Digital Representation of Information
(digital and binary)

- We all know that typical modern computers are ***digital***
  - (in addition to being *electronic* and *general purpose* as discussed previously)
  - As opposed to being *analog*
- And that they represent/manipulate numbers in ***binary***
  - As opposed to in *decimal* that we humans are more familiar with
- You probably have some idea about what *digital* and *binary* mean
  - Good to be clear about what they really mean
  - And to be clear about why most modern computers are digital and do things in binary

Digital Representation of Information
(what does digital mean)

- Why is a *digital* watch described as being ***digital***?
  - How is one that is *analog* different?

- Because it uses ***digits*** to tell time (# of values it can show is *finite*) ***discretely*** (not continuously)
  - An analog one uses *hands* instead
  - Hand positions must be interpreted (to tell what time)
    - They are (perceived to be) moving *continuously* with time
    - # of values that can be shown are (perceived to be) *infinite*

- It is a *digital system*
  - Deals with *discrete data discretely*
    - # of values involved is *discrete* (*finite*) → fixed by the set of digits used
    - Values displayed only *at discrete points* (*not continuously*)

Digital Representation of Information
(digital system)

- Any set that is restricted to a ***finite*** number of elements contains *discrete* data
  - *E.g. 1*: the 10 decimal digits
  - *E.g. 2*: the 26 letters of the alphabet
  - *E.g. 3*: the 52 playing cards
  - *E.g. 4*: the 64 squares of a chessboard

Associated verbs:
discretize, digitize
Meaning via example.

- Our typical computer is another example of ***digital system***
  - It deals with (manipulates) *discrete* elements of data
- Early computers → used mostly for numeric computations
  - (Do you know what the 2 main applications were back then?)
  - Discrete data involved → finite set of values with each value being some combination of the numeric digits
  - The term "*digital* computer" originated from such applications

Digital Representation of Information
(signals and binary signals)

- It's found convenient to represent discrete elements of data in digital systems using transiently varying (pulsing) physical quantities called ***signal***s
- The most common and detectable (measurable) signals are ***electrical*** signals such as *voltages* and *currents*
- Technologies leading to the development of devices that implement the use of electrical signals in digital systems shaped up in such a way that ***electronic*** devices called *transistor*s became predominant
- Most modern digital systems are thus electronic and they use ***binary*** signals
- *What does binary mean* and *why binary*?

Digital Representation of Information
(what binary means)

- Binary → used in situations that involve *2* in some way
  - A *binary signal* is one that distinguishes between only *2 states*
- The 2 states can be used to represent 2 different values
- Many different ways can be used to indicate a 2-state condition such as the 2 states of a binary signal
  - "ON" and "OFF"
  - "TRUE" and "FALSE"
  - "HIGH" and "LOW"
  - (and so on)
- But the most common way uses the digits **1** and **0**
  - Each is called a ***binary digit*** or ***bit***

Digital Representation of Information
(what binary means)

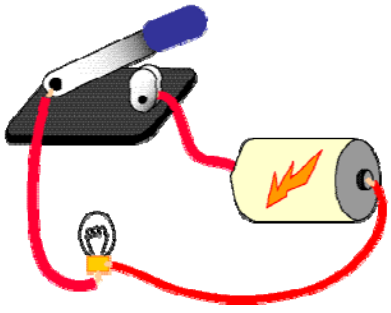
- Which of the following is a binary device?
  - A toggle light switch
  - A light dimmer
  - The ignition switch of an automobile
  - The hour hand of a clock
  - A button on a hand calculator
  - The volume control on a stereo
- Answers:
  - Yes, No, No, No, Yes, No
  - Can you explain why?

Digital Representation of Information
(why binary)

- They are at least 3 reasons why binary signals are used
- *Simplicity*
  - Device → easy/cheap to build
  - Device → small (can put lots of them in very little space)
- *Unambiguity*
  - Easy to discern
  - Noise immunity
  - Reproducibility
- *Universality*
  - Can represent any kind of data

Digital Representation of Information
(why binary - simplicity)

- Which of the following is easier to build?
  - A toggle (ON/OFF) light switch
  - A light dimmer
- Answer/reason is obvious
- Same is true for the tiny transistors inside a silicon chip
- Also means…
  - Cheap, small, reliable
  - Millions of them can be put in a small space

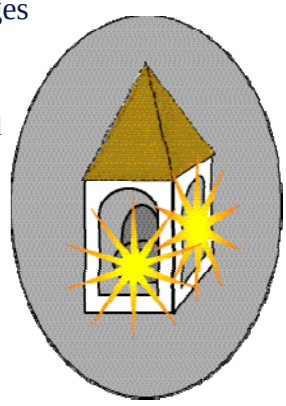


Digital Representation of Information
(why binary - Unambiguity)

- Which of the following is easier to determine?
  - Exactly how bright a light is
  - Whether a light is ON or OFF
- Answer is again obvious
- But do you see why lack of ambiguity is a tremendous advantage?
  - The tale of Paul Revere illustrates this

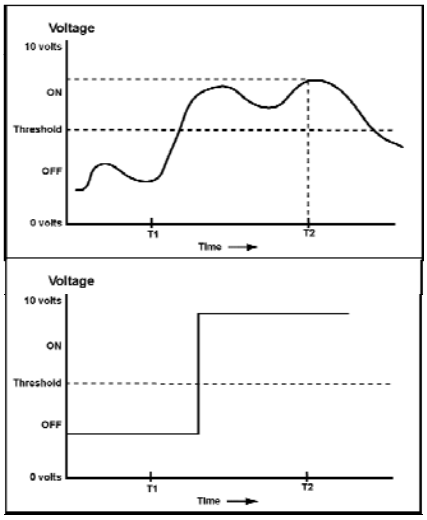
Digital Representation of Information
(why binary - Unambiguity)

- Paul Revere is to ride through Middlesex villages and farms to raise the alarm of British attack:
  - He waits for news on attack of British troops
  - He expects to see a signal lantern in the tower of Old North Church telling him how the attack comes
  - How should the signal be?
  - *Bright* if by sea and *not so bright* if by land?
    - The signal came on and Paul Revere's famous ride is delayed for several hours
    - He must figure out just how bright the signal is
  - 1 lantern if by land and 2 lanterns if by sea?
    - All he has to do is count



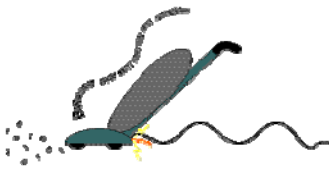
Digital Representation of Information
(why binary - Unambiguity)

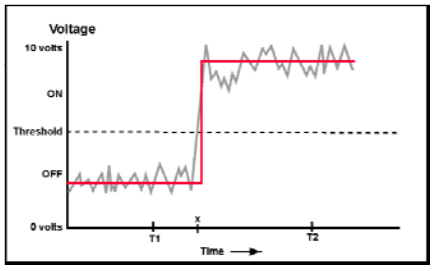
- Electrical signals such as voltages are really *analog*
  - Continuously changing values
- But what we want is discrete *binary* signals
  - What must be done?
- If a ***threshold*** is agreed upon
  - Any voltage below threshold is counted as OFF, ON otherwise
- But why not multiple thresholds?



Digital Representation of Information
(why binary - Unambiguity)

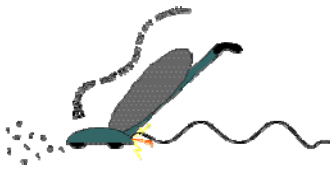
- Because the real world is full of noise
- Using one threshold (thus binary signals) is the most tolerant to noise
  - Compared to using multiple thresholds (thus multi-level signals)

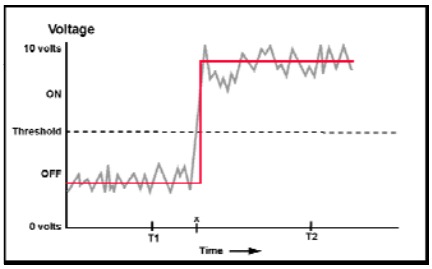




Digital Representation of Information
(why binary - Unambiguity)

- Something might still be wrong with our binary signal
  - ON or OFF at time *x*?
  - It's difficult to tell
  - It's different for different noise
  - How to deal with the problem?
- ***Clock*** to the rescue
  - Signal is sampled only at particular times
  - Signal changes occur between times of sampling



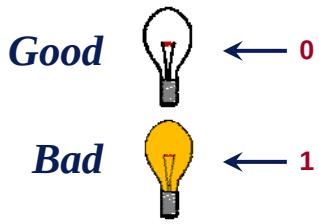


Digital Representation of Information
(why binary - Universality)

- Binary signals can be used to represent just about any types of data
  - By the way, what types of data should a modern electronic computer typically be able to handle?
- Because *any finite set of numbers* can be represented by a collection of binary signals (will use ***bit***s from here on)
- And different types of data can be represented by using some agreed upon ***encoding scheme***s
  - *E.g. 1*: characters encoded using numbers (ASCII codes)
  - *E.g. 2*: machine instructions encoded using numbers (opcodes)
- But how are bits to represent any finite set of numbers?

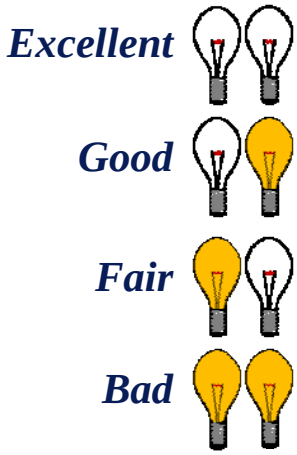
Digital Representation of Information
(how bits represent any finite set of numbers)

- What if we want to represent **2** different things (numbers included)
  - 1 bit (switch) can be used to distinguish up to **2** (= 2^1) quality ratings



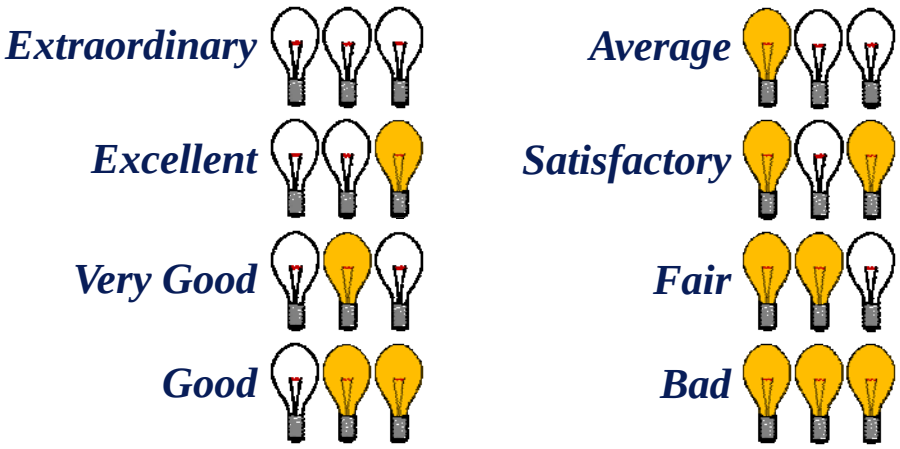
Digital Representation of Information
(how bits represent any finite set of numbers)

- What if we want to represent **4** different things (numbers included)
  - 2 bits (switches) can be used to distinguish up to **4** (= 2^2) quality ratings



Digital Representation of Information
(how bits represent any finite set of numbers)

- What if we want to represent **8** different things (numbers included)
  - 3 bits (switches) can be used to distinguish up to **8** (= 2^3) quality ratings



Digital Representation of Information
(how bits represent any finite set of numbers)

- The preceding illustration allows us to generalize as follows
  - n bits (switches) can be used to distinguish up to 2^n possible states/conditions concerning a certain characteristic of interest (about something)
- ***Quick quiz:*** If we use *n* bits to represent *nonnegative* numbers starting from 0 (*i.e.*, 0, 1, 2, 3, …), what is the *largest* number we can represent?
  - ***Answer:*** $(2^n - 1)$ → Do you see why it is $(2^n - 1)$ and not 2^n ?
- ***Quick quiz:*** If I want to represent students' grades (A, B, C, D or F) using the binary system, what is the *minimum* number of bits that I will need to store the grade of *each* student?
  - ***Answer:*** 3

Digital Representation of Information
(how anything can be represented in binary)

- Chinese and Japanese characters also have been translated into bit patterns
  - Computers can manipulate these symbols just as easily as western characters coded in ***ASCII*** (which uses ***8*** bits)
- ***Unicode*** is a coding scheme created by an international committee on how to represent characters using ***16*** (or more) bits
  - *E.g.:* 111110011111110 is the 16-bit unicode representation for theChinese/Japanese character shown here

Digital Representation of Information
(how anything can be represented in binary)

- Suppose the international committee decides to represent a new Chinese character
- How can they do it?
  - Find a bit pattern not yet used to represent any symbol
  - Assign the new character to that bit pattern
- Correspondence between human language symbols and bit patterns is arbitrary
  - What is important is that enough bits are used so that there are enough bit patterns to uniquely represent all the symbols of the language

Digital Representation of Information
(how anything can be represented in binary)

- If you are still not convinced that anything can be represented in binary, think about it this way:
  - Pick some subject (can be anything)
  - Use English sentences to describe it
  - Represent the English sentences in ASCII
  - Now the subject is represented in binary
- Notice that the above says nothing about how easy or difficult the representation can be used
  - Some schemes are very useful, others almost useless
  - Figuring out how to represent things in useful ways is one major challenge for computer scientists
  - Much work is done in recent years to figure out how to best represent multimedia data such as video and audio data

Digital Representation of Information
(question)

- We've just seen that anything can be represented in *binary*
- Can anything also be represented in *decimal*?
  - And in others (*octal*, *hexadecimal*, …) as well?

(recall)

- Simplicity → small, easy/cheap to build
- Unambiguity → discernability, noise immunity, reproducibility
- Universality → can represent any kind of data