EELE 461/561 – Digital System Design

Module #1 – Digital Signaling

• Topics
  1. Course Overview
  2. Signaling Definitions
  3. Signal Composition

• Textbook Reading Assignments
  1. 1.1 – 1.7, 2.1, 2.10

• What you should be able to do after this module
  1. Describe what signal integrity is and why it is important
  2. Understand the terminology used in digital signaling
  3. Describe and use the risetime-bandwidth product
  4. Describe the frequency components of a digital signal

Course Overview

• Instructor: Brock J. LaMeres
  Office: 533 Cobleigh Hall
  Phone: (406) - 994 - 5987
  Email: lameres@ece.montana.edu
  Web: www.coe.montana.edu/ee/lameres/

• Time / Location:
  Lecture: Monday, Wednesday, Friday
  9:00am – 9:50am
  110 EPS

• Textbook:
  "Signal & Power Integrity Simplified", Eric Bogatin,
  Prentice Hall, 2nd edition 2009

• Website: www.coe.montana.edu/ee/lameres/courses/eele461_spring12
  - all handouts and homework are found on the website
  - it is your responsibility to download assignments

Course Overview

• Office Hours: Check instructor website for most recent hours

• Requisites:
  - Pre-requisite EE308, EE334, EE371 (or consent of instructor)

• Grading:
  - Homework - 25%
  - Exam #1 - 25%
  - Exam #2 - 25%
  - Final Project - 25%

  - Homework Assignments are due at the beginning of class on indicated date.
  - Late homework will be accepted for one week after the due date with a penalty of 50% point reduction.
  - No credit will be given for assignments over one week late.
  - No make-up exams will be given. Plan on being available on the exam dates.

Course Content

• What is this course?
  - We will look at how to design and analyze digital communication links in a wireline medium
    (i.e., conducting wires vs. wireless)
  - A communication link is the circuitry (Tx, Rx, and interconnect) used to transfer information between
    logical blocks
    - ex: up to memory system or periphery
      computer-to-computer networking
  - We will look at the analog effects of a digital signal in order to understand how to design
    chip-to-chip communication links
  - We will learn how to create a noise budget that considers voltage and timing noise
  - We will see that the physical interconnect between IC’s tends to limit the speed at which
    data can be transferred
  - We will also see that at modern integrated circuit speeds, interconnect needs to be treated as
    a transmission line (as opposed to just a simple capacitance)
  - We will learn to use modern CAD tools to help design and analyze these links

Course Content

• What topics will be covered?
  1) Signaling (Exam #1 Topics)
  2) Interconnect Analysis
  3) Interconnect Fabrication and Modeling
  4) Noise Sources & Budgeting
  5) Power Distribution
  6) Link Architectures
  7) Measurement Techniques (Exam #2 Topics)
  8) Modern Bus Architectures
  9) Design Trade-offs

Course Overview
**Course Content**

**Why do we need this course?**

- We create all of our digital circuits using integrated circuit technology.
- In order for two digital circuits to communicate information, the transmitter (Tx) sends a signal to the receiver (Rx).
- As long as the DC specifications are met at the Rx, the receiver can determine whether a 1 or 0 was sent.
- As along as sufficient time is allowed before and after the associated clock, the Rx flip-flops can latch and store the information that was sent.

**Course Content**

**Why do we need this course?**

- So now all we need to do is meet DC and Timing specifications, and then go faster...
- In the beginning, integrated circuit performance was the limiting factor in going faster.

**Course Content**

**Why do we need this course?**

- We used to only care about the delay associated with the IC gates.
- On-chip and off-chip interconnect were modeled as capacitances, but they were secondary effects.
- As we moved beyond 1um process capability, the on-chip interconnect began to contribute more delay to the circuit than the gate itself.
- Also at this time, the rise times of the IC drivers became fast enough so that off-chip interconnect (i.e., PCB’s and cables) needed to be modeled as transmission lines.
- The mismatch in IC circuit performance and off-chip interconnect performance has to do with the manufacturing processes used (IC=photolithography/chemistry, Interconnect=chemistry/mechanical).

**Course Content**

**Why do we need this course?**

- Now, modern digital system speed is dominated by the interconnect between the integrated circuits making the Tx and Rx.
- The interconnect must be modeled not as simple wires, but as distributed capacitances, inductances, and impedances.
- This modeling also applies to the interconnect of the power distribution (i.e., VDD and GND).
- We now have to consider the analog behavior of the signal as it travels through the interconnect to understand how fast and how robust a digital system can be.
- This course will present the techniques to design, model, analyze, and measure a modern digital system at the physical level.
- This type of design and analysis is also called "Signal Integrity".

**Course Content**

**Signal Integrity**

- Signal integrity consists of 3 categories.
  1. Voltage Noise
     - sometimes just called Noise
     - what factors in the system eat into our Voltage Noise Margin
  2. Timing Noise
     - also called Jitter
     - the factors in the system eat into our Timing Margin
  3. Electromagnetic Interference (EM)
     - when our system creates unwanted energy that interferes with the standards set by the Federal Communication Commission (FCC)

**Signaling**

**Digital Signaling**

**Voltage Noise**

- what we’re really after is a stable region between VDD and VSS
Signalizing

• Digital Signaling
  Voltage Noise
  - this area of Signal Integrity can further be sub-categorized into 4 distinct sources of noise:
    1) Single Net Quality
    2) Cross-talk Noise
    3) Power Supply Quality
    4) EMI

Timing Noise
  - we need to ensure that the data is stable long enough to meet the setup/hold specification of the Rx

Timing Noise
  - the entire timing of a signal can be broken down into 3 distinct parts:
    1) Logical timing
      - i.e., the time for the gates to switch on chip
    2) Interconnect Propagation
      - the time we have to wait for the signal to propagate
      - the time that we have to wait for any distortions to settle out
    3) Receiver Setup/Hold
      - the time the data must remain stable around the receiver clock event

Electromagnetic Interference
  - The FCC sets standards for how much Electromagnetic energy can be transmitted in a given frequency band.
  - Some of the bands are licensed (i.e., you have to pay to use them)
  - If you make a product that isn’t intended to use one of the FCC bands, you cannot inadvertently transmit energy into an FCC band above a certain level.
  - This is considered breaking the law and you will not be able to ship your product if it does this.
  - Products undergo EMI testing prior to shipping.
  - As our systems go faster, it becomes easier for the energy to radiate out of our product because our small interconnect begins to look like an antenna

Eye Pattern
  - we can combine the voltage and timing specifications into one diagram called an "Eye Pattern"
Signaling

- **Digital Signaling**
  - Data Valid Window: the stable region in Voltage & Time around a timing event (i.e., a clock) that will guarantee a logic level is received.

- Unit Interval (UI): the time between transitions on the data line.

- Max Data Rate: the fastest we can transmit information on the data line:

\[ \text{Max Data Rate} = \frac{1}{\text{UI}} \]

Signal Composition

- **A Digital Link**
  - A digital communication link consists of 3 elements:
    1. A Driver (Tx)
    2. Interconnect
    3. A Receiver (Rx)

  - The Driver and Receiver are constructed using Integrated Circuit Technology (ICs).

  - The Interconnect consists of all wires that connect the Tx and Rx. These include:
    - On-chip traces, package leads, PCB traces, connectors, and cables.

- **Driver Risetime**
  - The speed of the driver risetime is important because it tells us how fast we can possibly switch logic states (i.e., how fast a frequency can be generated).

  - If the IC driver is NOT connected to any interconnect (on-chip or package), then the rise time will follow an RC exponential ramp where the R and C come from the Tx transistors.

  - The driver produces a digital edge.
  - We define the risetime \( t_{\text{rise}} \) as the time it takes for the edge to go from 10% of steady state to 90% of steady state.

  - If we try to switch too fast, there isn’t enough time for the rise time to get to its final value. This leads to data loss.

  - The relative speed of the risetime to Period (T) is arbitrary, we can use a rule of thumb that it needs to be ~10% of the Fastest Period.

  - We’ll see how this is derived when we look at the Fourier Transform of a Square Wave:

\[ \text{rise} \frac{t}{T} \approx \frac{1}{10} \text{rise} \text{min} \text{max} \]
Let's Solve the RC circuit and derive the expressions for risetime and bandwidth.

We are interested in what our forcing function looks like in the Frequency Domain because it becomes easier to observe and some things are easier to work with.

We can think of this as a Linear Time Invariant (LTI) System where we have:

1. A forcing function, \( x(t) \) (i.e., the driver edge and pattern)
2. A transfer function, \( h(t) \) (i.e., the response of the interconnect)
3. The system output, \( y(t) \) (i.e., what is seen at the Rx)

Note that the driver rise time is NOT always what we get at the Receiver end of the interconnect – run longer than necessary.

We want sufficient bandwidth for accuracy, but not too much that our simulations take more simulation time to compute. This helps us scale the cost of our materials and construction for the application.

Let's set an exponential approach.

One of the main operator transforms that we take advantage of is convolution in the time domain becomes multiplication in the frequency domain.

\[
V_{in}(s) = \frac{1}{s + \frac{1}{RC}}
\]

\[
V_{out}(s) = \frac{1}{s + \frac{1}{RC}}
\]

\[
v(t) = x(t) * h(t)
\]

\[
y(t) = X(s)H(s)
\]
**Signal Composition**

- **Risetime and Bandwidth**
  - Let's now solve for Bandwidth in the Frequency Domain
  
  We go back to an earlier form of the RC solution:
  
  \[ V_{out}(s) = V_{in}(s) \left( \frac{1}{1 + RCs} \right) \]
  
  We are after the frequency at which the Magnitude of \( V_{out}/V_{in} \) is equal to 0.7.
  
  In a logarithmic scale, this attenuation can also be described as where the Amplitude is 3dB less than the Amplitude at DC.
  
  We can use 1/2 to make the math come out easier since \( \sqrt{2} \approx 0.7 \).

**Signal Composition**

- **Risetime and Bandwidth**
  - We can relate the risetime to the bandwidth in what is known as the "Risetime Bandwidth Product"
  
  \[ t_{rise} = 2.2 \cdot BW \]
  
  \[ BW = \frac{1}{2 \pi f_{3dB}} \]

**Signal Composition**

- **Risetime and Bandwidth**
  - Now we can transform into the time domain:
  
  \[ V_{out}(s) = (1 - e^{-s}) \cdot u(t) \]
  
  \[ V_{out}(t) = (1 - e^{-t}) \cdot u(t) \]

  Now we can plug in \( u(t-1) \) for \( t=0 \)

  This is the generic Time Domain solution for an exponential ramp (i.e., a rising edge)

  \[ RC \cdot \alpha = \frac{1}{s} \]
  
  \[ RC \cdot s = \frac{1}{\alpha} \]

  Then, the risetime in the time domain is:

  \[ t_{rise} = (2.2) \cdot RC \cdot (1) \cdot RC \]

  \[ t_{rise} = 2.2 \cdot RC \]

Ex: how much spectral energy is created by a driver with a 1ns step?

\[ f_{3dB} = \frac{1}{2 \pi} \]

\[ BW = \frac{1}{2 \pi f_{3dB}} \]

\[ t_{rise} \cdot BW = 0.35 \]

\[ (t_{rise}) \cdot BW = 0.35 \]

\[ BW = \frac{0.35}{1 ns} \]

\[ BW = 350 MHz \]
Signal Composition

- **Risetime Bandwidth Product**
  - This expression also illustrates that as the risetime gets faster, higher frequencies are generated which makes sense:
    \[ t_{\text{rise}} \cdot BW = 0.35 \]

- **System Risetime & Bandwidth**
  - We found a very powerful and quick way to convert between a Time Domain metric (risetime) and a Frequency Domain metric (BW) using the Risetime Bandwidth Product:
    \[ t_{\text{rise}} \cdot BW = 0.35 \]

Note that this rule-of-thumb is used on a single element (i.e., a single driver, a single interconnect network, etc.).

- Remember that BW is when the sine wave at f_{3dB} is reduced to 70% of its value at steady state.

- Let's see how we'd use this:
  - ex) We have a digital link with a driver outputting a risetime of 300ps. The interconnect system looks like an RC network with bandwidth of 2GHz. What is the risetime we would expect to see at the receiver?
    - first convert the interconnect bandwidth to a risetime using the Risetime Bandwidth Product:
      \[ t_{\text{rise}} = \frac{BW}{2} = \frac{0.35}{2} = 0.175 \text{ ns} \]
    - next use the RMS sum to find the equivalent (or composite) risetime of the system:
      \[ t_{\text{sys}} = t_{\text{rise}} \sqrt{2} = 0.347 \text{ ns} \]
Signal Composition

• Square Waves
  - Let’s now look at the composition of a square wave.
  - We are interested in the Frequency Domain representation of square waves because this is what a digital driver will ultimately output a data pattern which will have a square wave spectrum.
  - We can visualize how much bandwidth a link will need to transmit a certain data pattern by looking at the data patterns spectrum.
  - For now, let’s assume that the risetime is instantaneous (we’ll incorporate real risetimes at the end).

Signal Composition

• Frequency Domain Basics
  - The Frequency Domain is completely made up by engineers and scientists. The only “Real” domain is the Time Domain.
  - We create the Frequency Domain in order to visualize problems in a more intuitive manner.
  - Also, some mathematical operations are easier in the Frequency Domain.
  - The Frequency Domain only contains Sine Waves

• Fourier Analysis tells us that any repetitive waveform in the Time Domain can be represented by one or more sine waves in the Frequency Domain.

- DC offset information for a waveform is represented in the Frequency Domain by placing magnitude information at 0 Hz.

- Fourier Analysis tells us that any repetitive waveform in the Time Domain can be represented by one or more sine waves in the Frequency Domain.

- We can transform back and forth between the domains using the Fourier Transform.
  - The Fourier Integral will transform any repetitive function:
    - Time to Frequency Domain
    - Frequency to Time Domain

Discrete Fourier Transform (DFT)
- In the real world, we perform this transformation on discrete points of the function.
- The number of points is fixed prior to the transform.
- The pattern is assumed to repeat indefinitely every T seconds.
- This is called a Discrete Fourier Transform (DFT).

Fast Fourier Transform (FFT)
- A special case of the DFT is when the number of points is a power of 2 (256, 512, 1024, ...).
- This special case enables the Matrix Math to be sped up considerably.
- This is called a Fast Fourier Transform (FFT) and is the most common algorithm used today.
- The FFT can be 100-10,000 times faster than a general DFT.

Signal Composition

• Fourier Transform of Square Waves
  - One way to construct the Fourier Transform of a Square Wave is to construct the Square Wave using multiple, basic functions.
  - The Fourier Transform not only allows functions to be transformed, but also operations.
  - This means we can construct a complex expression for a waveform, and then transform the individual parts and operations.
  - There are two common operations that we use when looking at Square Waves

<table>
<thead>
<tr>
<th>Operation</th>
<th>Time Domain</th>
<th>Frequency Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Scaling”</td>
<td>$f(at)$</td>
<td>$\frac{1}{</td>
</tr>
<tr>
<td>“Convolution”</td>
<td>$f(t) \ast g(t)$</td>
<td>$F(f) \cdot G(f)$</td>
</tr>
</tbody>
</table>
Signal Composition

• Fourier Transform of Basic Functions

Rectangle Function \( \Pi(t) \)
- A Rectangle Function is a pulse with unit width and height.
- We can use this to represent a single pulse of a square wave with a finite pulse width \( W \).
- This transforms into the \( \text{sinc} \) function with zero crossings at integer multiples of \( 1/W \).

\[
\text{Function: } \Pi(t) \quad \text{sinc}(f)
\]

Signal Composition

• Fourier Transform of Basic Functions

Shaw Function \( \delta(t) \)
- A Shaw Function is an infinite series of impulse functions.
- This transforms into another Shaw function in the Frequency Domain

\[
\text{Functions: } \Pi(t) \quad \delta(f)
\]

Signal Composition

• Fourier Transform of Basic Functions

Square Wave
- In the Time Domain, if we convolve a Rectangle Function with a Shaw Function, we will get a repetitive sequence of pulses.
- If \( W \) is set to \( T/2 \), then we will have a 50% duty cycle square wave (a special case).

\[
\text{If } W = T/2 \quad \text{then} \quad \text{50% duty cycle square wave}
\]

Signal Composition

• Fourier Composition of a Square Wave

- Notice that when we have a 50% duty cycle, the \( \text{sinc} \) envelope eliminates all even harmonics.
- We are left with only the DC offset, the Fundamental Frequency, and Odd Harmonics.
- This is what most people are familiar with when they talk about the Fourier spectrum of a square wave.
- However, this illustrates that if we alter the duty cycle, we will start getting even harmonics due to the \( \text{sinc} \) function spreading out.

\[
\text{If } W = T/2 \quad \text{then} \quad \text{50% duty cycle square wave}
\]
Signal Composition

- Fourier Composition of a Square Wave
- We now have a transform for an ideal square wave.

Ex) a 1V square wave (V_LOW = -0.5V, V_HIGH = +0.5V) will be made up of sine waves with Amplitudes:

<table>
<thead>
<tr>
<th>Harmonic</th>
<th>Amplitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fundamental</td>
<td>0.637 V</td>
</tr>
<tr>
<td>3rd Harmonic</td>
<td>0.212 V</td>
</tr>
<tr>
<td>5th Harmonic</td>
<td>0.127 V</td>
</tr>
<tr>
<td>7th Harmonic</td>
<td>0.091 V</td>
</tr>
<tr>
<td>9th Harmonic</td>
<td>0.071 V</td>
</tr>
<tr>
<td>11th Harmonic</td>
<td>0.058 V</td>
</tr>
</tbody>
</table>

Notice that as we add more harmonics to the fundamental, we get a waveform that looks more and more like an ideal square wave.

- If we add all the harmonics (up to infinity), we will get a perfect square wave with instantaneous rise times.
Signal Composition

- **Fourier Composition of a Square Wave**
  - However, we know that no system can output or transmit infinite frequency.
  - The big question for digital system designers becomes…
    - How many harmonics do I need to get a square wave that is good enough?

- **Fundamental + 3rd harmonic**
  - This is an arbitrary question and depends on how fast of a risetime each system needs.
  - But, we can tie this question back to one of our rules relating the risetime to the period of the square wave.
    - Earlier we stated that the risetime should be ~10% of the period.

\[
T_{\text{rise}} \leq (0.1) \cdot T_{\text{period}}
\]

- **Square Wave Composition**
  - Let's start with only the fundamental frequency and see what percentage of the period that the risetime takes.
  - Then we'll add harmonics and see how it changes:
    - \( \% \text{ of } T_{\text{period}} = \frac{\text{rise time}}{T_{\text{period}}} \times 100 \)

<table>
<thead>
<tr>
<th>Harmonic Combinations</th>
<th>% of Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fundamental</td>
<td>22%</td>
</tr>
<tr>
<td>Fundamental + 3rd</td>
<td>11%</td>
</tr>
<tr>
<td>Fundamental + 3rd + 5th</td>
<td>5.7%</td>
</tr>
<tr>
<td>Fundamental + 3rd + 5th + 7th</td>
<td>4.4%</td>
</tr>
<tr>
<td>Fundamental + 3rd + 5th + 7th + 9th</td>
<td>3.6%</td>
</tr>
</tbody>
</table>

- **Spectral Content**
  - If we look at the risetime bandwidth product of a Square Wave made up of a Fourier Series of sine waves, we get:

\[
\text{Risetime Bandwidth Product} = \frac{\text{rise time}}{T_{\text{period}}} 
\]

- **Risetime Bandwidth Product**
  - This shows that for different shaped risetimes, we can simply change the risetime BW product to get more accurate results.
    - RC step, use 0.35
    - Gaussian step, use 0.4
  - But remember, this is an approximation to get a gut feel. So we don't need to get too hung up on the exact number as long as it is around 0.35.
  - One more nice thing is that the Fourier Series Representation has a Gaussian Distribution by nature, which means it follows the Central Limit Theorem.
  - That means that even if we choose to use 0.4 in our risetime BW product, we can still use a sum-of-squares expression to describe the composite risetime.
Signal Composition

- Fourier Composition of a Square Wave
  - Once again, remember that we are talking about the stimulated energy of the driver.
  - What gets to the Receiver in our digital link is another story.
  - At first glance, we think this is pretty straightforward because an interconnect system will simply attenuate the forcing function’s harmonics.
  - In reality, when we look at the distributed nature of an interconnect system, we see that reflections can actually cause the harmonics to be amplified!!
  - This makes the analysis a little more interesting.
  - Next, we look how to model the interconnect in order to understand how its response will affect the wave shape of the forcing function.