Universal Serial Bus Signal Integrity Analysis for High-Speed Signaling

Todd West and Vikram Jandhyala  
{jandhyala}@ee.washington.edu

Dept of EE, University of Washington  
Seattle WA, 98195-2500
Universal Serial Bus Signal Integrity Analysis for High-Speed Signaling

Todd West and Vikram Jandhyala
{jandhyal}@ee.washington.edu

Dept of EE, University of Washington at Seattle
Seattle WA, 98195-2500

University of Washington, Dept. of EE, UWEETR-2001-0009
Dec 11, 2001

Abstract

The Universal Serial Bus transmits differential data over twisted-pair cables originally designed for signaling at 12Mbit/s. Because of the recent specification of a 480Mbit/s signaling speed with the existing cabling, the high frequency behavior and temporal response of these cables is of interest. In this work, two- and three- dimensional field solvers are used to characterize the field behavior within the cables. The results are used to perform SPICE simulations of complete time-domain USB packets, which indicate that existing cables support the higher bit rates well, and that the quality of signaling is not limited by the cable construction. The simulations match well with the limited measurement data available for 480Mbit/s signaling.

1. Universal Serial Bus Cables

The Universal Serial Bus (USB) is a general-purpose technology for connecting peripherals to personal computers. A twisted pair carries differential signals at signal rates of 1.5, 12, and 480 Mbits/s. Additionally, transitions on one member of the pair are used to signify ends of packets at all signal rates and starts of packets at 480 Mbits/s. The presence of both common- and differential-mode signals makes impedance control involved. The presence of a power pair, which eliminates the need for power supplies for USB peripherals, further complicates the situation [1]. Figure 1 shows the USB cross section. The six conductors i.e. signal lines D+ and D-, power pair Vbus and ground, drain, and shield are shown, along with the insulations polyvinyl chloride (PVC) and high-density polyethylene (HDPE). The outer dimension of the shield is approximately 3mm. The power pair is 20 American wire gauge (AWG), the signal pair is 20-28 AWG, and the drain wire is 28 AWG. The signal lines D+ and D- can be in an unbalanced configuration, owing to their twisted pair rotation along the cable.

USB cabling has been shipping in high volume since 1998, and hence it is important to estimate signal integrity in existing USB cables, developed for 1.5 Mbit/s and 12 Mbit/s signaling in 1995, for the new specification of 480 Mbit/s adopted in 2000 [2]. This necessitates the use of electromagnetic and SPICE solvers. Not many signal integrity results for USB systems are available in the public literature, and it is expected that the simulation approach and results presented here will enable the signal integrity analysis of other existing and future high-speed hardware connector schemes [3].
2. Electromagnetic and Circuit Simulation of USB Cables


Ansoft’s two-dimensional transmission line solver was used to compute differential- and common-mode impedances as a function of frequency. Two models, a simplified single-conductor model, and a more complex stranded conductor model, were used. Figure 2 shows the impedances for the two models in a balanced configuration. Similar simulations were run for rotated signal pairs to obtain average impedances. Also of interest is the propagation loss and velocity in the cable. This is shown for two cable rotations and for both models in Figure 3, along with the conservative USB specification limit, which is easily achieved by the cable model. The propagation velocity was modeled using an average of velocities of eigenmodes weighted by the relative strength of the eigenmode in a typical USB signal packet. The power spectral density of a USB signal has been studied in the past [6]. The resulting velocity is plotted in Figure 5, and is close to measured velocities that are observed to be near 2.1 X 10^8 m/s.

For a three-dimensional analysis of the USB cable, Ansoft’s Eminence software was used to simulate small sections of the cable including signal pair rotations that were linked together to obtain a SPICE-level representation. Figure 6 shows USB signal outputs from the simulation. Arrows 1 and 2 show transition errors that occur because the differential parts of the single ended transitions start before the slower common-mode components arrive. The differential power drives a voltage into the victim line until the common-mode signal arrives to correct the output. Another effect, shown by Arrow 3, is the upward shift of the first few byte of the packet, which occurs due to an impedance mismatch between common-mode terminations and the cable’s common-mode impedance. These effects are also visible in a measured USB output signal, as shown in Figure 7.

3. Conclusions

The USB simulation based on electromagnetic models and SPICE simulation was able to predict the time-domain effects present in measured data. Also, the cable model studied was able to function with 480 Mbit/s signaling and this is an indicator that existing cabling can handle the faster bit rate. More extensive simulation would require more accurate cable models, and full-wave three-dimensional electromagnetic simulation of an electrically long cable. It is expected that similar simulations will benefit the signal integrity studies of other high-speed connection schemes and will enhance public domain information.
References


Figure Captions

Figure 1: Schematic USB cross section, showing the six conductors and insulation. As shown by the arrows, the signal pair can be in a rotated position owing to the twisted-pair cable manufacturing process.

Figure 2: Common mode and differential impedances for the stranded multi-conductor and simple models, as a function of frequency.

Figure 3: Frequency dependence of common-mode and differential-mode propagation losses for the multi-stranded and simple models, for two signal pair rotation angles, balanced (0), and extreme unbalanced (90). Also shown is the USB specification limit.

Figure 4: Frequency dependence of common-mode and differential-mode propagation velocities for the multi-stranded and simple models, for two signal pair rotation angles, balanced (0), and extreme unbalanced (90).

Figure 5: A simulated output USB signal, showing the two differential signals and the common mode signal. The numbered arrows depict different common-mode effects.

Figure 6: A measured output USB signal, showing the two differential signals and the common mode signal. The numbered arrows depict different common-mode effects.
Figure 1: Schematic USB cross section, showing the six conductors and insulation. As shown by the arrows, the signal pair can be in a rotated position owing to the cable manufacturing process

T.J. West and V. Jandhyala
Figure 2: Common mode and differential impedances for the stranded multi-conductor and simple models, as a function of frequency.

T.J. West and V. Jandhyala
Figure 3: Frequency dependence of common-mode and differential-mode propagation losses for the multi-stranded and simple models, for two signal pair rotation angles, balanced (0), and extreme unbalanced (90). Also shown is the USB specification limit.
Figure 4: Frequency dependence of common-mode and differential-mode propagation velocities for the multi-stranded and simple models, for two signal pair rotation angles, balanced (0), and extreme unbalanced (90).

T.J. West and V. Jandhyala
Figure 5: A simulated output USB signal, showing the two differential signals and the common mode signal. The numbered arrows depict different common-mode effects.
Appendix

A Study of Universal Serial Bus Signal Integrity
Todd West, 2000.8.7
EE600B thesis project
Supervisory Committee: Professors Jandhyala (advisor), Allstot, and Sahr
Electrical Engineering Department, University of Washington

Abstract

The Universal Serial Bus transmits differential data over twisted-pair cables designed for signaling at 12Mbit/s. Because of their construction, impedance control within these cables is difficult and has become a subject of some concern with the recent specification of a 480Mbit/s signaling speed that’s expected to operate using existing 12Mbit/s cabling. Two and three dimensional electromagnetic field solvers were used to characterize the field behavior within the cables and the results were used to perform Spice simulations of complete USB packets. Both the field solver and Spice results agree well with experimental measurements and the Spice simulations indicate the cables support 480Mbit/s signaling quite well and that the quality of the signaling is limited by the connectors, boards, and chip packages needed to complete the signaling connection, not the cable construction. The simulations match well with limited measurement data.

T.J. West and V. Jandhyala

Figure 6: A measured output USB signal, showing the two differential signals and the common mode signal. The numbered arrows depict different common mode effects.
Introduction

Increasingly, electromagnetic field solvers are required to capture the behavior of electrical structures like transmission lines, discrete components, and chip packages. But, because of their high computational requirements, it’s impractical to use field solvers to simulate structures’ behavior at the circuit or port level. While it’s increasingly common to reduce the field solver results to a simpler circuit model that can be simulated by one of the many variants of Spice (or a comparable tool, such as Ansoft Serenade) there is no standard technique for performing this simplification.

Since I’ll likely be intimately involved with this simulation pipeline in the future, I chose to devote the conclusion of my Master’s degree to exploring it in detail by way of Universal Serial Bus (USB). USB is an Intel backed, general purpose technology for connecting external peripherals like mice, keyboards, printers, scanners, and whatnot to PCs through a cable with a single, half duplex twisted pair.

USB cabling is unusual. While the twisted pair usually carries differential signals, signal rates of 1.5, 12, and 480Mbit/s are used. Each of these three speeds occasionally uses unbalanced signals, and the unbalanced signals in each speed have their own quirks. Figure 1 shows typical signaling at 12Mbit/s (left plot) and 480Mbit/s (right plot). In the 12Mbit/s packet, the wires in the differential pair—D+ (green) and D- (blue)—undergo single ended, rather than differential, transitions to indicate the end of the packet (EOP), which occurs between 1.4 and 1.6 µs. In the 480Mbit/s packet, the single ended transitions occur at the start (10ns) and end (210ns) of the packet.

---

1 Mbit/s is 1 megabit per second, mega being 10^6, not 2^20.
Single ended transitions are a mixture of differential and common mode signals, so USB cables require a controlled common mode impedance as well as a controlled differential impedance. While these are reasonably easy to achieve with an isolated signal pair, a pair of power wires is enclosed with the differential pair inside of the cable’s shield. The 5V supply available from the power pair eliminates many USB devices’ need for independent power supplies, but the power pair’s presence within the cable substantially complicates control of the signal pair’s impedance.

Figure 3 shows simplified cross sections for a USB cable with the signal pair at 0 and 90° rotations. All six copper conductors in the cable are labeled (drain, 5V/Vbus, ground, D+, D-, and the shield), as are the PVC (polyvinylchloride) and HDPE (high density polyethelene) insulation jackets on the power and signal pairs. In both cross sections, the shield’s outer diameter is 3mm, which is fairly typical for USB cables. The complete cable assembly is somewhat larger in cross section because the shield is enclosed in a PVC jacket that adds about 1.5mm to the diameter. For most purposes, the maximum cable length is 5m, with the majority of cables shipping running in the 1.5–2.5m range.

---

2 There’s no convention for describing the twisted pair’s orientation, so I’ve defined it as the angle of a ray that starts at the middle of D- and passes through the middle of D+. Even this convention doesn’t have much meaning if you use different cable orientation than the scheme shown in Figure 3.

3 The power pair in the cross sections shown is 20 gauge AWG (American wire gauge) and the signal pair is 28 AWG. The signal pair is fixed at 28 gauge, but the power pair can actually be 28, 26, 24, 22, or 20 gauge. I chose to model the 20/28 configuration since it’s the longest USB cables that have the most effect on USB signals. Certain DC resistance specs on the power pair and some de facto standards in the USB cable industry mean nearly all of the long cables worth modeling have 20 gauge power pairs.

4 Devices signaling at 1.5Mbit/s are restricted to 3m cables. The minimum cable length is limited to about 4 inches by mechanical tolerances on the assembling machinery, but the shortest cables shipping in any kind of volume are all around 1m long. Some USB devices, such as USB versions of parallel-port hardware keys, have no cable and just plug directly into a USB port. It’s also becoming increasingly common for PC motherboard manufacturers to put USB devices down on the board and link them with a pair of microstrips.
The signal pair is balanced in the 90° position because the line of symmetry between D+ and D- is the same as the cable’s line of symmetry. As the pair rotates away from 90°, the geometry becomes progressively less balanced, reaching its worst at 0°. Since the signal pair has 180° rotational symmetry, 270° is also a balanced configuration and 180° is also unbalanced. The common mode impedance also changes as the pair rotates since changes in the signal pair’s position change the strength of the coupling between the signal lines and the shield, ground, and drain wire.

Since USB cabling has been shipping in volume since 1998, the immediate question is not how to mitigate variations in the cables’ impedance, but how bad the cables actually are. Cable quality is particularly important because the cable spec was finalized in 1995 for 1.5 and 12Mbit/s signaling rates. The 480Mbit/s signaling spec was set early in 2000, with the expectation that USB cables will handle 480Mbit/s signals, regardless of whether the cables were built before or after the adoption of 480Mbit/s signaling.5

Intel, HP, NEC, Phillips, and others have investigated cable quality experimentally, but industry understanding of electromagnetic field behavior within USB cables remains limited and only simple cable models have been developed for circuit simulators. Therefore, the rest of this paper is devoted to developing and assessing electromagnetic and circuit models for a USB cable of typical performance. Section 0 discusses the electromagnetic models and field solver results, while section 0 describes the development of circuit models and circuit simulation results. Not only does it turn out that the cables are surprisingly good, but the models and simulations agree well with experimental results.

5 This decision was—and remains—the subject of substantial controversy.
Electromagnetic Modeling of USB Cables

There are many kinds of field solvers. For this project, time and resource limitations dictated the use of the two and three dimensional field solvers in Ansoft’s signal integrity tools suite. The two dimensional solver, SI2D, analyzes the cross-section of a uniform transmission line and calculates characteristic impedances, signal propagation velocities, and propagation modes. SI2D solves for one particular frequency at a time, but provides a lot of information about what happens at that frequency, so the majority of the cable electromagnetic analysis was done by running SI2D frequency sweeps on a few specific cable geometries.

Ansoft’s three dimensional solver, SI3D, was used as a sanity check on SI2D’s results. SI3D provides inductance, capacitance, and resistance matrices for volume conduction and at 100MHz.6 While they aren’t discussed here, the volume conduction results are useful since SI2D, like many field solvers, solves equations whose numerical conditioning is proportional to the square of the analysis frequency. At 10kHz, SI2D’s solutions clearly suffered from conditioning problems, and instabilities occasionally manifested themselves at 100kHz. As a result, SI2D’s operating frequency was kept at or above a fairly conservative 1MHz.

Two Dimensional Models and SI2D Results

Model Setup

Four cable geometries were modeled in SI2D; the two simplified geometries shown in Figure 3 and the two more realistic geometries shown in Figure 4. Analyzing the simplified geometries takes substantially less time than handling the many wires of the realistic geometries, so the simplified geometries were used for initial explorations of cable behavior. However, USB cables need to be flexible, so stranded cables and a braided shield are used, rather than the solid wires and shield of the simplified geometries. Virtually all USB cabling strands seven wires together to form each of the cable’s five inner conductors, while the braided shield generally covers somewhere between 65 and 95% of the cable’s surface area at the shield’s radius. The shield shown in Figure 4 uses 80 wires to achieve about 85% coverage, which is about typical for well-built USB cables.

While the cable strands in Figure 4 are in one fixed position, many USB cables are built without any control over the strands’ orientation. Not only is there no strong guarantee of the strands’ orientation, but that orientation can change along the cable’s length. The shield is braided, so the conductors’ position in the shield shifts as a function of position along the cable, and the actual shield topology has more depth than the ring-like structure shown above. The shield is frequently tinned, so skin effect losses in such a shield will be somewhat higher than the ones discussed here.7 Also, the voids within the cable shield are often filled with shredded plastic to hold all of the

---

6 Volume conduction occurs when the skin depth is comparable to the thickness of the conductors being modeled. For copper wires in USB cables, the upper limit on volume conduction is about 25kHz.
7 The vast majority of cables use crimped connections to mate the cable conductors to the plugs at the cables’ ends, so tinning of the shield isn’t necessary mechanically and some cable manufacturers ship with untinned shields. However, the USB spec calls out a tinned shield, presumably in an attempt to increase the shield’s thickness—and therefore its effectiveness at low frequencies—even though standard industry practice is to DC couple the shield and ground wire to system ground at both ends of the cable, so noise is quite capable of infiltrating the cable by coupling through system ground. Further, the drain wire really serves no purpose within the shielded cable, and some manufacturers leave it out.
internal conductors in place. Rather than model all of these shreds, any area within the cable not occupied by a wire or insulation is filled with a uniform dielectric with $\varepsilon_r = 1.5$, which seems about right for the effective dielectric constant of a space that’s 40% plastic and 60% air. Most plastics have dielectric constants in the 2.1 to 2.6 range, though there’s substantial variation even within one particular kind of plastic. HDPE insulation, like that used on the signal pair, varies from about 2.2 to 2.4 and PVC insulation—which is on the power pair—runs from around 3 up to 8, with the most common values being in the 4 to 5 range. Dielectric constants of 2.3 and 4.5 were used for HDPE and PVC, respectively.

Since a 2D field solver assumes a cross section that's independent of length, the only way to assess the effect of positional variations is to put many different geometries into the solver and compare the results. The same is true for variations in dielectric constants. Unfortunately, time constraints made the development of tools that could do this infeasible. For similar reasons, the power and signal pair insulation actually fills the interstices between cable strands, rather than just surrounding the outside of the cables. It seems reasonable, however, to assume most of these departures from actuality inject only small errors into the simulation results.8

Figure 4 typical USB cable cross sections with the signal pair at 0° (left) and 90° (right)

Solution Process

SI2D is a finite element solver which solves differential equations on a mesh of triangles, such as the one shown in Figure 5. Adaptive meshing is used to create a high triangle density in areas where fields change rapidly and lower densities in areas where the fields change more slowly. For capacitance, SI2D solves the electrostatic equation

$$\nabla \cdot (\varepsilon_r \varepsilon_0 \nabla \phi(x,y)) = \rho(x,y)$$

8 Perhaps the largest effect is the relative cable orientation within the signal pair. Even this is small. Rotating both cables 30°, so that they’re facing each other broadside (as in the 0° orientation) to each other rather than point to point (as in the 90° orientation) increases the differential impedance by 0.7Ω.
for the charge density $\rho$ and the voltage $\phi$ is found from known boundary conditions. $x$ and $y$ are right and up in all of the 2D figures in this paper and $z$ points out of the page.

**Figure 5** SI2D finite element mesh for 90° many conductor geometry’s 10GHz impedance solution

The inductance solution comes from solving for the $z$ directed current density $J_z$ using the magnetic vector potential $A_z$.

$$J_z(x, y) = \nabla \times \left( \frac{1}{\mu_0} \nabla \times A_z(x, y) \right)$$

The equations for the impedance solution are

$$\nabla \times \frac{1}{\mu_0} (\nabla \times A) = \sigma (-j \omega A - \nabla \phi), \quad I_j = \int_{s_j} (\sigma + j\omega\varepsilon)(-j \omega A - \nabla \phi) ds$$
and the admittance solution is governed by

$$\nabla \cdot (\sigma + j\omega\varepsilon)\nabla \phi = 0, \ \nabla \cdot (J_z + j\omega D) = 0$$

where $D$ is the electric flux density. All six of these equations can be easily derived from Maxwell’s equations and solved numerically reasonably quickly. Solution times ranged from a fraction of a second for small 600 triangle meshes up to about 45 minutes for the largest meshes, which had around 46,000 triangles.

### Simulation Results: Characteristic Impedance

Limitations within SI2D meant that frequency sweeps had to be run manually, rather than in batch mode. This meant efficient use of overnight or weekend computing runs was impractical. Consequently, each of the four two dimensional model geometries was analyzed only at 1MHz, 10MHz, 100MHz, 1GHz, and 10GHz. Each analysis produces four matrices—inductance, capacitance, impedance, and admittance—that describe the coupling between the cable’s conductors.

The four matrices are analogous to the LC and RLGC models of a single transmission line. The characteristic impedance of a single conductor transmission line is $Z_0 = \sqrt{\frac{L}{C}}$ for a lossless line and

$$Z_0 = \sqrt{\frac{L}{C}} = \sqrt{\frac{R + j\omega L}{G + j\omega C}}$$

for a lossy line. With multiple conductors, $L$, $C$, $Z$, and $Y$ become the matrices returned by SI2D, and the characteristic impedance matrix $Z_0$ is found from $LC$ or $ZY$. The voltages and currents in the transmission lines are described by $\mathbf{v} = Z_0 \mathbf{i}$. For a USB cable, $Z_0$ is symmetric and this system of equations is:

$$
\begin{bmatrix}
V_{D-} \\
V_{D+} \\
V_{\text{drain}} \\
V_{\text{ground}} \\
V_{\text{shield}} \\
V_{\text{bus}}
\end{bmatrix} = 
\begin{bmatrix}
Z_{--} & Z_{+-} & Z_{d-} & Z_{g-} & Z_{s-} & Z_{v-} \\
Z_{+-} & Z_{++} & Z_{d+} & Z_{g+} & Z_{s+} & Z_{v+} \\
Z_{d-} & Z_{d+} & Z_{dd} & Z_{gd} & Z_{sd} & Z_{vd} \\
Z_{g-} & Z_{g+} & Z_{gd} & Z_{gg} & Z_{sg} & Z_{vg} \\
Z_{s-} & Z_{s+} & Z_{sd} & Z_{sg} & Z_{ss} & Z_{vs} \\
Z_{v-} & Z_{v+} & Z_{vd} & Z_{vg} & Z_{vs} & Z_{vv}
\end{bmatrix} 
\begin{bmatrix}
I_{D-} \\
I_{D+} \\
I_{\text{drain}} \\
I_{\text{ground}} \\
I_{\text{shield}} \\
I_{\text{bus}}
\end{bmatrix}
$$

From the standpoint of USB signaling, the excitations of interest are the 1A differential signal $[I_{D-}, I_{D+}] = [-1 1]$ and the 1A common mode signal $[I_{D-}, I_{D+}] = [1 1]$.

When $Z_-$ is equal to $Z_{++}$, the system above can be recast in terms of decoupled differential and common mode signals $V_{\text{diff}} = V_{\text{D+}} - V_{\text{D-}}, I_{\text{diff}} = 0.5(I_{\text{D+}} - I_{\text{D-}})$, $V_{\text{cm}} = 0.5(V_{\text{D+}} + V_{\text{D-}}), I_{\text{cm}} = 0.5(I_{\text{D+}} + I_{\text{D-}})$:

---

9 Or multiples thereof.
Where \( Z_{\text{diff}} = 2(Z_{++} - Z_{+}) \) is the differential characteristic impedance and \( Z_{\text{cm}} = Z_{++} + Z_{+} \) is a common mode characteristic impedance. However, it’s not clear what \( Z_{\text{cm}} \) represents an impedance to, since the common mode return path isn’t explicit. This question is further complicated by SI2D apparent computation of the elements of \( Z_0 \) with respect to whatever it considers to be infinity. The resulting \( Z_0 \) isn’t directly usable, since the common mode impedance of interest is the signal pair’s impedance with respect to the return paths in the USB cable, not some unclear location used by the field solver.

One way of resolving this issue is to find the differential impedances of D+ and D- with respect to the other conductors in the cable—in addition their mutual differential impedance—and then compute the common mode impedance by putting the differential impedances in parallel:

\[
\begin{bmatrix}
V_{\text{diff}} \\
V_{\text{cm}} \\
V_{\text{drain}} \\
V_{\text{ground}} \\
V_{\text{shield}} \\
V_{\text{Vbus}}
\end{bmatrix}
= 
\begin{bmatrix}
Z_{\text{diff}} & 0 & 2(Z_{d+} - Z_{d-}) & 2(Z_{g+} - Z_{g-}) & 2(Z_{s+} - Z_{s-}) & 2(Z_{V+} - Z_{V-}) \\
0 & 2 & Z_{d+} + Z_{d-} & Z_{g+} + Z_{g-} & Z_{s+} + Z_{s-} & Z_{V+} + Z_{V-} \\
2(Z_{d+} - Z_{d-}) & Z_{d+} + Z_{d-} & Z_{dd} & Z_{gd} & Z_{sd} & Z_{Vd} \\
2(Z_{g+} - Z_{g-}) & Z_{g+} + Z_{g-} & Z_{gd} & Z_{gg} & Z_{sg} & Z_{Vg} \\
2(Z_{s+} - Z_{s-}) & Z_{s+} + Z_{s-} & Z_{sd} & Z_{sg} & Z_{ss} & Z_{Vs} \\
2(Z_{V+} - Z_{V-}) & Z_{V+} + Z_{V-} & Z_{Vd} & Z_{Vg} & Z_{Vs} & Z_{VV}
\end{bmatrix}
\begin{bmatrix}
I_{\text{diff}} \\
I_{\text{cm}} \\
I_{\text{drain}} \\
I_{\text{ground}} \\
I_{\text{shield}} \\
I_{\text{Vbus}}
\end{bmatrix}
\]

This approach results in the impedances shown on the left side of Figure 6. The common mode impedance target called out by the USB specification is 30Ω and the differential impedance target is 90Ω. Particularly encouraging is that the difference between the simple model and the many conductor model is a very weak function of frequency. This indicates the simplistic model does a good job of describing the signal pair’s impedances in a balanced configuration that, at least with the signal pair at 90°, getting identical results out of the simple and many conductor models is mostly a matter of tuning the cable’s cross section.

However, when the signal pair is at 0°, \( Z_- \) and \( Z_+ \) aren’t equal, so there’s no manipulation of the characteristic impedance matrix that can decouple the differential and common mode signals, so it’s something of a misnomer to speak of the signal pair as a differential pair. While this makes sense physically—the pair’s geometry is unbalanced, so there’s no reason to expect that putting in differential currents will result in purely differential voltages—it poses the question of how to quantify how unbalanced the pair is. One way of doing this is to define “differential” impedances for D+ and D-: \( Z_{\text{diff D+}} = 2(Z_- - Z_+) \) and \( Z_{\text{diff D-}} = 2(Z_+ - Z_-) \). The results of this approach are shown in on the right side of Figure 6. Again, both the simple and many conductor models show the same frequency dependence, though the simple models have a wider impedance spread.
The black traces are the average of $Z_{\text{diff}_D-}$ and $Z_{\text{diff}_D+}$ for the many conductor model. If the signal pair has a sufficiently high number of twists per inch, the impedance discontinuities at the 0 and 180° positions will be both electrically small and separated by an electrically short distance. In this case, the discontinuities will average together to produce a “lumped” characteristic impedance. As Figure 6 shows, the averaged differential and common mode impedances are fairly close to the impedances of the 90 and 270° positions. This is fortunate, since it means the power pair’s presence within the cable shield has a small impact on the signal pair’s impedance at sufficiently low frequencies. Figure 7 takes this approach one step further, showing the average impedance when an entire twist is electrically small.

Loosely speaking, electrically small means a length of 10 to 15% of wavelength. The length of a signal pair twist is generally around 3cm, and wavelengths within the cable are around 20cm at 1GHz, which suggests 1GHz as a nominal frequency at which the twists’ impedance discontinuities start to become visible. 1GHz is high enough that it’s reasonably accurate to consider the cable as a uniform transmission line, though this approximation is something of a stretch for 480Mbit/s signals. Figure 8 shows spectrograms of the common mode and differential signals of the USB packets in Figure 2, as well as spectrograms of the signals on D+ and D-. These spectra are fairly representative of the USB population as a whole and, while the 12Mbit/s power spectral density drops to the noise floor well below 50MHz, the 480Mbit/s signals’ power spectral density is noticeable out to around 2.5GHz. However, the 480Mbit/s PSD’s magnitude remains at or below -
40dB for the 1–2.5GHz band, so nearly all of the signals’ energy propagates in the DC to 1GHz range where twist discontinuities are negligible.

Figure 8 spectrograms of 12Mbit/s (left) and 480Mbit/s (right) USB packets from Figure 2, colorbars are in dB

This is a very significant result, since it suggests cable models shouldn’t need to descend to the level of individual twists in order to be accurate. Instead, the cable can be described in terms of an averaged model that can represent a much larger section of cable than a single twist. Not only does this hypothesis greatly simplify the cable model, but the simulation results of section 0 strongly suggest it’s correct.

Simulation Results: Propagation Velocity and Loss

Besides characteristic impedance, two main cable performance parameters regulated by the USB specification are propagation velocity and loss. Loss can be extracted from the Y and Z matrices that describe a lossy multiconductor transmission line, while velocity can be extracted from either L and C or Y and Z. Both the loss and velocity extractions provide answers in terms of eigenmodes, which are special excitations of the transmission line system in which no crosstalk occurs between conductors. Each eigenmode has its own propagation velocity and loss and the eigenmodes form a spanning basis, so any excitation can be uniquely described in terms of eigenmodes.

This means two things. First, a cable’s energy loss per unit length can be computed by expressing the excitation of interest in terms of eigenmodes, finding the amount of input energy dissipated by each eigenmode, and adding up the results. The resulting loss figures are shown on the left side of Figure 9, along with the USB specification’s rather conservative attenuation limit. The computed losses agree fairly well with network analyzer results from several sources. For all practical purposes, these losses are negligible—even the worst-case common mode loss is less than -0.75dB at 1GHz. Since the bulk of USB signal energy is at frequencies well below 1GHz, amplitude attenuation from cable losses is only one or two percent.

Second, the eigenmode decomposition means even a pure sinusoid will be distorted as it propagates along the cable because the coupling amongst the conductors in the cable will break the sinusoid up

---

10 Actually, the USB spec limits a cable’s maximum attenuation, not attenuation per unit length. I’ve translated the spec’s loss limits to loss per unit length by dividing them by the maximum cable length of 5m.
into several smaller sinusoids that all travel at different velocities. The net result is akin to dispersion, though the underlying mechanism is different. This phenomena means there’s no single number that describes the propagation velocity of a signal, and USB cable eigenmode velocities tend to range from around $1.75 \times 10^8$ m/s up to $2.1 \times 10^8$ m/s, with one fast mode at about $2.45 \times 10^8$ m/s. Fortunately, both common mode and differential mode excitations tend to excite just two or three of the six eigenmodes strongly and, fairly consistently, its the fast, low loss modes that are excited.

Figure 9 aggregate propagation loss (left) and excitation averaged propagation velocities (right)

Since most of the signal energy is concentrated in a couple modes of comparable speed, it’s not unreasonable to summarize the cable’s propagation velocity in terms of a weighted average of the modes’ average propagation velocities. The right side of Figure 9 shows the average propagation velocities when each eigenmode is weighted by the normalized magnitude of its excitation. These results are in excellent agreement with oscilloscope measurements, which generally yield propagation velocities close to $2.1 \times 10^8$ m/s.\(^\text{11}\)

Three Dimensional Model and SI3D Results

Model Setup

\(^\text{11}\) There is a healthy spread from one manufacturer to another, but the population average is right around $2.1 \times 10^8$ m/s, with nearly all good cables running between $2.0 \times 10^8$ m/s and $2.2 \times 10^8$ m/s.
Figure 10 shows the SI3D model used to find the impedance and admittance matrices of one complete twist, using the same color scheme as the SI2D models. The twist is 3cm long and, though the shield and insulation aren’t shown, the cross sections at 0, 90, 180, and 270° are essentially the same as the simple cross sections shown in Figure 3. However, real USB cables have bulges where the pair rotates through 45, 135, 225, and 315°, and modeling these deformations is beyond the capabilities of SI3D’s geometry constructor. As a result, the shield’s inner diameter was increased from 2.8mm to 3.0mm and the power pair and drain wire were shifted 0.15mm to the left of their Figure 3 locations. This provides enough room for the twisted pair to rotate within the cylindrical shield used in the model without the signal pair’s insulation having to occupy the same space as the power pair insulation.12

Solution Process

---

12 Simulating this simplified geometry required 790MB of RAM. Back of the envelope calculations suggest a many conductor simulation would require 50GB of memory and consume nearly a week of processor time.
SI3D is a method of moments solver which integrates field quantities over triangular surface meshes. This approach is attractive for 3D simulations because only the surfaces between objects need to be meshed, which tends to minimize the total number of triangles needed to represent the model. The capacitance between conductors in the model is found by applying 1V to the conductor of interest and integrating over each conductor to find the total charge. This approach solves for one row or column of the capacitance matrix at a time—the capacitance matrix is symmetric, so it doesn’t really matter which—and each matrix element is

\[
C_{ij} = \frac{Q_{ij}}{lV} = \sum_{k=1}^{N_{\text{triangles}}} \int_{\sigma_{ij}} I_i \sigma_k (r')dr'
\]

where the summation is over all triangles on conductor j. \(\sigma_k\) is the pulse basis function on triangle k and \(I_i\) is the basis function’s strength when 1V is applied to conductor i. The AC inductance is computed from

\[
L_{ij} = \int_{\gamma_{KS}, x=0} A_j \cdot K_i \, ds, \quad A_j = \frac{\mu_0}{4\pi} \int_{\gamma_{KS}, x=0} \frac{K_i}{r} \, ds
\]

Where \(A_j\) is the magnetic vector potential created by applying 1A to conductor j and \(K_i\) is the surface current induced by the current sourced into conductor i. A surface impedance approximation is used with \(K\) to find the AC resistance. The volume conduction computations are
similar, though the surface integrals are replaced with volume integrals and $K$ is replaced by the volume current $J$.

$$L_{ij} = \int_{V_{ij} \times 0} A_j \cdot J \cdot \partial v, \quad A_j = \frac{\mu_0}{4\pi} \int_{V_{ij} \times 0} \frac{J}{r} \partial v$$

The 3D model converged at 21,691 triangles for the capacitance solution, 47,647 tetrahedra for volume inductance and resistance, and 167,873 triangles for AC inductance and resistance. Total run time for all three solutions was about 11 hours.

**Simulation Results**

As with SI2D, SI3D doesn’t model dielectric loss, so SI3D’s admittance matrix is just a capacitance matrix. Unlike SI2D, SI3D’s capacitance computation is frequency independent, and SI3D’s impedance computations are done at 100MHz. Additionally, SI3D treats the structure it models as electrically small, so modal information isn’t available.

These differences make a direct comparison of SI2D and SI3D results difficult. Since SI2D comes up with modal wavelengths around 2m at 100MHz, a single twist is electrically small. Thus, it’s reasonable to expect SI3D’s 100MHz $Y$ and $Z$ to be comparable to averaged impedance and admittance matrices found by SI2D at 100MHz. One way of finding the average matrices is

$$Z = 0.25Z_0^1 + 0.25Z_{180}^1 + 0.5Z_{90}^1$$
$$Y = 0.25Y_0^1 + 0.25Y_{180}^1 + 0.5Y_{90}^1$$

Where the 180° matrices are the 0° matrices with the signal pair’s rows and columns swapped to account for the pair’s reversed orientation. While this approach is simplistic, it does provide an important gauge for how well an average of the 2D results for the two different signal pair orientations reflects the overall nature of the cable.

For the resistance component of $Z$, the results are close, as are the results for $Y$. The 3D resistances are, on average, 83% of the 2D resistances, with the results tightly clustered between about 76% and 92%. The spread for admittance is about twice as wide, but the 3D capacitances average 85% of the 2D capacitances. The agreement between 2D and 3D is actually better than this number suggests—surprisingly, SI3D finds no coupling between the drain wire and the signal pair. The four resulting zero elements in the 3D admittance matrix cause about a 10% drop in the average admittance ratio.

Equally surprising is the clustering of the 3D inductances around 51% of the 2D inductances. While the 3D mutual and self inductances of the signal pair are 58 and 71% of the 2D results, the rest of the matrix is in the high 40s. The cause of this large discrepancy is not clear, and investigation of additional signal pair orientations in SI2D seems called for.

---

**Circuit Modeling of USB Cables**
In theory, connecting many individual RLCG sections in series will accurately model a system of multiconductor transmission lines. However, this approach creates extremely large Spice models that tend to exhibit convergence problems and long run times, so three alternate approaches have become popular.

1. Instead of using many RLCG sections, one RLCG section is created for some length of line, and then lossless transmission lines are attached to the section’s inputs and outputs in order to model propagation delay as well as coupling. While this approach is simple, it fails to account for the differing propagation velocities among eigenmodes, so this class of models doesn’t really represent crosstalk’s distributed nature. These models have to rely on transformers to represent mutual inductances, which can lead to substantial accuracy problems. Most existing USB cable do models use this scheme to model 1m chunks of cable, but only model one signal line in order to get around the mutual inductance problem. This means they can model either common mode propagation or differential propagation, but not both. Since the most problematic parts of USB signaling are single ended transitions that excite both modes, modeling just one line is a substantial limitation. Further, this type of model can’t represent the differing propagation velocities of common mode and differential signals, which substantially limits its utility in modeling USB signals.

2. Use controlled sources, usually voltage controlled voltage sources, to decompose an input signal into its eigenmodes, run the eigenmodes excitations through transmission lines to account for propagation delay, and use another set of controlled sources to translate the eigenmode excitations back into normal signals. This form of model only allows signals to propagate in one direction along the lines, but loss can be modeled by decreasing the sources’ gains, using lossy transmission line models, or by adding loss networks to the connections between the sources. Unfortunately, changing source gains represents a frequency independent loss, and most lossy transmission line models are also frequency independent. Lossy line models tend to suffer from poor accuracy and slow convergence, leading to simulations that take a long time to run and produce bad results. Additionally, the number of controlled sources necessary is $2N^2$, $N$ being the number of conductors modeled. For a six conductor USB cable, this means 72 interdependent sources, which leads to insoluble convergence problems.

3. Use a multiconductor transmission line model that’s built into a Spice tool. These models generally only handle frequency independent loss, and their accuracy is a subject of unending controversy. At least for USB cables, these models consistently produce results with far too much loss.

Given all of these constraints, the only viable implementation—short of writing one’s own circuit solver—is a simplified modal model with custom loss networks. For USB’s signal pair, the natural way to implement such a model is to split input signals into differential and common mode, propagate them using the twist-averaged common mode and differential results, and put them back together at the end. I pushed this implementation one step farther by adding real terminations to the differential and common mode transmission lines, which makes the model capable of simulating termination mismatches and reflections on the transmission lines. Signals still can only be injected at one end of the model, but the model is capable of handling reflections that make round trips through the cable. This is a plus for USB simulations, as reflections in real USB hardware often make at least one round trip of the bus before being damped to negligible levels.

---

13 I encountered a new record while working on this thesis. One model I set up took over three hours on an 866MHz Pentium III to simulate a single bit and produced results that were off by an order of magnitude.
Figure 12 shows the resulting schematic. The leftmost block provides the input signals and R3 and R4 average the inputs together to get the common mode signal. The two transmission line blocks handle common mode signals (top) and differential signals (bottom), approximating the cable loss with a second order lowpass section. The two rightmost blocks provide the output signals on D+ (top) and D- (bottom), while the bottom most block provides the minus one half of the differential signal necessary to find the output signal on D-. The terminations are set at USB’s standard 45Ω common mode and 90Ω differential and, based on the averaged SI2D results presented above, I chose to 35Ω common mode and 85Ω differential for the transmission lines’ impedances.\textsuperscript{14} The loss networks and propagation delays were set for a 5m long cable. Simulations spanning entire USB packets can be run in a few seconds.

\textsuperscript{14} There aren’t any frequency dependent lossless transmission line models.
Figure 13 shows a 480Mbit/s packet with the same data as 480Mbit/s packet in Figure 14. Since the data in Figure 14 comes from a full bus with cables, plugs, receptacles, microstrips, non ideal termination resistors, package parasitics, different impedances, and what not, the simulation results don’t exactly match the measured ones. The simulation does seem to capture all of the significant cable effects. The overall shapes of the output eyes are similar, and the simulation captures the three common mode effects at the packet’s start and end. Two of these effects are the glitches that the lower two red arrows point to. These glitches occur because the differential parts of the single ended transitions start a couple hundred picoseconds before the common mode components arrive, so the differential energy is able to drive around 100mV into the victim line before the common mode energy pushes the bit back into place.

A more subtle effect is the upwards shift of the first couple bytes of the packet, which is easily seen in Figure 13. This shift occurs because of the mismatch between USB’s 45Ω common mode terminations and USB cables’ 35Ω common mode impedance, which produces reflections that make several round-trips before they die out—it’s difficult to see with the noisy data used for Figure 13, but noiseless data shows the whole packet actually shifts up and down by a few millivolts as the common mode reflections settle. The same effect may be present in Figure 14, but the data is far from conclusive.\textsuperscript{15}

\textsuperscript{15} In case you’re wondering, the large dip just after the glitch in the end of the measured spike is ground bounce from the transmitter’s drivers switching off.
480Mbit/s USB receivers always operate on the differential signal amplitude, so these artifacts of the skew between differential and common mode signals aren’t directly visible. They do impact the receiver’s common mode budget, and the glitches have enough amplitude to mandate either a common mode range increase from 250mV to 350–400mV or the design of a data recovery circuit that can handle them. Additionally, USB’s common mode mistermination is the leading source of 12Mbit/s signal integrity violations, so a cable model that’s capable of representing them is quite useful.

Conclusions

Despite the large difference between SI3D and SI2D’s inductances, averaging together SI2D’s results for the two many conductor geometries creates in characteristic impedances, propagation velocities, and loss figures for USB cables that consistently agree with measured results. Translating these results into a Spice simulation is somewhat difficult because of Spice’s limited ability to handle frequency dependent impedances and the spread of propagation velocities created by the excitation of multiple eigenmodes within the cabling. It is however, possible to create simplified Spice models which capture the most significant effects of USB cabling on USB signals with reasonably high degree of accuracy.

The substantially higher amount of jitter and reflection in the measured data of Figure 14 suggests 480Mbit/s USB signal integrity is governed not by the cable that comprises most of the signal path, but by plug, receptacle, and package parasitics. However, any substantial increase in signal rate beyond 480Mbit/s will hazard the lumped-twist assumption made in this paper and probably require new cabling.

References

Publicly Accessible Sources

16 Just how big the glitches would be was a topic of considerable debate during the design of the 480Mbit/s signaling layer. Speaking as someone who spent considerable time explaining differing common mode and differential propagation velocities were, in fact, causal, it’s quite gratifying to have simulation data that corroborates experiment.

17 Albeit not as useful as a cable with a 45Ω common mode impedance, which would eliminate the mismatch. Unfortunately, building such a cable would require the off-center enclosure of the signal wires within their insulation and consistent control of wire’s relative orientation. Unless the cabling industry has been holding its cards unusually close to its chest, there’s no cable tooling that’s close to being able to do this, so building this kind of cable would require establishing completely new production tools, which is unlikely to happen any time soon.

18 Preliminary experiments with expanding the cable model of Figure 12 appear to confirm this hypothesis.

19 Substantial likely means 2+Gbit/s. It’s this author’s hunch that Future I/O, 1394b, and any faster version of USB will all end up losing to an out of the box extension of Serial ATA.
Confidential or Restricted Industry Sources


Access to Code and Projects

All the code and data mentioned here totals about 2GB, with much of it being protected by various forms of NDAs, CITRs, or other access restrictions. However, most of the thesis specific stuff, such as the Ansoft SI2D and SI3D models and Spice projects can be made available. If you’d like to get your hands on something, email Todd at twest@agora.rdrop.com.