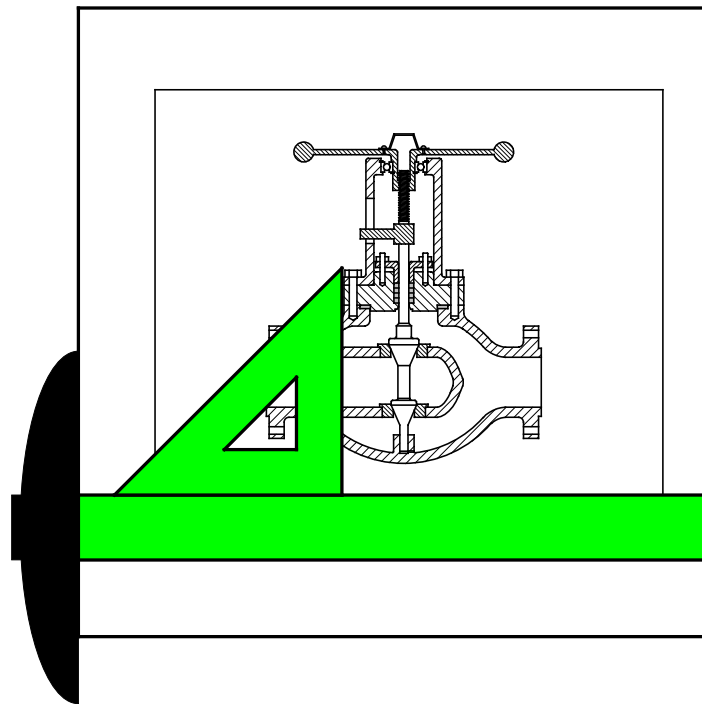


User's Guide

DeltaCad®

Professional
for Windows™

By  *Midnight Software, Inc.*



Copyright © 1995-2004 by  *Midnight Software, Inc.* All rights reserved.

The contents of this manual are subject to copyright © laws and international treaties. No part of this manual may be reproduced in any form without written permission from Midnight Software, Inc.

DeltaCad® is a registered trademark of Midnight Software, Inc.

Basic Scripting Engine provided by Cypress Software, Inc., Copyright 1993-1999. All rights reserved.

Microsoft® and Windows™ are trademarks of Microsoft Corporation.
All other trademarks are the property of their respective companies.

 *Midnight Software, Inc.*

For contact info. (E-Mail, FAX, address, etc.) go to www.deltacad.com/contact.html

MIDNIGHT SOFTWARE, INC. LICENSE AGREEMENT

MIDNIGHT SOFTWARE, INC. ("LICENSOR") IS WILLING TO LICENSE THIS SOFTWARE TO YOU ONLY IF YOU ACCEPT ALL OF THE TERMS IN THIS LICENSE AGREEMENT. PLEASE READ THE TERMS CAREFULLY BEFORE YOU INSTALL THE SOFTWARE, BECAUSE BY INSTALLING THE SOFTWARE YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THESE TERMS, LICENSOR WILL NOT LICENSE THIS SOFTWARE TO YOU, AND IN THAT CASE YOU SHOULD RETURN THIS PRODUCT PROMPTLY, INCLUDING THE PACKAGING AND ALL WRITTEN MATERIALS, TO THE PLACE OF PURCHASE FOR A FULL REFUND.

Ownership of the Software

1. The Licensor software program ("DeltaCad") and the accompanying written materials are owned by Licensor (or its suppliers) and are protected by United States copyright laws, by laws of other nations, and by international treaties.

Grant Of License

2. Licensor grants to you the right to use one copy of the Software on a single computer. You may load one copy into permanent memory of one computer and may use that copy only on that same computer. You may install the Software on a network server, provided that you have a License for each station of the network which has access to the software.

Restrictions on Use and Transfer

3. If the Software is supplied on 3.5" disks, 5.25" disks, a CDROM, or a downloadable file you may use only the size disks or file appropriate for your computer. You may not use the other size disks or file on another computer or loan, rent, transfer, or assign them to another user except as part of the permanent transfer of the software and all written materials (as provided for below).

4. You may not copy the Software, except that (1) you may make one copy of the Software solely for backup or archival purposes, and (2) you may transfer the Software to a single hard disk provided you keep the original solely for backup or archival purposes. You may not copy the written materials.

5. You may permanently transfer the Software and accompanying written materials (including the most recent update and all prior versions) if you retain no copies and the transferee agrees to be bound by the terms of this Agreement. Such a transfer terminates your license. You may not rent or lease the Software or otherwise transfer or assign the right to use the Software, except as stated in this paragraph.

6. You may not reverse engineer, decompile, or disassemble the Software.

Limited Warranty

7. Licensor warrants that the Software will perform substantially in accordance with the accompanying written materials for a period of 90 days from the date of your receipt of the Software. Any implied warranties on the Software are limited to 90 days. Some states do not allow limitations on duration of an implied warranty, so the above limitation may not apply to you.

8. LICENSOR DISCLAIMS ALL OTHER WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT, WITH RESPECT TO THE SOFTWARE AND THE ACCOMPANYING WRITTEN MATERIALS. This limited warranty gives you specific legal rights. You may have others, which vary from state to state.

9. LICENSOR'S ENTIRE LIABILITY AND YOUR EXCLUSIVE REMEDY SHALL BE, AT LICENSOR'S CHOICE, EITHER (A) RETURN OF THE PRICE PAID OR (B) REPLACEMENT OF THE SOFTWARE THAT DOES NOT MEET LICENSOR'S LIMITED WARRANTY AND WHICH IS RETURNED TO LICENSOR WITH A COPY OF YOUR RECEIPT. Any replacement Software will be warranted for the remainder of the original warranty period or 30 days, whichever is longer. These remedies are not available outside the United States of America.

10. This Limited Warranty is void if failure of the Software has resulted from modification, accident, abuse, or misapplication.

11. IN NO EVENT WILL LICENSOR BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY LOSS OF PROFITS, LOST SAVINGS, OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF YOUR USE OR INABILITY TO USE THE SOFTWARE. Because some states do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

12. This agreement is governed by the laws of the State of Washington.

13. If you have any questions concerning this Agreement or wish to contact Licensor for any reason, go to www.deltacad.com/contact.html to obtain contact information (Phone, FAX, e-mail, address, etc.).

14. U.S. Government Restricted Rights. The Software and documentation are provided with Restricted Rights. Use, duplication, or disclosure by the Government is subject to restrictions set forth in subparagraph (c)(1) of The Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1)(ii) and (2) of Commercial Computer Software - Restricted Rights at 48 CFR 52.227-19, as applicable. Supplier is Midnight Software, Inc., of Seattle, Washington.

Contents

Overview	1
Introduction	1
Technical support	2
System Requirements	3
Installation	5
Installing DeltaCad	5
Registering DeltaCad	5
Ordering or Upgrading DeltaCad.....	6
Starting and Exiting DeltaCad	6
Removing DeltaCad	8
Sample Drawings & Libraries	9
Tutorial	11
Learning DeltaCad.....	11
Start DeltaCad	13
Drawing the Calculator	13
Printing the Calculator	28
Saving and Exiting DeltaCad.....	30
End of Tutorial	31
Getting Started	33
Screen Layout	33
Drawing Area	39
Angles.....	40
Layer.....	41
View Scale	42
Drawing Scale	43
Initial Size and Position	46
Drawing Data	47
How to Use Help.....	48
Keyboard	49
Mouse.....	55
AutoSave	57

Drawing	59
Point.....	59
Point Objects.....	59
Point Toolbar functions	60
Draw a point	61
Draw a point offset equally from the last two points	62
Draw all points in an object.....	63
Draw mid-points	63
Line	64
Line Toolbar functions.....	64
Draw a line with two points	65
Draw a continued line	65
Draw a horizontal line	66
Draw a vertical line.....	67
Draw an angled line	68
Draw a perpendicular line	69
Draw a parallel line.....	70
Circle.....	71
Circle Toolbar functions	71
Draw a circle with center and radius.....	72
Draw a circle within a box	72
Draw a circle with three points.....	73
Draw a circle tangent to three lines.....	74
Draw a circular arc using the center	75
Draw a circular arc using three points.....	76
Draw an ellipse.....	77
Draw an ellipse within a parallelogram	78
Draw an elliptical arc.....	79
Draw a parallel circle.....	80
Draw a circle with two points.....	80
Draw a 180 degree arc with two points.....	81
Shapes.....	82
Shapes Toolbar functions	82
Draw a rectangle using two corners.....	83
Draw a rectangle using the center and corner.....	83
Draw a parallelogram.....	84
Draw a triangle	85
Draw a triangle, square, pentagon, etc.....	85
Draw a polygon	86
Text.....	87
Text Toolbar functions	87
Draw single line text.....	89
Draw multiple line text.....	89

Dimension.....	90
Dimension Toolbar functions	90
Draw a horizontal dimension.....	91
Draw a vertical dimension.....	92
Draw an angled dimension	93
Draw an angled dimension with angled extension lines	94
Draw a text pointer arrow.....	95
Draw a diameter dimension	96
Draw a radius dimension	96
Draw an angle dimension	97
Draw a point-to-point dimension	98
Draw an X,Y coordinate dimension	98
Spline.....	99
Spline Toolbar functions	99
Draw a spline	100
Draw a continuous spline.....	100
Add a point to a spline	101
Remove a point from a spline	101
Move a spline point	102
Draw a parallel spline.....	102
Editing Individual Objects	103
Undo	103
Edit object data.....	103
Edit grouped objects.....	104
Move an object point	105
Erase an object.....	105
Slide an endpoint.....	106
Create a corner.....	107
Create a chamfer.....	108
Create a radius	109
Split a line, circle, arc, ellipse, or elliptical arc	110
Move to the top of the Z-Order	110
Move to the bottom of the Z-Order	110

Editing Selected Objects	111
Selecting objects	111
Set the Base Point for selected objects.....	112
Move selected objects	112
Rotate selected objects	113
Mirror selected objects	114
Scale selected objects.....	115
Group selected objects.....	116
Erase selected objects	116
Change selected text.....	117
Change layer of selected objects	118
Change color of selected objects	118
Change line type/cross-hatch pattern of selected objects.....	119
Change line weight of selected objects	119
Slide endpoints of selected lines	120
Viewing the Drawing	121
View objects within a rectangle	121
Center view around a point.....	122
View all	122
Previous views.....	123
Change view scale	123
Zoom-In	123
Zoom-Out.....	124
View a single layer.....	124
View Toolbar Tab	125
Printing	127
Print	127
Setting the Print Region(s)	128
Printer Setup.....	130
Print Preview	131
Offline Printing	132
Misc.	133
File Operations	133
Clipboard	138
Options	140
Extract Group Data	148
Window	151
Calculator.....	154

Macro Language	155
Macro Language.....	155
Run macro	157
Edit macro	158
Macro Toolbar Tab	159
Macro functions listed by type	160
File functions.....	160
Layer functions.....	160
Group functions.....	160
Misc. functions	161
Object creation functions	162
Object parameter functions.....	162
Object retrieval functions	163
Print region functions	163
Select functions.....	164
Symbol functions.....	164
View functions	164
Alphabetical list of functions	165
dcAddLayer	165
dcAddPrintRegion	165
dcAddSymbol	165
dcChangedSinceLastSave.....	165
dcChgColorSelObjs	166
dcChgLayerSelObjs	166
dcChgLineTypeSelObjs	167
dcChgLineWeightSelObjs	167
dcClose	167
dcCloseWithoutSaving	168
dcCopyBMPFiletoClipboard.....	168
dcCopyCurrentPicturetoFile.....	168
dcCopySelObjstoClipboard.....	169
dcCopySymboltoClipboard	169
dcCreateBox	169
dcCreateCircle	170
dcCreateCircleEx	170
dcCreateDim	170
dcCreateLine.....	171
dcCreatePoint	171
dcCreateShape	172
dcCreateSpline	172
dcCreateText.....	173
dcCreateTriangle	173

dcDeleteLayer	173
dcDeletePrintRegion	173
dcDeleteSymbol	174
dcDoesLayerExist	174
dcEndPrintPreview	174
dcEraseCurrentObject	174
dcEraseSelObjs	175
dcEraseTempPoints	175
dcExportScreenToBMPFile	175
dcGetActiveWindow	176
dcGetAreaofCurrentObject	176
dcGetBookmark	176
dcGetCircleData	177
dcGetCircleParms	178
dcGetCurrentLayer	178
dcGetCurrentPrintRegion	178
dcGetDeltaCadDwgName	179
dcGetDeltaCadProgramDirectory	179
dcGetDeltaCadVersion	180
dcGetDeltaCadWindowHandle	180
dcGetDimData	180
dcGetDimParms	181
dcGetDrawingData	182
dcGetDrawingScale	182
dcGetDrawingWindowHandle	183
dcGetFirstGroupName	183
dcGetFirstObject	184
dcGetFirstSelectedObject	184
dcGetGroupData	185
dcGetInitialSizeandPos	186
dcGetLayerName	186
dcGetLengthofCurrentObject	186
dcGetLineData	187
dcGetLineParms	188
dcGetNextGroupName	188
dcGetNextObject	188
dcGetNextSelectedObject	189
dcGetNumberOfDeltaCadDwgs	189
dcGetNumberOfLayers	189
dcGetNumberOfPrintRegions	190
dcGetObjectAt	190
dcGetObjectBox	191
dcGetObjectLayer	191

dcGetPictureData.....	192
dcGetPointData.....	193
dcGetPointParms.....	193
dcGetPrintRegionData.....	194
dcGetPrintRegionName.....	194
dcGetSelCopyMode.....	195
dcGetShapeData.....	195
dcGetShapesParms.....	196
dcGetSplineData.....	196
dcGetSplineParms.....	197
dcGetTextData.....	197
dcGetTextParms.....	198
dcGetViewScaleandRect.....	199
dcGroupSelObjs.....	200
dclsCurrentObjectSelected.....	200
dclsLayerOn.....	200
dclsPrintRegionLayerOn.....	201
dclsSelected.....	201
dcMirrorSelObjs.....	201
dcMoveSelObjs.....	202
dcNew.....	202
dcNewSymbolFile.....	202
dcOpen.....	203
dcOpenGroupWindow.....	203
dcPasteClipboardData.....	203
dcPasteSpecialClipboardData.....	204
dcPrint.....	204
dcPrintPreview.....	204
dcReadInput.....	204
dcRenameLayer.....	205
dcRenamePrintRegion.....	205
dcRestoreAllParms.....	205
dcRestoreBookmark.....	206
dcRotateSelObjs.....	206
dcSave.....	206
dcSaveAllParms.....	206
dcSaveAs.....	207
dcScaleSelObjs.....	207
dcSelectAll.....	207
dcSelectCurrentObject.....	208
dcSelectObjInBox.....	208
dcSetActiveWindow.....	208
dcSetBookmark.....	209

dcSetCircleParms	209
dcSetCurrentLayer	210
dcSetCurrentPrintRegion	210
dcSetDimParms	210
dcSetDrawingData	211
dcSetDrawingScale	212
dcSetDrawingWindowMode	212
dcSetInitialSizeandPos	213
dcSetLineParms	213
dcSetPointParms	213
dcSetSelCopyMode	214
dcSetSelectBase	214
dcSetShapesParms	214
dcSetSplineParms	215
dcSetTextParms	216
dcSetViewScale	216
dcTurnLayerOff	216
dcTurnLayerOn	217
dcTurnPrintRegionLayerOff	217
dcTurnPrintRegionLayerOn	217
dcUnGroupAllGroups	218
dcUnGroupCurrentObject	218
dcUnSelectAll	218
dcUnSelectCurrentObject	218
dcUpdateDisplay	219
dcViewAll	219
dcViewBox	220
dcViewCenteredOnXY	220
dcViewPrevious	220
Troubleshooting	221
DeltaCad troubleshooting web site	221
Computer lock up	221
Display, Printing or FAX problems	221
Printing not to scale	221
Drawing prints blank sheet of paper	222
Drawing zooms out too far when opening	222
Pen Plotter Problems	222
Dimension text and arrowheads too small	223
DeltaCad will not print in color	223
Disk full problems	223
Index	225

Overview

Introduction

DeltaCad® is a powerful, easy to learn CAD (Computer Aided Design) program designed to meet all of your CAD needs. It can produce accurately scaled architectural or mechanical drawings, or just pretty pictures to paste into your favorite word processor (All of the Figures and Tables in this manual, except screen shots, were produced with DeltaCad). DeltaCad is more than just a paint program, because you can edit, scale, move, rotate, copy, etc. individual objects, not just paint pixels. DeltaCad allows you to zoom in to draw fine details or zoom out to see the whole drawing.

DeltaCad can be extended, by writing your own custom macros in the BASIC programming language. You can use macros to automatically create objects, or to read in data, which you have drawn.

DeltaCad can be used for Drafting, House plans, Decks, Landscape design, Newsletters, Brochures, Flyers, Graphs & Charts, Presentations, Instructions, Office or Furniture Layouts, Technical Illustrations, Business Cards, Forms, Signs, Labels, Organization Charts, Flow Charts, Home or School projects, Maps, Schematics, Pretty Pictures, etc.

DeltaCad includes the following features:

1. Simplified user interface makes it easy to learn and use.
2. On-Screen instructions help guide the user.
3. Draws accurately scaled pictures with absolute precision.
4. Draws points, lines, circles, ellipses, arcs, splines, text, dimensions, squares, rectangles, triangles, or any misc. shape (Filled or Empty).
5. Pastes pictures into your favorite word processor.
6. Create your own symbols.
7. Calculate length and area.
8. Contains a full feature BASIC macro programming language.
9. Creates solid, hidden, center, phantom, cutting, break, stitch, arrow, or double arrow lines, or many different cross-hatch patterns.
10. Easily Zoom in and out and move to any part of the drawing.
11. Automatic object snapping makes it easy to create drawings exactly to scale.

12. Edit, move, copy, mirror, rotate, scale, change color, change line type, change cross-hatch pattern, change line weight of any object.
13. Undo the last operation with a single key stroke.
14. Edits any two lines to create a corner, radius, or chamfer.
15. Draws on up to 256 layers.
16. Comes complete with many sample drawings and symbol libraries.
17. Supports Metric or English units in decimal or fractions.
18. Reads and writes .DXF files for compatibility with other CAD programs.
19. True 32 bit program for improved performance.
20. Print Preview shows what the print will look like.
21. Rubber banding helps show how objects will look as they are being drawn.
22. Objects may be grouped into a single object to save memory, and simplify the drawing. Groups may be easily edited, ungrouped, or copied to other parts of the drawing.
23. Create Bill-of-Materials from grouped objects.
24. Works well on older computers.

Technical support

If you have any questions, comments, suggestions, or problems you should go to the web page shown below. You will find contact info. (e-mail, FAX, address, etc.), and a form to send your questions. If reporting a problem, please describe the problem as clearly as possible, and include the exact text of any error message. Problems, which can not be re-created, are difficult to correct. Please feel free to send in suggestions for improvements.

Midnight Software, Inc.

www.deltacad.com/contact.html

System Requirements

Personal computer with a 80386 (or later) processor.

Microsoft® Windows™ 95, 98, NT(4.0), 2000, ME, XP (or later). An older version of DeltaCad is included on the CDROM to provide support for Windows 3.1, some features described in this manual do not exist in the older version.

- VGA or better color graphics card and monitor compatible with the operating system. 24-bit color is recommended if you are displaying bitmaps in your drawings.
- 4 MB of extended memory.
- Approx. 5 MB of free disk space.
- Math coprocessor recommended, not required.
- Mouse and printer compatible with the operating system.

Installation

Installing DeltaCad

Perform the following steps to install DeltaCad:

1. If using the NTFS file system (on Windows NT, 2000, or XP) log into an account with “Administrator” privileges.
2. If Windows™ is set up to automatically run CDROM programs simply insert the DeltaCad CDROM into the drive and follow the instructions. NOTE: It may take a few seconds for Windows™ to start the setup program.
3. If Windows™ is not set up to automatically run CDROM programs insert the DeltaCad CDROM (or floppy disk) into drive D: (or, A: or B:).
4. Pick Run from the Start menu. Type D:\SETUP (or, A:\SETUP or B:\SETUP) and click the OK button with the mouse.
5. Follow the on-screen instructions to install DeltaCad.
6. If you are a new DeltaCad user, it is recommended that you read through the Tutorial section of the manual or help system.
7. If you have previously installed the DEMO version of DeltaCad you can run the setup program to install the full version. The setup program will re-write all of the DeltaCad files. It will not affect any other files, which you have created.

If you encounter any problems while running SETUP.EXE, see the Troubleshooting section of the manual or the help system.

Registering DeltaCad

Please take the time to register your copy of DeltaCad so you can be informed about product upgrades. You can register your copy on the internet at <http://www.deltacad.com/register> . If you purchased your copy of DeltaCad directly from Midnight Software, Inc., you are already registered, and do not need to do so.

Ordering or Upgrading DeltaCad

To order or upgrade DeltaCad please go to our web site (<http://www.deltacad.com>).

Starting and Exiting DeltaCad

Starting DeltaCad



To start DeltaCad pick **Programs/DeltaCad** from the **Start** menu.

Setting the default disk directory


The default disk directory is the disk directory that is used when DeltaCad is started from the Start menu. It is the directory that is displayed when **Open** is picked from the DeltaCad **File** menu. To change the default directory pick **Options/Default Directory** from the **Options** menu, enter the new default directory, and pick **OK**.

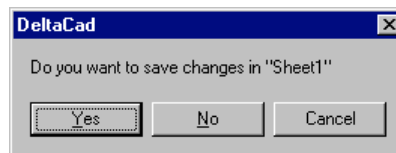
Calibrating the Screen

The first time DeltaCad is run you should calibrate the screen so images will appear the correct size and shape. If you are using the NTFS file system (on Windows NT or 2000) you will need to be logged in to an account with "Administrator" privileges. To calibrate the screen perform the following steps:

- Start DeltaCad.
- Maximize the DeltaCad window by clicking the mouse on the  button in the upper right corner of the DeltaCad Window. If the  button is currently displayed in the upper right corner of the window, it is already maximized.
- Select **Options/Screen Size** from the **Options** menu. Use a ruler and measure the actual height and width of the image on your computer screen, then enter the height and width and click the OK button.
NOTE: This is the entire area of your monitor that can emit light, not just the window size.
- Exit DeltaCad. The new screen size will not be used until DeltaCad is restarted.

Exiting DeltaCad

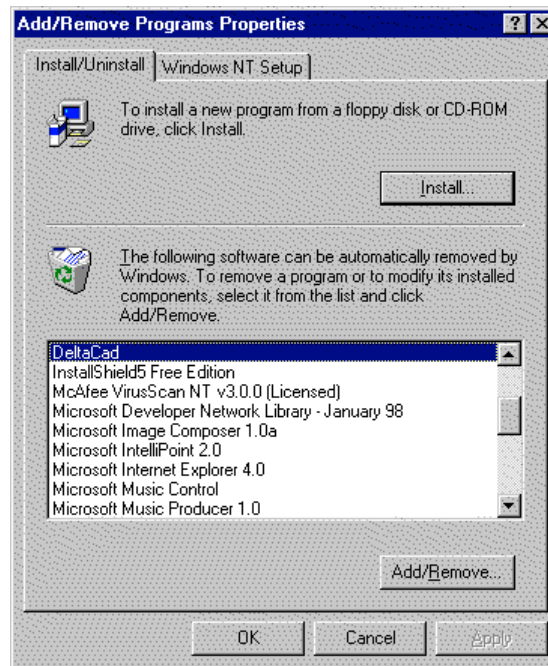
To exit DeltaCad simply pick **Exit** from the DeltaCad **File** menu, or click the  symbol on the title bar. If there are any open drawings which have changed their data since the last save, you will be prompted whether you want to save the changes to the drawing, as shown below.



Removing DeltaCad

Perform the following steps to remove DeltaCad from your computer:

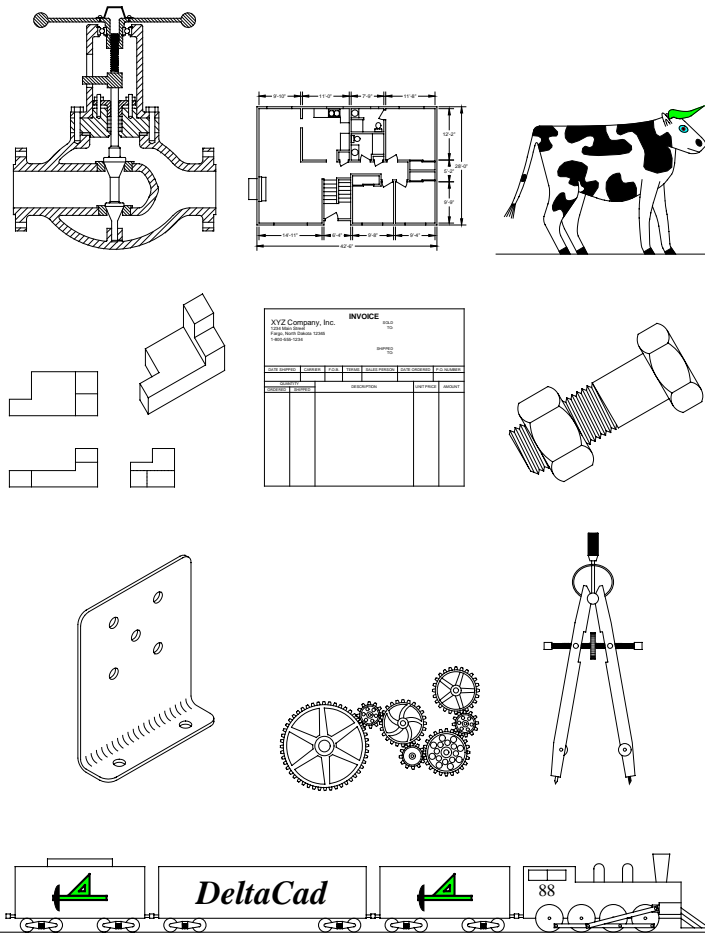
1. If you are using the NTFS file system (on Windows NT or 2000), log into an account with “Administrator” privileges.
2. Pick **Settings/Control Panel** from the **Start** menu, double click the **Add/Remove Programs** icon, pick the tab marked **Install/Uninstall** or **Change or Remove**, pick **DeltaCad** from the list of programs which can be removed, click **Add/Remove** or **Change/Remove**, and follow the on-screen instructions.



Sample Drawings & Libraries

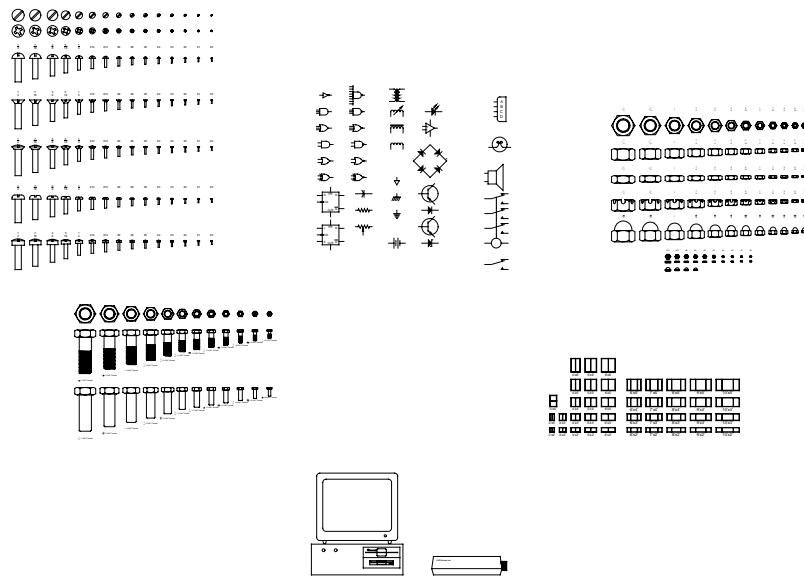
Sample Drawings

The sample drawings are located in the program directory (C:\Program Files\DeltaCad). The sample drawings are provided to give you an idea of DeltaCad's wide range of capabilities. The following are some of the samples supplied with DeltaCad.



Symbol Libraries

In addition to the sample drawings, DeltaCad contains many symbol libraries. The symbol libraries consist of hundreds of professionally drawn symbols and shapes that can be copied into your drawings. To copy a symbol into your drawing select **Symbols** from the **File** menu, pick the symbol you like, click OK, and then click the mouse where you would like to paste the symbol into your drawing. The following are some of the symbol libraries supplied with DeltaCad.



Tutorial

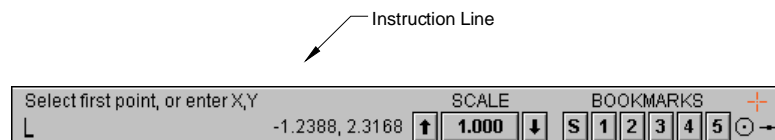
Learning DeltaCad

The following is a brief tutorial designed to show how to use some of the features of DeltaCad. This tutorial will demonstrate the basic features of DeltaCad by drawing the calculator shown below. There are many different ways to draw an object in DeltaCad. The following tutorial is designed to illustrate several different ways to create a DeltaCad drawing. The tutorial does not draw the calculator in the simplest way possible.

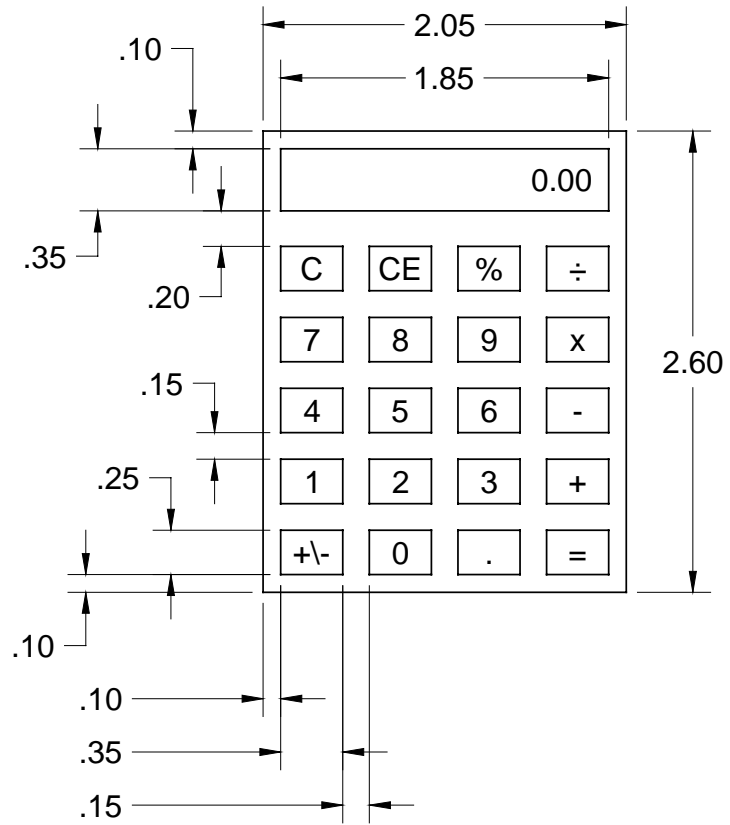
The calculator is a very simply drawing, but it is a good example of how DeltaCad can be used to draw almost any object.

As you read through the tutorial, you should try out each step on your own computer. After completing the tutorial you should read the rest of the manual and try out each new feature. DeltaCad is designed to be easy to learn by experimenting.

REMEMBER: Always read the instruction line on the status bar.



The figure below shows the dimensions of the various parts of the calculator.
This figure may be useful as you proceed through the tutorial.



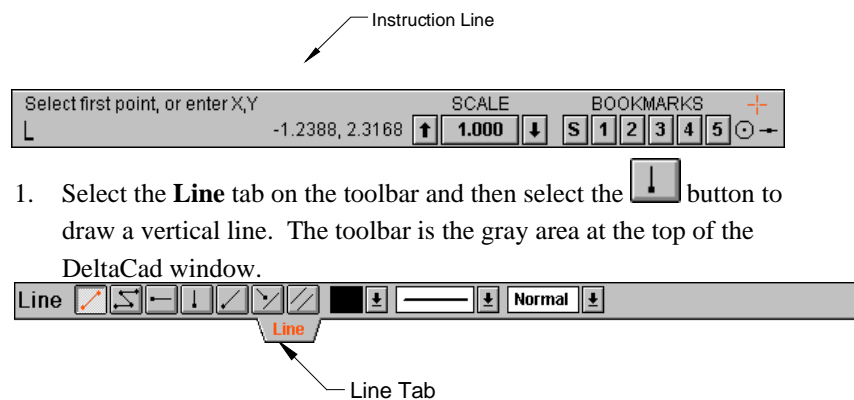
Start DeltaCad


Start DeltaCad by clicking the **Programs/DeltaCad** item on the **Start** menu. Then resize the DeltaCad window so it fills most of the screen, or maximize the DeltaCad window.



Drawing the Calculator


Draw the outline rectangle of the calculator

The first step will be to draw the vertical line on the left side of the calculator. As you proceed through the following steps, always read the instruction line on the Status Bar. The status bar is the gray area at the bottom of the DeltaCad window. The instruction line will always indicate what type of input DeltaCad is expecting next.



1. Select the **Line** tab on the toolbar and then select the  button to draw a vertical line. The toolbar is the gray area at the top of the DeltaCad window.



To select the **Line** tab move the mouse to the word "**Line**", on the toolbar, and then click the left mouse button. To select the  button, move the mouse over the  button and click the left mouse button.

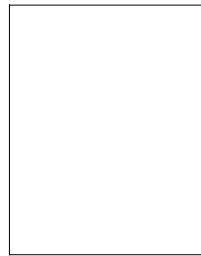
When the  toolbar button is selected it will change to its highlight color.


2. Move the mouse so the pointer is near the lower left corner of the DeltaCad window, and click the left mouse button to start drawing a line.

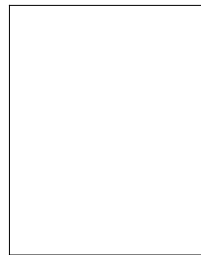
3. Now as you move the mouse, a temporary vertical line is shown on the screen. Move the mouse so the temporary line goes upward from the starting point, and then type **2.6** at the keyboard and press the ENTER key. This will create a vertical line 2.6 inches long. Now the first line should be shown on the screen.



4. Select the **Line** tab on the toolbar and then select the  button to draw a horizontal line.
5. Now move the mouse to the top of the vertical line that was just created. Notice that as you move the mouse near the end of the line a small set of cross-hairs  appear. While the cross-hairs are on the screen click the mouse button. The cross-hairs are used to indicate that the **exact** endpoint of the existing line will be used. This provides an easy way to draw a line from the **exact** endpoint of an existing line.
6. Move the mouse so the temporary line goes to the right from the starting point, and then type **2.05** at the keyboard and press the ENTER key.
7. Move the mouse to the bottom of the existing vertical line. While the cross-hairs are on the screen click the mouse button to start a new horizontal line. Now move to mouse to the right side of the horizontal line (created above) at the top of the figure, and click the mouse. This will create a horizontal line which is the same length as the first horizontal line.







8. Select the **Line** tab on the toolbar and then select the  button to draw a line between two points.
9. Select the endpoint on the right side of the upper horizontal line.
10. Select the endpoint on the right side of the lower horizontal line.

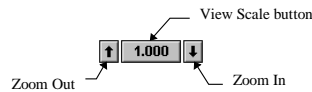


11. At this point there should be a rectangle on the screen.

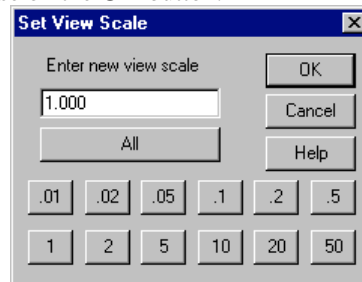
Viewing the drawing area


This section is provided to show you how to move around, zoom in, and zoom out in a DeltaCad drawing. You may want to come back to this section after the calculator drawing is completed.

1. Click the mouse on the , , , or  symbols on the horizontal or vertical scroll bars at the bottom and right side of the DeltaCad drawing window and observe how to move around in the drawing.
2. Click the mouse on the Zoom-In and Zoom-Out buttons on the status bar at the bottom of the DeltaCad window and observe the effect. Zooming out gives you a way to see the whole drawing, while zooming in allows you to see small details.

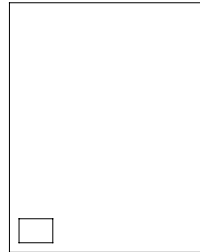






3. Click the mouse on the View Scale button on the status bar and then click one of the pre-defined scales, or enter a scale in the box provided, and click the mouse on the **OK** button.



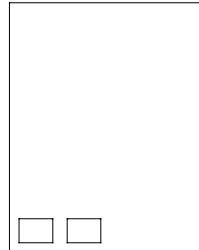
4. Select the **View** tab on the toolbar and then select the  button, or select **All** from the **View** menu, or hold down the control key and press the **A** key. This will scale the drawing so the entire drawing fits on the display.
5. Repeat step 3 above and change the View Scale to **1.0**. This will size the drawing correctly for the rest of the tutorial.




Draw the first calculator key

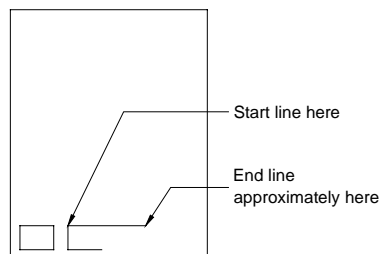



1. Select the **Point** tab on the toolbar and then select the  button to create a point. This point will be used as a reference point to draw the first calculator key.
2. Click the mouse at the lower left corner of the rectangle on the screen, and then type **0.1,0.1** on the keyboard and press the ENTER key. This will create a temporary point that is offset 0.1 inches to the right, and 0.1 inches above the rectangle corner. The temporary point will be used to draw the first calculator key.
3. Select the **Line** tab on the toolbar and then select the  button to draw a horizontal line.
4. Click the mouse on the point created above, move the mouse to the right from the starting point, and enter **0.35** on the keyboard.
5. Select the **Line** tab on the toolbar and then select the  button to draw a vertical line.
6. Click the mouse on the left (or right) side of the horizontal line just created, move the mouse up from the horizontal line, and enter **0.25** on the keyboard.
7. Repeat the previous step for the right side of the calculator key.
8. Select the **Line** tab on the toolbar and then select the  button to draw a line between two points.
9. Draw a line completing the calculator key by clicking the mouse at the free end of both of the vertical lines of the calculator key.

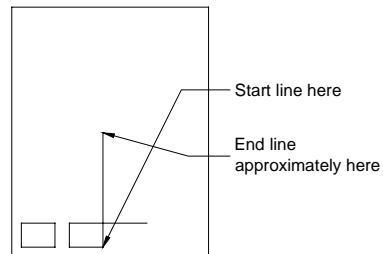
Draw the second calculator key




1. Select the **Point** tab on the toolbar and then select the  button to create a point.
2. Click the mouse at the lower right corner of the first calculator key, and then enter **0.15,0** on the keyboard. This will create a temporary point at the corner of the second calculator key.
3. Select the **Line** tab on the toolbar and then select the  button to draw a line.
4. Click the mouse on the point just created.
5. Create the left vertical line of the second key by entering **0,0.25** on the keyboard. This will create a line that goes 0.25 inches vertically from the starting point.
6. Click the mouse at the bottom of the line just created, and enter **0.35,0** on the keyboard to create the bottom horizontal line.
7. Select the **Line** tab on the toolbar and then select the  button to draw a horizontal line.
8. Click the mouse on the top of the vertical line created above, then move the mouse approximately as shown below and press the mouse button to end the line.

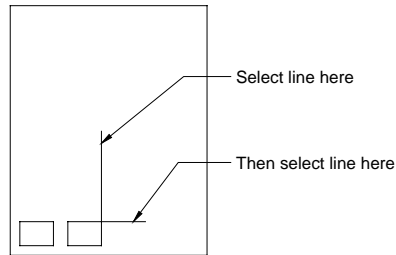


9. Select the **Line** tab on the toolbar and then select the  button to draw a vertical line.
10. Click the mouse on the right side of the first horizontal line created above, then move the mouse approximately as shown below and press the mouse button to end the line.

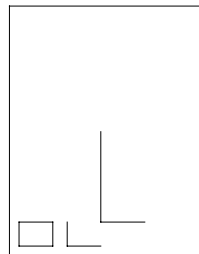


11. The following steps will demonstrate how to make a corner with two lines, and how to use the undo function.
12. Select the **Edit** tab on the toolbar and then select the  button to make a corner.

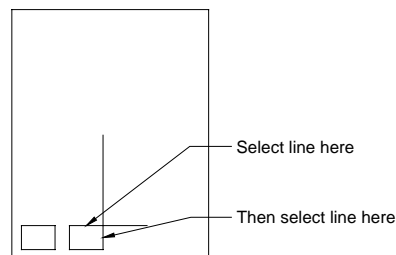
13. Select the two long lines of the second calculator key as shown below. It is important to select the lines close to the locations shown below.



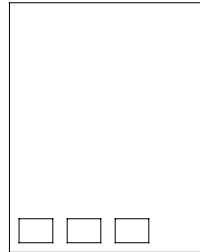
14. The previous step created the figure shown below. This did not create the corner as needed to complete the calculator key. This is because the corner function trims or extends the lines based on which side of the intersection point they are selected on. The following two steps will correct the problem to correctly draw the second calculator key.






15. Select **Undo** from the **Edit** menu to undo the previous corner function.
16. Select the two lines of the second calculator key as shown below. It is important to select the lines close to the locations shown below.

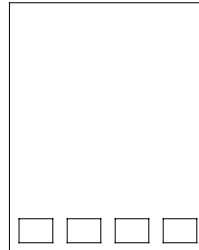




Draw the third calculator key



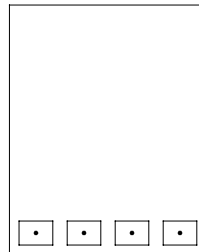
1. Select the **Line** tab on the toolbar and then select the  button to draw a parallel line.
2. Move the mouse to the vertical line on the right side of the second calculator key. Notice that when the mouse is moved near the line, the line changes to its highlight color. While the mouse is over the line, click the mouse button to select the line.
3. Move the mouse to the right from the starting line, and then enter **0.15** on the keyboard. This will create a new line that is offset 0.15 inches from the first line.
4. Move the mouse to the line just created, and click the mouse to select the line.
5. Move the mouse to the right from the starting line, and then enter **0.35** on the keyboard.
6. Select the **Line** tab on the toolbar and then select the  button to draw a line between two points.
7. Draw a line between the bottoms of the two vertical lines by clicking the mouse at the bottom ends of each line.
8. Select the **Line** tab on the toolbar and then select the  button to draw a parallel line.
9. Move the mouse to the horizontal line just created, and then click the mouse to select the line.
10. Move the mouse to the top of one of the two vertical lines of the third calculator key, and click the mouse on the end of the line.


Draw the fourth calculator key



1. Select the **Point** tab on the toolbar and then select the  button to create a point.
2. Click the mouse at the lower right corner of the third calculator key, and then enter **0.15,0** on the keyboard. This will create a temporary point at the corner of the fourth calculator key.
3. Select the **Shapes** tab on the toolbar and then select the  button.
4. Click the mouse on the point just created.
5. Enter **0.35,0.25** on the keyboard. This will create a rectangle that is 0.35 inches wide and 0.25 inches tall.

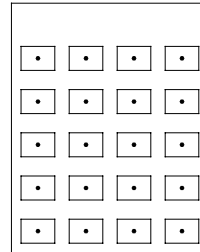
Draw center points inside each calculator key.




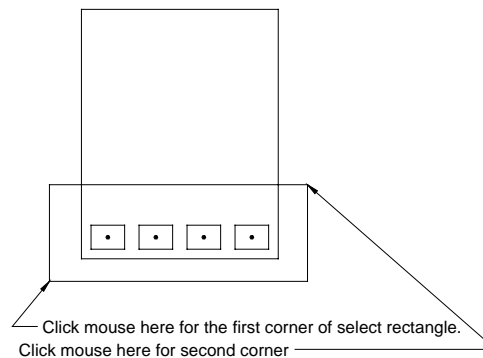
1. Select the **Point** tab on the toolbar and then select the  button to create a point.
2. Click the mouse at the lower left corner of the first calculator key.
3. Click the mouse at the upper right corner of the first calculator key to create a point in the center of the key. The point function will create a mid-point if two **existing** object points are picked.
4. Repeat the above two steps for each of the keys.

Create all remaining calculator keys.



This section will copy the first four calculator keys to create all of the other keys.

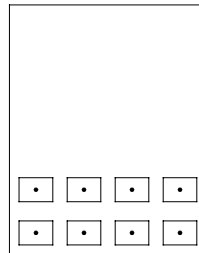


1. Select the **Select** tab on the toolbar and then select the  button.
2. Click the mouse approximately as shown in the figure below to select all of the calculator keys, and part of the outline rectangle.



3. Notice that all of the objects in the select rectangle have changed to their highlight color.
4. The next step is to unselect the three lines of the outline rectangle, without unselecting the calculator keys.
5. Press, and hold down, the control key. Click the mouse on each of the three lines of the outline rectangle. Then release the control key.
6. If the control key is held down then all objects which are picked will toggle to their opposite selection state. If an object was selected, it will unselect. If an object was unselected, it will select.
7. At this point only the four calculator keys should be highlighted.

8. Select the  button on the toolbar to select the move function.
9. Click the  button on the toolbar to instruct DeltaCad to leave a copy behind after the move function. Notice that the copy button turned red to indicate the copy function is turned on.
10. Enter **0,0.4** on the keyboard to copy the objects 0.4 inches upward from the starting location.

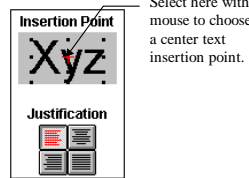


11. Now the second row of calculator keys has been created. Notice that the Instruction line on the status bar reads as follows:
Select new base point, enter X,Y, or press ENTER for (0,0.4)
12. Press the ENTER key three times to copy the calculator keys three more times.
13. Press the Escape key to unselect all objects.

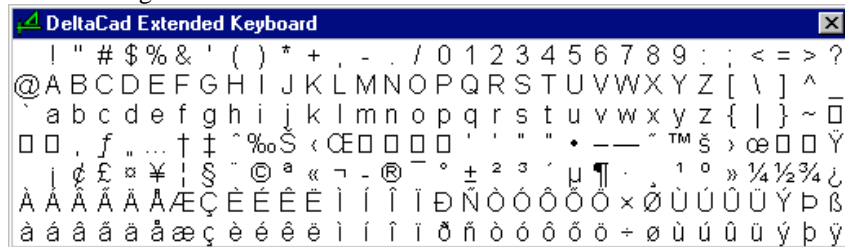
Fill in calculator key text.

C	CE	%	÷
7	8	9	x
4	5	6	-
1	2	3	+
+/-	0	.	=

1. Select the **Text** tab on the toolbar and then select the **Tx** button.
2. Select the **XYZ** button on the toolbar to change the location of the text insertion point.
3. Select a center text insertion point as shown below.




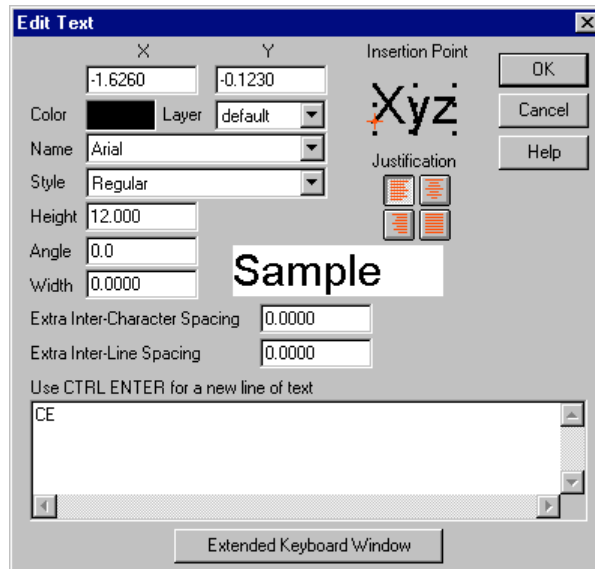
4. Click the mouse on one of the calculator key center points, then type in the calculator key text, and press ENTER.
5. Repeat the previous step for all of the keys.
6. The symbol for the "divide by" key is not on most computer keyboards. To enter non-standard text, select the text insertion point, press the F12 key, click the mouse on the desired symbol, and then press the ENTER key. The "divide by" symbol is on the bottom row about 3/4 of the way to the right side of the table.



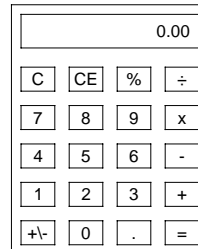
Changing Text








Use the following procedure to change an existing piece of text in a DeltaCad drawing. Try out the procedure by changing some of the calculator key text, and then change the text back to the correct calculator key text.

1. Select the **Edit** tab on the toolbar and then select the  button to edit text.
2. Select the piece of text to edit by clicking the mouse near the baseline of the text. As the mouse is moved near the text baseline, a red baseline will appear on the screen.
3. Edit the appropriate text data in the popup window, and then click the **OK** button.

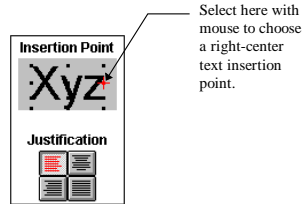



Draw the calculator display




1. Select the **Point** tab on the toolbar and then select the  button.
2. Click the mouse at the upper left corner of the calculator outline rectangle, and then enter **0.1,-0.1** on the keyboard. This will create a temporary point at the upper left corner of the calculator display.
3. Select the **Shapes** tab on the toolbar and then select the  button.
4. Click the mouse on the point just created.
5. Enter **1.85,-0.35** on the keyboard to create a rectangle.
6. Select the **Point** tab on the toolbar and then select the  button.
7. Click the mouse on the "Snap to Midpoint"  indicator at the lower right corner of the status bar. After the indicator is selected it will change to red .
8. Click the mouse somewhere on the line that forms the right edge of the rectangle created above, and then enter an offset of **-0.15,0** on the keyboard. This will create the text insertion point for the calculator display text.
9. Select the **Text** tab on the toolbar and then select the  button.
10. Select the  button on the toolbar to change the location of the text insertion point to right-center.

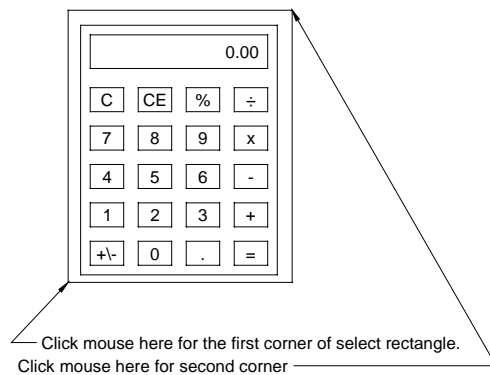
11. Select a right-center text insertion point as shown below.



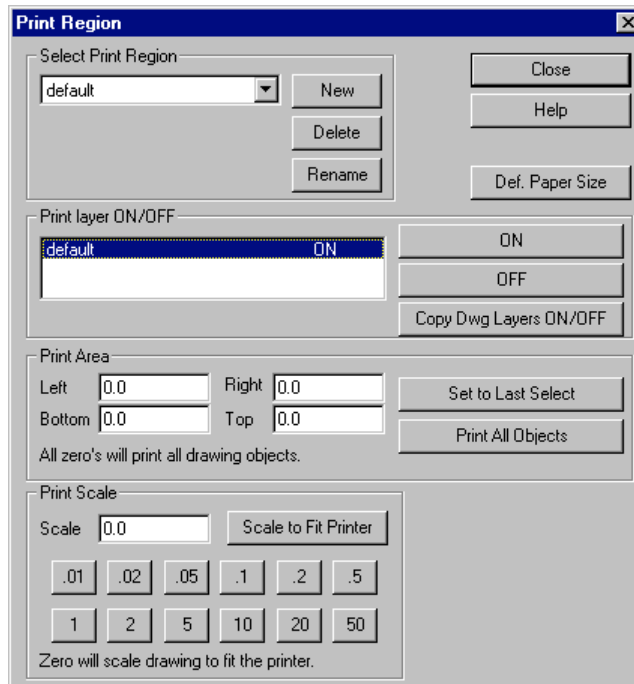
12. Click the mouse on the text insertion point created above, and then enter **0.00** for the display text.
13. Select the **Point** tab on the toolbar and then select the  button. This will erase the temporary points that were created.

Printing the Calculator

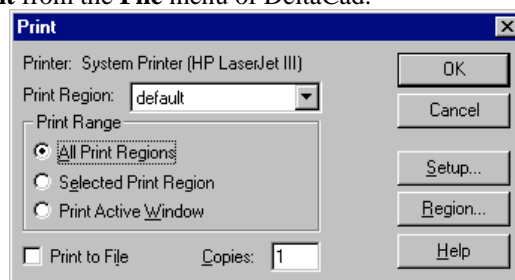
1. In order to print the calculator to scale a print region must be set. A print region is a rectangular part of the drawing that is printed at a particular scale.
If you just want a quick print, where all drawing objects are printed, and the print is scaled to fit the printer you can use the default print region.
2. Select the **Select** tab on the toolbar and then select the  button.
3. Click the mouse approximately as shown in the figure below to select the part of the calculator drawing to be printed.



4. Press the Escape key to unselect everything.
5. Select **Set Print Region** from the **File** menu of DeltaCad.



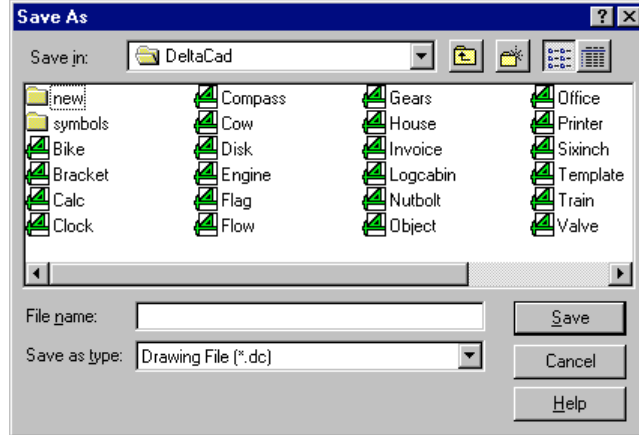
6. Click the mouse on the **Set to Last Select** button. This will change the Left, Right, Bottom, and Top boxes to the coordinates of the rectangle that was selected above.
7. Set the print scale by clicking the mouse in the Scale box, and then change the scale to **1.0**, or click the mouse on the **1** button in the Print Scale section of the popup window.
8. Click the mouse on the **Close** button.
9. Select **Print** from the **File** menu of DeltaCad.




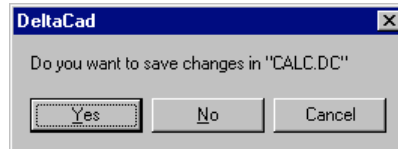
10. Click the mouse on the **OK** button to print the calculator.

Saving and Exiting DeltaCad

1. To save the existing drawing pick **SaveAs** from the DeltaCad **File** menu.



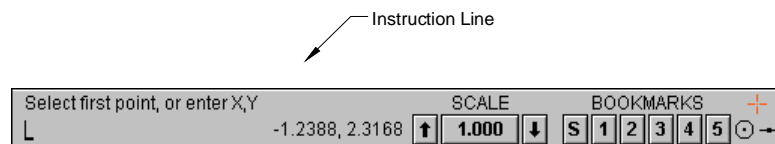
2. Click the mouse in the File Name box and enter a name for the calculator file. If no file extension is entered, DeltaCad will add **.DC** for the file extension. If desired, the disk drive or directory may be changed by selecting the new drive or directory. After the filename, drive, and directory are correct select the **OK** button with the mouse.
3. To exit DeltaCad pick **Exit** from the DeltaCad **File** menu, or double click the  button on the title bar. If there are any open files which have changed since the last time they were saved a window will appear prompting you whether the drawing should be saved or not.



End of Tutorial

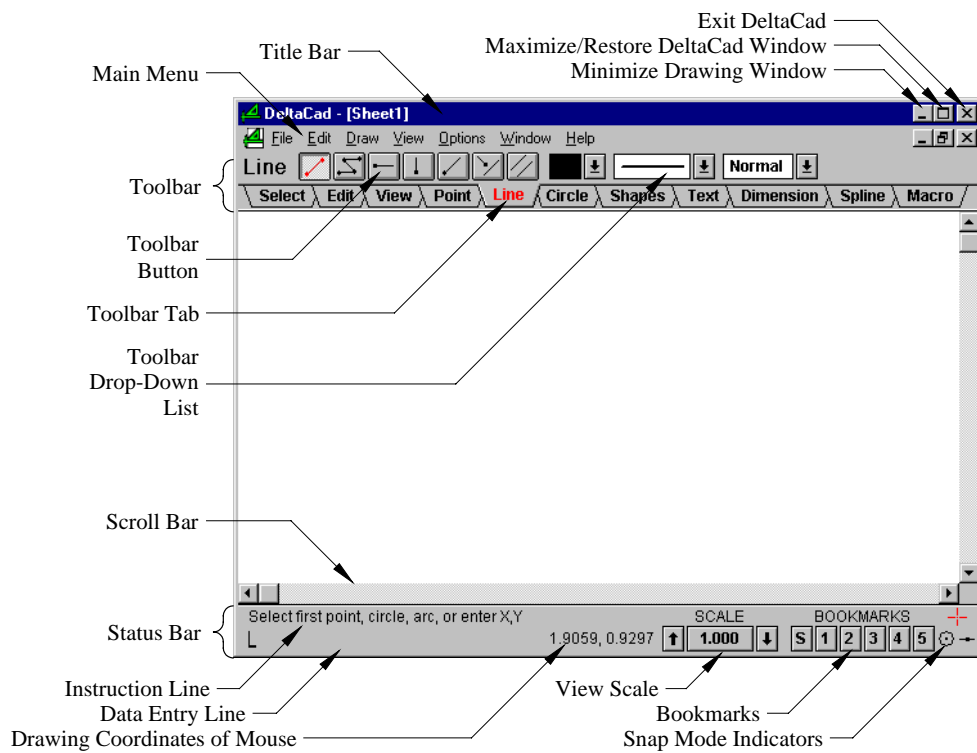
This ends the DeltaCad tutorial. At this point you should have a basic working knowledge of the main features of DeltaCad. To learn the rest of DeltaCad's features, read each section of the manual and then try out each new feature on your own computer.

REMEMBER: Always read the instruction line on the status bar.

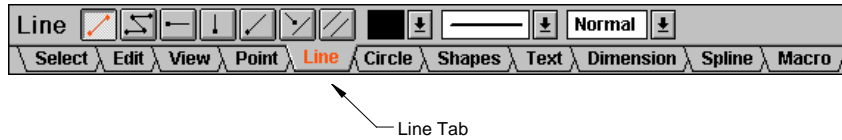


Getting Started

Screen Layout



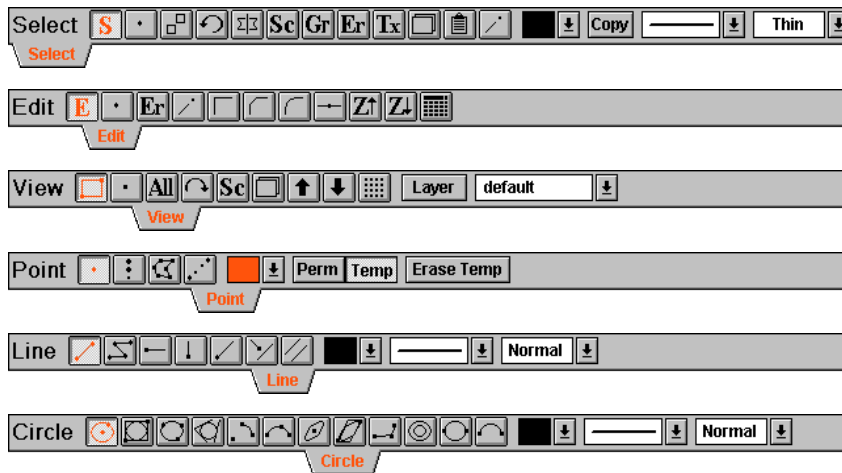
Toolbar

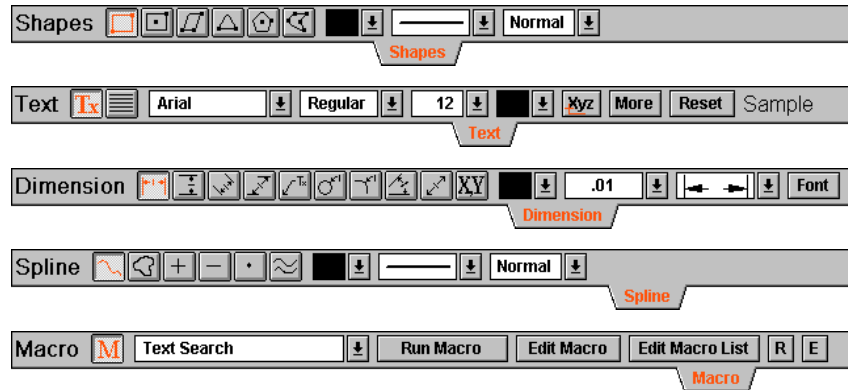


The toolbar is the part of the DeltaCad window that is located just below the main menu. The toolbar has the following three main sections:

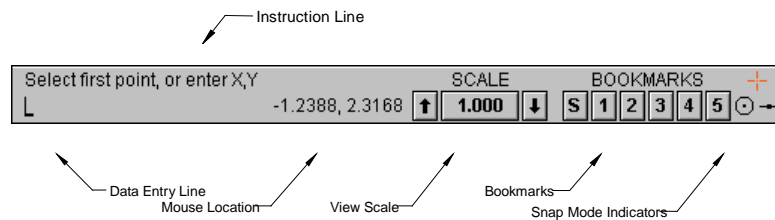
1. **Buttons:** The buttons are used to select which function to perform. As the mouse is moved over each toolbar button, a small window will appear which displays a line of text explaining what each button is used for. This means you do not have to memorize what the symbol on each toolbar button means.
2. **Drop-Down Lists:** The drop-down lists are used to select colors, line types, line weights, text height, text orientation, text justification, etc.
3. **Tabs:** The tabs are used to select the major type of object to draw. As each tab is selected all of the buttons and drop-down lists shown above the tabs will change.

The toolbar may be toggled on or off by selecting **Toolbar** from the **View** menu. There are actually nine different toolbars. To select the desired toolbar, click the mouse over the toolbar tab. For example, to select the **Circle** toolbar, click the mouse over the word "**Circle**" on the toolbar, then the **Circle** toolbar will be displayed. The following figures show all of the toolbars.





Status Bar



The status bar provides you with information, and is used to enter information on the Data Entry Line. It is located at the bottom of the DeltaCad window. The status bar may be toggled on or off by selecting **Statusbar** from the **View** menu. The status bar contains the following six sections:

1. **Instruction Line:** The instruction line provides you with short instructions for the current function in progress. This helps guide you through the steps needed to create an object. The instruction line makes it easy for a new user to learn about a new function by exploring.
The instruction line also displays a pop-up description of each menu item. To display the menu item text, hold the mouse button down as it is moved over the menu item.
2. **Data Entry Line:** This line is the main keyboard interface to DeltaCad. When the Instruction Line prompts you to enter data, this is where it is displayed as it is typed in.
3. **Mouse Location:** This portion of the Date Entry Line gives the current location of the mouse (in drawing coordinates). The mouse location is only displayed if you have not entered any data with the keyboard.
4. **View Scale:** This section displays the current View Scale, allows you to Zoom In, Zoom Out, or to change the View Scale. If the Up Arrow is picked with the mouse, the drawing Zooms Out, and if the Down

Arrow is picked the drawing Zooms In. If the box displaying the View Scale is picked with the mouse a window will appear which allows you enter a new View Scale, or to pick one of the standard scales.

5. **Bookmarks:** This section shows the five bookmarks in DeltaCad. Bookmarks are location and scale holders. This provides a convenient way to return to a particular part of a drawing. To set a bookmark you pick the "S" and then pick one of the bookmark numbers "1-5". Then any time you need to return to the bookmark location you simply pick the bookmark number "1-5" with the mouse.



6. **Snap Mode Indicators:** This section displays the current snap mode. The upper indicator will toggle between snap to nearest point, snap to grid, or snap off. The lower indicators are only visible when in the snap to nearest point mode, and they are used to snap to the center of a circle, or to the midpoint of a line. For a further description of the snap modes go to the **Mouse** section later in this chapter.



Main Menu

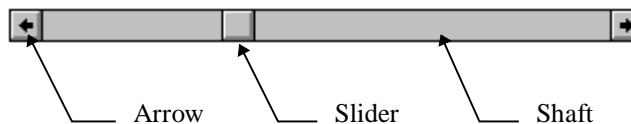


This is the main menu for DeltaCad. To execute a menu function, click the mouse over the desired menu function. After the menu item has been selected from the main menu bar, a drop down menu will appear. To pick an item from the drop down menu just click the mouse over the desired function. The main menu contains all of the standard Windows™ functions, plus all of the DeltaCad drawing functions and options. The following list shows all of the DeltaCad menu items:

- **File:** New, Open, Symbols, Close, Save, SaveAs, Delete File, Import, Export, Print, Printer Setup, Set Print Region, Print Preview, Open Most Recently Used File, Exit.
- **Edit:** Undo, Cut, Copy, Paste, Paste Special, Delete, Select All, Unselect All, Select, Edit Object.
- **Draw:** Point, Line, Circle, Shape, Text, Dimension, Spline.
- **View:** Zoom-In, Zoom-Out, All, Scale, Box, Point, Previous, Set Layer, View Layer, Snap to Nearest Point, Snap to Grid, Snap Off, Grid, Offset, Extended Keyboard, Calculator, Toolbar, Statusbar.
- **Options:** Macro, Drawing Scale, Drawing Init. Size and Pos., Drawing Data, Options, Extract Group Data.
- **Window:** New Window, Tile, Tile Horiz, Tile Vert, Cascade, Arrange Icons, Close All
- **Help:** DeltaCad Help, Current Toolbar Button, DeltaCad PDF Help, Basic DeltaCad Extension, Basic Macro Language Ref., Basic Macro Dialog Design Ref., DeltaCad License, Order DeltaCad, Register, About DeltaCad.

Scroll Bars

A Scroll bar is the horizontal bar at the bottom of each drawing window, and the vertical bar on the right side of each drawing window. Scroll bars are used to move around in the drawing. Scroll bars consist of the following three areas:



Arrow The arrow at each end of a scroll bar is used to scroll the drawing in the direction of the arrow. To scroll the drawing to the right you would pick the arrow at the right end of the scroll bar. To scroll the drawing by a smaller amount (1/10), hold down the control key when the mouse is clicked on the scroll arrow.

Shaft The scroll shaft is the portion of the scroll bar between the arrow and the slider. The scroll shaft is used to scroll the drawing by a larger amount. Each time the scroll shaft is picked, the drawing scrolls almost one entire screen width (there is a small overlap for clarity).


Slider The slider is the small box located somewhere in the middle of the scroll bar. The slider is a pointer to show where in the drawing the current view is located. For example, if the currently displayed portion of the drawing is approximately 2/3 of the way from the left to the right edge of the drawing, then the slider will be displayed 2/3 of the way from the left to the right edge of the scroll bar.

The slider may also be used to move around in the drawing. To move the slider you have to drag it with the mouse. To drag the slider, position the mouse over the slider, and press the mouse button down. Then, while holding the mouse button down, move the slider to its new location, and release the mouse button. After the slider has been moved, the drawing will move to the new location shown by the slider. This is useful for a large drawing when you need to move a large distance.


Mouse Wheel


If your mouse has a wheel it can be used to move up, down, left, or right in your drawing. You can also use it to zoom in and out. To move up or down, just turn the wheel. To move left or right, hold down the SHIFT key, and turn the wheel. To zoom in or out, hold down the CTRL key, and turn the wheel.

Minimize

When the minimize button  is picked with the mouse the DeltaCad window, or the drawing window, is reduced to an icon. This is useful when there are several windows overlapping each other.

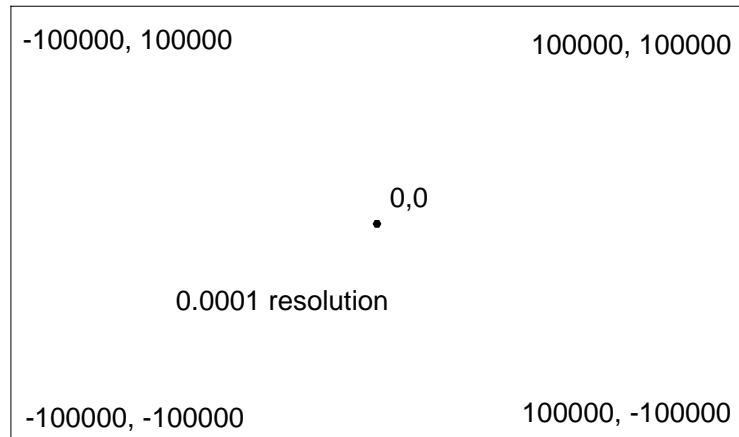
Maximize/Restore

When the maximize button  is picked the window is enlarged to its maximum size (usually the entire screen). After the window is maximized, the maximize button changes to the restore button shown below.

When the restore button  is picked the window is restored to the size it was before it was maximized. After the window is restored to its previous size, the restore button changes to the maximize button shown above.

Drawing Area

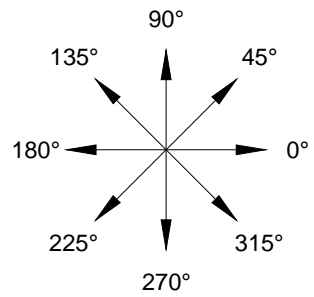
The drawing area



The drawing area consists of an area 200,000 units wide and tall. The drawing area goes from -100000.0000 to +100000.0000 units both horizontally and vertically, and it has a resolution of 0.0001 units. Your drawing may exist anywhere in this area. Most drawings only use a small portion of the total area available. The units used may be inches, centimeters, millimeters, or any other units. See the **Drawing Scale** section below for information on how to set the size of the units.

Angles

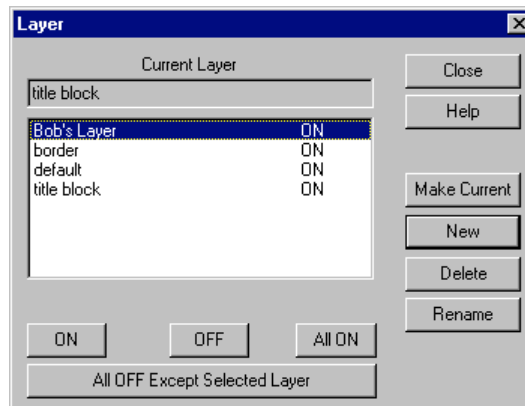
All angles in DeltaCad are entered in degrees, where zero is pointing to the right, and the angles get larger as they go counterclockwise. See the figure below:



Layer

There are 256 different layers that can be used for drawing. Each layer can be given a name, and can be turned on or off for viewing or printing. Layers are a way to organize the various objects in a drawing. You can gather common objects, such as the objects that make up the drawing border, and put them all on the same layer.

When you pick the **Set Layer** command on the **View** menu, or pick the **Layer** button on the **View** toolbar a pop-up window will appear. To change the layer data first select which layer to change by picking the layer in the list, then select the appropriate button to create a New layer, Delete a layer, Rename a layer, Set the Current Layer, or turn a layer on or off. The current layer may also be changed using the current layer drop-down list on the **View** toolbar.



Each new drawing object is placed on the **Current Layer**. You can change the layer of an object by editing the individual object, or by selecting one (or more) objects and then changing the layer of all selected objects.

Non-Printing Layer

It is useful to place objects on a non-printing layer when you want to include information in a drawing but do not want the information to show up on the final printout. To create non-printing information simply create a new layer using the **Set Layer** command on the **View** menu, draw your non-printing objects, then turn the layer off for printing using the **Set Print Region** command on the **File** menu. After creating the non-printing objects, don't forget to switch back to a printing layer using **Set Layer** on the **View** menu.

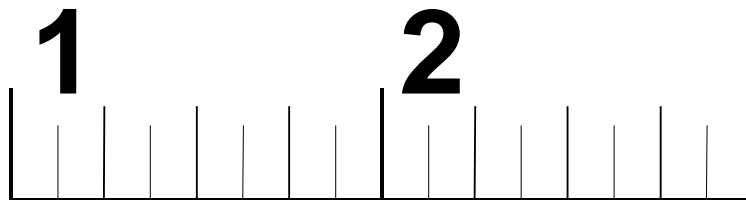
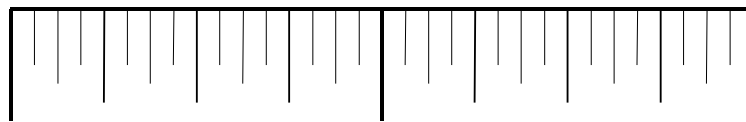
View Scale

View scale is the ratio of the displayed size of the object to the actual size of the object (times the drawing scale, see Drawing Scale below). View scale allows you to zoom in to draw fine details, or zoom out to see the whole drawing. The current view scale is shown on the status bar.

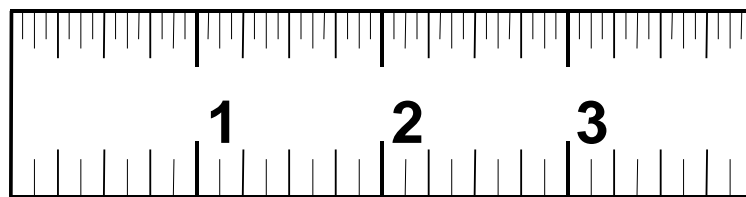


View Scale

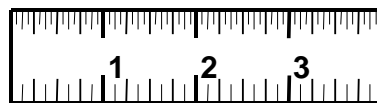
For example, if an object is displayed at twice its normal size the view scale would be one half (0.500). See the following examples.



View Scale = 0.5



View Scale = 1.0



View Scale = 2.0



View Scale = 10.0

Drawing Scale

Drawing scale is just like view (or print) scale, except it is applied to the drawing **before** the view scale. So the actual scale factor used is the drawing scale times the view scale. Drawing scale is used to set the size of a drawing unit. The following tables illustrates how to set the drawing scale for various drawings:

Drawing Scale	Set up drawing for units in
1.00	Display in Inches
2.54	Display in Centimeteres
120.0	Display 10 Feet in 1 Inch
48.0	1/4"=1'
96.0	1/8"=1'

Drawing scale is calculated by dividing the number of drawing units to be displayed by the distance to display the drawing units (in inches).

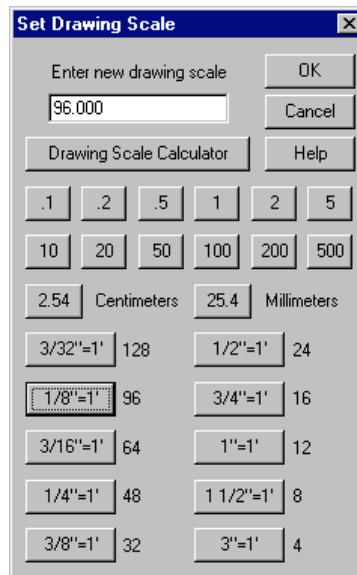
For example, if you want your drawing in meters and you want to display 10 meters in 9 millimeters (or 0.3543 inches) you would set the drawing scale to 28.22 (or $10 \div 0.3543$). Another example is if you want your drawing in inches, and you want to display 12 inches in 1/4 inch (i.e. 1/4"=1') you would set the drawing scale to 48 (or $12 \div 1/4$). You can have DeltaCad perform this calculation for you by picking the **Drawing Scale Calculator** button, see the figure below.

Drawing scale is used mostly for metric drawings or drawings which are much larger than the printed output, such as house plans. For example, the sample drawing HOUSE.DC was drawn using a drawing scale of 96.00 (1/8"=1'). This means that the house is 96 times larger than it will appear at a view scale of 1.00. So every 8 feet (96 inches) of house will be displayed in 1 inch (if the view scale is 1.0). This allows the house drawing to fit on a piece of letter size paper if it is printed at a print scale of 1.0.

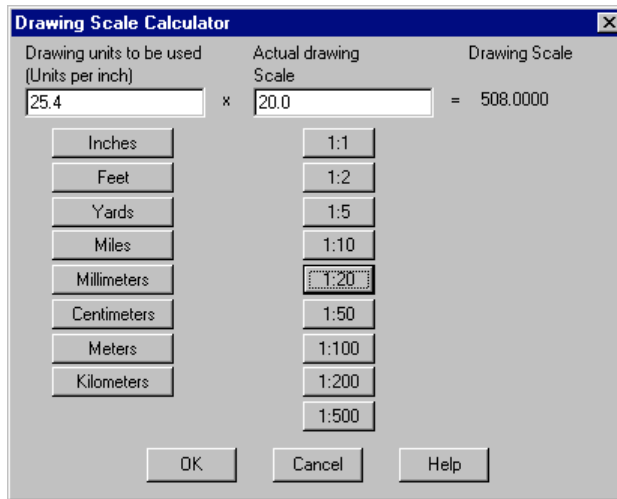
Text height is scaled with the view scale only, not the drawing scale. This allows you to input text height in familiar sizes (such as 12 point), even though the drawing may be very large. For example, to add text to the house plan drawing you would enter the text height you want on the final printout, such as 12 point (assuming a print scale of 1.0). If the house drawing was drawn at a drawing scale of 1.0 you would have to use 1152 point text to make the text visible on the final printout, and you would have to print the house at 1/96th scale to make it fit on a piece of letter size paper.

Arrow head length and the length of the dashes and gaps in dashed lines are also only scaled with the view scale, not the drawing scale. The default sizes of arrow head, dash, and gap lengths are designed to be viewed and printed at a scale of 1.0. This will make dashed lines look correct no matter what drawing scale is used.

To change the drawing scale of the current drawing, select **Drawing Scale** from the **Options** menu. Enter the new scale or pick one of the pre-defined scale buttons and then click the **OK** button.

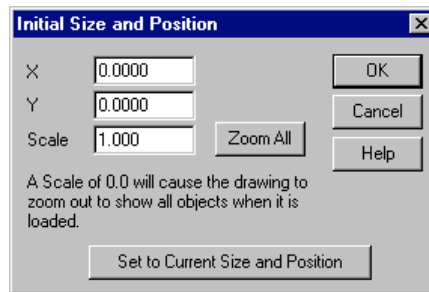


The figure shown below shows how to use the Drawing Scale Calculator to calculate the drawing scale for a drawing in millimeters with an actual scale of 1:20. To use the Drawing Scale Calculator pick **Drawing Scale** from the **Options** menu, then pick the **Drawing Scale Calculator** button.



Initial Size and Position

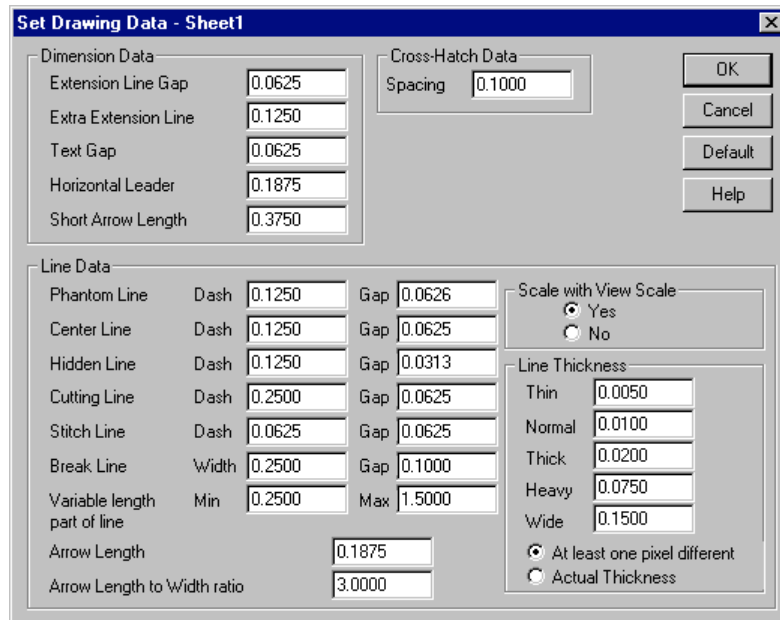
The initial size and position is the X,Y coordinates and scale used when a drawing is first opened. If you have a large drawing, and always want to go to a certain place when the drawing is opened, you would set an initial size and position. The default initial size and position is the origin 0,0 with a scale of 0.0. A scale of 0.0 causes the drawing to zoom out so the entire drawing can be seen. To set the initial size and position select **Drawing Init Size and Pos** from the **Options** menu, then enter the X,Y coordinates of the center point, and the scale.



The simplest way to set the initial size and position is to move the drawing to where you would like it to be at startup, then select **Drawing Init Size and Pos** from the **Options** menu, click the **Set to Current Size and Position** button, and then click the **OK** button. If you want the drawing to zoom out to see all drawing object set the Scale to 0.0, or click the **Zoom All** button.

Drawing Data

To change the miscellaneous data associated with the current drawing, select **Drawing Data** from the **Options** menu, and then change the desired data. This function allows you to customize your drawings to meet your specific needs. Miscellaneous data, such as the length of the dash and gap for each line type, arrow length and width, line thickness, etc. may be changed.



The dialog box titled "Set Drawing Data - Sheet1" is used for configuring drawing parameters. It is divided into several sections:

- Dimension Data:** Includes fields for Extension Line Gap (0.0625), Extra Extension Line (0.1250), Text Gap (0.0625), Horizontal Leader (0.1875), and Short Arrow Length (0.3750).
- Cross-Hatch Data:** Includes a Spacing field (0.1000).
- Line Data:** A table for configuring various line types:

Line Type	Property	Value
Phantom Line	Dash	0.1250
	Gap	0.0626
Center Line	Dash	0.1250
	Gap	0.0625
Hidden Line	Dash	0.1250
	Gap	0.0313
Cutting Line	Dash	0.2500
	Gap	0.0625
Stitch Line	Dash	0.0625
	Gap	0.0625
Break Line	Width	0.2500
	Gap	0.1000
Variable length part of line	Min	0.2500
	Max	1.5000
Arrow Length		0.1875
Arrow Length to Width ratio		3.0000
- Scale with View Scale:** Radio buttons for Yes (selected) and No.
- Line Thickness:** Fields for Thin (0.0050), Normal (0.0100), Thick (0.0200), Heavy (0.0750), and Wide (0.1500). Radio buttons for "At least one pixel different" (selected) and "Actual Thickness".

Buttons for OK, Cancel, Default, and Help are located on the right side of the dialog.

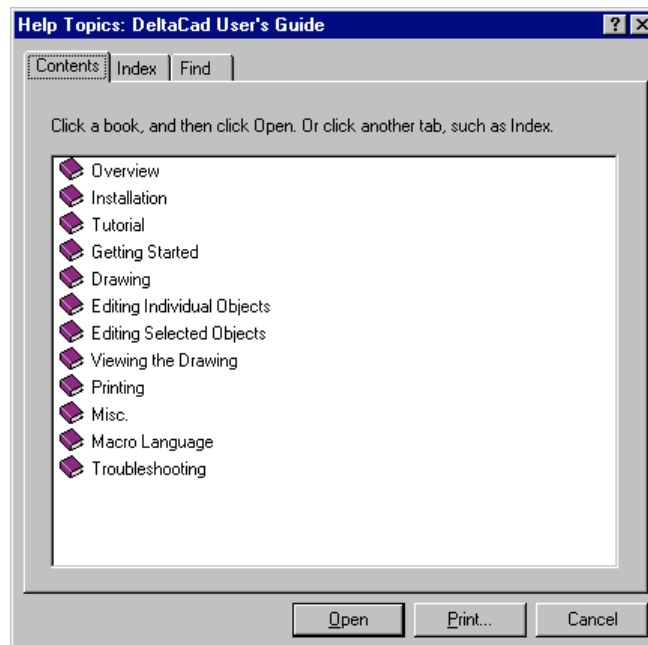
How to Use Help

DeltaCad provides an extensive on-line help system. The entire text of the DeltaCad manual is contained in the help system. You may also print out the entire manual if you have the Adobe Acrobat Reader. The Manual.pdf file is located in the DeltaCad folder (C:\Program Files\DeltaCad).

There are two ways to access the help system from within DeltaCad.

F1 Function Key: When the F1 function key is pressed, the help system will start, and the help system will display the help topic for the currently selected toolbar button.

Help Menu: When DeltaCad Help is picked from the Help menu it will start the help system. You may pick the Contents, Index, or Find tab to search the help system. When DeltaCad PDF Help is picked from the Help menu it will open the help file using Adobe Acrobat Reader. This is very helpful if you wish to print the entire contents of the help system.

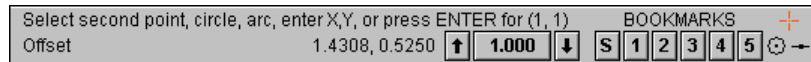


You can get help for the Basic Language Macro functions in DeltaCad by picking one of the following from the Help menu:

- Basic Macro Extension
- Basic Macro Language Ref.
- Basic Macro Dialog Design Ref.

Keyboard

All keyboard input in DeltaCad is entered on the Data Entry Line (except in popup windows) of the Status Bar.



↖ Data Entry Line

The following sections describe all of the special features of the Data Entry Line.

Feet, Inches, and Fractions

DeltaCad allows numeric input in Feet and Inches by using the ' and " symbols respectively. Also fractions may be input by separating two numbers with the / symbol. See the following examples.

Entered as	Equal to
2' 5"	29.0000
3' 6 1/2	42.5000
1'3 1/2"	15.5000
1' 3 1/2	15.5000
1'3 1/2"	15.5000

Computed Numeric Input

DeltaCad will accept numeric input in a wide variety of forms. Simple mathematical operations (+, -, *, /) and various functions (sine, cosine, tangent, etc.) may be performed as the data is entered into DeltaCad. The following table shows all of the operations that may be performed.

NOTE: The computed data input is only for data input on the Data Entry Line. Numbers input in popup windows will not be evaluated as shown below.

Symbol	Operation	Symbol	Operation
+	Addition	arcsin	Inverse sine
-	Subtraction	arccos	Inverse cosine
*	Multiplication	arctan	Inverse tangent
/	Division	ln	Natural Logarithm
()	Parenthesis	invln	Inverse Natural Log
** or ^	Raised to a power	log	Base 10 Logarithm
sin	sine	invlog	Inverse Base 10 Log
cos	cosine	dtr	Degrees to Radians
tan	tangent	rtd	Radians to Degrees
sqrt	square root	sqr	Square

The maximum and minimum numeric numbers DeltaCad will accept are 100000 and -100000 respectively. If a larger or smaller number is entered DeltaCad will truncate the number at 100000 or -100000. The following table shows how data is interpreted by DeltaCad.

Entered as	Equal to
$1' 3 \frac{1}{2}'' * 2 + 3$	34.0000
$1' 3 \frac{1}{2}'' + 3 * 2$	21.5000
$3 + 1' 3 \frac{1}{2}''$	18.5000
$5' 3 \frac{1}{2}'' * 1' 3 \frac{1}{2}''$	984.2500
sin(30)	0.5000
sqr(cos(45))*2	1.0000
(((sqr(cos(45))*2))))	1.0000
sqr(cos(45))*2	1.0000
sqr (cos(45))*2	1.0000
sqr(cos(45))*2	1.0000
sqr(cos (45))*2	1.0000
sqr(cos(45)) * 2	1.0000
(sqr(cos(45)) * 2) + 0	1.0000
sqr(sqrt(sqrt(sqr(sqr(sqr(8))))))	8.0000
1	1.0000
-1	-1.0000
--1	1.0000
-(-1)^2	-1.0000
-(-1)^3	1.0000
-(-1) **3	1.0000
sqr(2^3 * 2^4 + 1+ 3 * (3 + 4))	100000.0000 (Maximum value)
sqr(2^3 * 2^4 + 1+ 3 * (3 + 4)) -7000	100000.0000 (Maximum value)
-sqr(2^3 * 2^4 + 1+ 3 * (3 + 4)) +7000	-100000.0000 (Minimum value)
- sqr(2^3 * 2^4 + 1+ 3 * (3 + 4))+7000	-100000.0000 (Minimum value)
sin(30)	0.5000
cos(60)	0.5000
tan(45)	1.0000
arctan(1)	45.0000
arcsin(.5)	30.0000
arccos(.5)	60.0000
log(100)	2.0000
invlog(3)	1000.0000
invln(ln(2))	2.0000
invln(1)	2.7183
dtr(45)	0.7854
rtd(.785398164)	45.0000
sqr(2^sqr(sin(45)))	2.0000

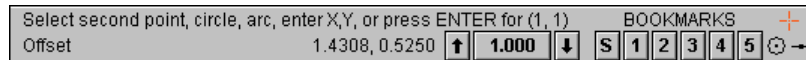
Special Text Input

DeltaCad provides a way to enter fractions with a horizontal bar between the upper and lower part of the fraction. There is also a way to create a fraction without the horizontal bar. To create a fraction you must enter `\\FR(,` followed by the fraction, followed by `)`. For a regular fraction, use the `/` symbol to separate the numbers in the upper and lower part of the fraction. To create a fraction without the horizontal bar, use the `\` symbol to separate the upper and lower part. See the following two examples:

Entered as	Displayed as
<code>5 \\FR(7/16)</code>	$5 \frac{7}{16}$
<code>2.125 \\FR(+.003\-.002)</code>	$2.125 \begin{matrix} +.003 \\ -.002 \end{matrix}$

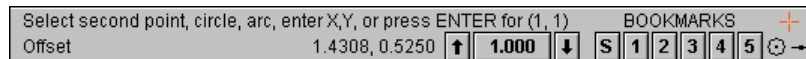
Offset F2

For most drawing functions (such as point, line, circle, etc.) which require you to enter a set of X,Y coordinates, the coordinates can either be the absolute coordinates, or an offset from the last set of coordinates entered. Pressing the **F2** function key will toggle DeltaCad between the absolute and offset modes. If the word "Offset" is shown on the Data Entry Line of the status bar then DeltaCad is in the offset mode. While in the offset mode the mouse location shown on the status bar will be the offset from the last point, not the actual X,Y coordinates. For example, when creating a line DeltaCad will prompt you to enter the position of the first endpoint, then DeltaCad will prompt you to enter the offset from the first point to the next point of the line.



Enter Key


For many of the drawing functions DeltaCad will provide a suggestion for the next thing to enter at the keyboard. The suggestion will appear at the end of the Instruction Line on the Status Bar. To use the suggestion, you simply press the ENTER key by itself. For example, when creating a line DeltaCad will prompt you to enter an offset, or just press the ENTER key to use the offset from the last line created.



If you pressed the ENTER key, in the above example, it would have the same effect as entering `1, 1`.

Right Mouse Button

The right mouse button in DeltaCad is a shortcut key which performs the same function as pressing the ENTER key. This is useful when you want to use the suggested input value shown on the status bar. For example, to create a series of equally spaced points starting at the end of a line perform the following:

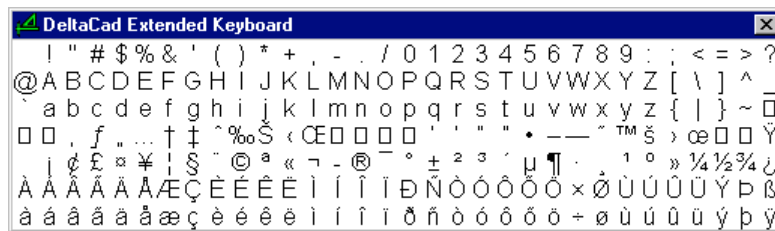
1. Select the **Point** tab on the toolbar and then select the  button to create a point.
2. Click the mouse at the end of the line, and then enter **0,-0.15** on the keyboard.
3. Press the right mouse button several times to create the additional points.



NOTE: The right mouse button only affects data on the Data Entry Line of the Status Bar (not data in popup windows), and the mouse must be over the DeltaCad window when the button is pressed.

Extended Keyboard

There are times when you need to enter a character that is not on the standard computer keyboard. To enter an extended character press the **F12** function key (or select **Extended Keyboard** from the **View** menu), then pick the extended character. When the ENTER key (or Escape Key) is pressed the extended keyboard window will disappear.



Shortcut Keys

The following shortcut keys are used in DeltaCad:




Key	Function	Key	Function
F1	Help	CTRL N	New
F2	Offset	CTRL O	Open
F3	Select	CTRL S	Save
F4	Edit	CTRL Y	Symbols
F5	View	CTRL F4	Close
F6	Point	CTRL P	Print
F7	Line	CTRL Z	Undo
F8	Circle	CTRL C	Copy
F9	Shapes	CTRL X	Cut
F10	Text	CTRL V	Paste
F11	Dimension	ALT F4	Exit
F12	Extended Keyboard	CTRL M	Toggle Snap Mode
		CTRL W	Print Preview
CTRL 1	Goto Bookmark 1	CTRL E	Edit
CTRL SHIFT 1	Set Bookmark 1	CTRL D	Slide Endpoint
CTRL 2	Goto Bookmark 2	CTRL L	Corner
CTRL SHIFT 2	Set Bookmark 2	CTRL H	Chamfer
CTRL 3	Goto Bookmark 3	CTRL R	Radius
CTRL SHIFT 3	Set Bookmark 3	CTRL T	Split
CTRL 4	Goto Bookmark 4	CTRL =	Calculator
CTRL SHIFT 4	Set Bookmark 4	CTRL A	View All
CTRL 5	Goto Bookmark 5	CTRL B	View Previous
CTRL SHIFT 5	Set Bookmark 5	Escape	End function
CTRL Up	Scroll Up	SHIFT Up	Scroll Up Less
CTRL Down	Scroll Down	SHIFT Down	Scroll Down Less
CTRL Left	Scroll Left	SHIFT Left	Scroll Left Less
CTRL Right	Scroll Right	SHIFT Right	Scroll Right Less
CTRL Page Down	Zoom In	SHIFT Page Down	Zoom In Less
CTRL Page Up	Zoom Out	SHIFT Page Up	Zoom Out Less

The **Delete** Key also has a special function. If text has been entered on the Data Entry Line of the Status Bar, then the Delete Key is used for text editing to delete characters. Otherwise the delete key is used to go to the Erase Object Function, or the Erase Selected Objects function, depending on whether there are any objects selected.


Mouse

The mouse is the primary input device for DeltaCad. It is used to select menu items, buttons, tabs, and to input X,Y locations in the drawing area.

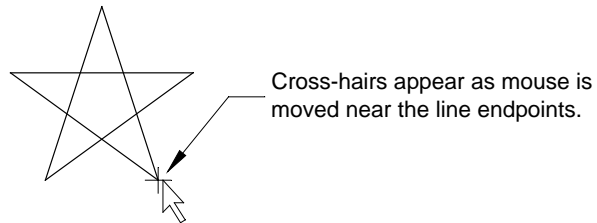
When selecting drawing locations there are three different input modes for the mouse. You can select between the modes by selecting them from the view menu, or by selecting the Snap Mode Indicator at the right side of the Status Bar. Clicking the mouse on the Snap Mode Indicator (or CTRL M) will toggle it between the three modes shown below.

-  Snap to Nearest Point
-  Snap to Grid
-  Snap Off



Snap to Nearest Point

If the **Snap to Nearest Point** mode is turned on and the mouse is moved near an existing object point a small set of cross-hairs  will appear. While the cross-hairs are displayed, if the mouse button is pressed, the entered point will be **exactly** at the object point. This is useful when you want to enter a point which is **exactly** the same as an existing point in the drawing.

You can also make the **Snap to Nearest Point** function work with a grouped object. If you add a “point object” to the group, then this function will snap to that point in the group. This can be useful when you want to snap to a specific point in a group.

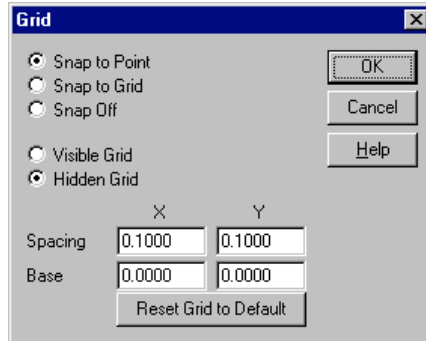


The **Snap to Nearest Point** function also contains the following two sub-functions:

1. **Snap to Center:**  This function snaps to the center of a circle or arc.
 2. **Snap to Midpoint:**  This function snaps to the midpoint of a line.
- The above two sub-functions must be turned on each time they are used. After each use they are turned off.

Snap to Grid

The **Snap to Grid** function is used to make sure all points entered are aligned with a grid. The grid spacing and base may be changed by selecting **Grid** from the **View** menu, or by picking the **Grid** button on the **View** toolbar, and then entering the data.



While in the **Snap to Grid** mode a small set of cross-hairs + will show where the grid point is located as the mouse is moved around the drawing.

Snap Off

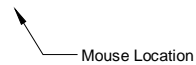
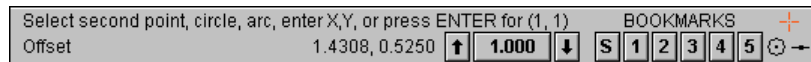
Sometimes it will be necessary to turn object snapping off to draw an object which is very close to an existing object point or grid point, but most of the time the **Snap to Nearest Point** or **Snap to Grid** mode should be left turned on.

Object Selection

Many of the drawing or editing functions in DeltaCad require you to select an existing object. When the mouse is positioned near an existing object, the object will change to its highlight color. To select an object the mouse button should be pressed while the object is highlighted.

Mouse Location

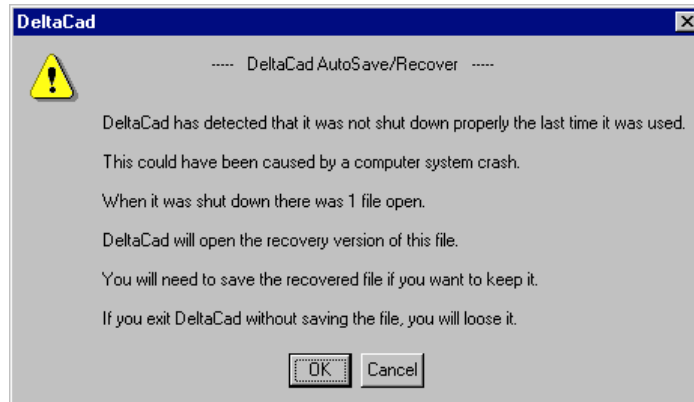
As the mouse is moved around the screen, the mouse location is shown on the status bar. The mouse location will be displayed either in X,Y drawing coordinates, or as an offset from the last point. The word "Offset" will appear at the beginning of the Data Entry Line if the mouse location is an offset from the last point.



The mouse location may also be displayed in a Feet/Inch mode, and the number of decimal places displayed may be changed by selecting **Options/Mouse Disp. Mode** from the **Options** menu.

AutoSave

DeltaCad has an auto-save feature that will save the drawing you are working on every few minutes. This way if there is a system crash you will not lose your work. You can change the auto-save time interval by picking **Options/AutoSave** from the **Options** menu, and enter the time interval. Each time DeltaCad is started it checks to see if it was shut down properly the last time it was run. If it detects that there was an open drawing which was not properly saved it will prompt you whether you want to restore the drawing.



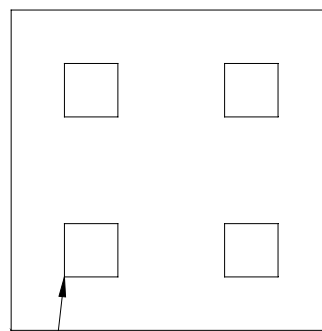
Drawing

Point

Point Objects


Points are the most misunderstood drawing element in most CAD programs. A point is defined as an object which has an X,Y location, but no size (i.e. an infinitely small dot). So what are points used for? Points are used as references when drawing other objects. For example, to draw the figure shown below perform the follow steps:

1. Create a 3 inch by 3 inch square.
2. Select the point function, and create a temporary point offset 1/2 inch by 1/2 inch from each of the corners.
3. Using the four temporary points created above, create the smaller 1/2 inch by 1/2 inch squares starting at each of the four temporary points.
4. Erase the four temporary points which were created.



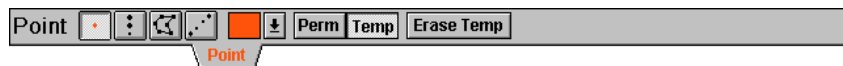
Select here with mouse.
Then enter an offset to get a point here.
Then create the smaller box.

The above figure would be difficult to draw if you did not have a way to temporarily mark the location of the starting point for the smaller squares. The only other way to draw the smaller squares would be to calculate their X,Y coordinates, and then enter them to draw the square, or to freehand the starting point of the square.


Points in grouped objects are not shown where the group is inserted, they are only shown if the group is edited by opening a group window. If the points are needed at the location where the group is inserted, the **Draw all points in an object** function  can be used to create the points.




Since points have a location, but no size, they are never printed. To print a dot on a drawing you should create a filled circle.


Point Toolbar functions



All of the following point toolbar functions may also be changed by selecting **Point/Point Options** from the **Draw** menu.


Color: Select the  drop down list to change the point color.

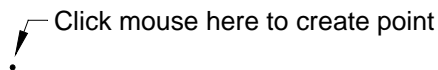
Perm/Temp points: Select the  or  buttons to select permanent or temporary points. The only difference between **Perm** and **Temp** points is **Temp** points can be erased by selecting the  button.


Erase Temp points: Select the  button to erase all temporary points. This function provides a quick and easy way to erase all of the temporary points in a drawing.

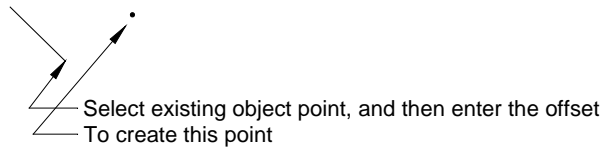
Draw a point

This function provides several different ways to draw a point.

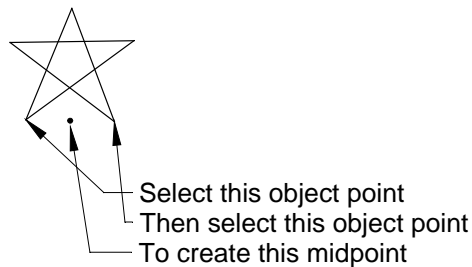
1. Select the **Point** tab on the toolbar and then select the  button, or select **Point/Point** from the **Draw** menu.
2. Any of the following may be used to create a point.
 - Click the mouse at the location for the new point, or enter the X,Y coordinates of the point with the keyboard.



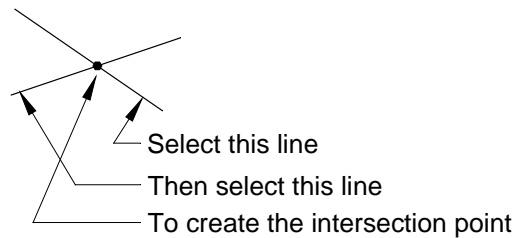
- Create an offset point by selecting an existing object point (an object point is any spot which displays the cross-hairs  as the mouse is moved over it) with the mouse, and then enter an X,Y offset with the keyboard. The offset may be repeated by pressing the ENTER key again.




- Create a single midpoint by selecting two existing object points with the mouse.

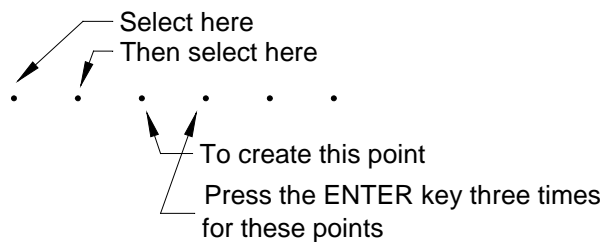


- Create an intersection point by select two existing objects, by select an existing object point and an existing object, or by select an existing object and an existing object point with the mouse.




Draw a point offset equally from the last two points

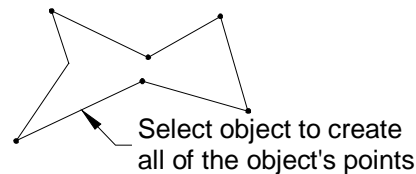
1. Select the **Point** tab on the toolbar and then select the  button, or select **Point/Equally Spaced** from the **Draw** menu.
2. Select the first object point with the mouse, or enter the X,Y coordinates of the point with the keyboard.
3. Select the second object point with the mouse, or enter the X,Y offset to the next point with the keyboard (press ENTER to use the last offset).
4. Press the ENTER key to repeat the offset. This step may be repeated.



Draw all points in an object


This function is used to create a point at all of the points associated with an object. For example, if a line is selected, this function will create a point at each end of the line. If a circular arc is selected, a point will be created at the two endpoints, and at the center point. If a grouped object is selected, a point will be created for each point in the grouped object.

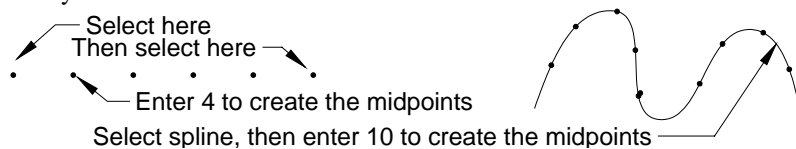
1. Select the **Point** tab on the toolbar and then select the  button, or select **Point/Object Points** from the **Draw** menu.
2. Select an object with the mouse.



Draw mid-points

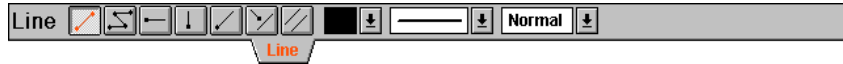
This function will create any number of midpoints between two points, or along an object.

1. Select the **Point** tab on the toolbar and then select the  button, or select **Point/Mid Points** from the **Draw** menu.
2. Select the first point with the mouse, or enter the X,Y coordinates of the point with the keyboard.
3. Select the second point with the mouse, or enter the X,Y offset from the first point to the second point with the keyboard (press ENTER to use the last offset).
4. Enter the number of midpoints with the keyboard, or press the ENTER key to use the last number of midpoints.
5. Or, just select an object, and enter the number of midpoints with the keyboard.




Line

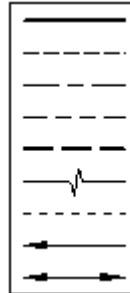
Line Toolbar functions




All of the following line toolbar functions may also be changed by selecting **Line/Line Options** from the **Draw** menu.


Color: Select the  drop down list to change the line color.

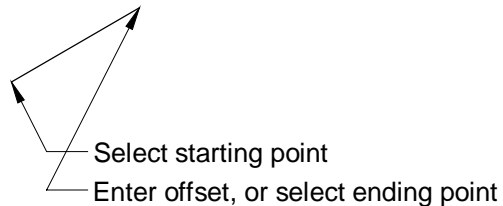
Line Type: Select the  drop down list to change the line type.



Line Weight: Select the  drop down list to change the line weight to Thin, Normal, Thick, Heavy, or Wide.


Draw a line with two points

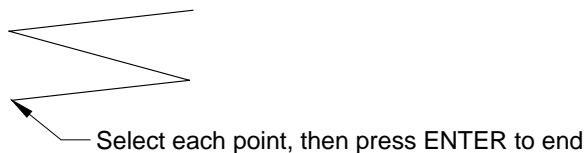
1. Select the **Line** tab on the toolbar and then select the  button, or select **Line/Line** from the **Draw** menu.
2. Select the first endpoint of the line with the mouse, select a circle or arc to create a tangent line, or enter the X,Y coordinates of the endpoint on the keyboard.
3. Select the second endpoint of the line with the mouse, select a circle or arc to create a tangent line, enter the X,Y offset from the first to the second endpoint on the keyboard, or press the ENTER key to use the offset of the last line created.




Draw a continued line

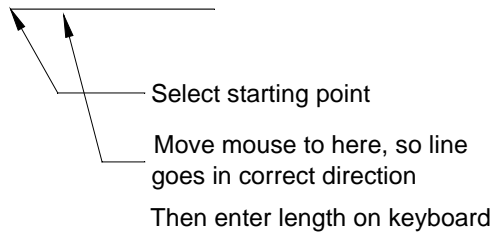
This function is used to create a line made up of many segments.

1. Select the **Line** tab on the toolbar and then select the  button, or select **Line/Continued** from the **Draw** menu.
2. Select the first endpoint of the line with the mouse, or enter the X,Y coordinates of the endpoint on the keyboard.
3. Select the next point of the line with the mouse, or enter the X,Y offset to the next point on the keyboard.
4. Repeat the previous step for each additional line segment.
5. Press the ENTER key after the last line segment has been drawn.




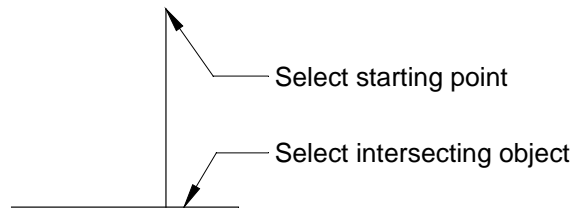
Draw a horizontal line

1. Select the **Line** tab on the toolbar and then select the  button, or select **Line/Horizontal** from the **Draw** menu.
2. Select the first endpoint of the line with the mouse, or enter the X,Y coordinates of the endpoint on the keyboard.
3. Enter the second endpoint using one of the following:
 - Select the second endpoint location with the mouse.
 - Move the mouse so the line goes in the desired direction, and then enter the length of the line on the keyboard, or just press ENTER to use the length of the last horizontal line which was created.
 - Select an intersecting object with the mouse. As the mouse is moved over the intersecting object, it will change to its highlight color.




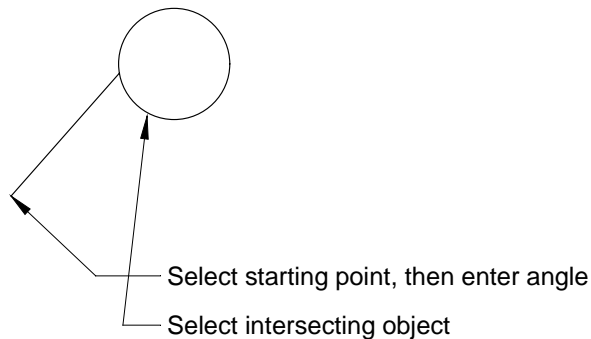
Draw a vertical line

1. Select the **Line** tab on the toolbar and then select the  button, or select **Line/Vertical** from the **Draw** menu.
2. Select the first endpoint of the line with the mouse, or enter the X,Y coordinates of the endpoint on the keyboard.
3. Enter the second endpoint using one of the following:
 - Select the second endpoint location with the mouse.
 - Move the mouse so the line goes in the desired direction, and then enter the length of the line on the keyboard, or just press ENTER to use the length of the last vertical line which was created.
 - Select an intersecting object with the mouse. As the mouse is moved over the intersecting object, it will change to its highlight color.




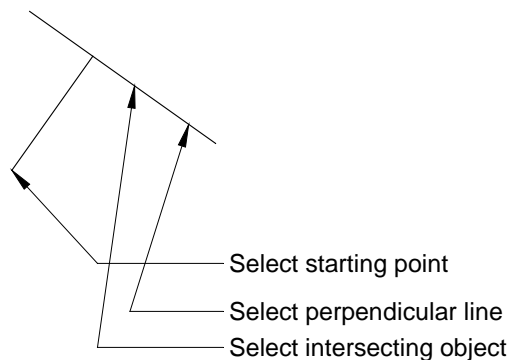
Draw an angled line

1. Select the **Line** tab on the toolbar and then select the  button, or select **Line/Angled** from the **Draw** menu.
2. Select the first endpoint of the line with the mouse, or enter the X,Y coordinates of the endpoint on the keyboard.
3. Enter the line angle using one of the following:
 - Move the mouse around until you get the desired angle, and then press the left mouse button.
 - Select an existing line which has the desired angle.
 - Select a circle or arc to create a tangent line.
 - Enter the desired angle on the keyboard. Enter the angle in degrees.
4. Enter the second endpoint using one of the following:
 - Select the second endpoint location with the mouse.
 - Move the mouse so the line goes in the desired direction, and then enter the length of the line on the keyboard, or just press ENTER to use the length of the last angled line which was created.
 - Select an intersecting object with the mouse. As the mouse is moved over the intersecting object, it will change to its highlight color.




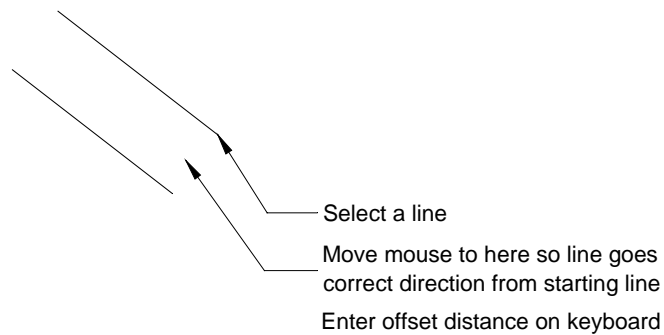
Draw a perpendicular line

1. Select the **Line** tab on the toolbar and then select the  button, or select **Line/Perpendicular** from the **Draw** menu.
2. Select the first endpoint of the line with the mouse, or enter the X,Y coordinates of the endpoint on the keyboard.
3. Enter the line angle perpendicular to the line using one of the following:
 - Move the mouse around until you get the desired perpendicular angle, and then press the mouse button.
 - Select an existing line which is perpendicular to the desired line.
 - Enter the desired perpendicular angle on the keyboard. Enter the angle in degrees.
4. Enter the second endpoint using one of the following:
 - Select the second endpoint location with the mouse.
 - Move the mouse so the line goes in the desired direction, and then enter the length of the line on the keyboard, or just press ENTER to use the length of the last perpendicular line which was created.
 - Select an intersecting object with the mouse. As the mouse is moved over the intersecting object, it will change to its highlight color.



Draw a parallel line

1. Select the **Line** tab on the toolbar and then select the  button, or select **Line/Parallel** from the **Draw** menu.
2. Select an existing line with the mouse.
3. Locate the new parallel line using one of the following:
 - Select the new location and click the mouse.
 - Move the mouse so the new line is the correct direction away from the existing line, and then enter the offset distance on the keyboard (or press ENTER to use the last parallel line offset distance).
 - Select a circle or arc to create a line tangent to the circle or arc.
4. Press the ENTER key to create more lines with the same offset as the first line. This step may be repeated.




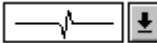
Circle

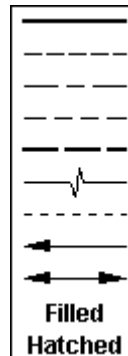
Circle Toolbar functions




All of the following circle toolbar functions may also be changed by selecting **Circle/Circle Options** from the **Draw** menu.


Color: Select the  drop down list to change the circle color.

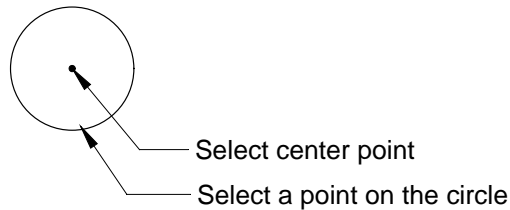
Line Type/Cross-Hatch: Select the  drop down list to change the line type, cross-hatch pattern, or select **Filled** to create a filled circle. To create a filled circle with a hole in it first create a filled circle in the desired color. Then select white for the color and create another filled circle on top of the first circle. Don't forget to change to color back from white.




Line Weight: Select the  drop down list to change the line weight to Thin, Normal, Thick, Heavy, or Wide.

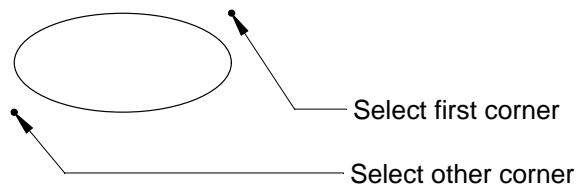
Draw a circle with center and radius

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Center-Radius** from the **Draw** menu.
2. Select the circle center point with the mouse, or enter the X,Y coordinates of the center point on the keyboard.
3. Select a point on the circle with the mouse, or enter the circle radius on the keyboard (press ENTER to use the radius of the last circle drawn).




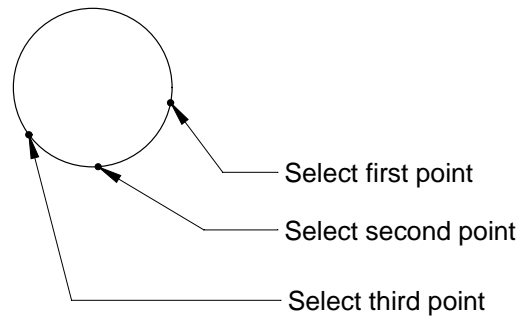
Draw a circle within a box

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Box** from the **Draw** menu.
2. Select one of the corners of the box to contain the circle with the mouse, or enter the X,Y coordinates of the corner on the keyboard.
3. Select the other corner with the mouse, or enter the X,Y offset from the first corner on the keyboard (press ENTER to use the offset of the last box circle drawn).




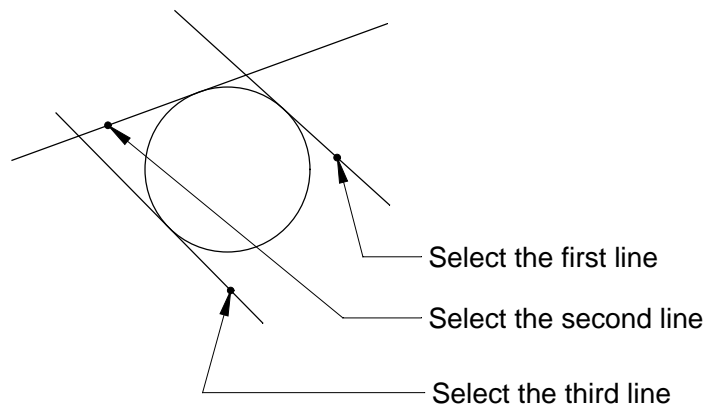
Draw a circle with three points

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Three Points** from the **Draw** menu.
2. Select a point on the circle with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the second point on the circle with the mouse, or enter the X,Y offset from the first to the second point of the circle with the keyboard (press ENTER to use the offset from the last three point circle drawn).
4. Select the third point on the circle with the mouse, or enter the X,Y offset from the second to the third point of the circle with the keyboard (press ENTER to use the offset from the last three point circle drawn).




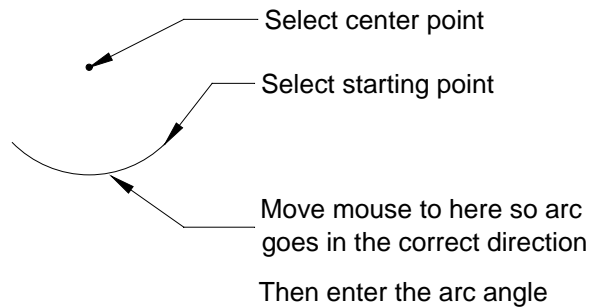
Draw a circle tangent to three lines

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Tangent** from the **Draw** menu.
2. Select the first line.
3. Select the second line, or enter the X,Y coordinates of the center.
4. Select the third line, or enter the X,Y coordinates of the center.




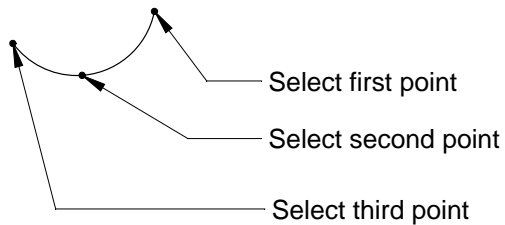
Draw a circular arc using the center

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Center-Radius Arc** from the **Draw** menu.
2. Select the arc center with the mouse, or enter the X,Y coordinates of the center on the keyboard.
3. Select an endpoint of the arc with the mouse, or enter the X,Y offset from the center to the endpoint of the arc on the keyboard (press ENTER to use the offset from the last arc drawn).
4. Enter the second endpoint using one of the following:
 - Select the second endpoint location with the mouse.
 - Move the mouse so the arc goes in the desired direction, and then enter the arc angle on the keyboard, or just press ENTER to use the angle of the last arc drawn.




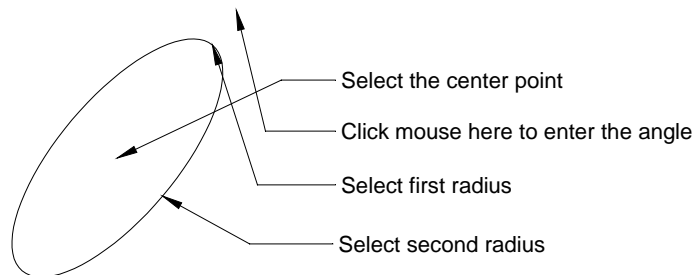
Draw a circular arc using three points

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Three Point Arc** from the **Draw** menu.
2. Select a point on the arc with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the second point on the arc with the mouse, or enter the X,Y offset from the first to the second point of the arc with the keyboard (press ENTER to use the offset from the last three point arc drawn).
4. Select the third point on the arc with the mouse, or enter the X,Y offset from the second to the third point of the arc with the keyboard (press ENTER to use the offset from the last three point arc drawn).




Draw an ellipse

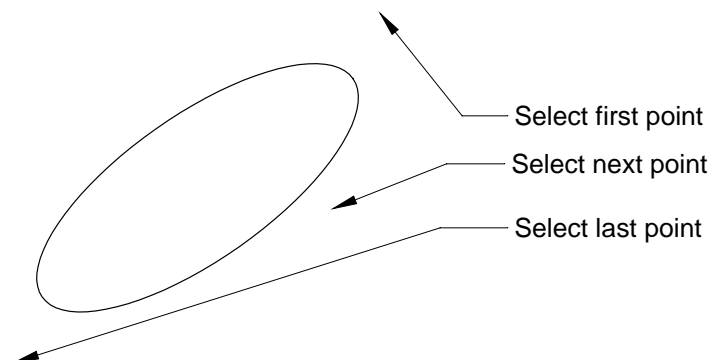
1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Ellipse** from the **Draw** menu.
2. Select the ellipse center with the mouse, or enter the X,Y coordinates of the center on the keyboard.
3. Enter the ellipse angle using one of the following:
 - Enter the angle on the keyboard, or press ENTER to use the angle of the last ellipse drawn.
 - Move the mouse around until you get the desired angle, and then press the mouse button.
 - Select an existing line which has the desired angle.
4. Select the ellipse's first radius by clicking the mouse, or by entering the radius on the keyboard (press ENTER to use the radius of the last ellipse drawn).
5. Select the ellipse's second radius by clicking the mouse, or by entering the radius on the keyboard (press ENTER to use the radius of the last ellipse drawn).



Draw an ellipse within a parallelogram

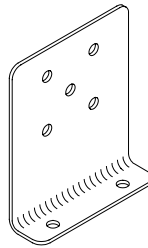
This function will draw an ellipse within a parallelogram such that the ellipse will intersect the midpoints of each side of the parallelogram. This function is useful in drawing 3-D isometric drawings.


1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Parallelogram** from the **Draw** menu.
2. Select the first point of the parallelogram with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the next point on the parallelogram with the mouse, or enter the X,Y offset from the first to the second point of the parallelogram with the keyboard (press ENTER to use the offset from the last ellipse drawn).
4. Select the third point on the parallelogram with the mouse, or enter the X,Y offset from the second to the third point of the parallelogram with the keyboard (press ENTER to use the offset from the last ellipse drawn).

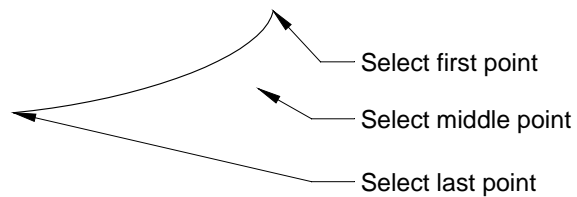


Draw an elliptical arc


This function will draw an elliptical arc segment such that the endpoints of the arc are exactly parallel with a line from the endpoints to the middle point used when drawing the arc. This function is useful when drawing 3-D isometric drawings.

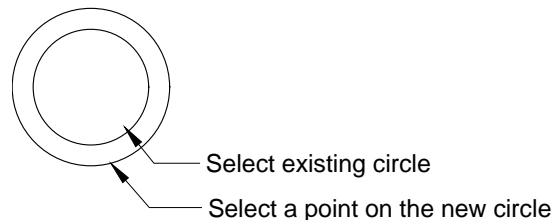


1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Elliptical Arc** from the **Draw** menu.
2. Select the first point on the elliptical arc with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the middle point of the elliptical arc with the mouse, or enter the X,Y offset from the first to the middle point of the elliptical arc with the keyboard (press ENTER to use the offset from the last elliptical arc drawn).
4. Select the last point of the elliptical arc with the mouse, or enter the X,Y offset from the middle to the last point of the elliptical arc with the keyboard (press ENTER to use the offset from the last elliptical arc drawn).




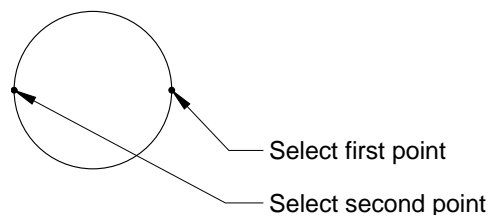
Draw a parallel circle

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Parallel Circle** from the **Draw** menu.
2. Select a circle.
3. Locate the new parallel circle using one of the following:
 - Select the new location and click the mouse.
 - Move the mouse so the new circle is the correct direction away from the existing circle, and then enter the offset distance on the keyboard (or press ENTER to use the last parallel circle offset distance).




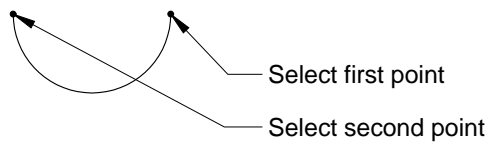
Draw a circle with two points

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Two Point** from the **Draw** menu.
2. Select the first circle point with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the second circle point with the mouse, enter the X,Y offset from the first to the second point on the keyboard, or press the ENTER key to use the offset of the last two point circle created.



Draw a 180 degree arc with two points

1. Select the **Circle** tab on the toolbar and then select the  button, or select **Circle/Two Point Arc** from the **Draw** menu.
2. Select the first arc point with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the second arc point with the mouse, enter the X,Y offset from the first to the second point on the keyboard, or press the ENTER key to use the offset of the last two point arc created. The arc will be drawn clockwise from the first point to the second point.





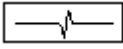

Shapes

Shapes Toolbar functions

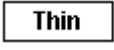



All of the following shape toolbar functions may also be changed by selecting **Shape/Shape Options** from the **Draw** menu.


Color: Select the   drop down list to change the shapes color.

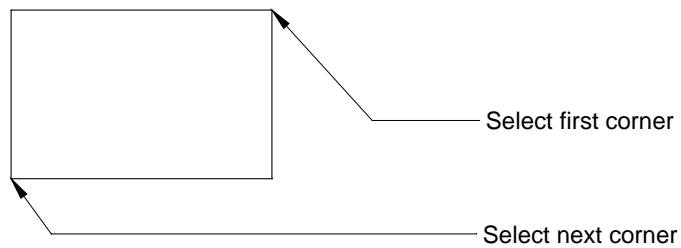
Line Type/Cross-Hatch: Select the   drop down list to change the line type, cross-hatch pattern, or select **Filled** to create a filled shape.




Line Weight: Select the   drop down list to change the line weight Thin, Normal, Thick, Heavy, or Wide.

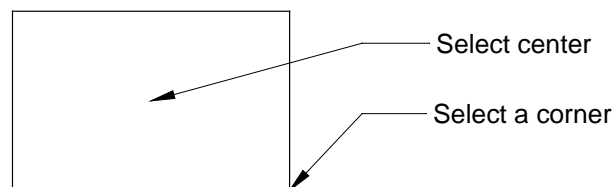
Draw a rectangle using two corners

1. Select the **Shapes** tab on the toolbar and then select the  button, or select **Shapes/Rectangle - Corners** from the **Draw** menu.
2. Select the first corner with the mouse, or enter the X,Y coordinates of the corner on the keyboard.
3. Select the second corner with the mouse, enter the X,Y offset from the first to the second corner on the keyboard, or press ENTER to use the offset of the last rectangle created.




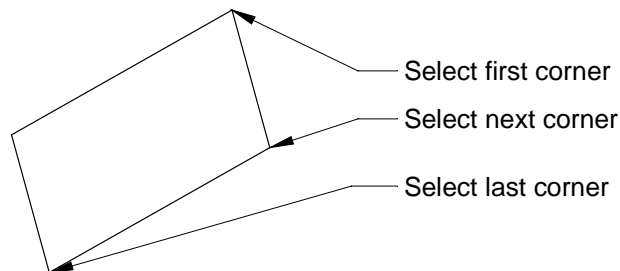
Draw a rectangle using the center and corner

1. Select the **Shapes** tab on the toolbar and then select the  button, or select **Shapes/Rectangle - Center** from the **Draw** menu.
2. Select the center of the rectangle with the mouse, or enter the X,Y coordinates of the center on the keyboard.
3. Select a corner with the mouse, enter the X,Y offset from the center to a corner on the keyboard, or press the ENTER key to use the offset of the last rectangle created.




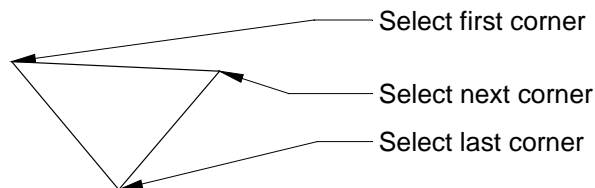
Draw a parallelogram

1. Select the **Shapes** tab on the toolbar and then select the  button, or select **Shapes/Parallelogram** from the **Draw** menu.
2. Select the first point on the parallelogram with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the next point on the parallelogram with the mouse, or enter the X,Y offset from the first to the second point of the parallelogram with the keyboard (press ENTER to use the offset from the last parallelogram drawn).
4. Select the third point on the parallelogram with the mouse, or enter the X,Y offset from the second to the third point of the parallelogram with the keyboard (press ENTER to use the offset from the last parallelogram drawn).




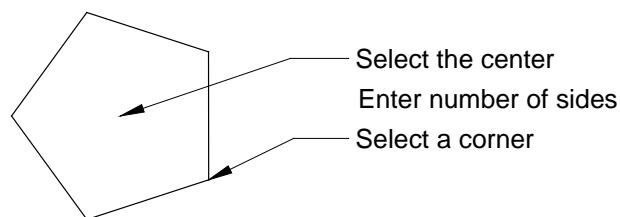
Draw a triangle

1. Select the **Shapes** tab on the toolbar and then select the  button, or select **Shapes/Triangle** from the **Draw** menu.
2. Select the first point on the triangle with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the second point on the triangle with the mouse, or enter the X,Y offset from the first to the second point of the triangle with the keyboard (press ENTER to use the offset from the last triangle drawn).
4. Select the third point on the triangle with the mouse, or enter the X,Y offset from the second to the third point of the triangle with the keyboard (press ENTER to use the offset from the last triangle drawn).




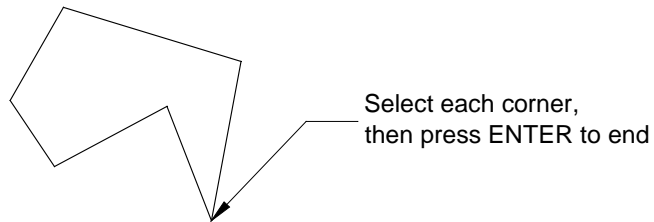
Draw a triangle, square, pentagon, etc.

1. Select the **Shapes** tab on the toolbar and then select the  button, or select **Shapes/Multi-Sided** from the **Draw** menu.
2. Select the center point with the mouse, or enter the X,Y coordinates of the center point on the keyboard.
3. Enter the number of sides with the keyboard.
4. Select a corner point with the mouse, or enter the X,Y offset from the center to a corner with the keyboard (press ENTER to use the offset from the last shape drawn).



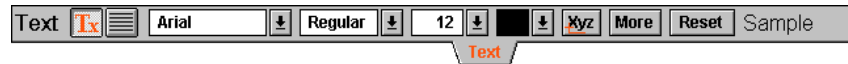
Draw a polygon

1. Select the **Shapes** tab on the toolbar and then select the  button, or select **Shapes/Polygon** from the **Draw** menu.
2. Select a point of the polygon with the mouse, or enter the X,Y coordinates of the point on the keyboard.
3. Select the next point of the polygon with the mouse, or enter the X,Y offset to the next point on the keyboard.
4. Repeat the previous step for each additional polygon segment.
5. Press the ENTER key after the last segment has been drawn.



Text

Text Toolbar functions



All of the following text toolbar functions may also be changed by selecting **Text/Text Options** from the **Draw** menu.


Text Font: Select the font drop down list to change the font. DeltaCad uses all of the True Type fonts currently installed on the Microsoft® Windows™ operating system.

Font Style: Select the font style drop down list to change the font style.



Text Height: Select the text height drop down list to change the text height. If the desired height is not in the list of choices, choose **other** at the top or bottom of the list, and then enter the correct height in the popup window.

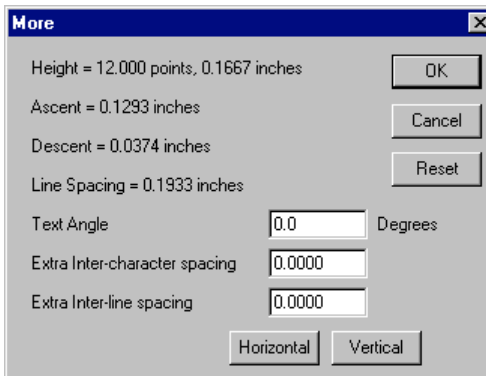



Color: Select the drop down list to change the text color.

Insertion Point and Justification: Select the  button to change the location of the text insertion point or the type of justification. The insertion point can be at the left, center, or right, and at the bottom, baseline, center, or top of the text. The justification can be left, center, right, or both.



Angle and Spacing: Select the  button to change the text angle, the character spacing, or the line spacing. If an angle or spacing is entered the  button will change to red to indicate that a non-standard angle or spacing has been entered. The color will change back to black when the angle and spacing are changed back to their default values.




Reset: Select the  button to change all of the text options back to their default values.


NOTE: The default values may be changed by choosing **Options/Text** from the DeltaCad **Options** menu.

Draw single line text

This function is for text which does not have a right margin, and will not automatically wrap to the next line.


1. Select the **Text** tab on the toolbar and then select the  button, or select **Text/Text** from the **Draw** menu.
2. Select the text insertion point with the mouse, or enter the X,Y coordinates on the keyboard.
3. Enter the text on the keyboard, or select an existing piece of text with the mouse (To start a new line of text, hold down the control key and press the ENTER key).

DeltaCad by Midnight Software, Inc.

 Select starting point, then enter text

Draw multiple line text

This function is for text which does have a right margin, and will automatically wrap to the next line.

1. Select the **Text** tab on the toolbar and then select the  button, or select **Text/Multi Line** from the **Draw** menu.
2. Select the text insertion point with the mouse, or enter the X,Y coordinates on the keyboard.
3. Enter the desired width to the right margin with the keyboard or the mouse.
4. Enter the text on the keyboard, or select an existing piece of text with the mouse (To start a new line of text, hold down the control key and press the ENTER key).

Now is the time for all good
men to come to the aid of their
country!!!


 Select starting point, enter width, then enter text.

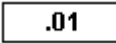

Dimension

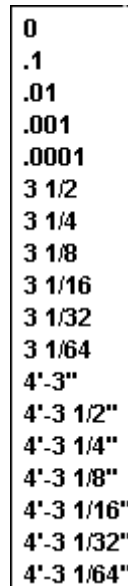
Dimension Toolbar functions

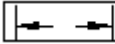



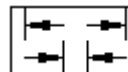
All of the following dimension toolbar functions may also be changed by selecting **Dimension/Dimension Options** from the **Draw** menu.

Color: Select the   drop down list to change the dimension color.

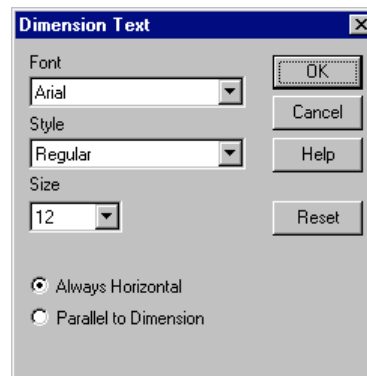
Precision type: Select the   drop down list to choose the type of precision to be used.




Arrow Direction: Select the   drop down list to choose whether the dimension arrows should point toward each other or away.

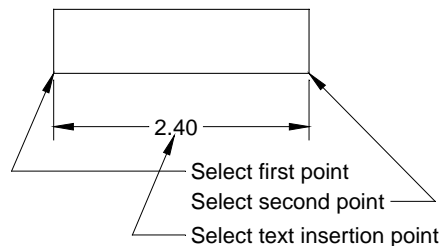


Dimension Font: Select the **Font** button to choose the dimension font to be used. If the font or size is changed to a value other than the default values the **Font** button will change to red to indicate that a non-standard font will be used. When the font is set back to the default value the button will change back to black.




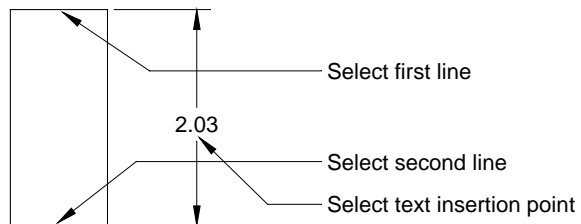
Draw a horizontal dimension

1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Horizontal** from the **Draw** menu.
2. Select a line or a point for the first extension line with the mouse, or enter the X,Y coordinates of a point on the keyboard.
3. Select a line or a point for the second extension line with the mouse, or enter the X,Y offset from the first point on the keyboard.
4. Select the text insertion point with the mouse, or enter the X,Y offset from the second point on the keyboard.




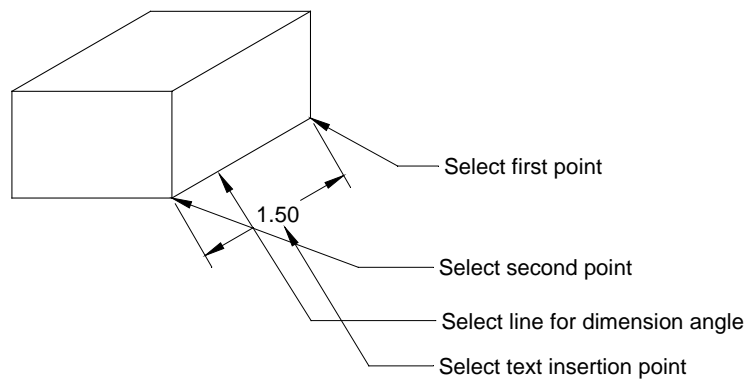
Draw a vertical dimension

1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Vertical** from the **Draw** menu.
2. Select a line or a point for the first extension line with the mouse, or enter the X,Y coordinates of a point on the keyboard.
3. Select a line or a point for the second extension line with the mouse, or enter the X,Y offset from the first point on the keyboard.
4. Select the text insertion point with the mouse, or enter the X,Y offset from the second point on the keyboard.




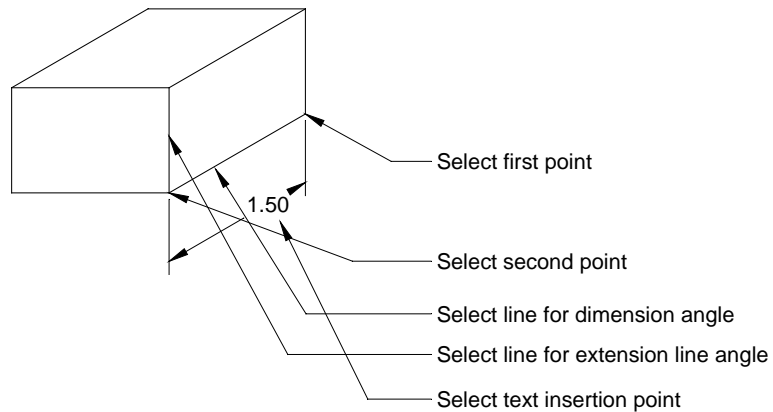
Draw an angled dimension

1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Angled** from the **Draw** menu.
2. Select a line or a point for the first extension line with the mouse, or enter the X,Y coordinates of a point on the keyboard.
3. Select a line or a point for the second extension line with the mouse, or enter the X,Y offset from the first point on the keyboard.
4. Enter the dimension angle by selecting an existing line with the mouse, or enter the angle on the keyboard.
5. Select the text insertion point with the mouse, or enter the X,Y offset from the second point on the keyboard.




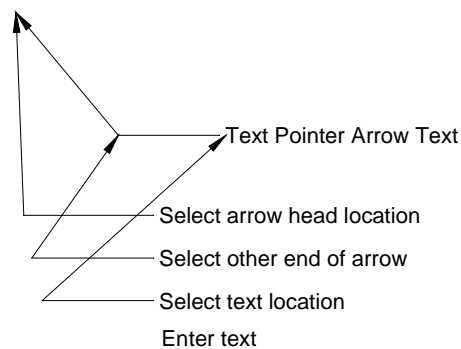
Draw an angled dimension with angled extension lines

1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Angled Angled** from the **Draw** menu.
2. Select a line or a point for the first extension line with the mouse, or enter the X,Y coordinates of a point on the keyboard.
3. Select a line or a point for the second extension line with the mouse, or enter the X,Y offset from the first point on the keyboard.
4. Enter the dimension angle by selecting an existing line with the mouse, or enter the angle on the keyboard.
5. Enter the extension line angle by selecting an existing line with the mouse, or enter the angle on the keyboard.
6. Select the text insertion point with the mouse, or enter the X,Y offset from the second point on the keyboard.




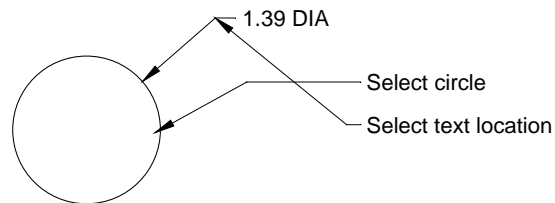
Draw a text pointer arrow

1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Arrow** from the **Draw** menu.
2. Select the arrow head location with the mouse, or enter the X,Y coordinates of the arrow head on the keyboard.
3. Select the other end of the arrow with the mouse, or enter the X,Y offset from the arrow head on the keyboard (press ENTER to use the offset from the last arrow drawn).
4. Select the text insertion point with the mouse, or enter the X,Y offset from the last point on the keyboard (press ENTER to use the offset from the last arrow drawn).
5. Enter the text on the keyboard.




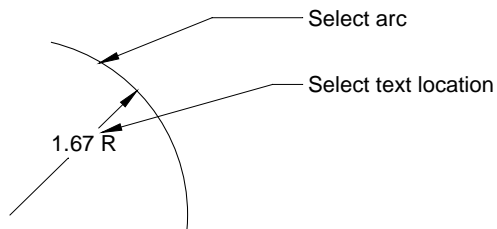
Draw a diameter dimension

1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Diameter** from the **Draw** menu.
2. Select a circle or arc with the mouse.
3. Select the text insertion point with the mouse, or enter the X,Y coordinates on the keyboard.




Draw a radius dimension

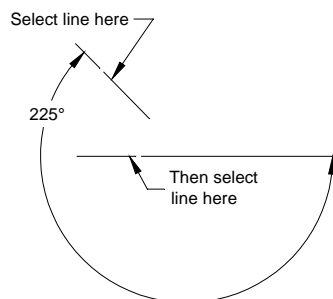
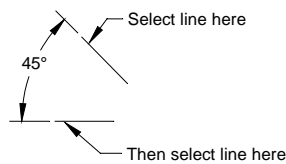
1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Radius** from the **Draw** menu.
2. Select a circle or arc with the mouse.
3. Select the text insertion point with the mouse, or enter the X,Y coordinates on the keyboard.




Draw an angle dimension

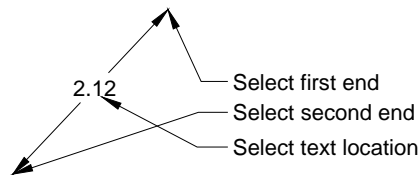
1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Angle** from the **Draw** menu.
2. Select a line for the angle dimension with the mouse. Angle dimensions are drawn in a counter-clockwise direction, so make sure the correct line is picked first. When a line is picked, the line can have two different angles (180 degrees off). The angle used will be the angle of the end of the line closest to the mouse pick location (only the displayed portion of the line is used in determining the angle). The two examples below show the mouse locations where the first and second lines were picked to create the angle dimensions shown.
3. Select the second line for the angle dimension as described above.
4. Select the text insertion point with the mouse, or enter the X,Y coordinates on the keyboard. If you accidentally started to draw the dimension in a clockwise direction you may press the ENTER key to swap the direction, and then select the text insertion point.

The following two examples illustrate how to select the angle for the bottom line.




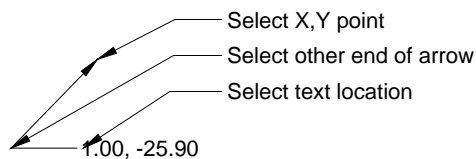
Draw a point-to-point dimension

1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/Point-to-Point** from the **Draw** menu.
2. Select the first endpoint with the mouse, or enter the X,Y coordinates on the keyboard.
3. Select the second endpoint with the mouse, or enter the X,Y offset on the keyboard.
4. Select the text location with the mouse, or enter the X,Y offset on the keyboard.



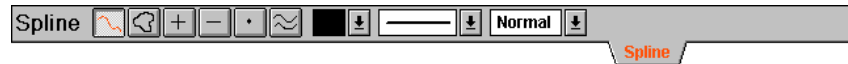
Draw an X,Y coordinate dimension

1. Select the **Dimension** tab on the toolbar and then select the  button, or select **Dimension/X,Y Coordinate** from the **Draw** menu.
2. Select the X,Y location with the mouse, or enter the X,Y coordinates on the keyboard.
3. Select the other end of the arrow with the mouse, or enter the X,Y offset from the X,Y location on the keyboard (press ENTER to use the offset from the last X,Y dimension).
4. Select the text insertion point with the mouse, or enter the X,Y offset from the last point on the keyboard (press ENTER to use the offset from the last X,Y dimension).





Spline

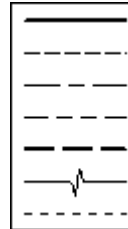
Spline Toolbar functions

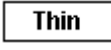



All of the following spline toolbar functions may also be changed by selecting **Spline/Spline Options** from the **Draw** menu.


Color: Select the   drop down list to change the spline color.

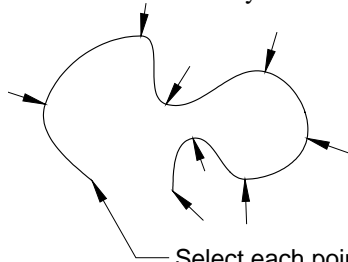
Line Type: Select the   drop down list to change the line type.



Line Weight: Select the   drop down list to change the line weight Thin, Normal, Thick, Heavy, or Wide.


Draw a spline

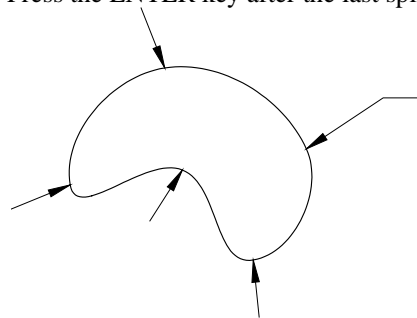
1. Select the **Spline** tab on the toolbar and then select the  button, or select **Spline/Spline** from the **Draw** menu.
2. Select the first endpoint of the spline with the mouse, or enter the X,Y coordinates of the endpoint on the keyboard.
3. Select the next point of the spline with the mouse, or enter the X,Y offset to the next point on the keyboard.
4. Repeat the previous step for each additional spline segment.
5. Press the ENTER key after the last spline segment has been drawn.



Select each point, then press ENTER to end


Draw a continuous spline

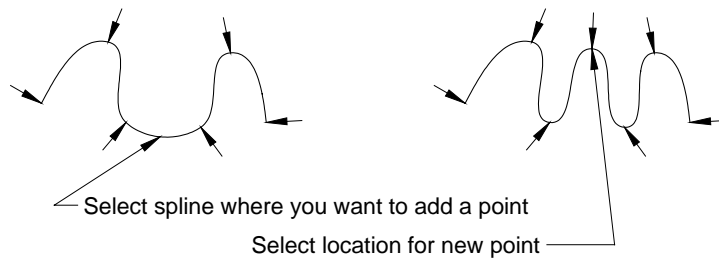
1. Select the **Spline** tab on the toolbar and then select the  button, or select **Spline/Continuous** from the **Draw** menu.
2. Select the first point of the spline with the mouse, or enter the X,Y coordinates of the first point on the keyboard.
3. Select the next point of the spline with the mouse, or enter the X,Y offset to the next point on the keyboard.
4. Repeat the previous step for each additional spline segment.
5. Press the ENTER key after the last spline segment has been drawn.




Select each point, then press ENTER to end

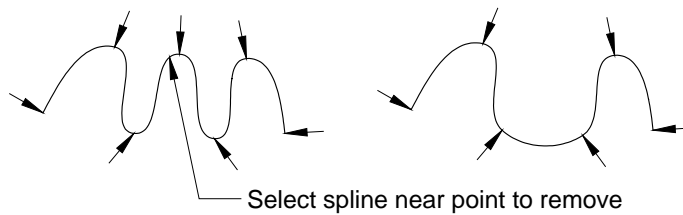
Add a point to a spline

1. Select the **Spline** tab on the toolbar and then select the  button, or select **Spline/Add Point** from the **Draw** menu.
2. Select an existing spline near where you want to add a new spline point.
3. Select the new spline point with the mouse, or enter the X,Y coordinates of the new point on the keyboard.




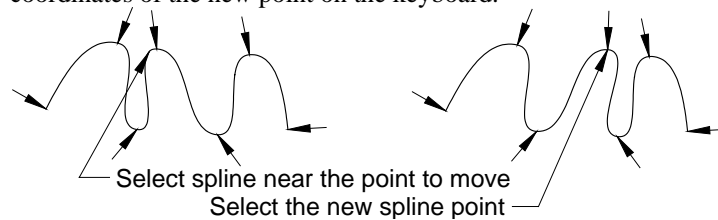
Remove a point from a spline

1. Select the **Spline** tab on the toolbar and then select the  button, or select **Spline/Remove Point** from the **Draw** menu.
2. Select an existing spline near the point you want to remove.




Move a spline point

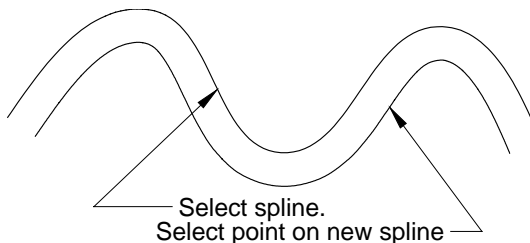
1. Select the **Spline** tab on the toolbar and then select the  button, or select **Spline/Move Point** from the **Draw** menu.
2. Select an existing spline near the point you want to move.
3. Select the new spline point with the mouse, or enter the X,Y coordinates of the new point on the keyboard.



Draw a parallel spline

1. Select the **Spline** tab on the toolbar and then select the  button, or select **Spline/Parallel** from the **Draw** menu.
2. Select an existing spline.
3. Locate the new parallel spline using one of the following:
 - Select the new location and click the mouse.
 - Move the mouse so the new spline is the correct direction away from the existing spline, and then enter the offset distance on the keyboard (or press ENTER to use the last parallel spline offset distance).

To create a spline that is parallel to the old spline will require that the new spline have a more points than the original spline.




Editing Individual Objects

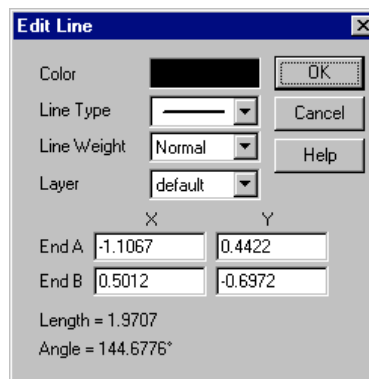
Undo

The Undo function is used to undo the last drawing operation. To undo the last operation simply select **Undo** from the **Edit** menu, or press control **Z**. The undo function is a toggle function. So if undo is performed a second time it will undo the first undo. Try creating an object, then press control **Z** several times and observe the effect.

Edit object data


This function is used to edit the data (color, line type, line weight, text, X,Y locations, etc.) associated with an object. This function is also useful just to view an object's data. This window also lists lengths, areas, and angles for some objects.

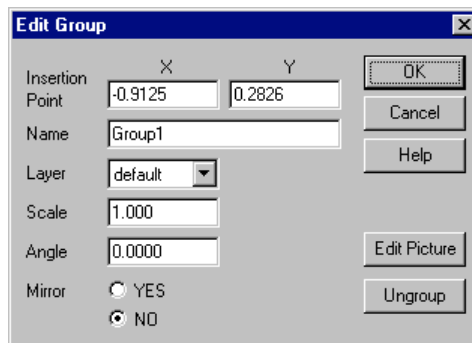
1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Edit Object** from the **Edit** menu.
2. Select the object to edit with the mouse.
3. Edit the appropriate data in the popup window, and then click the **OK** button.



Edit grouped objects

This function is used to edit a grouped object (Grouped objects are created by using the **Group Selected Objects** function).

1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Edit Object** from the **Edit** menu.
2. Select the grouped object to edit with the mouse.
3. Edit the appropriate data in the popup window, and then click the **OK** button, click the **Edit Picture** button, or click the **Ungroup** button.




When the **Edit Picture** button is clicked a new drawing window will appear which contains the group to be edited. Notice that the group name is part of the window title. Any changes made in the group window will immediately be shown on all windows that use the group. After completing all edits to the group, close the group window just like closing any other window.

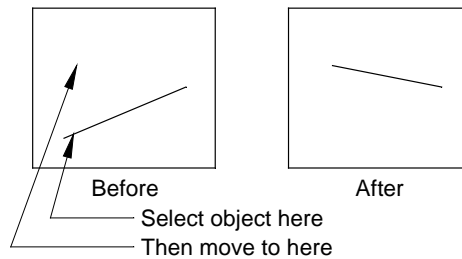
While editing a grouped object, DeltaCad will not allow the last object to be removed from a group. This is to prevent having an inserted group with no objects in it. To delete a group, simply erase all of the places the group is inserted, do not try to delete all objects from the group.

The **Ungroup** function is provided to ungroup objects that were previously grouped. To ungroup a grouped object, simply click the Ungroup button, and then click the YES button when prompted if you are sure you want to ungroup the object.

Move an object point


This function is used to move one of the points associated with an object (point, line, circle, etc.).

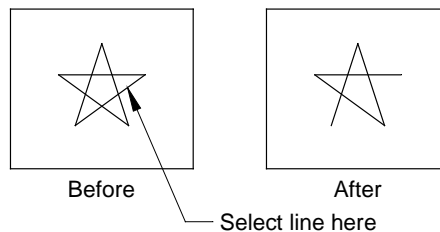
1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Move Endpoint** from the **Edit** menu.
2. Select the object that will have one of its points moved. If the object contains more than one point, the point closest to the mouse location when the object was picked is the one which will be used.
3. Select the new point location with the mouse, or enter the X,Y offset from the old location to the new location (press ENTER to use the offset from the last moved point).



Erase an object


This function is used to erase an object. If this function is used by mistake, simply select **Undo** from the **Edit** menu.

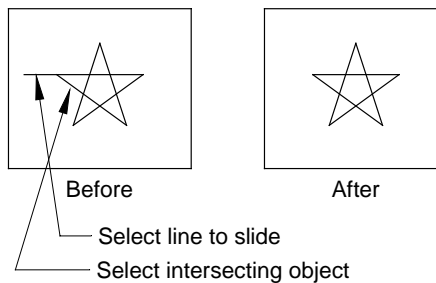
1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Erase Object** from the **Edit** menu.
2. Select the object to erase with the mouse.



Slide an endpoint


This function is used to slide the endpoint of a line or arc. If this function is used on a shape the shape will be converted into lines, and then the function will be applied.

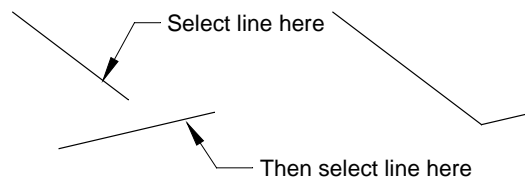
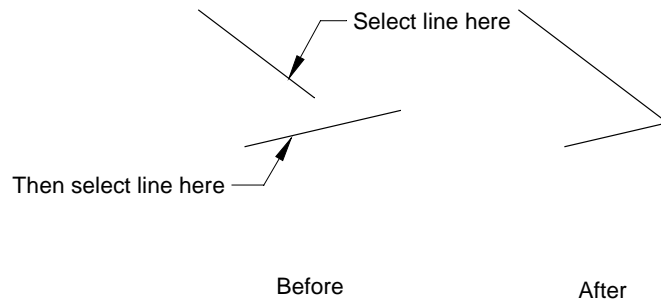
1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Slide Endpoint** from the **Edit** menu.
2. Select a line or arc with the mouse. The end of the object that will be moved is the end closest to the mouse when the object is selected.
3. Select a new endpoint using one of the following:
 - Select a new location with the mouse.
 - Select an intersecting object.
 - Enter the X,Y coordinates of the new location with the keyboard.
4. If the wrong end of the line was picked in step 2 above, you can press the ENTER key to swap ends.



Create a corner


This function is used to trim or extend two lines to make a corner. If this function is used on a shape the shape will be converted into lines, and then the function will be applied.

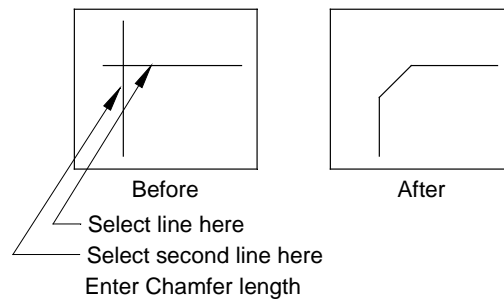
1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Create Corner** from the **Edit** menu.
2. Select the first line or arc with the mouse. The line or arc must be selected on the side of the intersection point that is to be retained. The intersection point is the point where the two lines or arcs will meet. See the example below.
3. Select the second line with the mouse.



Create a chamfer


This function is used to trim or extend two lines and add a new line to make a chamfer. If this function is used on a shape the shape will be converted into lines, and then the function will be applied.

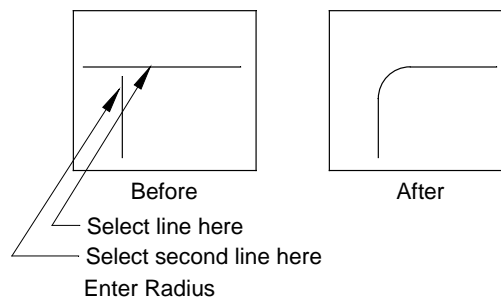
1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Create Chamfer** from the **Edit** menu.
2. Select the first line or arc with the mouse
3. Select the second line or arc with the mouse.
4. Enter the chamfer length on the keyboard, or press ENTER to use the last chamfer length.



Create a radius


This function is used to trim or extend two lines and add a new arc segment to make a radius. If this function is used on a shape the shape will be converted into lines, and then the function will be applied.

1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Create Radius** from the **Edit** menu.
2. Select the first line or arc with the mouse
3. Select the second line or arc with the mouse.
4. Enter the radius on the keyboard, or press ENTER to use the last radius.



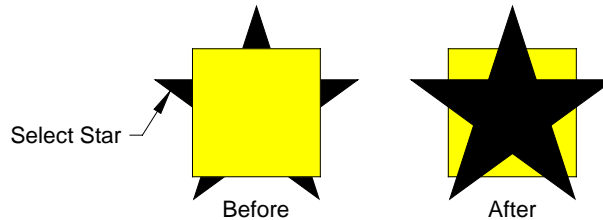
Split a line, circle, arc, ellipse, or elliptical arc


This function is used to split a line, circle, arc, ellipse, or elliptical arc into two new objects. If this function is used on a shape the shape will be converted into lines, and then the function will be applied.

1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Split a Line** from the **Edit** menu.
2. Select the object to split with the mouse. The object will be split at the mouse location where the object was selected.


Move to the top of the Z-Order

This function is used to move an object to the top of the Z-order. The Z-order is the order which objects are stacked on top of each other. In the left figure below the square is at the top of the Z-order and the star is at the bottom. In the right figure below the star is at the top of the Z-order and the square is at the bottom. This function is used mostly with overlapping filled objects.



1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Z-order Top** from the **Edit** menu.
2. Select the object to move to the top of the Z-order with the mouse.


Move to the bottom of the Z-Order

1. Select the **Edit** tab on the toolbar and then select the  button, or select **Edit/Z-order Bottom** from the **Edit** menu.
2. Select the object to move to the bottom of the Z-order with the mouse.

Editing Selected Objects


Selecting objects

This function is used to select objects. Selected objects are displayed in their highlight color until they are unselected. After objects are selected, they can be moved, copied, rotated, mirrored, etc.


1. Select the **Select** tab on the toolbar and then select the  button, or select **Select/Select Object** from the **Edit** menu.
2. Select objects using one of the following methods:
 - Move the mouse over the object, and then click the mouse button.
 - Select objects within a box by clicking the mouse button at the two box corners. All objects within the box will be selected.
 - Pick **Select All** from the **Edit** menu to select all drawing objects.
3. To select, or unselect, additional objects hold down the control key and repeat the above step. This will toggle the selected objects. Objects that were selected will become unselected, and unselected objects will become selected.
4. To unselect all objects press the Escape key, pick **Unselect All** from the **Edit** menu, or select any of the following tabs on the toolbar:
 - Edit, Point, Line, Circle, Shapes, Text, Dimension, Spline, Macro.


Set the Base Point for selected objects

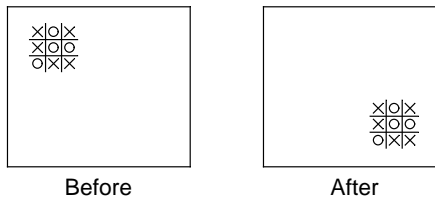
The basepoint for selected objects is used as a reference point for moving, rotating, scaling, etc. Unless the base point is set with this function, the base point will be set to the center of the selected objects.

1. Select the desired objects.
2. Select the  button on the **Select** toolbar, or select **Select/Set Base Point** from the **Edit** menu.
3. Select the base point with the mouse, or enter the X,Y coordinates of the base point with the keyboard.

Move selected objects

This function is used to move or copy selected objects. To move objects from one DeltaCad drawing window to another select the objects in one drawing window, switch to another drawing window, and then use the move  function.


1. Select the desired objects, and set the base point if needed.
2. Select the  button on the **Select** toolbar, or select **Select/Move Objects** from the **Edit** menu.
3. If you want to make a copy of the data before it is moved, pick the **Copy** button on the toolbar.
4. Select the new location with the mouse, or enter the X,Y offset from the old location to the new location with the keyboard (press ENTER to use the last offset). As the mouse is moved, an outline box will be displayed to show the new location.
5. To move the selected objects the same amount again, press the ENTER key. This step may be repeated.
6. You can also use the arrow keys (up, down, right, & left) to move the selected objects one pixel at a time.

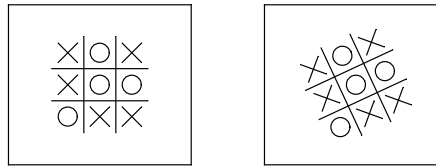


Since the move function can be repeated simply by pressing the ENTER key, it is sometimes useful to type in a very small offset, and then just keep pressing the ENTER key to move the objects to their final position.

Rotate selected objects

This function is used to rotate selected objects around the Base Point.

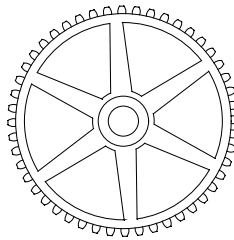
1. Select the desired objects, and set the base point if needed.
2. Select the  button on the **Select** toolbar, or select **Select/Rotate Objects** from the **Edit** menu.
3. If you want to make a copy of the data before it is rotated, pick the **Copy** button on the toolbar.
4. Enter the rotation angle (counter-clockwise is positive) with the keyboard, or press ENTER to use the last rotation angle.
5. To rotate the selected objects the same amount again, press the ENTER key. This step may be repeated.



Before

After


Rotate 25°

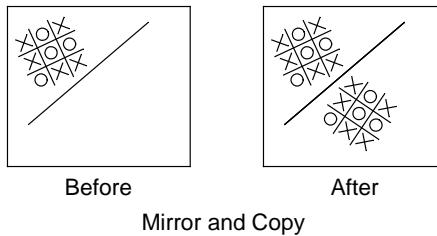


Rotate and Copy 7.5 degrees
to make each gear tooth.

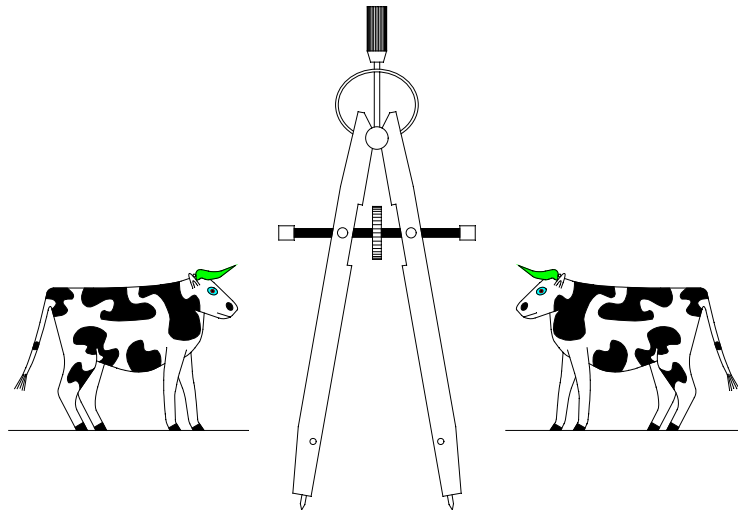
Mirror selected objects

This function is used to create a mirror image of all selected objects. You may need to create a mirror line before starting this function.

1. Select the desired objects.
2. Select the  button on the **Select** toolbar, or select **Select/Mirror Objects** from the **Edit** menu.
3. If you want to make a copy of the data before it is mirrored, pick the **Copy** button on the toolbar.
4. Select a mirror line with the mouse.



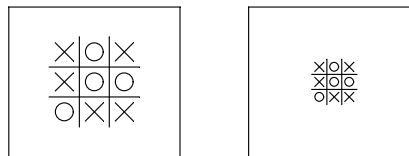
The mirror function is useful for creating symmetric or reversed objects.



Scale selected objects

This function is used to scale all selected objects around the Base Point.

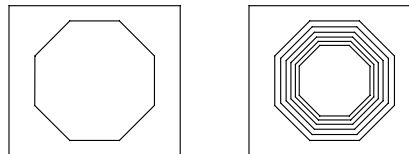
1. Select the desired objects, and set the base point if needed.
2. Select the **Sc** button on the **Select** toolbar, or select **Select/Scale Objects** from the **Edit** menu.
3. If you want to make a copy of the data before it is scaled, pick the **Copy** button on the toolbar.
4. Enter the scale factor on the keyboard, or press ENTER to use the last scale factor.
5. To scale the selected objects the same amount again, press the ENTER key. This step may be repeated.



Before

After

Scale 0.5




Before

After

Scale 0.9 & copy, five times


Group selected objects

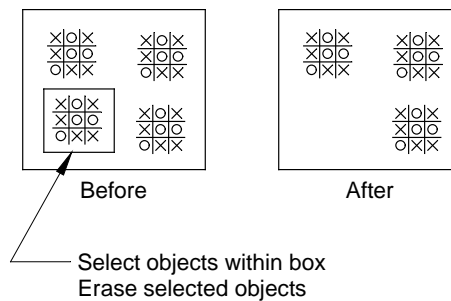
This function is used to group selected objects. Once a group of objects is grouped into a single grouped object, the object may be moved, copied, mirrored, scaled, etc. just like a single object. Grouped objects can also save memory. If a grouped object is copied many times in a drawing, DeltaCad only needs to keep one copy of the objects which make up the group.

1. Select the desired objects to group together, and set the base point if needed.
2. Select the  button on the **Select** toolbar, or select **Select/Group Objects** from the **Edit** menu.
3. Enter a name for the group, or press ENTER to use the suggested name.

Erase selected objects


This function is used to erase all selected objects. If this function is used by mistake, simply select **Undo** from the **Edit** menu.

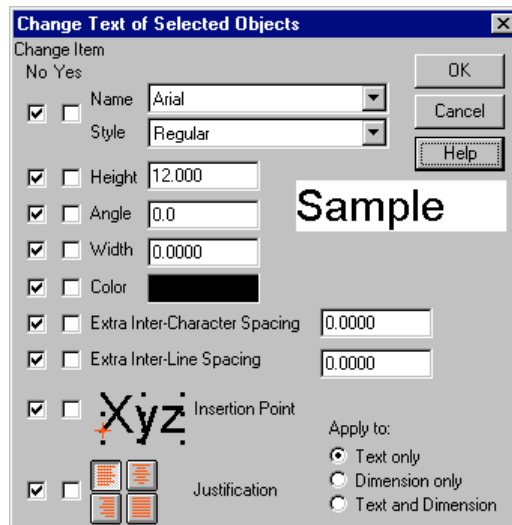
1. Select the desired objects.
2. Select the  button on the **Select** toolbar, or select **Select/Erase Objects** from the **Edit** menu.
3. Press the ENTER key to erase all selected objects.



Change selected text


This function is used to change the properties (color, size, justification, alignment, etc.) of all selected text. Any non-text objects which are selected will be ignored.

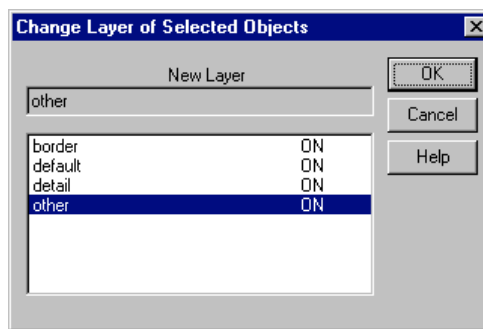
1. Select the desired objects.
2. Select the  button on the **Select** toolbar, or select **Select/Edit Text** from the **Edit** menu.
3. Change the appropriate data in the popup window, and then press the **OK** button with the mouse.



Change layer of selected objects

This function is used to change the layer of all selected objects.

1. Select the desired objects.
2. Select the  button on the **Select** toolbar, or select **Select/Change Layer** from the **Edit** menu.
3. Pick one of the layers in the list of existing layers, and then press the **OK** button with the mouse.



Change color of selected objects

This function is used to change the color of all selected objects.

NOTE: The new color will not appear on the screen until after the objects are unselected.

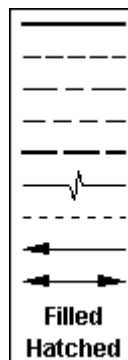
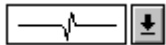
1. Select the desired objects.
2. Select a new color with the color drop-down list on the toolbar



Change line type/cross-hatch pattern of selected objects

This function is used to change the line type or cross-hatch pattern of all selected objects. If some of the selected objects do not have a line type (such as text) they will be ignored.



1. Select the desired objects.
2. Select a new line type with the line type drop-down list on the toolbar



Change line weight of selected objects


This function is used to change the line weight of all selected objects. If some of the selected objects do not have a line weight (such as text) they will be ignored.

1. Select the desired objects.
2. Select a new line weight with the line weight drop-down list on the

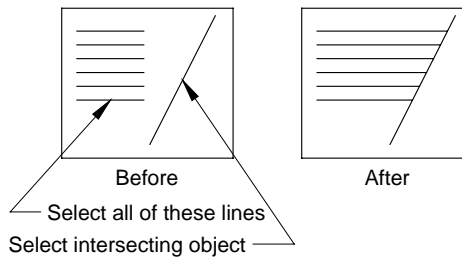
toolbar   and then choose **Thin**, **Normal**, **Thick**, **Heavy**, or **Wide**.

Slide endpoints of selected lines

This function is used slide the endpoints of all selected lines.

1. Select the desired lines.
2. Select the  button on the **Select** toolbar, or select **Select/Slide Endpoints of Lines** from the **Edit** menu.
3. Select a new endpoint using one of the following:
 - Select a new location with the mouse.
 - Select an intersecting object.
 - Enter the X,Y coordinates of the new location with the keyboard.

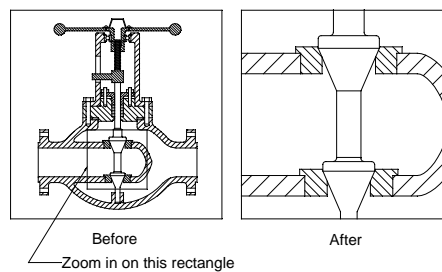
The end moved will be the end closest to the new endpoint.




Viewing the Drawing

View objects within a rectangle


This function is used to zoom in on a rectangular section of a drawing. The following example shows a drawing before and after it is zoomed in on.



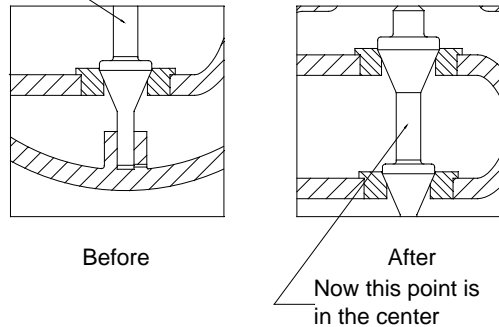
1. Select the **View** tab on the toolbar and then select the  button, or select **Box** from the **View** menu.
2. Select the first corner of the rectangle with the mouse, or enter the X,Y coordinates on the keyboard.
3. Select the other corner of the rectangle with the mouse, or enter the X,Y offset from the first corner on the keyboard.

Center view around a point


This function is used to center the displayed image around a point.

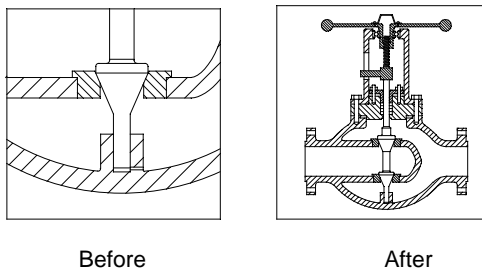
1. Select the **View** tab on the toolbar and then select the  button, or select **Point** from the **View** menu.
2. Select the new center point with the mouse, or enter the new center points X,Y coordinates on the keyboard.

Select here to center




View all

The View All function is used to view all of the objects in a drawing. To view all objects select the **View** tab on the toolbar and then select the  button, select **All** from the **View** menu, or hold down the control key and press the **A** key.




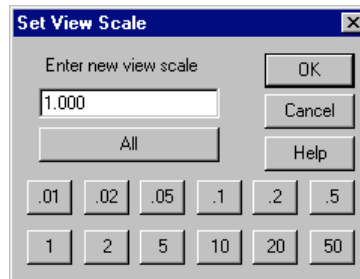
Previous views

When ever DeltaCad changes the current view of a drawing (by scaling, moving up, down, left, or right, or by restoring a bookmark) it keeps track of the last five views. To go back to the previous view of a drawing simply click the  button on the **View** toolbar, select **Previous** from the **View** menu, or press control **B** on the keyboard. This step may be repeated to go back further. If you try to go back further than five views, DeltaCad will start over with the view you started with. This function is convenient if you need to zoom in to look at something, and then you want to go back to where you were.

Change view scale



This function is used to change the View Scale to one of the pre-defined values, or any entered value.

1. Select the **View** tab on the toolbar and then select the  button, select the scale box on the status bar, or select **Scale** from the **View** menu.
2. When the popup window appears select one of the standard scale buttons, or enter the desired scale and click the **OK** button with the mouse.





Zoom-In

This function is used to zoom in on the drawing. This function decreases the view scale by a factor of 1.414.

Select the **View** tab on the toolbar and then select the  button, select the  button on the status bar, or select **Zoom In** from the **View** menu.


Zoom-Out

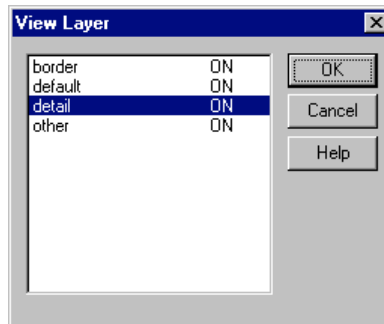
This function is used to zoom out on the drawing. This function increases the view scale by a factor of 1.414.

Select the **View** tab on the toolbar and then select the  button, select the  button on the status bar, or select **Zoom Out** from the **View** menu.

View a single layer

This function is used to temporarily view the objects on a single layer.

1. Select the **View** tab on the toolbar and then select the  button, or select **View Layer** from the **View** menu.
2. Pick the layer to view with the mouse, and then click the **OK** button.
3. To end this function press the Escape key, or select another drawing function from the toolbar.



View Toolbar Tab

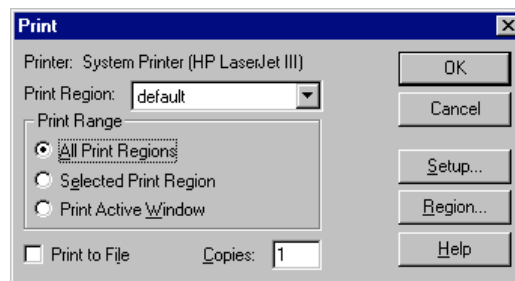


The **View** tab on the toolbar is used to move around in DeltaCad. The View tab is different from all of the other toolbar tabs. If any of the other tabs are picked, the current function in progress ends. The view tab is designed so the current function does not end when it is picked. For example, assume that you are drawing a rectangle. After you enter the first corner of the rectangle you may select the view tab to change the scale, and then return to the Shapes tab to continue drawing the rectangle. This allows you to perform View tab functions in between steps of the various drawing functions.

Printing

Print

To print the active drawing select **Print** from the **File** menu, and then perform the following steps.

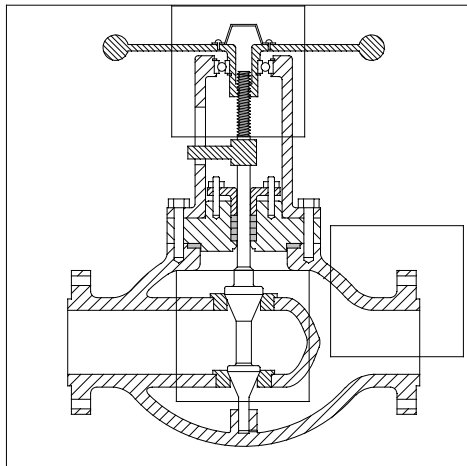


1. Select what to print.
 - **All Print Regions** will print all currently defined print regions (print regions are described below).
 - **Selected Print Region** will print the print region selected in the **Print Region** drop down list.
 - **Print Active Window** will print the active window as currently displayed on the computer screen.
2. Select the number of copies, and enter it in the **Copies** box.
3. If a different printer or orientation is desired, select the **Setup** button, pick a printer, and pick portrait or landscape printing orientation.
4. Select the **Region** button if the current print region needs to be changed, or if a new print region is needed.
5. Select the **OK** button to print the drawing.

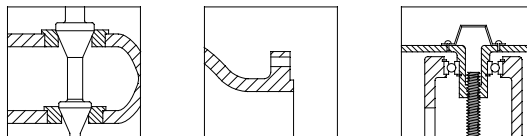
Setting the Print Region(s)

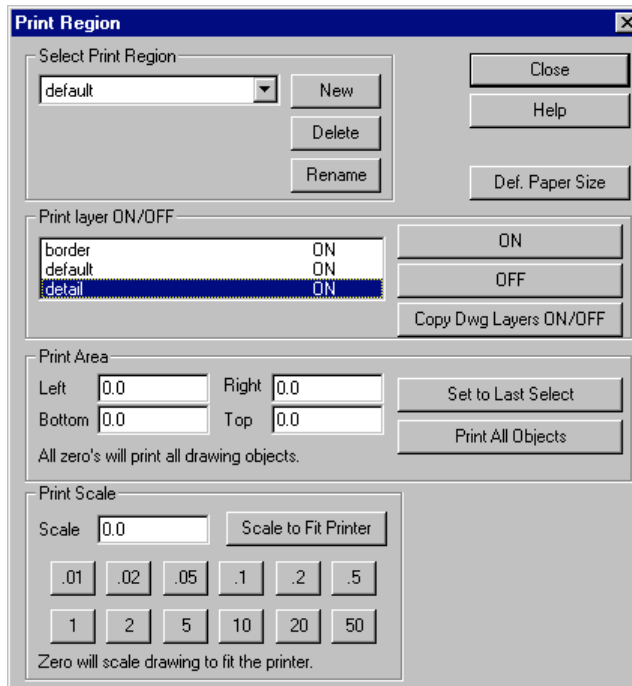
Print regions are used in DeltaCad to define a rectangular area of a drawing to be printed. A print region consists of a name, rectangle, scale, and list of layers to print. A drawing may have any number of print regions. This is convenient when working with a large drawing, and you only want to print a few small areas. When ever a new drawing is created a **default** print region is created. The **default** print region will print all layers, and the print will be scaled so all drawing objects will fit on the printer paper.

Prints will always be centered on the printer paper. If the drawing is printed at a specific scale, and the drawing will not fit on the printer paper, DeltaCad will make as many prints as necessary to print the entire drawing. You may use the Print Preview function to see how many sheets of paper will be required to print the drawing.



The drawing of the valve, shown above, contains three rectangles. If these rectangles represent three print regions, the three drawings shown below will be printed.





Adding, Deleting, or Renaming a print region

Select **Set Print Region** from the **File** menu, and then pick the **New**, **Delete**, or **Rename** button to perform the desired action. DeltaCad will not let you delete the last print region (i.e. every drawing must have at least one print region).

For all of the following print region functions you should first select which print region to change by selecting a print region in the drop-down list in the Select Print Region section of the popup window.

Selecting which layer to print

Select **Set Print Region** from the **File** menu. Ensure the correct print region is displayed in the **Select Print Region** drop down list. Then select which layers should be printed. To print the objects on a layer, select the layer in the **Print layer ON/OFF** list and then pick the **ON** button. To not print a layer, select the layer, and then pick the **OFF** button. To change all of the print layer ON/OFF's to match the current drawing layer ON/OFF's click the **Copy Dwg Layer ON/OFF** button.

Setting the Print Area

Enter the Left, Right, Top, and Bottom drawing coordinates of the area to be printed using one of the following methods.

1. Enter the Left, Right, Top, and Bottom coordinates in the appropriate boxes.
2. Pick the **Set to Last Select** button to use the last selected rectangle. This is the simplest way to define a print area. This function requires that you use the **Select** function to select the area to be printed prior to entering this function. Then when the **Set to Last Select** button is picked, the Left, Right, Top, and Bottom boxes will change to the coordinates of the last selected rectangle.
3. Pick the **Print All Objects** button. This will change the Left, Right, Top, and Bottom boxes to 0.0. If all of the boxes contain zeros, DeltaCad will set the print area to include all drawing objects.

Setting the Print Scale

This function is used to set the print scale. Enter the scale to be used for printing, pick the button for one of the pre-defined scales, or pick the **Scale to Fit Printer** button to have DeltaCad scale the print to fit the printer paper. When the **Scale to Fit Printer** button is picked, the scale box changes to 0.0.

Viewing the default printer paper size

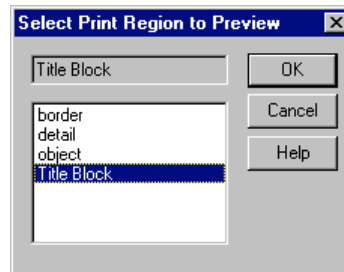
This function is used to display the paper size of the default printer. Pick the **Def. Paper Size** button to display the paper size of the default printer.

Printer Setup

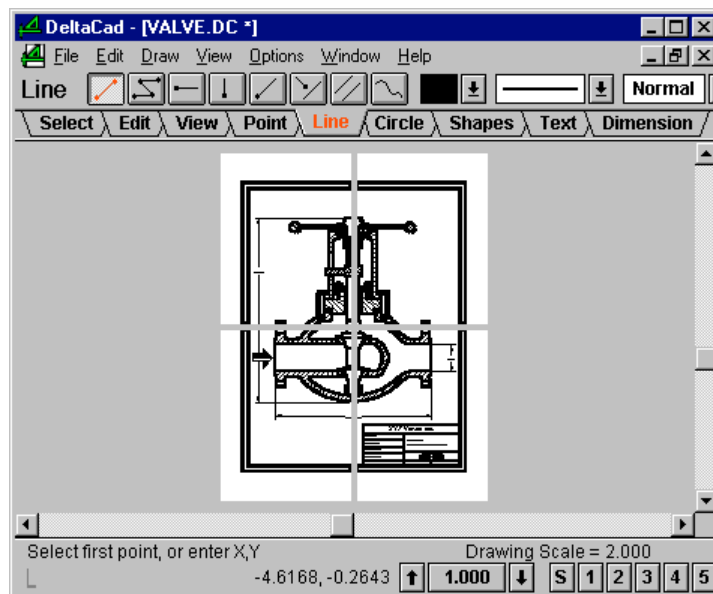
The Printer Setup function is the standard Windows™ printer setup popup window. This window allows you to select a printer from the installed printers or select printing in portrait or landscape. This function is available by selecting **Print** from the **File** menu and then selecting the **Setup** button, or by selecting **Printer Setup** from the **File** menu.

Print Preview

The print preview function is used to show how a drawing will appear when printed using the default printer. If the drawing only has one print region defined, it will be used. Otherwise a popup window will prompt you to select which print region to preview.



This function is useful in showing how a large drawing will be split up and printed on several sheets of paper.



Offline Printing

There are times when you may want to print your DeltaCad drawing from a computer which does not have DeltaCad installed. Such as printing at a blueprint shop. There are two ways to send your drawing to another computer for printing.

The first method requires that the printing computer have a program that can read and print .DXF files. You just save your drawing as a .DXF file, take it to the other computer, open it in the other program, and print the drawing. Most blueprint shops can read (and print) .DXF files. This method can cause small changes in the printed output due to limitations in the .DXF file format. In most cases this will not be a problem.

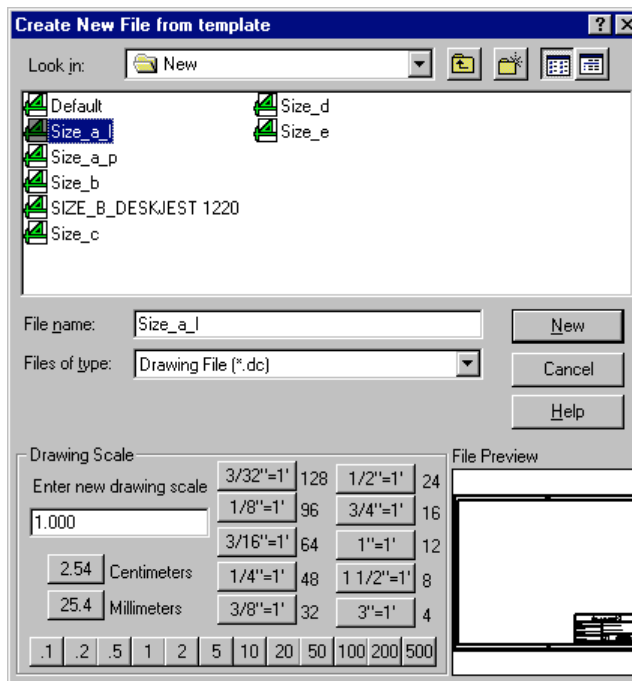
The second way is to find out what kind of printer your blueprint shop has. Then install the printer driver for their printer on your computer, except have it print to a FILE instead of the printer port (LPT1:). Then print your drawing to the new printer driver, take the print file to the blueprint shop, and copy it to their printer.

Misc.

File Operations

New

The New function is used to create a new DeltaCad drawing. When **New** is picked from the **File** menu a window will pop up which prompts you to pick a template drawing to base the new drawing on, and to enter the Drawing Scale for the new drawing.



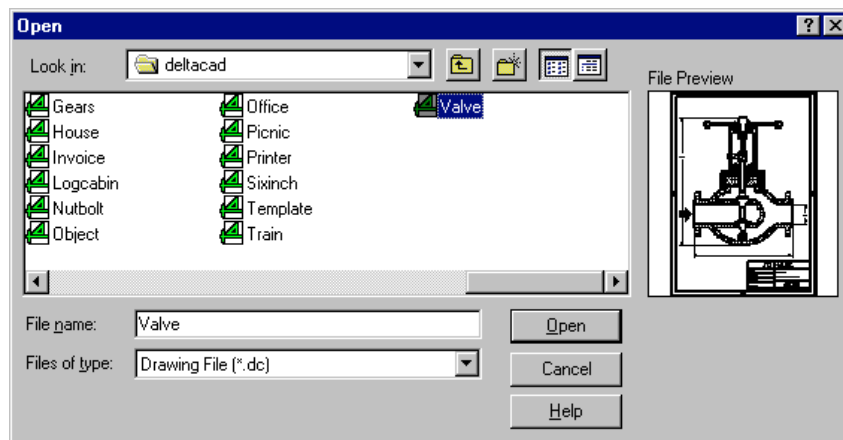
After the template drawing is picked a new drawing window will appear on the screen. There are a maximum of 9 drawing windows which may be open at once. Once the new window is created, it will have a window title such as **Sheet1**. The window title will change to the filename after the drawing has been saved.

To create a new template, simply create a DeltaCad drawing and save it in the DeltaCad **New** directory (C:\Program Files\DeltaCad\New).

When DeltaCad is started the **DEFAULT.DC** drawing in the **New** directory is used as a template. So if you would like to change the way DeltaCad starts, change the **DEFAULT.DC** drawing.

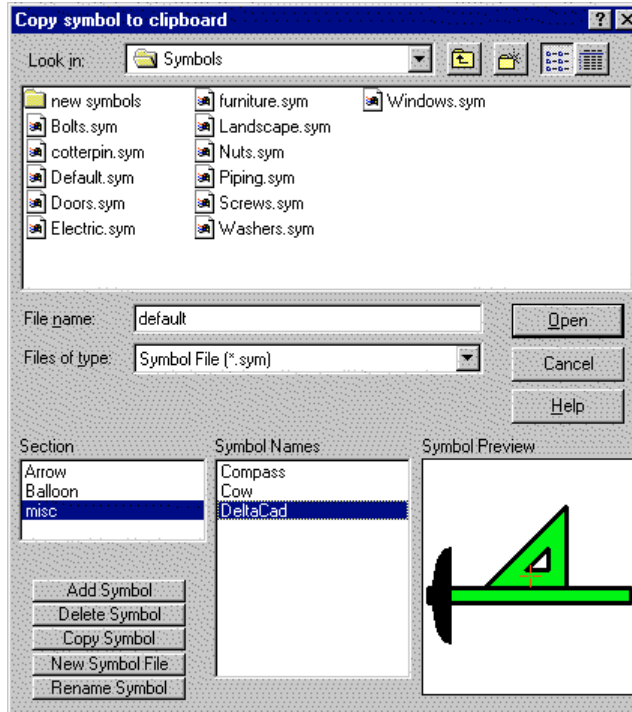
Open

The Open function is used to open an existing DeltaCad drawing, or .DXF file. When **Open** is picked from the **File** menu a window will pop up which allows you to select an existing DeltaCad drawing. If the file extension of the file name picked is .DXF, then the file will be read in as a .DXF file. All other file extension will be assumed to be DeltaCad files.



Symbol Library

The Symbol Library function is used to copy one of the DeltaCad symbols into a drawing. This function copies the selected symbol onto the clipboard, when OK is picked, and then prompts the user to paste the symbol into the drawing.



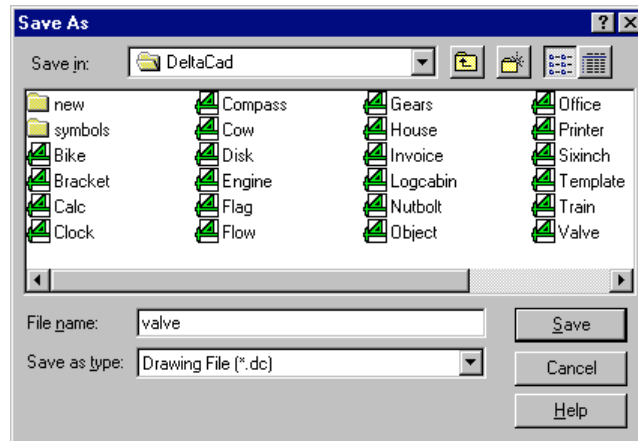
To add a new symbol to a .SYM file copy the new symbol onto the clipboard (select symbol and pick Copy from the Edit menu), pick the symbol file you want, and pick **Add Symbol**, then enter a section and symbol name. To Delete a symbol from a .SYM file pick the file, section, and symbol name, and then pick the **Delete Symbol** button. To copy a symbol to the clipboard pick the file, section, and symbol name, then pick the **Copy Symbol** button. The copy symbol function is useful when move symbols from one file to another. To move a symbol simply pick the symbol, copy it to the clipboard, delete the original symbol, and add the new symbol from the clipboard. To create a new symbol file copy the new symbol onto the clipboard, pick the **New Symbol File** button, and enter the file name, section name, and symbol name.

Save

The Save function is used to save the active drawing in DeltaCad. To save the drawing simply select **Save** from the **File** menu. If the drawing has never been saved, the **SaveAs** (described below) window will appear.

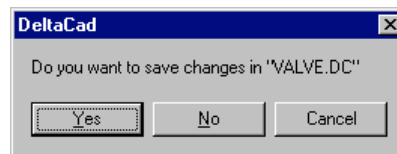
SaveAs

The SaveAs function is used to save a DeltaCad drawing for the first time, or to save a drawing with a different name. When **SaveAs** is picked from the **File** menu a window will appear which allows you to pick a file name to save the drawing as. If the file extension is .DXF then the file will be saved as a .DXF file.



Close

The Close function is used to exit the currently active DeltaCad drawing. If the drawing has not been saved since the last change was made a window will pop up asking you whether or not to save the drawing.

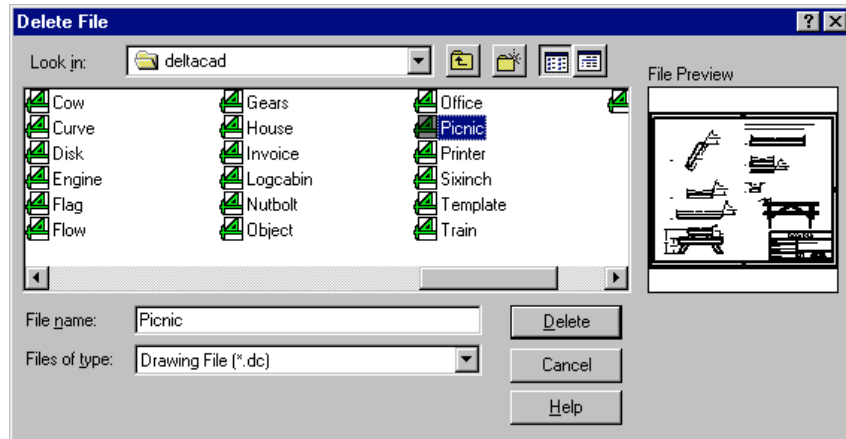


Most Recently Used Files

At the bottom of the **File** menu DeltaCad maintains a list of the four most recently used drawing files. To open one of the most recently used files simply pick it from the **File** menu.

Delete File

The Delete File function is used to delete a file. When **Delete File** is picked from the **File** menu a window will appear which allows you to pick a file to delete.



Import File

The Import File function is used to import a file. To import a bitmap file (.BMP) pick **Import/Bitmap File** from the **File** menu, browse to the file you want, and pick OK. This will copy the bitmap to the clipboard, and move you to the **Paste data from clipboard** toolbar button on the **Select** toolbar tab. Then just click the mouse at the location where you want to paste the picture into your drawing.

Export Screen to File

The Export function exports what is displayed on your screen to a file. To export your screen to a bitmap (.BMP) file) pick **Export/Bitmap File** from the **File** menu, enter a filename, and pick OK. You may also enter the resolution (dots-per-inch) to be used. Enter zero to use the screen resolution.

Clipboard

The following functions are used to cut and paste data to and from the clipboard. The clipboard is a feature of Microsoft® Windows™ which provides an easy way to exchange information between different programs. You should be careful when using Cut and Copy because they both erase any data which was already on the Clipboard.

Copy

The copy function is used to transfer selected data to the clipboard. The data is added to the clipboard in several different formats (Native DeltaCad format, Windows™ Metafile, Bitmap, and Text). The Text format is added to the clipboard only if a single text or dimension object is selected.

1. Select the desired data as described earlier.
2. Select **Copy** from the **Edit** menu.


Cut

The Cut function is the same as the Copy function, except the data is also deleted from the drawing.

1. Select the desired data as described earlier.
2. Select **Cut** from the **Edit** menu.

Paste

The paste function is used to transfer data from the clipboard to the DeltaCad drawing. The paste function requires that the Clipboard data be either in a Bitmap format, Text format, or in the Native DeltaCad data format. This function is useful when transferring text or pictures from another program to DeltaCad.

1. Select the **Select** tab on the toolbar and then select the  button, or select **Paste** from the **Edit** menu.
2. Select the insertion point with the mouse, or enter the X,Y coordinates of the insertion point with the keyboard. As the mouse is moved around an outline rectangle is shown on the screen. You may scroll the drawing or change the scale before pasting the data.
3. This function may be repeated. Select another insertion point, or press the escape key (or select any drawing function) to end.

Paste Special

The paste special function is the same as the paste function except you are given a choice of which clipboard data format to use, where the paste function uses the default format.



Copy a Figure to a Word Processor

The most frequent use of the Clipboard is to copy a DeltaCad drawing to a word processing document. To copy a figure to a word processing document, select the desired object in DeltaCad using the Select function, then pick **Copy** from the **Edit** menu of DeltaCad. Then switch to the word processor and pick **Paste** or **Paste Special** from the **Edit** menu of the word processor.

Copy Text to a Word Processor from DeltaCad

To copy text from DeltaCad to a word processor you should select the text to copy, then pick **Copy** from the **Edit** menu to add the text to the Clipboard. Then switch to the word processor and select **Paste** from the **Edit** menu to add the text to the document. The text will only copy to the Clipboard (in a Text format) if it is the only object selected, so it is important to only select one object (the text).

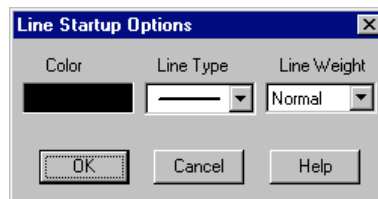
Copy Text to DeltaCad from a Word Processor

To copy text from a word processor to DeltaCad select the text in the word processor and pick **Copy** from the **Edit** menu of the word processor. Then pick **Paste** from the DeltaCad **Edit** menu and select the text insertion point. The text will be inserted using the font from the **Text** toolbar.

Options

Point, Line, Circle, Shapes, Text, Dimension, Spline

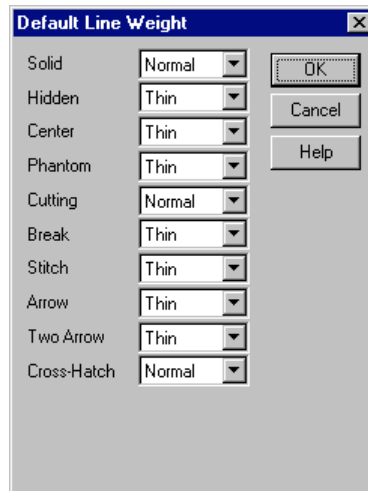
1. To change the default startup options for any of the drawing objects, select one of the above drawing objects from the **Options** section of the **Options** menu.
2. When the popup window appears change the desired option, and then click the **OK** button with the mouse.



Default Line Weight

The default line weight is the line weight selected when a new line type is selected.

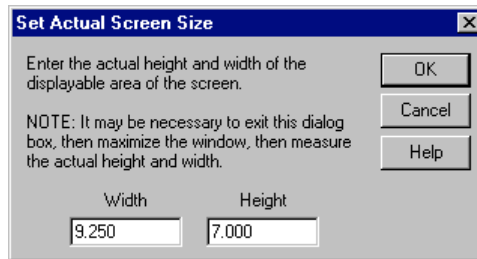
1. To change the default line weight select **Options/Default Line Weight** from the **Options** menu.
2. When the popup window appears select Thin, Normal, or Thick for each line type, and then click the **OK** button with the mouse.



Screen Size

This function is used to set the screen size. This will require you to maximize the DeltaCad Window, and then use a ruler to measure the exact size of the image on your computer monitor. This function is used so the images on the screen will appear the correct size and shape. If you are using the NTFS file system (on Windows NT or 2000) you will need to be logged into an account with “Administrator” privileges.

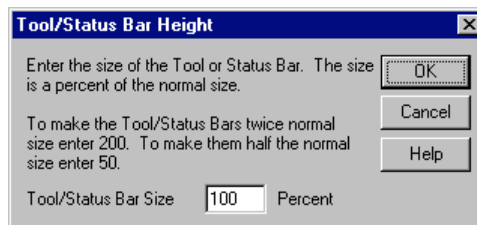
1. Select **Options/Screen Size** from the **Options** menu.
2. Enter the actual screen size and then click **OK** with the mouse. **NOTE:** This is the entire area of your monitor that can emit light, not just the window size.
3. Exit DeltaCad and then restart DeltaCad for the new options to take effect.



Tool and Status Bar

This function is used to change the height of the Tool bar and Status bar. This is useful on computers with very large monitors.

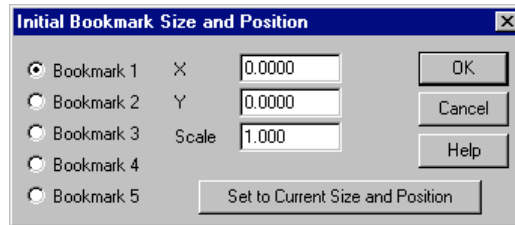
1. Select **Options/Tool/Status Bar Size** from the **Options** menu.
2. Enter the percent of normal size (i.e. 75 will make the Tool and Status bars 3/4 of their original size).
3. Click **OK** with the mouse.



Bookmarks

This function is provided so you can specify the initial bookmark scale and location to be used when DeltaCad is started.

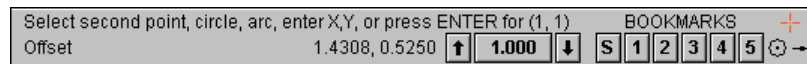
1. Select **Options/Bookmarks** from the **Options** menu.
2. Select Bookmark (1 thru 5) with the mouse.
3. Enter the scale and position.
4. Select another bookmark, or click **OK** with the mouse.



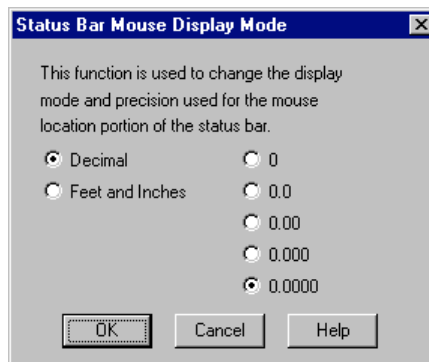
The **Set to Current Size and Position** button may be used to set the bookmark to the size and position of the currently active DeltaCad drawing.

Mouse Location Display Mode

This function is used to set the display mode to be used to display the mouse location on the status bar. You can change whether to display distances in the Feet/Inch mode or the Decimal mode, and you can change the number of decimal places displayed.



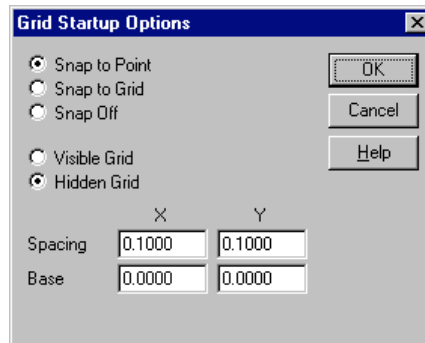
Mouse Location



Grid

This function is used to set the initial grid settings used when DeltaCad is started.

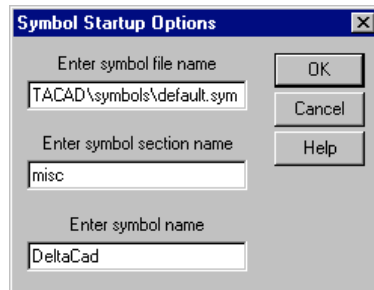
1. Select **Options/Grid** from the **Options** menu.
2. Pick one of the three Snap modes.
3. Set the initial grid spacing and base.
4. Click **OK** with the mouse.



Default Symbol

This function is used to set the initial symbol which is displayed when **Symbols** is picked from the **File** menu.

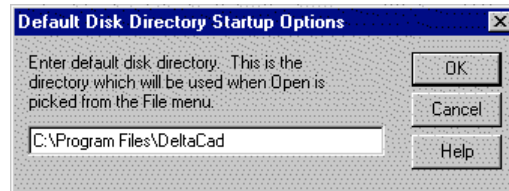
1. Select **Options/Default Symbol** from the **Options** menu.
2. Enter the symbol file name, section name, and symbol name.
3. Click **OK** with the mouse.



Default Directory

The default disk directory is the disk directory that is used when DeltaCad is started from the Start menu. It is the directory that is displayed when **Open** is picked from the DeltaCad **File** menu.

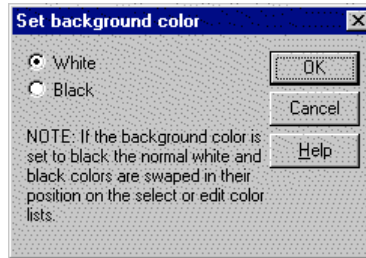
1. Select **Options/Default Directory** from the **Options** menu.
2. Enter the new default directory name.
3. Click **OK** with the mouse.



Background Color

This function is used to set the background color to either black or white.

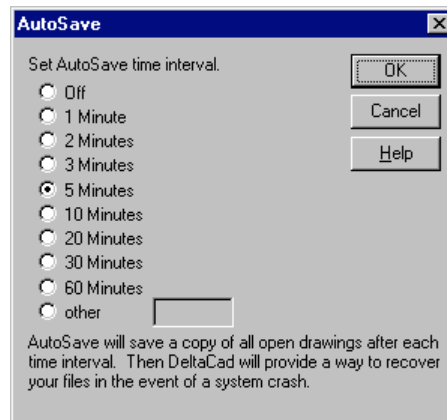
1. Select **Options/Background Color** from the **Options** menu.
2. Pick either black or white.
3. Click **OK** with the mouse.



AutoSave

Every few minutes DeltaCad saves your drawing, so it can be restored in case of a system crash. You can set the time interval between saves.

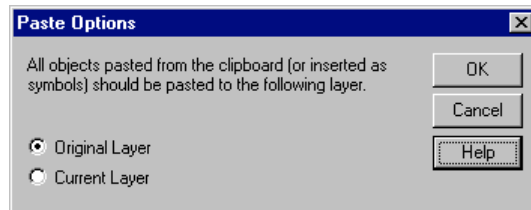
1. Select **Options/AutoSave** from the **Options** menu.
2. Pick the new autosave time interval.
3. Click **OK** with the mouse.



Pasting Option

DeltaCad will let you choose if you want pasted objects to go onto the current layer, or on the original layer the object was drawn on.

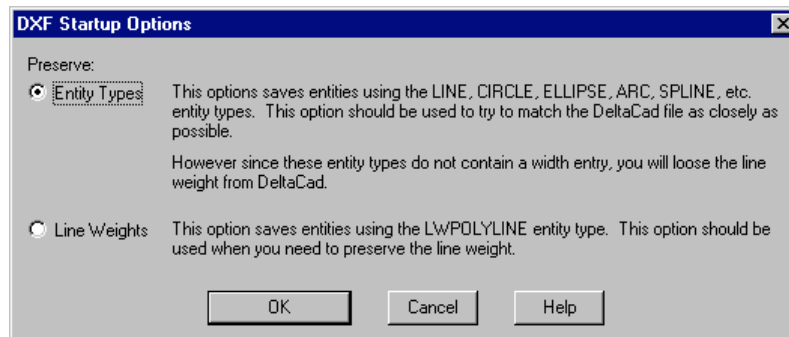
1. Select **Options/Pasting Options** from the **Options** menu.
2. Pick the option you want.
3. Click **OK** with the mouse.



DXF Options

This function is used to set the default options to be used when exporting a drawing to a .DXF file.

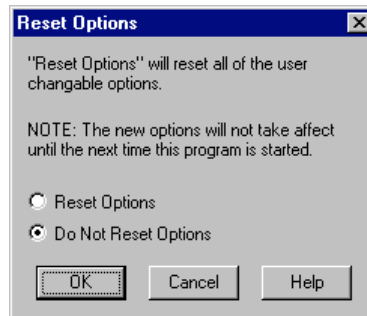
1. Select **Options/DXF Options** from the **Options** menu.
2. Pick the option you want.
3. Click **OK** with the mouse.



Reset Options

This function is provided to reset all of the user options to their default values. This function will not take effect until the next time DeltaCad is started.

1. Select **Options/Reset Options** from the **Options** menu.
2. When the popup window appears, select **Reset Options**, and then click **OK** with the mouse.
3. Exit DeltaCad and then restart DeltaCad for the new options to take effect.



Extract Group Data

The Extract Group Data function is used to create a Bill of Materials, or to create a list of all groups used in the drawing. To use this function pick **Extract Group Data** from the **Options** menu, enter the appropriate information in the popup window, and then pick **OK** with the mouse.



This function gives you a choice of three output devices for the data. The data can be sent to the printer, a file, or the clipboard. If data is sent to the Clipboard it can be easily imported into a word processing program, or a spreadsheet. There is also a choice for separating the fields with spaces, or tabs. If the data is going to be moved to a word processor which uses proportionally spaced fonts it may be better to separate the fields using tabs.

There is also a choice whether to include nested groups. A nested group is a group which is used on another group, and then used on the main drawing.

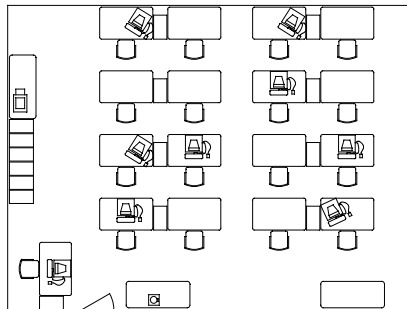
Group names which begin with an asterisk (*) will not be included. This provides an easy way to exclude some groups from the or Bill of Materials or Group Data list.

Bill of Materials

The Bill of Materials list is a list which shows the quantity of all groups which are used on a drawing. To create a Bill of Materials you should make groups for all objects to be included, and then execute the **Extract Group Data** function on the **Options** menu. The following Bill of Materials was generated from the office.dc sample drawing which is provided with DeltaCad.

BILL OF MATERIALS

QTY.	NAME
17	Desk
15	File
9	Computer
17	Chair
3	Table
1	Coffee
1	Printer



Group Data

The Group Data list is a list which shows the name and location of all groups used on a drawing, The following Group Data list was generated from the office.dc sample drawing provided with DeltaCad.

```

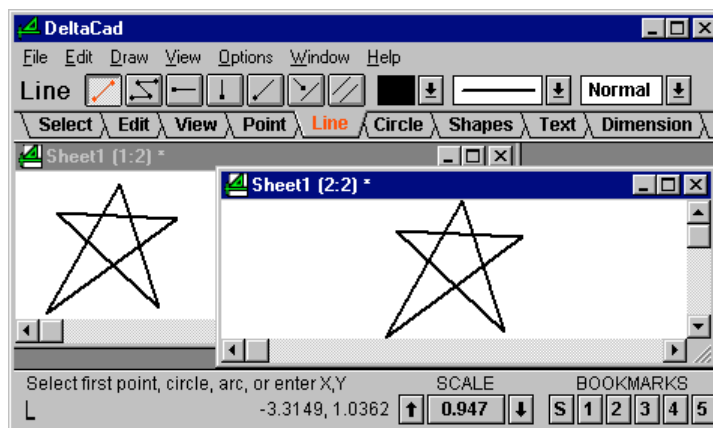
--- GROUP DATA ---
X      Y      NAME      X      Y      NAME
-74.2692  17.1790  Desk      175.7308 -156.8210  Chair
-35.7692  12.6790  File      -74.2692 -198.8210  Desk
2.7308    17.1790  Desk      -35.7692 -203.3210  File
-56.5710  16.0127  Computer  2.7308   -198.8210  Desk
98.7308   17.1790  Desk      98.7308  -198.8210  Desk
137.2308  12.6790  File      137.2308 -203.3210  File
175.7308  17.1790  Desk      175.7308 -198.8210  Desk
116.4290  16.0127  Computer  -74.2692 -228.8210  Chair
-74.2692  -12.8210  Chair     2.7308   -228.8210  Chair
2.7308    -12.8210  Chair     98.7308  -228.8210  Chair
98.7308   -12.8210  Chair     175.7308 -228.8210  Chair
175.7308  -12.8210  Chair     102.7308 -50.8210   Computer
-74.2692  -54.8210  Desk      179.7308 -122.8210  Computer
-35.7692  -59.3210  File      6.7308   -122.8210  Computer
2.7308    -54.8210  Desk      -70.2692 -194.8210  Computer
98.7308   -54.8210  Desk      163.8209 -194.9064  Computer
137.2308  -59.3210  File      -154.2693 -260.8211  Desk
175.7308  -54.8210  Desk      -158.7693 -299.3211  File
-74.2692  -84.8210  Chair     -184.2693 -260.8211  Chair
2.7308    -84.8210  Chair     -150.2693 -264.8211  Computer
98.7308   -84.8210  Chair     -191.2692 -54.8210   Table
175.7308  -84.8210  Chair     -192.7693 -99.3210   File
-74.2692  -126.8210  Desk     -192.7693 -115.3210  File
-35.7692  -131.3210  File     -192.7693 -131.3210  File
2.7308    -126.8210  Desk     -192.7693 -147.3210  File
-56.5710  -127.9873  Computer -192.7693 -163.3210  File
98.7308   -126.8210  Desk     -192.7693 -179.3210  File
137.2308  -131.3210  File     181.7308  -289.8210  Table
175.7308  -126.8210  Desk     -42.9638  -295.9187  Coffee
-74.2692  -156.8210  Chair    -38.2692  -289.8210  Table
2.7308    -156.8210  Chair    -194.1109 -70.3201   Printer
98.7308   -156.8210  Chair

```

Window

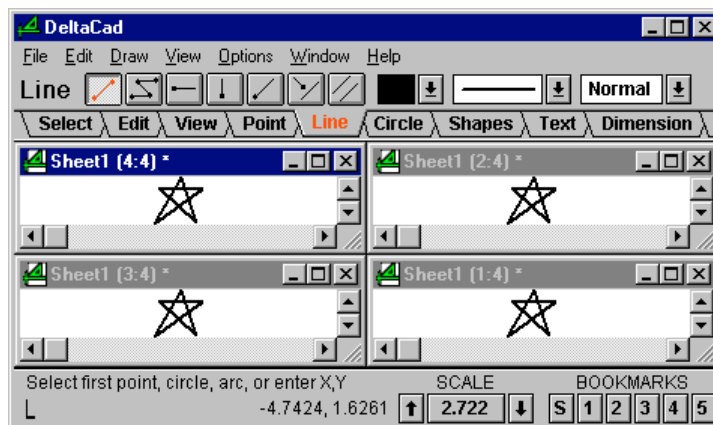
New Window

The New Window function in DeltaCad is used to create another window for the currently active drawing. This allows you to view one part of a drawing, and make changes to another part of the drawing. All drawing changes made to the drawing will appear in all of the drawing windows. To create a New Window pick **New Window** from the **Window** menu. There are a maximum of 9 windows which may be open at the same time.



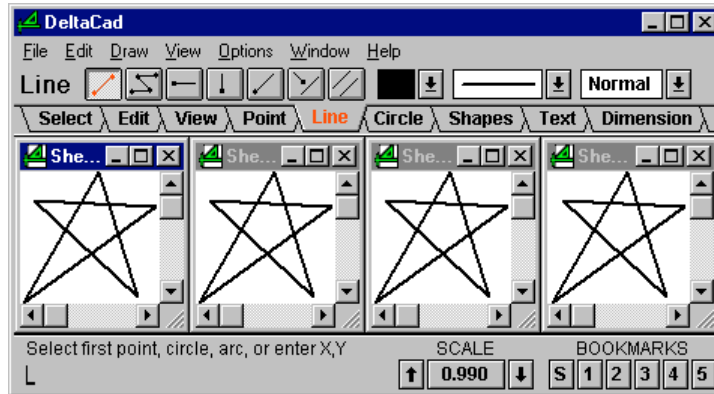
Tile

The Tile function is used to arrange all of the currently open DeltaCad drawing Windows. This function will move and size all open windows in a tile pattern so that none of the windows overlap



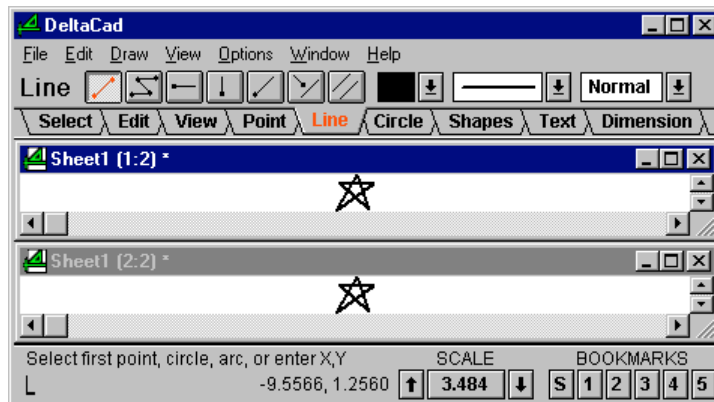
Tile Horizontally

The Tile Horizontally function is used to arrange all of the currently open DeltaCad drawing Windows in a horizontal pattern. This function will move and size all open windows in a horizontal pattern so that none of the windows overlap



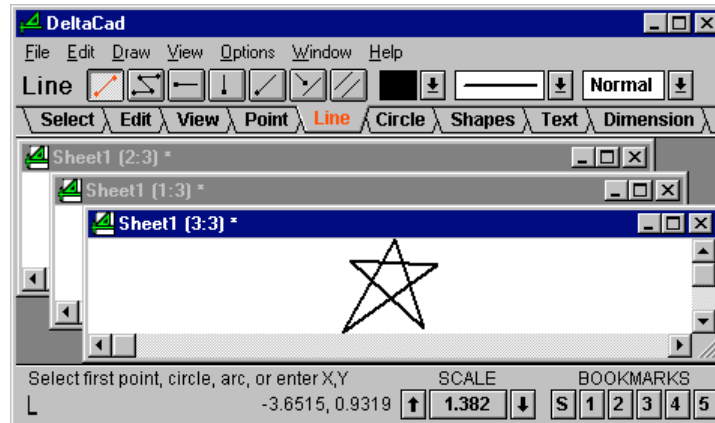
Tile Vertically

The Tile Vertically function is used to arrange all of the currently open DeltaCad drawing Windows in a vertical pattern. This function will move and size all open windows in a vertical pattern so that none of the windows overlap



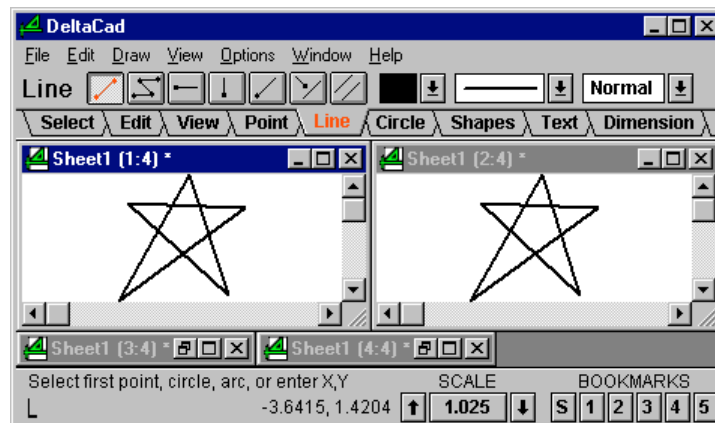
Cascade

The Cascade function is used to arrange all of the currently open DeltaCad drawing Windows in an overlapped pattern. This function will move and size all open windows in an overlapped format.



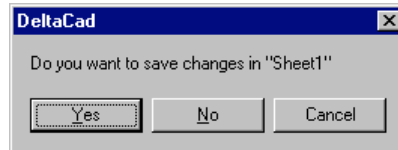
Arrange Icons

The Arrange Icons function is used to arrange the icons of any minimized drawing windows in a horizontal pattern at the bottom of the DeltaCad window.




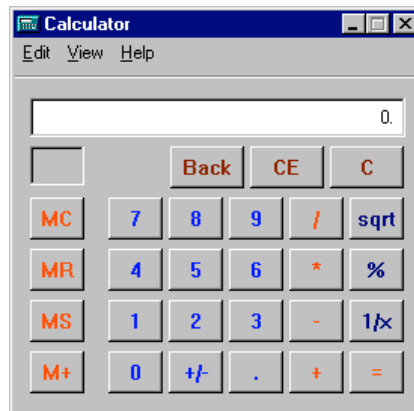
Close All

The Close All function is used to close all open drawing windows. If any of the drawing were not saved since the last changes were made you will be prompted whether the drawing should be saved prior to closing the window.



Calculator

The Calculator function is used as a convenient way to temporarily switch to the Microsoft® Windows™ calculator program. To start the calculator program select the **Edit** tab on the toolbar and then select the  button.



Macro Language

Macro Language

DeltaCad provides a Basic macro language to automate drawing tasks, or as a way to retrieve info. from a drawing. The macro language is the familiar BASIC programming language. A few examples are included with DeltaCad (in the program directory c:\Program Files\DeltaCad), and there are many examples included in the function descriptions in the manual and help system. To look at the samples pick “E” button on the Macro toolbar tab to start the editor, then pick **Open** from the **File** menu to open a sample macro.

Basic Scripting Engine provided by Cypress Software Inc., Copyright 1993-1999, all right reserved.

Cypress Software, Inc.
7541 N. Invergordon Rd.
Paradise Valley, AZ 85253

Phone: 602-922-4883

Fax: 602 951-8047

web: <http://www.cypressinc.com>

e-mail: cypress@cypressinc.com

For a list of functions see the sections shown below. For a description of the BASIC Programming Language pick **Basic Macro Language Ref** from the **Help** menu. The BASIC language ref. is also included in your DeltaCad directory in the BASIC.PDF file so you can print it out.

File functions

Layer functions

Group functions

Misc. functions

Object creating functions

Object parameter functions

Object retrieval functions

Print region functions

Select functions

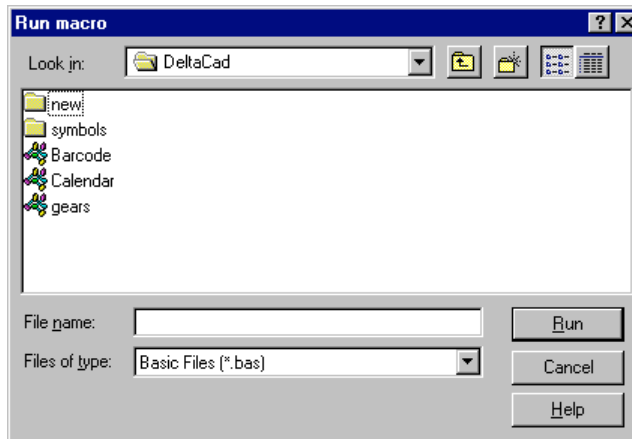
Symbol functions

View functions

Run macro

Use one of the following methods to run a macro.

1. Pick the “**R**” button on the Macro toolbar tab, pick the macro file to run, and pick the **Run** button..
2. Pick **Macro/Run** from the **Options** menu, pick the macro file to run, and pick the **Run** button.

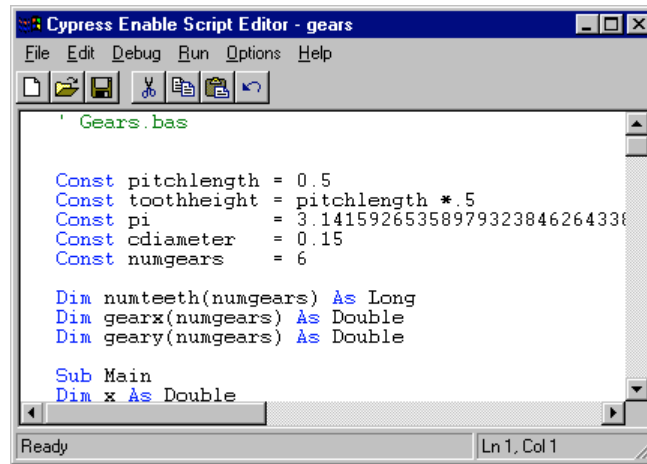


3. If the macro is on the macro drop-down list on the Macro toolbar tab, just pick the macro, and pick the **Run Macro** button.

Edit macro

Use one of the following methods to edit a macro.

1. Pick the “E” button on the Macro toolbar tab to start the macro editor. Then pick **Open** from the **File** menu in the editor to open a macro.
2. Pick **Macro/Edit** from the **Options** menu to start the macro editor. Then pick **Open** from the **File** menu in the editor to open a macro.



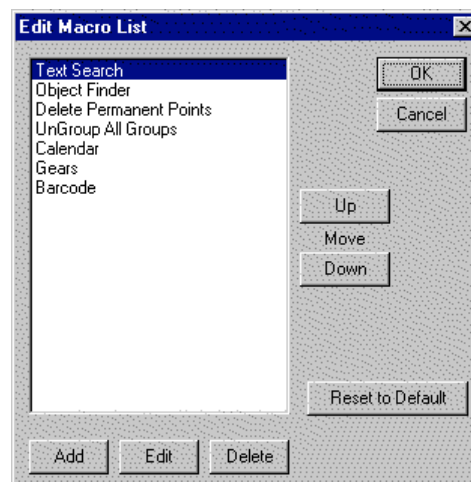
3. If the macro is on the macro drop-down list on the Macro toolbar tab, just pick the macro, and pick the **Edit Macro** button.

Macro Toolbar Tab

The Macro toolbar tab provides a quick and easy way to run and edit macros. Just pick the macro from the drop-down list, and then pick the **Run Macro** or **Edit Macro** button.



To add, delete, or edit the macro drop-down list, pick the **Edit Macro List** button on the Macro toolbar tab.



Macro functions listed by type

File functions

- dcChangedSinceLastSave
- dcClose
- dcCloseWithoutSaving
- dcOpen
- dcNew
- dcSave
- dcSaveAs
- dcSetDrawingWindowMode
- dcPrint
- dcPrintPreview
- dcEndPrintPreview

Layer functions

- dcAddLayer
- dcDoesLayerExist
- dcDeleteLayer
- dcGetCurrentLayer
- dcGetLayerName
- dcGetNumberOfLayers
- dcIsLayerOn
- dcRenameLayer
- dcSetCurrentLayer
- dcTurnLayerOff
- dcTurnLayerOn

Group functions

- dcGetFirstGroupName
- dcGetNextGroupName
- dcOpenGroupWindow
- dcUnGroupAllGroups
- dcUnGroupCurrentObject

Misc. functions

dcCopyBMPFiletoClipboard
dcEraseTempPoints
dcExportScreentoBMPFile
dcGetActiveWindow
dcGetBookmark
dcGetDeltaCadDwgName
dcGetDeltaCadProgramDirectory
dcGetDeltaCadVersion
dcGetDeltaCadWindowHandle
dcGetDrawingData
dcGetDrawingScale
dcGetDrawingWindowHandle
dcGetInitialSizeandPos
dcGetNumberOfDeltaCadDwgs
dcReadInput
dcRestoreBookmark
dcSetActiveWindow
dcSetBookmark
dcSetDrawingScale
dcSetInitialSizeandPos
dcSetDrawingData
dcUpdateDisplay

Object creation functions

- dcCreateBox
- dcCreateCircle
- dcCreateCircleEx
- dcCreateDim
- dcCreateLine
- dcCreatePoint
- dcCreateShape
- dcCreateSpline
- dcCreateText
- dcCreateTriangle

Object parameter functions

- dcGetCircleParms
- dcGetDimParms
- dcGetLineParms
- dcGetPointParms
- dcGetShapesParms
- dcGetSplineParms
- dcGetTextParms

- dcSetCircleParms
- dcSetDimParms
- dcSetLineParms
- dcSetPointParms
- dcSetShapesParms
- dcSetSplineParms
- dcSetTextParms

- dcSaveAllParms
- dcRestoreAllParms

Object retrieval functions

dcGetFirstObject
dcGetFirstSelectedObject
dcGetNextObject
dcGetNextSelectedObject
dcGetObjectAt

dcGetCircleData
dcGetDimData
dcGetLineData
dcGetGroupData
dcGetPictureData
dcGetPointData
dcGetShapeData
dcGetSplineData
dcGetTextData
dcCopyCurrentPicturetoFile

dcGetAreaofCurrentObject
dcGetLengthofCurrentObject
dcGetObjectBox
dcGetObjectLayer

dcIsCurrentObjectSelected
dcSelectCurrentObject
dcUnselectCurrentObject
dcEraseCurrentObject

Print region functions

dcAddPrintRegion
dcDeletePrintRegion
dcGetCurrentPrintRegion
dcGetNumberofPrintRegions
dcGetPrintRegionData
dcGetPrintRegionName
dcIsPrintRegionLayerOn
dcRenamePrintRegion
dcSetCurrentPrintRegion
dcTurnPrintRegionLayerOff
dcTurnPrintRegionLayerOn

Select functions

- dcChgColorSelObjs
- dcChgLayerSelObjs
- dcChgLineTypeSelObjs
- dcChgLineWeightSelObjs
- dcCopySelObjstoClipboard
- dcEraseSelObjs
- dcGetSelCopyMode
- dcGroupSelObjs
- dcIsCurrentObjectSelected
- dcIsSelected
- dcMoveSelObjs
- dcMirrorSelObjs
- dcPasteClipboardData
- dcPasteSpecialClipboardFormat
- dcRotateSelObjs
- dcScaleSelObjs
- dcSelectAll
- dcSelectCurrentObject
- dcSelectObjInBox
- dcSetSelCopyMode
- dcSetSelectBase
- dcUnSelectAll
- dcUnSelectCurrentObject

Symbol functions

- dcAddSymbol
- dcCopySymboltoClipboard
- dcDeleteSymbol
- dcNewSymbolFile

View functions

- dcGetViewScaleandRect
- dcSetViewScale
- dcViewAll
- dcViewBox
- dcViewCenteredOnXY
- dcViewPrevious

Alphabetical list of functions

dcAddLayer

dcAddLayer(*layername*)

This function adds a new layer with the name given by *layername*.

Example:

```
Sub Main
dcAddLayer "outline"
End Sub
```

Related Topics: dcDeleteLayer, dcRenameLayer, dcDoesLayerExist

dcAddPrintRegion

dcAddPrintRegion(*prname, scale, left, bottom, right, top, orientation*)

This function adds a new print region with the name given by *prname*. The print scale is given by *scale*. A *scale* of 0.0 causes the drawing to be scaled to fit the printer paper. The print area is given by *left, bottom, right, and top*. Entering all zero's for the print area makes DeltaCad print the entire drawing. *orientation* can be set to either dcPortrait or dcLandscape.

Example:

```
Sub Main
dcAddPrintRegion "outline", 1.0, 0, 0, 0, 0, dcPortrait
End Sub
```

Related Topics: dcDeletePrintRegion, dcRenamePrintRegion

dcAddSymbol

dcAddSymbol(*symbolfilename, sectionname, symbolname*)

This function adds the DeltaCad object on the clipboard to the symbol file given by *symbolfilename*. It is assigned to the section name given by *sectionname*, and the symbol name given by *symbolname*.

Example:

```
Sub Main
dcAddSymbol "c:\deltacad\symbols\new.sym", "def", "newsym"
End Sub
```

Related Topics: dcDeleteSymbol, dcNewSymbolFile, dcCopySymboltoClipboard

dcChangedSinceLastSave

dcChangedSinceLastSave

This function indicates if the current DeltaCad drawing has been changed since the last save.

Return Value: True if the drawing has been changed since the last save, otherwise False.

Example:

```
Sub Main
If (dcChangedSinceLastSave) Then
MsgBox "The drawing has been changed."
Else
MsgBox "The drawing has not been changed."
End If
End Sub
```

Related Topics: dcOpen, dcClose, dcSave, dcSaveAs

dcChgColorSelObjs

dcChgColorSelObjs(*color*)

This function changes the color of all selected objects to the color given in *color*.

color is the DeltaCad color number. The color number starts with zero, and increases in the same order that the colors are displayed in the color selection box in DeltaCad (i.e. 0 is black, 1 is red, 2 is green, 3 is yellow, etc.). You can also use the following pre-defined color values: dcBLACK, dcRED, dcGREEN, dcYELLOW, dcBLUE, dcPURPLE, dcBROWN, dcDARKGRAY, dcLIGHTGRAY, dcLIGHTBLUE, dcDARKGREEN, dcGREENYELLOW, dcDARKBLUE, dcDARKPURPLE, dcGRAYBLUE, dcWHITE

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcChgColorSelObjs dcGREEN
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll

dcChgLayerSelObjs

dcChgLayerSelObjs(*newlayer*)

This function changes the layer of all selected objects to the name given in *newlayer*.

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcChgLayerSelObjs "object"
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll

dcChgLineTypeSelObjs

dcChgLineTypeSelObjs(*linetype*)

This function changes the line type of all selected objects to the *linetype* given in *linetype*.

linetype is the DeltaCad line type (NOTE: *linetype* for “lines” will not be set to Filled or Hatched). This may be set to one of the following: dcSOLID, dcHIDDEN, dcCENTER, dcPHANTOM, dcCUTTING, dcBREAK, dcSTITCH, dcARROW, dcTWOARROW, dcFILL, dcHATCH, dcHATCHNOOUTLINE. If using dcHATCH or dcHATCHNOOUTLINE you may add numbers (0 to 39) to the value to get the various hatch types.

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcChgLineTypeSelObjs dcCENTER
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll

dcChgLineWeightSelObjs

dcChgLineWeightSelObjs(*lineweight*)

This function changes the line weight of all selected objects to the *lineweight* given in *lineweight*.

lineweight is the DeltaCad line weight. It can be one of the following values: dcTHIN, dcNORMAL, dcTHICK, dcHEAVY or dcWIDE

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcChgLineWeightSelObjs dcTHICK
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll

dcClose

dcClose

This function closes the currently open DeltaCad window.

Example:

```
Sub Main
dcOpen "C:\DeltaCad\lines.dc"
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcSave
dcClose
End Sub
```

Related Topics: dcCloseWithoutSaving, dcNew, dcOpen, dcSave, dcSaveAs

dcCloseWithoutSaving

dcCloseWithoutSaving

This function closes the currently open DeltaCad window without prompting the user to save the drawing (if it has been changed since the last save).

Example:

```
Sub Main
dcOpen "C:\DeltaCad\lines.dc"
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcSave
dcCloseWithoutSaving
End Sub
```

Related Topics: dcNew, dcOpen, dcSave, dcSaveAs

dcCopyBMPFiletoClipboard

dcCopyBMPFiletoClipboard(*filename*)

This function copies the bitmap (.BMP) file given by *filename* to the clipboard.

Example:

```
Sub Main
dcCopyBMPFiletoClipboard "c:\program files\deltacad\scr.bmp"
dcPasteSpecialClipboardData 1, 2, dcBitmapFormat
End Sub
```

Related Topics: dcExportScreentoBMPFile, dcPasteClipboardData, dcPasteSpecialClipboardData

dcCopyCurrentPicturetoFile

dcCopyCurrentPicturetoFile(*filename*)

This function copies the current picture to the file with the name given by *filename*.

Example:

```
Sub Main
'This sub copies the first picture to a file
Dim ot As Long
Dim pictype As Long
Dim x As Double
Dim y As Double
Dim w As Double
Dim h As Double
Dim orig_w As Double
Dim orig_h As Double
Dim layer As String
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcPicture Then
    layer = Space(255)
```

```

    dcGetPictureData pictype, x, y, w, h, orig_w, orig_h, layer
    If pictype = dcPicBMP Then
        dcCopyCurrentPictureToFile "c:\program
files\deltacad\pic.bmp"
    End If
    MsgBox layer
    Exit Do
    End If
    ot = dcGetNextObject
Wend
End Sub

```

Related Topics: dcGetPictureData

dcCopySelObjstoClipboard

dcCopySelObjstoClipboard

This function copies selected objects to the clipboard.

Example:

```

Sub Main
dcSelectAll
dcCopySelObjstoClipboard
End Sub

```

Related Topics: dcSelectAll, dcUnSelectAll, dcPasteClipboardData, dcPasteSpecialClipboardData

dcCopySymboltoClipboard

dcCopySymboltoClipboard(*symbolfilename*, *sectionname*, *symbolname*)

This function copies the symbol given by *sectionname* and *symbolname* in the file given by *symbolfilename* to the clipboard.

Example:

```

Sub Main
dcCopySymboltoClipboard "c:\deltacad\symbols\new.sym", _
    "def", "newsym"
End Sub

```

Related Topics: dcAddSymbol, dcDeleteSymbol, dcNewSymbolFile

dcCreateBox

dcCreateBox(*left*, *bottom*, *right*, *top*)

This function creates a rectangle at the location specified by the *left*, *bottom*, *right*, *top* coordinates.

Return Value: True if successful, otherwise False.

Example:

```

Sub Main
For x = 1 to 5
    dcCreateBox -x, 0, x, 3

```

```
Next x
End Sub
```

Related Topics: dcSetShapesParms, dcGetShapesParms

dcCreateCircle

dcCreateCircle(*x*, *y*, *r*)

This function creates a circle at the point specified by *x* and *y*, with a radius as specified by *r*.

Return Value: True if successful, otherwise False.

Example:

```
Sub Main
For r = 1 to 5
  dcCreateCircle 0, 0, r
Next r
End Sub
```

Related Topics: dcSetCircleParms, dcGetCircleParms

dcCreateCircleEx

dcCreateCircleEx(*cx*, *cy*, *xa*, *ya*, *xb*, *yb*, *hrad*, *wrad*, *angle*, *arrow*)

This function creates a circle, arc, ellipse, or elliptical arc. *cx* and *cy* are the center, *xa/ya* and *xb/yb* are the endpoints of an arc. If you are drawing a complete circle (i.e. not an arc) then *xa/ya* and *xb/yb* should be the same point. If you are drawing an arc it is drawn counterclockwise from *xa/ya* to *xb/yb*. *hrad* is the height radius, and *wrad* is the width radius. If drawing a circle (i.e. not an ellipse) set *hrad* and *wrad* to the same value. *angle* is the angle of the ellipse in degrees (it is ignored for circles). *arrow* indicates which end of the arc segment should get an arrow head. 1 is end A, and 2 is end B. The *arrow* value is only used if the line type is set to an arrow line.

Return Value: True if successful, otherwise False.

Example:

```
Sub Main
dcCreateCircleEx 0, 0, 1.41, 1.41, 1.41, 1.41, 1, 2, 45, 0
End Sub
```

Related Topics: dcSetCircleParms, dcGetCircleParms

dcCreateDim

dcCreateDim(*xa*, *ya*, *xb*, *yb*, *xt*, *yt*, *dimangle*, *extlineangle*, *dimtype*, *extline*, *text*)

This function creates a dimension object. *xa/ya* and *xb/yb* specify the ends of the dimension, *xt/yt* specify the text location of the dimension. *dimangle* is the dimension angle in degrees, and *extlineangle* is the extension line angle in

degrees. *dimtype* and *extline* are as specified below. *text* is the actual text to be displayed

dimtype

- 0 Horizontal dimension.
- 1 Vertical dimension.
- 2 Angled dimension.
- 3 Angled with angled extension lines dimension.
- 4 Arrow dimension.
- 5 Diameter dimension.
- 6 Radius dimension.
- 7 Angle dimension.

extline

- 0 No extension lines.
- 1 Extension line at end A only.
- 2 Extension line at end B only.
- 3 Both extension lines.

Return Value: True if successful, otherwise False.

Example:

```
Sub Main
dcCreateDim -2, 0, 2, 0, 0, 1, 0, 90, 0, 3, "4.0"
End Sub
```

Related Topics: dcSetDimParms, dcGetDimParms

dcCreateLine

dcCreateLine(*xa*, *ya*, *xb*, *yb*)

This function creates a line between the points at the locations specified by *xa*, *ya*, and *xb*, *yb*.

Return Value: True if successful, otherwise False.

Example:

```
Sub Main
For x = 1 to 5
  dcCreateLine x, 5, x, 10
Next x
End Sub
```

Related Topics: dcSetLineParms, dcGetLineParms

dcCreatePoint

dcCreatePoint(*x*, *y*)

This function creates a point at the location specified by *x* and *y*.

Return Value: True if successful, otherwise False.

```
Example:  
Sub Main  
dcCreatePoint 1.2, 3.4  
For x = 1 to 5  
    dcCreatePoint x, 5  
Next x  
End Sub
```

Related Topics: dcSetPointParms, dcGetPointParms

dcCreateShape

dcCreateShape(*pt*, *num*)

This function creates a shape. *pt* is an array of Double numeric values where every other array element is the X and Y coordinate of the shape point. *num* is the number of points.

Return Value: True if successful, otherwise False.

```
Example:  
Sub Main  
Dim a(8) As Double  
a(1) = -0.5  
a(2) = 0.5  
a(3) = 0  
a(4) = 0  
a(5) = 0.5  
a(6) = 0.5  
a(7) = 0  
a(8) = -0.5  
dcCreateShape a(1), 4  
End Sub
```

Related Topics: dcSetShapesParms, dcGetShapesParms

dcCreateSpline

dcCreateSpline(*pt*, *num*, *closed*)

This function creates an open or closed spline. *pt* is an array of Double numeric values where every other array element is the X and Y coordinate of the spline point. *num* is the number of points, and *closed* specifies whether the array is closed or open.

Return Value: True if successful, otherwise False.

```
Example:  
Sub Main  
Dim a(8) As Double  
a(1) = -0.5  
a(2) = 0.5  
a(3) = 0  
a(4) = 0  
a(5) = 0.5  
a(6) = 0.5  
a(7) = 0
```

```
a(8) = -0.5
dcCreateSpline a(1), 4, True
End Sub
```

Related Topics: dcSetLineParms, dcGetLineParms, dcSetShapesParms, dcGetShapesParms

dcCreateText

dcCreateText(*x*, *y*, *w*, *text*)

This function creates a text object at the location specified by *x/y*. *w* is the width of the text. *w* is only used for multi-line text. *text* is the actual text to be displayed

Return Value: True if successful, otherwise False.

Example:

```
Sub Main
dcCreateText 1, 2, 0, "DeltaCad"
End Sub
```

Related Topics: dcSetTextParms, dcGetTextParms

dcCreateTriangle

dcCreateTriangle(*xa*, *ya*, *xb*, *yb*, *xc*, *yc*)

This function creates a triangle at the location specified by the three points *xa/ya*, *xb/yb*, and *xc/yc*.

Return Value: True if successful, otherwise False.

Example:

```
Sub Main
dcCreateTriangle -1, 0, 1, 1, 1, 0
End Sub
```

Related Topics: dcSetShapesParms, dcGetShapesParms

dcDeleteLayer

dcDeleteLayer(*layername*)

This function deletes the layer with the name given by *layername*.

Example:

```
Sub Main
dcAddLayer "outline"
dcDeleteLayer "outline"
End Sub
```

Related Topics: dcAddLayer, dcRenameLayer, dcDoesLayerExist

dcDeletePrintRegion

dcDeletePrintRegion(*prname*)

This function deletes the print region with the name given by *prname*.

Example:

```
Sub Main
dcAddPrintRegion "outline", 1.0, 0, 0, 0, 0, dcPortrait
dcDeletePrintRegion "outline"
End Sub
```

Related Topics: dcAddPrintRegion, dcRenamePrintRegion

dcDeleteSymbol

dcDeleteSymbol(*symbolfilename*, *sectionname*, *symbolname*)

This function deletes the symbol given by *symbolname* from the section given by *sectionname* in the symbol file given by *symbolfilename*.

Example:

```
Sub Main
dcDeleteSymbol "c:\deltacad\symbols\new.sym", "def", "newsym"
End Sub
```

Related Topics: dcAddSymbol, dcNewSymbolFile, dcCopySymboltoClipboard

dcDoesLayerExist

dcDoesLayerExist (*layername*)

This function determines if the layer given by *layername* exists in the drawing.

Return Value: True if the layer exists, otherwise False.

Example:

```
Sub Main
If dcDoesLayerExist("outline") Then
MsgBox "Layer exists"
Else
MsgBox "Layer does not exist"
End If
End Sub
```

Related Topics: dcAddLayer, dcDeleteLayer, dcRenameLayer

dcEndPrintPreview

dcEndPrintPreview

This function ends the current print preview window.

Example:

```
Sub Main
dcPrintPreview
End Sub
```

Related Topics: dcPrintPreview

dcEraseCurrentObject

dcEraseCurrentObject

This function deletes the current object. The current object is set in `dcGetFirstObject`, `dcGetNextObject`, `dcGetFirstSelectedObject`, `dcGetNextSelectedObject`, or `dcGetObjectAt`.

Example:

```
Sub Main
  'This sub erases all lines
  Dim ot As Long
  ot = dcGetFirstObject("")
  While ot <> dcNone
    If ot = dcLine Then
      dcEraseCurrentObject
    End If
    ot = dcGetNextObject
  Wend
End Sub
```

Related Topics: `dcGetFirstObject`, `dcGetNextObject`, `dcGetObjectAt`, `dcGetFirstSelectedObject`, `dcGetNextSelectedObject`

dcEraseSelObjs

dcEraseSelObjs

This function erases all selected objects.

Example:

```
Sub Main
  dcSelectObjInBox 0, 0, 2, 2
  dcEraseSelObjs
End Sub
```

Related Topics: `dcSelectAll`, `dcUnselectAll`

dcEraseTempPoints

dcEraseTempPoints

This function erases all temporary points.

Example:

```
Sub Main
  dcEraseTempPoints
End Sub
```

Related Topics: `dcCreatePoint`

dcExportScreentoBMPFile

dcExportScreentoBMPFile(*filename*, *dpi*)

This function exports the current screen to the bitmap (.BMP) file given by *filename*. The file will have the resolution (dots-per-inch) given by *dpi*. If *dpi* is 0 the current screen resolution will be used.

Example:

```
Sub Main
```

```
dcExportScreenToBMPFile "c:\program files\deltacad\scr.bmp", 300
End Sub
```

Related Topics: dcCopyBMPFiletoClipboard

dcGetActiveWindow

dcGetActiveWindow(*filename*)

This function gets the filename of the active DeltaCad window. The string specified by *filename* must be large enough to hold the result or an error will occur. See the example below.

Example:

```
Sub Main
Dim fn As String
fn = space(256)
dcGetActiveWindow fn
MsgBox "The current filename is - " + fn
End Sub
```

Related Topics: dcSetActiveWindow

dcGetAreaofCurrentObject

dcGetAreaofCurrentObject

This function gets the area of the current object. This function only works with Circles, Shapes, and Splines.

Return Value: The area is returned. If a non-supported object is used (such as a line), zero is returned.

Example:

```
Sub Main
Dim ot As Long
ot = dcGetFirstObject("")
While ot <> dcNone
  MsgBox dcGetAreaofCurrentObject
  ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetBookmark

dcGetBookmark(*num, x, y, scale*)

This function gets the bookmark *x*, *y*, and *scale* for bookmark number *num*.

Example:

```
Sub Main
Dim j As Long
Dim x As Double
Dim y As Double
Dim scale As Double
```

```

For j=1 to 5
dcGetBookmark j, x, y, scale
MsgBox "Bookmark"+Format(j)+" = "+Format(x)+" "+Format(y)+" , _
      "+Format(scale)
Next j
End Sub

```

Related Topics: dcSetBookmark, dcRestoreBookmark

dcGetCircleData

dcGetCircleData(*cx, cy, xa, ya, xb, yb, hrad, wrad, angle, arrow, color, linetype, lineweight, layer*)

This function gets all of the data associated with the current circle object. *cx,cy* is the center of the circle, *xa, ya, xb, yb* are the arc endpoints, if *xa,ya* and *xb,yb* are the same point then it is a circle (not an arc). *hrad* is the height radius, and *wrad* is the width radius. For a circle *hrad* and *wrad* will be the same, but for an ellipse they will be different. *angle* is the ellipse angle. *arrow* is set to 1 if there is an arrowhead on end A, and is set to 2 if there is an arrowhead on end B. *arrow* is only valid if the line type is dcARROW. *color* is the circle color, *linetype* is the linetype, *lineweight* is the lineweight, *layer* is the layername.

The variable for *layer* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetCircleData. See the example below.

Example:

```

Sub Main
'This sub finds all circles
Dim ot As Long
Dim cx As Double
Dim cy As Double
Dim xa As Double
Dim ya As Double
Dim xb As Double
Dim yb As Double
Dim wrad As Double
Dim hrad As Double
Dim angle As Double
Dim arrow As Long
Dim color As Long
Dim lt As Long
Dim lw As Long
Dim layer As String
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcCircle Then
    layer = Space(255)
    dcGetCircleData cx, cy, xa, ya, xb, yb, hrad, wrad, _
      angle, arrow, color, linetype, lineweight, layer
    MsgBox Format(cx)+" "+Format(cy)
  End If
  ot = dcGetNextObject

```

```
Wend  
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetCircleParms

dcGetCircleParms(*color, linetype, lineweight*)

This function gets all of the current parameters used to create a circle.

See dcSetCircleParms for a description of *color, linetype* or *lineweight*.

Example:

```
Sub Main  
Dim color As Long  
Dim linetype As Long  
Dim lineweight As Long  
dcGetCircleParms color, linetype, lineweight  
MsgBox "Color = " + Format(color)  
MsgBox "LineType = " + Format(linetype)  
MsgBox "LineWeight = " + Format(lineweight)  
End Sub
```

Related Topics: dcSetCircleParms, dcSaveAllParms, dcRestoreAllParms

dcGetCurrentLayer

dcGetCurrentLayer(*layername*)

This function gets the current layer and puts it into the variable *layername*.

The variable for *layername* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetCurrentLayer. See the example below.

Example:

```
Sub Main  
Dim s As String  
s = space(255)  
dcGetCurrentLayer s  
MsgBox "Current Layer is " + s  
End Sub
```

Related Topics: dcSetCurrentLayer

dcGetCurrentPrintRegion

dcGetCurrentPrintRegion(*prname*)

This function gets the current print region and puts it into the variable *prname*.

The variable for *prname* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetCurrentPrintRegion. See the example below.

Example:

```
Sub Main
Dim s As String
s = space(255)
dcGetCurrentPrintRegion s
MsgBox "Current Print Region is " + s
End Sub
```

Related Topics: dcSetCurrentPrintRegion

dcGetDeltaCadDwgName

dcGetDeltaCadDwgName(*num*, *dwgname*)

This function gets the filename for any open DeltaCad drawing. *num* is the drawing number. It must be less than or equal to the number returned by dcGetNumberOfDeltaCadDwgs. *dwgname* is the variable to receive the filename.

The variable for *dwgname* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetDeltaCadDwgName. See the example below.

Example:

```
Sub Main
Dim j As Long
Dim n As Long
Dim s As String
n = dcGetNumberOfDeltaCadDwgs
MsgBox "Number of open drawings = " + Format(n)
For j=1 To n
s = Space(255)
dcGetDeltaCadDwgName j, s
MsgBox s
Next j
End Sub
```

Related Topics: dcGetNumberOfDeltaCadDwgs

dcGetDeltaCadProgramDirectory

dcGetDeltaCadProgramDirectory(*prokdir*)

This function gets the DeltaCad program disk directory and puts it in the *prokdir* variable.

The variable for *prokdir* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetDeltaCadProgramDirectory. See the example below.

Example:

```
Sub Main
Dim s As String
s = space(255)
dcGetDeltaCadProgramDirectory s
MsgBox "Program Directory = " + s
End Sub
```

dcGetDeltaCadVersion

dcGetDeltaCadVersion(*major, minor, relmonth, relday, reyear*)

This function gets the DeltaCad major and minor version numbers in *major* and *minor*, and the release date of the program in *relmonth*, *relday* and *reyear*.

Example:

```
Sub Main
Dim major As Long
Dim minor As Long
Dim relmonth As Long
Dim relday As Long
Dim reyear As Long
dcGetDeltaCadVersion major, minor, relmonth, relday, reyear
MsgBox "DeltaCad version " + Format(major) + "." + Format(minor)
MsgBox "Release Date " + Format(relmonth) + "/" + Format(relday) + _
      "/" + Format(reyear)
End Sub
```

dcGetDeltaCadWindowHandle

dcGetDeltaCadWindowHandle

This function gets the window handle for the main DeltaCad window. This can be used to call Windows functions.

Return Value: Is the window handle for the main active DeltaCad window drawing.

Example:

```
Sub Main
Dim h As Long
h = dcGetDeltaCadWindowHandle
End Sub
```

Related Topics: dcGetDrawingWindowHandle

dcGetDimData

dcGetDimData(*xa, ya, xb, yb, xt, yt, x, y, text, dimtype, extline, dimangle, extlineangle, height, font, style, color, layer*)

This function gets all of the data associated with the current dimension object. *xa, ya* and *xb, yb* are the coordinates of ends A and B. *xt, yt* are the coordinates of the text insertion point. *text* is the actual text displayed. *dimtype* is the type of dimension. *extline* tells if the extension lines are turned on or off. See dcCreateDim for a description of *dimtype* and *extline*. *dimangle* is the dimension angle in degrees, and *extlineangle* is the extension line angle. *height* is the text height in points. *font* and *style* are the font and style of the typeface used. *color* is the dimension color, and *layer* is the layername.

The variables for *text*, *font*, *style* and *layer* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetDimData. See the example below.

Example:

```

Sub Main
'This sub finds all dimensions
Dim ot As Long
Dim xa As Double
Dim ya As Double
Dim xb As Double
Dim yb As Double
Dim xt As Double
Dim yt As Double
Dim text As String
Dim dimtype As Long
Dim extline As Long
Dim dimangle As Double
Dim extlineangle As Double
Dim height As Double
Dim font As String
Dim style As String
Dim color As Long
Dim layer As String
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcDimension Then
    text = Space(255)
    font = Space(255)
    style = Space(255)
    layer = Space(255)
    dcGetDimData xa, ya, xb, yb, xt, yt, text, _
      dimtype, extline, dimangle, extlineangle, height, _
      font, style, color, layer
    MsgBox text
  End If
  ot = dcGetNextObject
Wend
End Sub

```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetDimParms

dcGetDimParms(*color, fontname, fontstyle, height, InOut, AlwaysHorz*)

This function gets all of the current parameters used to create a dimension object.

The variable for *fontname* and *fontstyle* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetDimParms. See the example below.

See dcSetDimParms for a description of *color, fontname, fontstyle, height, InOut, AlwaysHorz*.

Example:

```

Sub Main
Dim color As Long
Dim fontname As String

```

```
Dim fontstyle As String
Dim height As Double
Dim InOut As Boolean
Dim AlwaysHorz As Boolean
fontname = space(32)
fontstyle = space(32)
dcGetDimParms color, fontname, fontstyle, height, _
    InOut, AlwaysHorz
MsgBox "color = " + Format(color)
MsgBox "fontname = "+fontname
MsgBox "fontstyle = "+fontstyle
MsgBox "height = " + Format(height)
If(InOut) Then
MsgBox "Arrows Pointing in"
Else
MsgBox "Arrows Pointing out"
End If
If(AlwaysHorz) Then
MsgBox "Dimension text horizontal"
Else
MsgBox "Dimension text parallel to dimension"
End If
End Sub
```

Related Topics: dcSetDimParms, dcSaveAllParms, dcRestoreAllParms

dcGetDrawingData

dcGetDrawingData(*index*, *value*)

This function gets the drawing data item associated with the given *index* and stores it in the variable given by *value*.

When getting dcScalewithViewScale or dcAtLeastOnePixBigger, 1.0 means True and 0.0 means False.

See dcSetDrawingData for a list of valid *index* values.

Example:

```
Dim v As Double
dcSetDrawingData dcDimGap, 0.0625
dcSetDrawingData dcPhantomDash, 0.15
dcSetDrawingData dcScalewithViewScale, 0.0
dcGetDrawingData dcDimGap, v : MsgBox "Dim Gap = "+Format(v)
dcGetDrawingData dcScalewithViewScale, v
If v = 1.0 Then
MsgBox "Scale with view scale"
Else
MsgBox "Do not scale with view scale"
End If
```

Related Topics: dcSetDrawingData

dcGetDrawingScale

dcGetDrawingScale(*scale*)

This function gets the drawing scale.

Return Value: Is the drawing scale.

Example:

```
Sub Main
Dim ds as Double
ds = dcGetDrawingScale
MsgBox "Drawing Scale = "+Format(ds)
End Sub
```

Related Topics: dcSetDrawingScale

dcGetDrawingWindowHandle

dcGetDrawingWindowHandle

This function gets the window handle for the currently active DeltaCad drawing. This can be used to call Windows functions.

Return Value: Is the window handle for the currently active DeltaCad drawing.

Example:

```
Sub Main
Dim h As Long
h = dcGetDrawingWindowHandle
End Sub
```

Related Topics: dcGetDeltaCadWindowHandle

dcGetFirstGroupName

dcGetFirstGroupName(*gname*)

This function gets the first group name, and returns it in the *gname* variable. Use dcGetNextGroupName to get the rest of the group names.

Return Value: True if a group name is returned, and False if there are no groups.

Example:

```
Sub Main
Dim gname As String
gname = Space(255)
If dcGetFirstGroupName (gname) Then
  MsgBox gname
  gname = Space(255)
  While dcGetNextGroupName (gname)
    MsgBox gname
    gname = Space(255)
  Wend
Else
  MsgBox "Can't find first group name"
End If
End Sub
```

Related Topics: dcGetNextGroupName , dcOpenGroupWindow

dcGetFirstObject

dcGetFirstObject(*groupname*)

This function gets the first object in the group name given by *groupname*. If retrieving data from the main drawing (i.e. not in a group) pass an empty string to *groupname*. After this function is called use the data retrieval functions to get the actual data on a drawing object. See the example below.

Return Value: Is the type of object found, or dcNone if nothing is found. The following is a list of valid object types: dcPoint, dcLine, dcSpline, dcCircle, dcShape, dcText, dcDimension, dcGroup, dcPicture.

Example:

```
Sub Main
'This sub finds all red lines
Dim ot As Long
Dim xa As Double
Dim ya As Double
Dim xb As Double
Dim yb As Double
Dim color As Long
Dim lt As Long
Dim lw As Long
Dim layer As String
Dim arrow As Long
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcLine Then
    layer = Space(255)
    dcGetLineData xa, ya, xb, yb, color, lt, lw, layer, arrow
    If color = dcRED Then
      MsgBox Format(xa)+", "+Format(ya)+ _ " "+Format(xb)+ _
        ", "+Format(yb)
    End If
  End If
  ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetNextObject

dcGetFirstSelectedObject

dcGetFirstSelectedObject

This function gets the first selected object. After this function is called use the data retrieval functions to get the actual data on a drawing object. See the example below.

Return Value: Is the type of object found, or dcNone if nothing is found. The following is a list of valid object types: dcPoint, dcLine, dcSpline, dcCircle, dcShape, dcText, dcDimension, dcGroup, dcPicture.

Example:

```
Sub Main
'This sub finds all selected lines
```

```

Dim ot As Long
Dim xa As Double
Dim ya As Double
Dim xb As Double
Dim yb As Double
Dim color As Long
Dim lt As Long
Dim lw As Long
Dim layer As String
Dim arrow As Long
ot = dcGetFirstSelectedObject
While ot <> dcNone
  If ot = dcLine Then
    layer = Space(255)
    dcGetLineData xa, ya, xb, yb, color, lt, lw, layer, arrow
    MsgBox Format(xa)+", "+Format(ya)+ _ "    "+Format(xb)+ _
           ", "+Format(yb)
  End If
  ot = dcGetNextSelectedObject
Wend
End Sub

```

Related Topics: dcGetNextSelectedObject

dcGetGroupData

dcGetGroupData(*x, y, name, scale, angle, mirror, layer*)

This function gets all of the data associated with the current group object. *x,y* is the group insertion point, *name* is the group name. *scale* is the insertion scale. *angle* is the insertion angle. If *mirror* is True then the group is mirrored horizontally before being displayed. *layer* is the layername.

The variable for *layer* and *name* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetGroupData. See the example below.

Example:

```

Sub Main
  'This sub finds all groups
  Dim ot As Long
  Dim x As Double
  Dim y As Double
  Dim name As String
  Dim scale As Double
  Dim angle As Double
  Dim mirror As Boolean
  Dim layer As String
  ot = dcGetFirstObject("")
  While ot <> dcNone
    If ot = dcGroup Then
      name = Space(255)
      layer = Space(255)
      dcGetGroupData x, y, name, scale, angle, mirror, layer
      MsgBox name
    End If
    ot = dcGetNextObject
  End While
End Sub

```

```
Wend  
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetInitialSizeandPos

dcGetInitialSizeandPos(*x,y, scale*)

This function gets the initial size and position used when a DeltaCad drawing is opened. If the *scale* is 0.0 the drawing will be zoomed to fill the window when the drawing is opened.

Example:

```
Sub Main  
Dim x As Double  
Dim y As Double  
Dim scale As Double  
dcGetInitialSizeandPos x, y, scale  
MsgBox Format(x)+", "+Format(y)+", "+Format(scale)  
End Sub
```

Related Topics: dcSetInitialSizeandPos

dcGetLayerName

dcGetLayerName(*layernumer, layername*)

This function gets the *layername* for any *layernumber*.

The variable for *layername* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetLayerName. See the example below.

Example:

```
Sub Main  
Dim n As Long  
Dim j As Long  
Dim s As String  
n = dcGetNumberOfLayers  
For j = 1 To n  
s = space(255)  
dcGetLayerName j, s  
MsgBox "Layer name is - " + s  
Next j  
End Sub
```

Related Topics: dcGetNumberOfLayers

dcGetLengthofCurrentObject

dcGetLengthofCurrentObject

This function gets the length of the current object. This function only works with Lines, Circles, Shapes, and Splines.

Return Value: The area is returned. If a non-supported object is used (such as a dimension), zero is returned.

Example:

```
Sub Main
Dim ot As Long
ot = dcGetFirstObject("")
While ot <> dcNone
    MsgBox dcGetLengthofCurrentObject
    ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetLineData

dcGetLineData(*xa, ya, xb, yb, color, linetype, lineweight, layer, arrow*)

This function gets all of the data associated with the current line object. *xa, ya, xb, yb* are the line endpoints, *color* is the line color, *linetype* is the linetype, *lineweight* is the lineweight, *layer* is the layername, and *arrow* is set to 1 if there is an arrowhead on end A, and is set to 2 if there is an arrowhead on end B. *arrow* is only valid if the line type is dcARROW.

The variable for *layer* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetLineData. See the example below.

Example:

```
Sub Main
'This sub finds all red lines
Dim ot As Long
Dim xa As Double
Dim ya As Double
Dim xb As Double
Dim yb As Double
Dim color As Long
Dim lt As Long
Dim lw As Long
Dim layer As String
Dim arrow As Long
ot = dcGetFirstObject("")
While ot <> dcNone
    If ot = dcLine Then
        layer = Space(255)
        dcGetLineData xa, ya, xb, yb, color, lt, lw, layer, arrow
        If color = dcRED Then
            MsgBox Format(xa)+", "+Format(ya)+ _ "    "+Format(xb)+ _
                ", "+Format(yb)
        End If
    End If
    ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetLineParms

dcGetLineParms(*color, linetype, linewidth*)

This function gets all of the current parameters used to create a line.

See dcSetLineParms for a description of *color*, *linetype* or *linewidth*.

Example:

```
Sub Main
Dim color As Long
Dim linetype As Long
Dim linewidth As Long
dcGetLineParms color, linetype, linewidth
MsgBox "Color = " + Format(color)
MsgBox "LineType = " + Format(linetype)
MsgBox "LineWeight = " + Format(linewidth)
End Sub
```

Related Topics: dcSetLineParms, dcSaveAllParms, dcRestoreAllParms

dcGetNextGroupName

dcGetNextGroupName(*gname*)

This function gets the next group name, and returns it in the *gname* variable.

Use dcGetFirstGroupName to get the first the group name.

Return Value: True if a group name is returned, and False if there are no more groups.

Example:

```
Sub Main
Dim gname As String
gname = Space(255)
If dcGetFirstGroupName (gname) Then
  MsgBox gname
  gname = Space(255)
  While dcGetNextGroupName (gname)
    MsgBox gname
    gname = Space(255)
  Wend
Else
  MsgBox "Can't find first group name"
End If
End Sub
```

Related Topics: dcGetFirstGroupName, dcOpenGroupWindow

dcGetNextObject

dcGetNextObject

This function gets the next object. Use this function after calling `dcGetFirstObject`. After this function is called use the data retrieval functions to get the actual data on a drawing object.

Return Value: Is the type of object found, or `dcNone` if nothing is found. The following is a list of valid object types: `dcPoint`, `dcLine`, `dcSpline`, `dcCircle`, `dcShape`, `dcText`, `dcDimension`, `dcGroup`, `dcPicture`.

Example: See the example in `dcGetFirstObject`

Related Topics: `dcGetFirstObject`

dcGetNextSelectedObject

dcGetNextSelectedObject

This function gets the next selected object. Use this function after calling `dcGetFirstSelectedObject`. After this function is called use the data retrieval functions to get the actual data on a drawing object.

Return Value: Is the type of object found, or `dcNone` if nothing is found. The following is a list of valid object types: `dcPoint`, `dcLine`, `dcSpline`, `dcCircle`, `dcShape`, `dcText`, `dcDimension`, `dcGroup`, `dcPicture`.

Example: See the example in `dcGetFirstSelectedObject`

Related Topics: `dcGetFirstSelectedObject`

dcGetNumberofDeltaCadDwgs

dcGetNumberofDeltaCadDwgs

This function determines number of open DeltaCad drawings.

Return Value: Is the number of open DeltaCad drawings.

Example:

```
Sub Main
Dim j As Long
Dim n As Long
Dim s As String
n = dcGetNumberofDeltaCadDwgs
MsgBox "Number of open drawings = " + Format(n)
For j=1 To n
s = Space(255)
dcGetDeltaCadDwgName j, s
MsgBox s
Next j
End Sub
```

Related Topics: `dcGetDeltaCadDwgName`

dcGetNumberofLayers

dcGetNumberofLayers

This function determines number of layers in the current drawing.

Return Value: Is the number of layers.

Example:

```
Sub Main
Dim n As Long
Dim j As Long
Dim s As String
n = dcGetNumberOfLayers
For j = 1 To n
s = space(255)
dcGetLayerName j, s
MsgBox "Layer name is - " + s
Next j
End Sub
```

Related Topics: dcGetLayerName

dcGetNumberOfPrintRegions

dcGetNumberOfPrintRegions

This function determines number of print regions in the current drawing.

Return Value: Is the number of print regions.

Example:

```
Sub Main
Dim n As Long
Dim j As Long
Dim s As String
n = dcGetNumberOfPrintRegions
For j = 1 To n
s = space(255)
dcGetPrintRegionName j, s
MsgBox "Print Region name is - " + s
Next j
End Sub
```

Related Topics: dcGetPrintRegionName

dcGetObjectAt

dcGetObjectAt(*x*, *y*, *howclose*)

This function gets the object at the *x,y* coordinates given. The *x,y* coordinates must be within the distance given by *howclose*. After this function is called use the data retrieval functions to get the actual data on a drawing object.

Return Value: Is the type of object found, or dcNone if nothing is found. The following is a list of valid object types: dcPoint, dcLine, dcSpline, dcCircle, dcShape, dcText, dcDimension, dcGroup, dcPicture.

Example:

```
Sub Main
Dim ot As Long
Dim xa As Double
Dim ya As Double
Dim xb As Double
```

```

Dim yb As Double
Dim color As Long
Dim lt As Long
Dim lw As Long
Dim layer As String
Dim arrow As Long
ot = dcGetObjectAt(3, -5, 0.01)
If ot = dcLine Then
    layer = Space(255)
    dcGetLineData xa, ya, xb, yb, color, lt, lw, layer, arrow
    If color = dcRED Then
        dcEraseCurrentObject
    Else
        dcSelectCurrentObject
    End If
End If
End Sub

```

Related Topics: dcSelectCurrentObject, dcEraseCurrentObject, dcGetLineData

dcGetObjectBox

dcGetObjectBox *ll, bb, rr, tt*

This function gets left, bottom, right, and top coordinates of a box that bounds the object. They are returned in the *ll*, *bb*, *rr*, and *tt* variables.

Example:

```

Sub Main
Dim ot As Long
Dim ll As Double
Dim rr As Double
Dim tt As Double
Dim bb As Double
ot = dcGetFirstObject("")
While ot <> dcNone
    dcGetObjectBox ll, bb, rr, tt
    MsgBox "Left   = "+Format(ll)+ Chr(13)+ _
           "Right  = "+Format(rr)+ Chr(13)+ _
           "Top    = "+Format(tt)+ Chr(13)+ _
           "Bottom = "+Format(bb)
    ot = dcGetNextObject
Wend
End Sub

```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetObjectLayer

dcGetObjectLayer *lname*

This function gets the layer name of the current object, and returns it in the *lname* variable.

Example:

```

Sub Main

```

```
Dim ot As Long
Dim lname As String
ot = dcGetFirstObject("")
While ot <> dcNone
    lname = space(255)
    dcGetObjectLayer lname
    MsgBox lname
    ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetPictureData

dcGetPictureData(*pictype, x, y, w, h, orig_w, orig_h, layer*)

This function gets all of the data associated with the current picture object. *pictype* is the type of picture. Currently the only picture type supported is bitmaps, so *pictype* will always be equal to dcPicBMP. *x* and *y* are the coordinates of the insertion point. *w* and *h* are the width and height. *orig_w* and *orig_h* are the original width and height of the picture when it was inserted into DeltaCad. *layer* is the layername.

The variable for *layer* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetPictureData. See the example below.

Example:

```
Sub Main
'This sub copies the first picture to a file
Dim ot As Long
Dim pictype As Long
Dim x As Double
Dim y As Double
Dim w As Double
Dim h As Double
Dim orig_w As Double
Dim orig_h As Double
Dim layer As String
ot = dcGetFirstObject("")
While ot <> dcNone
    If ot = dcPicture Then
        layer = Space(255)
        dcGetPictureData pictype, x, y, w, h, orig_w, orig_h, layer
        If pictype = dcPicBMP Then
            dcCopyCurrentPictureToFile "c:\program
files\deltacad\pic.bmp"
        End If
        MsgBox layer
        Exit Do
    End If
    ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject, dcCopyCurrentPictureToFile

dcGetPointData

dcGetPointData(*x*, *y*, *color*, *layer*, *perm*)

This function gets all of the data associated with the current point object. *x,y* is the location of the point, *color* is the point color, *layer* is the layername, and *perm* is set to True if it is a permanent point, otherwise it is set to False.

The variable for *layer* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetPointData. See the example below.

Example:

```
Sub Main
'This sub finds all points
Dim ot As Long
Dim x As Double
Dim y As Double
Dim color As Long
Dim layer As String
Dim perm As Boolean
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcPoint Then
    layer = Space(255)
    dcGetPointData x, y, color, layer, perm
    MsgBox Format(x)+", "+Format(y)
  End If
  ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetPointParms

dcGetPointParms(*color*, *permtemp*)

This function gets all of the current parameters used to create a point.

See dcSetPointParms for a description of *color* and *pointperm*.

Example:

```
Sub Main
Dim color As Long
Dim pointperm As Boolean
dcGetPointParms color, pointperm
If(pointperm) Then
  MsgBox "Create Permanent point in color " + Format(color)
Else
  MsgBox "Create Temporary point in color " + Format(color)
```

```
End If
End Sub
```

Related Topics: dcSetPointParms, dcSaveAllParms, dcRestoreAllParms

dcGetPrintRegionData

dcGetPrintRegionData(*prnumer*, *prscale*, *prleft*, *prbottom*, *prright*, *prtop*, *orient*)

This function gets the print region data for the print region with the name given by *prname*. The print scale is returned in the variable *prscale*. The print area is returned in the variables *prleft*, *prbottom*, *prright*, and *prtop*. The printer paper orientation is returned in *orient*. *orient* may be either dcPortrait or dcLandscape

Example:

```
Sub Main
Dim scale As Double
Dim prleft As Double
Dim prbottom As Double
Dim prright As Double
Dim prtop As Double
Dim orient As Long
dcAddPrintRegion "nregion", 2.5, 1.2, 3.4, 5.6, 7.8, dcPortrait
dcGetPrintRegionData "nregion", scale, prleft, prbottom, _
    prright, prtop, orient
MsgBox "Scale = " + Format(scale)
MsgBox "Left = " + Format(prleft)
MsgBox "Bottom = " + Format(prbottom)
MsgBox "Right = " + Format(prright)
MsgBox "Top = " + Format(prtop)
If orient = dcPortrait Then
    MsgBox "Portrait"
Else
    MsgBox "Landscape"
End If
End Sub
```

Related Topics: dcGetPrintRegionName

dcGetPrintRegionName

dcGetPrintRegionName(*prnumer*, *prname*)

This function gets the *prname* for any *prnumber*.

The variable for *prname* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetLayerName. See the example below.

Example:

```
Sub Main
Dim n As Long
Dim j As Long
Dim s As String
n = dcGetNumberOfPrintRegions
For j = 1 To n
```

```
s = space(255)
dcGetPrintRegionName j, s
MsgBox "Print Region name is - " + s
Next j
End Sub
```

Related Topics: dcGetNumberOfPrintRegions

dcGetSelCopyMode

dcGetSelCopyMode(*mode*)

This function gets the Select functions copy mode. The mode is set with dcSetSelCopyMode.

Return Value: True if the select copy mode is on, otherwise False.

Example:

```
Sub Main
Dim m As Boolean
m = dcGetSelCopyMode
If m Then
MsgBox "Select copy mode is on"
Else
MsgBox "Select copy mode is off"
End If
End Sub
```

Related Topics: dcGetSelCopyMode

dcGetShapeData

dcGetShapeData(*vertex, num, color, linetype, lineweight, layer*)

This function gets all of the data associated with the current shape object. *vertex* is an array where every other number will be the X and Y coordinate of the shape point. *num* is a variable to receive the number of shape points. *num* must be set to the maximum number of points that the *vertex* array can hold before calling this function (this will be half the size of the array). *color* is the shape color, *linetype* is the linetype, *lineweight* is the lineweight, and *layer* is the layername.

The variables for *vertex* and *layer* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetShapeData. See the example below.

Example:

```
Sub Main
'This sub finds all shape points
Dim ot As Long
Dim vert(64) As Double
Dim num As Long
Dim j As Long
Dim color As Long
Dim lt As Long
Dim lw As Long
```

```
Dim layer As String
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcShape Then
    layer = Space(255)
    num = 32
    dcGetShapeData vert(1), num, color, lt, lw, layer
    For j=1 to num
      MsgBox Format(vert(j*2-1))+", "+Format(vert(j*2))
    Next j
  End If
  ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetShapesParms

dcGetShapesParms(*color, linetype, lineweight*)

This function gets all of the current parameters used to create a shape.

See dcSetShapesParms for a description of *color*, *linetype* or *lineweight*.

Example:

```
Sub Main
Dim color As Long
Dim linetype As Long
Dim lineweight As Long
dcGetShapesParms color, linetype, lineweight
MsgBox "Color = " + Format(color)
MsgBox "LineType = " + Format(linetype)
MsgBox "LineWeight = " + Format(lineweight)
End Sub
```

Related Topics: dcSetShapesParms, dcSaveAllParms, dcRestoreAllParms

dcGetSplineData

dcGetSplineData(*vertex, num, color, linetype, lineweight, layer, open*)

This function gets all of the data associated with the current spline object. *vertex* is an array where every other number will be the X and Y coordinate of the spline point. *num* is a variable to receive the number of spline points. *num* must be set to the maximum number of points that the *vertex* array can hold before calling this function (this will be half the size of the array). *color* is the spline color, *linetype* is the linetype, *lineweight* is the lineweight, *layer* is the layername, and *open* is set to True for an open spline and to False for a closed spline.

The variables for *vertex* and *layer* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling dcGetSplineData. See the example below.

Example:

```

Sub Main
'This sub finds all spline points
Dim ot As Long
Dim vert(64) As Double
Dim num As Long
Dim j As Long
Dim color As Long
Dim lt As Long
Dim lw As Long
Dim layer As String
Dim opensp As Boolean
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcSpline Then
    layer = Space(255)
    num = 32
    dcGetSplineData vert(1), num, color, lt, lw, layer, opensp
    For j=1 to num
      MsgBox Format(vert(j*2-1))+", "+Format(vert(j*2))
    Next j
  End If
  ot = dcGetNextObject
Wend
End Sub

```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcGetSplineParms

dcGetSplineParms(*color, linetype, lineweight*)

This function gets all of the current parameters used to create a spline.

See dcSetSplineParms for a description of *color, linetype* or *lineweight*.

Example:

```

Sub Main
Dim color As Long
Dim linetype As Long
Dim lineweight As Long
dcGetSplineParms color, linetype, lineweight
MsgBox "Color = " + Format(color)
MsgBox "LineType = " + Format(linetype)
MsgBox "LineWeight = " + Format(lineweight)
End Sub

```

Related Topics: dcSetSplineParms, dcSaveAllParms, dcRestoreAllParms

dcGetTextData

dcGetTextData(*x, y, text, xchar, xline, angle, height, width, font, style, flags, color, layer*)

This function gets all of the data associated with the current text object. *x,y* are the coordinates of the text insertion point. *text* is the actual text displayed.

xchar and *xline* are extra character or line spacing. *angle* is the text angle in degrees. *height* is the text height in points. *width* is the width of the text. *width* is only used for multi-line text. *font* and *style* are the font and style of the typeface used. *flags* indicate the location of the insertion point and the justification. See `dcSetTextParms` for a description of flags. *color* is the text color, and *layer* is the layername.

The variables for *text*, *font*, *style* and *layer* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling `dcGetTextData`. See the example below.

Example:

```
Sub Main
'This sub finds all text
Dim ot As Long
Dim x As Double
Dim y As Double
Dim text As String
Dim xchar As Double
Dim xline As Double
Dim angle As Double
Dim height As Double
Dim width As Double
Dim font As String
Dim style As String
Dim flags As Long
Dim color As Long
Dim layer As String
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcText Then
    text = Space(255)
    font = Space(255)
    style = Space(255)
    layer = Space(255)
    dcGetTextData x, y, text, xchar, xline, angle, height, _
      width, font, style, flags, color, layer
    MsgBox text
  End If
  ot = dcGetNextObject
Wend
End Sub
```

Related Topics: `dcGetFirstObject`, `dcGetNextObject`, `dcGetObjectAt`, `dcGetFirstSelectedObject`, `dcGetNextSelectedObject`

dcGetTextParms

dcGetTextParms(*color*, *fontname*, *fontstyle*, *angle*, *height*, *flags*, *excharspacing*, *exlinespacing*)

This function gets all of the current parameters used to create a text object.

The variable for *fontname* and *fontstyle* must be large enough to receive the result, or an error will occur. Be sure to make the size of the variable large enough before calling `dcGetTextParms`. See the example below.

See `dcSetTextParms` for a description of *color*, *fontname*, *fontstyle*, *angle*, *height*, *flags*, *excharspacing*, *exlinespacing*.

Example:

```
Sub Main
Dim color As Long
Dim fontname As String
Dim fontstyle As String
Dim angle As Double
Dim height As Double
Dim flags As Long
Dim excharspacing As Double
Dim exlinespacing As Double
fontname = space(32)
fontstyle = space(32)
dcGetTextParms color, fontname, fontstyle, angle, height, _
    flags, excharspacing, exlinespacing
MsgBox "color = " + Format(color)
MsgBox "fontname = "+fontname
MsgBox "fontstyle = "+fontstyle
MsgBox "angle = " + Format(angle)
MsgBox "height = " + Format(height)
MsgBox "flags = " + Format(flags)
MsgBox "excharspacing = " + Format(excharspacing)
MsgBox "exlinespacing = " + Format(exlinespacing)
```

Related Topics: `dcSetTextParms`, `dcSaveAllParms`, `dcRestoreAllParms`

dcGetViewScaleandRect

dcGetViewScaleandRect(*scale*, *left*, *bottom*, *right*, *top*)

This function gets the view scale and the displayed rectangle in drawing units.

Example:

```
Sub Main
Dim scale As Double
Dim vLeft As Double
Dim vBottom As Double
Dim vRight As Double
Dim vTop As Double
dcGetViewScaleandRect scale, vLeft, vBottom, vRight, vTop
MsgBox "Scale = "+Format(scale)
MsgBox "Left = "+Format(vLeft)
MsgBox "Bottom = "+Format(vBottom)
MsgBox "Right = "+Format(vRight)
MsgBox "Top = "+Format(vTop)
End Sub
```

Related Topics: `dcSetViewScale`, `dcViewCenteredOnXY`

dcGroupSelObjs

dcGroupSelObjs(*groupname*)

This function groups selected objects into a new group with the name given by *groupname*.

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcSetSelectBase 0.75, 0.75
dcGroupSelObjs "newgroup"
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll, dcSetSelectBase

dcIsCurrentObjectSelected

dcIsCurrentObjectSelected

This function determines if the current object is selected. The current object is set in dcGetFirstObject, dcGetNextObject, dcGetFirstSelectedObject, dcGetNextSelectedObject, or dcGetObjectAt.

Return Value: True if selected, otherwise False.

Example:

```
Sub Main
' This sub unselects all lines
Dim ot As Long
ot = dcGetFirstObject("")
While ot <> dcNone
  If ot = dcLine Then
    If dcIsCurrentObjectSelected Then
      dcUnselectCurrentObject
    End If
  End If
  ot = dcGetNextObject
Wend
End Sub
```

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject, dcUnselectCurrentObject

dcIsLayerOn

dcIsLayerOn(*layername*)

This function determines if the layer given by *layername* is on or off.

Return Value: True if the layer is on, otherwise False.

Example:

```
Sub Main
dcAddLayer "outline"
If dcIsLayerOn("outline") Then
```

```
MsgBox "Layer is On"  
Else  
MsgBox "Layer is Off"  
End If  
End Sub
```

Related Topics: dcTurnLayerOn, dcTurnLayerOff, dcGetCurrentLayer

dcIsPrintRegionLayerOn

dcIsPrintRegionLayerOn(*prname*, *layername*)

This function determines if the layer given by *layername* in the print region given by *prname* is on or off.

Return Value: True if the layer is on, otherwise False.

Example:

```
Sub Main  
dcAddPrintRegion "nregion", 1.0, 0, 0, 0, 0, dcPortrait  
dcAddLayer "outline"  
If dcIsPrintRegionLayerOn("nregion", "outline") Then  
MsgBox "Layer is On"  
Else  
MsgBox "Layer is Off"  
End If  
End Sub
```

Related Topics: dcTurnPrintRegionLayerOn, dcTurnPrintRegionLayerOff, dcGetCurrentPrintRegion

dcIsSelected

dcIsSelected

This function determines if there are any selected objects in the current window.

Return Value: True if there are any selected objects in the current window, otherwise False.

Example:

```
Sub Main  
Dim m As Boolean  
m = dcIsSelected  
If m Then  
MsgBox "There are selected objects"  
Else  
MsgBox "There are not any selected objects"  
End If  
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll

dcMirrorSelObjs

dcMirrorSelObjs(*xa*, *ya*, *xb*, *yb*)

This function mirrors all selected objects using the line defined by the endpoints *xa/ya* and *xb/yb*.

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcMirrorSelObjs 0, 0, 3, 2
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll, dcSetSelCopyMode

dcMoveSelObjs

dcMoveSelObjs(*offsetx*,*offsety*)

This function moves all selected objects by the amount specified by *offsetx*, and *offsety*.

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcMoveSelObjs 1.2, 3.4
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll, dcSetSelectBase, dcSetSelCopyMode

dcNew

dcNew(*filename*)

This function creates a new DeltaCad drawing based on an existing drawing name given by *filename*. If an empty string is passed to this function the default.dc drawing in the DeltaCad new directory is used.

Example:

```
Sub Main
dcNew ""
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcSave
dcCloseWithoutSaving
End Sub
```

Related Topics: dcCloseWithoutSaving, dcSave, dcSaveAs, dcClose

dcNewSymbolFile

dcNewSymbolFile(*symbolfilename*, *sectionname*, *symbolname*)

This function adds the DeltaCad object on the clipboard to a new symbol file given by *symbolfilename*. It is assigned the section name given by *sectionname*, and the symbol name given by *symbolname*.

Example :

```
Sub Main
dcNewSymbolFile "c:\deltacad\symbols\new.sym", "def", "newsym"
End Sub
```

Related Topics: dcAddSymbol, dcDeleteSymbol, dcCopySymboltoClipboard

dcOpen

dcOpen(filename)

This function opens the DeltaCad file with the name given by *filename*.

Example :

```
Sub Main
dcOpen "C:\DeltaCad\lines.dc"
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcSave
dcClose
End Sub
```

Related Topics: dcSave, dcSaveAs, dcClose, dcCloseWithoutSaving

dcOpenGroupWindow

dcOpenGroupWindow (groupname)

This function opens a window for the group given by *groupname*.

Return Value: True if the group exists, otherwise False.

Example :

```
Sub Main
If dcOpenGroupWindow ("Star") Then
  dcCreateLine 1, 0, 1, 3
  dcCreateLine 2, 0, 2, 3
  dcCreateLine 0, 1, 3, 1
  dcCreateLine 0, 2, 3, 2
  dcClose
Else
  MsgBox "Can't open group window"
End If
End Sub
```

Related Topics: dcGetFirstGroupName, dcGetNextGroupName

dcPasteClipboardData

dcPasteClipboardData(x, y)

This function inserts the DeltaCad object on the clipboard into the current drawing at the location given by *x* and *y*.

Example :

```
Sub Main
dcCopySymboltoClipboard "c:\deltacad\symbols\new.sym", _
  "def", "newsym"
```

```
dcPasteClipboardData 2, 3
End Sub
```

Related Topics: dcCopySelObjstoClipboard, dcPasteSpecialClipboardData

dcPasteSpecialClipboardData

dcPasteSpecialClipboardData(*x*, *y*, *datatype*)

This function inserts the data on the clipboard into the current drawing at the location given by *x* and *y*, in the format given by *datatype*. *datatype* can be one of the following: dcDeltaCadFormat, dcTextFormat, or dcBitmapFormat.

Example:

```
Sub Main
dcPasteSpecialClipboardData 2, 3, dcBitmapFormat
End Sub
```

Related Topics: dcCopySelObjstoClipboard, dcPasteClipboardData

dcPrint

dcPrint

This function prints the currently active DeltaCad drawing window.

Example:

```
Sub Main
dcPrint
End Sub
```

Related Topics: dcPrintPreview

dcPrintPreview

dcPrintPreview

This function the print preview window for the currently active DeltaCad drawing.

Example:

```
Sub Main
dcPrintPreview
MsgBox "Press ENTER to end"
dcEndPrintPreview
End Sub
```

Related Topics: dcEndPrintPreview

dcReadInput

dcReadInput(*text*)

This function reads the *text* variable and interprets it the same way as on the DeltaCad data input line (on the Status bar).

Return Value: The value of the *text* variable is returned.

Example:
Sub Main
Dim inputtext As String
inputtext = "sqrt(2.0)"
MsgBox dcReadInput(inputtext)
End Sub

dcRenameLayer

dcRenameLayer(newlayername, oldlayername)

This function renames layer *oldlayername* to *newlayername*.

Example:
Sub Main
dcAddLayer "outline"
dcRenameLayer "object", "outline"
End Sub

Related Topics: dcAddLayer, dcDeleteLayer, dcDoesLayerExist

dcRenamePrintRegion

dcRenamePrintRegion(newprname, oldprname)

This function renames print region *oldprname* to *newprname*.

Example:
Sub Main
dcAddPrintRegion "outline", 1.0, 0, 0, 0, 0, dcPortrait
dcRenamePrintRegion "object", "outline"
End Sub

Related Topics: dcAddPrintRegion, dcDeletePrintRegion

dcRestoreAllParms

dcRestoreAllParms

This function restores all parms that were saved by a previous call to dcSaveAllParms.

Example:
Sub Main
dcSaveAllParms
dcSetLineParms dcRED, dcSOLID, dcNORMAL
dcCreateLine 1, 0, 1, 3
dcCreateLine 2, 0, 2, 3
dcCreateLine 0, 1, 3, 1
dcCreateLine 0, 2, 3, 2
dcRestoreAllParms
dcViewAll
End Sub

Related Topics: dcSaveAllParms

dcRestoreBookmark

dcRestoreBookmark(*num*)

This function restores the DeltaCad view to the x, y, and scale set for bookmark number *num*.

Example:

```
Sub Main
dcSetBookmark 2, 50, 75, 2.5
dcRestoreBookmark 2
End Sub
```

Related Topics: dcGetBookmark, dcSetBookmark

dcRotateSelObjs

dcRotateSelObjs(*angle*)

This function rotates selected objects by the *angle* given. The *angle* is in degrees. A positive angle goes counterclockwise.

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcSetSelectBase 0.25, 0.25
dcRotateSelObjs 90
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll, dcSetSelectBase, dcSetSelCopyMode

dcSave

dcSave

This function saves the currently open DeltaCad file.

Example:

```
Sub Main
dcOpen "C:\DeltaCad\lines.dcd"
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcSave
dcClose
End Sub
```

Related Topics: dcNew, dcOpen, dcClose, dcCloseWithoutSaving, dcSaveAs

dcSaveAllParms

dcSaveAllParms

This function saves all of the parms for all object types. It can be very convenient to save all parms at the beginning of a program, and then restore them at the end of a program. Use dcRestoreAllParms to restore the saved parms.

Example:

```
Sub Main
dcSaveAllParms
dcSetLineParms dcRED, dcSOLID, dcNORMAL
dcCreateLine 1, 0, 1, 3
dcCreateLine 2, 0, 2, 3
dcCreateLine 0, 1, 3, 1
dcCreateLine 0, 2, 3, 2
dcRestoreAllParms
dcViewAll
End Sub
```

Related Topics: dcRestoreAllParms

dcSaveAs

dcSaveAs(*filename*)

This function saves the currently open DeltaCad file with the filename given by *filename*.

Example:

```
Sub Main
dcOpen "C:\DeltaCad\lines.dc"
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcSaveAs "C:\DeltaCad\newlines.dc"
dcClose
End Sub
```

Related Topics: dcNew, dcOpen, dcClose, dcCloseWithoutSaving, dcSave

dcScaleSelObjs

dcScaleSelObjs(*scalefactor*)

This function scales selected objects by the *scalefactor* given. If the *scalefactor* is greater than 1.0 it increases the size. If it is less than 1.0 it decreases the size.

Example:

```
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcSetSelectBase 0.75, 0.75
dcScaleSelObjs 2.5
dcUnselectAll
End Sub
```

Related Topics: dcSelectAll, dcUnselectAll, dcSetSelectBase, dcSetSelCopyMode

dcSelectAll

dcSelectAll

This function selects all objects.

Return Value: True if successful, otherwise False.

Example:
Sub Main
dcSelectAll
dcMoveSelObjs 1.2, 3.4
dcUnselectAll
End Sub

Related Topics: dcUnselectAll

dcSelectCurrentObject

dcSelectCurrentObject

This function selects the current object. The current object is set in dcGetFirstObject, dcGetNextObject, dcGetFirstSelectedObject, dcGetNextSelectedObject, or dcGetObjectAt

Example:
Sub Main
'This sub selects all lines
Dim ot As Long
ot = dcGetFirstObject("")
While ot <> dcNone
 If ot = dcLine Then
 dcSelectCurrentObject
 End If
 ot = dcGetNextObject
Wend
End Sub

Related Topics: dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject, dcUnselectCurrentObject, dcIsCurrentObjectSelected

dcSelectObjInBox

dcSelectObjInBox(*left, bottom, right, top*)

This function selects all objects inside the box defined by *left, bottom, right, top*.

Return Value: True if successful, otherwise False.

Example:
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcMoveSelObjs 1.2, 3.4
dcUnselectAll
End Sub

Related Topics: dcSelectAll, dcUnselectAll

dcSetActiveWindow

dcSetActiveWindow(*filename*)

This function sets the active DeltaCad window to the DeltaCad drawing with the filename given by *filename*.

Example:

```

Sub Main
dcOpen "C:\DeltaCad\lines.dc"
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcOpen "C:\DeltaCad\points.dc"
dcCreatePoint 0, 0
dcCreatePoint 0, 1
dcSetActiveWindow "C:\DeltaCad\lines.dc"
End Sub

```

Related Topics: dcGetActiveWindow

dcSetBookmark

dcSetBookmark(*num, x,y, scale*)

This function sets the bookmark *x, y*, and *scale* for bookmark number *num*.

Example:

```

Sub Main
Dim j As Long
Dim x As Double
Dim y As Double
Dim scale As Double
dcSetBookmark 1, 0, 0, 2.5
dcSetBookmark 2, 0, 0, 5
dcSetBookmark 3, 0, 0, 10
dcSetBookmark 4, 10, 20, 2.5
dcSetBookmark 5, 75, 50, 5
For j=1 to 5
MsgBox "Bookmark"+Format(j)+" = "+Format(x)+" , "+Format(y)+" , _
      "+Format(scale)
Next j
End Sub

```

Related Topics: dcGetBookmark, dcRestoreBookmark

dcSetCircleParms

dcSetCircleParms(*color, linetype, lineweight*)

This function sets all of the parameters used to create a circle.

color is the DeltaCad color number. The color number starts with zero, and increases in the same order that the colors are displayed in the color selection box in DeltaCad (i.e. 0 is black, 1 is red, 2 is green, 3 is yellow, etc.). You can also use the following pre-defined color values: dcBLACK, dcRED, dcGREEN, dcYELLOW, dcBLUE, dcPURPLE, dcBROWN, dcDARKGRAY, dcLIGHTGRAY, dcLIGHTBLUE, dcDARKGREEN, dcGREENYELLOW, dcDARKBLUE, dcDARKPURPLE, dcGRAYBLUE, dcWHITE

linetype is the DeltaCad line type. This may be set to one of the following: dcSOLID, dcHIDDEN, dcCENTER, dcPHANTOM, dcCUTTING, dcBREAK, dcSTITCH, dcARROW, dcTWOARROW, dcFILL, dcHATCH, dcHATCHNOOUTLINE. If using dcHATCH or dcHATCHNOOUTLINE you

may add numbers (0 to 39) to the value to get the various hatch types. See the example below.

lineweight is the DeltaCad line weight. It can be one of the following values: dcTHIN, dcNORMAL, dcTHICK, dcHEAVY or dcWIDE

```
Example:  
Sub Main  
dcSetCircleParms dcGREEN, dcFILL, dcNORMAL  
dcSetCircleParms dcBLUE, dcHATCH+2, dcTHIN  
End Sub
```

Related Topics: dcGetCircleParms, dcSaveAllParms, dcRestoreAllParms

dcSetCurrentLayer

dcSetCurrentLayer(*layername*)

This function sets the current layer to the layer with the name given by *layername*.

```
Example:  
Sub Main  
dcAddLayer "outline"  
dcSetCurrentLayer "outline"  
End Sub
```

Related Topics: dcGetCurrentLayer

dcSetCurrentPrintRegion

dcSetCurrentPrintRegion(*prname*)

This function sets the current print region to the print region with the name given by *prname*.

```
Example:  
Sub Main  
dcAddPrintRegion "outline", 1.0, 0, 0, 0, 0, dcPortrait  
dcSetCurrentPrintRegion "outline"  
End Sub
```

Related Topics: dcGetCurrentPrintRegion

dcSetDimParms

dcSetDimParms(*color, fontname, fontstyle, height, InOut, AlwaysHorz*)

This function sets all of the parameters used to create a dimension object.

color is the DeltaCad color number. The color number starts with zero, and increases in the same order that the colors are displayed in the color selection box in DeltaCad (i.e. 0 is black, 1 is red, 2 is green, 3 is yellow, etc.). You can also use the following pre-defined color values: dcBLACK, dcRED, dcGREEN, dcYELLOW, dcBLUE, dcPURPLE, dcBROWN, dcDARKGRAY,

dcLIGHTGRAY, dcLIGHTBLUE, dcDARKGREEN, dcGREENYELLOW,
dcDARKBLUE, dcDARKPURPLE, dcGRAYBLUE, dcWHITE

fontname and *fontstyle* are the name and style of the font to be used.

height is the point size of the font.

InOut - Set to True to make the dimension arrows point in. Set to False to make the dimension arrows point out

AlwaysHorz - Set to True to make the dimension text always horizontal. Set to False to make the dimension text parallel to the dimension.

Example:

```
Sub Main
dcSetDimParms dcBLUE, "Arial", "Regular", 16, True, True
End Sub
```

Related Topics: dcGetDimParms, dcSaveAllParms, dcRestoreAllParms

dcSetDrawingData

dcSetDrawingData(*index*, *value*)

This function sets the drawing data item associated with the given *index* to the *value* given.

When setting dcScalewithViewScale or dcAtLeastOnePixBigger, set to 1.0 for True and 0.0 for False.

The following are all of the valid *index* values.

- dcDimGap
- dcDimExtEx
- dcDimTxGap
- dcDimVerLd
- dcDimShortArrowLen
- dcPhantomDash
- dcPhantomGap
- dcCenterDash
- dcCenterGap
- dcHiddenDash
- dcHiddenGap
- dcCuttingDash
- dcCuttingGap
- dcStitchDash
- dcStitchGap
- dcBreakGap
- dcBreakWidth
- dcMinFixedLen
- dcMaxVarLen
- dcArrowLen
- dcArrowWL

dcHatchSpacing
dcLineThin
dcLineNormal
dcLineThick
dcLineHeavy
dcLineWide
dcScalewithViewScale
dcAtLeastOnePixBigger

Example:

```
Dim v As Double
dcSetDrawingData dcDimGap, 0.0625
dcSetDrawingData dcPhantomDash, 0.15
dcSetDrawingData dcScalewithViewScale, 0.0
dcGetDrawingData dcDimGap, v : MsgBox "Dim Gap = "+Format(v)
dcGetDrawingData dcScalewithViewScale, v
If v = 1.0 Then
MsgBox "Scale with view scale"
Else
MsgBox "Do not scale with view scale"
End If
```

Related Topics: dcGetDrawingData

dcSetDrawingScale

dcSetDrawingScale(*scale*)

This function sets the drawing scale to the value defined by *scale*.

Example:

```
Sub Main
dcSetDrawingScale 96
End Sub
```

Related Topics: dcGetDrawingScale

dcSetDrawingWindowMode

dcSetDrawingWindowMode(*mode*)

This function sets drawing window mode specified by *mode*. *mode* may be any of the following.

mode

dcMaximizeWin - This maximizes the current window.

dcMinimizeWin - This minimizes the current window.

dcNormalWin - This returns the current window to its normal size.

Example:

```
Sub Main
dcSetDrawingWindowMode dcMaximizeWin
End Sub
```

Related Topics: dcGetDrawingWindowHandle

dcSetInitialSizeandPos

dcSetInitialSizeandPos(*x,y, scale*)

This function sets the initial size and position used when a DeltaCad drawing is opened. If the *scale* is set to 0.0 the drawing will be zoomed to fill the window when the drawing is opened.

Example:

```
Sub Main
dcSetInitialSizeandPos 0, 0, 10
End Sub
```

Related Topics: dcGetInitialSizeandPos

dcSetLineParms

dcSetLineParms(*color, linetype, linewidth*)

This function sets all of the parameters used to create a line.

color is the DeltaCad color number. The color number starts with zero, and increases in the same order that the colors are displayed in the color selection box in DeltaCad (i.e. 0 is black, 1 is red, 2 is green, 3 is yellow, etc.). You can also use the following pre-defined color values: dcBLACK, dcRED, dcGREEN, dcYELLOW, dcBLUE, dcPURPLE, dcBROWN, dcDARKGRAY, dcLIGHTGRAY, dcLIGHTBLUE, dcDARKGREEN, dcGREENYELLOW, dcDARKBLUE, dcDARKPURPLE, dcGRAYBLUE, dcWHITE

linetype is the DeltaCad line type. This may be set to one of the following: dcSOLID, dcHIDDEN, dcCENTER, dcPHANTOM, dcCUTTING, dcBREAK, dcSTITCH, dcARROW, dcTWOARROW. (NOTE: Filled or Hatched lines are not valid for lines).

linewidth is the DeltaCad line weight. It can be one of the following values: dcTHIN, dcNORMAL, dcTHICK, dcHEAVY or dcWIDE

Example:

```
Sub Main
dcSetLineParms dcRED, dcSOLID, dcWIDE
End Sub
```

Related Topics: dcGetLineParms, dcSaveAllParms, dcRestoreAllParms

dcSetPointParms

dcSetPointParms(*color, permtemp*)

This function sets all of the parameters used to create a point.

color is the DeltaCad color number. The color number starts with zero, and increases in the same order that the colors are displayed in the color selection box in DeltaCad (i.e. 0 is black, 1 is red, 2 is green, 3 is yellow, etc.). You can also use the following pre-defined color values: dcBLACK, dcRED, dcGREEN, dcYELLOW, dcBLUE, dcPURPLE, dcBROWN, dcDARKGRAY,

dcLIGHTGRAY, dcLIGHTBLUE, dcDARKGREEN, dcGREENYELLOW,
dcDARKBLUE, dcDARKPURPLE, dcGRAYBLUE, dcWHITE

Set *permtemp* to True to make a permanent point, or set it to False to make a temporary point.

Example:
Sub Main
dcSetPointParms dcBLUE, False
End Sub

Related Topics: dcGetPointParms, dcSaveAllParms, dcRestoreAllParms

dcSetSelCopyMode

dcSetSelCopyMode(*mode*)

This function sets the Select functions copy mode. If *mode* is set to True the select functions (such as dcMoveSelObjs) will leave a copy behind when performing their functions.

Example:
Sub Main
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcSelectAll
dcSetSelCopyMode True
dcMoveSelObjs 2, 3
dcUnselectAll
End Sub

Related Topics: dcGetSelCopyMode

dcSetSelectBase

dcSetSelectBase(*x, y*)

This function sets the select base point to *x/y*. The base point is used for rotating, scaling, or grouping selected objects. This function must be called after the objects are selected.

Example:
Sub Main
dcSelectObjInBox 0, 0, 2, 2
dcSetSelectBase 0.25, 0.25
dcMoveSelObjs 1.2, 3.4
dcUnselectAll
End Sub

Related Topics: dcSelectAll, dcUnselectAll

dcSetShapesParms

dcSetShapesParms(*color, linetype, lineweight*)

This function sets all of the parameters used to create a shape.

color is the DeltaCad color number. The color number starts with zero, and increases in the same order that the colors are displayed in the color selection box in DeltaCad (i.e. 0 is black, 1 is red, 2 is green, 3 is yellow, etc.). You can also use the following pre-defined color values: dcBLACK, dcRED, dcGREEN, dcYELLOW, dcBLUE, dcPURPLE, dcBROWN, dcDARKGRAY, dcLIGHTGRAY, dcLIGHTBLUE, dcDARKGREEN, dcGREENYELLOW, dcDARKBLUE, dcDARKPURPLE, dcGRAYBLUE, dcWHITE

linetype is the DeltaCad line type. This may be set to one of the following: dcSOLID, dcHIDDEN, dcCENTER, dcPHANTOM, dcCUTTING, dcBREAK, dcSTITCH, dcARROW, dcTWOARROW, dcFILL, dcHATCH, dcHATCHNOOUTLINE. If using dcHATCH or dcHATCHNOOUTLINE you may add numbers (0 to 39) to the value to get the various hatch types. See the example below.

lineweight is the DeltaCad line weight. It can be one of the following values: dcTHIN, dcNORMAL, dcTHICK, dcHEAVY or dcWIDE

Example:

```
Sub Main
dcSetShapesParms dcGREEN, dcFILL, dcNORMAL
dcSetShapesParms dcBLUE, dcHATCH+2, dcTHIN
End Sub
```

Related Topics: dcGetShapesParms, dcSaveAllParms, dcRestoreAllParms

dcSetSplineParms

dcSetSplineParms(*color, linetype, lineweight*)

This function sets all of the parameters used to create a spline.

color is the DeltaCad color number. The color number starts with zero, and increases in the same order that the colors are displayed in the color selection box in DeltaCad (i.e. 0 is black, 1 is red, 2 is green, 3 is yellow, etc.). You can also use the following pre-defined color values: dcBLACK, dcRED, dcGREEN, dcYELLOW, dcBLUE, dcPURPLE, dcBROWN, dcDARKGRAY, dcLIGHTGRAY, dcLIGHTBLUE, dcDARKGREEN, dcGREENYELLOW, dcDARKBLUE, dcDARKPURPLE, dcGRAYBLUE, dcWHITE

linetype is the DeltaCad line type. This may be set to one of the following: dcSOLID, dcHIDDEN, dcCENTER, dcPHANTOM, dcCUTTING, dcBREAK, dcSTITCH, dcARROW, dcTWOARROW. (NOTE: Filled or Hatched lines are not valid for splines).

lineweight is the DeltaCad line weight. It can be one of the following values: dcTHIN, dcNORMAL, dcTHICK, dcHEAVY or dcWIDE

Example:

```
Sub Main
dcSetSplineParms dcRED, dcSOLID, dcWIDE
End Sub
```

Related Topics: dcGetSplineParms, dcSaveAllParms, dcRestoreAllParms

dcSetTextParms

dcSetTextParms(*color, fontname, fontstyle, angle, height, flags, excharspacing, exlinespacing*)

This function sets all of the parameters used to create a text object.

color is the DeltaCad color number. The color number starts with zero, and increases in the same order that the colors are displayed in the color selection box in DeltaCad (i.e. 0 is black, 1 is red, 2 is green, 3 is yellow, etc.). You can also use the following pre-defined color values: dcBLACK, dcRED, dcGREEN, dcYELLOW, dcBLUE, dcPURPLE, dcBROWN, dcDARKGRAY, dcLIGHTGRAY, dcLIGHTBLUE, dcDARKGREEN, dcGREENYELLOW, dcDARKBLUE, dcDARKPURPLE, dcGRAYBLUE, dcWHITE

fontname and *fontstyle* are the name and style of the font to be used.

angle is the angle to draw the text in degrees.

height is the point size of the font.

flags are set to specify the location of the text insertion point and the justification. Add the following three values to get the number for *flags*.

Position 0 = Left, 1 = Center, 2 = Right

Position 0 = Top, 4 = Center, 8 = Baseline, 12 = Bottom

Justification 0 = Left, 16 = Center, 32 = Right

excharspacing is the extra character spacing between characters

exlinespacing is the extra line spacing between lines of text

Example:

```
Sub Main
dcSetTextParms dcRED, "Times New Roman", "Bold", 0, 16, 8, 0, 0
End Sub
```

Related Topics: dcGetTextParms, dcSaveAllParms, dcRestoreAllParms

dcSetViewScale

dcSetViewScale(*scale*)

This function sets the view scale to the value defined by *scale*.

Example:

```
Sub Main
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcSetViewScale 2.0
End Sub
```

Related Topics: dcGetViewScaleandRect

dcTurnLayerOff

dcTurnLayerOff(*layername*)

This function turns off the layer with the name given by *layername*. NOTE: You can not turn off the current layer.

Example:

```
Sub Main
dcAddLayer "outline"
dcTurnLayerOff "outline"
End Sub
```

Related Topics: dcIsLayerOn, dcTurnLayerOn, dcGetCurrentLayer

dcTurnLayerOn

dcTurnLayerOn(*layername*)

This function turns on the layer with the name given by *layername*.

Example:

```
Sub Main
dcAddLayer "outline"
dcTurnLayerOn "outline"
End Sub
```

Related Topics: dcIsLayerOn, dcTurnLayerOff, dcGetCurrentLayer

dcTurnPrintRegionLayerOff

dcTurnPrintRegionLayerOff(*prname*, *layername*)

This function turns off the layer with the name given by *layername*, in the print region with the name given by *prname*.

Example:

```
Sub Main
dcAddLayer "outline"
dcAddPrintRegion "nregion", 1.0, 0, 0, 0, 0, dcPortrait
dcTurnPrintRegionLayerOff "nregion", "outline"
End Sub
```

Related Topics: dcTurnPrintRegionLayerOn

dcTurnPrintRegionLayerOn

dcTurnPrintRegionLayerOn(*prname*, *layername*)

This function turns on the layer with the name given by *layername*, in the print region with the name given by *prname*.

Example:

```
Sub Main
dcAddLayer "outline"
dcAddPrintRegion "nregion", 1.0, 0, 0, 0, 0, dcPortrait
dcTurnPrintRegionLayerOn "nregion", "outline"
End Sub
```

Related Topics: dcTurnPrintRegionLayerOff

dcUnGroupAllGroups

dcUnGroupAllGroups

This function ungroups all groups in the drawing.

Example:

```
Sub Main
dcUnGroupAllGroups
End Sub
```

Related Topics: dcUnGroupCurrentObject

dcUnGroupCurrentObject

dcUnGroupCurrentObject

This function ungroups the current group object.

Example:

```
Sub main
Dim ot As Long
Dim found As Boolean
found = true
While found
found = false
ot = dcGetFirstObject("")
While ot <> dcNone
If ot = dcGroup Then
dcUnGroupCurrentObject
found = true
End If
ot = dcGetNextObject
Wend
Wend
End Sub
```

Related Topics: dcUnGroupAllGroups, dcGetFirstObject, dcGetNextObject, dcGetObjectAt, dcGetFirstSelectedObject, dcGetNextSelectedObject

dcUnSelectAll

dcUnSelectAll

This function unselects all selected objects.

Example:

```
Sub Main
dcSelectAll
dcMoveSelObjs 1.2, 3.4
dcUnSelectAll
End Sub
```

Related Topics: dcSelectAll

dcUnSelectCurrentObject

dcUnSelectCurrentObject

This function unselects the current object. The current object is set in `dcGetFirstObject`, `dcGetNextObject`, `dcGetFirstSelectedObject`, `dcGetNextSelectedObject`, or `dcGetObjectAt`

Example:

```
Sub Main
  'This sub unselects all lines
  Dim ot As Long
  ot = dcGetFirstObject("")
  While ot <> dcNone
    If ot = dcLine Then
      If dcIsCurrentObjectSelected Then
        dcUnselectCurrentObject
      End If
    End If
    ot = dcGetNextObject
  Wend
End Sub
```

Related Topics: `dcGetFirstObject`, `dcGetNextObject`, `dcGetObjectAt`, `dcGetFirstSelectedObject`, `dcGetNextSelectedObject`, `dcSelectCurrentObject`, `dcIsCurrentObjectSelected`

dcUpdateDisplay**dcUpdateDisplay**(*onoff*)

This function turns display updates on or off. If display updates are turned off it can speed up creating a drawing. If *onoff* is set to False it turns off display updates. If *onoff* is set to True it turns on display updates, and updates the display. Always remember to turn display updates back on at the end of your program.

Example:

```
Sub Main
  dcUpdateDisplay False
  dcCreateLine 0, 0, 1, 1
  dcCreateLine 0, 1, 1, 0
  dcUpdateDisplay True
End Sub
```

dcViewAll**dcViewAll**

This function changes the view scale and position so all objects are displayed.

Example:

```
Sub Main
  dcCreateLine 0, 0, 1, 1
  dcCreateLine 0, 1, 1, 0
  dcViewAll
End Sub
```

Related Topics: `dcViewBox`, `dcViewCenteredOnXY`, `dcViewPrevious`

dcViewBox

dcViewBox(*left, bottom, right, top*)

This function centers and scales the displayed view so the box defined by *left, bottom, right, and top* fits on the display.

Example:

```
Sub Main
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcViewBox -0.5, -0.5, 1.5, 1.5
End Sub
```

Related Topics: dcViewAll, dcViewCenteredOnXY, dcViewPrevious

dcViewCenteredOnXY

dcViewCenteredOnXY(*x, y*)

This function centers the displayed view on the point given by *x/y*.

Example:

```
Sub Main
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcViewCenteredOnXY 0.5, 0.5
End Sub
```

Related Topics: dcViewBox, dcViewAll, dcViewPrevious

dcViewPrevious

dcViewPrevious

This function sets the view scale and position to where it was before the last change. This function can be used to go back to the last five scales and positions by calling the function up to five times.

Example:

```
Sub Main
dcCreateLine 0, 0, 1, 1
dcCreateLine 0, 1, 1, 0
dcViewBox -0.5, -0.5, 1.5, 1.5
dcViewPrevious
End Sub
```

Related Topics: dcViewBox, dcViewCenteredOnXY, dcViewAll

Troubleshooting

DeltaCad troubleshooting web site

If none of the solutions below help you can look on the internet at <http://www.deltacad.com/troubles.html> for the latest troubleshooting info. for DeltaCad.

Computer lock up

If the computer locks up it is usually caused by an outdated or incorrect video driver for your video card. You should check to see if you have the latest video driver for your video card. Most video card manufacturers have a web site where you can get the latest video driver for your card.

Display, Printing or FAX problems

Most problems with the computer display, the printer, or the FAX modem are a result of an older device driver. If you encounter a one of these problems you should contact the manufacturer of the device and obtain a copy of the latest version of the device driver.

Printing not to scale

Most problems people have with the printer not printing to scale are because they have not set the print region correctly. To set a print region pick **Set Print Region** from the **File** menu, then enter the print area and print scale. The default print scale of 0.0 causes the drawing to be scaled to fit the printer paper, and the default print area (all zero's) causes the entire drawing to be printed.

Drawing prints blank sheet of paper

The default print area in DeltaCad is set to print all objects. If the left, bottom, right, top print area numbers are all set to zero DeltaCad will print all objects. If you have accidentally drawn an object a long ways away from the main part of your drawing then DeltaCad will try to print it and the rest of your drawing. The result is that the print is zoomed out so far that nothing is actually printed. You can correct the problem by erasing the object, or by setting the print area to be exactly what you want printed. To set the print area pick **Set Print Region** from the **File** menu.

Drawing zooms out too far when opening

The default initial size and position is set to zoom the drawing out so all objects are displayed when opening a drawing. If you have accidentally drawn an object a long ways away from the main part of your drawing then DeltaCad will try to display it and the rest of your drawing. The result is that the drawing will zoom way out so you probably won't see anything. You can correct this problem by erasing the object. Sometimes the object is hard to find so it is easier to select the part of the drawing you want and copy it to a new drawing. You could also specify what to display when the drawing is opened by picking **Drawing Init Size and Pos** from the **Options** menu.

Pen Plotter Problems

There are two common problems with printing to pen plotters. Most pen plotter device drivers will not print True Type Fonts, so Windows will substitute the closest plotter font which is installed on the system. Some plotter device drivers will draw lines too thick. To solve the line thickness problem pick **Drawing Data** from the **Options** menu and enter in a line thickness which is thinner than the pen thickness which the device driver is using (try entering 0.001"), also pick the **Actual Thickness** button located beneath the Line Thickness section of the Drawing Data pop-up window.

Dimension text and arrowheads too small

If your dimension text and arrowheads are too small it is usually because you have not entered a Drawing Scale. To set the Drawing Scale pick **Drawing Scale** from the **Options** menu. The font size and arrowhead sizes are scaled with the drawing scale. This way you can enter normal size numbers. For Example, on a large drawing of a house (with a drawing scale of 96) you can enter 12 for the text point size, but the actual point size used will be 1152 (12 x 96). You can also change the size of the dimension text by picking the **Font** button on the **Dimension** toolbar tab, and you can change the size of the arrowheads by picking **Drawing Data** from the **Options** menu.

DeltaCad will not print in color

DeltaCad will only print in color if the printer device driver for your printer supports color. There are some color printers on the market which do not have color device drivers at this time, and some printer drivers have color options which must be set by picking **Printers** in **Control Panel**.

Disk full problems

If the hard disk on your computer is full, or very close to full, you will experience a wide variety of problems from DeltaCad, and most other programs. When printing DeltaCad copies the drawing data to a print spooler file, and then Microsoft® Windows™ sends the data to the printer. However if the disk is full, there will be no room for the print spooler file, so you will not get your print. Any operation which places data on the clipboard also requires some free disk space. Many other operations require temporary disk space to execute, so always make sure that the hard disk on your computer is not full.

Index

A

Add a point to a spline	101
Angle Dimension	93, 97
Angled Line	68
Angles	40
Arrange Icons	153
Arrow Direction	90
AutoSave	57

B

Base Point	24, 112
Bill of Materials	148, 149
Bookmarks	36, 140

C

Calculator	154
Cascade	153
Center view around a point	122
Chamfer	108
Change color of selected objects	118
Change layer of selected objects	118
Change line type/cross-hatch pattern of selected objects	119
Change line weight of selected objects	119
Change selected text	117
Change view scale	123
Circle Toolbar functions	71
Clipboard	138, 139
Clipboard - Copy	138
Clipboard - Cut	138
Clipboard - Paste	1, 138
Close	136
Color - Circle	71

Color - Dimension	90
Color - Line	64
Color - Point	60
Color - Shapes	82
Color - Text	87
Comments	2
Computed Numeric Input	49
Computer lock up	221
Continued Line	65
Copy	138, 139
Corner	107
Create a chamfer	108
Create a corner	107
Create a radius	109
Cut	138

D

Data Entry Line	35, 49, 50, 52, 53, 54
dcAddLayer	165
dcAddPrintRegion	165
dcAddSymbol	165
dcChangedSinceLastSave	165
dcChgColorSelObjs	166
dcChgLayerSelObjs	166
dcChgLineTypeSelObjs	167
dcChgLineWeightSelObjs	167
dcClose	167
dcCloseWithoutSaving	168
dcCopyBMPFiletoClipboard	168
dcCopyCurrentPicturetoFile	169
dcCopySelObjstoClipboard	169
dcCopySymboltoClipboard	169
dcCreateBox	169
dcCreateCircle	170
dcCreateCircleEx	170
dcCreateDim	171
dcCreateLine	171
dcCreatePoint	172
dcCreateShape	172
dcCreateSpline	173
dcCreateText	173
dcCreateTriangle	173
dcDeleteLayer	173

DeltaCad User's Guide

dcDeletePrintRegion	174	dcGetPrintRegionData	194
dcDeleteSymbol	174	dcGetPrintRegionName	195
dcDoesLayerExist	174	dcGetSelCopyMode	195
dcEndPrintPreview	174	dcGetShapeData	195, 196
dcEraseCurrentObject	175	dcGetShapesParms	196
dcEraseSelObjs	175	dcGetSplineData	196, 197
dcEraseTempPoints	175	dcGetSplineParms	197
dcExportScreenToBMPFile	176	dcGetTextData	198
dcGetActiveWindow	176	dcGetTextParms	199
dcGetAreaofCurrentObject	176	dcGetViewScaleandRect	199
dcGetBookmark	177	dcGroupSelObjs	200
dcGetCircleData	177	dcIsCurrentObjectSelected	200
dcGetCircleParms	178	dcIsLayerOn	200
dcGetCurrentLayer	178	dcIsPrintRegionLayerOn	201
dcGetCurrentPrintRegion	178, 179	dcIsSelected	201
dcGetDeltaCadDwgName	179	dcMirrorSelObjs	202
dcGetDeltaCadProgramDirectory	179	dcMoveSelObjs	202
dcGetDeltaCadVersion	180	dcNew	202
dcGetDeltaCadWindowHandle	180	dcNewSymbolFile	203
dcGetDimData	180, 181	dcOpen	203
dcGetDimParms	181, 182	dcOpenGroupWindow	203
dcGetDrawingData	182	dcPasteClipboardData	204
dcGetDrawingScale	183	dcPasteSpecialClipboardData	204
dcGetDrawingWindowHandle	183	dcPrint	204
dcGetFirstGroupName	183	dcPrintPreview	204
dcGetFirstObject	184	dcReadInput	204
dcGetFirstSelectedObject	185	dcRenameLayer	205
dcGetGroupData	185	dcRenamePrintRegion	205
dcGetInitialSizeandPos	186	dcRestoreAllParms	205
dcGetLayerName	186	dcRestoreBookmark	206
dcGetLengthofCurrentObject	187	dcRotateSelObjs	206
dcGetLineData	187	dcSave	206
dcGetLineParms	188	dcSaveAllParms	206
dcGetNextGroupName	188	dcSaveAs	207
dcGetNextObject	188	dcScaleSelObjs	207
dcGetNextSelectedObject	189	dcSelectAll	208
dcGetNumberofDeltaCadDwgs	189	dcSelectCurrentObject	208
dcGetNumberofLayers	190	dcSelectObjInBox	208
dcGetNumberofPrintRegions	190	dcSetActiveWindow	209
dcGetObjectAt	190	dcSetBookmark	209
dcGetObjectBox	191	dcSetCircleParms	210
dcGetObjectLayer	192	dcSetCurrentLayer	210
dcGetPictureData	192	dcSetCurrentPrintRegion	210
dcGetPointData	193		
dcGetPointParms	193		

dcSetDimParms	211	Draw a continued line	65
dcSetDrawingData	212	Draw a continuous spline	100
dcSetDrawingScale	212	Draw a diameter dimension	96
dcSetDrawingWindowMode	212	Draw a horizontal dimension	91
dcSetInitialSizeandPos	213	Draw a horizontal line	66
dcSetLineParms	213	Draw a line with two points	65
dcSetPointParms	214	Draw a parallel circle	80
dcSetSelCopyMode	214	Draw a parallel line	70
dcSetSelectBase	214	Draw a parallel spline	102
dcSetShapesParms	215	Draw a parallelogram	84
dcSetSplineParms	215	Draw a perpendicular line	69
dcSetTextParms	216	Draw a point	61
dcSetViewScale	216	Draw a point offset equally from the last two points	62
dcTurnLayerOff	217	Draw a point-to-point dimension	98
dcTurnLayerOn	217	Draw a polygon	86
dcTurnPrintRegionLayerOff	217	Draw a radius dimension	96
dcTurnPrintRegionLayerOn	217	Draw a rectangle using the center and corner	83
dcUnGroupAllGroups	218	Draw a rectangle using two corners	83
dcUnGroupCurrentObject	218	Draw a spline	100
dcUnSelectAll	218	Draw a text pointer arrow	95
dcUnSelectCurrentObject	219	Draw a triangle, square, pentagon, etc.	85
dcUpdateDisplay	219	Draw a vertical dimension	92
dcViewAll	219	Draw a vertical line	67
dcViewBox	220	Draw all points in an object	63
dcViewCenteredOnXY	220	Draw an angle dimension	97
dcViewPrevious	220	Draw an angled dimension	93
Default Symbol	140	Draw an angled dimension with angled extension lines	94
Delete File	137	Draw an angled line	68
Delete Key	54	Draw an ellipse	77
DeltaCad troubleshooting web site	221	Draw an ellipse within a parallelogram	78
DeltaCad will not print in color	223	Draw an elliptical arc	79
Diameter Dimension	96	Draw an X Y coordinate dimension	98
Dimensin text and arrowheads too small	223	Draw mid-points	63
Dimension Font	91	Draw multiple line text	89
Dimension Toolbar functions	90	Draw single line text	89
Disk full problems	223	Drawing - Circle	71
Display, Printing or FAX problems	221	Drawing - Dimension	90
Draw a 180 degree arc with two points	81	Drawing - Line	64
Draw a circle tangent to three lines	74	Drawing - Point	59
Draw a circle with center and radius	72	Drawing - Shapes	82
Draw a circle with three points	73	Drawing - Text	87
Draw a circle with two points	80		
Draw a circle within a box	72		
Draw a circular arc using the center	75		
Draw a circular arc using three points	76		

DeltaCad User's Guide

Drawing Area	39
Drawing Data	47
Drawing prints blank sheet of paper	222
Drawing Scale	42, 43, 44, 45
Drawing the Calculator	13
Drawing zooms out too far when opening	222
DXF	134, 136

E

Edit grouped objects	104
Edit Macro	158
Edit object data	103
Ellipse	77, 110
Elliptical Arc	79, 110
End of Tutorial	31
Enter Key	52, 53
Equally Offset Point	62
Erase	54, 59, 105, 116
Erase an object	105
Erase selected objects	54, 116
Erase Temp Points	60
Exiting DeltaCad	6, 30
Extended Keyboard	53
Extract Group Data	148

F

Feet, Inches, and Fractions	49
File - Close	133
File - Most Recently Used	133
File - New	133
File - Open	133
File - Save	133
File - SaveAs	133
File functions	160
File Operations	133
Font - Dimension	90
Font - Text	87
Font Style	87
Fractions	49, 52

G

Grid	55, 56, 143
Group	104, 116
Group Data	148, 150

Group Functions	160
Group selected objects	116

H

Help - Contents	48
Help - F1 Function Key	48
Help - How to Use Help	48
Help - Search for Help On	48
Horizontal Dimension	91
Horizontal Line	14, 15, 17, 18, 19, 21, 66
How to Use Help	48

I

Initial Size and Position	46
Insertion Point	25, 27, 28, 88, 91, 138, 139
Installation	5
Installing DeltaCad	5
Instruction Line	13, 24, 35, 52
Intersection Point	62, 107
Introduction	1
Isometric	79

J

Justification	34, 88, 117
---------------	-------------

K

Keyboard	49, 52, 53
----------	------------

L

Layer	41, 118, 124, 129
Layer functions	160
Learning DeltaCad	11
Line Toolbar functions	64
Line Type	47, 64, 71, 82, 103, 119, 140
Line Weight	64, 71, 82, 103, 119, 140

M

Macro Language	155, 156
Macro Toolbar Tab	159
Main Menu	34, 37
Maximize	13, 38, 141
Midpoint	61
Minimize	38
Mirror selected objects	114
Misc. functions	161
Most Recently Used Files	136
Mouse	55, 56, 57
Move a spline point	102
Move an object point	105
Move selected objects	112
Move to the bottom of the Z-Order	110
Move to the top of the Z-Order	110

N

New	133, 134, 135
New Window	151

O

Object creating functions	162
Object parameter functions	162
Object retrieval functions	163
Offline Printing	132
Offset F2	52
Open	134, 136
Options	37, 43, 88, 140, 141, 146, 147
Options - Default Line Weight	140
Options - Reset	140
Options - Screen Size	140
Ordering or Upgrading DeltaCad	6

P

Parallel Line	21, 70
Paste	1, 138, 139
Pen Plotter Problems	222
Pentagon	85
Perm/Temp Points	60
Perpendicular Line	69
Point	59, 60
Point Toolbar functions	60

Polygon	86
Precision	90
Previous views	123
Print	28, 29, 41, 59, 127, 128, 129, 130
Print - All Print Regions	127
Print - Print Active Window	127
Print - Selected Print Region	127
Print Area	130
Print Preview	131
Print Region	28, 127, 128, 129
Print region functions	163
Print Scale	29, 43, 130
Printer Setup	130
Printing not to scale	221
Printing the Calculator	28

R

Radius	109
Radius Dimension	96
Registering DeltaCad	5
Remove a point from a spline	101
Removing DeltaCad	8
Reset	87, 147
Restore	38
Right Mouse Button	53
Rotate selected objects	113
Run macro	157

S

Sample	2, 43
Sample Drawings & Libraries	9
Save	134, 136
SaveAs	136
Saving and Exiting DeltaCad	30
Scale - Drawing	42
Scale - Print	28, 43, 128
Scale - View	33, 42, 123
Scale selected objects	115
Screen Layout	33
Screen Size	141
Scroll bar	37, 38
Scroll Bar - Arrow	33

DeltaCad User's Guide

Scroll Bar - Shaft	33	Toolbar Tab - Line	13, 64, 65
Scroll Bar - Slider	33	Toolbar Tab - Point	13, 60, 61
Search for Help On	48	Toolbar Tab - Shapes	13, 82, 83, 125
Select Additional Objects	111	Toolbar Tab - Text	13, 87, 89
Select functions	164	Triangle	85
Select Objects	111	Tutorial	11, 12
Selecting objects	111	U	
Set the Base Point for selected objects	112	Undo	103
Setting the Print Region(s)	128	UnInstall	8
Setup	5	Unselect	23, 24, 111
Shapes Toolbar functions	82	V	
Shortcut Keys	54	Vertical Dimension	92
Slide an endpoint	106	Vertical Line	13, 14, 17, 18, 19, 21, 67
Slide endpoints of selected lines	120	View a single layer	124
Slider	38	View all	122
Snap	55, 56	View functions	164
Special Text Input	49	View objects within a rectangle	121
Spline	100	View Scale	35, 36, 42, 123
Spline Toolbar functions	99	View Toolbar Tab	125
Split	110	W	
Split a line, circle, arc, ellipse, or elliptical arc	110	Wheel	38
Start DeltaCad	13	Window	151
Starting and Exiting DeltaCad	6	Window - Arrange Icons	151
Starting DeltaCad	6	Window - Cascade	151
Status Bar	13, 16, 24, 27, 35, 49, 52, 53, 54, 123	Window - New Window	151
Suggestions	2, 49	Window - Tile	151
Symbol	135	Window - Tile Horizontally	151
Symbol functions	164	Window - Tile Vertically	151
System Requirements	3	Working Directory	6
T		Z	
Technical support	2	Zoom In	1, 16, 35, 38, 121, 123
Text - Multiple Line	89	Zoom Out	1, 16, 35, 124
Text - Single Line	89	Z-Order	110
Text Font	87		
Text Height	34, 87		
Text Pointer Arrow	95		
Text Toolbar functions	87		
Tile	151, 152		
Tile Horizontally	152		
Tile Vertically	152		
Toolbar	34, 37		
Toolbar Tab - Circle	71, 72		
Toolbar Tab - Dimension	90, 91		