**DrawControls**                    Draw all controls visible in a window

 #include <Controls.h>                                      **Control Manager**

  void              **DrawControls(**_theWindow_ **);**
  WindowPtr      _theWindow_ ;       window to update

   **DrawControls** draws all of a window's currently active and visible controls.
   This (or **UpdtControl**) should be used upon detecting an update event for a
   window that contains controls.

    _theWindow_ is a pointer to a 108-byte GrafPort structure (actually a 156-byte
              WindowRecord).  It is typically a value obtained from
              EventRecord.message after calling **WaitNextEvent**.

     **Returns**: none
   _____

 Notes:   This can be used at any time to draw the controls in a window.  It works by
          drawing all the controls, and letting the off-screen or covered controls be
          clipped.  The 128K ROM **UpdtControl** function is more efficient since it
          doesn't waste time drawing outside of the window's visible region.

          The most common usage is to call **DrawControls** to redraw scroll bars
          upon detecting an update event as in this code skeleton:

```
    if ( GetNextEvent( everyEvent, &myEvent ) ) {
        switch ( myEvent.what ) {
        case updateEvt:
                   updtWin = (WindowPtr)myEvent.message;
                   GetPort( &savePort );
                   SetPort( updtWin)
                   BeginUpdate( updtWin );
                   DrawControls( updtWin );
                   DrawGrowIcon( updtWin);   /* if needed */
                   MyDrawWin( updtWin);        /* draw window contents */
                   EndUpdate( updtWin );
                   SetPort( savePort )
                   break;
        case mouseDown:
              .
              . ... etc ...
              .
        }
     }
```

          The 'size icon' (if used) is also part of the content region of a window and
          will need to be redrawn when it is uncovered.