Optional Limits

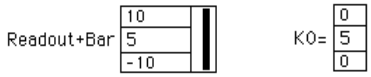
Whenever the numeric readout is visible, the optional limit values may be displayed too.

```
// Set limits font size to 10 points. Readout widths unchanged.
ValDisplay valdisp2 barmisc={10,50}
ValDisplay valdisp0 barmisc={10,1000}
```

Optional Title

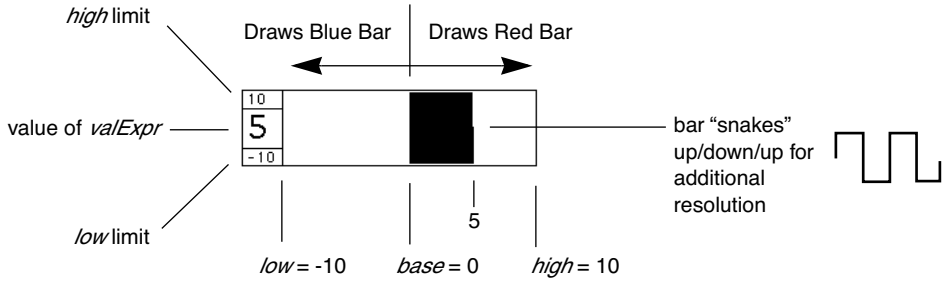
The control title steals horizontal space from the numeric readout and the bar, pushing them to the right. You may need to increase the control width to prevent them from disappearing.

```
// Add titles. Readout widths, control widths unchanged.
ValDisplay valdisp2 title="Readout+Bar"
ValDisplay valdisp0 title="K0="
```



The limits values *low*, *high*, and *base* and the value of *valExpr* control how the bar, if any, is drawn. The bar is drawn from a starting position corresponding to the *base* value to an ending position determined by the value of *valExpr*, *low* and *high*. *low* corresponds to the left side of the bar, and *high* corresponds to the right. The position that corresponds to the *base* value is linearly interpolated between *low* and *high*.

For example, with *low* = -10, *high*=10, and *base*= 0, a *valExpr* value of 5 will draw from the center of the bar area (0 is centered between -10 and 10) to the right, halfway from the center to the right of the bar area (5 is halfway from 0 to 10):



You can force the control to not draw bars with fractional parts by specifying mode=3.

Killing Controls

You can kill (delete) a control from within a procedure using the **KillControl** operation (page V-468). This might be useful in creating control panels that change their appearance depending on other settings.

You can interactively kill a control by selecting it with the arrow tool or the Select Control submenu and press Delete.

Getting Information About Controls

You can use the **ControlInfo** operation (page V-89) to obtain information about a given control. This is useful to obtain the current state of a checkbox or the current setting of a pop-up menu.

ControlInfo is usually used for control panels that have a Do It button or equivalent. When the user clicks the button, its action procedure calls ControlInfo to query the state of each relevant control and acts accordingly.