



```

; Divide the crystal oscillator frequency by two to clock the C011 link
; adapter
Clk5M := /Clk5M

; A state machine to increment the DAC address after each write (QValid),
; and to pulse LDAC and zero the address after 8 writes or if there is a
; read from the link adapter (IAck * IValid).

LDAC := /DnReset * /LDAC * A1 * A0 * CSMSB * WR * /QValid
      + /DnReset * /LDAC * /Iack * /QValid
      + /DnReset * /LDAC * /WR * /Iack * /QValid
      + /DnReset * LDAC * /A1 * /A0 * /CSMSB * /WR
      + /DnReset * LDAC * /A1 * /A0 * /CSMSB * /WR

A1 := /DnReset * /LDAC * A1 * /Iack * QValid
     + /DnReset * /LDAC * A1 * /WR * /Iack * QValid
     + /DnReset * /LDAC * /A1 * A0 * CSMSB * WR * /Iack * QValid
     + /DnReset * /LDAC * /A1 * A0 * CSMSB * WR * /Iack * QValid
     + /DnReset * /LDAC * A1 * /CSMSB * WR * /Iack * QValid
     + /DnReset * /LDAC * A1 * /IValid * QValid
     + /DnReset * /LDAC * A1 * /WR * /IValid * QValid
     + /DnReset * /LDAC * /A1 * A0 * CSMSB * WR * /IValid * QValid
     + /DnReset * /LDAC * A1 * /A0 * WR * /IValid * QValid
     + /DnReset * /LDAC * A1 * /CSMSB * WR * /IValid * QValid

A0 := /DnReset * /LDAC * /A0 * CSMSB * WR * /Iack * QValid
     + /DnReset * /LDAC * /A0 * /CSMSB * /Iack * QValid
     + /DnReset * /LDAC * /A0 * /WR * /Iack * QValid
     + /DnReset * /LDAC * /A0 * CSMSB * WR * /Iack * QValid
     + /DnReset * /LDAC * /A0 * CSMSB * WR * /Iack * QValid
     + /DnReset * /LDAC * /A0 * /CSMSB * /WR * /IValid * QValid
     + /DnReset * /LDAC * /A0 * /WR * /IValid * QValid
     + /DnReset * /LDAC * /A0 * CSMSB * WR * /IValid * QValid
     + /DnReset * /LDAC * /A0 * CSMSB * WR * /IValid * QValid

CSMSB := /DnReset * /LDAC * /CSMSB * WR * /Iack * QValid
        + /DnReset * /LDAC * CSMSB * /WR * /Iack * QValid
        + /DnReset * /LDAC * CSMSB * /WR * /Iack * QValid
        + /DnReset * /LDAC * CSMSB * WR * QValid
        + /DnReset * LDAC * /A1 * /A0 * /CSMSB * /WR
        + /DnReset * LDAC * /A1 * /A0 * /CSMSB * /WR
        + /DnReset * LDAC * /CSMSB * WR * /IValid * QValid
        + /DnReset * LDAC * CSMSB * /WR * /IValid * QValid
        + /DnReset * LDAC * CSMSB * WR * /IValid * QValid

WR := /DnReset * /LDAC * WR * /Iack * QValid
     + /DnReset * /LDAC * WR * /Iack * QValid
     + /DnReset * /LDAC * WR * /IValid * QValid

; Other outputs

; Wait until a byte is received (QValid) before sending a byte (IValid),
; to avoid having a transputer interpret it as the start of a boot after
; a reset.
IValid := /DnReset * /IValid * /Iack * QValid
        + /DnReset * /IValid * /Iack

; The eqn for /CSLSB must be identical to that for CSMSB
/CSLSB := /DnReset * /LDAC * /CSMSB * WR * /Iack * QValid
        + /DnReset * /LDAC * CSMSB * /WR * /Iack * QValid
        + /DnReset * /LDAC * CSMSB * /WR * /Iack * QValid
        + /DnReset * /LDAC * CSMSB * WR * QValid
        + /DnReset * LDAC * /A1 * /A0 * /CSMSB * /WR
        + /DnReset * LDAC * /A1 * /A0 * /CSMSB * /WR
        + /DnReset * LDAC * /CSMSB * WR * /IValid * QValid
        + /DnReset * LDAC * CSMSB * /WR * /IValid * QValid
        + /DnReset * LDAC * CSMSB * WR * /IValid * QValid

; The eqn for QAck must be identical to that for WR
QAck := /DnReset * /LDAC * WR * /Iack * QValid
      + /DnReset * /LDAC * WR * /Iack * QValid
      + /DnReset * /LDAC * WR * /IValid * QValid

LAReset = DnReset

```

Max data rate 960kbyte/s
(using 20Mbit/s link)
= 120k samples/s on 4 channels

Pins	J1,J2	J6	U1	U2	U3	U4	U5
VCC	1,2	1,6	28	4	24	1	
GND	3,4,5	3,4	14	2	12	7	2

Transputer Link 4 Channel D to A
 © Copyright 1994 Dave Keenan
 Smalltalk Computing, 116 Bowman Parade, Bardon QLD 4065, Australia.
 Tel: +61 7 366 2660, Fax: +61 7 366 2660,
 Email: davek@peg.pegasus.oz.au

Revision History:
 31-Mar-94 DK, Commenced
 28-May-94 DK, Completed version 1.0
 1-Jun-94 DK, Updated to version 1.1

