```
;---------------------- PIN Declarations ---------------------------------

PIN   1  CLK       COMBINATORIAL ; INPUT  pixel clock (PIXCLKB) from DPB
PIN   2  CLAMP     COMBINATORIAL ; INPUT  CLAMP-DP pulse from DPB
PIN   3  XH        COMBINATORIAL ; INPUT  XHATCH test pattern from U117
PIN   4  EV        COMBINATORIAL ; INPUT  EVENBIT from shift register U99
PIN   5  OD        COMBINATORIAL ; INPUT  ODDBIT from shift register U88
PIN   6  R         COMBINATORIAL ; INPUT  RED-ATT - red color attribute
PIN   7  G         COMBINATORIAL ; INPUT  GREEN-ATT - green color attribute
PIN   8  B         COMBINATORIAL ; INPUT  BLUE-ATT - blue color attribute
PIN   9  C0        COMBINATORIAL ; INPUT  TRANSPARENT attribute (active hi)
PIN   10 C1        COMBINATORIAL ; INPUT  INVERSE attribute (active low)
PIN   11 S0        COMBINATORIAL ; INPUT  S2-S0 - background selection
PIN   12 GND                               code, as follows:
PIN   13 S1        COMBINATORIAL ; INPUT    0XX = one of 4 xhatch patterns
PIN   14 S2        COMBINATORIAL ; INPUT    100 = white field
                                 ;          101 = grayscale
                                 ;          110 = external video
                                 ;          111 = black field
PIN   15 VBL       COMBINATORIAL ; INPUT  vertical blanking waveform
PIN   16 /EXT      COMBINATORIAL ; OUTPUT internal/external video select
PIN   17 /GRAY     COMBINATORIAL ; OUTPUT turns on grayscale generator
PIN   18 /BLUE     COMBINATORIAL ; OUTPUT blue digital video
PIN   19 /GREEN    COMBINATORIAL ; OUTPUT green digital video
PIN   20 /RED      COMBINATORIAL ; OUTPUT red digital video
PIN   21 HBLANK    COMBINATORIAL ; INPUT  horizontal blanking wfm from DPB
PIN   22 BLNK_WFM  COMBINATORIAL ; OUTPUT video blanking waveform BLANK-WFM
PIN   23 CBLANK    COMBINATORIAL ; OUTPUT composite blanking wfm for U28
PIN   24 VCC

;---------------------- Boolean Equation Segment -------------------------

EQUATIONS

GREEN = /HBLANK */VBL*(/CLK*(C1*/OD*G+/C1*OD*G + /S2*C1*/C0*OD*XH
        + /S2*/C1*/C0*/OD*XH + S2*/S1*/S0*C1*/C0*OD
        + S2*/S1*/S0*/C1*/C0*/OD) + CLK*(C1*/EV*G + /C1*EV*G
        + /S2*C1*/C0*EV*XH + /S2*/C1*/C0*/EV*XH
        + S2*/S1*/S0*C1*/C0*EV + S2*/S1*/S0*/C1*/C0*/EV))

BLUE = /HBLANK*/VBL*(/CLK*(C1*/OD*B + /C1*OD*B + /S2*C1*/C0*OD*XH
       + /S2*/C1*/C0*/OD*XH + S2*/S1*/S0*C1*/C0*OD
       + S2*/S1*/S0*/C1*/C0*/OD) + CLK*(C1*/EV*B + /C1*EV*B
       + /S2*C1*/C0*EV*XH + /S2*/C1*/C0*/EV*XH
       + S2*/S1*/S0*C1*/C0*EV + S2*/S1*/S0*/C1*/C0*/EV))

RED = /HBLANK*/VBL*(/CLK*(C1*/OD*R +/C1*OD*R +/S2*C1*/C0*OD*XH
      + /S2*/C1*/C0*/OD*XH + S2*/S1*/S0*C1*/C0*OD
      + S2*/S1*/S0*/C1*/C0*/OD) + CLK*(C1*/EV*R +/C1*EV*R
      + /S2*C1*/C0*EV*XH + /S2*/C1*/C0*/EV*XH
      + S2*/S1*/S0*C1*/C0*EV + S2*/S1*/S0*/C1*/C0*/EV))

GRAY = /HBLANK*/VBL*(/CLK*(S2*/S1*S0*/C1*/C0*/OD + S2*/S1*S0*C1*/C0*OD)
       + CLK*(S2*/S1*S0*/C1*/C0*/EV + S2*/S1*S0*C1*/C0*EV))

EXT = /HBLANK*/VBL*(/CLK*(S2*S1*/S0*/C1*/C0*/OD + S2*S1*/S0*C1*/C0*OD)
      + CLK*(S2*S1*/S0*/C1*/C0*/EV + S2*S1*/S0*C1*/C0*EV))

BLNK_WFM = /CLAMP * (HBLANK + VBL)

CBLANK = CLAMP + VBL
```