

Junque - an open-ended activity indicator

This little program demo's how you can implement an activity indicator for those open-ended routines that you might have in your project.

We're all familiar with the "% complete" bars that give us progress reports on some activity or other, such as installations, sorts, and the like. The problem I ran into was letting a user know that the computer hadn't gone off into the Bit Zone while it did some math - the math functions were kind of 'open-ended', in that I couldn't consistently tell how long it would take to complete the calculations. Rather than just drop in a plain old message box that didn't particularly gain me anything, I decided to come up with something that shows Things Are Happening, but didn't (erroneously) claim when they'd be done.

The indicator harkens back to the days when Real Computers had flashing lights on the front panel - you knew *something* was happening because the lights flashed. All it does is come up with a random pattern of 'on' and 'off' display lights - each time the routine is called, it comes up with a different set of on/off lights, and it can be called as often (or seldom) as you like.

The whole thing is accomplished without using any custom controls, VBX's, or anything else - just what comes with Visual Basic. The only 'tricky' parts are how you set up the series of LED's (actually graphics images in VB2 - you can use icons in VB1): if you play with the form, you'll see that there are actually TWO LEDs in each position, with one of them (the 'light' one) being invisible on startup. The restraining factor is the amount of patience you have in how many LED's you want to include. In the demo, there are 8 visible, and 8 invisible; they ALL have the same name, with only their index values changing. The updating process is speeded up *considerably* by numbering all the LEDs of the same type (lit or unlit) in sequence, followed by the other LEDs *in the same order*. In the demo, they're indexed as:

```
1 2 3 4 5 6 7 8 <<<<< position
0 1 2 3 4 5 6 7 <<<< LIT index
8 9 10 11 12 13 14 15 <<<< UNLIT index
```

By using this sequence, you can reference a single position's LEDs by simply referencing the lit LEDs index; with the unlit LED's index being the same as the lit's, plus the total number of LED's. See the example above.

Note that you can put the "working on it" or "thinking about it" or other notice in either the border of the form, or as a text box; you'll probably want to make it a separate form that you can call whenever you need it.

This little technique - though probably not original - is placed in the 'public domain' for anyone to use.

Have fun!