```
TITLE "Scancode to U2 conversion";
%--------------------------------------------------------------------
    PS/2 input module : ZDDD -> U2 converter.
%--------------------------------------------------------------------
INCLUDE "SC_TO_BCD.inc";

SUBDESIGN SC_TO_U2
(
    GEN                 : input;        -- 20MHz clock
    RESET               : input;        -- Reset signal from uC

    SC_Z[7..0]          : input;        -- Z
    SC_D0[7..0]         : input;        -- D
    SC_D1[7..0]         : input;        -- D
    SC_D2[7..0]         : input;        -- D

    U2_MS[7..0]         : output;       -- Output most significant (not
used)
    U2_LS[7..0]         : output;       -- Output less significant
)
VARIABLE
    SIGN                : DFF;          -- Sign of number
    S2B0                : SC_TO_BCD;    -- Digit 0 decoder
    S2B1                : SC_TO_BCD;    -- Digit 1 decoder
    S2B2                : SC_TO_BCD;    -- Digit 2 decoder
    BCD_D0[7..0]        : DFF;          -- Digit 0 in BCD
    BCD_D1[7..0]        : DFF;          -- Digit 1 in BCD
    BCD_D2[7..0]        : DFF;          -- Digit 2 in BCD
    D0[7..0]            : DFF;          -- Digit 0 multiplied
    D1[7..0]            : DFF;          -- Digit 1 multiplied
    D2[7..0]            : DFF;          -- Digit 2 multiplied
    BINARY[7..0]        : DFF;          -- Binary from ZDDD
    NEGATOR[8..0]       : DFF;          -- For binary negation
    NEGATED[8..0]       : DFF;          -- For binary negation
    NEGATION[8..0]      : DFF;          -- For binary negation
    RESULT[7..0]        : DFF;          -- Converter final output
BEGIN
    % Clock %
    SIGN.CLK            = GEN;
    BCD_D0[].CLK        = GEN;
    BCD_D1[].CLK        = GEN;
    BCD_D2[].CLK        = GEN;
    D0[].CLK            = GEN;
    D1[].CLK            = GEN;
    D2[].CLK            = GEN;
    BINARY[].CLK        = GEN;
    NEGATOR[].CLK       = GEN;
    NEGATED[].CLK       = GEN;
    NEGATION[].CLK      = GEN;
    RESULT[].CLK        = GEN;

    % Reset %
    SIGN.CLRN           = RESET;
    BCD_D0[].CLRN       = RESET;
    BCD_D1[].CLRN       = RESET;
```

```
    BCD_D2[].CLRN       = RESET;
    D0[].CLRN           = RESET;
    D1[].CLRN           = RESET;
    D2[].CLRN           = RESET;
    BINARY[].CLRN       = RESET;
    NEGATOR[].CLRN      = RESET;
    NEGATED[].CLRN      = RESET;
    NEGATION[].CLRN     = RESET;
    RESULT[].CLRN       = RESET;

    % SC to BCD %
    S2B0.SC[] = SC_D0[];
    BCD_D0[] = S2B0.BCD[];

    S2B1.SC[] = SC_D1[];
    BCD_D1[] = S2B1.BCD[];

    S2B2.SC[] = SC_D2[];
    BCD_D2[] = S2B2.BCD[];

    % Multiplication %
    D0[] = BCD_D0[];
    D1[] = BCD_D1[]*10;
    D2[] = BCD_D2[]*100;

    % Sum %
    BINARY[] = D0[] + D1[] + D2[];

    % Sign %
    if( SC_Z[] == H"4E" ) then
        -- Sign is "-" so convert to U2
        NEGATOR[] = B"100000000";
        NEGATED[] = (GND, BINARY[7..0]);
        NEGATION[] = NEGATOR[] - NEGATED[];
        RESULT[] = (NEGATION[7..0]);
    else
        RESULT[] = BINARY[];
    end if;

    % Return result %
    U2_LS[] = RESULT[];
    % Future extension? %
    U2_MS[] = (GND, GND, GND, GND, GND, GND, GND, GND);
END;
```