

Using Quickdraw Globals in an INIT

```
// Using Quickdraw Globals in an INIT
// This is a short bit of code that shows how to Initialize your own set of QuickDraw
// globals at INIT time. After you initialize your quickdraw globals, you can
// access any of them by using "qd.<theglobal>".
```

```
// Assumes inclusion of <MacHeader>
#include <GestaltEqu.h>
```

```
typedef struct {
    char          privates[76];
    long          randSeed;
    BitMap screenBits;
    Cursor arrow;
    Pattern       dkGray;
    Pattern       ltGray;
    Pattern       black;
    Pattern       white;
    GrafPtr       thePort;
    long          qdend;
} QDGlobals;
```

```
pascal void main ()
{
```

```
    long          oldA5,
                  result,
                  dummy;

    QDGlobals      qd;
    SysEnvRec     environment;
    CGrafPort     gp;
    OSErr         err;
```

```
    oldA5 = SetA5((long) &qd.qdend); // Tell A5 to point to our 'fake' QD Globals
    InitGraf(&qd.thePort);           // Initialize our QD Globals
```

```
    // This code tests the screen device type, so we can tell whether to open up a
    // Color or B&W GrafPort. If you don't plan to draw in color, you can skip this
    // code, and just use OpenPort.
```

```
    err = Gestalt (gestaltQuickdrawVersion, &result);
    // If your development system does not provide
    // glue for Gestalt, you may need to check for its
    // existence. See Using the Gestalt Manager for details
```

```
    if ((err == noErr) && (result >= gestalt8BitQD))
        OpenCPort(&gp);
```

```
    else
        OpenPort((GrafPtr)&gp);
```

```
        dummy = SetA5(oldA5);           // Restore A5 to its previous value
```

```
}
```