# ECE 218 — Digital Systems

Based on lectures by Dr. Suresh Borkar

Notes taken by Pablo C. Bedolla Ortiz

Spring 2025

These notes are my own and are not endorsed by the lecturers. I have often made significant modifications to them after the lectures, so they may not accurately reflect the content presented in class. Any errors are almost certainly my own.

**Information**

Information is defined as a set of datum received or communicated to resolve an uncertainty about a particular fact or circumstance. After receiving data, we know more about it. The greater the uncertainty about some fact $n$, the more information received will have conveyed.

# 0 Information

We quantify information as it allows us to objectively measure the amount of uncertainty reduction or surprise contained within a piece of data received or observed.

## 0.1 Quantification of Information

**Definition** (Calude Shannon, Theory of Information, 1948)**.** Given a discrete random variable $x_n \in X$ of $n$ possible distinct choices, the probability that $x_n$ will take on the value $x_k$ is given by the associated probability $p_k$. The smaller the probability $p_k$ is, the more likely is it that $x_n$ will take on that particular value $x_k$. The information received upon learning that the choice was $x_k$ is defined by:

$$I(x_k) = \log_2\left(\frac{1}{p_k}\right)$$

From this, we can derive that $\frac{1}{p_k} \propto x_k$. Term inside of the log is the uncertainty of that particular choice. The operation:

$$\log_r\left(\frac{1}{p_k}\right)$$

is an information measurement for a number system of *radix* $r$, therefore $I(x_k)$ being an information measurement for binary digits $\{0, 1\}$. In the case that you are encountered with an $n$ equally probable choice which you receive data narrowing it down to $m$ choices, the probability that the data would be sent is $m \times \frac{1}{n}$. Thus, we define the amount of information you have received as:

$$I(\text{data}) = \log_r\left(\frac{1}{m \times \frac{1}{n}}\right) = \log_r\left(\frac{n}{m}\right)$$

**Example** (Information Content of a Coin Toss)**.** Suppose that you flip a \$0.25 fair coin. The discrete possible choices are $\{c_1, c_2\}$. Let $n = 2$ for the cardinality of the set of choices, and $m$ be the resulting outcome, then the information received is

$$\log_r\left(\frac{n}{m}\right) = \log_2\left(\frac{2}{1}\right) = 1 \text{ bit}$$

It would take one bit to represent each possibility in this set of choices. In the case of a fractional bit such as the rolling of two-die

$$\log_2\left(\frac{36}{1}\right) = 5.17 \text{ bits}$$

then it is only reasonable to use 6-bits to represent the information.

## 0.2 Entropy

The entry of a random variable $x$ is an average of information received when learning the value of the random variable. It is how much information is contained in each piece of data received. Thus, the expected value:

$$H(x) = E\left(I(x)\right) = \sum_{i=1}^{N} p_i \times \log_2 \frac{1}{p_1}$$

**Example** (Random of Four)**.** Let $X = \{A, B, C, D\}$ where $n$ can take on one of four values. We define the choices and their associated probabilities:

| $n_i$ | $p_i$ | $\log_2\left(\frac{1}{p_i}\right)$ |
|---|---|---|
| A | 1/3 | 1.58 bits |
| B | 1/2 | 1 bit |
| C | 1/12 | 3.58 bits |
| D | 1/12 | 3.58 bits |

We can then calculate the entropy:

$$H(x) = \left(\frac{1}{3}\right)(1.58) + \left(\frac{1}{2}\right)(1) + 2\left(\frac{1}{12}\right)(3.58) = 1.626 \text{ bits}$$

We can represent $X$ with a maximum of two bits.

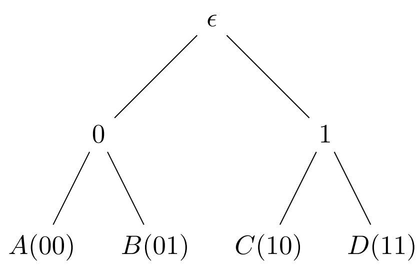
## 0.3 Encodings

An encoding is an unambiguous mapping between bit strings and the members of the set of data to be encoded. We can represent these unambiguous encoding as binary trees where the labels on the path from the root to the leaf give an encoding for that particular leaf. We observe fixed-length encoding tree ($\epsilon$), where all choices are equally likely of occurring (and we have no reason to expect otherwise). This encoding uses the least enough bits to represent the information content.

**System Encoding Scheme**

$$A \to 00$$
$$B \to 01$$
$$C \to 10$$
$$D \to 11$$

**Fixed-Length Tree**




Since we have $n$ elements, all equally probably $1/n$, we define the entropy of that discrete set as

$$\sum_{i=1}^{n}\left(\frac{1}{n}\right)\log_2\left(\frac{1}{\frac{1}{n}}\right) = \log_2(n)$$

As an example, the American Standard Code for Information Interchange (ASCII) encoding standard uses 7-bits to represent the 94 printing characters

$$\log_2(94) = 6.555$$

Thus, we can see

$$\log_2(\text{radix}) \ \textbf{ or } \ \log_2(\text{radix}) + 1$$

is a method we can use for the least enough bits to represent information

## 0.4 Number Systems and Radix Conversion

We use the decimal system in our everyday lives such that each position has a *weight* that is a power of 10. A number expressing in a base-$r$ system will have coefficients multiplied by powers of $r$ (radix). Let $S$ be the set of numbers $\{0, \ldots, 9\}$. We describe a number system whose elements $e \in S$ and is of a base $r$ where an arbitrary sum between $n$ and $m$ can be defined as

$$e_n \times r^n + e_{n-1} \times r^{n-1} + \cdots + e_2 \times r^2 + e_1 \times r^1 + e_0 + e_{-1} \times r^{-1} + \cdots + e_{-m} \times r^{-m}$$

We then see that $r$ is a value equal to the cardinality $|S|$ and its elements $e_n$ are between $0 \leq e < r$. A base-10 system would consist of the set

$$S_{10} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

where $r = |S_{10}|$ and contains $r - 1$ elements from $0 \to (r-1)$. For systems with $e \geq 10$, we use the letters $\{A_{10}, B_{11}, C_{12}, D_{13}, E_{14}, F_{15}\} \to \{10, 11, 12, 13, 14, 15\}$. To convert any number $n$ to $\{\text{Binary}, \text{Octal}, \text{Hexadecimal}\}$,

**Example** (Base-5 System)**.** We define a base-5 system defined by the set $B = \{0, 1, 2, 3, 4\}$. A number $n$ with $i$ digits such that every $n_i \in B$ is a base-5 number. If $i = 6$, then we can represent its base-10 equivalent number with the polynomial sequence as

$$\begin{aligned}(4320.10)_5 &= \left(4 \times 5^3\right) + \left(3 \times 5^2\right) + \left(2 \times 5^1\right) + \left(0 \times 5^0\right) + \left(1 \times 5^{-1}\right) + \left(0 \times 5^{-2}\right) \\ &= (585.2)_{10}\end{aligned}$$

### 0.4.1 Binary System

Binary digits, *bits*, are $e \in \{0, 1\}$. We refer to $2^{10}$ as K (kilo), $2^{20}$ as M (mega) $2^{30}$ as G (giga), and $2^{40}$ as T (tera). This is computer work. One *byte* is equal to eight bits.