

ModifyGraph (axes)

```
jackbarb[][0]= 40           // length of stem
jackbarb[][1]= 45*pi/180    // angle (45deg)
jackbarb[][2]= p           // wind speed code
SetDimLabel 1,2,windBarb,jackbarb
ModifyGraph mode=3,arrowMarker(jack)={jackbarb,1,10,0.5,0}
ModifyGraph margin(top)=62,margin(right)=84
```

See also **Wind Barb Plots** on page II-329.

Inline arrows and barb sharpness.

```
Make/O/N=20 wavex=cos(x/3),wavey=sin(x)
Display wavey vs wavex
ModifyGraph mode=3,arrowMarker={_inline_,1,20,.5,0,barbSharp= 0.2}
```

Use direct color mode to individually color each point in a trace:

```
Make jack=sin(x/8)
Make/N=(128,3)/B/U jackrgb
Display jack
ModifyGraph mode=3,marker=19
jackrgb= enoise(128)+128
ModifyGraph zColor(jack)={jackrgb,*,*,directRGB}
```

Use masking.

```
Make/N=100 jack= (p&1) ? sin(x/8) : cos(x/8)
Display jack

Make/N=100 mjack= (p&1) ? 0 : NaN           // just to show NaN can be used
ModifyGraph mask(jack)={mjack,0,NaN}

// now switch which points are shown
mjack= (p&1) ? NaN : 0
```

See Also

Trace Names on page II-282, **Programming With Trace Names** on page IV-87.

ModifyGraph (axes)

```
ModifyGraph [/W=winName/Z] key [(axisName)] = value
[, key [(axisName)] = value]...
```

This section of ModifyGraph relates to modifying the appearance of axes in a graph.

Parameters

Each *key* parameter may take an optional *axisName* enclosed in parentheses.

axisName is "left", "right", "top", "bottom" or the name of a free axis such as "vertCrossing". For instance, "ModifyGraph axThick(left)=0.5" sets the axis thickness for only the left axis.

If "(*axisName*)" is omitted, all axes in the graph are affected. For instance, "ModifyGraph standoff=0" disables axis standoff for all axes in the graph.

The parameter descriptions below omit the optional "(*axisName*)".

axisClip= <i>c</i>	Specifies one of three clipping modes for traces.
<i>c</i> =0:	Clips traces to a plot rectangle as defined by the pair of axes used by a given trace (default).
<i>c</i> =1:	Plots traces on an axis with a restricted range (as set by axisEnab) to extend to the full range of the normal plot rectangle.
<i>c</i> =2:	Traces extend outside the normal plot rectangle to the full extent of the graph area.

axisEnab={*lowFrac*,*highFrac*}

Restricts the length of an axis to a subrange of normal. The axis is drawn from *lowFrac* to *highFrac* of graph area height (vertical axis) or width (horizontal axis). For instance, {0.1,0.75} specifies that the axis is drawn from 10% to 75% of the graph area height/width, instead of the normal 0% to 100%. AxisEnab is discussed in **Creating Split Axes** on page II-347 and **Creating Stacked Plots** on page II-324.

axisOnTop= <i>t</i>	Specifies drawing level of axis and associated grid lines. <i>t</i> =0: Draws axis before traces and images (default). <i>t</i> =1: Draws the axis after all traces and images.
axOffset= <i>a</i>	Specifies the distance from default axis position to actual axis position in units of the width of a zero character (0) in a tick mark label. Unlike margin, axOffset adjusts to changes in the size of the graph.
axThick= <i>t</i>	Specifies the axis thickness in points.
barGap= <i>fraction</i>	Sets the fraction of the width available for bars to be used as gap between bars. barGap sets the gap between bars within a single category while catGap sets the gap between categories.
btLen= <i>p</i>	Sets the length of major ("big") tick marks to <i>p</i> points. If <i>p</i> is zero, it uses the default length. <i>p</i> may be fractional.
btThick= <i>p</i>	Sets the thickness of major ("big") tick marks to <i>p</i> points. If <i>p</i> is zero, it uses the default thickness. <i>p</i> may be fractional.
catGap= <i>fraction</i>	The value for catGap is the fraction of the category width to be used as gap. The gap is divided equally between the start and end of the category width. A value of 0.2 would use 20% of the available space for the gap and leave 80% of the available space for the bars. catGap sets the gap between categories while barGap sets the gap between bars within a single category.
dateFormat={ <i>languageName</i> , <i>yearFormat</i> , <i>monthFormat</i> , <i>dayOfMonthFormat</i> , <i>dayOfWeekFormat</i> , <i>layoutStr</i> , <i>commonFormat</i> }	Sets the custom date format used in the active graph. Note: Use a custom date format only if you turn it on via a ModifyGraph dateInfo command. The last parameter to the ModifyGraph dateInfo command must be -1 to turn on the custom date format. Parameters are the same as for the LoadWave/R flag except for the last one. <i>commonFormat</i> selects the common date format to use in the Modify Axis dialog. The legal values correspond to the choices in the Common Format pop-up menu of the Modify Axis dialog. They are:

Value	Date Format	Value	Date Format
1	mm/dd/yy	16	mm/yy
2	mm-dd-yy	17	mm.yy
3	mm.dd.yy	18	Abbreviated month and year
4	mmddyy	19	Full month and year
6	dd/mm/yy	21	mm/dd
7	dd-mm-yy	22	dd.mm
8	dd.mm.yy	23	Abbreviated month and day
9	ddmmyy	24	Full month and day
11	yy/mm/dd	26	Abbreviated date without day of week
12	yy-mm-dd	27	Abbreviated date with day of week
13	yy.mm.dd	28	Full date without day of week
14	yymmdd	29	Full date with day of week

ModifyGraph (axes)

If the *commonFormat* parameter is negative, then it will select the Use Custom Format radio button in the Modify Axis dialog rather than Use Common Format and will then use the absolute value of *commonFormat* to determine which item to select in the Common Format pop-up menu.

`dateInfo={sd,tm,dt}` Controls formatting of date/time axes.

`sd=0:` Show date in the date&time format.

The date is always suppressed if *tm*=2 (elapsed time).

The time is always suppressed if there are fewer than one ticks per day and *tm*=0 (12-hour time) or *tm*=1 (24-hour time).

`sd=1:` Suppress date.

The date is always suppressed if *tm*=2 (elapsed time).

`sd=2:` Suppress time.

The time is always shown if *tm*=2 (elapsed time).

sd=2 requires Igor Pro 9.00 or later.

`tm=0:` 12 hour (AM/PM) time.

`tm=1:` 24 hour (military) time.

`tm=2:` Elapsed time.

`dt=-1:` Custom date as specified via the *dateFormat* keyword.

`dt=0:` Short dates (2/22/90).

`dt=1:` Long dates (Thursday, February 22, 1990).

`dt=2:` Abbreviated dates (Thurs, Feb 22, 1990).

`font="fontName"` Sets the axis label font, e.g., `font(left)="Helvetica"`.

`freePos(freeAxName)=p`

Sets the position of the *free* axis relative to the edge of the plot area to which the axis is anchored. *p* is in points. i.e., if the axis was made via `/R=axName` then the axis is placed *p* points from the right edge of the plot area. Positive is away from the central plot area. *freeAxName* may not be any of the standard axes: "left", "bottom", "right" or "top".

`freePos(freeAxName)={crossAxVal,crossAxName}`

Positions the *free* axis so it will cross the perpendicular axis *crossAxName* where it has a value of *crossAxVal*. *freeAxName* may not be any of the standard axis names "left", "bottom", "right", or "top", though *crossAxName* may.

You can position a free axis as a fraction of the distance across the plot area by using `kwFraction` for *crossAxName*. *crossAxVal* must then be between 0 and 1; any values outside this range are clipped to valid values.

`fsiz=s`

Autosizes (*s*=0) tick mark labels and axis labels.

If *s* is between 3 and 99 then the labels are fixed at *s* points.

<code>fstyle=f</code>	<p><i>f</i> is a bitwise parameter with each bit controlling one aspect of the font style for the axis and tick mark labels as follows:</p> <p>Bit 0: Bold</p> <p>Bit 1: Italic</p> <p>Bit 2: Underline</p> <p>Bit 4: Strikethrough</p> <p>See Setting Bit Parameters on page IV-12 for details about bit settings.</p>
<code>ftLen=p</code>	Sets the length of 5th (or emphasized minor) tick marks to <i>p</i> points. If <i>p</i> is zero, it uses the default length. <i>p</i> may be fractional.
<code>ftThick=p</code>	Sets the thickness of 5th (or emphasized minor) tick marks to <i>p</i> points (fractional). If <i>p</i> is zero, it uses the default thickness.
<code>grid=g</code>	<p>Controls grid lines.</p> <p><i>g</i>=0: Grid off.</p> <p><i>g</i>=1: Grid on.</p> <p><i>g</i>=2: Grid on major ticks only.</p>
<code>gridEnab={lowFrac,highFrac}</code>	Restricts the length of axis grid lines to a subrange of normal. The grid is drawn from <i>lowFrac</i> to <i>highFrac</i> of graph area height (if axis is horizontal) or width (if axis is vertical).
<code>gridHair=h</code>	<p>Sets the grid hairline thickness (<i>h</i> =0 to 3; 0 for thicker lines, 3 for thinner; default is 2). If <i>h</i>=0, the thickness of grid lines on major tick marks is the same as the axis thickness, half for a minor tick and one tenth for a subminor tick (log axis only). As <i>h</i> increases these thicknesses decrease by a factor of 2^{<i>h</i>}. If you want to see the effect of different values of <i>gridHair</i>, you will need to print a sample graph because you generally can't see the effect of thin lines on the screen. Also see the example experiment "Examples:Graphing Techniques:Graph Grid Demo".</p>
<code>gridStyle=g</code>	<p>Sets the grid style to various combinations of solid and dashed lines. In the following discussion, major, minor and subminor refer to grid lines the corresponding tick marks. Subminor ticks are used only on log axes when there is a small range and sufficient room (they correspond to hundredths of a decade). The different grid styles are solid, dotted, dashed, and blank. The possible grids are as follows:</p> <p><i>g</i>=0: Same as mode 1 if graph background is white else uses mode 5.</p> <p><i>g</i>=1: Major dotted, minor and subminor dashed.</p> <p><i>g</i>=2: All dotted.</p> <p><i>g</i>=3: Major solid, minor dotted, subminor blank.</p> <p><i>g</i>=4: Major and minor solid, subminor dotted.</p> <p><i>g</i>=5: All solid.</p> <p>Also see the example experiment "Examples:Graphing Techniques:Graph Grid Demo".</p>
<code>highTrip=h</code>	If the extrema of an axis are between its <i>lowTrip</i> and its <i>highTrip</i> then tick mark labels use fixed point notation. Otherwise they use exponential (scientific or engineering) notation.
<code>lblLatPos=p</code>	Sets a lateral offset for the axis label. This is an offset parallel to the corresponding axis. <i>p</i> is in points. Positive is down for vertical axes and to the right for horizontal axes.
<code>lblLineSpacing=linespace</code>	

ModifyGraph (axes)

	<p>Specifies an adjustment to the normal line spacing for multi-line axis labels. <code>linespace</code> is points of extra (plus or minus) line spacing. For negative values, a blank line may be necessary to avoid clipping the bottom of the last line.</p> <p><code>lblLineSpacing</code> affects all lines of the axis label. See also the <code>\sa</code> and <code>\sb</code> line spacing escape codes described under General Escape Codes on page III-54. They can be used to affect the spacing between individual lines.</p>
<code>lblMargin=<i>l</i></code>	Specifies the distance from the edge of graph to a label in points.
<code>lblPos=<i>p</i></code>	<p>Sets the distance from an axis to the corresponding axis label in points. If $p=0$, it automatically picks an appropriate distance.</p> <p>This setting is used only if the given graph edge has at least one free axis. Otherwise, the <code>lblMargin</code> setting is used to position the axis label.</p>
<code>lblPosMode=<i>m</i></code>	<p>Affects the meaning and usage of <code>lblPos</code>, <code>lblLatPos</code>, and <code>lblMargin</code> parameters. Mainly for use when you have multiple axes on a side and you need axis labels to be properly positioned even as you make graph windows dramatically larger or smaller.</p> <p><i>m</i>=0: Default compatibility mode (Margin or Axis absolute depending on presence of free axis).</p> <p><i>m</i>=1: Margin absolute.</p> <p><i>m</i>=2: Margin scaled.</p> <p><i>m</i>=3: Axis absolute.</p> <p><i>m</i>=4: Axis scaled.</p> <p>The absolute modes are measured in points whereas scaled modes have similar values but automatically expand or contract as the axis font height changes. Mode 0 is the default and results in no change relative to previous versions of Igor Pro that used <code>lblMargin</code> unless a given side used a free axis in which case it used <code>lblPos</code> in absolute mode. The margin modes measure relative to an edge of the graph while the axis modes measure relative to the position of the axis. When using stacked axes, use either margin modes. With multiple nonstacked axes, use Axis scaled if the graph edge is not using a fixed margin or use axis absolute if it is.</p>
<code>lblRot=<i>r</i></code>	Rotates the axis label by <i>r</i> degrees. <i>r</i> is a value from -360 to 360. Rotation is counterclockwise and starts from the label's normal orientation.
<code>linTkLabel=<i>tl</i></code>	<p><i>tl</i>=1 attaches the data units with any exponent or prefix to each tick label on a normal axis. <i>tl</i>=0 removes them.</p> <p>In Igor Pro 9.00 and later, <i>tl</i>=2 suppresses the space character normally displayed before units.</p>
<code>log=<i>l</i></code>	<p>Controls axis log mode.</p> <p><i>g</i>=0: Normal axis.</p> <p><i>g</i>=1: Log base 10.</p> <p><i>g</i>=2: Log base 2.</p>
<code>logHTrip=<i>h</i></code>	Same as <code>highTrip</code> but for log axes.
<code>logLabel=<i>l</i></code>	Sets the maximum number of decades in a log axis before minor tick labels are suppressed.
<code>logLTrip=<i>l</i></code>	Same as <code>lowTrip</code> but for log axes.
<code>loglinear=<i>l</i></code>	Switches to a linear tick method (<i>l</i> =1) on a log axis if the number of decades of ranges is less than 2. It switches to a linear tick exponent method if the number of decades is greater than five.

logmtlOffset= <i>o</i>	Offsets the minor tick mark labels on a log axis by <i>o</i> fractional points relative to the default tick mark label position. Positive is away from the axis. Added in Igor Pro 9.00.
logTicks= <i>t</i>	Sets the maximum number of decades in log axis before minor ticks are suppressed.
lowTrip= <i>l</i>	If the extrema of an axis are between its lowTrip and its highTrip then tick mark labels use fixed point notation. Otherwise they use exponential (scientific or engineering) notation.
manminor={ <i>number</i> , <i>emphasizeEvery</i> }	Specifies how to draw minor ticks in manual tick mode. There will be <i>number</i> ticks between each major (labeled) tick. You will usually want to set this to 4 to make 5 divisions, or 9 to make 10 divisions. A medium-sized tick (an emphasized minor tick) will be drawn every <i>emphasizeEvery</i> minor tick.
manTick={ <i>cantick</i> , <i>tickinc</i> , <i>exp</i> , <i>digitsrt</i> [, <i>timeUnit</i>]}	<p>Turns on manual tick mode. The tick from which all other ticks are calculated is the cononic tick (<i>cantick</i>). The numerical spacing between ticks is set by <i>tickinc</i>. <i>cantick</i> and <i>tickinc</i> are multiplied by 10^{exp}. The number of digits to the right of the decimal point displayed in the tick labels is set by <i>digitsrt</i>.</p> <p>The optional parameter <i>timeUnit</i> is used with Date/Time axes to specify the units of <i>tickinc</i>. In this case, <i>tickinc</i> must be an integer. The value of <i>timeUnit</i> is one of the following keywords:</p> <p>second, minute, hour, day, week, month, year</p> <p>On a date/time axis, the <i>exp</i> and <i>digitsrt</i> keywords are ignored, but must be present. You can set them to zero.</p>
manTick=0	Turns off manual tick mode.
margin= <i>m</i>	<p>Sets a fixed margin from the edge of the window to the axis in points. Used principally to make axes of multiple graphs on a page line up when “stacked”. You can use the left, right, bottom, and top axis names (even if an axis with that name doesn’t exist) to adjust the graph plot area. See Types of Axes on page II-279.</p> <p><i>m</i>=0: Sets “automatic” margin size (dependent on the length and height of tick marks and labels).</p> <p><i>m</i>=-1: Sets the margin to “none”, or 0. The axis is drawn at the graph window’s edge.</p>
minor= <i>m</i>	Disables (<i>m</i> =0) or enables (<i>m</i> =1) minor ticks.
mirror= <i>m</i>	<p>Controls axis mirroring.</p> <p><i>m</i>=1: Right axis mirroring left or top mirroring bottom.</p> <p><i>m</i>=2: Mirror axis without tick marks.</p> <p><i>m</i>=3: Mirror axis with tick marks and tick labels.</p> <p><i>m</i>=0: No mirroring.</p>
mirrorPos= <i>pos</i>	Specifies the position of the mirror axis relative to the normal position. <i>pos</i> is a value between 0 and 1.
noLabel= <i>n</i>	<p>Controls axis labeling.</p> <p><i>n</i>=0: Normal labels.</p> <p><i>n</i>=1: Suppresses tick mark labels.</p> <p><i>n</i>=2: Suppresses tick mark labels and axis labels.</p>

ModifyGraph (axes)

notation= <i>n</i>	Uses engineering (<i>n</i> =0) or scientific (<i>n</i> =1) notation for tick mark labels. Affects tick mark labels displayed exponentially. See highTrip and lowTrip. Does not affect log axes.
nticks= <i>n</i>	Specifies the approximate number of ticks marks (<i>n</i>) on axis.
prescaleExp= <i>exp</i>	Multiplies axis range by 10^{exp} for tick labeling and <i>exp</i> is subtracted from the axis label exponent. In other words, the exponent is moved from the tick labels to the axis label. (This affects the display only, not the source data.)
sep= <i>s</i>	Specifies the minimum number of screen points (<i>s</i>) between minor ticks.
standoff= <i>s</i>	Suppresses (<i>s</i> =0) or enables (<i>s</i> =1) axis standoff. Axis standoff prevents waves or markers from covering the axis.
stLen= <i>p</i>	Sets the length of minor ("small") tick marks to <i>p</i> points. If <i>p</i> is zero, it uses the default length. <i>p</i> may be fractional.
stThick= <i>p</i>	Sets the thickness of minor ("small") tick marks to <i>p</i> points. If <i>p</i> is zero, it uses the default thickness. <i>p</i> may be fractional.
tick= <i>t</i>	Sets tick position. <i>t</i> =0: Outside axis. <i>t</i> =1: Crossing axis. <i>t</i> =2: Inside axis. <i>t</i> =3: None. In a category plot, adding 4 to the usual values for the tick keyword will place the tick marks in the center of each category rather than at the edges. In a normal (non-category) plot specifying <i>t</i> =4 or <i>t</i> =5 replaces tick marks and axis lines with rectangular regions with alternating fills. This feature, which was added in Igor Pro 8.00, is used for cartographic plots. <i>t</i> =4 gives even fills and <i>t</i> =5 gives odd fills. The btLen keyword determines the thickness of the fill area. It is recommended that you use mirror axes and axisOnTop=1 as shown in this example: <pre>Make jack=sin(x/8) Display jack ModifyGraph axisOnTop=1 ModifyGraph mirror=1 ModifyGraph standoff=0 ModifyGraph tick=4</pre>
tickEnab={ <i>lowTick</i> , <i>highTick</i> }	Restricts axis ticking to a subrange of normal. Ticks are drawn and labelled only if they fall within this inclusive numerical range.
tickExp= <i>te</i>	<i>te</i> =1 forces tick labels to exponential notation when labels have units with a prefix. <i>te</i> =0 turns this off.
tickUnit= <i>tu</i>	Suppresses (<i>tu</i> =1) or turns on (<i>tu</i> =0) units labels attached to tick marks.
tickZap={ [<i>v1</i> [<i>v2</i> [<i>v3</i>]]] }	Suppresses drawing of the tick mark label for values given in the list. This is useful when you have crossing axes to prevent tick mark labels from overlapping. The list may contain zero, one, two or three values. The values must be exact to suppress the label.

tkLblRot= r	Rotates the tick mark labels by r degrees. r is a value from -360 to 360. Rotation is counterclockwise and starts from the label's normal orientation.
tlOffset= o	Offsets the tick mark labels by o fractional points relative to the default tick mark label position. Positive is away from the axis.
ttLen= p	Sets the length of subminor ("tiny") tick marks to p points. If p is zero, it uses the default length. p may be fractional. Subminor ticks are used only in log axes.
ttThick= p	Sets the thickness of subminor ("tiny") tick marks to p points. If p is zero, it uses the default thickness. p may be fractional.
userticks={ <i>tickPosWave</i> , <i>tickLabelWave</i> }	<p>Draws axes with purely user-defined tick mark positions and labels, overriding other settings. <i>tickPosWave</i> is a numeric wave containing the desired positions of the tick marks, and <i>tickLabelWave</i> is a text wave containing the labels. See User Ticks from Waves on page II-313 for an example.</p> <p>The tick mark labels can be multiline and use styled text. For more details, see Fancy Tick Mark Labels on page II-358.</p> <p><i>tickPosWave</i> need not be monotonic. Igor will plot a tick if a value is in the range of the axis. Both linear and log axes are supported.</p> <p>Graph margins will adjust to accommodate tick labels. This will not prevent overlap between labels, which you will need fix yourself.</p>
userticks=0	Removes the userticks setting returning the axis to the previously-set mode.
useTsep= t	$t=1$ displays a thousand's separator character between every group of three digits in the tick mark label (e.g., "1,000" instead of "1000"). The default is $t=0$.
zapLZ= t	Removes ($t=1$) leading zeros from tick mark labels. For example 0.5 becomes .5 and -0.5 becomes -.5. Default is $t=0$.
zapTZ= t	Removes ($t=1$) trailing zeros from tick mark labels. The the radix point will also be removed if all digits are zero. Default is $t=0$.
zero=z	<p>Controls the zero line.</p> <p>$z=0$: A zero line at $x=0$ or $y=0$.</p> <p> The line style is set to $z-1$. See ModifyGraph (traces) on page V-613, <i>lStyle</i> keyword, for details on line styles.</p> <p>$z=1$: No zero line.</p>
zeroThick= zt	Sets the thickness of the zero line in points, from 0.0 to 5.0 points. $zt=0.0$ means the zero line thickness automatically follows the thickness of the axis; this is the default. You can use 0.1 for a thin zero line thickness.
ZisZ= t	$t=1$ uses the single digit 0 as the zero tick mark label (if any) regardless of the number of digits used for other labels. Default is $t=0$.