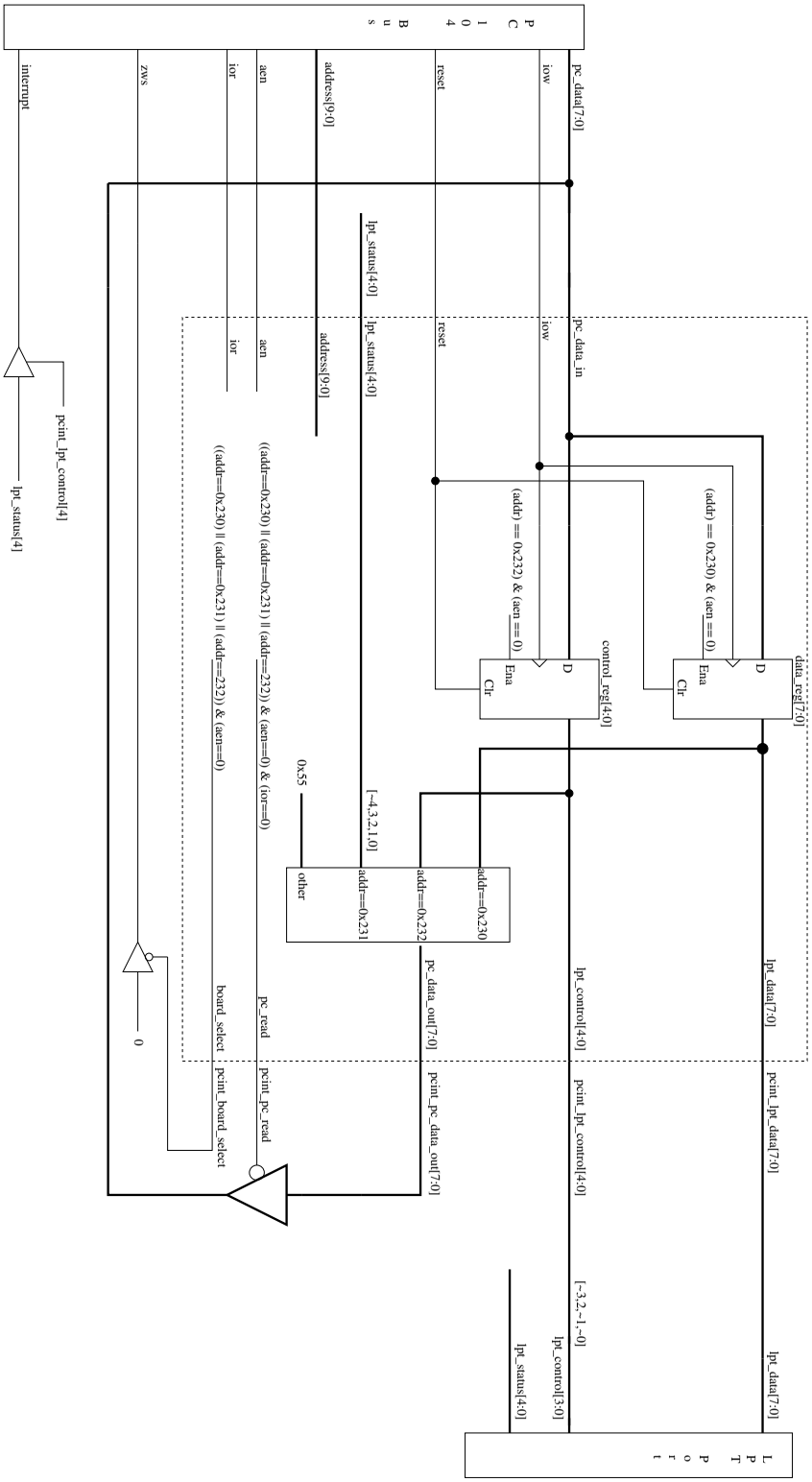
Parallel Port Block Diagram



```
/*-----------------------------------------------------------------------
    constants.v - file to hold all the global constants

    Bill Rison 1/30/12
-----------------------------------------------------------------------*/

/*-- Address constants  -- */
parameter LPT_DATA_ADDR     =10'h230;
parameter LPT_STATUS_ADDR   =10'h231;
parameter LPT_CONTROL_ADDR  =10'h232;

parameter DEFAULT_OUT       = 8'h55; // if nothing cool is happening
```

```
/*********************************************************************
    parallel.v - Parallel port replacement Verilog code

    Bill Rison, 1/30/11
*********************************************************************/

module parallel (
    /* The following are the PC Interface lines */
    inout [7:0] pc_data,        // data to/from PC
    input [9:0] address,        // PC address lines
    input ior, iow, aen, reset, // control lines from PC
    inout interrupt,           // Interrupt output
    inout io_cs16,
    inout zws,

    /* The board has a 25 MHz clock */
    input sys_clock,            // 25 Mhz system clock

    input [4:0] lpt_status,
    output [7:0] lpt_data,
    output [3:0] lpt_control,

    /* Leave the following lines alone.  They are wired to other components on
     * the board */
    inout [7:0] HC11_data,
    input [11:0] sample_data,
    input fifo1_ef,fifo1_hf,fifo1_ff,
    input fifo2_ef,fifo2_hf,fifo2_ff,
    input gps,
    input HC11_strb,
    input [3:0] time_select,
    input write_time,
    output [15:0] fifo_data,
    output fifo1_r,fifo1_w,fifo1_rst,
    output fifo2_r,fifo2_w,fifo2_rst,
    output fifo_oe,
    output HC11_stra,
    output HC11_reset,
    output test_pulse,
    output local_gps

);

`include "constants.v"

/* Wires out of PC Interface  */
wire [7:0] pcint_pc_data_out;
wire [7:0] pcint_lpt_data;
```

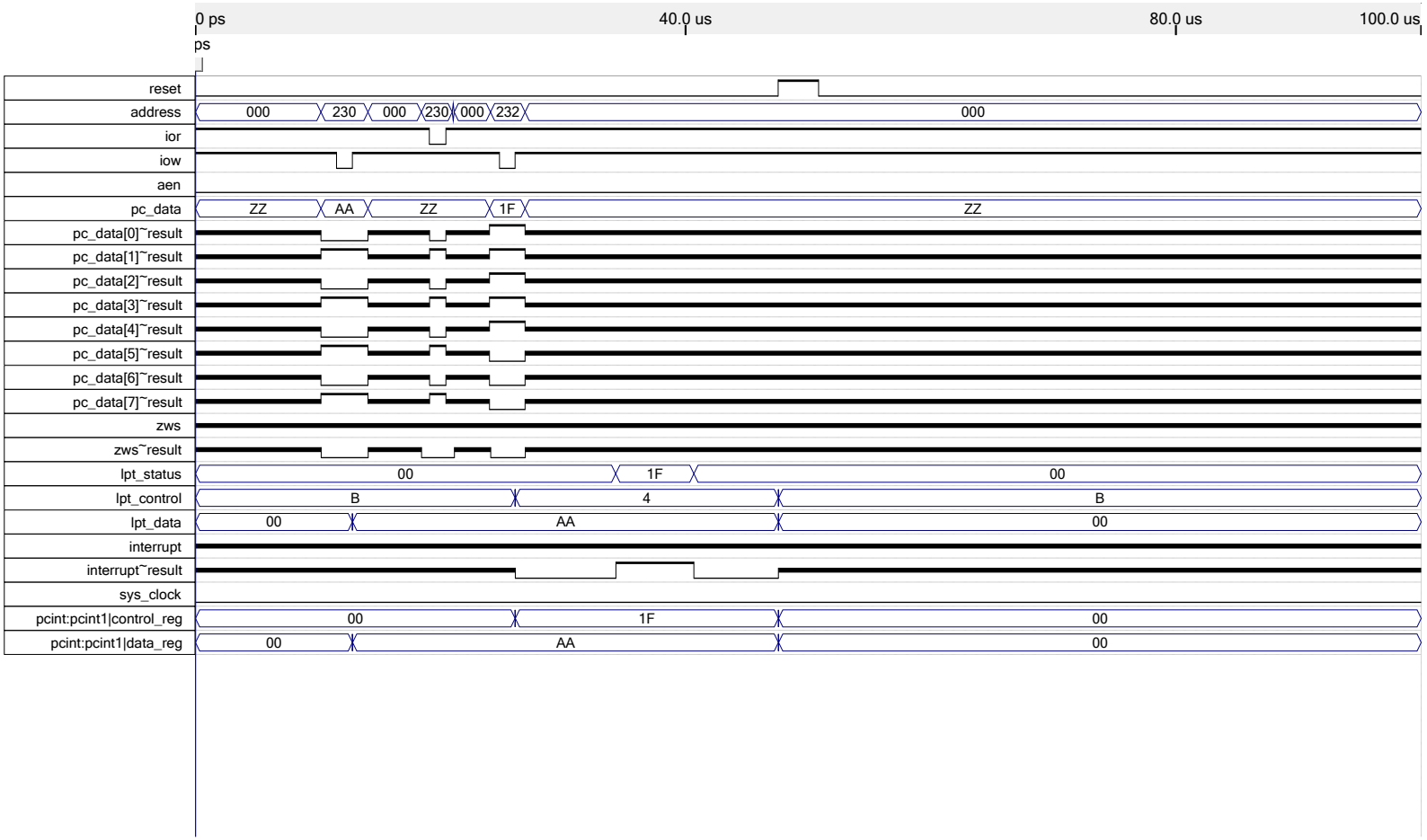
```
wire [4:0] pcint_lpt_control;
wire pcint_pc_read;
wire pcint_board_select;

pcint pcint1 (
    .address      (address),
    .pc_data_in (pc_data),
    .ior (ior), .iow (iow), .aen (aen), .reset (reset),
    .lpt_status (lpt_status),

    .pc_data_out (pcint_pc_data_out),
    .lpt_data (pcint_lpt_data),
    .lpt_control (pcint_lpt_control),
    .pc_read (pcint_pc_read),
    .board_select (pcint_board_select)
);

assign interrupt = (pcint_lpt_control[4]) ? lpt_status[3] : 1'bz;           // Interrupt output
assign zws = pcint_board_select ? 1'bz : 1'b0;
assign pc_data[7:0] =  pcint_pc_read ? 8'hzz : pcint_pc_data_out[7:0];
assign lpt_data[7:0] = pcint_lpt_data[7:0];
assign lpt_control[3:0] = {~pcint_lpt_control[3],pcint_lpt_control[2],~pcint_lpt_control[1:0]};

assign io_cs16 = 1'bz;

/* Unused outputs */
assign HC11_data =  8'hzz;
assign fifo_data = 15'h0000;
assign fifo1_r = 1'b0;
assign fifo1_w = 1'b0;
assign fifo1_rst = 1'b0;
assign fifo2_r = 1'b0;
assign fifo2_w = 1'b0;
assign fifo2_rst = 1'b0;
assign fifo_oe = 1'b0;
assign HC11_stra = 1'b0;
assign HC11_reset = 1'b0;
assign test_pulse = 1'b0;
assign local_gps = 1'b0;

endmodule
```

```
module pcint(
  input [9:0] address,                          // PC 104 address
  input [7:0] pc_data_in,                       // Data from PC 104 bus
  input ior, iow, aen, reset,                         // Control from PC 104 bus

  input [4:0] lpt_status,                      // External status lines

  output [7:0] pc_data_out,                    // Data to write to PC 104 bus
  output [7:0] lpt_data,                       // Data to put on LPT_DATA lines
  output [4:0] lpt_control,                    // External control lines
  output pc_read, board_select                 // When to put data on PC 104 bus
);

`include "constants.v"

reg [4:0] control_reg;
reg [7:0] data_reg;

assign board_select =
       (( (address[9:0] == LPT_DATA_ADDR) ||
          (address[9:0] == LPT_STATUS_ADDR) ||
          (address[9:0] == LPT_CONTROL_ADDR)) && (aen == 1'b0)) ?
          1'b0 : 1'b1;

assign pc_read =
       (( (address[9:0] == LPT_DATA_ADDR) ||
          (address[9:0] == LPT_STATUS_ADDR) ||
          (address[9:0] == LPT_CONTROL_ADDR)) && (aen == 1'b0) && (ior == 1'b0)) ?
           1'b0 : 1'b1;


/* Write to threshold and control registers on rising edge of IOW */
always @(posedge reset or posedge iow) begin
    if (reset == 1'b1)  begin
            data_reg[7:0] <= 8'h00;
            control_reg[4:0] <= 5'h00;
    end
    else begin
        if ((aen == 1'b0) && (address[9:0] == LPT_DATA_ADDR))
                data_reg[7:0] <= pc_data_in[7:0];

        if ((aen == 1'b0) && (address[9:0] == LPT_CONTROL_ADDR))
             control_reg[4:0] <= pc_data_in[4:0];
    end
end

/* Read registers by setting pc_data_out for appropriate addresses */
```

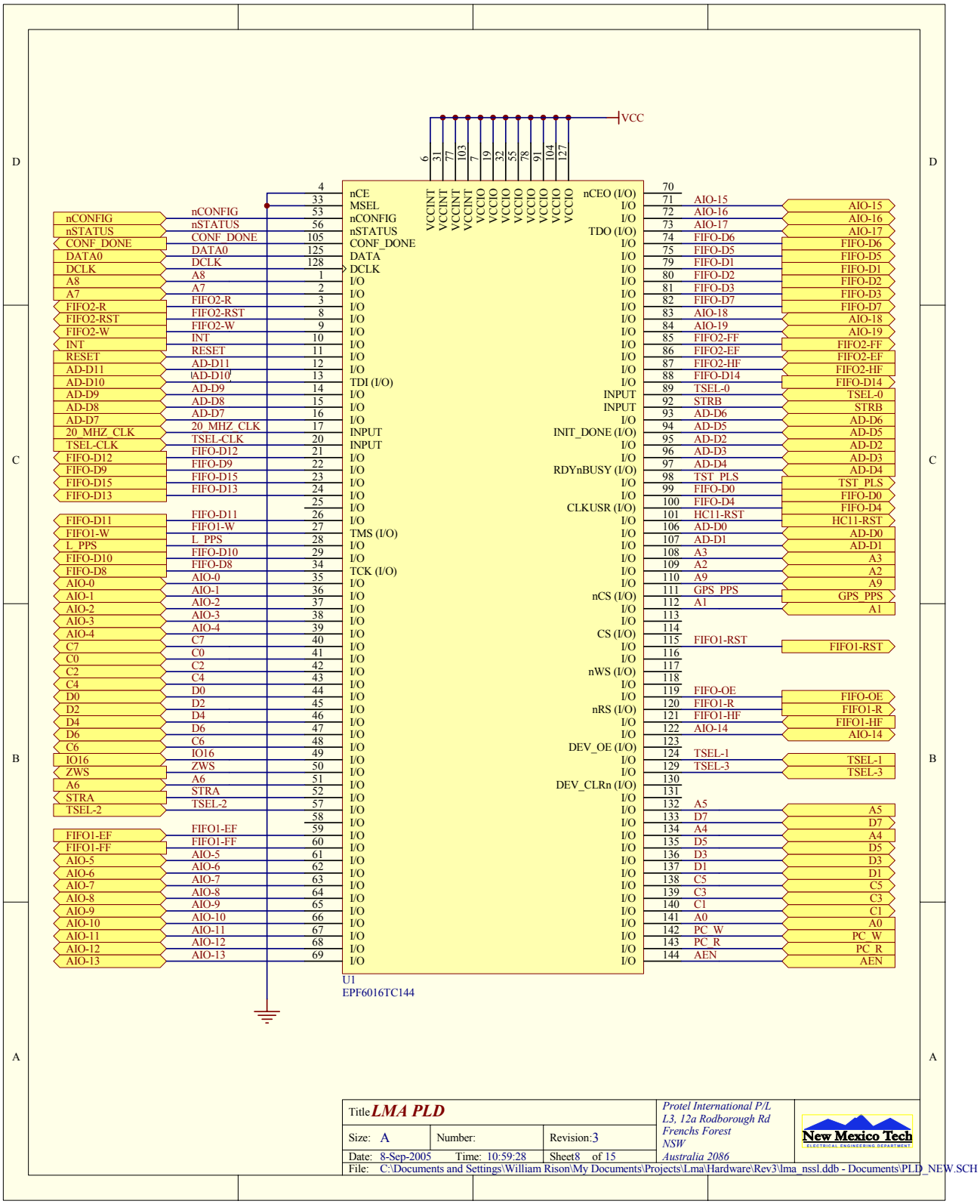
```
assign pc_data_out[7:0] = (address[9:0] == LPT_DATA_ADDR) ? data_reg[7:0] :
                          (address[9:0] == LPT_CONTROL_ADDR) ? {3'b111, control_reg[4:0]} :
                          (address[9:0] == LPT_STATUS_ADDR) ?
                               {~lpt_status[4], lpt_status[3:0],3'b111} :
                          DEFAULT_OUT;

assign lpt_data = data_reg;
assign lpt_control = control_reg;

endmodule
```

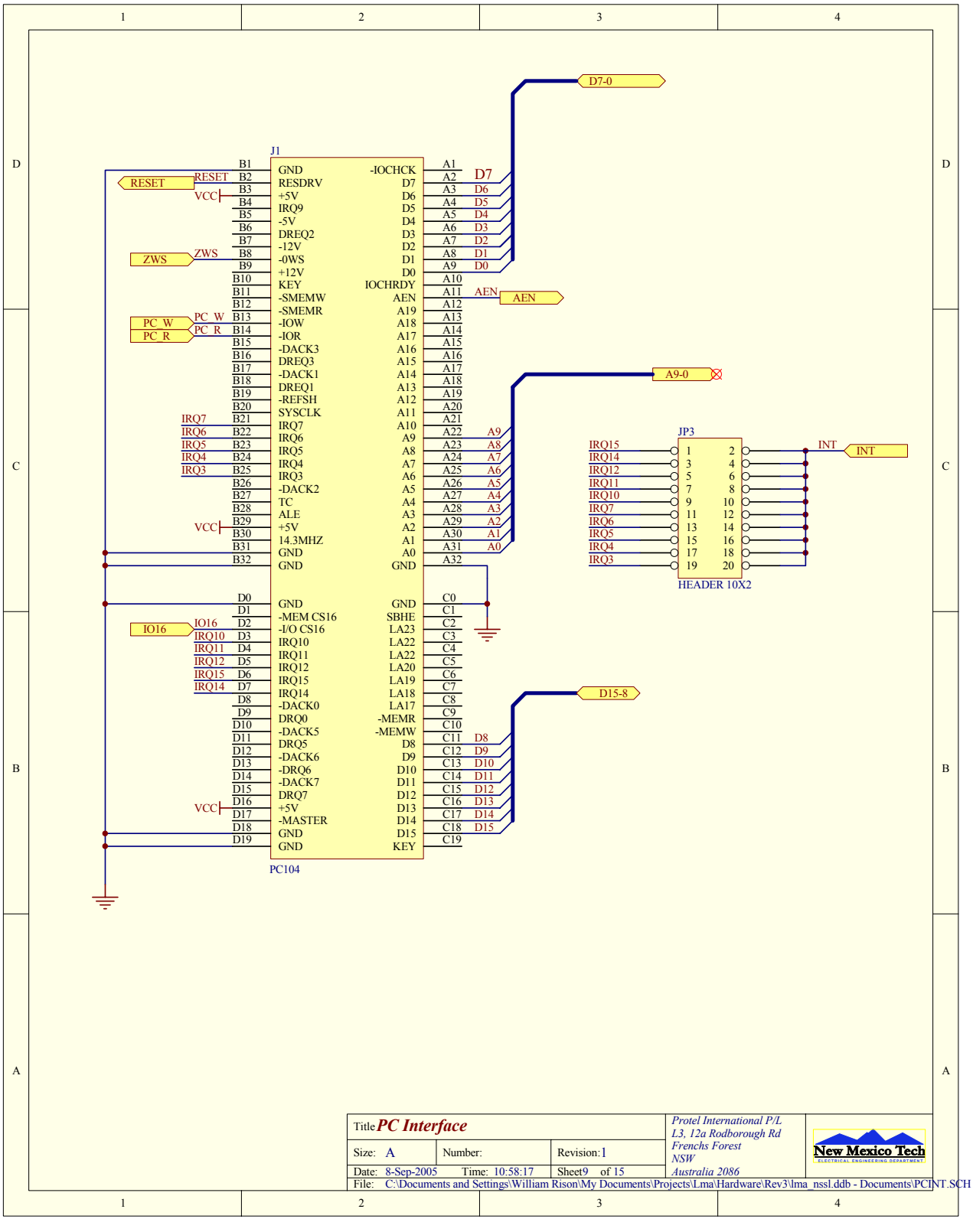
```
%************************************************************************
    parallel.inc - This file contains the macro function definitions used in the
              parallel.v file.
    Bill Rison, 1/30/12
************************************************************************%

%* Macro Function prototypes *%

% PC Interface Module %
FUNCTION pcint
    (
    address[9..0],
    pc_data_in[7..0],
    ior,
    iow,
    aen,
    reset,
    lpt_status[4:0]
    )
    RETURNS
    (
        pc_data_out[7..0],
        lpt_data[7..0],
        lpt_control[4..0],
        pc_read,
        board_select
    );
```

Date: January 31, 2012 parallel.vwf Project: parallel



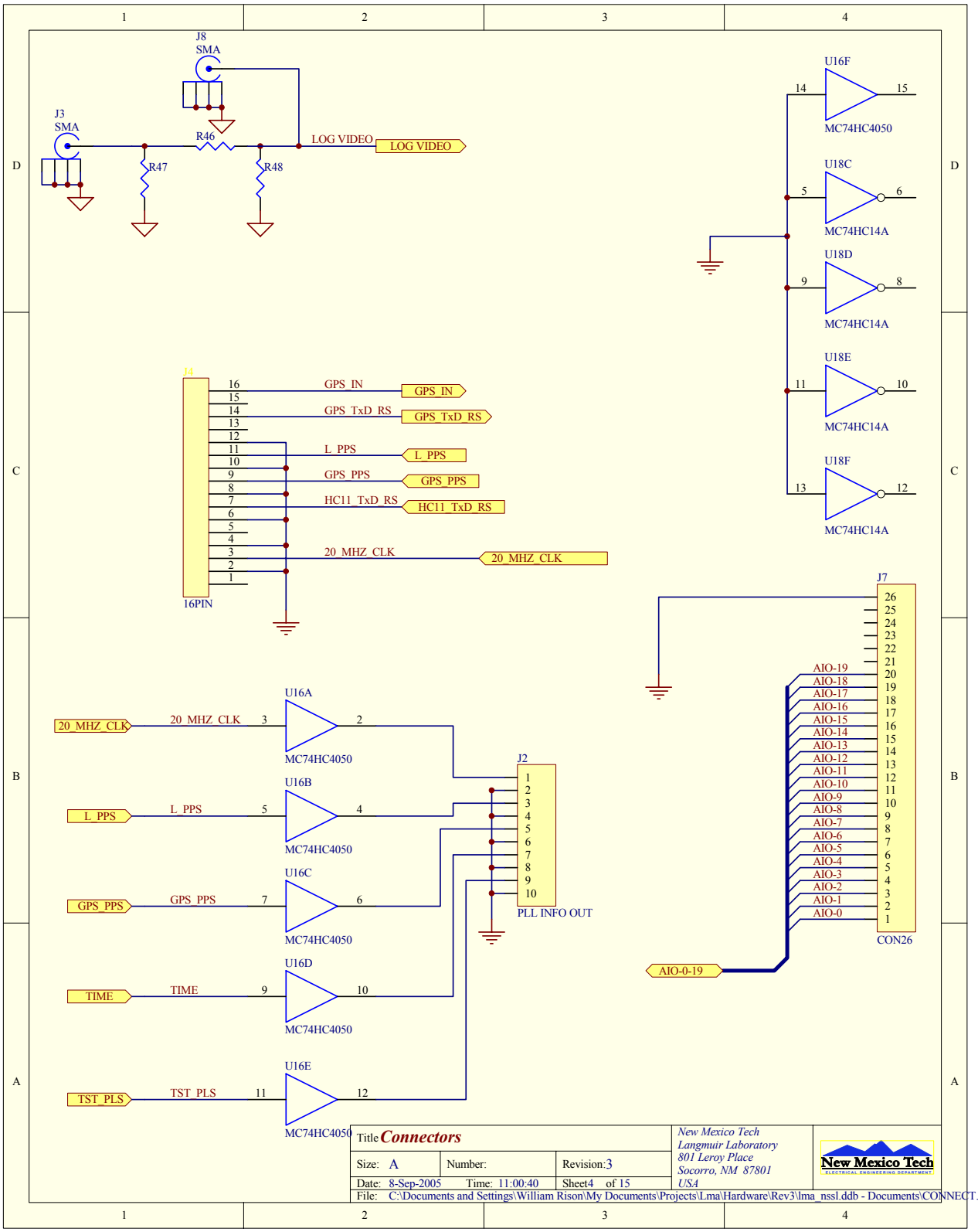
Revision: parallel


VCC
U1
EPF6016TC144
nCONFIG
nSTATUS
CONF_DONE
DATA0
DCLK
A8
A7
FIFO2-R
FIFO2-RST
FIFO2-W
INT
RESET
AD-D11
AD-D10
AD-D9
AD-D8
AD-D7
20_MHZ_CLK
TSEL-CLK
FIFO-D12
FIFO-D9
FIFO-D15
FIFO-D13
FIFO-D11
FIFO1-W
L_PPS
FIFO-D10
FIFO-D8
AIO-0
AIO-1
AIO-2
AIO-3
AIO-4
C7
C0
C2
C4
D0
D2
D4
D6
C6
IO16
ZWS
A6
STRA
TSEL-2
FIFO1-EF
FIFO1-FF
AIO-5
AIO-6
AIO-7
AIO-8
AIO-9
AIO-10
AIO-11
AIO-12
AIO-13
AIO-15
AIO-16
AIO-17
FIFO-D6
FIFO-D5
FIFO-D1
FIFO-D2
FIFO-D3
FIFO-D7
AIO-18
AIO-19
FIFO2-FF
FIFO2-EF
FIFO2-HF
FIFO-D14
TSEL-0
STRB
AD-D6
AD-D5
AD-D2
AD-D3
AD-D4
TST_PLS
FIFO-D0
FIFO-D4
HC11-RST
AD-D0
AD-D1
A3
A2
A9
GPS_PPS
A1
FIFO1-RST
FIFO-OE
FIFO1-R
FIFO1-HF
AIO-14
TSEL-1
TSEL-3
A5
D7
A4
D5
D3
D1
C5
C3
C1
A0
PC_W
PC_R
AEN
Title LMA PLD
Size: A
Number:
Revision:3
Date: 8-Sep-2005
Time: 10:59:28
Sheet8 of 15
File: C:\Documents and Settings\William Rison\My Documents\Projects\Lma\Hardware\Rev3\lma_nssl.ddb - Documents\PLD_NEW.SCH
Protel International P/L
L3, 12a Rodborough Rd
Frenchs Forest
NSW
Australia 2086
New Mexico Tech
ELECTRICAL ENGINEERING DEPARTMENT



Title **PC Interface**			*Protel International P/L* *L3, 12a Rodborough Rd* *Frenchs Forest* *NSW* *Australia 2086*
Size: A	Number:	Revision: 1	
Date: 8-Sep-2005	Time: 10:58:17	Sheet 9 of 15	
File: C:\Documents and Settings\William Rison\My Documents\Projects\Lma\Hardware\Rev3\lma_nssl.ddb - Documents\PCINT.SCH			

New Mexico Tech
ELECTRICAL ENGINEERING DEPARTMENT


J8
SMA
J3
SMA
R46
R47
R48
LOG VIDEO
LOG VIDEO
J4
16PIN
GPS_IN
GPS_TxD_RS
L_PPS
GPS_PPS
HC11_TxD_RS
20_MHZ_CLK
20_MHZ_CLK
U16A
U16B
U16C
U16D
U16E
MC74HC4050
20_MHZ_CLK
L_PPS
GPS_PPS
TIME
TST_PLS
J2
PLL INFO OUT
U16F
U18C
U18D
U18E
U18F
MC74HC4050
MC74HC14A
J7
CON26
AIO-19
AIO-18
AIO-17
AIO-16
AIO-15
AIO-14
AIO-13
AIO-12
AIO-11
AIO-10
AIO-9
AIO-8
AIO-7
AIO-6
AIO-5
AIO-4
AIO-3
AIO-2
AIO-1
AIO-0
AIO-0-19
Title Connectors
Size: A
Number:
Revision:3
Date: 8-Sep-2005
Time: 11:00:40
Sheet4 of 15
File: C:\Documents and Settings\William Rison\My Documents\Projects\Lma\Hardware\Rev3\lma_nssl.ddb - Documents\CONNECT.SCH
New Mexico Tech
Langmuir Laboratory
801 Leroy Place
Socorro, NM 87801
USA
New Mexico Tech
ELECTRICAL ENGINEERING DEPARTMENT