutive bytes may be used (typically, two, four or eight). When negative numbers are required, they are usually stored in two's complement notation. Other arrangements are possible, but are usually not seen outside of specialized applications or historical contexts. A computer may store any kind of information in memory as long as it can be somehow represented in numerical form. Modern computers have billions or even trillions of bytes of memory.

◆ How Does It Work?

One can think about binary by comparing it with our usual numbers. We use a base ten system. This means that the value of each position in a numerical value can be represented by one of ten possible symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, or 9. We are all familiar with these and how the decimal system works using these ten symbols. When we begin counting values, we should start with the symbol 0, and proceed to 9 when counting. We call this the "ones," or "units" place.

The "ones" place, with those digits, might be thought of as a multiplication problem. 5 can be thought of as 5×10^0 (10 to the zero power, which equals 5×1, since any number to the zero power is one). As we move to the left of the ones place, we increase the power of 10 by one. Thus, to represent 50 in this same manner, it can be thought of as 5×10^1, or 5×10.

$$500 = (5 \text{ x } 10^2) + (0 \text{ x } 10^1) + (0 \text{ x } 10^0)$$

$$5834 = (5 \text{ x } 10^3) + (8 \text{ x } 10^2) + (3 \text{ x } 10^1) + (4 \text{ x } 10^0)$$

When we run out of symbols in the decimal numeral system, we "move to the left" one place and use a "1" to represent the "tens" place. Then we reset the symbol in the "ones" place back to the first symbol, zero.

Student Resource
Binary Basics (continued)

Binary is a base two system which works just like our decimal system, however with only two symbols which can be used to represent numerical values: 0 and 1. We begin in the "ones" place with 0, then go up to 1. Now we are out of symbols, so to represent a higher value, we must place a "1" in the "twos" place, since we don't have a symbol we can use in the binary system for 2, like we do in the decimal system.

In the binary numeral system, the value represented as 10 is (1 × 21) + (0 × 20). Thus, it equals "2" in our decimal system.

Binary-to-decimal equivalence:

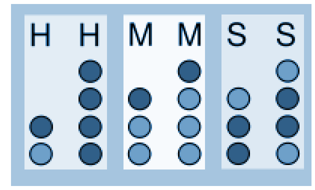
$$1_2 = 1 \text{ x } 2^0 = 1 \text{ x } 1 = 1_{10}$$

$$10_2 = (1 \text{ x } 2^1) + (0 \text{ x } 2^0) = 2 + 0 = 2_{10}$$

$$101_2 = (1 \text{ x } 2^2) + (0 \text{ x } 2^1) + (1 \text{ x } 2^0) = 4 + 0 + 1 = 5_{10}$$

Here is another way of thinking about it: When you run out of symbols, for example 11111, add a "1" on the left end and reset all the numerals on the right to "0", producing 100000. This also works for symbols in the middle. Say the number is 100111. If you add one to it, you move the leftmost repeating "1" one space to the left (from the "fours" place to the "eights" place) and reset all the numerals on the right to "0", producing 101000.

Give Binary a Try!



Student Worksheet A: What Time is it?

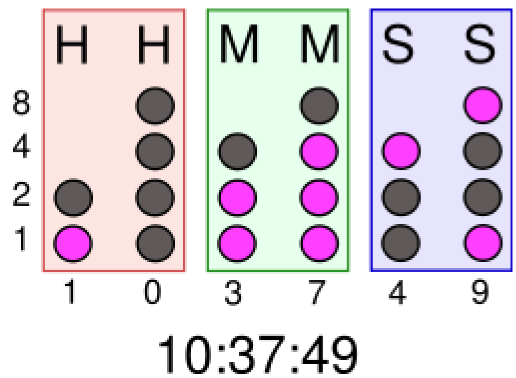
A fun and easy way to learn how binary coding works is to learn how to tell time using the binary system. This worksheet will help you learn the code and how it can be read using a digital binary clock.

◆ What Time is It?

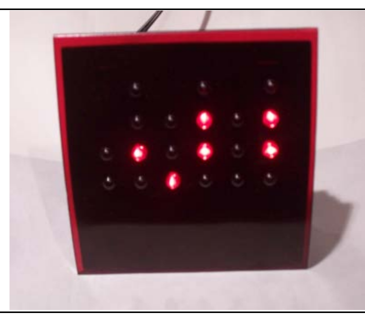
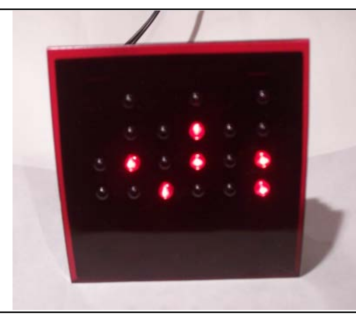
The following clock is set up in an array with numbers represented in the following structure:

Hours		Minutes		Seconds	
	8		8		8
	4	4	4	4	4
2	2	2	2	2	2
1	1	1	1	1	1

Based on which light are illuminated at any given time, you can determine the hour, minute, and seconds. In the illustration below, the time is 10 hours, 37 minutes and 49 seconds.



What time do the following binary clocks say?

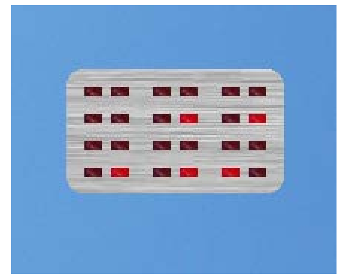
		
Time:	Time:	Time:

◆ **Binary Software Download**

Working as a team of students on one computer, visit one of the following websites and download a binary clock onto your computer.

- Free Binary Clock for Computer Desktop (www.sb-software.com/binaryclock)
- Another Free Binary Clock for Windows-based Computer Desktop (www.goldsofts.com/soft/321/37385/Scotts_Binary_Clock.html)
- MAC Free Binary Clock (http://mac.softpedia.com/get/Dashboard-Widgets/Calculate-Convert/Binary-Clock.shtml)

Complete the following questions:



1. How did your team decide which software to download?

2. How long did the download take? Was it easier or harder than you expected?

3. Once installed, what options did your software offer…which did you try? Which did you prefer? Why? (For example, some offer the option of switching from a vertical to a horizontal view, allows for different looks, or allows you to switch between a 24 or 12 hour clock)

4. Do you think that binary clocks will ever be more popular than standard digital or analog clocks? Why or why not?

5. Why do you think binary code is so important to software engineers?

6. What do you think the future will bring? If you were an engineer, what changes could you envision to clocks and codes for the future?

Give Binary a Try!

Student Worksheet B: Team Engineering

◆ You are a team of engineers which has to tackle the challenge of building your own binary clock. You have been provided with a kit which your team will use to build a functional electric binary clock.

◆ Activity Steps

1. Review the various Student Reference Sheets.

2. Your team has been provided with a binary clock kit. You'll need to follow step by step instructions and work as a team comparing the directions and the materials.

3. Work as a team to construct your clock. Make decisions about how your team will divide up the work, manage the parts, go through the steps. You are acting as manufacturing engineers on this project, determining the best way to create your product.

4. Predict in the box below how much time you estimate it will take to complete the clock.

5. Build your clock -- remember teamwork!

6. Complete the question/reflection area below.

7. Present your clock to the group along with a verbal summary of your reflections.

◆ Questions/Reflections

1. Did your clock work? If not, what do you think went wrong?

◆ Questions/Reflections (continued)

2. What obstacles did you face during construction? How did you overcome these?

3. How did your actual manufacturing time compare with your estimated time? What do you think caused the difference?

4. Did your teams' plan for dividing up the work end up being how the work was completed, or did you change strategies during the project? If you changed your approach to the work, why?

5. Present your binary clock to the class and discuss how your team approached the work and how your plan differed from the actual execution.

6. Did you think that another team had a better approach to completing this project? If so, what would you have done differently in retrospect? Answers might be dividing up the work differently, organizing parts differently, or keeping track of steps.

For Teachers:
Alignment to Curriculum Frameworks

Note: Lesson plans in this series are aligned to one or more of the following sets of standards:

- U.S. Science Education Standards (http://www.nap.edu/catalog.php?record_id=4962)
- U.S. Next Generation Science Standards (http://www.nextgenscience.org/)
- International Technology Education Association's Standards for Technological Literacy (http://www.iteea.org/TAA/PDFs/xstnd.pdf)
- U.S. National Council of Teachers of Mathematics' Principles and Standards for School Mathematics (http://www.nctm.org/standards/content.aspx?id=16909)
- U.S. Common Core State Standards for Mathematics (http://www.corestandards.org/Math)
- Computer Science Teachers Association K-12 Computer Science Standards (http://csta.acm.org/Curriculum/sub/K12Standards.html)

◆National Science Education Standards Grades K-4 (ages 4 - 9)

CONTENT STANDARD A: Science as Inquiry

As a result of activities, all students should develop

- ✦ Abilities necessary to do scientific inquiry

CONTENT STANDARD B: Physical Science

As a result of the activities, all students should develop an understanding of

- ✦ Light, heat, electricity, and magnetism

CONTENT STANDARD E: Science and Technology

As a result of activities, all students should develop

- ✦ Abilities of technological design
- ✦ Understanding about science and technology

◆National Science Education Standards Grades 5-8 (ages 10 - 14)

CONTENT STANDARD A: Science as Inquiry

As a result of activities, all students should develop

- ✦ Understandings about scientific inquiry

◆National Science Education Standards Grades 9-12 (ages 14-18)

CONTENT STANDARD B: Physical Science

As a result of their activities, all students should develop understanding of

- ✦ Interactions of energy and matter

CONTENT STANDARD E: Science and Technology

As a result of activities, all students should develop

- ✦ Abilities of technological design
- ✦ Understandings about science and technology

CONTENT STANDARD F: Science in Personal and Social Perspectives

As a result of activities, all students should develop understanding of

- ✦ Science and technology in local, national, and global challenges

CONTENT STANDARD G: History and Nature of Science

As a result of activities, all students should develop understanding of

- ✦ Historical perspectives

For Teachers:
Alignment to Curriculum Frameworks (continued)

◆**Next Generation Science Standards Grades 3-5 (Ages 8-11)**

Waves and Their Applications in Technologies for Information Transfer

Students who demonstrate understanding can:

- 4-PS4-3. Generate and compare multiple solutions that use patterns to transfer information.

Engineering Design

Students who demonstrate understanding can:

- 3-5-ETS1-1.Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
- 3-5-ETS1-2.Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.

◆**Next Generation Science Standards Grades 6-8 (Ages 11-14)**

Engineering Design

Students who demonstrate understanding can:

- MS-ETS1-2 Evaluate competing design solutions using a systematic process to determine how well they meet the criteria and constraints of the problem.

◆**Principles and Standards for School Mathematics**

Number and Operations Standard

As a result of activities, all students should develop

- Understand numbers, ways of representing numbers, relationships among numbers, and number systems.
- Compute fluently and make reasonable estimates.

Connections Standard

As a result of activities, all students should develop

- Understand how mathematical ideas interconnect and build on one another to produce a coherent whole.
- Recognize and apply mathematics in contexts outside of mathematics.

◆**Common Core State Standards for School Mathematics Grade 3-8 (ages 8-14)**

Number & Operations in Base Ten

- Generalize place value understanding for multi-digit whole numbers.
    - CCSS.Math.Content.4.NBT.A.1 Recognize that in a multi-digit whole number, a digit in one place represents ten times what it represents in the place to its right.
- Understand the place value system.
    - CCSS.Math.Content.5.NBT.A.1 Recognize that in a multi-digit number, a digit in one place represents 10 times as much as it represents in the place to its right and 1/10 of what it represents in the place to its left.
- Apply and extend previous understandings of arithmetic to algebraic expressions.
    - CCSS.Math.Content.6.EE.A.1 Write and evaluate numerical expressions involving whole-number exponents.

Alignment to Curriculum Frameworks (continued)

◆Standards for Technological Literacy - All Ages

The Nature of Technology

- Standard 3: Students will develop an understanding of the relationships among technologies and the connections between technology and other fields of study.

Technology and Society

- Standard 7: Students will develop an understanding of the influence of technology on history.

Design

- Standard 9: Students will develop an understanding of engineering design.

The Designed World

- Standard 17: Students will develop an understanding of and be able to select and use information and communication technologies.

◆CSTA K-12 Computer Science Standards Grades K-3 (ages 5-8)

5.1 Level 1: Computer Science and Me (L1)

- Computational Thinking (CT)
  1. Use technology resources (e.g., puzzles, logical thinking programs) to solve age-appropriate problems.
  4. Recognize that software is created to control computer operations.
  5. Demonstrate how 0s and 1s can be used to represent information.
- Collaboration (CL)
  2. Work cooperatively and collaboratively with peers, teachers, and others using technology.
- Computer Practice and Programming (CPP)
  4. Construct a set of statements to be acted out to accomplish a simple task.
- Computers and Communication Devices (CD)
  1. Use standard input and output devices to successfully operate computers and related technologies.

◆CSTA K-12 Computer Science Standards Grades 3-6 (ages 8-11)

5.1 Level 1: Computer Science and Me (L1)

- Computational Thinking (CT)
  3. Demonstrate how a string of bits can be used to represent alphanumeric information.
- Collaboration (CL)
  2. Use online resources (e.g., email, online discussions, collaborative web environments) to participate in collaborative problem solving activities for the purpose of developing solutions or products.
- Computer Practice and Programming (CPP)
  1. Use technology resources (e.g., calculators, data collection probes, mobile devices, videos, educational software, and web tools) for problem-solving and self-directed learning.
  4. Gather and manipulate data using a variety of digital tools.
  10. Gather and manipulate data using a variety of digital tools.

◆CSTA K-12 Computer Science Standards Grades 6-9 (ages 11-14)

5. 2 Level 2: Computer Science and Community (L2)

- Computational Thinking: (CT)
  7. Represent data in a variety of ways including text, sounds, pictures, and numbers.
  14. Examine connections between elements of mathematics and computer science including binary numbers, logic, sets and functions.
- Collaboration (CL)
  3. Collaborate with peers, experts, and others using collaborative practices such as pair programming, working in project teams, and participating in group active learning activities.
  4. Exhibit dispositions necessary for collaboration: providing useful feedback, integrating feedback, understanding and accepting multiple perspectives, socialization.

◆CSTA K-12 Computer Science Standards Grades 6-9 (ages 11-14)

5. 2 Level 2: Computer Science and Community (L2)

- Computing Practice & Programming (CPP)
  1. Select appropriate tools and technology resources to accomplish a variety of tasks and solve problems.
- Computers & Communications Devices (CD)
  1. Recognize that computers are devices that execute programs.
- Community, Global, and Ethical Impacts (CI)
  4. Evaluate the accuracy, relevance, appropriateness, comprehensiveness, and bias of electronic information sources concerning real-world problems.