Give Binary a Try! - TryEngineering.org Powered by IEEE

tryengineering.org/teacher/lesson-plans/give-binary-a-try

Give Binary a Try!

This lesson focuses on how binary codes function and binary applications for computer engineers. Students complete an activity to learn how to download and install binary clock software and read an online binary clock.

- 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.

Age Levels: 8-18

Materials & Preparation

Build Materials (For each team)

Required Materials

- Internet Access
- Set up classroom PC computer(s) so students can download and install binary clock software.
  - Free Binary Clock for Computer Desktop (www.sb-software.com/binaryclock)
  - Microsoft Free Binary Clock (https://download.cnet.com/Windows-Binary-Clock/3000-2350_4-10448812.html)

Materials for Advanced Students

Binary Clock Kit https://www.tindie.com/products/applemountain/binary-clock-kit-with-red-green-and-blue-lights/

Engineering Design Challenge

Design Challenge

You are part of a team of engineers given the challenge of downloading and installing binary clock software. You will try some options in the software. Discuss as a team whether you think binary clocks will ever be more popular than standard digital or analog clocks.

Advanced Students: 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

Criteria

- Download and install software
- Try software options
- Team discussion

Constraints

Use only the materials provided.

Activity Instructions & Procedures

1. Break class into teams of 2-4.

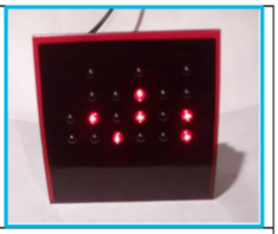
2. Hand out the Give Binary a Try worksheet. Have students complete Student Worksheet A: What Time is It? The solution appears below:**Student Worksheet Solution: What Time Is It?**
   A fun and easy way to learn how binary coding works is to learn how to tell time using the binary system. This worksheet 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 lights 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.

VanVoorhis

What time do the following binary clocks say?

		
Time:	Time:	Time:

3. Discuss the topics in the Background Concepts Section about binary basics.
4. Review the Engineering Design Process, Design Challenge, Criteria, Constraints and Materials.
5. Provide each team with their materials.
    - Free Binary Clock for Computer Desktop
    - Microsoft Free Binary Clock
6. Explain that students must download and install binary clock software from the links provided.
7. Announce the amount of time they have (1 hour recommended).
8. Use a timer or an on-line stopwatch (count down feature) to ensure you keep on time. Give students regular “time checks” so they stay on task. If they are struggling, ask questions that will lead them to a solution quicker.
9. Students meet and complete the challenge.
10. Teams discuss whether they think binary clocks will ever be more popular than standard digital or analog clocks.
11. As a class, discuss the student reflection questions.
12. For more content on the topic, see the “Digging Deeper” section.

Advanced Student Option

Have students complete a binary clock kit, complete Student Worksheet B and answer the reflection questions below.

Binary Clock Kit

Student Reflection (engineering notebook)

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?

Advanced Students Reflection

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

Time Modification

The lesson can be done in as little as 1 class period for older students. However, to help students from feeling rushed and to ensure student success (especially for younger students), split the lesson into two periods giving students more time to brainstorm, test ideas and finalize their design. Conduct the testing and debrief in the next class period.

Engineering Design Process

Watch Video At: https://youtu.be/0YVGm8cOSdA

Background Concepts

Binary Basics

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 × 100 (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.

$500 = (5 \times 10^2) + (0 \times 10^1) + (0 \times 10^0)$

$5834 = (5 \times 10^3) + (8 \times 10^2) + (3 \times 10^1) + (4 \times 10^0)$

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.

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:

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

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

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

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.

Dig Deeper

Internet Connections

- Building a Binary Clock (https://www.instructables.com/id/Easy-Binary-Clock/) or (https://www.instructables.com/id/24-Hour-Binary-Clock-with-Bamboo-Case/)
- Binary Clock Kit
- Free Binary Clock for Computer Desktop
- Microsoft Free Binary Clock

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)

Writing Activity

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

Curriculum Alignment

Alignment to Curriculum Frameworks

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

- S. Science Education Standards (http://www.nap.edu/catalog.php?record_id=4962)
- 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)
- S. National Council of Teachers of Mathematics' Principles and Standards for School Mathematics (http://www.nctm.org/standards/content.aspx?id=16909)
- 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

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

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

CONTENT STANDARD A: Science as Inquiry

As a result of activities, all students should develop

Understandings about scientific inquiry

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

- Personal and community health
- 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

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

- Environmental quality
- Natural and human-induced hazards
- 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

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.

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.

Related Engineering Fields and Degrees

There are many different types of engineering fields that involve designing products and processes. Here are just some of the related engineering fields.

Environmental Engineering

Industrial Engineering

Electrical Engineering

Computer Engineering

Download the Engineering Fields Infographic: How will YOU change the world?

Student Worksheet

Student Worksheet A: What Time is it?

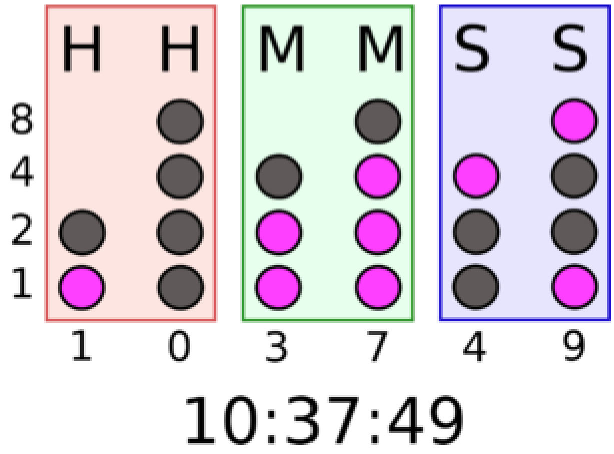
A fun and easy way to learn how binary coding works is to learn how to tell time using the binary system. This worksheet will help you learn the code and how it can be read using a digital binary clock.

What Time is It?

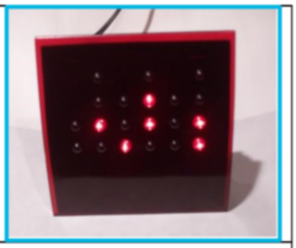
The following clock is set up in an array with numbers represented in the following structure:

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
- Another Free Binary Clock for Windows-based Computer Desktop (www.goldsofts.com/soft/321/37385/Scotts_Binary_Clock.html)
- MAC Free Binary Clock

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?

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?

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.