

Allegro® X User Guide: Allegro® X Timing Environment

Product Version 23.1

September 2023

Last updated on: October 2012

© 2023 Cadence Design Systems, Inc. All rights reserved.

Portions © Apache Software Foundation, Sun Microsystems, Free Software Foundation, Inc., Regents of the University of California, Massachusetts Institute of Technology, University of Florida . Used by permission. Printed in the United States of America.

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

Allegro Platform Products contain technology licensed from, and copyrighted by: Apache Software Foundation, 1901 Munsey Drive Forest Hill, MD 21050, USA © 2000-2005, Apache Software Foundation. Sun Microsystems, 4150 Network Circle, Santa Clara, CA 95054 USA © 1994-2007, Sun Microsystems, Inc. Free Software Foundation, 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA © 1989, 1991, Free Software Foundation, Inc. Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, © 2001, Regents of the University of California. Daniel Stenberg, © 1996 - 2006, Daniel Stenberg. UMFPACK © 2005, Timothy A. Davis, University of Florida, (davis@cise.ulv.edu). Ken Martin, Will Schroeder, Bill Lorensen © 1993-2002, Ken Martin, Will Schroeder, Bill Lorensen. Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, Massachusetts, USA © 2003, the Board of Trustees of Massachusetts Institute of Technology. vtkQt, © 2000-2005, Matthias Koenig. All rights reserved.

Trademarks: Trademarks and service marks of Cadence Design Systems, Inc. contained in this document are attributed to Cadence with the appropriate symbol. For queries regarding Cadence's trademarks, contact the corporate legal department at the address shown above or call 800.862.4522.

Open SystemC, Open SystemC Initiative, OSCI, SystemC, and SystemC Initiative are trademarks or registered trademarks of Open SystemC Initiative, Inc. in the United States and other countries and are used with permission. All other trademarks are the property of their respective holders.

Restricted Permission: This publication is protected by copyright law and international treaties and contains trade secrets and proprietary information owned by Cadence. Unauthorized reproduction or distribution of this publication, or any portion of it, may result in civil and criminal penalties. Except as specified in this permission statement, this publication may not be copied, reproduced, modified, published, uploaded, posted, transmitted, or distributed in any way, without prior written permission from Cadence. Unless otherwise agreed to by Cadence in writing, this statement grants Cadence customers permission to print one (1) hard copy of this publication subject to the following conditions:

1. The publication may be used only in accordance with a written agreement between Cadence and its customer.
2. The publication may not be modified in any way.
3. Any authorized copy of the publication or portion thereof must include all original copyright, trademark, and other proprietary notices and this permission statement.
4. The information contained in this document cannot be used in the development of like products or software, whether for internal or external use, and shall not be used for the benefit of any other party, whether or not for consideration.

Disclaimer: Information in this publication is subject to change without notice and does not represent a commitment on the part of Cadence. Except as may be explicitly set forth in such agreement, Cadence does not make, and expressly disclaims, any representations or warranties as to the completeness, accuracy or usefulness of the information contained in this document. Cadence does not warrant that use of such information will not infringe any third party rights, nor does Cadence assume any liability for damages or costs of any kind that may result from use of such information. Cadence is committed to using respectful language in our code and communications. We are also active in the removal and/or replacement of inappropriate language from existing content. This product documentation may however contain material that is no longer considered appropriate but still reflects long-standing industry terminology. Such content will be addressed at a time when the related software can be updated without end-user impact.

Restricted Rights: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor.

Contents

1

<u>Allegro Timing Environment (ATE)</u>	5
<u>ATE Flow</u>	5
<u>Delay Overview</u>	6
<u>Differential Pair delay</u>	6
<u>Static Phase</u>	6
<u>Dynamic Phase</u>	6
<u>General Delay</u>	6
<u>Min Delay</u>	6
<u>Max Delay</u>	6
<u>Min/Max Delay</u>	6
<u>Matched Delay</u>	7
<u>Relative Delay</u>	7
<u>Allegro Timing Environment (ATE)</u>	8
<u>Timing Vision</u>	8
<u>Auto-Interactive Phase Tuning (AiPT)</u>	9
<u>Auto-Interactive Delay Tuning (AiDT)</u>	10
<u>Timing Vision</u>	11
<u>Timing Group</u>	11
<u>Timing Group Menu Path</u>	11
<u>Timing Vision Options</u>	12
<u>Display Controls</u>	13
<u>Timing Modes</u>	15
<u>DRC</u>	16
<u>Smart</u>	16
<u>Timing Mode - Off</u>	17
<u>Data-Tip Hover Over Information</u>	17
<u>Smart Data - Calculation Methods and Color Coding</u>	18
<u>Smart Data - Goal Calculation Methods - General Delay</u>	18
<u>Impact of Modes on Color Coding of Delay</u>	19
<u>Calculation and Color Coding of Differential Pair Phase</u>	24
<u>Timing Vision Phase Calculation Methods</u>	27

Allegro X User Guide: Allegro X Timing Environment

<u>Timing Vision Phase Calculation Methods: Two Pin Nets</u>	27
<u>Timing Vision Phase Calculation Methods: Multi Pin Nets</u>	30
<u>Timing Vision Use Model</u>	35
<u>Timing Vision Setup</u>	36
<u>Timing Vision Use Model: Final Checks</u>	38
<u>Auto-Interactive Phase Tuning (AiPT)</u>	39
<u>AiPT Options</u>	39
<u>Compensation Loc</u>	40
<u>Compensation Techniques</u>	41
<u>Combining Effects of Options</u>	48
<u>AiPT Use Model</u>	49
<u>Timing Vision Use Model: AiPT Setup</u>	50
<u>AiPT Selection Techniques</u>	52
<u>AiPT Feedback</u>	53
<u>Auto-Interactive Delay Tuning (AiDT)</u>	55
<u>AiDT Options</u>	55
<u>Auto-I Delay Tune Parameters</u>	56
<u>AiDT Use Model</u>	61
<u>Timing Vision Use Model: AiDT Setup</u>	61
<u>AiDT Selection Techniques</u>	64
<u>AiDT Feedback</u>	64

Allegro Timing Environment (ATE)

Solving delay whether it is a general delay or differential pair phase delays is a time consuming process. As designs get bigger or more complex in terms of numbers of delay constraints and their precision, this process requires automation to reduce the burden on the designer.

This automation provides the data in a simpler form to the designer about the nets which are affected. The automation also provides compensation techniques that are industry accepted standards.

The purpose of this chapter is to provide methodologies for solving timing relationships (differential phase, match groups and relative match groups) and other delay related constraints using the Allegro Timing Environment (ATE). This environment contains technology focused on visualizing and solving the delay issues. The chapter presents a general use model for PCB designers showing the concepts and principles for using these tools using an industry standard interface - DDR3.

ATE Flow

This ATE flow includes the following:

- Delays
- Allegro Timing Environment Use Model
- Automation (Tools) to Help Solve Delay

Delay Overview

Delay is generally described as the process of adding additional length into a routed solution in order to slow down a signal for a specific purpose. The various types of delay a design could contain fall into two categories:

Differential Pair delay

Static Phase

A form of delay that measures the difference in length between the positive/negative halves of a differential signal.

Dynamic Phase

A form of delay that pre/post corner compensates for the mismatched lengths when a differential pair makes a turn. When a turn happens, one signal is longer. Since it is on the outside edge making it take longer to reach its destination. Dynamic phase adjustments compensate for this by adding an appropriate delay bump in the shorter or inside member. The bumps slow it down and make sure the pair arrives at the destination at the same time.

General Delay

Min Delay

Where specific nets connections must be routed greater than the minimum constraint length.

Max Delay

Where specific nets connections must be routed shorter than the maximum constraint length.

Min/Max Delay

Where specific nets connections must be routed greater than the minimum constraint length and shorter than the maximum constraint length.

Matched Delay

Where specific nets/connections must be the same length within some tolerance.

Relative Delay

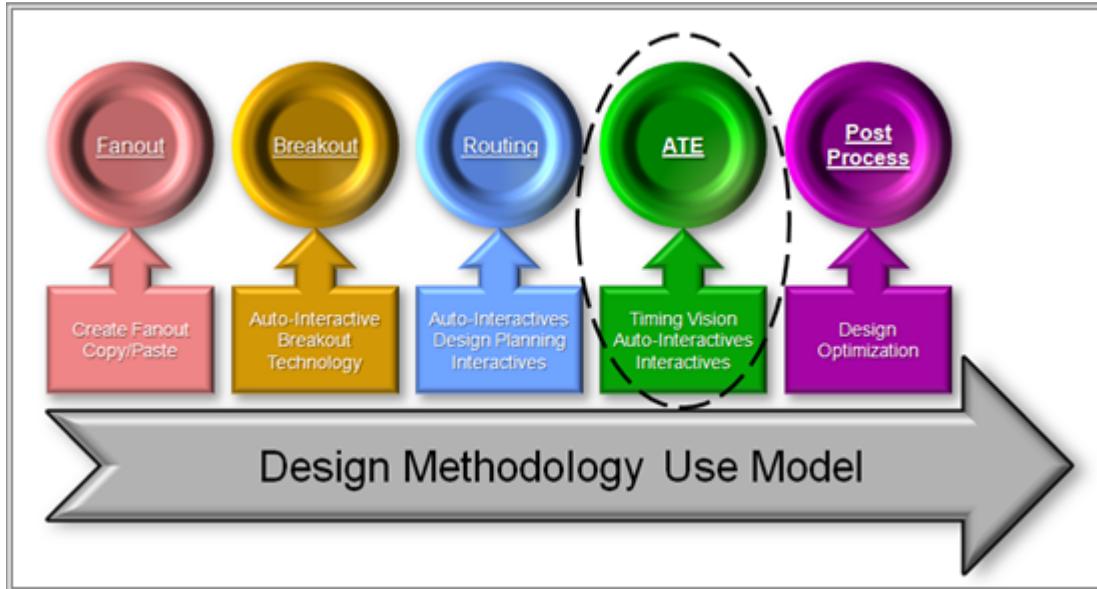
Where specific nets/connections must lead or lag some other net that controls the delay for the relative members. Relative delay may span many nets (a group). A one relative member could be in multiple groups creating a nested relationship.

Allegro PCB Editor and APD have a built in environment and tools to help you handle the delay problem in an efficient auto-interactive manner. This environment is called Allegro Timing Environment.

Allegro Timing Environment (ATE)

When designing a layout a basic flow is usually followed as shown in the following figure. In this flow, delay resolution is a very time consuming task depending on how many interfaces and nets require delay compensation.

Figure 1-1 ATE Location in the Design Flow



ATE Technologies

ATE is a set of tools designed to provide automation and specialized graphics to help solve the delay problem. Within the ATE toolset, there is a general process that involves starting the special graphics for visualizing the delay issues. You would then solve differential phase, and the general delay problem.

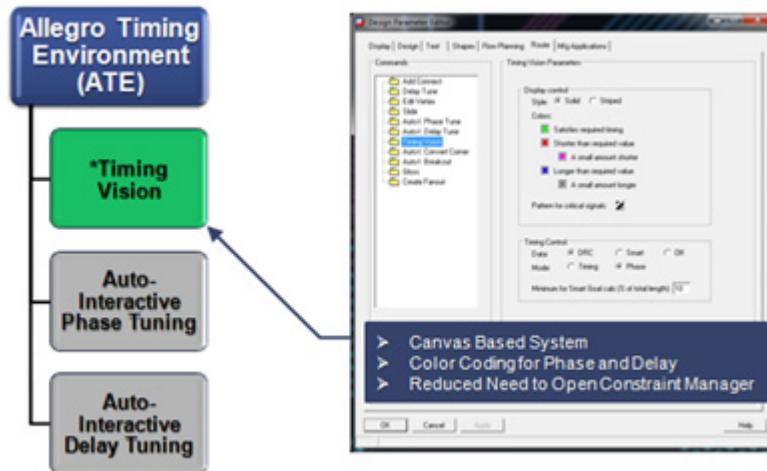
Timing Vision

Timing Vision provides visual aides to help solve delay by using a color coded system. This graphical environment figures out what is good and what needs to be adjusted to solve the remaining delay constraints in the design. These graphics calculate how long/short any

Allegro X User Guide: Allegro X Timing Environment

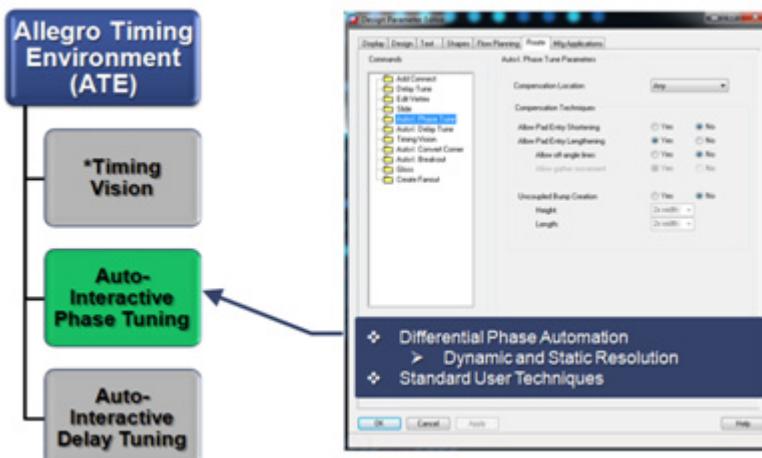
Allegro Timing Environment (ATE)

particular connection is in relation to a critical net. A critical net can be a target or just the physically longest net in the match group.



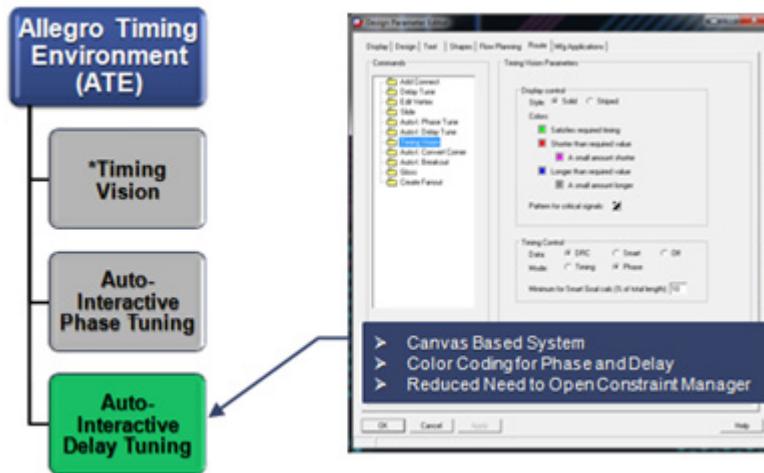
Auto-Interactive Phase Tuning (AiPT)

Auto-Interactive Phase Tuning helps to solve differential pair static and/or dynamic phase delays. This automation tries to solve the phase compensation problem automatically using industry standard techniques.



Auto-Interactive Delay Tuning (AiDT)

Auto-Interactive Delay Tuning performs similar functionality to AiPT, but focuses on the general delay problem. This automation tries to solve the phase compensation problem automatically using industry standard techniques.



These three technologies are designed to work together to solve the delay constraints in the design.

Timing Vision

Timing Vision is part of the Allegro Timing Environment (ATE) that provides special graphics to let you see real-time delay and phase information directly on the routing canvas. These special graphic techniques include: custom cline coloring, stipple patterns and customized data tip information to define the delay problem in the simplest terms possible.

When the Timing Vision environment is active, it temporarily takes control of the etch, pin and via color to properly display the timing relationships or color coding. When you exit the environment, the original color schemes are restored.

Timing Vision uses data found in the system to create color code scheme to guide you to a fast, graphical understanding of the timing problems in the design.

Timing Vision requires a Timing Group which is a group of selected nets.

Timing Group

A Timing Group is a collection of nets that you want to focus on for delay resolution. Timing group does not have to be a match group. When a net becomes a member of the Timing Group it gets color coded. The color coding depends on how good or bad the net is in terms of meeting its delay goals. This color code is determined by analyzing the nets existing clines, delay constraints (Min/Max, matched, relative and their associated targets), and the choice of data mode to apply the appropriate color scheme to the interconnect of the Timing Group. There is only one Timing Group for the entire design.

When you exit the editor (Allegro or APD/SIP), Timing Groups are not stored in the database. You need to recreate them when you open or reopen a design. When you exit the editor and reopen the database, the Timing Vision color scheme will not be shown.

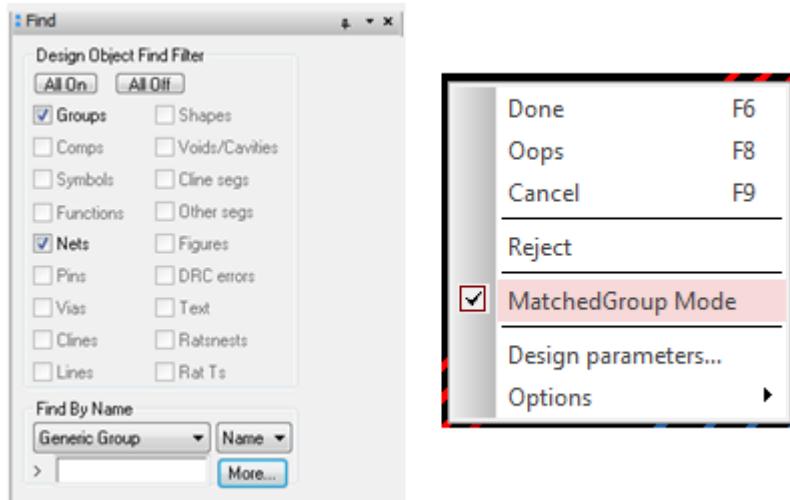
Timing Group Menu Path

Timing Groups are created using the command:

Route – Timing Vision

It restricts the Find Filter to allow only Nets and Group. However, if you use the right-menu button you will get a pop-up with *MatchedGroup Mode*. This mode automatically adds all members of a match group when you pick some particular net while in the Timing Group command. The *MatchedGroup Mode* becomes a set option until you turn it off.

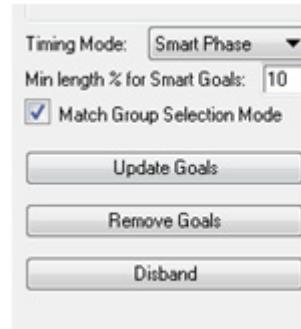
Figure 1-2 Timing Group Find Filter Choices and RMB Options



You can modify an existing Timing Group at any time by executing the Timing Group command and using the **Ctrl** key. You can add new members or remove existing members. The number of nets added or removed depends on the Find Filter settings and if *MatchedGroup Mode* is enabled or not.

You can delete the Timing Group completely by using the *Disband* command on the Options tab using Timing Vision command.

Figure 1-3 Timing Vision Disband Option



Timing Vision Options

You can set the parameters for Timing Vision in the Design Parameters dialog box on the Route Tab.

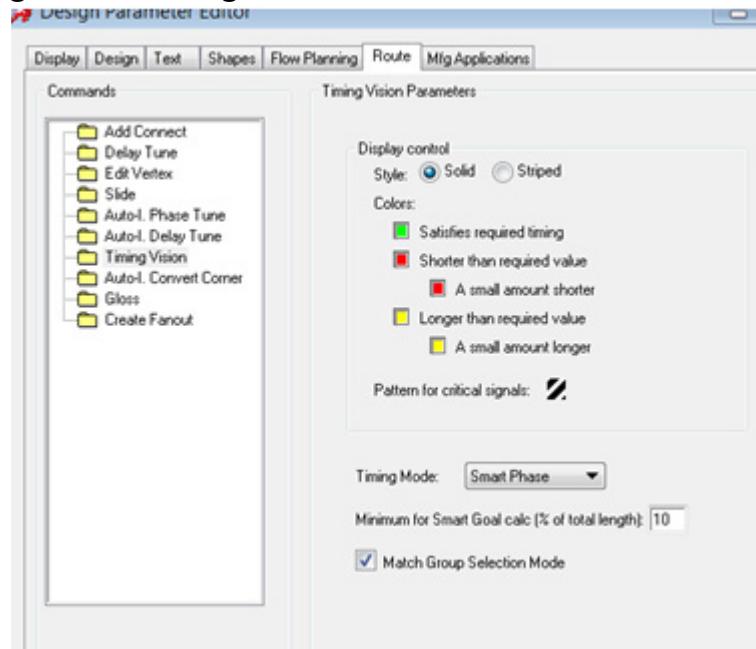
To access these options from the pull-down menus use the following path -

Setup – Design Parameters – Route – Timing Vision

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

Figure 1-4 Timing Vision: Design Parameters



Display Controls

- Style: sets the display of the data. The two options are: Solid or Striped.

Note: If this option is enabled, the stipple pattern option for *Pattern for critical signals* is grayed out.

- Satisfies Required Timing: is the color that is used to color code nets/pns/clines that meet the delay requirements.

Figure 1-5 Example: Satisfies Required Timing



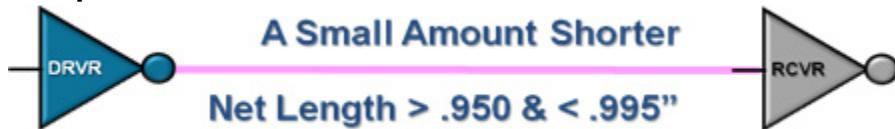
- Shorter than required: is the color that is used to color code nets/rats/clines that are more than 5% shorter than the required delay constraints.

Figure 1-6 Example: Shorter than required



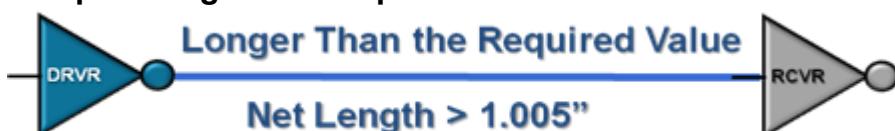
- A small amount shorter: is the color that is used to color code nets/rats/clines that are less than 5% shorter than the required delay constraints.

Figure 1-7 Example: A small amount shorter



- Longer than required: is the color that is used to color code nets/rats/clines that are more than 5% longer than the required delay constraints.

Figure 1-8 Example: Longer than required



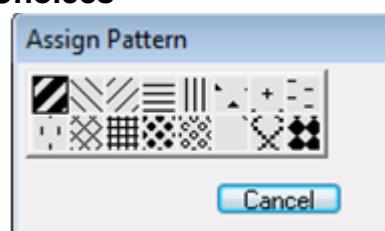
- A small amount longer: is the color that is used to color code nets/rats/clines that are less than 5% longer than the required delay constraints.

Figure 1-9 Example: A small amount longer



- Pattern for critical signals: is the stipple pattern for definition of critical signals in the Timing Environment. A critical signal is defined as:
 - A target as defined in constraint manager
 - The currently longest routed net in the Match Group at the time the smart data was updated (assuming Smart Timing mode is used).

Figure 1-10 Stipple Pattern Choices



Timing Mode section of the dialog do the following:

- Data - lets you pick the type of data used by Timing Vision to determine the proper color coding. There are two primary calculation modes (DRC and Smart):
 - DRC: uses Constraint Manager numbers to display the appropriate DRC status/color code for each affected cline on the canvas. The custom data tips show all delay/phase constraint information.
 - Smart: examines the relationship between all the clines in the current Timing Group. This option compares them to the required delay constraints (min/max, matched, relative and their associated targets) and computes a min/max range for each clines. This range should meet the Timing Group's delay constraints. The data tip show all delay/phase constraint information. The data tip uses a two character code to help you determine how to get the affected nets into compliance. This calculation method also looks at the existing route data and figures out the nets that are truly controlling the length. It may not actually be the target. Generally it is the longest net.

Both DRC and Smart Modes support the following options:

- Timing: puts the system into the general delay mode. This mode analyze group of nets. Such as byte-lanes of a memory system.
- Phase: puts the system into a mode specifically targeted at static phase issues. This mode lets you analyze differential clocks, busses, etc.
- Off: turns off the current Timing Vision color code scheme. Restores the original color coding, and removes the Smart data entries from the data tips. But preserves the existing Timing Group relationships that are created previously (until you exit Allegro/SIP). Reselecting either DRC or Smart modes re-enables the Timing Vision color code scheme using the existing Timing Group relationships.
- Minimum for Smart Goal Calc (% of total length): lets you define the system to handle the delay distribution when multiple pins/rats are involved in any particular net. For example, it allows to bias delay to be in the longer leg of a connection such as between the driver and a DIMM pin while not adjusting the DIMM to DIMM connection.

Timing Modes

Once a Timing Group is created, Timing Vision has two ways of looking at the data for generating the color code scheme that it displays on the canvas. It includes:

DRC

The DRC Timing Mode is basically a direct usage of the nets interconnect that makes up the net and Constraint Manager view of whether a net satisfies delay or not. Since Constraint Manager is net level view, it means even though some pin-pair(s) of the net satisfies its delay requirement, some other pin-pair may not. This will cause the entire net to be out of spec for delay. To set Timing Vision to use DRC data for color coding, set the Timing Mode as follows:

- *Route – Timing Vision.* Right-click and choose *Timing Mode – DRC Timing or DRC Phase*
- *Route – Timing Vision.* *Options – Timing Mode – DRC Timing or DRC Phase*
- *Setup – Design Parameters – Route – Timing Vision – Timing Mode – DRC Timing or DRC Phase*

Smart

The Smart Timing Mode takes the interconnect that makes up the net and breaks it up into its appropriate pin-pairs. This mode calculates for color code and then provides a color scheme to figure out which pin-pair is correct and which is not. This is significantly different than DRC Timing Mode net level view. The Smart Timing Mode view allows some pin-pairs to be in delay compliance while others may not.

With the same data presented in a more appropriate/effective manner, it will be easier for you to solve the delay constraints of the design. To set Timing Vision to use DRC data for color coding, set the Timing Mode as follows:

- *Route – Timing Vision.* Right-click and choose *Timing Mode – DRC Timing or DRC Phase*
- *Route – Timing Vision.* *Options – Timing Mode – DRC Timing or DRC Phase*
- *Setup – Design Parameters – Route – Timing Vision – Timing Mode – DRC Timing or DRC Phase*

When using the Smart Timing Mode, Timing Vision uses a calculation to internally modify the tolerance range to get all the members of Match Groups into a Min/Max range that will meet the requirements of the system. It uses this tolerance to create a Smart Goal for each net or pin pair which is then color coded for display to on the canvas. This Smart Goal Min/Max range is applied to the pin pairs which then drive the Auto-Interactive Tools or assist the manual interactive with delay convergence. This makes delay convergence easier to resolve since it is allowing you to satisfy a defined min/max range for each net or pin pair.

Timing Mode - Off

To turn off the Timing Vision, set the Timing Mode as follows:

- *Route – Timing Vision.* Right-click and choose *Timing Mode – Off*
- *Route – Timing Vision. Options – Timing Mode – Off*
- *Setup – Design Parameters – Route – Timing Vision – Timing Mode – Off*

Data-Tip Hover Over Information

All objects in the Allegro database offer customizable data tip information when you hover over an object. Timing Vision provides additional information for clines and nets when the environment is set to either phase or delay modes. The following table shows the various types of information depending on the Find Filter settings when you hover over the cline of a particular net.

Table 1-1 Data-Tip Information vs. Find Filter Setting

Find Filter	Delay	Phase
Cline		
		<pre>Connect Line Bottom CPU0_BANK0_CHNLA_DQS_DP17 S [(Goal) DIMM5.161 to U95.AC29 min= 3494.422 max= 3502.713 length= 3279.614 MIL xx [(R0)y] DIMM5.161 to U95.AC29 min= 3492.713 max= 3106.898 length= 3279.614 MIL grp= DRVR_TO_DIMM_CPU[0].BANK[0].CHNLA_DATA-ECC xx [(R0)y] DIMM5.161 to U95.AC29 min= 3492.713 max= 3106.898 length= 3279.614 MIL grp= DRVR_TO_DIMM_CPU[0].BANK[0].CHNLA-BYTELANE-8 (Dly) DIMM5.161 to U95.AC29 max= 4300.000 length= 3279.614 MIL 2015.174 MIL</pre>
Net		
		<pre>Net "Cpu0_Bank0-chnla_Dqs_Dp17" Length: 2436.882 MIL S [(Goal) DIMM5.161 to U95.AC29 min= 3494.422 max= 3502.713 length= 3279.614 MIL [Goal] DIMM5.161 to DIMM6.161 min= 393.999 max= 403.999 length= 398.608 MIL xx [(R0)y] DIMM5.161 to U95.AC29 min= 3492.713 max= 3106.898 length= 3279.614 MIL grp= DRVR_TO_DIMM_CPU[0].BANK[0].CHNLA_DATA-ECC xx [(R0)y] DIMM5.161 to U95.AC29 min= 3492.713 max= 3106.898 length= 3279.614 MIL grp= DRVR_TO_DIMM_CPU[0].BANK[0].CHNLA-BYTELANE-8 (Dly) DIMM5.161 to DIMM5.161 max= 525.000 length= 398.608 MIL (Dly) DIMM5.161 to U95.AC29 max= 4300.000 length= 3279.614 MIL (RDly) DIMM6.161 to DIMM5.161 min= 393.999 max= 408.980 length= 398.608 MIL grp= DIMM_TO_DIMM_CPU[0].BANK[0].CHNLA_DATA-ECC (RDly) DIMM6.161 to DIMM5.161 min= 393.999 max= 408.980 length= 398.608 MIL grp= DIMM_TO_DIMM_CPU[0].BANK[0].CHNLA-BYTELANE-8</pre>

Smart Data - Calculation Methods and Color Coding

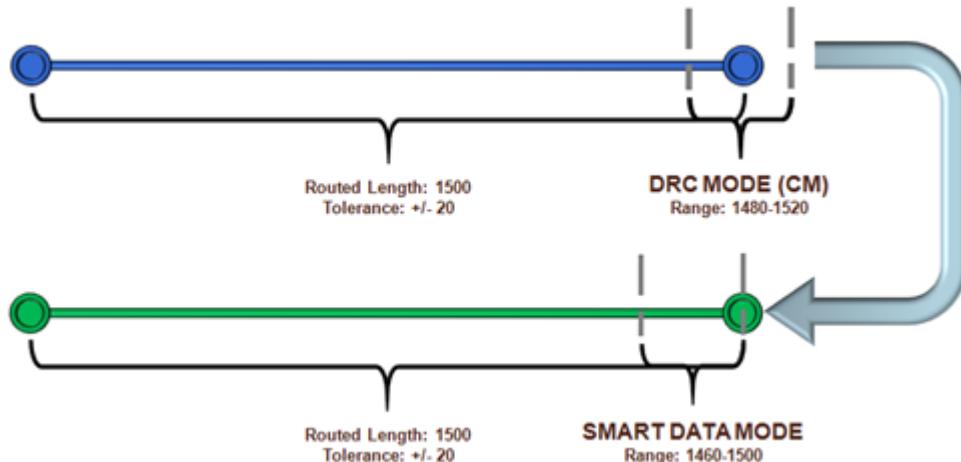
Timing vision uses the existing routing data to help find the longest, and shortest members of the nets in the Timing Group. It uses this data to calculate how much a net or pin-pair needs to be increased or decreased to get it into constraint compliance.

These calculations are also used to define the color code scheme Longer than required, A small amount longer, A small amount shorter, Shorter than required or Satisfies Required Timing options.

Smart Data - Goal Calculation Methods - General Delay

In the Smart Timing mode, Timing Vision subtracts the tolerance range from the longest routed length to keep all routes as short as possible.

Figure 1-11 Comparison of How DRC Mode and Smart Mode Evaluate Tolerance



This example uses standard blue color code to show the differences between DRC mode and Smart mode calculation methods. In the traditional DRC mode, Timing Vision uses the data from the Constraint Manager to create a color scheme that it shows on the canvas.

If you switch to the Smart Timing mode view, Timing Vision uses a calculation to internally modify the tolerance range to get all the members of the Match Group(s) into a Min/Max range that will meet the requirements of the system. It uses this tolerance to create a Smart Goal for each net or pin pair. This tolerance also drives the pin-pair color code and the manual and Auto-Interactive tools for solving the delay problem. This creates a defined min/max range for each net or pin pair.

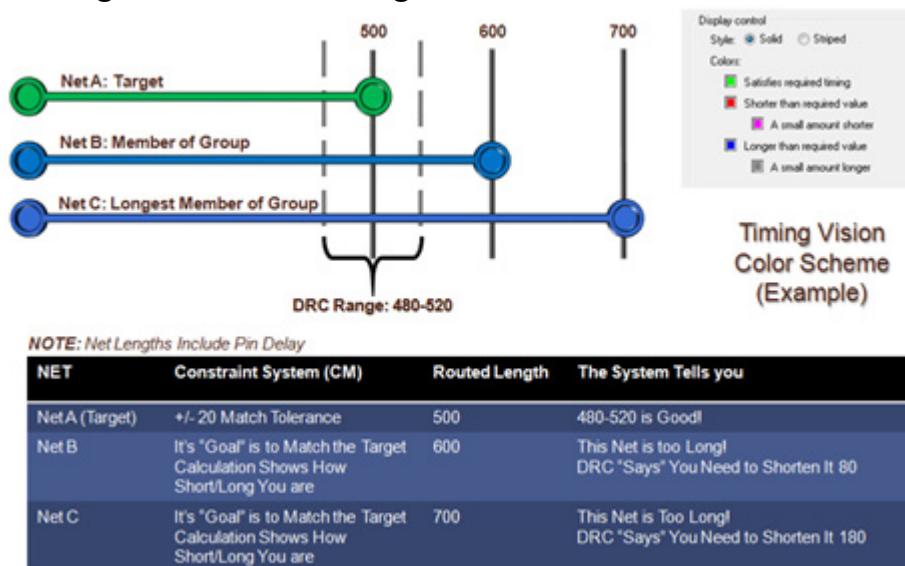
The tolerance range is subtracted from the longest routed length to keep all routes as short as possible.

For example, when nets have no target and use a simple match, smart goals take the existing tolerance range allowed by Constraint Manager. This tolerance is subtracted from the longest member to calculate the new min/max range. This implies that any other member related to this long route should now be easier to get within tolerance.

Impact of Modes on Color Coding of Delay

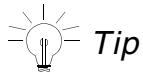
Using this data type, you would see that the only net that is green is the target. All the other nets would have the blue color code. It would make you think that all the nets need to be shorter to match the target. Satisfying delay may be possible if you could shorten the long routes enough. But, in many cases the target must lengthen to match the longest net. The longest may have its pins in physically longer locations, route around obstacles, and so on that make it longer than the target.

Figure 1-12 Timing Vision DRC Timing Mode Calculation



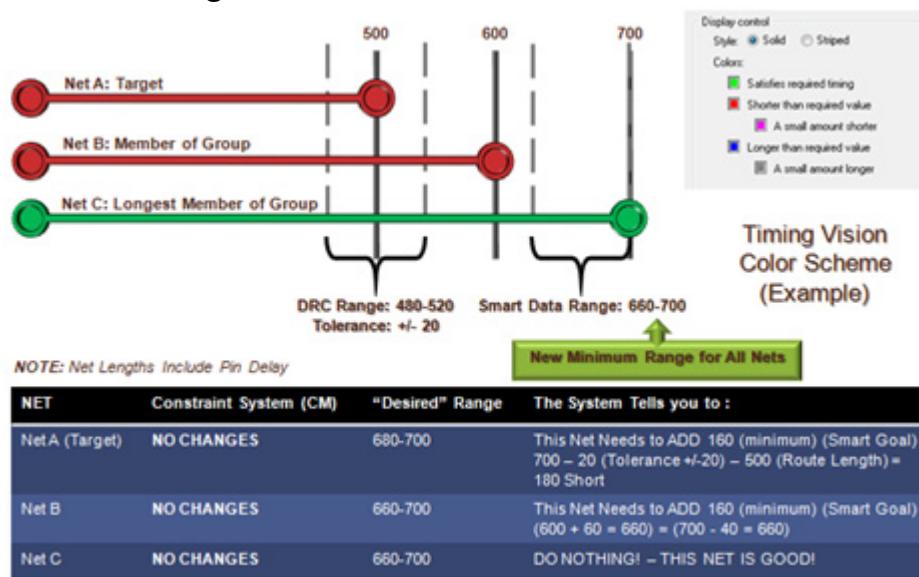
The following figure shows Smart Timing Mode view of the same problem. However Timing Vision using Smart Timing Mode that shows a completely different color scheme. In smart Mode, instead of the target being green (satisfied); the target and Net B will be red (short). They are shorter than the longest net. This implies that the longest net is the Net C and will be green.

So, as long as the longest routed net does not break a max length rule all the nets should be within the tolerance range.



If you want to make sure your interface, match group or net do not get too long for delay reasons, set a maximum length on at least one of the nets in the match group. This lets Timing Vision calculate smart goals that do not exceed the maximum length. This method ensures that the system works at full speed as designed by engineering.

Figure 1-13 Smart Timing Mode Calculation

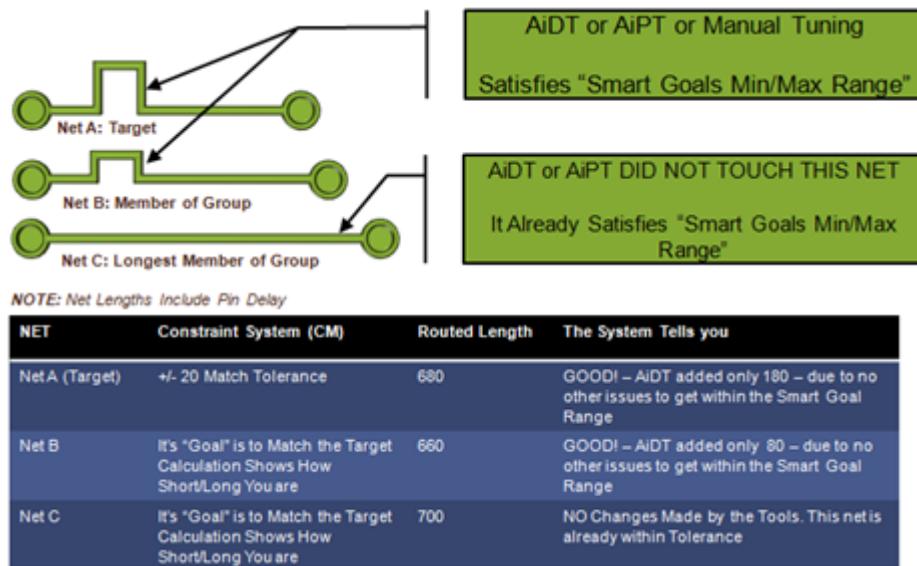


In the following figure, the target and Net B now fall in the range required by the longest net. All nets are now green, because the data is displayed in a more intelligent smart manner.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

Figure 1-14 Timing Vision End Results with Delay

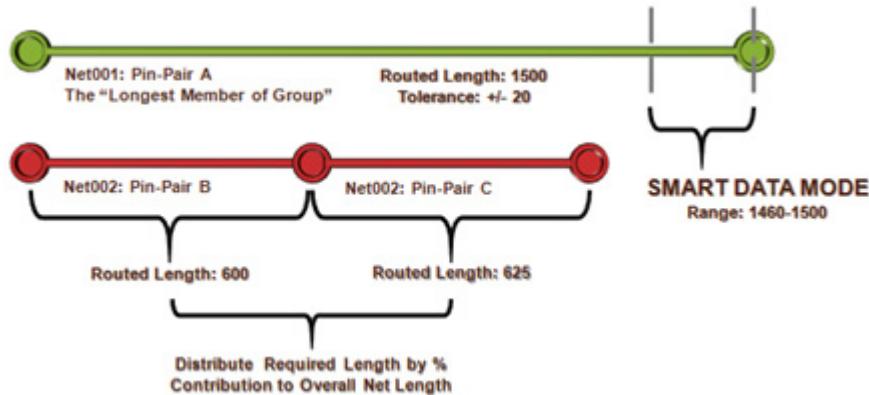


If you change the system back to DRC mode, the color code scheme will look the same. This implies all the nets are in green. That is the simplest form of how Timing Vision calculates general delay.

The next figure shows that the length of Net002 (a multiple pin-pair net) wants to equal the length of Net001. They are in the same match group. However, its routed length is only 1225 and is much shorter than the 1500 length of Net 001.

To achieve this, the pin pairs of Net002 must increase in length. Some calculations are to be made to increase the length of these pin-pairs equally.

Figure 1-15 Multi-Pin-Pair Matching to the Target or a Single Pin-Pair Net



The following figure shows the way smart Timing mode calculates the individual pin-pairs from net level data obtained from Constraint Manager. The table shows that Net001 is 1500

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

units long and is the longest member of the match group. Net002 is part of that match group and wants to increase in length to get within the timing tolerance. This tolerance is defined by the constraint system and controlled by Net001 since it is the longest net. In this case, there is no way to reduce the length of Net001. You need to increase the length of Net002 logically.

Note: The pin properties such as driver/receiver are ignored and are not considered in calculations. However, pin delay is used in Timing Vision calculations.

To achieve this Timing Vision looks at how much each pin pair contributes to the total length of Net002. In this example, pin-pair B contributes 48.98% and pin-pair C contributes 51.02% to the length of Net002. Timing Vision increases the length of each pin-pair by its contribution percentage to get Net002 within the smart range defined for Net001 which is 1460-to-1500.

There are two ways to achieve this:

- Pin-Pair B must now be greater than or equal to 715.10 as its “minimum” length and less than or equal to 735.10 as its “maximum”.
- Pin-Pair C must now be greater than or equal to 744.90 as its “minimum” length and less than or equal to 764.90 as its “maximum”.

Figure 1-16 Multi-Pin-Pair Calculation

Net							
Name	Route Length	Longest Route Length	Short by This Much = Longest Net Length - This Net Length	Tolerance (+/-)	Tolerance Max Range = 2 x Tolerance	Smart Goal Min (Net) = Route Length - 2 x Tolerance	Smart Goal Max (Net) = Route Length
Net001	1500.00	1500.00	0.00	20.00	40.00	1460.00	1500.00
Net002	1225.00	1500.00	275.00	20.00	40.00	1460.00	1500.00

Net001: Pin-Pair A
The “Longest Member of Group”
Net002: Pin-Pair B Net002: Pin-Pair C
Routed Length: 600 Routed Length: 625
Smart Goal Min: 715.10 Smart Goal Min: 744.90
Smart Goal Max: 735.10 Smart Goal Max: 764.90
SMART DATA MODE
Range: 1460-1500

Pin-Pair						
Net	Name	Route Length	% Contribution to Net Length	Tolerance per Pin Pair (Tolerance / # of Pairs)	Smart Goal Min (Pin-Pair) = %Contribution * Longest Net Smart Goal Min	Smart Goal Max = Smart Goal Min (Pin Pair) + Tolerance per Pin Pair
Net001	A	1500.00	100.00	40.00	1460.00	1500.00
Net002	B	600.00	48.98	20.00	715.10	735.10
	C	625.00	51.02	20.00	744.90	764.90
				Summation	1460.00	1500.00

If you add the minimums you will get 1460 which is equal to minimum smart length of Net001.

If you add up the maximums you will get 1500 which is equal to maximum smart length of Net001.

Allegro X User Guide: Allegro X Timing Environment

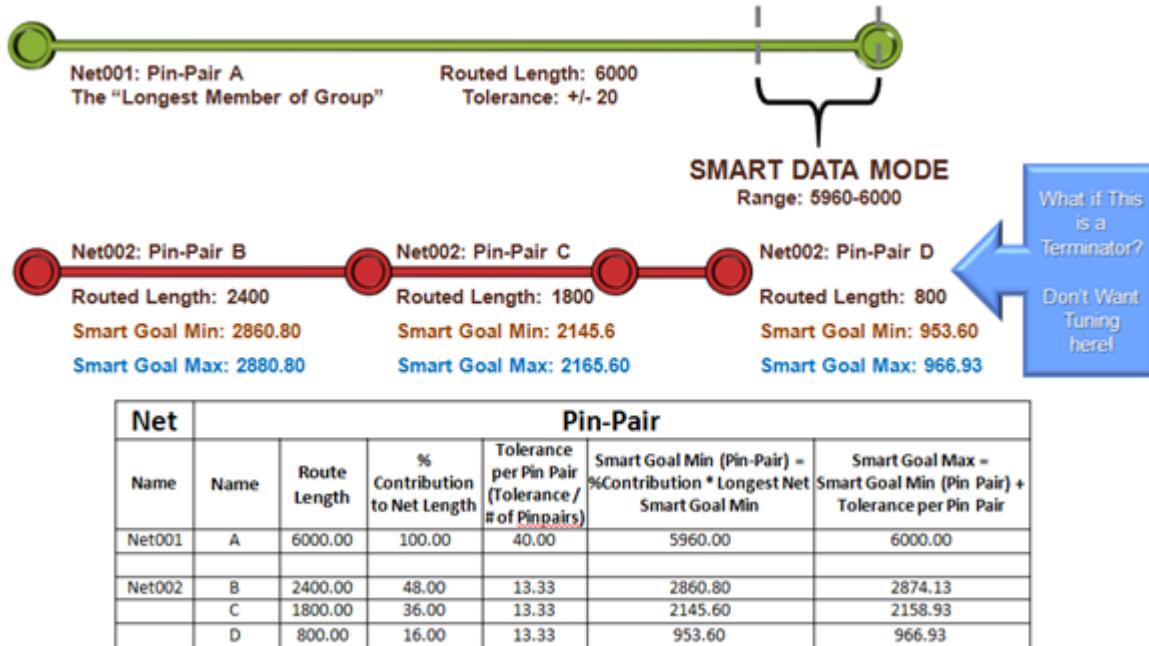
Allegro Timing Environment (ATE)

Now when you use the Auto-Interactive (AiPT or AiDT) or the interactive (manual) they try to create delay patterns that will put you inside a simple min/max range to satisfy delay. When satisfied, you get a green pin pair or net.

If you have multiple pin nets that may have different lengths between the pins/vias and some maybe termination. When dealing with termination, the placement of the parts is very close to the last load on the line making it a very short connection. So, you may not want to include this short connection in your overall delay tuning strategy or requirements. You can tell Timing Vision to filter out these short connections by using the Min Value % found on the *Design Parameters* form.

Here is the data on how Timing Vision calculated requirements for each pin-pair and its smart data minimum and maximum range values.

Figure 1-17 Understanding the Min Value % Effect on Timing Vision Calculations



The next figure shows how Timing Vision can be used to modify the length calculations for each of the pin pairs of a multi-pin net. In this example, the Minimum for Smart Goal Calc: (% of total length) field has been set to 20%. This implies that for any pin-pair of the net that contributes less than 20% of the nets overall length do not add. Also any additional length is not allowed to be added to a pin-pair that contributes less than this percentage.

In this example:

- Pin-Pair B must now be greater than or equal to 2946.67 as its “minimum” length and less than or equal to 2960.00 as its “maximum”.

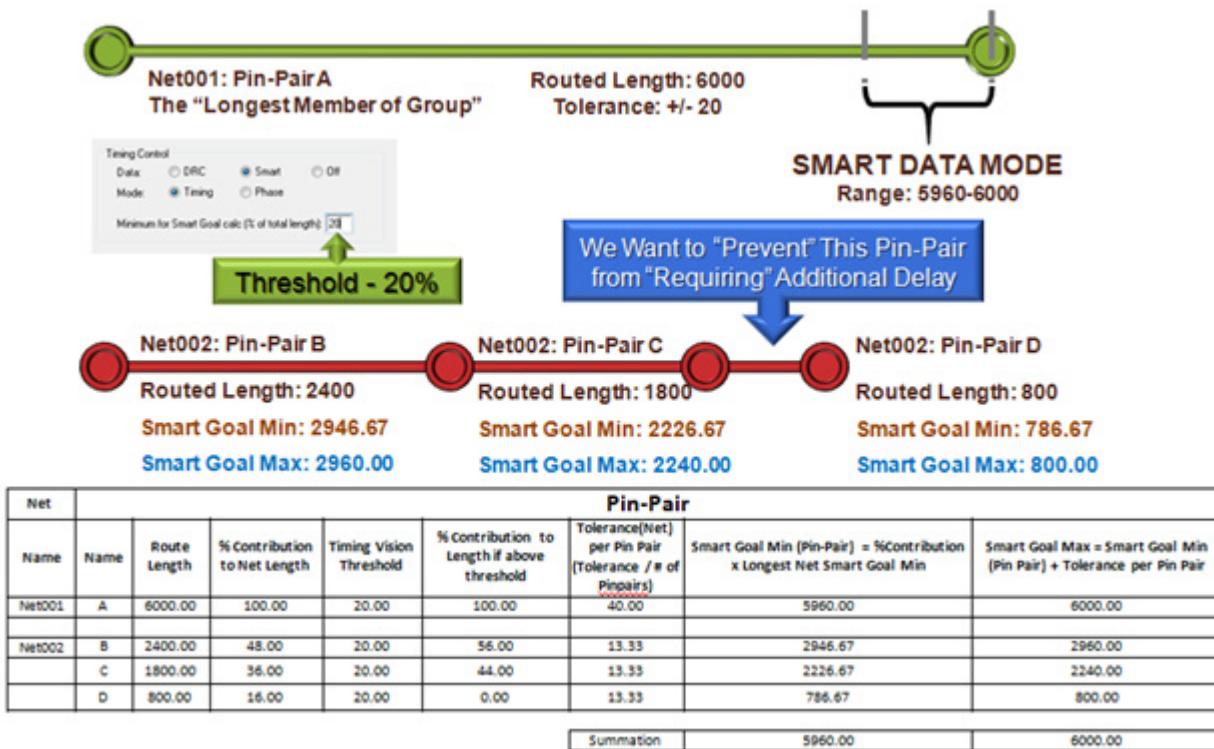
Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

- Pin-Pair C must now be greater than or equal to 2226.67 as its “minimum” length and less than or equal to 2240.00 as its “maximum”.
- Pin-Pair D must now be greater than or equal to 786.67 as its “minimum” length and less than or equal to 800.00 as its “maximum”. Since this is the routed length of pin-pair (maximum value), you need not to do anything and it remains satisfied.

Now adding up all of these lengths give the same minimum and maximum smart goals as Net001. So you are within the Smart Timing mode range for Net001 and the design will be delay constraint satisfied. In addition, when you run AiDT on Net002 - Pin-Pair D is already satisfied, so no additional length will be added to the current interconnect solution.

Figure 1-18 Using the Min Value to Remove Pin-Pairs from the Delay Calculation



Calculation and Color Coding of Differential Pair Phase

The following figure represents the way phase issues are shown in the various modes of the tools. In the top part of the image, the Canvas View is the basic view in the database before Timing Vision color codes the data based on the DRC Mode or the Smart Mode.

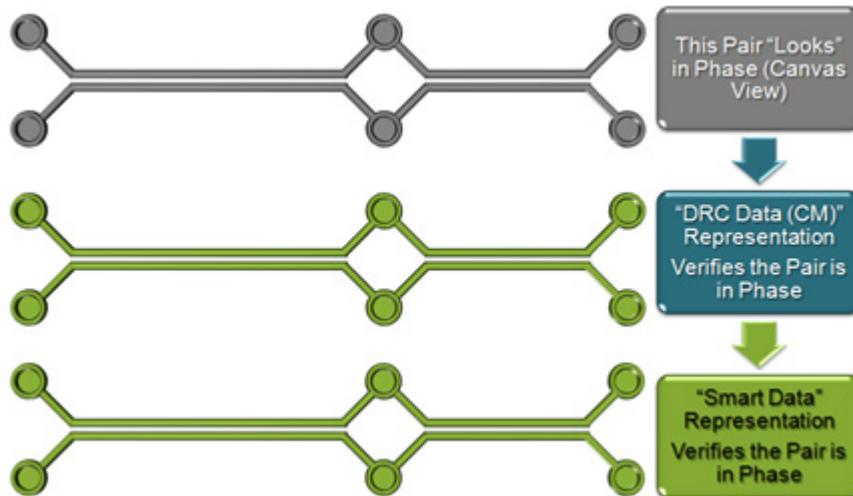
In the DRC Data representation/color scheme (middle image) the pairs satisfy the phase constraints and are color coded green.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

In the smart Timing Mode representation/color scheme (bottom image) the pairs satisfy the phase constraints and are color coded green.

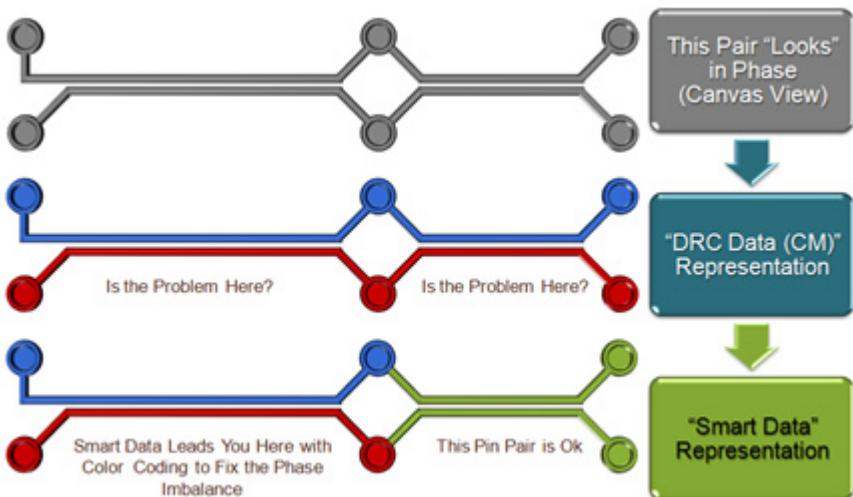
Figure 1-19 System, DRC Mode and Smart Mode view of Differential Phase Issues



Using the DRC Timing Mode, the red and blue color code only tells that the net has a phase issue. When zoomed out you still might not be able to see the left pin pair is the problem while the right pin pair is ok. Looking at full layer views of the design let you able to quickly zoom in, fix a problem, zoom out, fix another phase problem and so on.

But in Smart Mode you get the best view when zoomed out to see large areas of the design. The green will clearly stand out while the red/blue will lead you to the problem area. There will be no trying to fix the phase issue at the wrong end. This is due to the way smart Mode can calculate and color code the phase imbalance within each pin pair.

Figure 1-20 System, DRC Mode and Smart Mode view of Differential Phase Issues

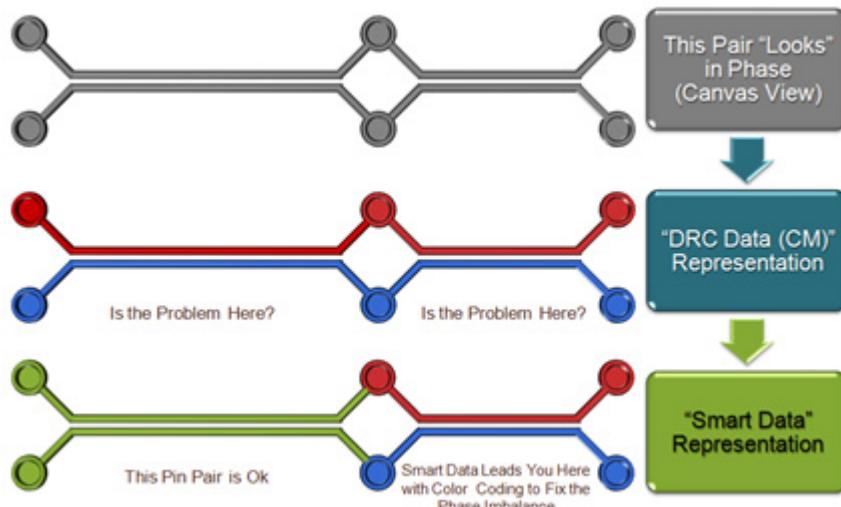


The next figure represents the way phase issues are shown in the various modes of the tools. In the top image, the Canvas View is the basic view in the database before Timing Vision color codes the data based on the DRC Timing Mode or the Smart Timing Mode.

In the DRC Data Representation/Color scheme one leg is color coded blue (for long) and red (for short). However, both pin pairs still look in phase this makes it hard to figure out what is wrong.

In the Smart Timing Mode Representation/Color scheme one pin pair satisfies the phase constraints and is color coded green. The second pin pair is color coded with one leg color coded blue (for long) and red (for short).

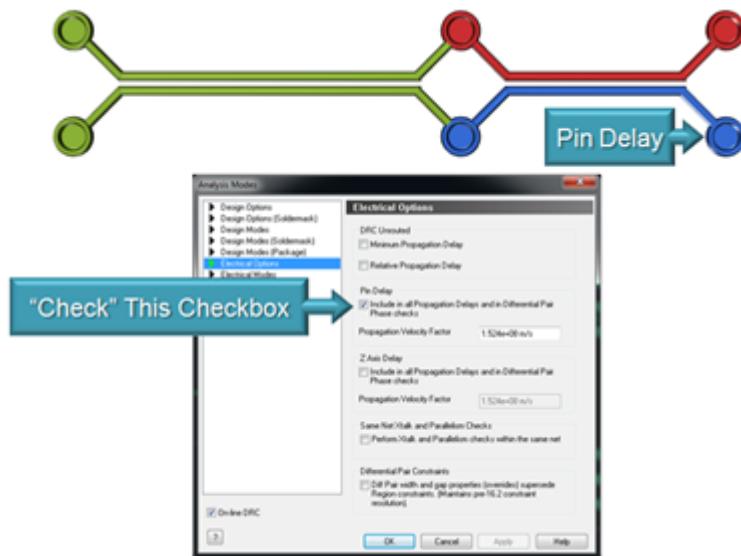
Figure 1-21 System, DRC Mode and Smart Mode view of Differential Phase Issues



The problem is that there is pin delay on the lower right pin that is causing the pair to be out of phase in this pin pair. This can be only detected if pin delay checking is turned on in Constraint Manager and there are pin delay properties or models on the pins of the nets.

The next figure shows the way phase issues with pin delays must be setup in Constraint Manager so that Timing Vision can properly use the pin delay properties.

Figure 1-22 Pin Delay impact on Smart Data



Timing Vision Phase Calculation Methods

Today's designs use differential pairs in ever increasing numbers. To make sure the signals of the pair arrive at the receiver at the same time, static phase constraints are a requirement. This static phase constraint is used by Timing Vision to generate the color scheme to solve this problem.

However, Timing Vision (Smart Mode) also calculates how far out of phase a net is based on its existing route solution at the time the smart goals were last updated. This data is also presented to you on the HUD (when using `slide`, `add_connect` and `delay tune` commands). The data tip show additional information to solve the static phase constraints in the design.

Timing Vision Phase Calculation Methods: Two Pin Nets

When a net is a simple two pin connection (driver-to-receiver), phase calculation is very simple. The static phase constraint is divided in half to create the Net Phase Tolerance. This tolerance ensures that the pair meets its phase constraints while adding the smallest amount of additional delay possible.

For example: $10 / 2 = +/- 5\text{mils}$ is the allowed Net Phase Tolerance range.

The Static Phase Constraint of the net is then split equally between all of the pin-pairs in the net to create the Pin-Pair Phase Goal.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

For this example: $10 / 1 = 10$ mils is the calculated Pin-Pair Phase Goal.

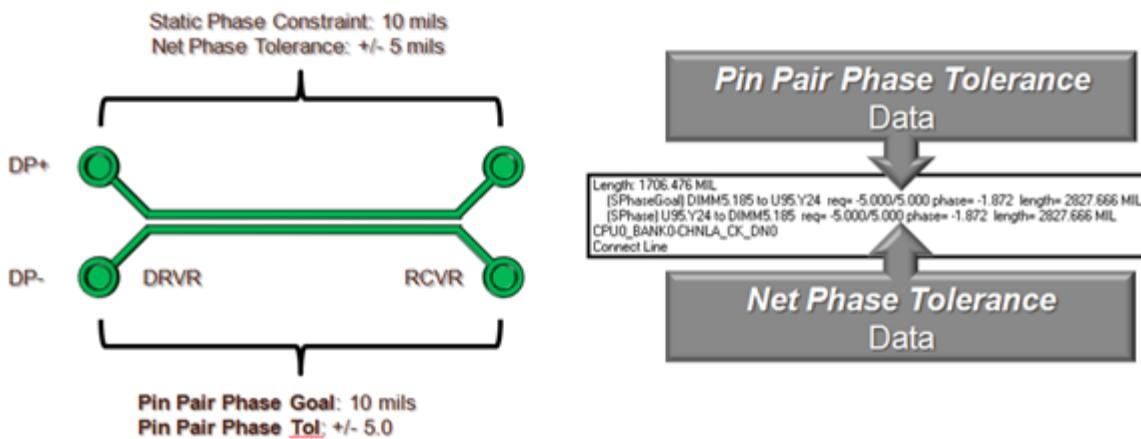
This Pin-Pair Phase Goal is then divided in half to create a tolerance range for this pin-pair. This calculated range is called the Pin-Pair Phase Tolerance (Tol).

If you hover over one of the clines (negative in this example), the data tip show you information about the net and the pin-pair you are hovering over. In this example the second line in the data tip represents the Net Phase Tolerance and how good or bad with respect to satisfying the Static Phase constraint for this net. The first line of the data tip reports the current pin-pair and Pin Pair Phase Tol. The data tip also shows how good or bad with respect to satisfying the Static Phase constraint for this pin-pair.

For this example: $10 / 2 = +/- 5.000$ mils is the calculated Pin-Pair Phase Tol.

If assume that one half of the pair is 1000 mils long, then the mate must be in the range of 995 to 1005 mils long to satisfy the static phase constraint.

Figure 1-23 Single Pin-Pair (2 Pin Net) Phase Calculation Example
Topology Example **Data-Tip Representation**



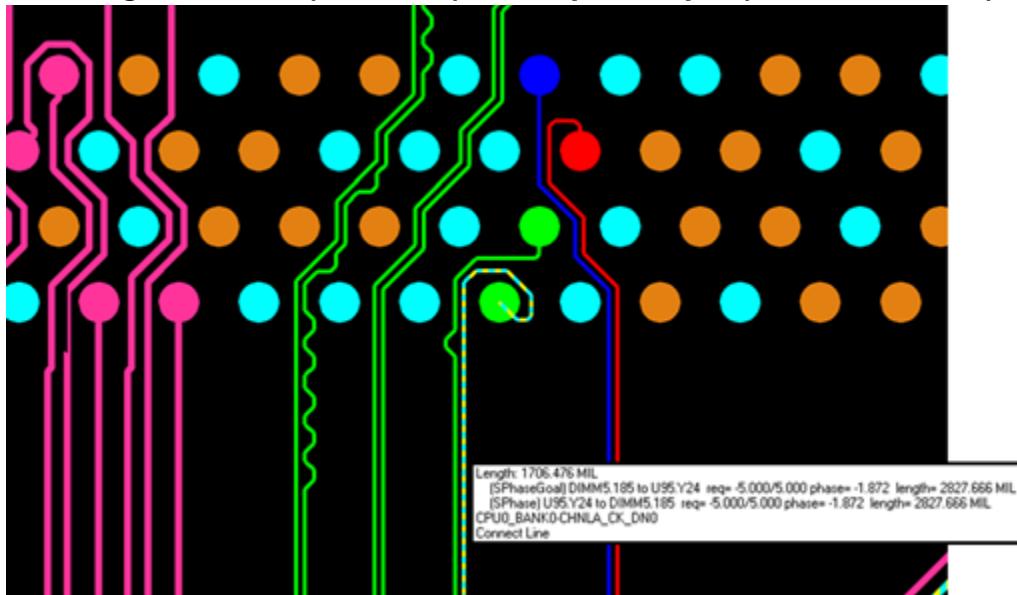
An Example:

Assumed Net Length (DP+): 1000 mils
Allowed Range (DP-): 995 -to-1005 mils

The flowering is a data tip example of a two pin net that meets its static phase constraints. The 10 mil tolerance was split in half to a +/- 5 constraint. This gives (interactively) and the auto-interactive tools (Auto-Interactive Phase Tuning) a range to get the + and – halves of the pair to fall within.

The data tip also displays the current imbalance of the pair. In this case the leg you are hovering over (striped) is -1.872 mils from being perfectly the same length in relation to its mate.

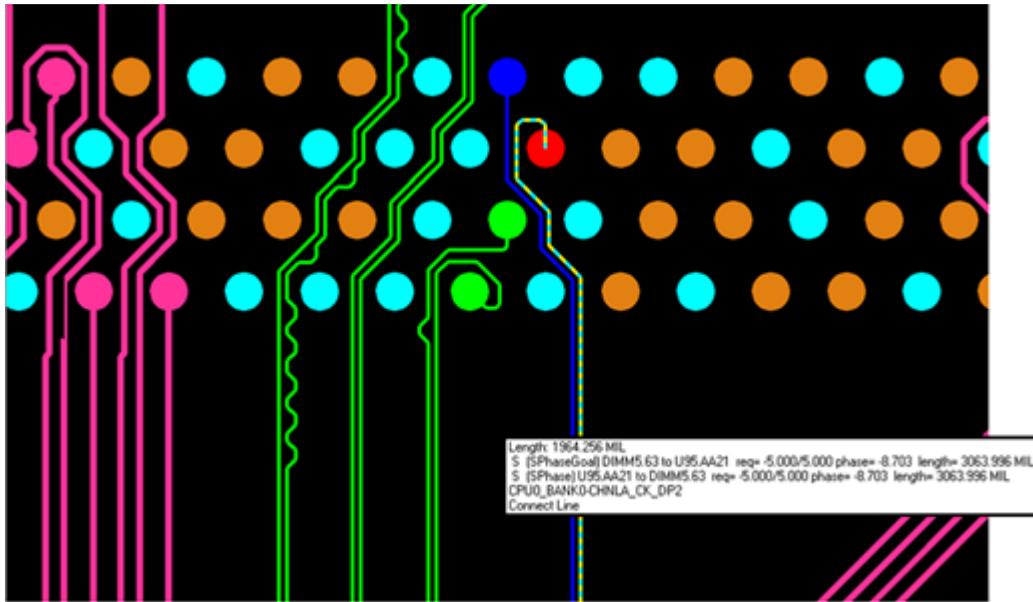
Figure 1-24 Single Pin-Pair (2 Pin Net) Data tip Example (Phase Satisfied)



Next is the data tip example of a two pin net that does not meet its static phase constraints. The 10 mil tolerance was split in half to a +/- 5 constraint.

The data tip also displays the current imbalance of the pair. In this case the leg you are hovering over (striped) is -8.703 mils from being perfectly the same length in relation to its mate. Since -8.703 is outside the calculated Pin-Pair Phase Tolerance of -5.000/5.000, this pair is failing its phase constraints.

Figure 1-25 Single Pin-Pair (2 Pin Net) Data tip Example (Phase NOT Satisfied)



Timing Vision Phase Calculation Methods: Multi Pin Nets

When phase is calculated for multi-pin nets, the nets Static Phase Constraint is divided in half to form the Net Phase Tolerance of the nets.

However, the Static Phase Constraint of the net is divided by the number of pin-pairs in the net to make a Pin-Pair Phase Goal for each pin-pair in the net. In this example, the net has two pin-pairs and a 10 mil Static Phase Constraint.

For example: $10 / 2 = 5\text{mils}$ is the calculated Pin-Pair Phase Goal.

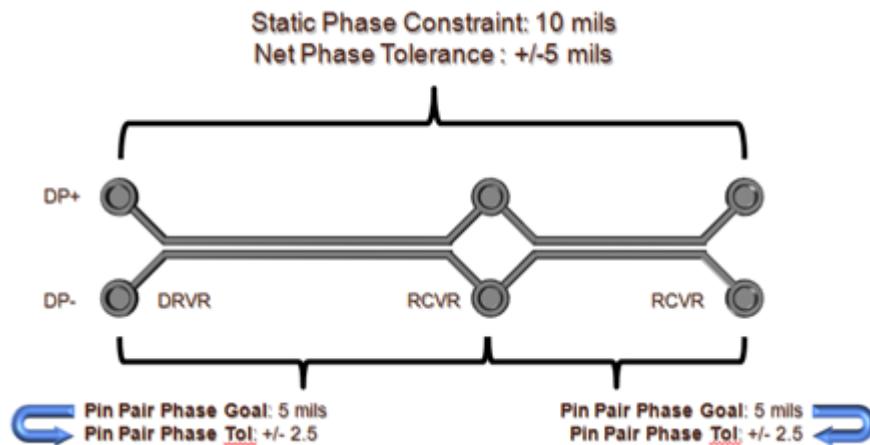
The Pin-Pair Phase Goal is then divided in half to form the Pin-Pair Phase Tolerance range for the manual (interactive) and auto-interactive tools to try and solve.

For this example: $5 / 2 = +/- 2.5 \text{ mils}$.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

Figure 1-26 Multi Pin-Pair (3 Pin Net) Phase Calculation Example



In this view, the updated data tip phase reporting shows that both nets are in phase. This new information on the data tip can help you find out how good or bad the phase matching or imbalance is.

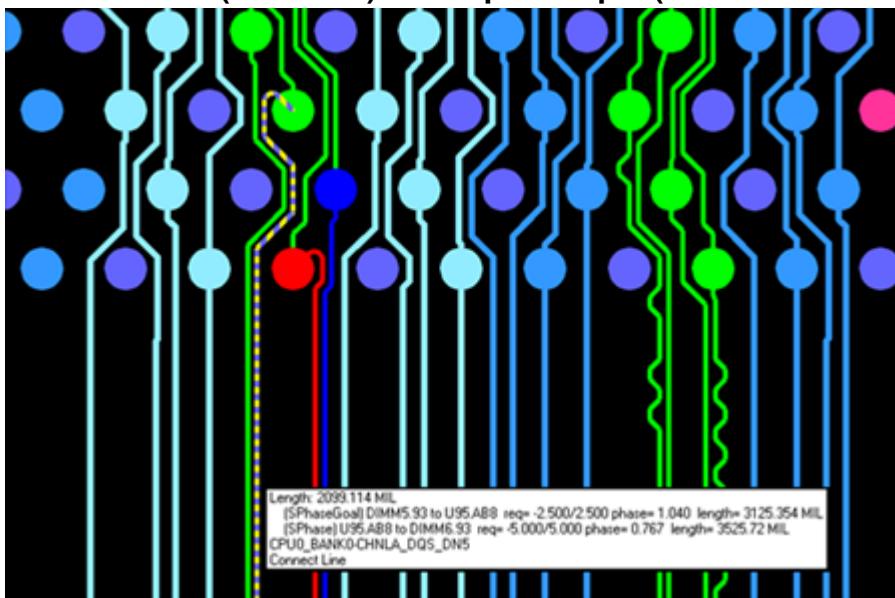
Next is the data tip example of a multi pin net that meets its static phase constraints. The 10 mil tolerance was split in half to a +/- 5 constraint. This gives you (interactively) and the auto-interactive tools (Auto-Interactive Phase Tuning) a range for the halves of the pair to fall within.

For example: $10 / 2 = 5.000$ mils is the calculated Pin-Pair Phase Goal.

This Pin-Pair Phase Goal is then divided in half to create to create the Pin Pair Phase Tolerance (Tol) which is shown on line one of the Data-Tip.

For this example: $5 / 2 = +/- 2.500$ mils is the calculated Pin-Pair Phase Goal.

Figure 1-27 Multi Pin-Pair (3 Pin Net) Data tip Example (Phase Satisfied)



If you hover over a net that has multiple pins the data tip show similar information as it did for the two pin net data tip. In this case however, it is showing information for whichever pin-pair you are hovering over.

In this example the Pin-Pair Phase Tol and imbalance for the U95.AB8 to DIMM5.93 connection on the first line is reported. While the second line reports the overall compliance of the net and it's Net Phase Tol and imbalance (if any).

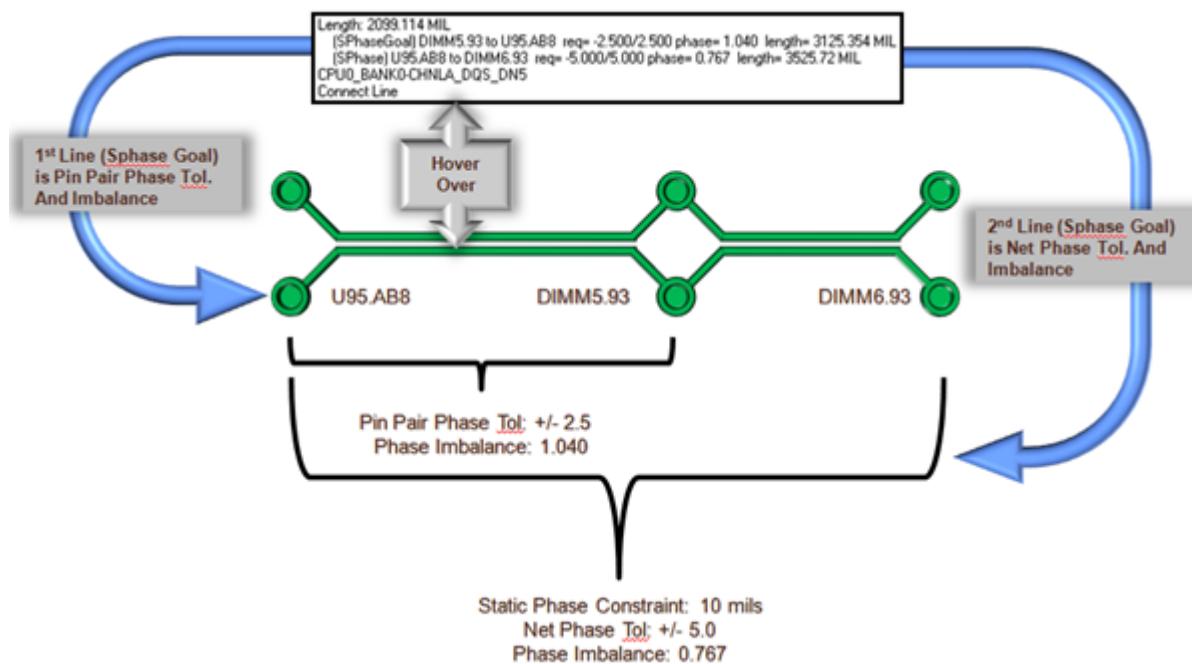
The data tip also displays the current imbalance of the pin-pair on the first line of the data tip. In this case the leg you are hovering over (striped in the previous image) is 1.040 mils (long) from being perfectly the same length in relation to its mate.

The current route solution is also satisfying its Net Phase Tol constraint and is only 0.767 (long) from the driver to the last receiver.

Allegro X User Guide: Allegro X Timing Environment

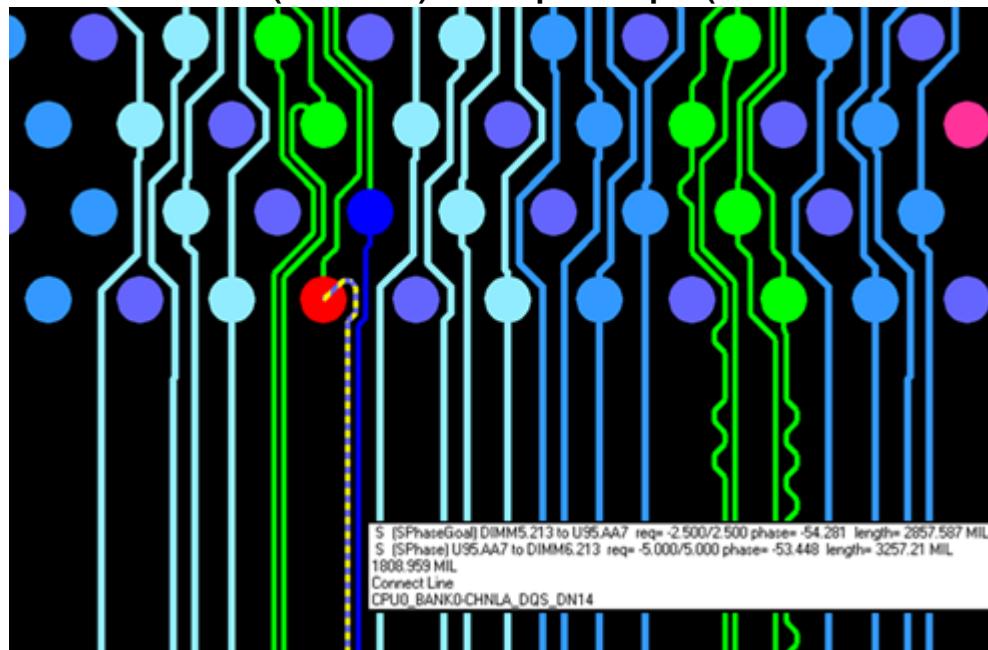
Allegro Timing Environment (ATE)

Figure 1-28 Multi Pin-Pair (3 Pin Net) Data tip Example



In the following image a data tip Phase Representation is shown when a multi-pin net is not in phase. In this example the Pin Pair Phase Tol and imbalance is: -54.281; while the Net Phase Tol and Imbalance is -53.448.

Figure 1-29 Multi Pin-Pair (3 Pin Net) Data tip Example (Phase NOT Satisfied)

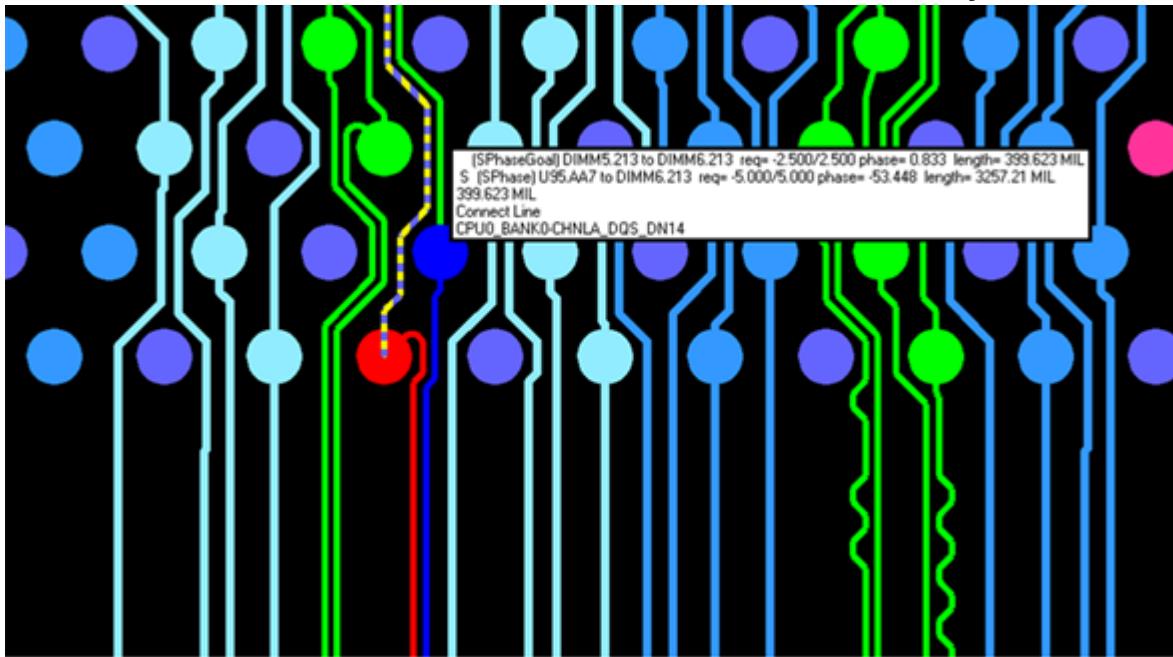


Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

To find out how good the good green pin-pair is, hover over it to get its data tip information. You can see from the data tip that the Pin-Pair Phase Tol and imbalance are very small only .833 mils. Therefore, this part of the net is very good and you need to fix the problem in the red pin-pair.

Figure 1-30 Examine the Other Pin-Pairs to see how Good/Bad they are



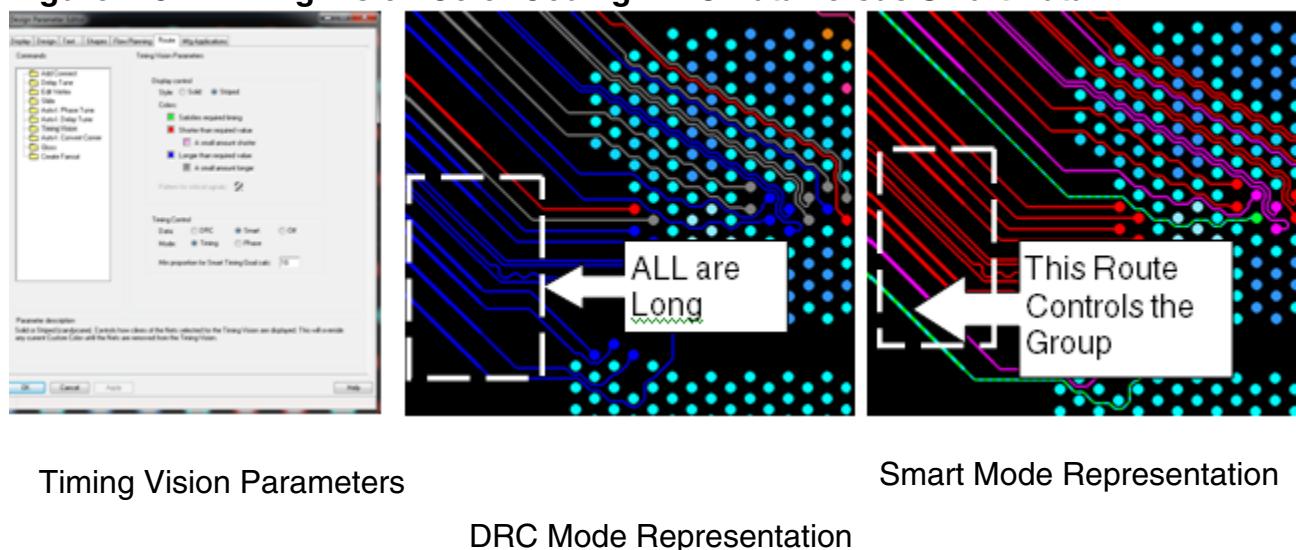
Timing Vision Use Model

Since Smart and DRC modes use different methods to generate the color code. The color coding representations for the same interface depends on the mode.

In the following image, the Timing Vision color code pattern is shown on the left. DRC Mode image is displayed in the middle, while the Smart Mode representation is displayed on the right.

DRC Mode considers many of them are long (dark blue) which could be a difficult task to resolve especially when the target (dark blue differential pair in the center) also appears long. But in the right hand image, the green member is the longest member since it is on the outer edge of the routing. This causes all the other members of the Timing Group to turn red since they are shorter than the green member. This means all of the red members need additional length or delay added to them to make them turn green. This includes the target which is a differential pair in the middle of routing in the image.

Figure 1-31 Timing Vision Color Coding: DRC Data versus Smart Data

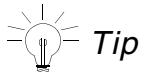


There is also a preferred use model for Timing Vision that relates to a traditional design flow for solving designs delay constraints. This delay strategy includes following steps:

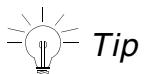
- Solve Phase Constraints: both Dynamic and Static
- Solve the General Delay problem
- Final DRC checks

Timing Vision Setup

To setup the environment for the first time you use ATE Timing Vision in the current session, you need to create the Timing Group and Update the Goals. This simple process should be repeated each time you start the design. Especially the Timing Group and Update Goals sections as these are not stored in the database. The color scheme is generally done only once.

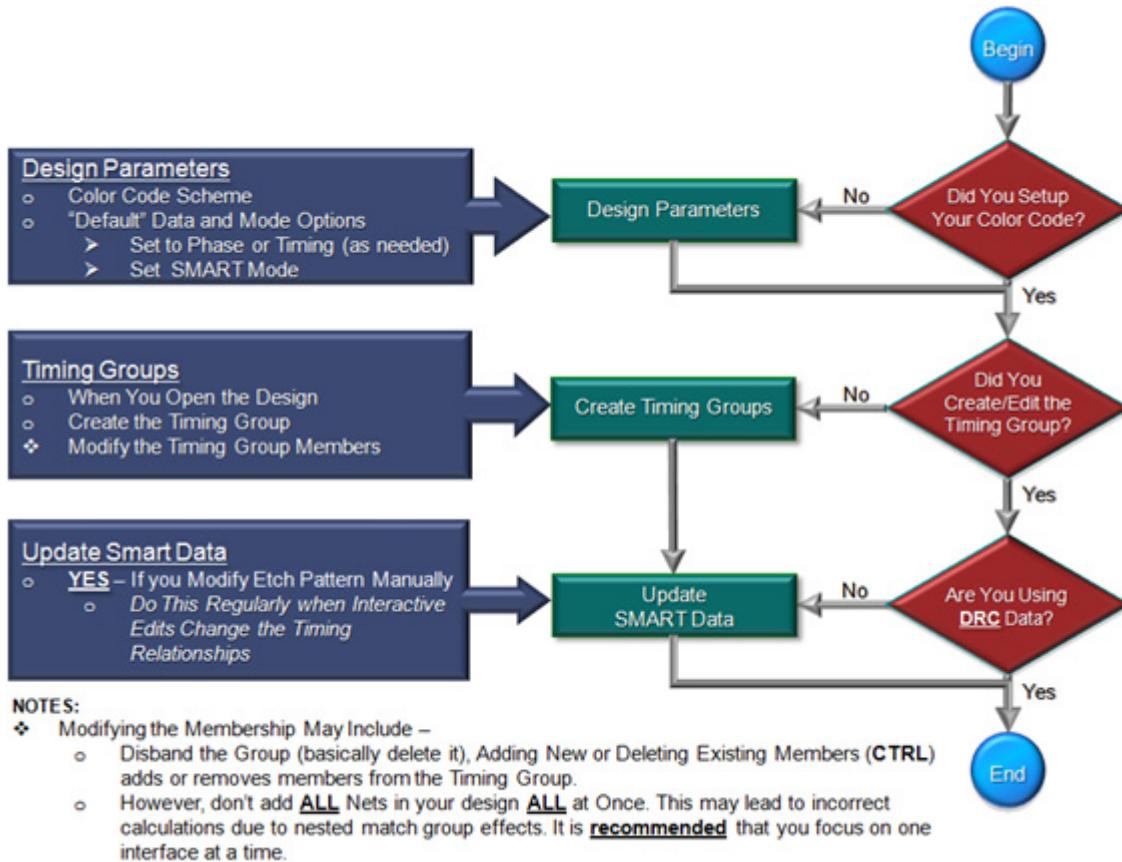


Tip
If you add or remove membership to the Timing Group you do not need to Update the Goals. Smart Data is calculated for the entire design when you execute the command. This may take a little time depending on the number of constrained nets in the design.



Tip
If you make manual etch edits, you could modify nets that are considered Critical Signals by Timing Vision. If you do or you want to modify some net and make it the new critical signal (by making it the longest) then you need to update the Goals. Otherwise you will not get the desired long net to properly help color code the remaining nets in the interface.

Figure 1-32 Timing Vision Initial Setup



Timing Vision Use Model

There are several possible use models for Timing Vision:

1. Select the entire interface and put it in the Timing Group. This will let you see/analyze the entire timing problem. It may help you decide that you need to start tuning in the center of the BGA first pushing space open using slide or some other tools before you start delay compensation.
2. If your routing is more z distributed through the cross-section, then there is not so much push/shove that has to be done. So adding a few match groups at a time may be easier. You can focus on the delay problem in smaller pieces.
3. If you have partial constraint definition, start by adding the match groups that are complete for now. Delay tune the match groups and add additional match groups as you get more constraints. In this use model you need to update the Goals as new constraints

are added to the design. Otherwise the tools will not be able to color code and delay compensate properly.

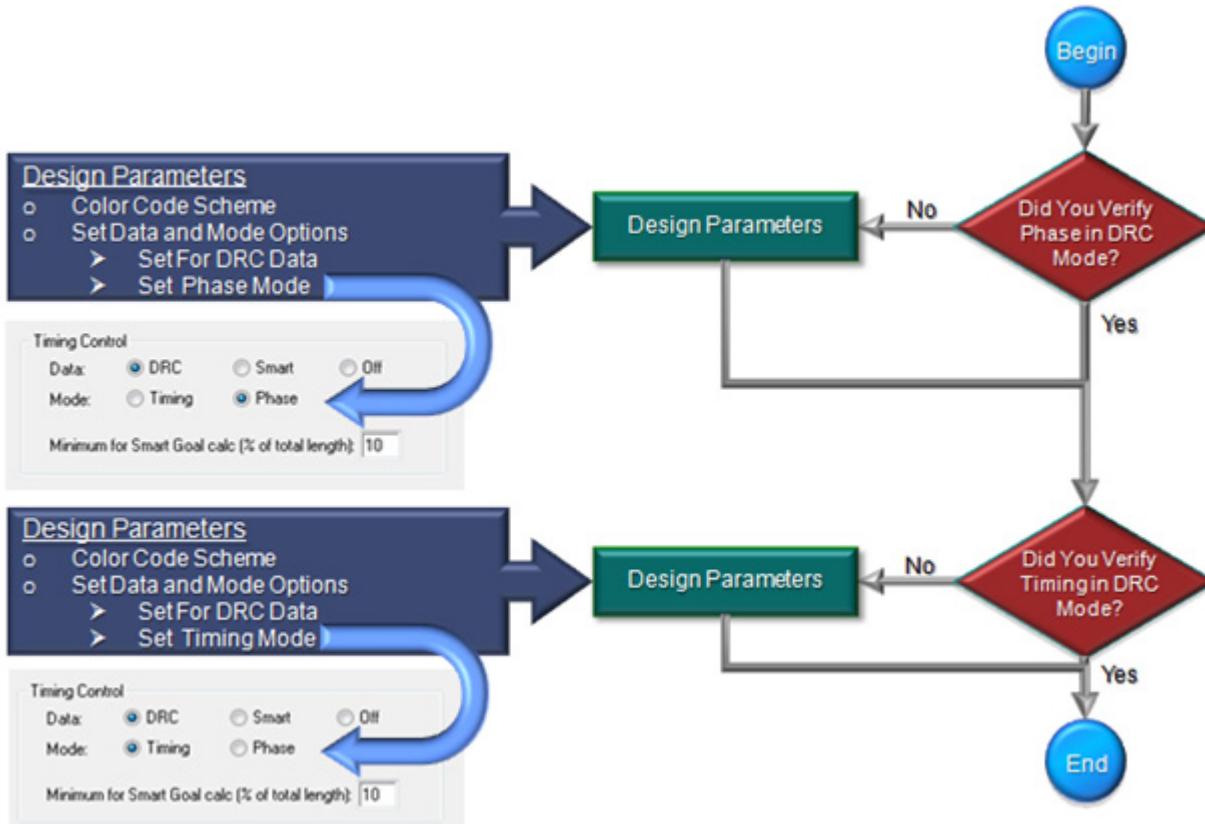
Timing Vision Use Model: Final Checks

Calculations and color code of Timing Vision are based on the actual route data and constraints found in Constraint Manager. But there is a possibility that you modified a constraint or changed a route path while not being in Timing Vision mode. Therefore it is always best to make one final check before you assume that delay has been satisfied.

To do this set the Data type for phase first, let the screen update the color code. Visually scan the canvas and look for any non-green items.

When you are sure the results are good, change the system to Timing Mode and again verify that the results are good, all green.

Figure 1-33 Change Back to DRC Mode for Final Checks



Auto-Interactive Phase Tuning (AiPT)

AiPT is an interactive command that specifically focuses on solving differential static and dynamic phase constraints by:

- Using information calculated by Timing Vision for how long or short the appropriate phase imbalances within the pair are. These phase mismatches are the amount that AiPT will try to add/remove from the differential pair.
- Using a prioritized list of user defined phase compensation techniques found on an options form that lets you modify them during command execution.
- Modifying the selection set using specialized algorithms focused on the phase problem only.

Note: When the AiPT command has finished executing, it may/not achieve 100% due to space limitations but generally it will modify the pair and produce high quality results wherever possible.

You can invoke the command from

Route – Unsupported Prototypes – Auto-Interactive Phase Tuning

AiPT Options

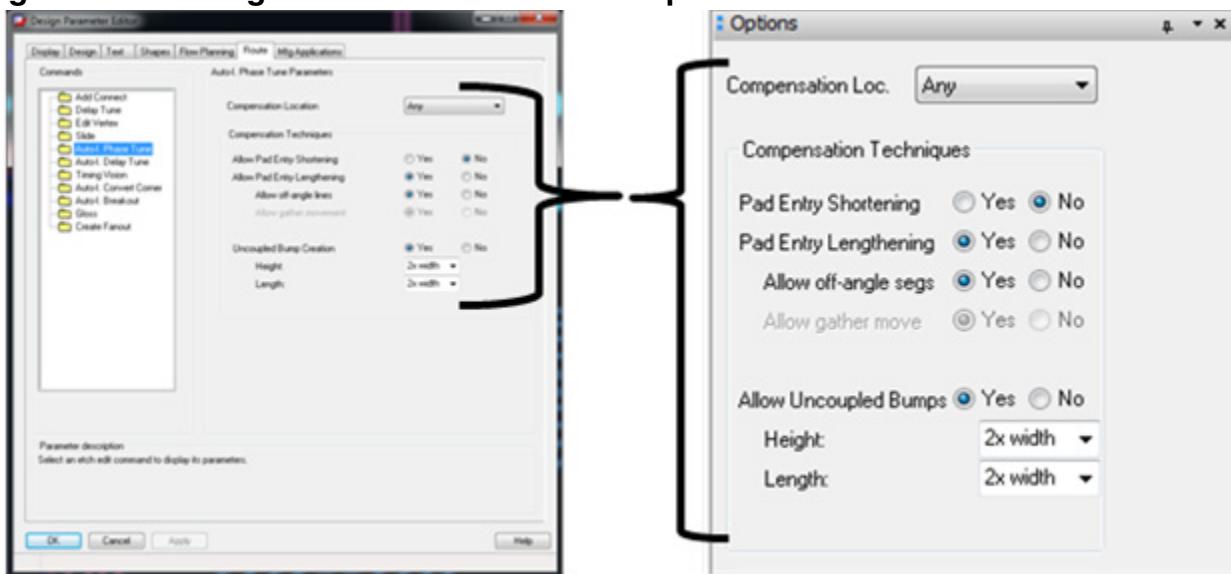
AiPT uses a set of compensation techniques that are considered industry standard. The tool is pre-defined with good values, that produce good quality results.

You can find these default Compensation Techniques in the *Design Parameters* dialog box as well as on the Options tab.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

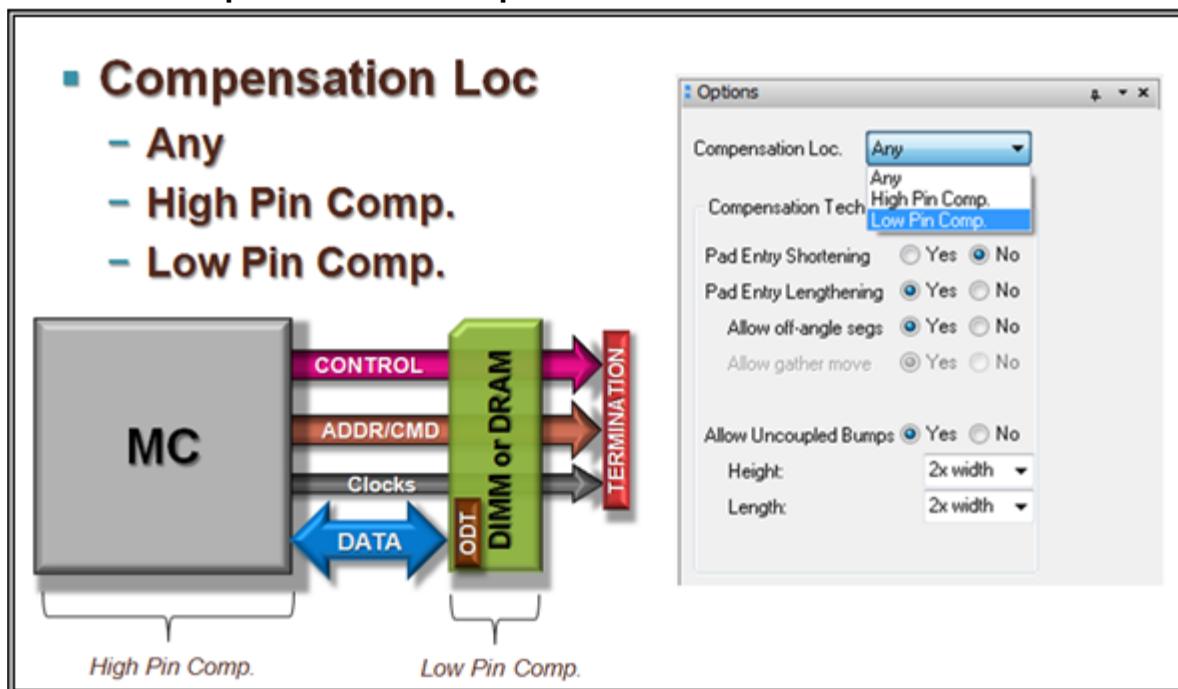
Figure 1-34 Design Parameters or Canvas Options- AiPT Parameters



Compensation Loc

Determines the location of where AiPT prefers to focus its compensation techniques. There are three choices - Any, High Pin Comp. and Low Pin Comp.

Figure 1-35 Compensation Techniques



In the example, the memory controller (MC) is the *High Pin Comp.* and the DIMM or DRAM is the *Low Pin Comp.* Definitions of the locations and how AiPT works in the various modes are defined as:

Any

- This option lets the tool place the allowed compensation technique preferably at either end of the differential pair when trying to satisfy static phase constraints. There are no restrictions regarding pin/via pad modifications and on the bump location depending on the Compensation Techniques selected.
- When working with Dynamic Phase constraints it could put phase compensation bumps anywhere along the cline paths from pin-to-pin when the *Allow Uncoupled Bumps* techniques is set to Yes.

High_Pin_Comp

- Specifies that only the end of the diff pair that connects to the highest pin count component can be modified in the pin/via pad entry area. For example, the tool can modify the BGA end of the memory system.

Low_Pin_Comp

- Specifies that only the end of the diff pair that connects to the lowest pin count component can be modified in the pin/via pad entry area. For example, the tool can modify the DIMM end of the memory system.

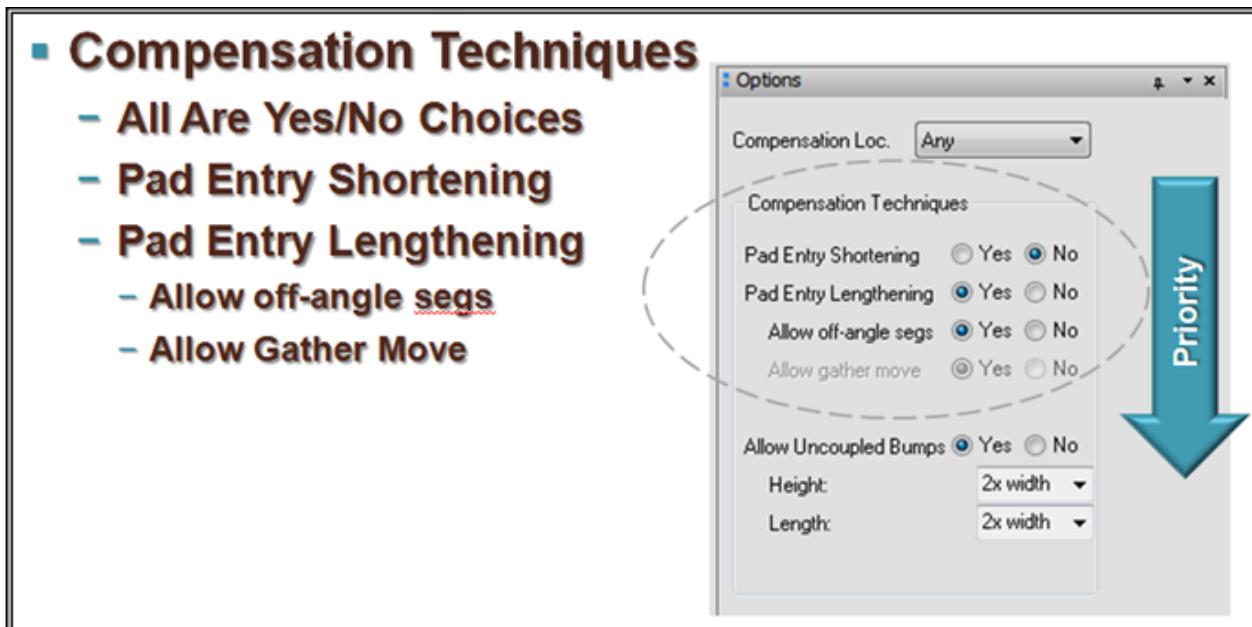
Note: Use of this option may reduce the ability of the tool to satisfy the phase requirements especially if the *Compensation Techniques* are restrictive or there is not enough room to perform the appropriate compensation techniques.

Note: Use of *High_Pin_Comp.* and *Low_Pin_Comp.* options do not affect the *Allowed Uncoupled Bumps* locations for solving Dynamic Phase.

Compensation Techniques

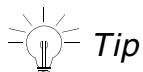
This section describes the techniques that the tool use at the appropriate locations set by *Compensation Loc.*

Figure 1-36 Compensation Techniques



These techniques are generally the best known behaviors that are widely used. They are all set to either a *Yes* or *No* state and do not have sub-options.

1. The order of the techniques on the Options tab are prioritized.
2. When using these options and the *Compensation Loc* is set to *Any*, quite often the area you are looking at may not be where the compensation is actually done. So, when you phase adjust a pair or some pairs make sure you look at the full length to see where the compensation happened.
3. Turning on (*Yes*) or off (*No*) any of the following compensation techniques may impact the ability of the tool to meet phase constraints.
4. Use of these compensation techniques may increase or decrease the amount of uncoupled length in the differential pairs. It is important to review the results when solving differential phase constraints.



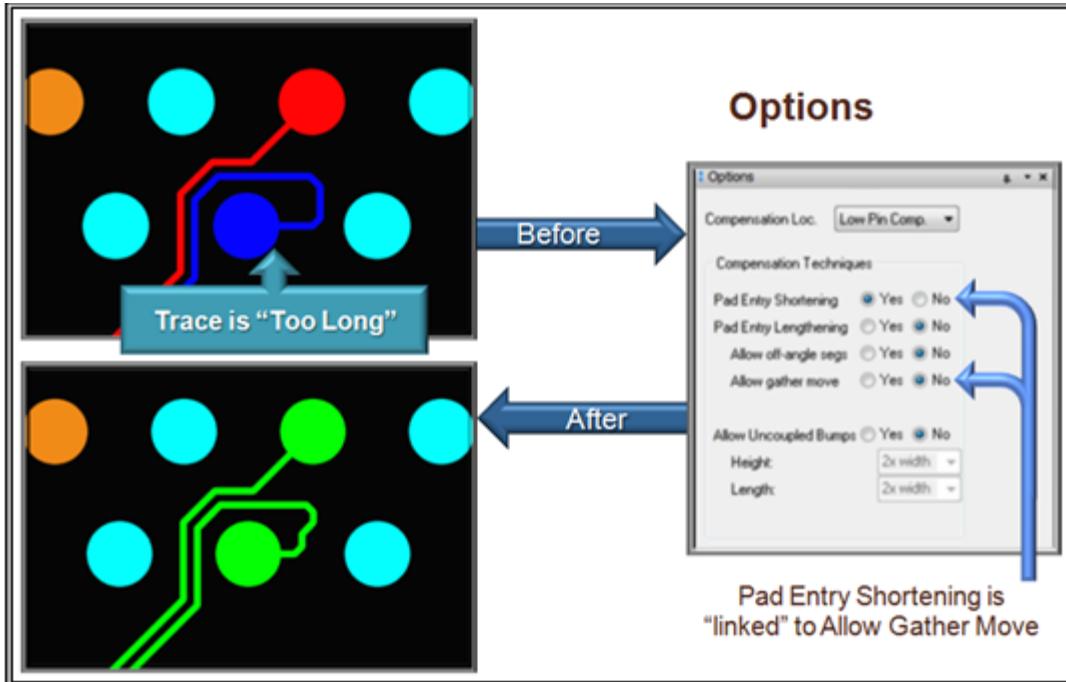
Tip
When phase does not appear satisfied, try using the *Oops* command from the right-click menu. Change a setting on the Options tab and run the pair again. This may provide the phase compensation required or route pattern desired.

Pad Entry Shortening

This technique enables (*Yes*) or disables (*No*) the tools ability to shorten the longer half of the pair by making modifications to the existing route pattern only in the region from the gather point to the pin or via as it tries to match the phase imbalance between the two halves of the pair. This technique uses the *Allow off-angle segs* technique if enabled.

Note: The following figure shows a possible use model for *Pad Entry Shortening*. It also points out a link between the *Pad Entry Shortening* compensation technique and the *Allow gather move* option. If you set *Pad Entry Shortening* to *No*, it automatically sets *Allow gather move* to *No*. However, if you set *Pad Entry Shortening* to *Yes*, it does not automatically set *Allow gather move* to *Yes*.

Figure 1-37 Pad Entry Shortening

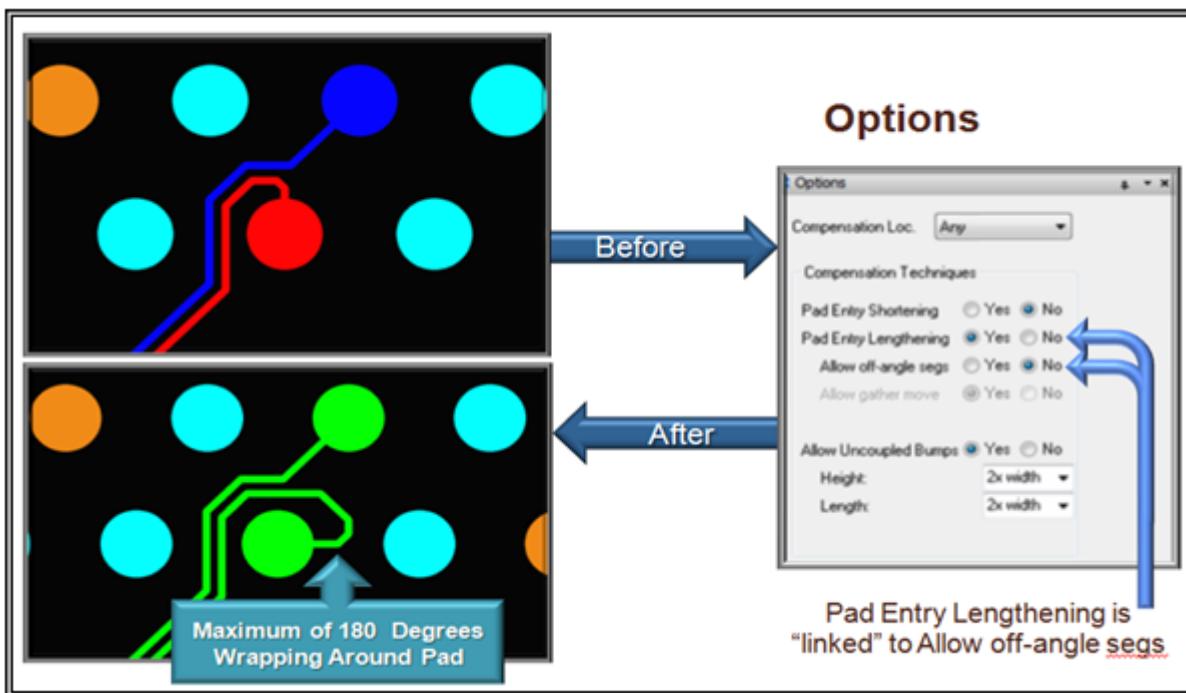


Pad Entry Lengthening

This technique enables (*Yes*) or disables (*No*) the tools ability to lengthen the shorter half of the pair. It focuses on the region from the gather point to the pin or via as it wraps around the pad in order to match the phase imbalance between the two halves of the pair. This technique uses only 45 degree segments in the wrap but it will use the *Allow off-angle segs* technique (if enabled).

Note: *Pad Entry Lengthening* will not wrap more than 180 degrees around the pad.

Figure 1-38 Pad Entry Lengthening Example

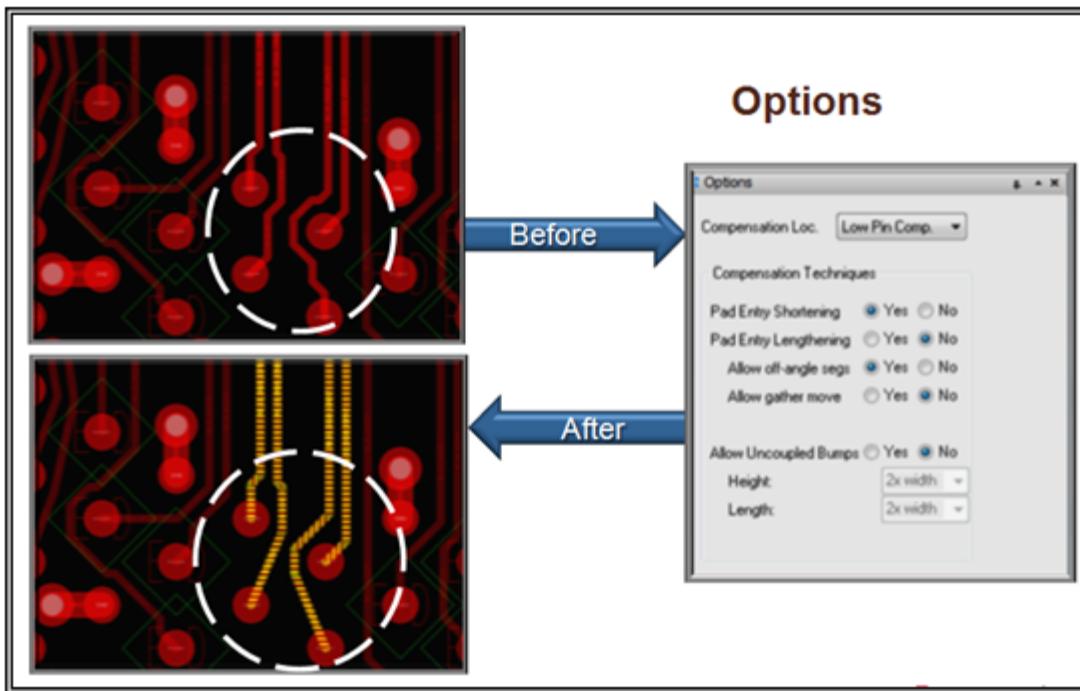


Allow off-angle segs

This option allows the tool to create off-angle (non 45/90 degree) pad entry segments only when trying to solve the phase compensation problem. This can be done in tight pin fields, or when slight shortening of one half of a pair is required.

Note: This technique need to have either *Allow off-angle segs* and/or *Allow gather move* set to Yes for this option to be available for use.

Figure 1-39 Allow off-angle segs example



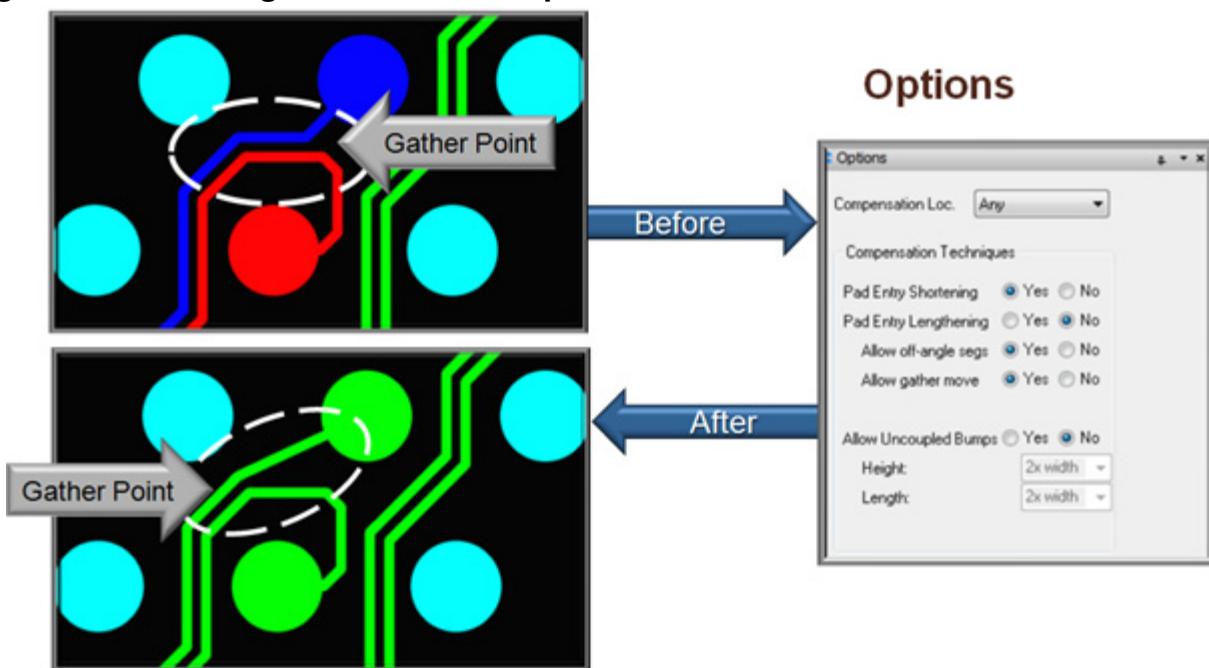
Allow Gather Move

This option allows the tool to modify the existing gather point of the differential pair. However, it must have either *Allow off-angle segs* and/or *Allow gather move* set to *Yes* for this option to be available for use.

In the following figure this option is used with *Pad Entry Shortening* to make the long leg (dark blue) shorter by entering at any angle and moving the gather point.

Note: Use of this technique will increase the amount of uncoupled length in the interconnect solutions. It is recommended to recheck the design for uncoupled length DRC markers when phase adjustment is done.

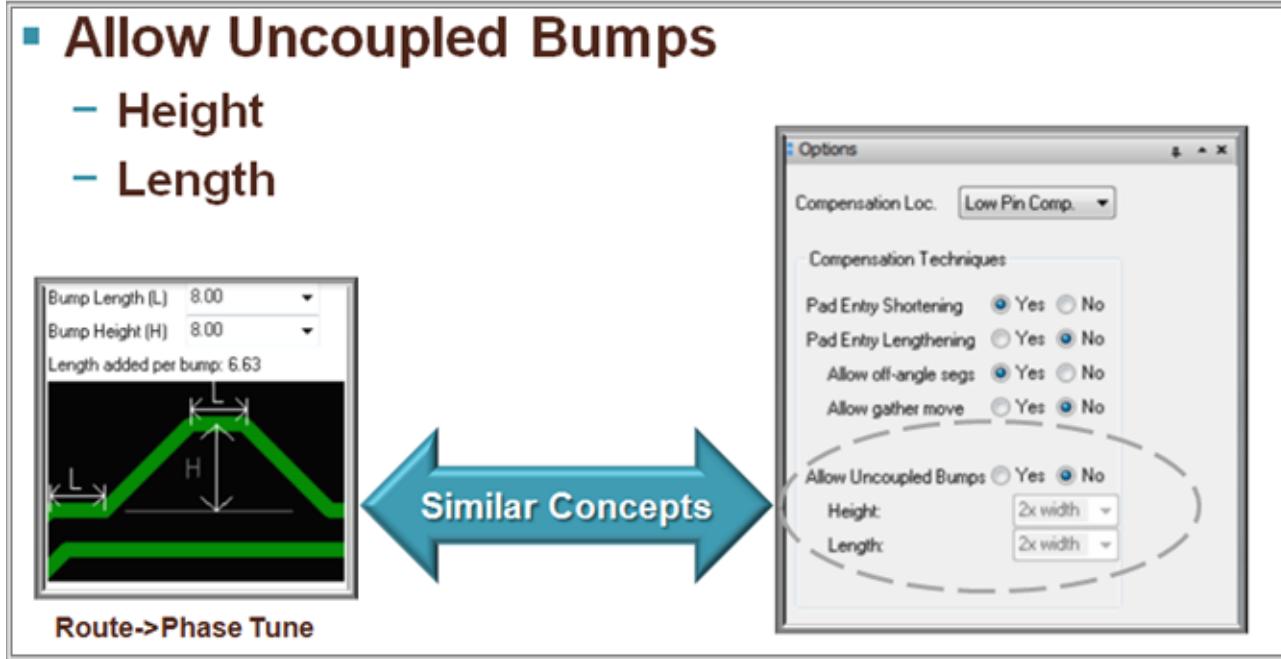
Figure 1-40 Allow gather move Example



Allow Uncoupled Bumps

This option and settings tell the tool to put phase compensation delay bumps into the clines to try and bring the pair within tolerance.

Figure 1-41 Allow uncoupled Bumps



The values that create the bumps are user definable and the tool tries to create as many bumps as needed to meet phase constraints. During execution AiPT will not push existing traces or vias to make space for phase bumps and it will not create DRC errors to existing traces or vias.

In order to define the bumps, there are two options -

Height

This key-in/pull-down value controls the size or distance that the delay bump will spread the pair apart. It is similar to the manual version of phase adjustment found in the *Route – Phase Tune* command. Its values can be specified in either line width or database units. Default units are in terms of line width (n X width).

Length

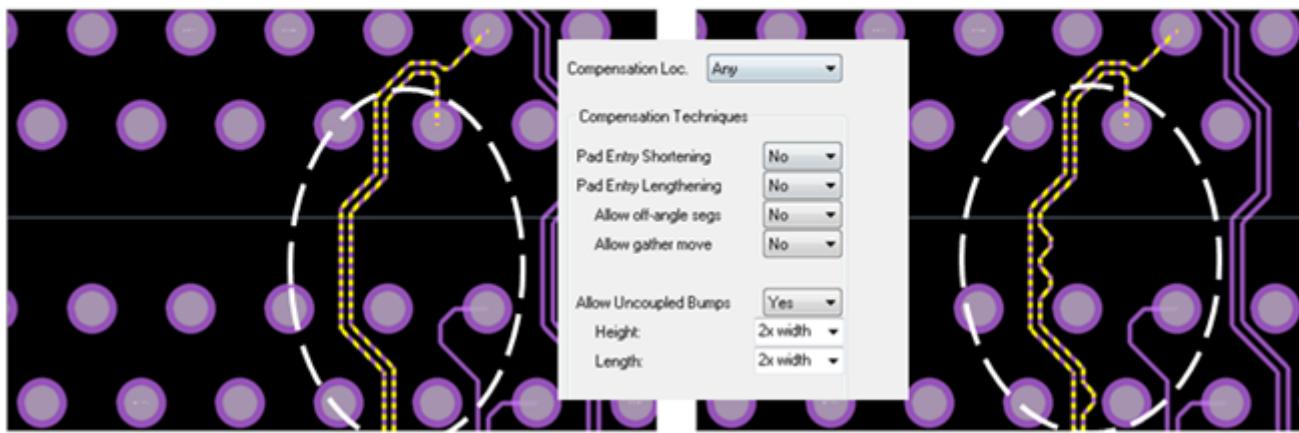
This key-in/pull-down value controls the length of each delay bump created. It is similar to the manual version of phase adjustment found in the *Route – Phase Tune* command. Its values can be specified in either line width or database units. Default units are in terms of line width (n X width).



Using the Allow Uncoupled Bump technique will add uncoupled sections into the differential pair. Each uncoupled section length gap is controlled by the Length and Height values set above. The tool will continue putting phase bumps into the pattern even if they create uncoupled violations.

Figure 1-42 Uncoupled Bump Example

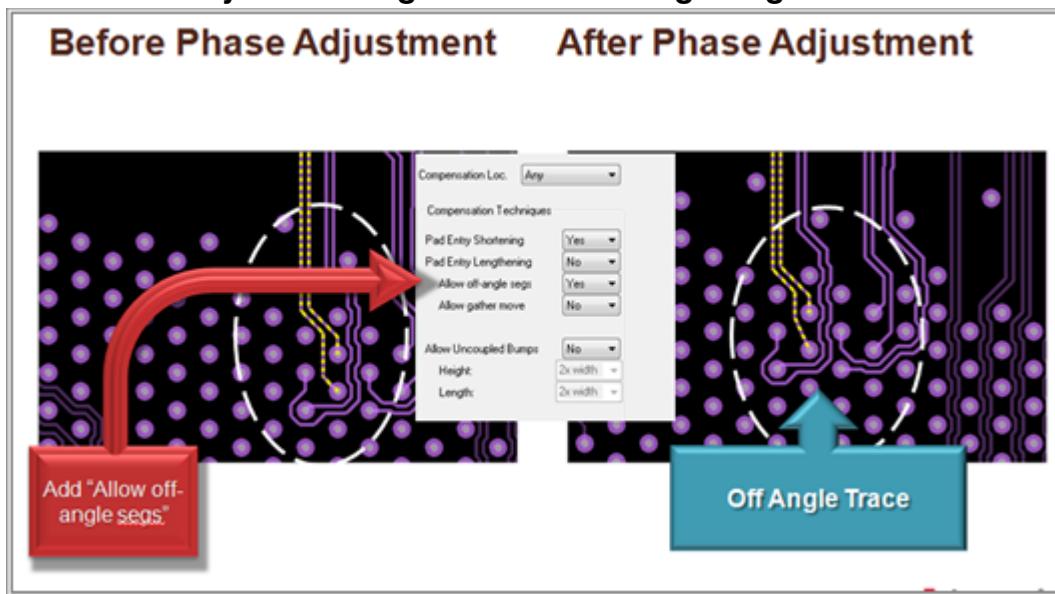
Before Phase Adjustment After Phase Adjustment



Combining Effects of Options

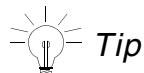
When you select multiple options at the same time, you can get the techniques to work together and change the behavior of the tools.

Figure 1-43 Pad Entry Shortening and Allow off-angle segs



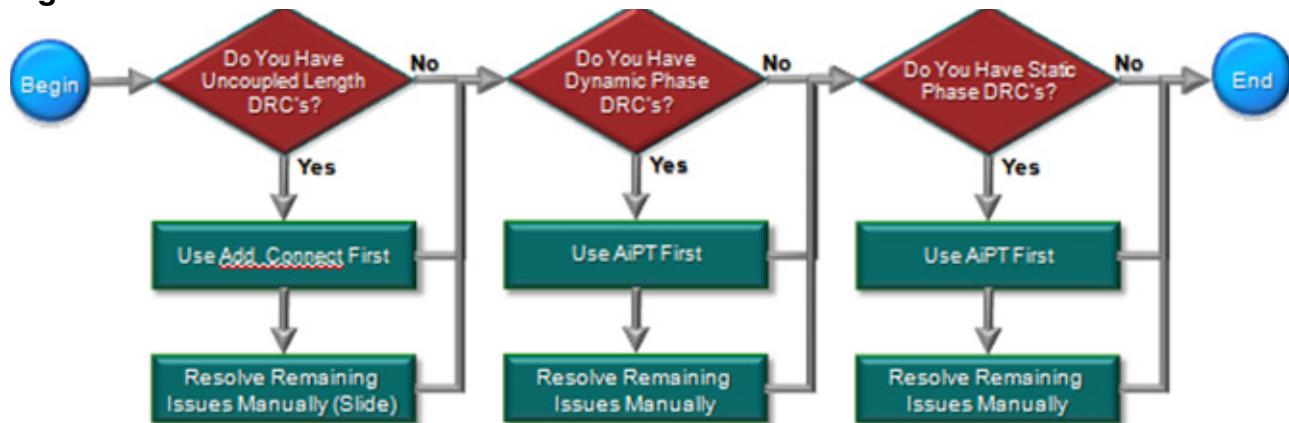
AiPT Use Model

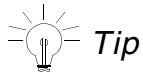
AiPT is focused on solving the phase problem. It does not solve uncoupled length and in fact it can increase it if used improperly. Depending on the delay to solve (Dynamic, Static or both) you can set the options and apply them.



It is recommended that you resolve any uncoupled length DRC error markers before using AiPT. When you have solved all of your Differential Phase issues, run DRC to resolve any uncoupled length DRC errors created by AiPT.

Figure 1-44 Differential Phase Flow Chart



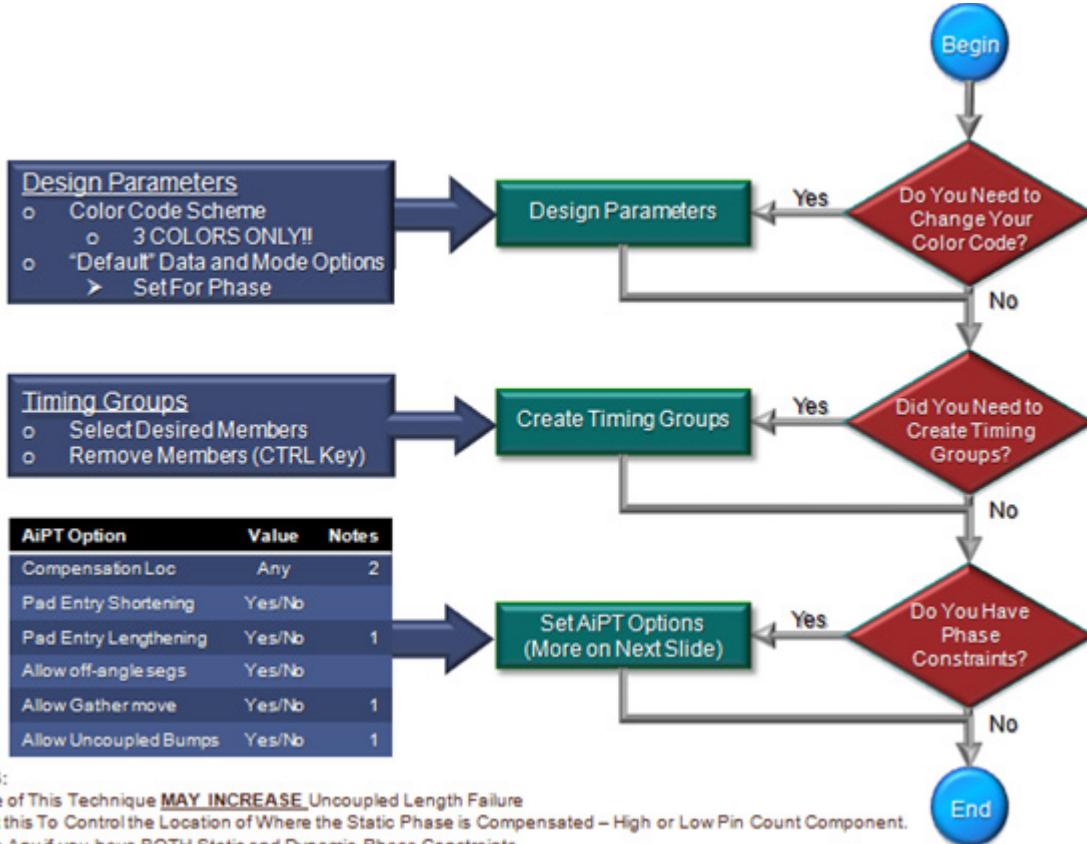


When you have both Dynamic and Static phase requirements it is recommended that you enable all the compensation techniques AiPT offers and run the tool.

Timing Vision Use Model: AiPT Setup

- AiPT may get the phase right but may increase uncoupled length depending on the selected options. This is due to some of the techniques AiPT can use to solve phase also has a negative side-effect of making uncoupled length longer, such as *Pad Entry Lengthening*, and *Allow Gather Move*.
- In phase mode, Timing Vision is a three color system. There is no *A little shorter* or a *A little longer* color code.

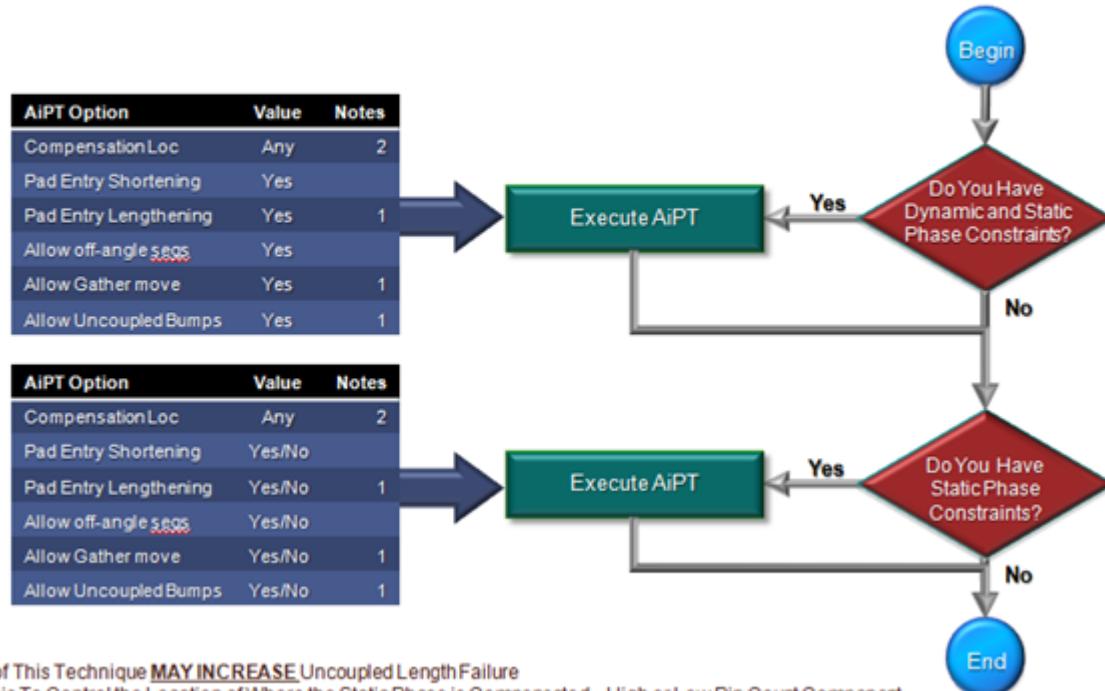
Figure 1-45 Timing Vision -Phase Setup



If you need to resolve some uncoupled length errors in the area around the pin field, allowing AiPT to modify this region with *Pad Entry Shortening*, and *Pad Entry Lengthening*, may

cause the creation of new uncoupled length errors. It is recommended to recheck for uncoupled length DRC's when allowing AiPT to use all of the options at the same time.

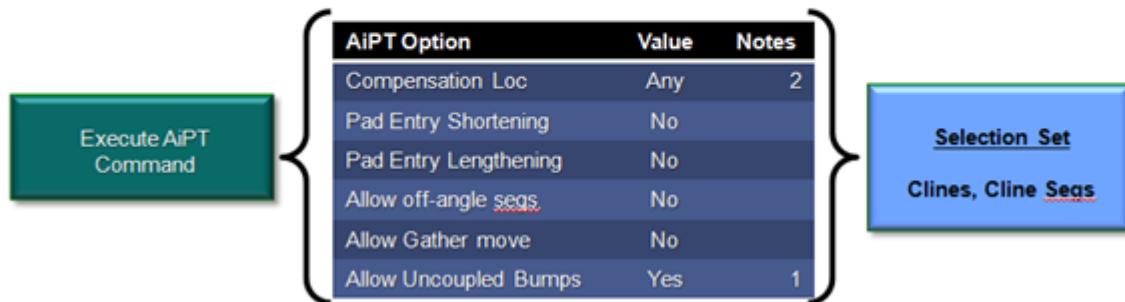
Figure 1-46 AiPT Preferred Use model and Options Selections
Routing Methodology
Solving Phase Dynamic & Static Simultaneously



If you prefer to solve Dynamic Phase only, the use model is the same, but the option choices are different and they are listed as:

Figure 1-47 Dynamic Phase Specific Setup

Routing Methodology Focusing AiPT to Solve Dynamic Phase Only



- Dynamic Phase is “Solved” by Inserting Bumps “Anywhere”
 - Relates to where the “DRC Line/Marker” Shows Up
- AiPT Needs to Insert Bumps in Clines “Anywhere”
 - Set Values as Shown to Turn OFF Compensation in Other Places
 - Normal Selection Set After Setting Options

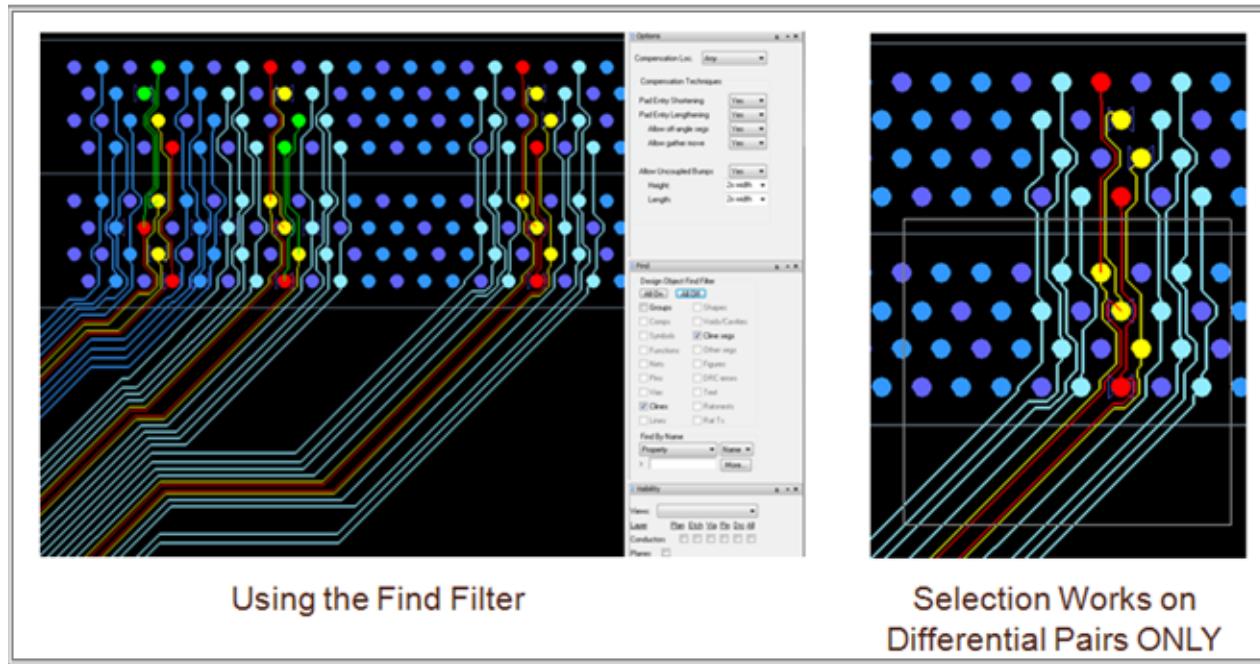
NOTES:

1. Use of This Technique MAY INCREASE Uncoupled Length Failure
2. Use Any because we are focusing on Dynamic Phase Constraints

AiPT Selection Techniques

AiPT works on selection sets. You can select clines, cline segs or groups in the Find Filter. You can also use the *Temp Group* to make selection of Net-Groups or user-defined custom Groups of rats and nets for specific selection control.

Figure 1-48 AiPT Selection Techniques



AiPT Feedback

When AiPT is run, there is information in the command window that gives some basic statistics on how well this attempt did. For example, in the following figure you can see 8 phase constraints (for 4 pairs), 6 phase violations (this means one pair is in phase) and 6 outside ideal range errors (none of the selected pairs are able to meet the phase constraint requirements).

Figure 1-49 Command Windows Statistics

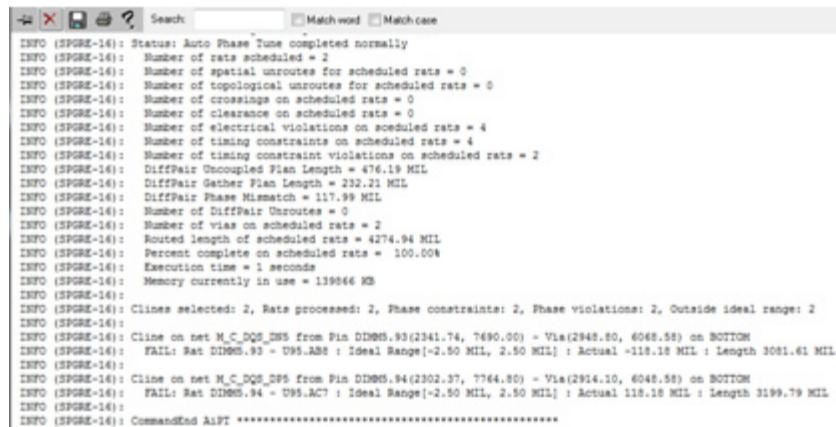
```
Command > last pick: 4137.590 8058.530
        No element found.
        Clines Selected: 8; Phase constraints: 8; Phase violations: 6; Outside ideal range: 6
        Clines Selected: 8; Phase constraints: 8; Phase violations: 0; Outside ideal range: 0
        Clines Selected: 8; Phase constraints: 8; Phase violations: 6; Outside ideal range: 6
Command >
```

For more extensive information, use the *File Viewer* option to see more statistics that may help you resolve an AiPT phase error.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

Figure 1-50 Example of Log File Detailed Output



The screenshot shows a terminal window with the Allegro X log file output. The window has standard operating system controls (minimize, maximize, close) at the top left. To the right of the controls are two checkboxes labeled "Match word" and "Match case". Below the title bar is a search bar with the placeholder "Search". The main area of the window contains the log file content, which is a series of informational messages from the Allegro X timing environment. The messages include statistics about scheduled routes, timing constraints, and specific route details for nets like M_C_DQS_DQS and M_C_DQMS_DQMS.

```
INFO (SPGRE-16): Status: Auto Phase Tune completed normally
INFO (SPGRE-16): Number of rats scheduled = 2
INFO (SPGRE-16): Number of spatial unroutes for scheduled rats = 0
INFO (SPGRE-16): Number of topological unroutes for scheduled rats = 0
INFO (SPGRE-16): Number of crossings on scheduled rats = 0
INFO (SPGRE-16): Number of clearance on scheduled rats = 0
INFO (SPGRE-16): Number of electrical violations on scheduled rats = 4
INFO (SPGRE-16): Number of timing constraints on scheduled rats = 4
INFO (SPGRE-16): Number of timing constraint violations on scheduled rats = 2
INFO (SPGRE-16): DiffPair Uncoupled Plan Length = 476.19 MIL
INFO (SPGRE-16): DiffPair Gather Plan Length = 232.21 MIL
INFO (SPGRE-16): DiffPair Phase Mismatch = 117.99 MIL
INFO (SPGRE-16): Number of DiffPair Unroutes = 0
INFO (SPGRE-16): Number of vias on scheduled rats = 2
INFO (SPGRE-16): Routed length of scheduled rats = 4274.94 MIL
INFO (SPGRE-16): Percent complete on scheduled rats = 100.00%
INFO (SPGRE-16): Execution time = 1 seconds
INFO (SPGRE-16): Memory currently in use = 139846 KB
INFO (SPGRE-16):
INFO (SPGRE-16): Clines selected: 2, Rats processed: 2, Phase constraints: 2, Phase violations: 2, Outside ideal range: 2
INFO (SPGRE-16):
INFO (SPGRE-16): Cline on net M_C_DQS_DQS from Pin D10M5.93(2341.74, 7690.00) - Via(2948.80, 6048.58) on BOTTOM
INFO (SPGRE-16): FAIL! Rat D10M5.93 - U95.ACT : Ideal Range[-2.50 MIL, 2.50 MIL] : Actual -118.18 MIL : Length 3081.61 MIL
INFO (SPGRE-16):
INFO (SPGRE-16): Cline on net M_C_DQS_DQS from Pin D10M5.94(2302.37, 7764.80) - Via(2914.10, 6048.58) on BOTTOM
INFO (SPGRE-16): FAIL! Rat D10M5.94 - U95.ACT : Ideal Range[-2.50 MIL, 2.50 MIL] : Actual 118.18 MIL : Length 3199.79 MIL
INFO (SPGRE-16):
INFO (SPGRE-16): CommandEnd APT *****
```

Auto-Interactive Delay Tuning (AiDT)

AiDT is an interactive command and part of the Allegro Timing Environment (ATE). This command operates on a user defined selection set of clines and/or cline segments to modify their route pattern to make them satisfy their individual/match group delay constraints.

It does this by -

- Using Constraint Manager or Smart Data to obtain how much each net/pin-pair is short of compliance with its delay constraints.
- Using a set of options to create the appropriate tuning pattern.

Note: When the AiDT command has finished executing, it may/not achieve 100% due to space limitations but generally it will modify the selection set and produce high quality results wherever possible.

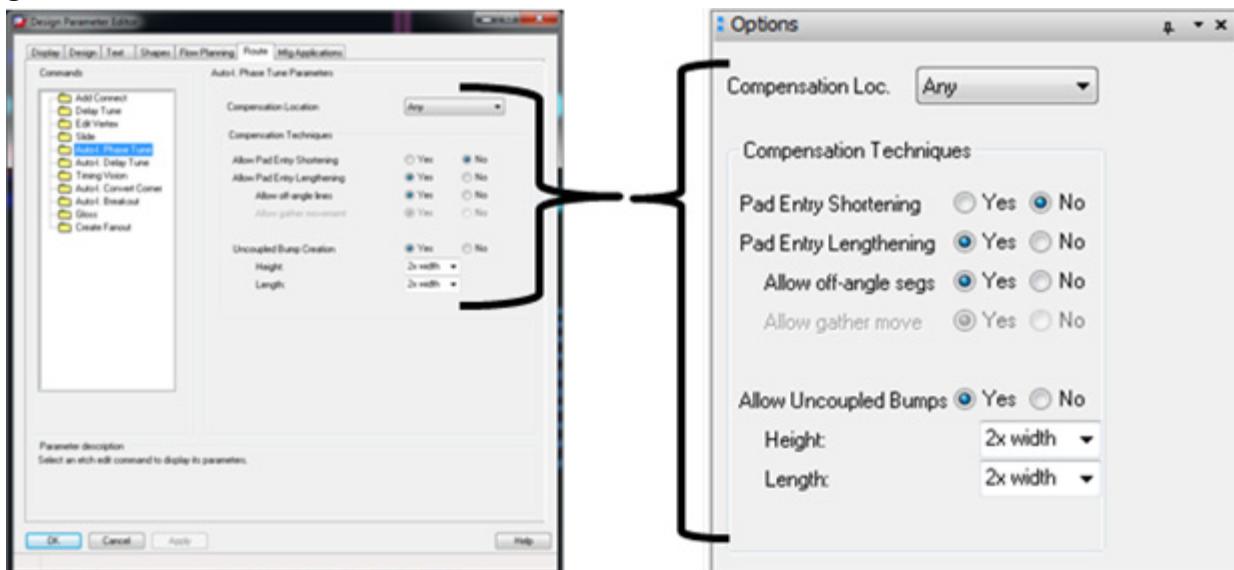
You can invoke the AiDT command from:

Route – Auto-Interactive Delay Tuning

AiDT Options

You can find the default techniques to solve delay in the *Design Parameters* dialog box. However, these options also can be found in the Options tab on the canvas when you invoke the AiDT command.

Figure 1-51



Auto-I Delay Tune Parameters

Override bundle level parameters

Design Planning Bundles have tuning parameter properties that can be applied to them. This allows you to set different parameters on different bundles. When AiDT executes it will read those parameters to create the desired tuning pattern. You can tell AiDT to ignore the Bundle parameters as its default method. You can also change these on the Options dialog when you run AiDT.

Allow in Constraint Areas

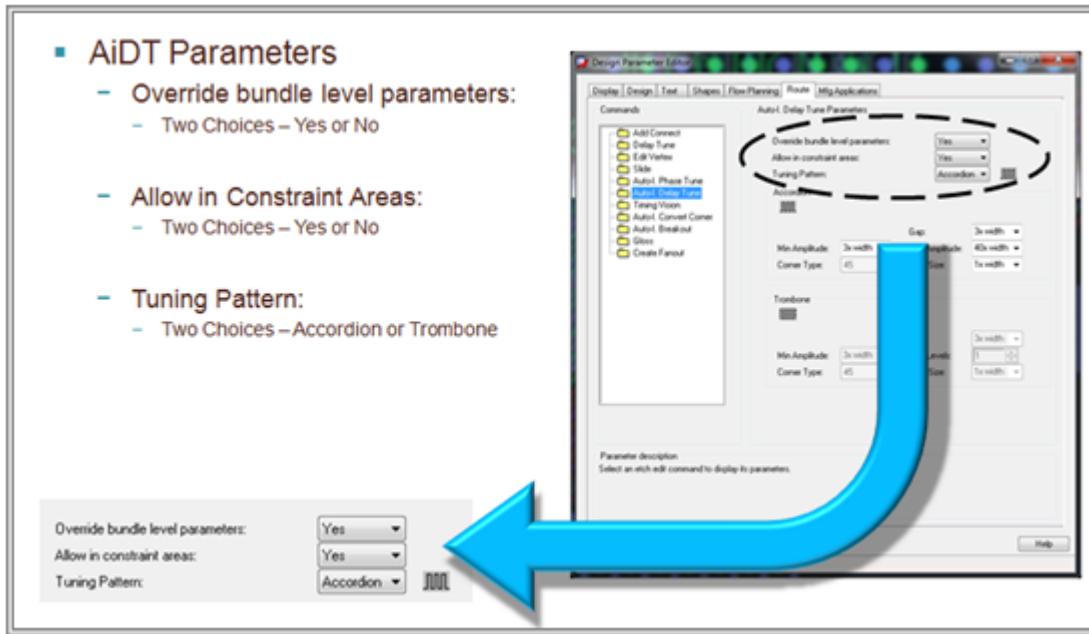
This parameter allows or disallows tuning pattern generation inside a constraint area. When you set it to *Yes*, AiDT is allowed to create tuning patterns inside the constraint region. When set to *No* AiDT will not allow tuning patterns to be created inside the constraint region.

Note: Generally, tuning in constraint regions increases routing using a reduced line size. This may cause impedance issues for the signals in the design. It is recommended to simulate after routing is complete to make sure that there are no Signal Integrity issues.

Tuning Pattern

This option lets you control the delay pattern that AiDT will generate. You can choose one of the options: Accordion or Trombone.

Figure 1-52 Bundle Options - Overrides, CNS Areas and Tuning Pattern



Accordion Parameters

You can set a variety of Accordion Parameters that include the following:

Min Amplitude

Controls the minimum bump size created by AiDT. Varying this can change the results generated by AiDT. Increasing this may also increase the amount of delay failures. This is due to the strict nature of the minimum bump creation. If a net/pin-pair requires one more partial bump, then AiDT will not create the additional bump. This is quite often the reason for delay failures. The default setting is: 3x the Trace Width. This can also be a keyed in numeric value.

Corner Type

Sets the desired corner type for the tuning pattern. The default setting is : 45 only.

Gap

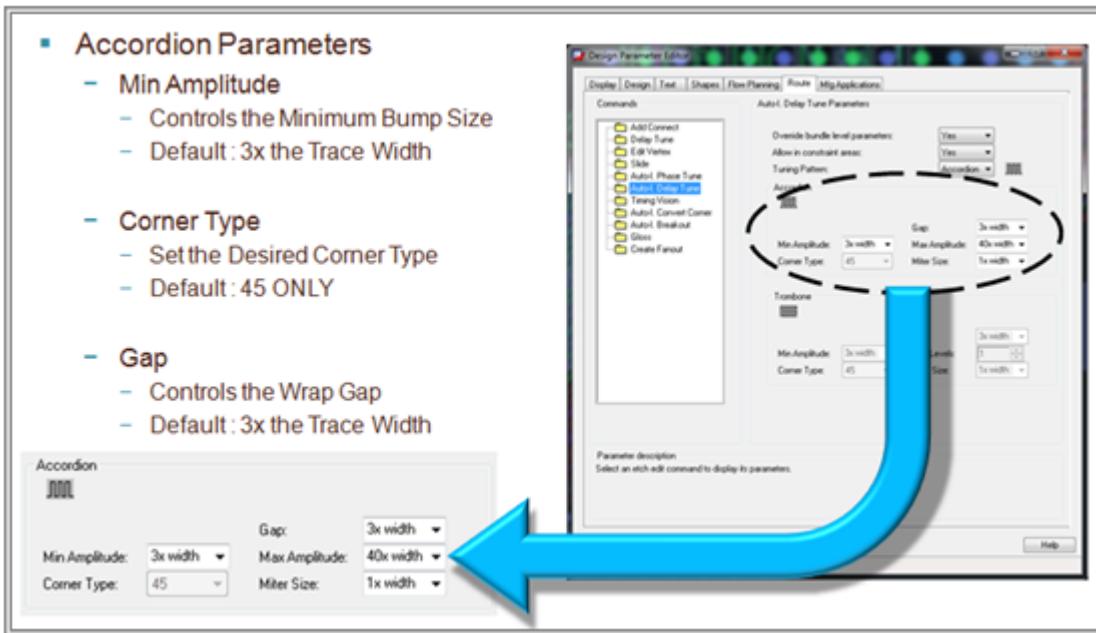
This setting controls the gap between the wraps in the pattern. You should pick a wrap gap that is greater than the trace to plane height. In general 3x the trace width gap is good. Also, the wrap-gap should be at least equal to the Same-Net Spacing Line-to-Line spacing value. AiDT will always use the larger clearance value either the Gap or the Same-Net Spacing Line-

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

to-Line spacing value. The default setting is: 3x the Trace Width. This can also be a keyed in numeric value.

Figure 1-53 Accordion Parameters



Max Amplitude

This value controls the maximum size of a tuning bump. You can use this value together with the *Min Amplitude* value to control how many bumps AiDT creates. In other words, increasing this size may create less bumps but they will be bigger. The default setting is: 40x the Trace Width. This can also be a keyed in numeric value.

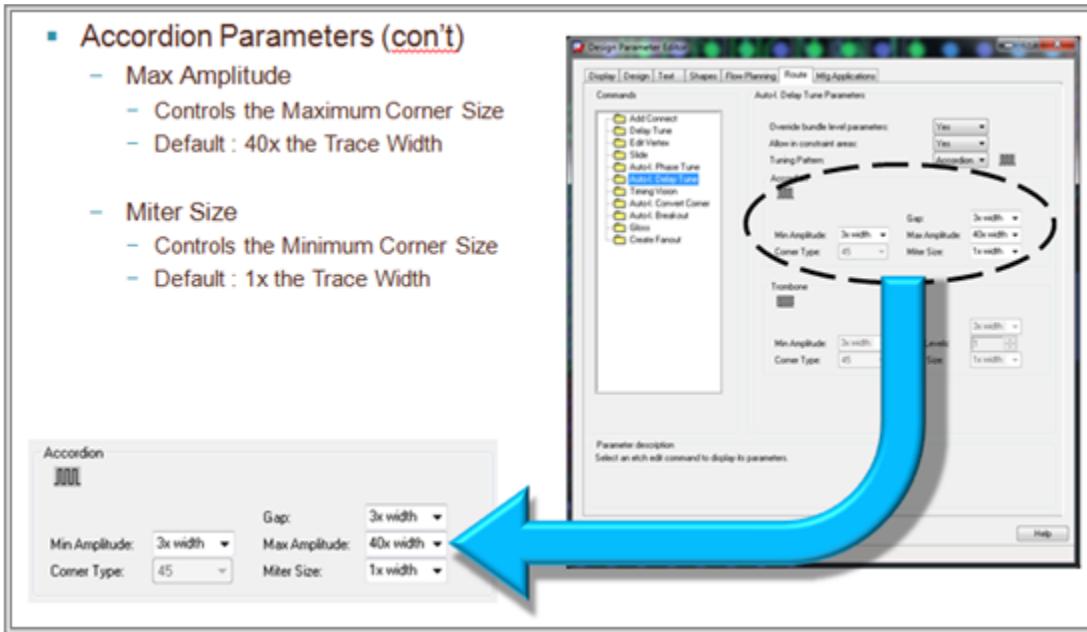
Miter Size

This value controls the minimum 45 degree corner size AiDT can create. The default setting is: 1x the Trace Width. This can also be a keyed in numeric value.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

Figure 1-54 Accordion Parameters (continued)



Trombone Parameters

AiDT can also create Trombone Patterns. You can set a variety of Trombone parameters that include the following:

Min Amplitude

Controls the minimum bump size created by AiDT. Varying this can change the results generated by AiDT. Increasing this may also increase the amount of delay failures. This is due to the strict nature of the minimum bump creation. If a net/pin-pair requires one more partial bump, then AiDT will not create the additional bump. This is quite often the reason for delay failures. The default setting is: 3x the Trace Width. This can also be a keyed in numeric value.

Corner Type

Sets the desired corner type for the tuning pattern. The default setting is: 45 only.

Gap

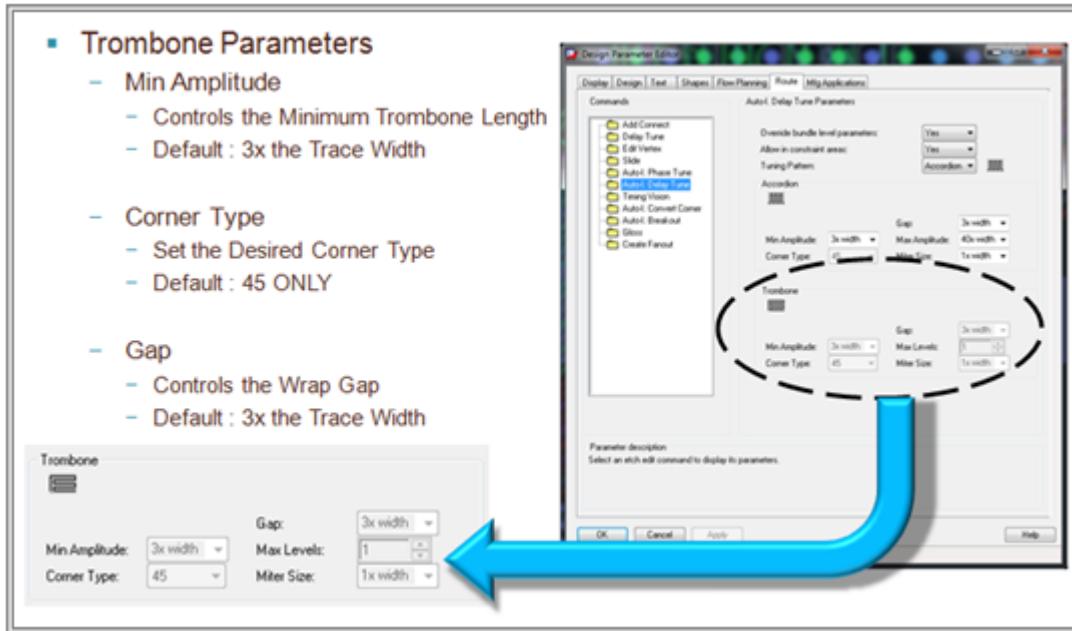
This setting controls the gap between the wraps in the pattern. You should pick a wrap gap that is greater than the trace to plane height. In general 3x the trace width is good. Also, the wrap-gap should be at least equal to the Same-Net Spacing Line-to-Line spacing value. AiDT

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

will always use the larger clearance value either the Gap or the Same-Net Spacing Line-to-Line spacing value. The default setting is: 3x the Trace Width. This can also be a keyed in numeric value.

Figure 1-55 Trombone Parameters



Max Levels

This value controls the maximum number of trombone wraps AiDT can create. The default setting is: 1.

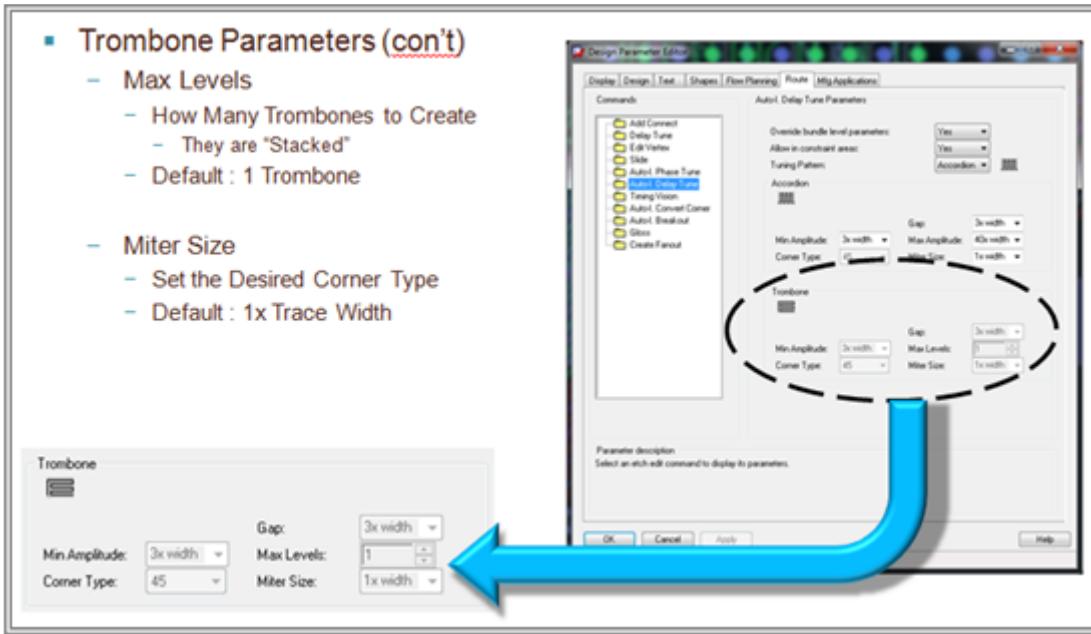
Miter Size

This value controls the minimum 45 degree corner size AiDT can create. The default setting is: 1x the Trace Width.

Allegro X User Guide: Allegro X Timing Environment

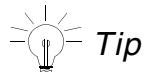
Allegro Timing Environment (ATE)

Figure 1-56 Trombone Parameters (continued)



AiDT Use Model

AiDT is focused on solving the delay problem. It does not disturb phase tuning structures created by AiDT when it adds delay structures.



It is recommended that you resolve any uncoupled length DRC error markers before using AiDT. When you have solved all of your Differential Phase issues, run DRC to resolve any uncoupled length DRC errors created by AiDT.

Timing Vision Use Model: AiDT Setup

The following figure illustrates the Timing Vision use model to setup the environment, create the Timing Group and Update the smart data for use with AiDT. This simple process should be repeated each time you start your design. Especially the Timing Group and Update goals sections as these are not stored in the database. The color scheme is generally done only once.

If you add or remove any membership to the Timing Group you need to Update the Goals. However, do not add all the nets in the design into the Timing Group as this can degrade system performance. Generally the impact is not bad, but there is no reason to include nets

Allegro X User Guide: Allegro X Timing Environment

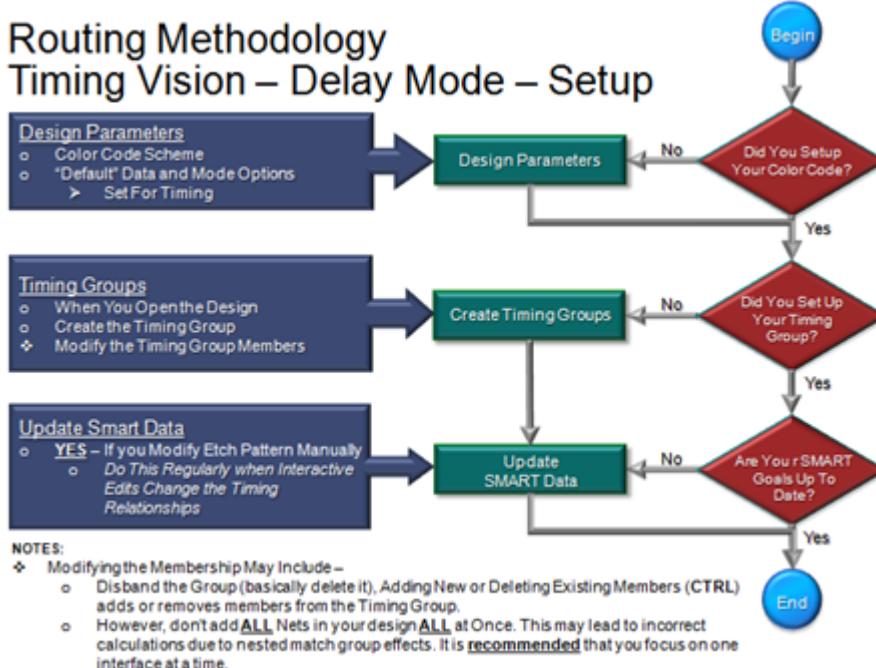
Allegro Timing Environment (ATE)

that do not have delay constraints or are incompletely specified by engineering. It is better to wait until the design is more fully developed before pulling all nets into the design.

If you make manual etch edits, you could modify nets that are considered Critical Signals by Timing Vision. If you do or you want to modify some net and make it the new critical signal (by making it the longest) then you need to update your smart data. Otherwise you will not get the desired long net to properly help color code the remaining nets in the interface.

Due to how all the nets of nested match groups can influence each other, it is not recommended that you put all of the nets in your design into the Timing Group when you initially create it. Therefore, if you have interfaces that have nets that control other nets lengths do the ones with the largest bit count first. These tend to be the ones that will use the most physical space on your design, then things like clocks, etc can be done last as they take up the least amount of route space and are in many cases able to float up/down as needed to satisfy the timing requirements for the whole interface.

Figure 1-57 Timing Vision Setup for AiDT



The next figure illustrates the three steps to success: SEE. These steps outline the setup of Timing Vision, Timing Groups and the setting and changing of options as needed. When you execute AiDT, the chances of success should be good. When it completes, examine the results for quality and failures. Make adjustments from there to options, selection sets, and so on. If the results are good, then move on to the next selection set and continue solving the delay problem.

Figure 1-58 AiDT - Three Steps to success SEE



The basic flow of the AiDT use model: using the SEE steps; Setup, Execute, Examine.

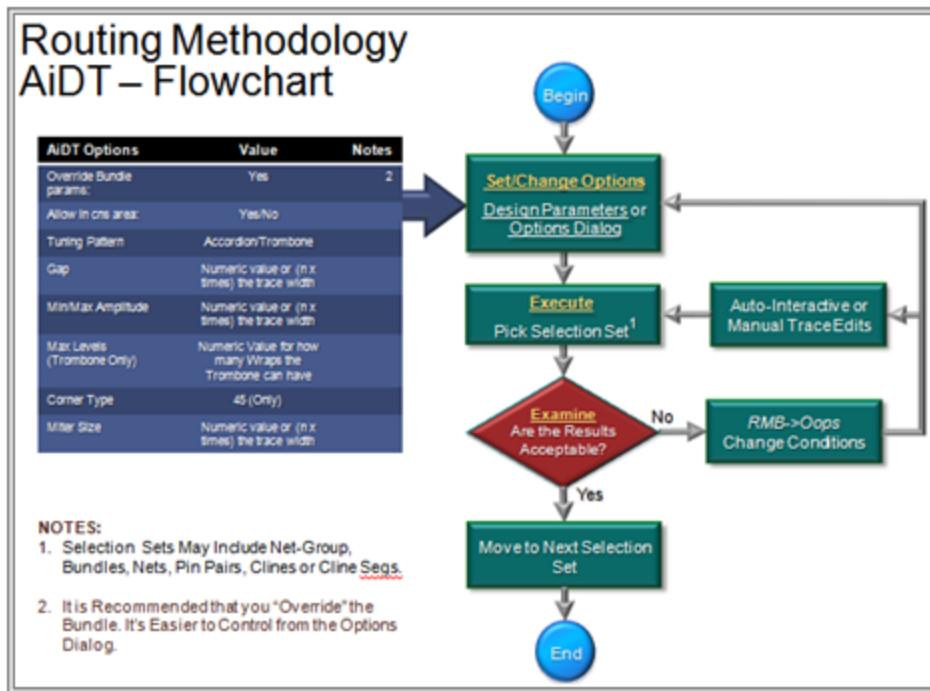
In this flow, you do the normal setup items you would for Timing Vision although Timing Vision is not required for AiDT to operate properly. However, Timing Vision presents valuable information to you before using AiDT as to which nets are not meeting delay and which are not. With this knowledge and the increased information provided by Timing Vision data tips you can make choice as to which pattern and parameters are to be used.

You can generally get good delay convergence on the first or second attempt. When AiDT finishes it will update the canvas with its results for this run. Take a quick look and see if the results meet your aesthetic and constraint goals. If the results look good, move on to the next selection set or use the *Oops* command to return to the pre-AiDT execution state. Make some changes either to the Options or maybe make some manual changes to the interconnect pattern and run it again.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

Figure 1-59 AiDT - General Use Model Flowchart



AiDT Selection Techniques

AiDT works on selection sets. You can select clines, cline segs or groups as your choices in the Find Filter. You can also use the *Temp Group* to make selections sets for more specific selections.

Note: Groups selects user defined groups of objects and AiDT will use its compensation techniques/algorithms on the clines of the group.

You also do not have to be careful about picking both halves of the differential pairs. Auto-Interactive Delay Tuning understands that it must tune both halves of the differential pair.

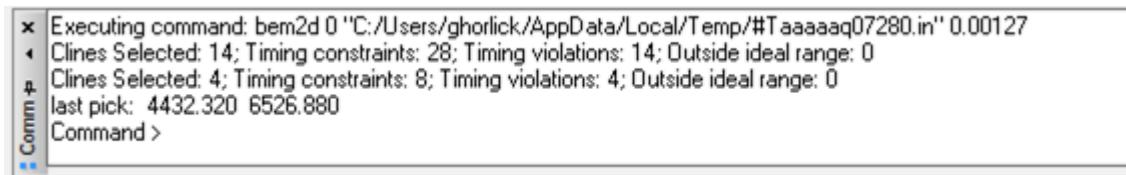
AiDT Feedback

When AiDT is run, there is information in the command window that gives some basic statistics on how well this attempt did. For example, in next figure, you can see 4 Clines selected and 8 Timing Constraints associated with them. When complete AiDT ended up with 4 Timing Violations and 0 were outside the ideal range.

Allegro X User Guide: Allegro X Timing Environment

Allegro Timing Environment (ATE)

Figure 1-60 Command Window Statistics

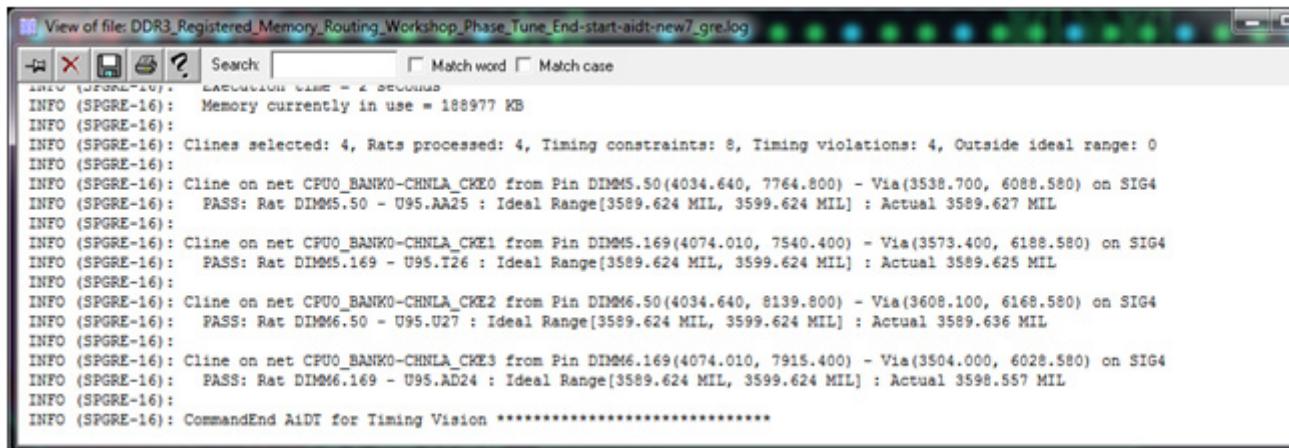


The screenshot shows a command window with the following text:

```
x Executing command: bem2d 0 "C:/Users/ghorlick/AppData/Local/Temp/#Taaaaaq07280.in" 0.00127
↳ Clines Selected: 14; Timing constraints: 28; Timing violations: 14; Outside ideal range: 0
↳ Clines Selected: 4; Timing constraints: 8; Timing violations: 4; Outside ideal range: 0
↳ last pick: 4432.320 6526.880
Command >
```

For more extensive information, use the *File Viewer* option to see more statistics that may help you resolve an AiDT delay failures.

Figure 1-61 Example of Log File Detailed Output



The screenshot shows a log file viewer window titled "View of file: DDR3_Registered_Memory_Routing_Workshop_Phase_Tune_End-start-aidt-new7.gre.log". The log file contains the following detailed output:

```
EVALUATION TIME = 2 SECONDS
INFO (SPGRE-16): Memory currently in use = 188977 KB
INFO (SPGRE-16):
INFO (SPGRE-16): Clines selected: 4, Rats processed: 4, Timing constraints: 8, Timing violations: 4, Outside ideal range: 0
INFO (SPGRE-16):
INFO (SPGRE-16): Cline on net CPU0_BANK0-CHNL0_CKE0 from Pin DIMM5.50(4034.640, 7764.800) - Via(3538.700, 6088.580) on SIG4
INFO (SPGRE-16): PASS: Rat DIMM5.50 - U95.AA25 : Ideal Range[3589.624 MIL, 3599.624 MIL] : Actual 3589.627 MIL
INFO (SPGRE-16):
INFO (SPGRE-16): Cline on net CPU0_BANK0-CHNL0_CKE1 from Pin DIMM5.169(4074.010, 7540.400) - Via(3573.400, 6188.580) on SIG4
INFO (SPGRE-16): PASS: Rat DIMM5.169 - U95.T26 : Ideal Range[3589.624 MIL, 3599.624 MIL] : Actual 3589.625 MIL
INFO (SPGRE-16):
INFO (SPGRE-16): Cline on net CPU0_BANK0-CHNL0_CKE2 from Pin DIMM6.50(4034.640, 8139.800) - Via(3608.100, 6168.580) on SIG4
INFO (SPGRE-16): PASS: Rat DIMM6.50 - U95.U27 : Ideal Range[3589.624 MIL, 3599.624 MIL] : Actual 3589.636 MIL
INFO (SPGRE-16):
INFO (SPGRE-16): Cline on net CPU0_BANK0-CHNL0_CKE3 from Pin DIMM6.169(4074.010, 7915.400) - Via(3504.000, 6028.580) on SIG4
INFO (SPGRE-16): PASS: Rat DIMM6.169 - U95.AD24 : Ideal Range[3589.624 MIL, 3599.624 MIL] : Actual 3598.557 MIL
INFO (SPGRE-16):
INFO (SPGRE-16): CommandEnd AiDT for Timing Vision *****
```