LocalToGlobal Obtain global (screen) value of local point

#include <Quickdraw.h> **Quickdraw**

```
void        LocalToGlobal(thePoint);
Point       *thePoint;      address of point to convert; receives result
```

LocalToGlobal converts the coordinates of a local point (relative to the current GrafPort origin) to global (screen) coordinates. It can then be compared to other global points or converted to the local coordinates of a different GrafPort.

thePoint is the address of a 4-byte Point structure, expressed in coordinates of the current GrafPort. Upon return, it will contain the coordinates of that same position, expressed to the global, screen coordinates.

Returns: none

Notes: The result of the conversion is based relative to coordinate (0,0) of the device's BitMap; typically the screen.

To convert the coordinates of a rectangle from local to global, you can apply this call to both corners; e.g.:

```
Rect    theRect;

LocalToGlobal( & topleft(theRect);
LocalToGlobal( &.botRight(theRect) );
```

Rectangles and other graphic elements (regions and polygons) can be converted to the global coordinate system via a 3-step sequence:

1. Use **LocalToGlobal** to obtain the global coordinates of one corner of a local item.
2. Use **SubPt** or **DeltaPoint** to determine the difference between the local and global coordinate systems.
3. Use **Offset**Xxx to reposition the item.

For instance, the following sequence converts a local Polygon to global coordinates:

```
Point       tmpPt,localPt;
PolyHandle  thePoly;

tmpPt=localPt=topLeft( (*thePoly)->polyBBox );    /* get corner */
LocalToGlobal( &tmpPt );                          /* convert to global */
SubPt( localPt, &tmpPt );                         /* find  difference */
OffsetPoly( thePoly, tmpPt.h, tmpPt.v );          /* move the item */
```