```
''****************************
''*  VGA Text 32x15 v1.0     *
''*  (C) 2006 Parallax, Inc.  *
''****************************
```

CON

```
  cols = 32
  rows = 15

  screensize = cols * rows
  lastrow = screensize - cols

  vga_count = 21
```

VAR

```
  long  col, row, color, flag

  word  screen[screensize]
  long  colors[8 * 2]

  long  vga_status    '0/1/2 = off/visible/invisible      read-only   (2
  long  vga_enable    '0/non-0 = off/on                   write-only
  long  vga_pins      '%ppptttt = pins                    write-only
  long  vga_mode      '%tihv = tile,interlace,hpol,vpol   write-only
  long  vga_screen    'pointer to screen (words)          write-only
  long  vga_colors    'pointer to colors (longs)          write-only
  long  vga_ht        'horizontal tiles                   write-only
  long  vga_vt        'vertical tiles                     write-only
  long  vga_hx        'horizontal tile expansion          write-only
  long  vga_vx        'vertical tile expansion            write-only
  long  vga_ho        'horizontal offset                  write-only
  long  vga_vo        'vertical offset                    write-only
  long  vga_hd        'horizontal display ticks           write-only
  long  vga_hf        'horizontal front porch ticks       write-only
  long  vga_hs        'horizontal sync ticks              write-only
  long  vga_hb        'horizontal back porch ticks        write-only
  long  vga_vd        'vertical display lines             write-only
  long  vga_vf        'vertical front porch lines         write-only
  long  vga_vs        'vertical sync lines                write-only
  long  vga_vb        'vertical back porch lines          write-only
  long  vga_rate      'tick rate (Hz)                     write-only
```

OBJ

```
  vga : "vga"
```

```
PUB start(basepin) : okay

'' Start terminal - starts a cog
'' returns false if no cog available
''
'' requires at least 80MHz system clock

  setcolors(@palette)
  out(0)


  longmove(@vga_status, @vga_params, vga_count)
  vga_pins := basepin | %000_111
  vga_screen := @screen
  vga_colors := @colors
  vga_rate := clkfreq >> 2

  okay := vga.start(@vga_status)


PUB stop

'' Stop terminal - frees a cog

  vga.stop


PUB str(stringptr)

'' Print a zero-terminated string

  repeat strsize(stringptr)
    out(byte[stringptr++])


PUB dec(value) | i

'' Print a decimal number

  if value < 0
    -value
    out("-")

  i := 1_000_000_000

  repeat 10
    if value => i
```

```
      out(value / i + "0")
      value //= i
      result~~
    elseif result or i == 1
      out("0")
    i /= 10


PUB hex(value, digits)

'' Print a hexadecimal number

  value <<= (8 - digits) << 2
  repeat digits
    out(lookupz((value <-= 4) & $F : "0".."9", "A".."F"))


PUB bin(value, digits)

'' Print a binary number

  value <<= 32 - digits
  repeat digits
    out((value <-= 1) & 1 + "0")


PUB out(c) | i, k

'' Output a character
''
''     $00 = clear screen
''     $01 = home
''     $08 = backspace
''     $09 = tab (8 spaces per)
''     $0A = set X position (X follows)
''     $0B = set Y position (Y follows)
''     $0C = set color (color follows)
''     $0D = return
''  others = printable characters

  case flag
    $00: case c
           $00: wordfill(@screen, $220, screensize)
                col := row := 0
           $01: col := row := 0
           $08: if col
                  col--
           $09: repeat
                  print(" ")
```

```
            while col & 7
          $0A..$0C: flag := c
                    return
          $0D: newline
          other: print(c)
    $0A: col := c // cols
    $0B: row := c // rows
    $0C: color := c & 7
  flag := 0


PUB setcolors(colorptr) | i, fore, back

'' Override default color palette
'' colorptr must point to a list of up to 8 colors
'' arranged as follows (where r, g, b are 0..3):
''
''               fore   back
''               ------------
'' palette  byte %%rgb, %%rgb     'color 0
''          byte %%rgb, %%rgb     'color 1
''          byte %%rgb, %%rgb     'color 2
''          ...

  repeat i from 0 to 7
    fore := byte[colorptr][i << 1] << 2
    back := byte[colorptr][i << 1 + 1] << 2
    colors[i << 1]     := fore << 24 + back << 16 + fore << 8 + back
    colors[i << 1 + 1] := fore << 24 + fore << 16 + back << 8 + back


PRI print(c)

  screen[row * cols + col] := (color << 1 + c & 1) << 10 + $200 + c & $F
  if ++col == cols
    newline


PRI newline | i

  col := 0
  if ++row == rows
    row--
    wordmove(@screen, @screen[cols], lastrow)   'scroll lines
    wordfill(@screen[lastrow], $220, cols)      'clear new line


DAT
```

```
vga_params              long    0               'status
                        long    1               'enable
                        long    0               'pins
                        long    %1000           'mode
                        long    0               'videobase
                        long    0               'colorbase
                        long    cols            'hc
                        long    rows            'vc
                        long    1               'hx
                        long    1               'vx
                        long    0               'ho
                        long    0               'vo
                        long    512             'hd
                        long    10              'hf
                        long    75              'hs
                        long    43              'hb
                        long    480             'vd
                        long    11              'vf
                        long    2               'vs
                        long    31              'vb
                        long    0               'rate

                        '       fore   back
                        '        RGB    RGB
palette                 byte    %%222, %%001    '0    white / dark blue
                        byte    %%330, %%110    '1   yellow / brown
                        byte    %%202, %%000    '2  magenta / black
                        byte    %%111, %%333    '3     grey / white
                        byte    %%033, %%011    '4     cyan / dark cyan
                        byte    %%020, %%232    '5    green / gray-green
                        byte    %%100, %%311    '6      red / pink
                        byte    %%033, %%003    '7     cyan / blue
```