

LED Demo

A simple demo illustrating the use of graphicboxes to represent state-changing LEDs.

```
nomainwin
```

```
WindowWidth = 150 : WindowHeight = 90
```

```
button      #main.bSwitch,  "Switch!", [switch], UL, 10, 12, 60, 25
graphicbox  #main.gbGreen,   100, 10, 17, 17
graphicbox  #main.gbRed,     100, 28, 17, 17
```

```
stylebits   #main.gbGreen,   0, _WS_BORDER, 0, 0
stylebits   #main.gbRed,     0, _WS_BORDER, 0, 0
```

```
open "LEDs" for window_nf as #main
main.isOpen = 1
```

```
#main          "trapclose [exit]"
#main.gbGreen  "down; fill buttonface; flush bkSeg; color black"
#main.gbGreen  "backcolor darkgreen; place 8 8; circlefilled 7; flush greenSeg"
#main.gbRed    "down; fill buttonface; flush bkSeg; color black"
#main.gbRed    "backcolor red; place 8 8; circlefilled 7; flush redSeg"
```

```
[loop]
    'Un-comment next line for data Tx/Rx or status monitor effect:
    'timer (int(rnd(1) * 420) + 1), [switch]
wait
```

```
[exit]
    timer (0)
    if (main.isOpen) then close #main
end
```

```
[switch]
    timer (0)
    greenColor$ = word$("dark", abs(state)) + "green"
    redColor$   = word$("dark", abs(not(state))) + "red"
    state       = not(state)

    #main.gbGreen  "delsegment greenSeg; backcolor "; greenColor$
    #main.gbGreen  "place 8 8; circlefilled 7; flush greenSeg; redraw"
```

```
#main.gbRed      "delsegment redSeg; backcolor "; redColor$  
#main.gbRed      "place 8 8; circlefilled 7; flush redSeg; redraw"  
goto [loop]
```