CSE 260 – Lecture 1

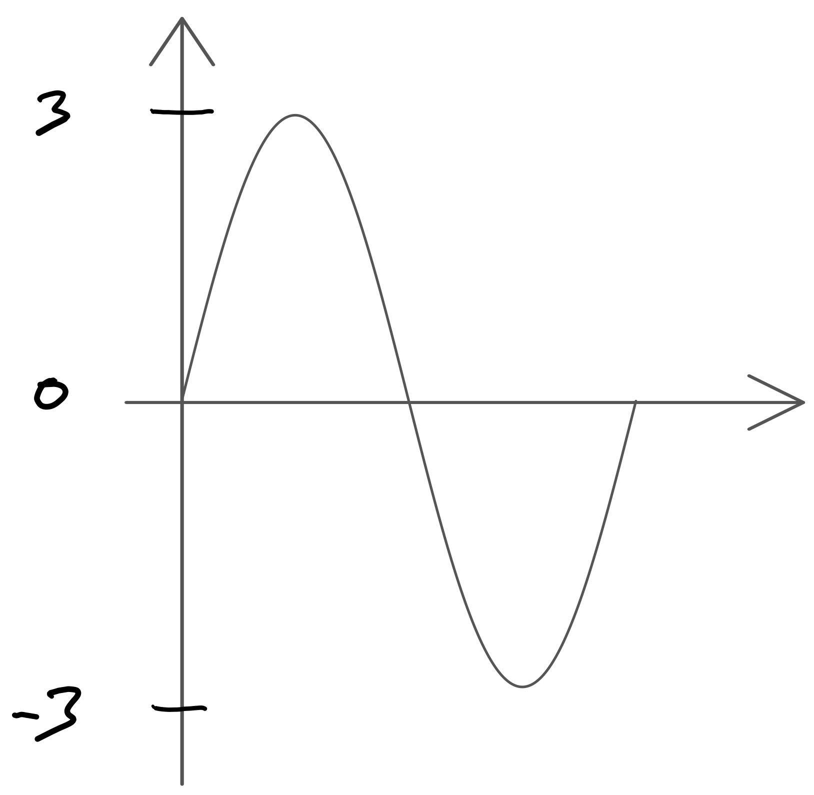
Analog:

⇒ Analog Data can vary over a continuous range of values.

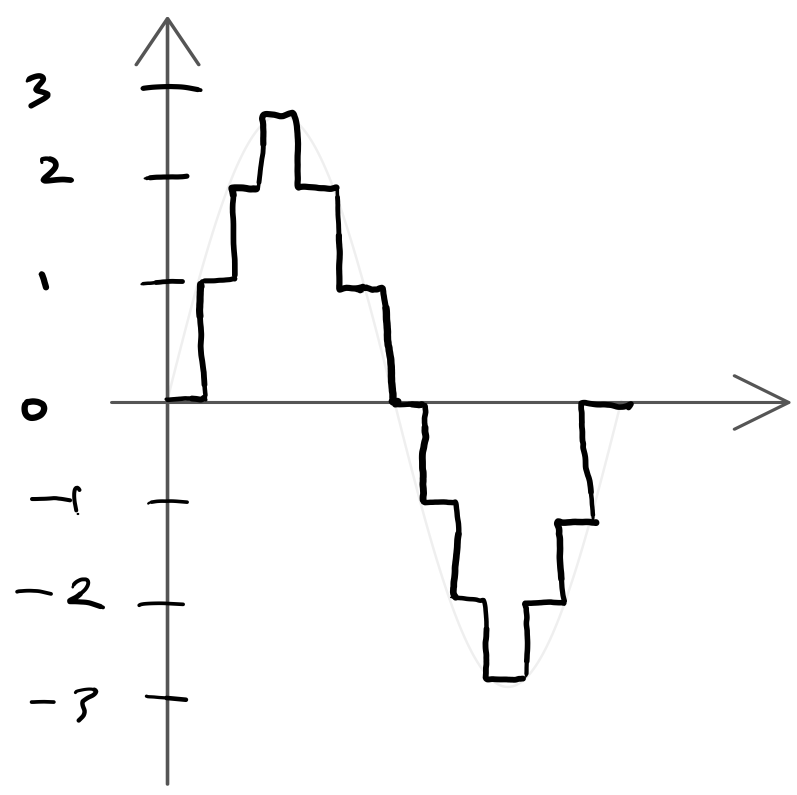
⇒ The data of our world is mostly analog.

Digital: Digital quantities can take on only discrete values.

⇒ Example: (0, 1), alphabet.



Analog plot of $f(x) = 3\sin x$



Digital plot of $f(x) = 3\sin x$

★ Since real world is mainly analog, we overcome this by following process:

Analog → Digital + processing → Processed Analog

Number System

→ A number system is a way of representing and expressing number using a consistent set of symbols & rules.

→ Example: Decimal number system:

→ Base = 10

→ Unique Digits = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Binary number system:

→ Base = 2

→ Unique Digits = 0, 1

Hexadecimal number system:

→ Base = 16

→ Unique Digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

(10) (11) (12) (13) (14) (15)

In this course, we mostly work with Binary code as computer can only understand if there is voltage (1) or not (0).

We represent numbers of multiple base in the following manner:

$$(25)_{10},\quad (110111)_2,\quad (AB12)_{16},\quad (A125)_{13}$$

↑ Base ↑ Base ↑ Base ↑ Base

Some formulas of number system

In Decimal (10),

1	digit	can	represent	[0 – 9]	total	10 = 10^1	values.
2	digits	"	"	[0 – 99]	"	100 = 10^2	"
3	"	"	"	[0 – 999]	"	1000 = 10^3	"
∴ n	"	"	"	[0 – (10^n – 1)]	"	= 10^n	"

Similarly in Binary (Base 2)

1	digit	can	represent	[0, 1]	total	2 = 2^1	values
2	digits	"	"	$[\underset{0}{00}, \underset{1}{01}, \underset{2}{10}, \underset{3}{11}]$	"	4 = 2^2	"
3	"	"	"	$[\underset{0}{000}, \underset{1}{001}, \underset{2}{010} \dots \underset{6}{110}, \underset{7}{111}]$	"	8 = 2^3	"
∴ n	digits	"	"	$[\underset{0}{0}, \underset{1}{1}, \underset{2}{01}, \dots \dots \underset{(2^n - 1)}{\dots}]$	total	2^n	values.

Similarly, in any Base B,

n digit can represent $\boxed{[0 - (B^n - 1)]}$ total $\boxed{B^n}$ values.

We can reverse this formula and say,

to represent M values, we need $\boxed{\text{Round\_up}(\log_B M)}$ digits.

Example:

* To represent $(547)_{10}$ in binary, we need $\log_2(547)$ bits

$$= 9.095$$
$$\approx 10 \text{ bits.}$$

* Similary, to represent $(547)_{10}$ in octal, we need $\log_8(547)$ digits

$$= 3.031$$
$$= 4 \text{ digits.}$$

* Can we represent $(247)_{10}$ in 8 bit Binary number?

$\Rightarrow$ Range of 8 bit binary number is,

$$[0 - (2^8 - 1)]$$
$$\Rightarrow [0 - 255]$$

$\therefore$ We can represent 247 in 8 bit Binary number.

* Can we represent 512 in 9 bit Binary?

$$[0 - (2^9 - 1)] = (0 - 511).$$

$\therefore$ No, we can not.

Number Conversion:

Based on how to convert, We can divide number conversion in 3 types.

Type 1: Decimal to other bases

Type 2: Other Base to Decimal

Type 3: Binary to Octal / Hexa Decimal & Vice Versa

Type 1

Decimal to other Base:

$$(\underbrace{43}_{\text{Whole number}} . \underbrace{3125}_{\text{Fractions}})_{10} \leftarrow \text{Base}$$

* Whole number → Repeated Division by Base R.
* Fraction number → Repeated Multiplication by Base R.

Example: $(43.3125)_{10} = (\,?\,)_2$

$$\begin{array}{r|l l}
2 & 43 & \\
2 & 21 & \rightarrow 1 \\
2 & 10 & \rightarrow 1 \\
2 & 5 & \rightarrow 0 \\
2 & 2 & \rightarrow 1 \\
2 & 1 & \rightarrow 0 \\
 & 0 & \rightarrow 1
\end{array}$$

MSB

LSB

We stop at 0.

$\therefore (43)_{10} = (101011)_2$

	Carry
$.3125 \times 2 = 0.625$	0
$.625 \times 2 = 1.25$	1
$.25 \times 2 = 0.5$	0
$.5 \times 2 = 1.0$	1

LSB

MSB

We stop when it reach 0 or the limit will be given

$\therefore (.3125)_{10} = (.0101)_2$

$\therefore (43.3125)_{10} = (101011.0101)_2$

LSB = Least Significant Bit

MSB = Most Significant Bit

$(34.215)_{10} = (?)_5$

$$
\begin{array}{r|l}
5 & 34 \\
\hline
5 & 6 \rightarrow 4 \\
\hline
5 & 1 \rightarrow 1 \\
\hline
 & 0 \rightarrow 1
\end{array}
\uparrow
$$

$(34)_{10} = (114)_5$

$.215 \times 5 = 1.075$	1
$.075 \times 5 = 0.375$	0
$.375 \times 5 = 1.875$	1
$.875 \times 5 = 4.375$	4
$.375 \times 5 = 1.875$	1
$.875 \times 5 = 4.375$	4
⋮	⋮

$(.215)_{10} = (101414\ldots)_5$

$\therefore (34.215)_{10} = (114.101414\ldots)_5$

Do it yourself :

- Convert the following decimal number to equivalent binary numbers: $(4195)_{10}$
- Convert the following decimal number to equivalent binary numbers: $(3785.65625)_{10}$
- Convert the following decimal number to equivalent binary numbers: $(4785.150263)_{10}$ [for infinite fractional part, just do 6-7 steps and use dots for the rest]
- Convert the following decimal number to equivalent base 5 numbers: $(4123)_{10}$
- Convert the following decimal number to equivalent hexadecimal numbers: $(513)_{10}$
- Convert the following decimal number to equivalent base 9 numbers: $(813)_{10}$

Type 2

Any Base to Decimal.

$(1101.101)_2 = (?)_{10}$

Positions

$$\begin{array}{ccccccccccc} \cdots & 3 & 2 & 1 & 0 & & -1 & -2 & -3 & \cdots \\ & 1 & 1 & 0 & 1 & . & 1 & 0 & 1 & \end{array}$$

$= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$

Base (Position)

$= (13.625)_{10}$

$\star\ (572.6)_8 = (?)_{10}$

$$\Rightarrow \begin{array}{ccccc} 2 & 1 & 0 & & -1 \\ 5 & 7 & 2 & . & 6 \end{array}$$

$= 5 \times 8^2 + 7 \times 8^1 + 2 \times 8^0 + 6 \times 8^{-1}$

$= (378.75)_{10}$

Do it yourself :

- Convert the following binary numbers to equivalent decimal numbers.
  - (a) $(101110001001)_2$
  - (b) $(11011.101)_2$

Type 3

Binary → Octal / Hexa Decimal

[for octal, we select 3 bits]

* $(010111011001 \cdot 101110)_2 = (?)_8$

$$(\underbrace{010}_{2}\,\underbrace{111}_{7}\,\underbrace{011}_{3}\,\underbrace{001}_{1} \cdot \underbrace{101}_{5}\,\underbrace{110}_{6})_2$$

$\therefore (2731 \cdot 56)_8$

* $(0101\ 1101\ 1001 \cdot 1011\ 1000)_2 = (?)_{16}$

$$(\underbrace{0101}_{5}\,\underbrace{1101}_{D}\,\underbrace{1001}_{9} \cdot \underbrace{1011}_{B}\,\underbrace{1000}_{8})_2$$

$= (5D9 \cdot B8)$

Octal / Hexa Decimal → Binary

$(5D9.B8)_{16} = (?)_2$

0101 1101 1001 · 1011 1000

$= (101\ 1101\ 1001 \cdot 10111)_2$

$(731 \cdot 56)_8 = (?)_2$

111 011 001 · 101 110

$= (111011001 \cdot 10111)_2$

* Do it yourself: $(AB \cdot 19)_{16} \rightarrow (?)_8$

[Hint: convert to binary
convert binary to octal.]

(Any Base) Base - R Addition

Decimal Addition :

We all know this.

$$\begin{array}{r} {\scriptstyle 11\ \ } \\ 1245 \\ 982 \\ \hline 2227 \end{array}$$

11 → Carry

$$\begin{array}{r} 4 \\ +8 \\ \hline 12 \end{array} \qquad \begin{array}{r} 1 \\ 9 \\ +2 \\ \hline 12 \end{array}$$

carry sum carry sum

Binary Addition :

$$\begin{array}{r} 1\ 1\ 1\ 1\ 1\ \ \\ 101101 \\ 100111 \\ \hline 1010100 \end{array}$$

$$\begin{array}{r} 1 \\ 1 \\ \hline 10 \end{array} \qquad \begin{array}{r} 1 \\ 1 \\ 1 \\ \hline 11 \end{array}$$

carry sum carry sum

Other Base :

$$\begin{array}{r} 2\ 1\ \ \ \\ (34)_5 \\ (41)_5 \\ (24)_5 \\ \hline 204 \end{array}$$

Base 5 range [0 – 4]

4+1+4 = 9 not in range.

$$\begin{array}{l} 5\,|\,9 \\ 5\,|\,1 \rightarrow 4 \\ \ \ \ 0 \rightarrow 1 \end{array} \uparrow$$

14

carry sum

1+3+4+2 = 10

$$\begin{array}{l} 5\,|\,10 \\ 5\,|\,2 \rightarrow 0 \\ \ \ \ 0 \rightarrow 2 \end{array} \uparrow$$

20

c s

∴ Ans: $(204)_5$

★ Do practice with other bases.

(Any Base) Base - R Multiplication

Binary:

```
    1011
  x  101
  ------
    1011
   0000x
  1011xx
  ------
  110111
```

Other Base:

$$(2A3C)_{16} \times (B7)_{16} = ?$$

Range [0 – 15]

```
     7  2  8
  1  4  1  5
     2 A 3 C
  x      B 7
  ----------
     1 2 7 A 4
(+) 1 D 0 9 4 x
  ------------
  1 E 3 0 E 4
```

$12(C) \times 7 = 84 \rightarrow$ out of range

16 | 84
16 | 5 → 4 ↑
0 → 5

54 (c: 5, s: 4)

$3 \times 7 + 5 = 26 \rightarrow$ not in range

16 | 26
16 | 1 → 10 (A) ↑
0 → 1

1A (c: 1, s: A)

$10(A) \times 7 + 1 = 71$

16 | 71
16 | 4 → 7 ↑
0 → 4

47 (c: 4, s: 7)

$7 \times 2 + 4 = 18$

16 | 18
16 | 1 → 2 ↑
0 → 1

12 (c: 1, s: 2)

$12(C) \times 11(B) = 132$

16 | 132
16 | 8 → 4
0 → 8

84 (c: 8, s: 4)

$3 \times 11(B) + 8 = 41$

16 | 41
16 | 2 → 9 ↑
0 → 2

29 (c: 2, s: 9)

$10(A) \times 11(B) + 2$
$= 112$

16 | 112
16 | 7 → 0 ↑
0 → 7

70 (c: 7, s: 0)

$2 \times 11(B) + 7$
$= 29$

16 | 29
16 | 1 → 13 (D)
0 → 1

1D (c: 1, s: D)

(Any Base) Base - R Subtraction

Decimal:

$$\begin{array}{rcccc} & & ^{+10} & ^{+10} & \\ & 1 & 2 & 0 & 5 \\ - & & ^{+1}8 & ^{+1}9 & 2 \\ \hline & 0 & 3 & 1 & 3 \end{array}$$

We add + Base

We add + 1.

Binary:

$$\begin{array}{rccccc} & & ^{+2} & ^{+2} & & \\ & 1 & 1 & 0 & 1 & 1 \\ ^{+1}- & & ^{+1}1 & 1 & 1 & 0 \\ \hline & 0 & 1 & 1 & 0 & 1 \end{array}$$

Other Base:

Hexa-Decimal

$$\begin{array}{rccc} & & ^{+16} & \\ & 4 & A & 6 \\ - & 1^{+1} & B & 3 \\ \hline & 2 & F & 3 \end{array}$$

$16 + 10(A) = 26$

$$\begin{array}{r} 26 \\ 11 \\ \hline 15\ (F) \end{array}$$

Base 6

$$\begin{array}{rcc} & & ^{+6} \\ & 5 & 4 \\ - & 3^{+1} & 5 \\ \hline & 1 & 5 \end{array}$$