

TITLE "NS486SXF EVB - PC104 I/F - Rev D";

```
-- C7X, 07NOV96 Complete desired rev. D board functionality with p_enal/ and p_ena2/
-- Includes:
--     Support for PCMCIA I/O access to overlap PC104SEL/ i/o space
--     DMA from flash and to/from PCMCIA by disabling read buffer
--     Optimized timing (in RDY) for PC/104 using 2 ready extensions on SXF
--     Added two stage delay to ready to get full 120 ns. of ready extension
--     Masking of PCMRDY into ready cycle if not in PCMCIA cycle
--     Eliminate RDY circuitry for on board UART operations
--
-- D, 13NOV96 Release of Rev. D PCB
-- Remove UART frequency divider
```

```
INCLUDE "rdy";
INCLUDE "74163";
```

```
-- Subdesign Section
SUBDESIGN evb
```

```
(
    iochrdy, pcmrdy                : INPUT;
    c_iow/, c_ior/, c_memr/, c_memw/ : INPUT;    -- SXF strobes in
    endxfr/                        : INPUT;      -- PC/104 end transfer (0ws)
    clock                          : INPUT;      -- Sysclk from SXF
    iocsl6/, memcs16/              : INPUT;      -- Size control from PC/104
    clkis50                        : INPUT;      -- Asserted for 50 MHz., else 40
    reset                          : INPUT;      -- Reset
    pc104sel/                      : INPUT;      -- Chip select for PC/104
    cs0/                           : INPUT;      -- Boot chip select
    sbhe/                          : INPUT;      -- Used to determine boot location
    meglen/                        : INPUT;      -- Chip select for 1st meg of PC/104
    uarten/                        : INPUT;      -- Chip select for uart
    c_dack/                        : INPUT;      -- DMA Acknowledge (combined)
    p_enal/, p_ena2/              : INPUT;      -- PCMCIA Enables
    c_rdy                          : OUTPUT;     -- Ready to the SXF
    c_cs16/                       : BIDIR;      -- Open collector CS16/
    sysclk                         : OUTPUT;     -- Synchronous sysclk for PC/104
    flash_sel/                    : OUTPUT;     -- Flash chip select out
    smemr/, smemw/                : OUTPUT;     -- Lower 1 meg memory strobes
    memr/, memw/                  : OUTPUT;     -- PC/104 16-bit mem strobes
    db_rden/, db_wren/            : OUTPUT;     -- Read/write enables for data buffer
    db_len/                       : OUTPUT;     -- Latch enable for output data buffer
    aen                           : OUTPUT;     -- Address Enable (for PC/104 dma)

    w10, w11                      : INPUT;      -- Spare pins on jumpers
)
```

```
-- Variable Section
VARIABLE
```

```
-- Node Declaration
pc104en      : NODE;
pc104boot, flashboot : NODE;
enable_8bit_strobes : NODE;
readstrobe   : NODE;
enb_dmareadbuf : NODE;
-- Instance Declaration
evb_rdy      : rdy;
clk_ctr      : 74163;

-- Register Declaration
sbhe_delay[3..1] : dff;
```

```
-- Logic Section
BEGIN
```

```
-- Sysclk counter
clk_ctr.clk      = clock;
clk_ctr.ldn      = ~(clk_ctr.qb & clk_ctr.qc);
clk_ctr.a        = VCC;
clk_ctr.b        = GND;
clk_ctr.c        = GND;
clk_ctr.d        = GND;
sysclk           = clk_ctr.qc;
```

```
-- Boot determination logic
-- Also share the first two flip flops as ready delay circuitry during non-reset (operational)
sbhe_delay[2..1].clk = clock;
sbhe_delay[3].clk    = !reset;
sbhe_delay[1].d      = (reset & sbhe/) # (!reset & ((pcmrdy # !(p_ena1/ # !p_ena2/)) & iochrdy));
sbhe_delay[2].d      = sbhe_delay[1].q;
sbhe_delay[3].d      = sbhe_delay[2].q;
flashboot            = sbhe_delay[3].q;  -- flash boot if sbhe/ held high!!!!
pc104boot             = !flashboot;      -- else boot from on board flash

-- flash sel signal, only during cs0 for booting, user other CS when PC/104 boot
flash_sel/           = !((pc104boot & !meglen/) # (flashboot & !cs0/));

-- smemr & smemw strobes are the 8-bit strobes on the PC/104 bus
enable_8bit_strobes = (pc104boot & !cs0/) # (flashboot & !meglen/);

-- the PC/104 ready circuitry, data buffer and 16-bit strobes are enabled by pc104en
pc104en               = (!pc104sel/ # (pc104boot & !cs0/))
                      & !(p_ena1/ # !p_ena2/ # !uarten/);

-- Connect up rdy circuitry;
evb_rdy.iocrdy        = sbhe_delay[2] # reset;  -- input the delayed ready, for ready during reset
evb_rdy.pcmrddy       = sbhe_delay[2] # reset;  -- just in case SXF doesn't ignore ready during no
n-cycles
evb_rdy.iow/          = c_iow/;
evb_rdy.ior/          = c_ior/;
evb_rdy.memr/         = c_memr/;
evb_rdy.memw/         = c_memw/;
evb_rdy.sel/          = !pc104en;
evb_rdy.reset/        = !reset;
evb_rdy.0ws/          = endxfr/;
evb_rdy.rdyclk        = clock;
evb_rdy.iocsl6/       = iocsl6/;
evb_rdy.memcs16/      = memcs16/;
evb_rdy.clkis50       = clkis50;

-- The output pins:
-- rdy logic outputs
c_rdy                 = evb_rdy.rdy;
c_cs16/               = TRI( evb_rdy.e_cs16/, !evb_rdy.e_cs16/ );

smemr/                = !(enable_8bit_strobes & !c_memr/);
smemw/                = !(enable_8bit_strobes & !c_memw/);
memr/                 = !(pc104en & !c_memr/);
memw/                 = !(pc104en & !c_memw/);

readstrobe            = !c_memr/ # !c_ior/;  -- (not an output);
enb_dmareadbuf        = !c_dack/ & !(p_ena1/ # !p_ena2/ # !flash_sel/);
db_rden/              = !(readstrobe & (pc104en # enb_dmareadbuf));
db_wren/              = readstrobe;
db_len/               = !(c_memw/ # !c_iow/);

aen                   = !(pc104en & c_dack/);
```

END;

TITLE "RDY - NS486SXF EVB Ready Circuitry - Rev C7X 07NOV96";

SUBDESIGN rdy

```
(
    iocrdy, pcmrdy          : INPUT;
    iow/, ior/, memr/, memw/ : INPUT;
    sel/                   : INPUT;    -- Strobe qualifier
    reset/                 : INPUT;
    0ws/, iocsl6/, memcs16/ : INPUT;    -- PC/104 Inputs
    rdyclk                 : INPUT;    -- clock
    clkis50                : INPUT;    -- 1 for 50M, 0 for 40M
    rdy, e_cs16/           : OUTPUT;    -- outputs for SXF
)
```

VARIABLE

```
-- Node Declaration
bytecyc, wordcyc          : node;
-- distinguish between byte and word access

wait                      : DFF;      -- Wait signal to SXF
waits                    : DFF;      -- Synchronized wait
endxfr                   : DFF;      -- Sync version of 0ws
endwaitd                 : DFF;      -- Delayed endwait

meml6, iol6              : node;      -- 16-bit memory, i/o cycle

wordio, wordmem, bytememio : node;
endwait                  : node;      -- End of wait

-- Instance Declaration
ctr[4..0]                 : DFF;
```

-- Logic Section

```
BEGIN
    ctr[].clrn = reset/;
    ctr[].clk  = rdyclk;
    IF waits
    THEN
        ctr[].d = ctr[].q + 1;
    ELSE
        ctr[].d = B"00000";
    END IF;

    -- 8/16 selection logic
    iol6 = !iocsl6/ & (!ior/ # !iow/);
    meml6 = !memcs16/ & (!memr/ # !memw/);
    wordcyc = iol6 # meml6;
    bytecyc = !wordcyc;

    e_cs16/ = !wordcyc;

    -- Delay decode logic
    wordio = iol6 & ((clkis50 & (ctr[2..0] == 4)) # (!clkis50 & (ctr[1..0] == 2)))
;
    wordmem = meml6 & ((clkis50 & (ctr[2..0] == 7)) # (!clkis50 & (ctr[2..0] == 5)))
;
    bytememio = bytecyc & ((clkis50 & (ctr[] == 22)) # (!clkis50 & (ctr[] == 17)));

    endwait = wordio # wordmem # bytememio # endxfr.q;

    -- zero wait state logic
    endxfr.d = meml6 & !0ws/;
    endxfr.clk = rdyclk;

    wait.clk = !sel/ & (!iow/ # !ior/ # !memr/ # !memw/);
    wait.d = VCC;
    wait.clrn = !(reset/ # endwaitd.q);

    endwaitd.d = endwait;
    endwaitd.clk = rdyclk;

    waits.d = wait.q;
    waits.clk = rdyclk;

    rdy = !wait.q & iocrdy & pcmrdy;
END;
```

Project Information y:\my briefcase\usr\cust\nsc\evb\max2work\max7\d\evb.rpt

MAX+plus II Compiler Report File

Version 7.0 08/26/96

Compiled: 12/13/96 10:18:39

Copyright (c) Altera Corporation, 1996. This file may contain proprietary and confidential information of Altera Corporation that may be used, copied, and disclosed only pursuant to the terms of Altera's Program License Agreement. Altera makes no claim to any end-user or third-party proprietary information that also may be contained in this file. This notice must be contained as part of this text at all times.

***** Project compilation was successful

NS486SXF EVB - PC104 I/F - Rev D

** DEVICE SUMMARY **

Chip/ POF	Device	Input Pins	Output Pins	Bidir Pins	LCs	Shareable Expanders	% Utilized
evb	EPM7032LC44-6	23	12	1	28	9	87 %
User Pins:		22	11	1			

** PROJECT COMPILATION MESSAGES **

Info: Reserved unused input pin 'w10' for future use because it has a pin assignment--pin is tri-stated and must be connected to your board

Info: Reserved unused input pin 'w11' for future use because it has a pin assignment--pin is tri-stated and must be connected to your board

** AUTO GLOBAL SIGNALS **

INFO: Signal 'clock' chosen for auto global Clock

** MULTIPLE PIN CONNECTIONS **

For node name '~174~2' (Same as node '~PIN001')

For node name '~174~1' (Same as node '~PIN002')

Connect: {evb@44, evb@18}

Project Information y:\my briefcase\usr\cust\nsc\evb\max2work\max7\d\evb.rpt

** PIN/LC/CHIP ASSIGNMENTS **

User Assignments	Actual Assignments (if different)	Node Name
evb@13		aen
evb@39		c_cs16/
evb@34		c_dack/
evb@5		c_ior/
evb@4		c_iow/
evb@26		clkis50
evb@43		clock
evb@2		c_memr/
evb@1		c_memw/
evb@38		c_rdy
evb@31		cs0/
evb@37		db_len/
evb@16		db_rden/
evb@36		db_wren/
evb@33		endxfr/
evb@17		flash_sel/
evb@14		iochrdy
evb@6		iocs16/
evb@8		meglen/
evb@9		memcs16/
evb@20		memr/
evb@21		memw/
evb@11		pcmrdy
evb@12		pc104sel/
evb@28		p_ena1/
evb@19		p_ena2/
evb@32		reset
evb@24		sbhe/
evb@41		smemr/
evb@40		smemw/
evb@7		sysclk
evb@29		uarten/
evb@25		w10
evb@27		w11

** FILE HIERARCHY **

```
|rdy:evb_rdy|
|74163:clk_ctr|
|74163:clk_ctr|p74163:sub|
```

***** Logic for device 'evb' compiled without errors.

Device: EPM7032LC44-6

Turbo: ON

Security: OFF

** ERROR SUMMARY **

Info: Chip 'evb' in device 'EPM7032LC44-6' has less than 20% of pins available for future logic changes -- if your project is likely to change, Altera recommends using a larger device

Device-Specific Information:y:\my briefcase\usr\cust\nsc\evb\max2work\max7\d\evb.rpt
evb

** INPUTS **

Pin	LC	LAB	Primitive	Code	Shareable			Fan-In		Fan-Out		Name
					Total	Shared	n/a	INP	FBK	OUT	FBK	
39	19	B	BIDIR	!	0	0	0	6	0	1	2	c_cs16/
34	(23)	(B)	INPUT		0	0	0	0	0	2	0	c_dack/
5	(2)	(A)	INPUT		0	0	0	0	0	2	2	c_ior/
4	(1)	(A)	INPUT		0	0	0	0	0	2	2	c_iow/
26	(30)	(B)	INPUT		0	0	0	0	0	0	2	clkis50
43	-	-	INPUT		0	0	0	0	0	0	0	clock
2	-	-	INPUT		0	0	0	0	0	4	2	c_memr/
1	-	-	INPUT		0	0	0	0	0	4	2	c_memw/
31	(26)	(B)	INPUT		0	0	0	0	0	7	1	cs0/
33	(24)	(B)	INPUT		0	0	0	0	0	0	1	endxfr/
14	(10)	(A)	INPUT		0	0	0	0	0	0	1	iochrdy
6	(3)	(A)	INPUT		0	0	0	0	0	1	2	iocs16/
8	(5)	(A)	INPUT		0	0	0	0	0	3	0	meglen/
9	(6)	(A)	INPUT		0	0	0	0	0	1	3	memcs16/
11	(7)	(A)	INPUT		0	0	0	0	0	0	1	pclrdrdy
12	(8)	(A)	INPUT		0	0	0	0	0	4	1	pc104sel/
28	(28)	(B)	INPUT		0	0	0	0	0	4	2	p_enal/
19	(14)	(A)	INPUT		0	0	0	0	0	4	2	p_ena2/
44	-	-	INPUT	s	0	0	0	0	0	0	0	~PIN001
32	(25)	(B)	INPUT		0	0	0	0	0	1	8	reset
24	(32)	(B)	INPUT		0	0	0	0	0	0	1	sbhe/
29	(27)	(B)	INPUT		0	0	0	0	0	4	1	uarten/
25	(31)	(B)	INPUT		0	0	0	0	0	0	0	w10
27	(29)	(B)	INPUT		0	0	0	0	0	0	0	w11

Code:

s = Synthesized pin or logic cell
t = Turbo logic cell
+ = Synchronous flipflop
! = NOT gate push-back
r = Fitter-inserted logic cell

** OUTPUTS **

Pin	LC	LAB	Primitive	Code	Shareable			Fan-In		Fan-Out		Name
					Total	Shared	n/a	INP	FBK	OUT	FBK	
13	9	A	OUTPUT	t !	0	0	0	6	1	0	0	aen
39	19	B	TRI	t !	0	0	0	6	0	1	2	c_cs16/
38	20	B	OUTPUT	t	0	0	0	1	2	0	0	c_rdy (rdy:evb_rdy
:163)												
37	21	B	OUTPUT	t !	0	0	0	2	0	0	1	db_len/
16	11	A	OUTPUT	t !	0	0	0	6	3	0	0	db_rden/
36	22	B	OUTPUT	t	0	0	0	2	0	1	1	db_wren/
17	12	A	OUTPUT	t !	0	0	0	2	1	1	0	flash_sel/
20	15	A	OUTPUT	t !	0	0	0	6	1	0	0	memr/
21	16	A	OUTPUT	t !	0	0	0	6	1	0	0	memw/
18	13	A	OUTPUT	t	0	0	0	0	1	0	0	~PIN002
41	17	B	OUTPUT	t !	0	0	0	3	1	0	0	smemr/
40	18	B	OUTPUT	t !	0	0	0	3	1	0	0	smemw/
7	4	A	FF	+ t	0	0	0	0	3	1	2	sysclk (74163:clk_c
tr p74163:sub :36)												

Code:

s = Synthesized pin or logic cell
t = Turbo logic cell
+ = Synchronous flipflop
! = NOT gate push-back
r = Fitter-inserted logic cell