

## Harman Outline 1A CENG 5131

### Numbers Real and Imaginary PDF

In Chapter 2, concentrate on 2.2 (MATLAB Numbers), 2.3 (Complex Numbers).

A. On  $\mathbf{R}$ , the distance of any real number from the origin is the magnitude, or *absolute value*, of the number. The absolute value is defined as

$$|x| = \begin{cases} x, & \text{if } x \geq 0, \\ -x, & \text{if } x < 0. \end{cases}$$

The absolute value is useful to define ranges of a variable that extends from negative to positive values. Thus, using the definition, the notation  $|t| \leq a$  states that

$$0 \leq t \leq a \text{ when } t > 0, \quad \text{but} \quad 0 \leq -t \leq a \text{ when } t < 0.$$

Stated in another manner, the statement  $|t| \leq a$  means that  $-a \leq t \leq a$ , and the real variable  $t$  is defined on the interval  $[-a, a]$  including the end points  $-a$  and  $a$ .

**Example** What is the interval on which  $|t + 2| \leq 5$  is defined? The statement  $|t + 2| \leq 5$  means  $-5 \leq t + 2 \leq 5$ , which becomes  $-7 \leq t \leq 3$ . So  $t$  is defined on the interval  $[-7, 3]$  in this case.

## B. Computer Numbers (Harman P 49 see Equations 2.2 and 2.3)

In modern computer systems, numbers are represented in the *binary* number system. The digits are 0 and 1 and the computer can store a finite number of binary digits to represent any number. For integers, the decimal value of a binary number is

$$N = \sum_{i=0}^{m-1} d_i 2^i = d_0 + d_1 \times 2 + d_2 \times 2^2 + \cdots + d_{m-1} \times 2^{m-1}, \quad (1)$$

where  $m$  is the number of digits and the  $d_i$  are binary digits.

The number range for a binary number with  $m$  digits is from

$$(000 \dots 0000)_2 \quad \text{to} \quad (11111 \dots 1111)_2,$$

where the notation  $(N)_2$  means that  $N$  is a number in base 2.<sup>1</sup> The most positive value sums to the decimal value  $2^m - 1$ . An 8-digit positive binary number thus has the maximum value of  $2^8 - 1$ , or 255. The number  $1011_2$  has the decimal value 11 using the expansion of Equation 1.

The binary number representation just discussed assumes that the binary point was located in a fixed position, yielding either an integer or a fraction as the interpretation of the internal machine representation. Unfortunately, the binary point is not actually stored with the number, but its position must be remembered by the programmer (or the program) to display the results in the usual form. This method of representation is called *fixed point*.

---

<sup>1</sup>In the text, numbers other than decimal numbers will have the base specified explicitly.

**Examples** between binary and decimal for fractions.

If a fraction is to be represented, a *binary point* is used to separate the integer from the fractional part of the number, just as the decimal point is used for decimal numbers. The binary number  $(0.11111 \dots 1111)_2$  has the decimal value

$$\begin{aligned} N &= \sum_{i=1}^m d_{-i} 2^{-i} = d_{-1} \times 2^{-1} + d_{-2} \times 2^{-2} + \dots + d_{-m} \times 2^{-m} \\ &= \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots + d_{-m} \times 2^{-m}, \end{aligned} \quad (2)$$

since  $d_{-i} = 1$  for  $i = 1, \dots, m$ . For a finite value of  $m$  digits, the sum is  $1 - 2^{-m}$ . With 16 binary digits, the value is

$$1 - 2^{-16} = 0.99998$$

to five places after the decimal.

For 8 bits- Ask MATLAB:

```
>> 1-2^(-8)
```

```
ans = 0.9961
```

The 16-bit base2 fraction can represent about 5 decimal places of fraction. MATLAB calculates its numbers to far greater precision. Thus, a MATLAB calculation for an algorithm may not be valid if the results are to be used on a microprocessor with a 16-bit data length.

The floating-point notation is the counterpart of scientific notation used in computers for number representation. The decimal equivalent is typically

$$N.n = f \times 2^e. \quad (3)$$

Check these floating point MATLAB numbers that define the Range of numbers:

```
>> realmax %Largest positive real number
```

```
ans = 1.797693134862316e+308
```

```
>> realmin %Smallest positive real number
```

```
ans = 2.225073858507201e-308
```

To define the Precision:

```
>> eps %Machine epsilon - EPS, with no arguments, is the distance from 1.0
to the next larger double precision number, that is EPS with no arguments
returns 2(-52), the smallest fraction. (See Harman p 52)
```

```
ans = 2.220446049250313e-016
```

Even with MATLAB's precision, numbers that are not sums of powers of 2 or 1/2 cannot be represented exactly as this example shows:

```
>> .42-0.5+.08 % You would say this is zero!! Not according to MATLAB
```

```
ans = -1.387778780781446e-017 NOT ZERO!
```

However,

```
>> 0.5-1.0+0.5
```

```
ans = 0,
```

since the fractions are  $-1/2 - 2^0 + 1/2$ .

The figures from Smith's book show clearly the problem with using too few digits to represent measured values.

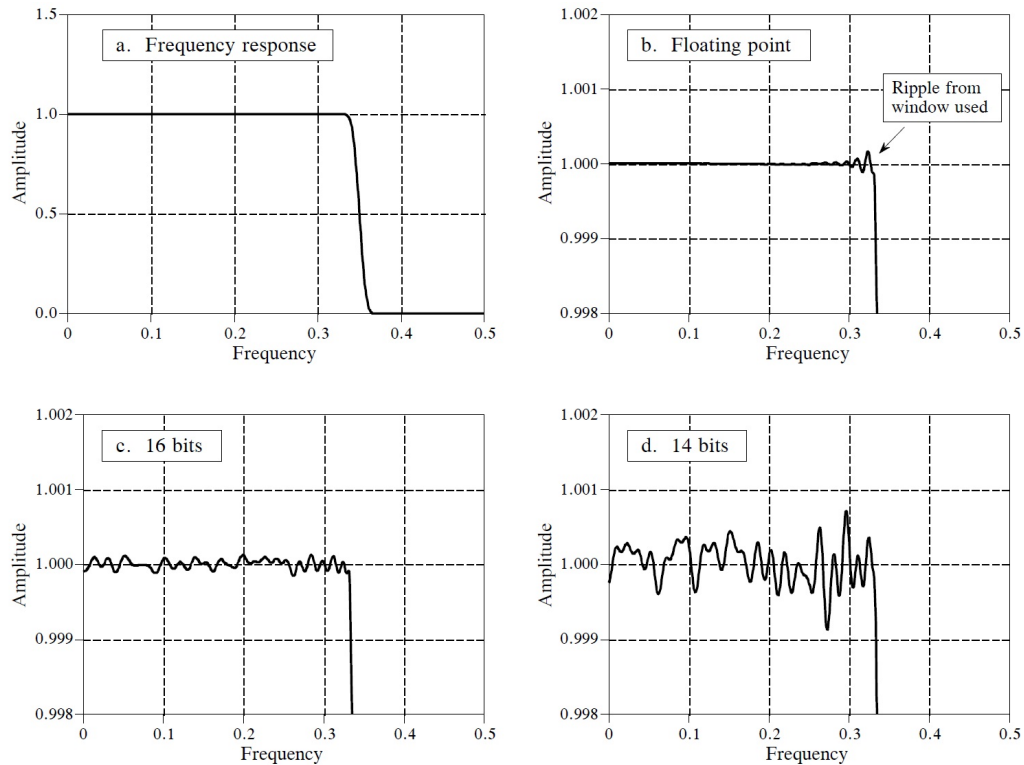


FIGURE 29-5 Round-off noise in the frequency response. Figure (a) shows the frequency response of a windowed-sinc low-pass filter, using a Blackman window and 150 points in the filter kernel. Figures (b), (c), and (d) show a more detailed view of this response by zooming in on the amplitude. When the filter kernel is represented in floating point, (b), the round-off noise is insignificant compared to the imperfections of the windowed-sinc design. As shown in (c) and (d), representing the filter kernel in fixed point makes round-off noise the dominate imperfection.

Figure 1: SmithCh29FixedFloatFilters

### C. Complex Numbers (Harman P 54-60)

Since imaginary values cannot be measured or seen, there may seem little use of complex numbers except for solution of certain mathematical equations such as polynomials with complex roots. This would be an error however since physical meaning can be attached to certain complex values. For real numbers the connection is clear in measurements such as 30 mph or 50 psi.

The number  $x + iy$  yields two pieces of information. It has the following representations

$$x + iy = re^{i\theta} = r(\cos \theta + i \sin \theta) = r\angle\theta$$

according to figure 2.2 and Equation 2.15 and 2.18 (Page 55-57).

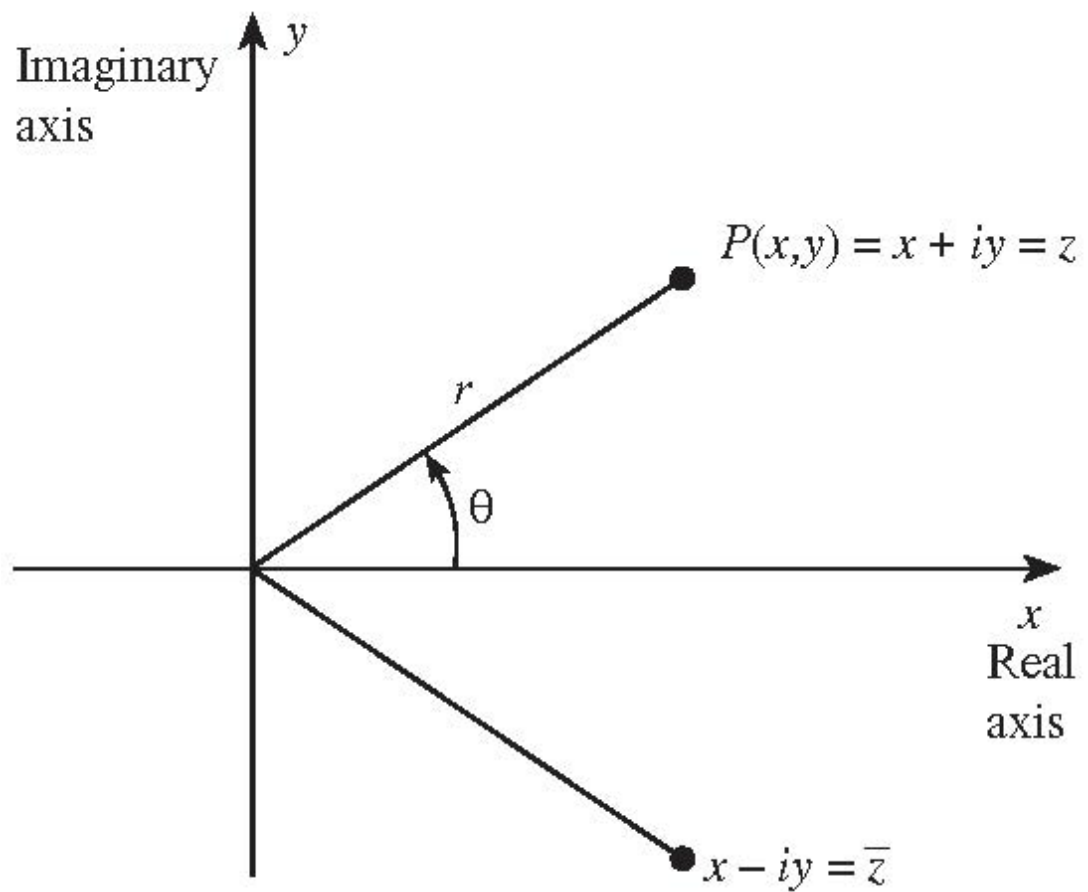


Figure 2: Complex Plane F02-2

**Example - Locus of  $|z| = 1$**  A *locus of points* is the set of points in a plane that are defined by a mathematical equation. For complex numbers, the locus is a 2-dimensional figure in the  $xy$  plane. For example, the locus of points for  $|z| = 1$  is

$$|z| = |x+iy| = \sqrt{x^2 + y^2} = 1 \text{ A circle of radius 1, centered at } (0, 0).$$

**Example Roots of 1** From Euler's formula  $e^{i\theta} = \cos \theta + i \sin \theta$  and DeMoivre's theorem

$$(\cos \theta + i \sin \theta)^N = \cos N\theta + i \sin N\theta$$

we can solve  $z^N = a$ . Let  $a = 1$ . For this complex number, the argument  $\theta$  is zero and the magnitude is 1 so we can write

$$z = a^{1/N} = e^{ip \ 2\pi/N} = \cos \frac{p \ 2\pi}{N} + i \sin \frac{p \ 2\pi}{N}$$

with the property that  $|a| = 1$  so that every one of these complex numbers lies on the unit circle in the complex plane.

Consider the values  $(p/N)2\pi$  for integers  $p = 0, 1, 2, 3$  when  $N = 4$ . Writing  $z = e^\alpha$  where  $\alpha = 2\pi p/4$ , these values correspond to the angles

$$\alpha = 0 \ (2\pi), \ \alpha = \pi/2, \ \alpha = \pi \ \text{and} \ \alpha = 3\pi/2$$

in the complex plane, respectively. The complex values allowed for  $z$  are then

$$\begin{aligned} z_0 &= e^0 &= 1 &= 1 \angle 0 \\ z_1 &= e^{i\pi/2} &= i &= 1 \angle 90 \\ z_2 &= e^{i\pi} &= -1 &= 1 \angle 180 \\ z_3 &= e^{i3\pi/2} &= -i &= 1 \angle 270. \end{aligned} \tag{4}$$

Note that the sum of the roots will be zero. This is true in general for the equations of the form  $z^N + a - 0 = 0$



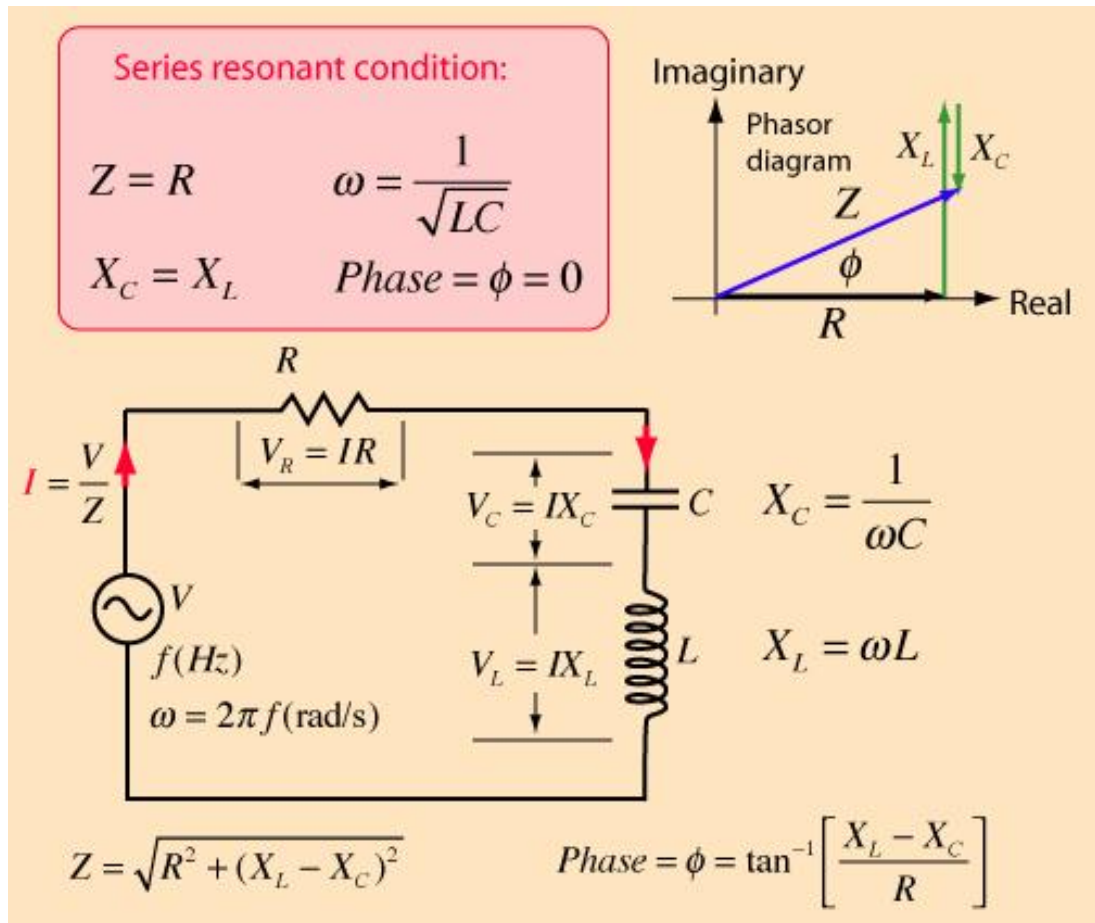


Figure 3: RLC Circuit

### Example of circuits

Consider the RLC circuit shown

Using Kirchhoff's voltage law and the laws for resistors, capacitors, and inductors,

$$v(t) = R i(t) + \frac{1}{C} \int i(t) dt + L \frac{di(t)}{dt} \quad (5)$$

If  $i(t) = I_m \cos(\omega t)$ , then substitute in Equation 5 we find

$$v(t) = RI_m \cos(\omega t) + \frac{I_m}{\omega C} \sin(\omega t) - \omega LI_m \sin(\omega t) \quad (6)$$

Consider the RLC circuit in the figure and the *algebraic* Equation 6 that results from a sinusoidal input to the circuit. Thus,

$$v(t) = RI_m \cos(\omega t) + \frac{I_m}{\omega C} \sin(\omega t) - \omega LI_m \sin(\omega t)$$

If the measured current is

$$i(t) = I_m \cos \omega t,$$

substituting  $i(t)$  and forming the result, the voltage is given by Equation 6. Here  $\omega = 2\pi f$ , the radian frequency.

Let's use  $j = \sqrt{-1}$  to avoid confusion with current.

Noting that  $I_m \cos \omega t = \text{Re}\{e^{j\omega t}\}$ , substitute into the differential equation to find

$$V = I_m [R \times \text{Re}\{e^{j\omega t}\} + \frac{1}{C} \times \text{Re}\{\frac{1}{j\omega} e^{j\omega t}\}] + L \times \text{Re}\{j\omega e^{j\omega t}\} \quad (7)$$

Using Euler's formula, the result is the same as before in Equation 6. If the current had been  $i(t) = I_m \sin \omega t$ , taking the imaginary part of the result in Equation 7 would yield the current.

For the special case of a sinusoidal input to a simple system described by a linear differential equation with constant coefficients, the result will always be a sinusoidal of the same frequency. If the current in such a series circuit is  $I_m \cos \omega t$ , the voltage across the circuit will be

$$v(t) = V_m \cos(\omega t + \theta)$$

where  $V_m$  is the *magnitude* of the voltage and  $\theta$  is the *phase* with respect to the input sinusoid. The voltage across the resistor is in phase with the current but the voltages across the inductor and capacitor are shifted by  $90^\circ$  from the current wave. Notice that  $\pm \sin \omega t = \cos(\omega t \mp \pi/2)$ , since

$$\cos(a + b) = \cos a \cos b - \sin a \sin b.$$

If we associate the value

$$j = e^{j\pi/2}$$

the imaginary value  $j$  represents a  $90^\circ$  shift. For the RLC circuit with sinusoidal input, the *impedance* relates voltage and current at the input terminals as

$$\mathbf{Z} = \frac{\mathbf{V}}{\mathbf{I}} = R + j\omega L + \frac{1}{j\omega C} \quad (8)$$

The table lists useful MATLAB commands for complex numbers (See Harman p 59).

<i>Command</i>	<i>Format</i>
<b>z=x+yi, z=x+yj</b>	Complex number
<b>z=r*exp(i*theta)</b>	Polar form
<b>abs</b>	Magnitude $ z  = \sqrt{x^2 + y^2}$
<b>angle</b>	Angle in radians $(-\pi, \pi)$ ; $\theta = \tan^{-1}(y/x)$
<b>conj</b>	Complex conjugate $x - yi$
<b>imag</b>	Complex imaginary part $y$
<b>real</b>	Complex real part $x$
<i>Plotting:</i>	
<b>compass</b>	Draws complex numbers as arrows on polar plot
<b>feather</b>	Draws complex numbers as arrows on linear plot

Table 1: Table of MATLAB Commands for Complex Numbers