LatticeECP-DSP sysDSP Usage Guide

September 2012 | Technical Note TN1057

Introduction

This technical note discusses how to access the features of the LatticeECP™-DSP sysDSP™ (Digital Signal Processing) Block described in the LatticeECP/EC Family data sheet. Designs targeting the sysDSP Block offer significant improvement over traditional LUT-based implementations. Table 15-1 provides an example of the performance and area benefits of this approach.

Table 15-1. sysDSP Block vs. LUT-based Multipliers

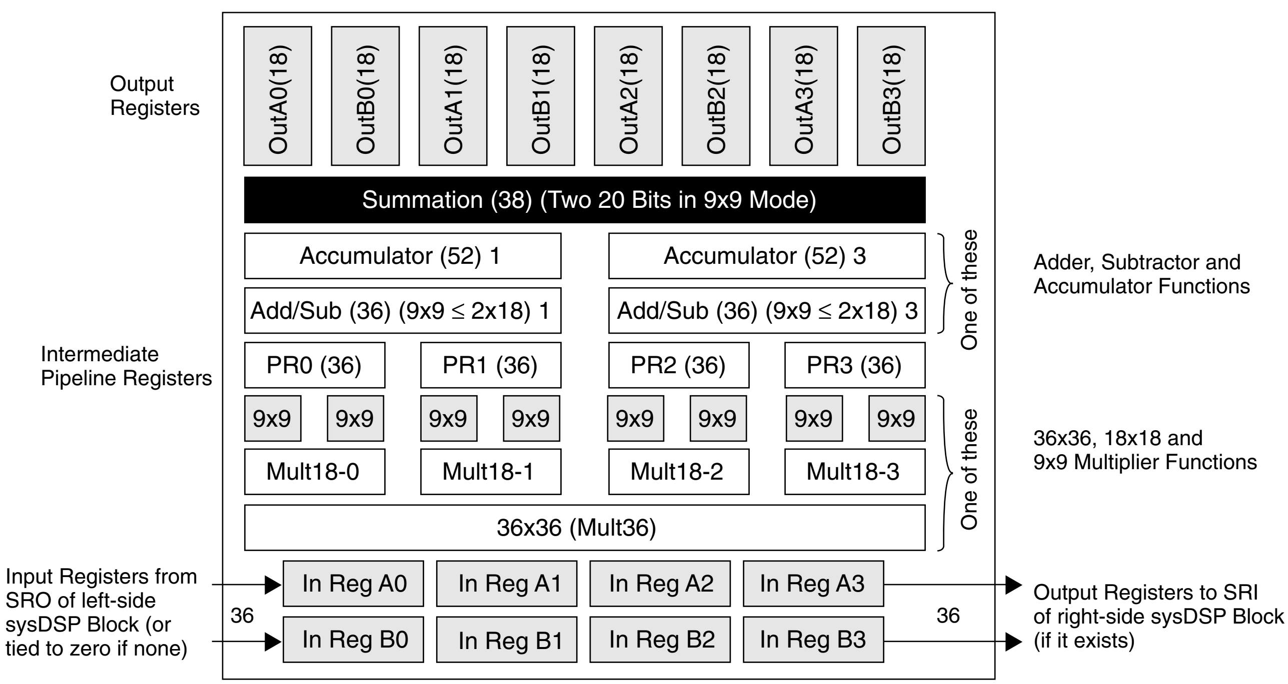
Multiplier Width	Register Pipelining	LatticeECP LFECP20E-5 Uses One DSP Block		LatticeEC LFEC20E-5 Uses LUTs	
		f_{MAX}	LUTs	f_{MAX}	LUTs
9x9	Input, Multiplier, Output	235	0	76	174
18x18	Input, Multiplier, Output	211	0	50	608
36x36	Input, Multiplier, Output	177	0	35	2225

sysDSP Block Hardware

Overview

The sysDSP Blocks are located in a row at the center of the LatticeECP-DSP device. A sysDSP Block block diagram is shown in Figure 15-1.

Figure 15-1. LatticeECP sysDSP Block Diagram



Note: Each sysDSP Block spans eight columns of PFUs.

sysDSP Blocks have three operating modes:

36x36 Mode
- One 36x36 multiplier

18x18 Mode
- Four Multipliers
- Two 52-bit MACs
- Two sums of two 18x18 multipliers each
- One sum of four 18x18 multipliers

9x9 Mode
- Eight Multipliers
- Two 34-bit MACs
- Four sums of two 9x9 multipliers each
- Two sums of four 9x9 multipliers each

sysDSP Block Software

Overview

The sysDSP Block of the LatticeECP-DSP device can be targeted in a number of ways.

- IPexpress™ in the Lattice ispLEVER® design tools allows the rapid creation of modules implementing sysDSP elements. These modules can then be used in HDL designs as appropriate.
- The coding of certain functions into a design's HDL allows the synthesis tools to inference the use of a sysDSP Block.
- The implementation of designs in Mathwork's Simulink tool using a Lattice Block set. The ispLEVER sysDSP portion of the ispLEVER tools then converts these blocks into HDL as appropriate.
- Instantiation of sysDSP primitives directly in the source code.

Targeting the sysDSP Block Using IPexpress

IPexpress allows you to graphically specify sysDSP elements. Once the element is specified, an HDL file is generated which can be instantiated in a design. IPexpress allows users to configure all ports and set all available parameters. The following show modules which target the sysDSP Block. For design examples using IPexpress, refer to EXAMPLES in your ispLEVER software. From the Project Navigator pull-down menu, go to File -> Open Example). The following four element types can be specified via IPexpress:

- MULT (Multiplier)
- MAC (Multiplier Accumulate)
- MULTADDSUB (Multiplier Add/Subtract)
- MULTADDSUBSUM (Multiply Add/Subtract and SUM)

MULT Module

The MULT Module configures elements to be packed into the sysDSP primitives. The Basic mode screen illustrated in Figure 15-2 consists of one clock (optional), one clock enable and one reset tied to all registers. Using the multiplier you can span multiple sysDSP Blocks. The SRI and SRO ports can only be enabled if inputs are less than 18 bits. The Advanced mode screen, illustrated in Figure 15-3, allows finer control over the register. In the Advanced mode you can control each register with independent clocks, clock enables and resets.

Figure 15-2. MULT Mode Basic Set-up

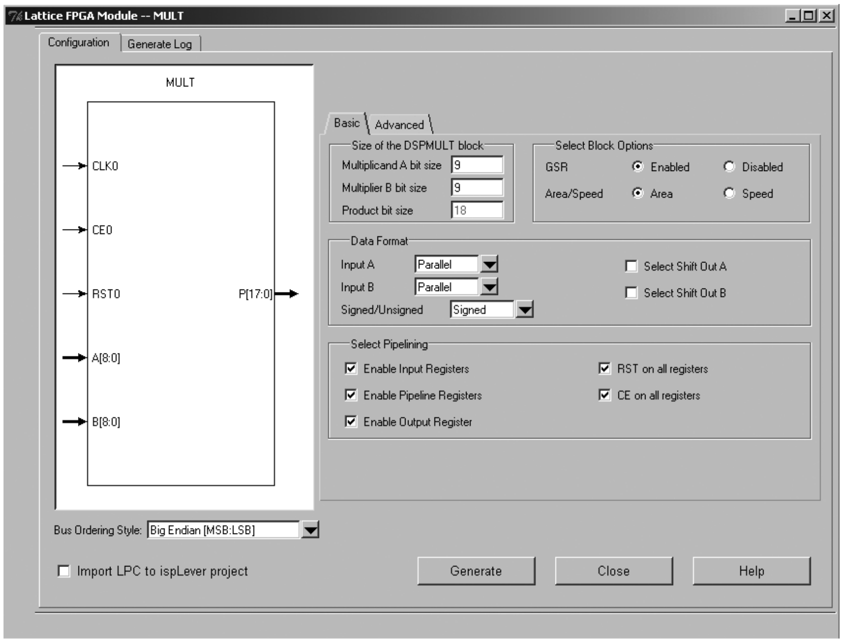
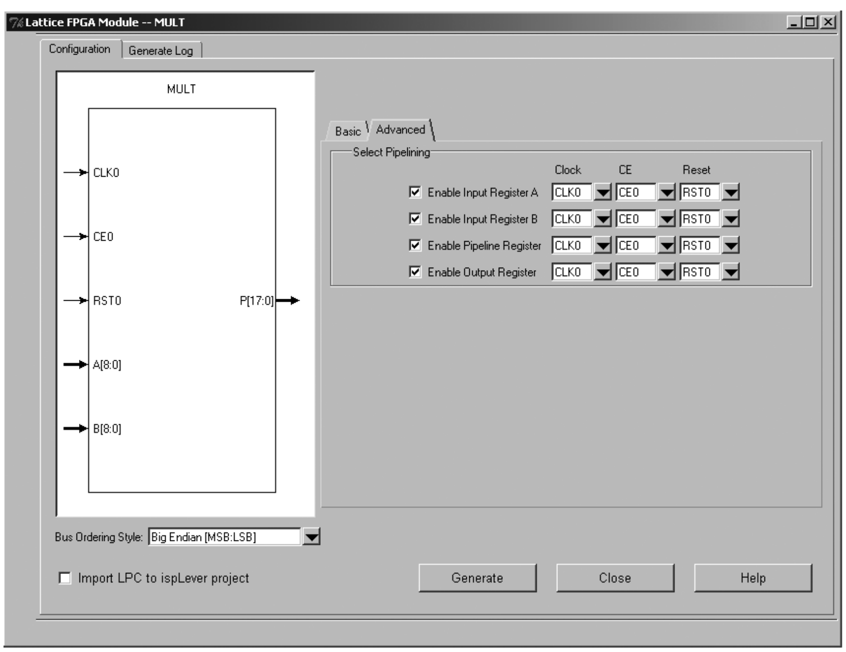


Figure 15-3. MULT Mode Advanced Set-up



MAC Module

The MAC Module configures multiply accumulate elements to be packed into the primitives. The Basic mode, shown in Figure 15-4, consists of one clock (optional), one clock enable and one reset tied to all registers. Because of the Accumulator, the output register is automatically enabled. The SRI and SRO ports can only be enabled if inputs are less than 18 bits. The Accumsload loads the accumulator with the value from the multiplier. This is required to initialize and load the first value of the accumulation. The Advanced mode, shown in Figure 15-5, allows finer control over the registers. In the Advanced mode you can control each register with independent clocks, clock enables and resets.

Figure 15-4. MAC Mode Basic Set-up

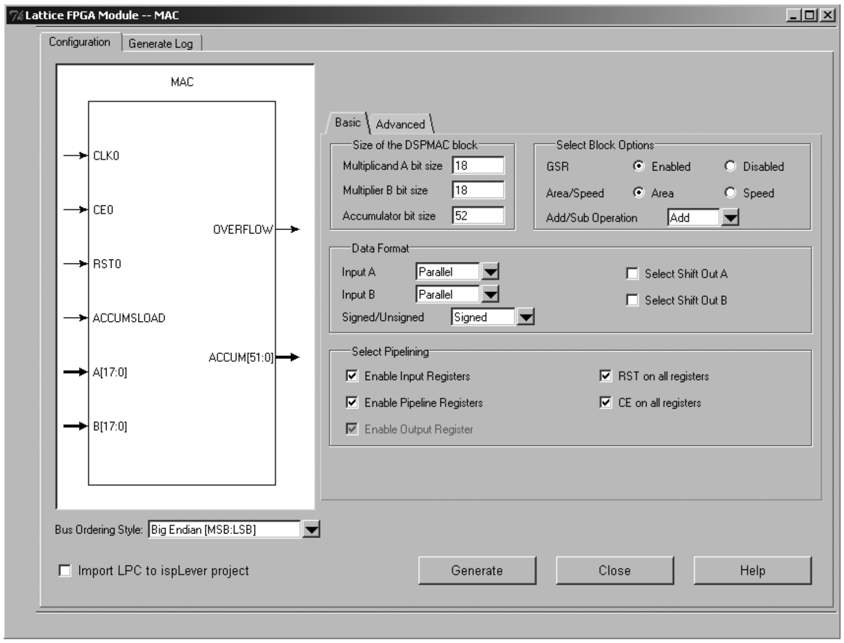
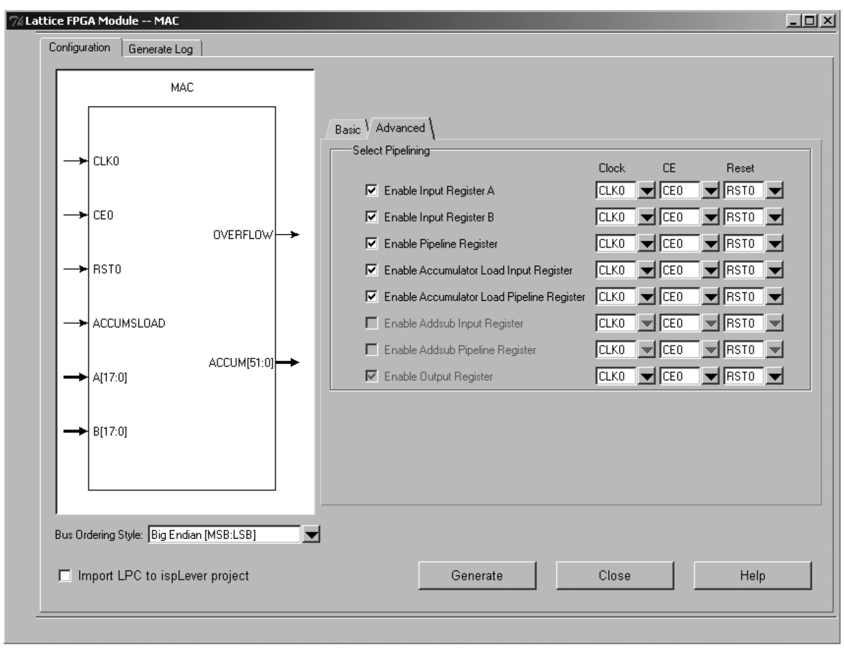


Figure 15-5. MAC Mode Advanced Set-up



MULTADDSUB Module

The MULTADDSUB GUI configures Multiplier Addition/Subtraction elements to be packed into the primitives MULT18X18ADDSUB or MULT9X9ADDSUB. The Basic mode, shown in Figure 15-6, consists of one clock (optional), one clock enable and one reset tied to all registers. Using the MULTADDSUB you can span multiple sysDSP Blocks. The SRI and SRO ports can only be enabled if inputs are less than 18 bits. The Advanced mode, shown in Figure 15-7, provides finer control over the registers. In the Advanced mode you can control each register with independent clocks, clock enables and resets.

Figure 15-6. MULTADDSUB Mode Basic Set-up

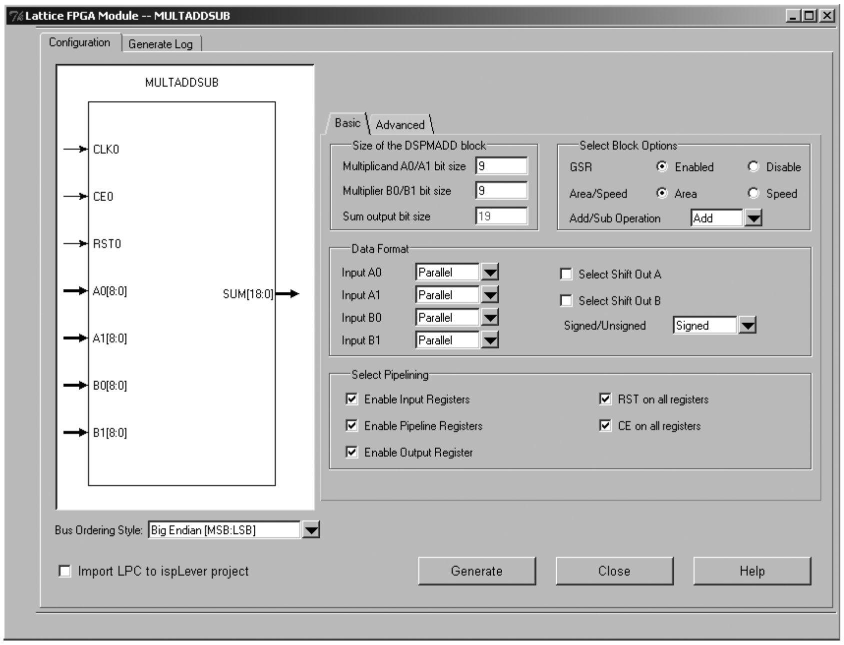
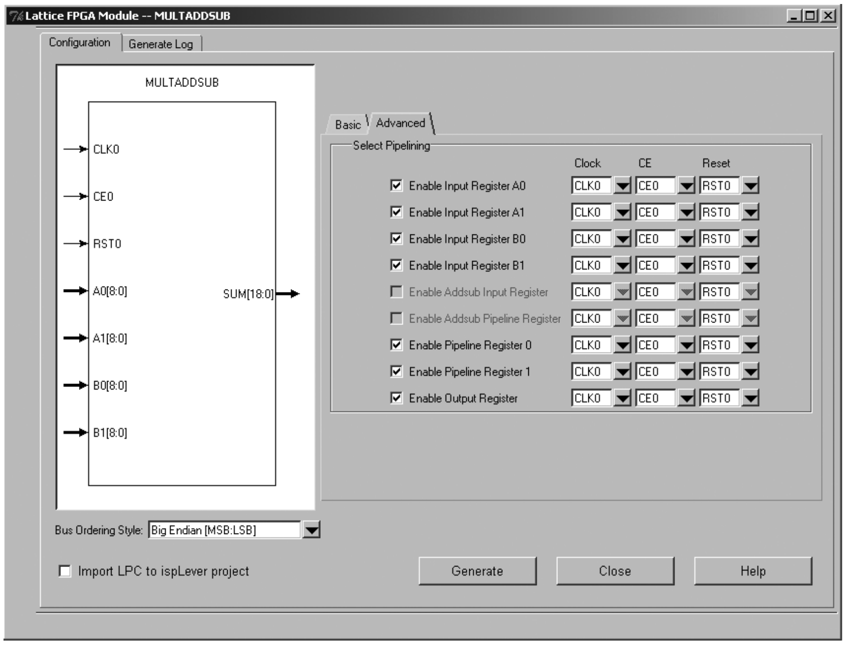


Figure 15-7. MULTADDSUB Mode Advanced Set-up



MULTADDSUBSUM Module

The MULTADDSUBSUM GUI configures Multiplier Addition/Subtraction Addition elements to be packed into the primitives MULT18X18ADDSUBSUM or MULT9X9ADDSUBSUM. The Basic mode, shown in Figure 15-8, consists of one clock (optional), one clock enable and one reset tied to all registers. Using the MULTADDSUBSUM you can span multiple sysDSP Blocks. The SRI and SRO ports can only be enabled if inputs are less than 18 bits. The Advanced mode, shown in Figure 15-9, provides finer control over the registers. In the Advanced mode you can control each register with independent clocks, clock enables and resets.

Figure 15-8. MULTADDSUBSUM Mode Basic Set-up

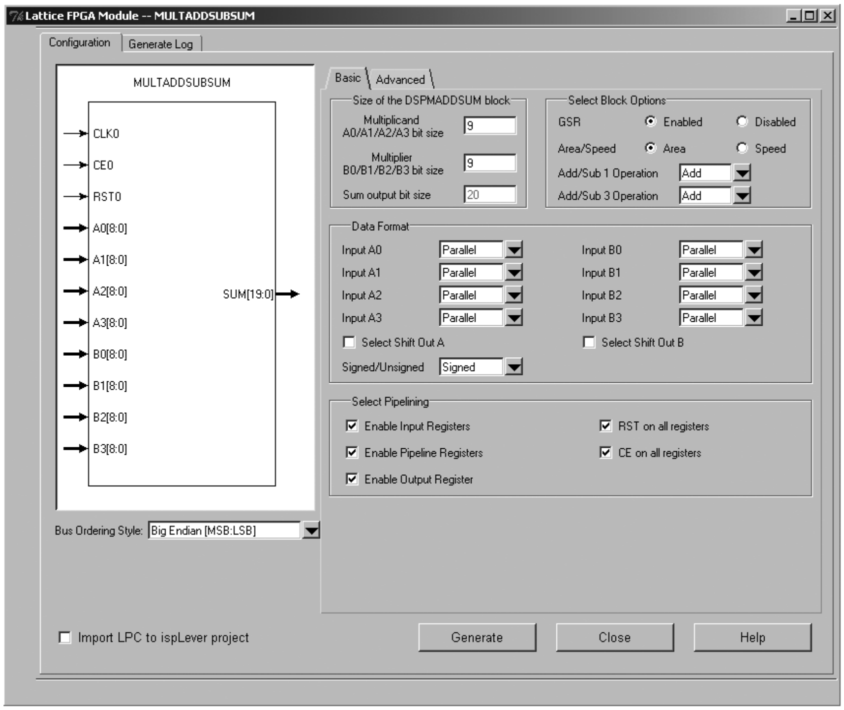
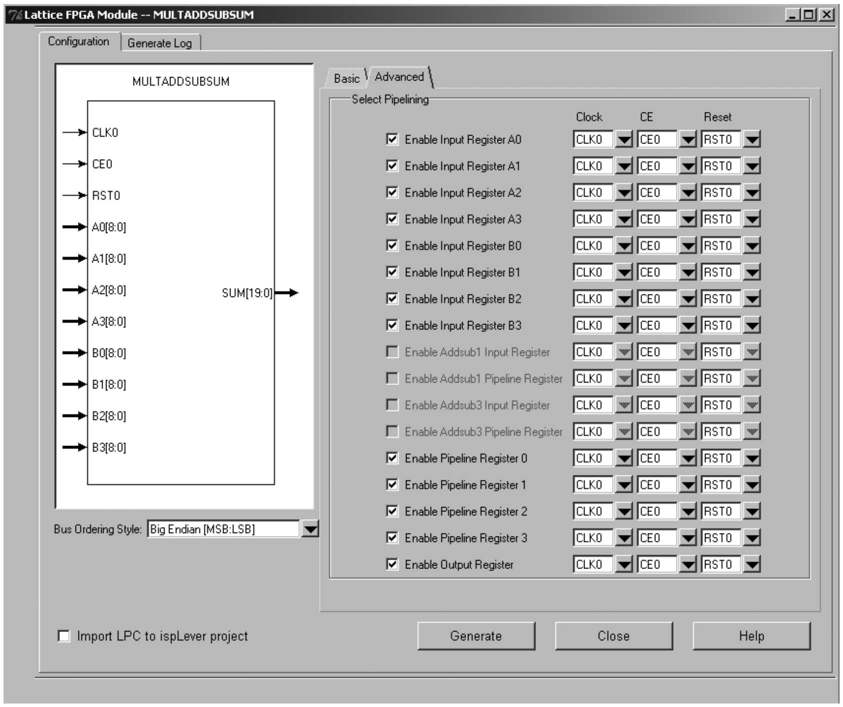


Figure 15-9. MULTADDSUBSUM Mode Advance 1 Set-up



Targeting the sysDSP Block by Inference

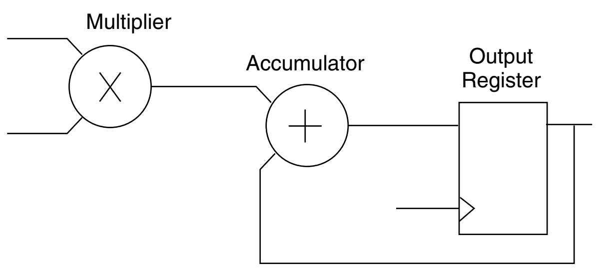
The Inferencing flow enables the design tools to infer sysDSP Blocks from a HDL design. It is important to note that when using the Inferencing flow; unless the code style matches the sysDSP Block results will not be optimal results. Consider the following Veriflog and VHDL examples:

```
// This Verilog example will be mapped into single MULT9X9MAC with the output register enabled
// This will be mapped into single MULT9X9MAC with the output register enabled
module mult_acc (dataout, dataax, dataay, clk);
   output [16:0] dataout;
   input  [7:0] dataax, dataay;
   input clk;
   reg   [16:0] dataout;

   wire [15:0] multa = dataax * dataay; // 9x9 Multiplier
   wire [16:0] adder_out;
   assign adder_out = multa + dataout; // Accumulator
   always @(posedge clk)
   begin
     dataout <= adder_out; // Output Register of the Accumulator
   end
endmodule
```

```
-- This VHDL example will be mapped into single MULT9X9MAC with the output register enabled
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_signed.all;

entity mult_acc is
port (
  clk   : in std_logic;
  dataax : in std_logic_vector(7 downto 0);
  dataay : in std_logic_vector(7 downto 0);
  dataout : out std_logic_vector(16 downto 0));
end;

architecture arch of mult_acc is
  signal multout : std_logic_vector(15 downto 0);
  signal adder_out : std_logic_vector(16 downto 0);
  signal dataout_reg : std_logic_vector(16 downto 0);
begin
  multout <= dataax * dataay; -- 9x9 Multiplier

  adder_out <= multout + dataout_reg;
  process (clk)
  begin
    if (clk'event and clk='1') then
      dataout_reg <= adder_out; -- Output Register of the Accumulator
    end if;
end process;
    dataout <= dataout_reg;
end arch;
```

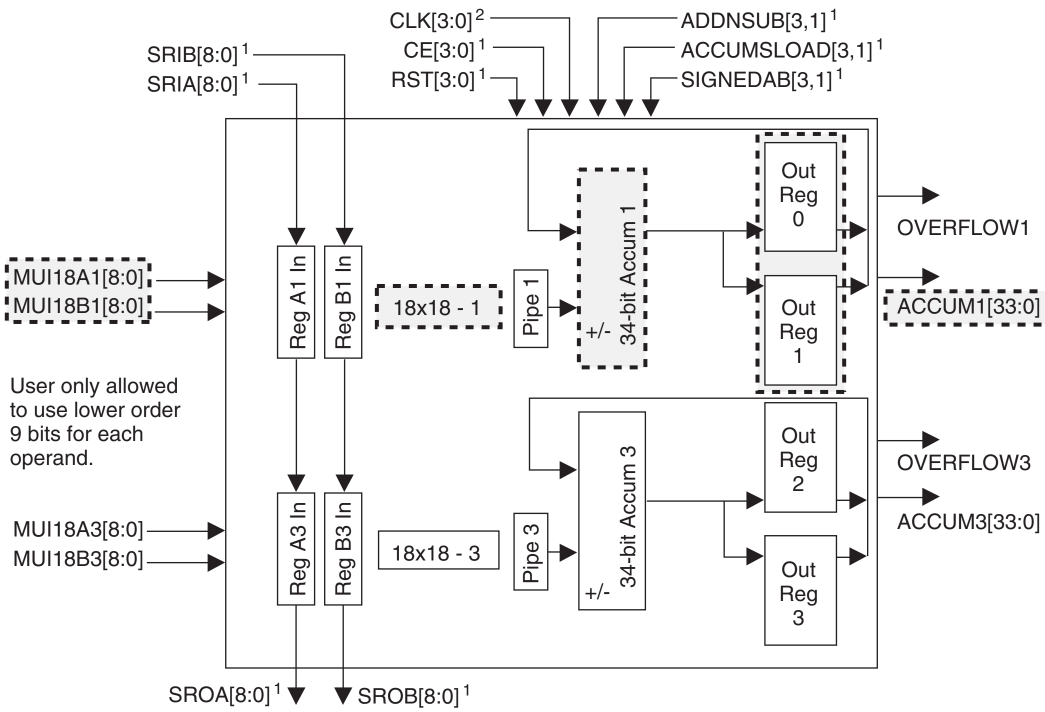
The above RTL will infer the following block diagram.

Figure 15-10. MULT9X9MAC Block Diagram



As you can see, this block diagram can be mapped directly into the sysDSP primitives If we were to add a test point between the multiplier and the accumulator, or two output registers, the code could not be mapped into a MULT9X9MAC of a sysDSP Block. So options you could include in the design would be input registers, pipeline registers, etc. For more inferring design examples refer to EXAMPLES in your ispLEVER software.

Figure 15-11. MAC9X9MAC Packed into a sysDSP Block



Notes:
1. These signals are optional.
2. At least one clock is required.

Targeting the sysDSP Block using Simulink

Simulink Overview

Simulink is a graphical add-on (similar to schematic entry) for Matlab, which is produced by Mathworks. For more information refer to the Simulink web site at www.mathworks.com/products/simulink/.

Why is Simulink Used?

- Simulink allows users to create algorithms using floating point numbers
- It helps users convert floating point algorithms into fixed point algorithms

How Does Simulink Fit into the Normal ispLEVER Design Flow?

Once you have converted have your algorithm working in fixed point you can use the Lattice ispDSP Block to create HDL files, which can be instantiated in your HDL design. Currently there is only support for VHDL.

What Does Lattice Provide?

Lattice provides a library of blocks for the Simulink tool, which include multipliers, adders, registers and other standard building blocks. Besides the basic building blocks there are a couple of unique Lattice blocks.

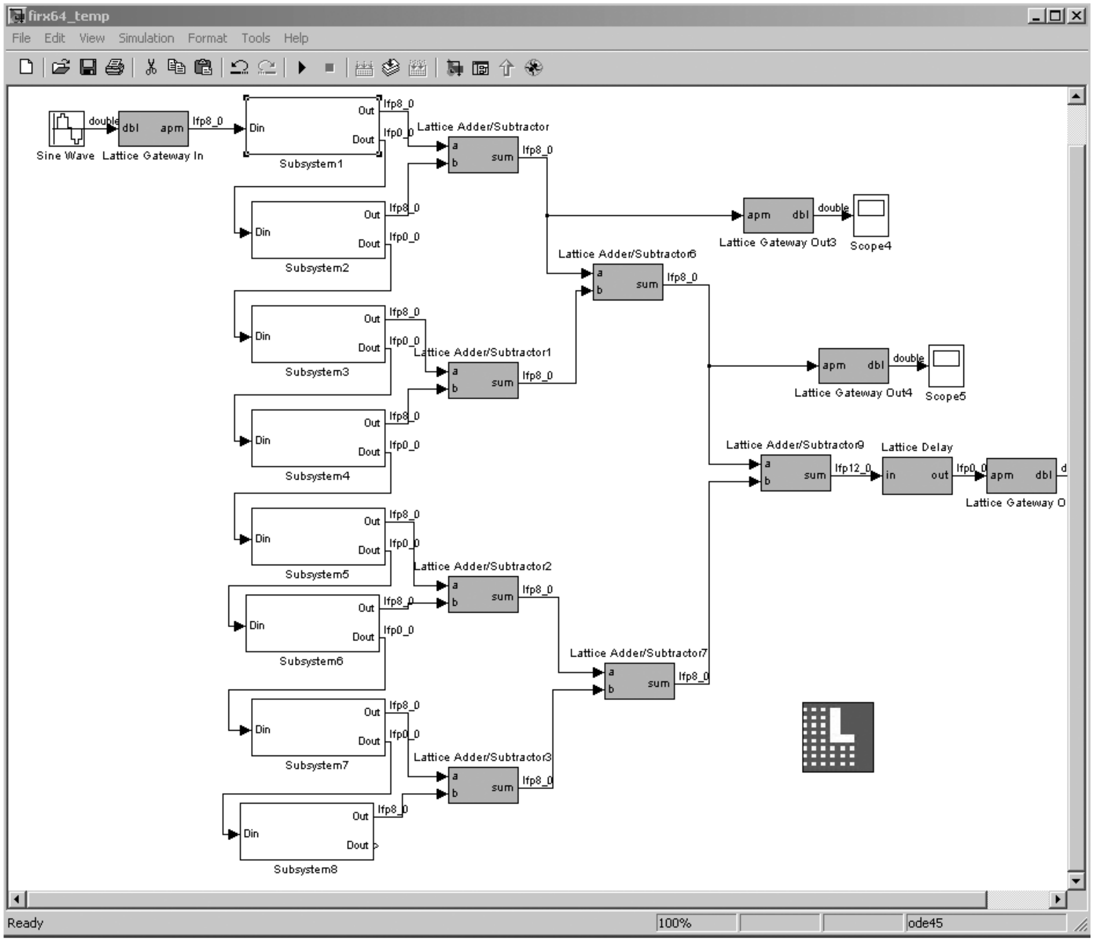
Gateways In and Out

Everything between Gateways In and Out represents HDL code. Everything before a Gateway In is the stimulus your test bench. Everything after a Gateway Out are the signals you will be monitoring in your test bench. Figure 15-12 is an example.

ispDSP

The ispDSP block is used to convert fixed point Simulink design into HDL files which can be instantiated in your HDL design. The ispDSP Block is identified by the Lattice logo and can be seen in Figure 21.

Figure 15-12. Simulink Design



Targeting the sysDSP Block by Instantiating Primitives

You can target the sysDSP Block by instantiating the sysDSP Block primitives into your design. The advantage of instantiating primitives is that it gives you access to all ports and set all available parameters. The disadvantage of this flow is that all this customization comes at a cost of extra coding for the user. Appendix A details the syntax for the sysDSP Block primitives.

sysDSP Blocks in the Report File

To check configuration of the sysDSP Blocks in your design, look at the MAP and Post PAR report files. The MAP Report File shows the mapped sysDSP components/primitives in your design. The POST PAR Report File shows the number of components in each sysDSP Block. The following gives an example of the sysDSP section from each of the Report Files.

MAP Report File

```
Number Of Mapped DSP Components:
--------------------------------
MULT36X36                  0
MULT18X18                  1
MULT18X18MAC               0
MULT18X18ADDSUB            0
MULT18X18ADDSUBSUM         0
MULT9X9                    0
MULT9X9MAC                 0
MULT9X9ADDSUB              0
MULT9X9ADDSUBSUM           0
```

Post PAR Report File

```
DSP Utilization Summary:
-------------------------------------
DSP Block #:               1      2      3      4      5      6      7
# of MULT36X36
# of MULT18X18             1
# of MULT18X18MAC
# of MULT18X18ADDSUB
# of MULT18X18ADDSUBSUM
# of MULT9X9
# of MULT9X9MAC
# of MULT9X9ADDSUB
# of MULT9X9ADDSUBSUM
```

Technical Support Assistance

Hotline: 1-800-LATTICE (North America)
+1-503-268-8001 (Outside North America)

e-mail: techsupport@latticesemi.com

Internet: www.latticesemi.com

Revision History

Date	Version	Change Summary
—	—	Previous Lattice releases.
September 2012	01.3	Updated document with new corporate logo.

Appendix A. DSP Block Primitives

MULT36X36 Primitive

```
MULT36X36(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,
A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A16,A17,
A18,A19,A20,A21,A22,A23,A24,A25,A26,A27,A28,A29,A30,A31,A32,A33,A34,A35,
B0,B1,B2,B3,B4,B5,B6,B7,B8,B9,B10,B11,B12,B13,B14,B15,B16,B17,
B18,B19,B20,B21,B22,B23,B24,B25,B26,B27,B28,B29,B30,B31,B32,B33,B34,B35,
P0,P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11,P12,P13,P14,P15,P16,P17,
P18,P19,P20,P21,P22,P23,P24,P25,P26,P27,P28,P29,P30,P31,P32,P33,P34,P35,
P36,P37,P38,P39,P40,P41,P42,P43,P44,P45,P46,P47,P48,P49,P50,P51,P52,P53,
P54,P55,P56,P57,P58,P59,P60,P61,P62,P63,P64,P65,P66,P67,P68,P69,P70,P71) is black-box
{
property REG_INPUTA_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_INPUTA_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_INPUTA_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property REG_INPUTB_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_INPUTB_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_INPUTB_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property REG_PIPELINE_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_PIPELINE_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_PIPELINE_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property REG_OUTPUT_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_OUTPUT_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_OUTPUT_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property REG_SIGNEDAB_0_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_SIGNEDAB_0_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_SIGNEDAB_0_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property REG_SIGNEDAB_1_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_SIGNEDAB_1_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_SIGNEDAB_1_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property GSR {“ENABLED”,”DISABLED”};
```

MULT18X18 Primitive

```
MULT18X18(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,
A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A16,A17,
B0,B1,B2,B3,B4,B5,B6,B7,B8,B9,B10,B11,B12,B13,B14,B15,B16,B17,
SRIA0,SRIA1,SRIA2,SRIA3,SRIA4,SRIA5,SRIA6,SRIA7,SRIA8,SRIA9,SRIA10,SRIA11,SRIA12,SRIA13,SRIA1
4,SRIA15,SRIA16,SRIA17,
SRIB0,SRIB1,SRIB2,SRIB3,SRIB4,SRIB5,SRIB6,SRIB7,SRIB8,SRIB9,SRIB10,SRIB11,SRIB12,SRIB13,SRIB1
4,SRIB15,SRIB16,SRIB17,
SROA0,SROA1,SROA2,SROA3,SROA4,SROA5,SROA6,SROA7,SROA8,SROA9,SROA10,SROA11,SROA12,SROA13,SROA1
4,SROA15,SROA16,SROA17,
SROB0,SROB1,SROB2,SROB3,SROB4,SROB5,SROB6,SROB7,SROB8,SROB9,SROB10,SROB11,SROB12,SROB13,SROB1
4,SROB15,SROB16,SROB17,
P0,P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11,P12,P13,P14,P15,P16,P17,
P18,P19,P20,P21,P22,P23,P24,P25,P26,P27,P28,P29,P30,P31,P32,P33,P34,P35) is black-box
{
property REG_INPUTA_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_INPUTA_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_INPUTA_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property REG_INPUTB_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_INPUTB_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_INPUTB_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property REG_PIPELINE_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
property REG_PIPELINE_CE {“CE0”,”CE1”,”CE2”,”CE3”};
property REG_PIPELINE_RST {“RST0”,”RST1”,”RST2”,”RST3”};
property REG_OUTPUT_CLK {“NONE”,”CLK0”,”CLK1”,”CLK2”,”CLK3”};
```

```
property REG_OUTPUT_CE {"CE0","CE1","CE2","CE3"};
property REG_OUTPUT_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_1_RST {"RST0","RST1","RST2","RST3"};
property SHIFT_IN_A {"FALSE", "TRUE"};
property SHIFT_IN_B {"FALSE", "TRUE"};
property GSR {"ENABLED","DISABLED"};
```

MULT18X18MAC Primitive

```
MULT18X18MAC(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,ADDNSUB,ACCUMS-
LOAD,
A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A16,A17,
B0,B1,B2,B3,B4,B5,B6,B7,B8,B9,B10,B11,B12,B13,B14,B15,B16,B17,
SRIA0,SRIA1,SRIA2,SRIA3,SRIA4,SRIA5,SRIA6,SRIA7,SRIA8,
SRIA9,SRIA10,SRIA11,SRIA12,SRIA13,SRIA14,SRIA15,SRIA16,SRIA17,
SRIB0,SRIB1,SRIB2,SRIB3,SRIB4,SRIB5,SRIB6,SRIB7,SRIB8,
SRIB9,SRIB10,SRIB11,SRIB12,SRIB13,SRIB14,SRIB15,SRIB16,SRIB17,
SROA0,SROA1,SROA2,SROA3,SROA4,SROA5,SROA6,SROA7,SROA8,
SROA9,SROA10,SROA11,SROA12,SROA13,SROA14,SROA15,SROA16,SROA17,
SROB0,SROB1,SROB2,SROB3,SROB4,SROB5,SROB6,SROB7,SROB8,
SROB9,SROB10,SROB11,SROB12,SROB13,SROB14,SROB15,SROB16,SROB17,
ACCUM0,ACCUM1,ACCUM2,ACCUM3,ACCUM4,ACCUM5,ACCUM6,ACCUM7,ACCUM8,
ACCUM9,ACCUM10,ACCUM11,ACCUM12,ACCUM13,ACCUM14,ACCUM15,ACCUM16,ACCUM17,
ACCUM18,ACCUM19,ACCUM20,ACCUM21,ACCUM22,ACCUM23,ACCUM24,ACCUM25,ACCUM26,
ACCUM27,ACCUM28,ACCUM29,ACCUM30,ACCUM31,ACCUM32,ACCUM33,ACCUM34,ACCUM35,
ACCUM36,ACCUM37,ACCUM38,ACCUM39,ACCUM40,ACCUM41,ACCUM42,ACCUM43,ACCUM44,
ACCUM45,ACCUM46,ACCUM47,ACCUM48,ACCUM49,ACCUM50,ACCUM51,OVERFLOW) is black-box
{
property REG_INPUTA_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE_RST {"RST0","RST1","RST2","RST3"};
property REG_OUTPUT_CLK {"CLK0","CLK1","CLK2","CLK3"};
property REG_OUTPUT_CE {"CE0","CE1","CE2","CE3"};
property REG_OUTPUT_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ACCUMSLOAD_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ACCUMSLOAD_0_CE {"CE0","CE1","CE2","CE3"};
```

```
property REG_ACCUMSLOAD_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ACCUMSLOAD_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ACCUMSLOAD_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ACCUMSLOAD_1_RST {"RST0","RST1","RST2","RST3"};
property SHIFT_IN_A {"FALSE", "TRUE"};
property SHIFT_IN_B {"FALSE", "TRUE"};
property GSR {"ENABLED","DISABLED"};
```

MULT18X18ADDSUB Primitive

```
MULT18X18ADDSUB(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,ADDNSUB,
A00,A01,A02,A03,A04,A05,A06,A07,A08,A09,A010,A011,A012,A013,A014,A015,A016,A017,
A10,A11,A12,A13,A14,A15,A16,A17,A18,A19,A110,A111,A112,A113,A114,A115,A116,A117,
B00,B01,B02,B03,B04,B05,B06,B07,B08,B09,B010,B011,B012,B013,B014,B015,B016,B017,
B10,B11,B12,B13,B14,B15,B16,B17,B18,B19,B110,B111,B112,B113,B114,B115,B116,B117,
SRIA0,SRIA1,SRIA2,SRIA3,SRIA4,SRIA5,SRIA6,SRIA7,SRIA8,
SRIA9,SRIA10,SRIA11,SRIA12,SRIA13,SRIA14,SRIA15,SRIA16,SRIA17,
SRIB0,SRIB1,SRIB2,SRIB3,SRIB4,SRIB5,SRIB6,SRIB7,SRIB8,
SRIB9,SRIB10,SRIB11,SRIB12,SRIB13,SRIB14,SRIB15,SRIB16,SRIB17,
SROA0,SROA1,SROA2,SROA3,SROA4,SROA5,SROA6,SROA7,SROA8,
SROA9,SROA10,SROA11,SROA12,SROA13,SROA14,SROA15,SROA16,SROA17,
SROB0,SROB1,SROB2,SROB3,SROB4,SROB5,SROB6,SROB7,SROB8,
SROB9,SROB10,SROB11,SROB12,SROB13,SROB14,SROB15,SROB16,SROB17,
SUM0,SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,SUM7,SUM8,
SUM9,SUM10,SUM11,SUM12,SUM13,SUM14,SUM15,SUM16,SUM17,
SUM18,SUM19,SUM20,SUM21,SUM22,SUM23,SUM24,SUM25,SUM26,
SUM27,SUM28,SUM29,SUM30,SUM31,SUM32,SUM33,SUM34,SUM35,
SUM36) is black-box
{
property REG_INPUTA0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA0_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA0_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTA1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA1_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA1_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB0_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB0_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB1_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB1_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE0_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE0_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE1_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE1_RST {"RST0","RST1","RST2","RST3"};
property REG_OUTPUT_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_OUTPUT_CE {"CE0","CE1","CE2","CE3"};
property REG_OUTPUT_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB_0_RST {"RST0","RST1","RST2","RST3"};
```

```
property REG_ADDNSUB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB_1_RST {"RST0","RST1","RST2","RST3"};
property SHIFT_IN_A0 {"FALSE", "TRUE"};
property SHIFT_IN_B0 {"FALSE", "TRUE"};
property SHIFT_IN_A1 {"FALSE", "TRUE"};
property SHIFT_IN_B1 {"FALSE", "TRUE"};
property GSR {"ENABLED","DISABLED"};
```

MULT18X18ADDSUBSUM Primitive

```
MULT18X18ADDSUBSUM(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,ADDNSUB1,
ADDNSUB3,
A00,A01,A02,A03,A04,A05,A06,A07,A08,A09,A010,A011,A012,A013,A014,A015,A016,A017,
A10,A11,A12,A13,A14,A15,A16,A17,A18,A19,A110,A111,A112,A113,A114,A115,A116,A117,
A20,A21,A22,A23,A24,A25,A26,A27,A28,A29,A210,A211,A212,A213,A214,A215,A216,A217,
A30,A31,A32,A33,A34,A35,A36,A37,A38,A39,A310,A311,A312,A313,A314,A315,A316,A317,
B00,B01,B02,B03,B04,B05,B06,B07,B08,B09,B010,B011,B012,B013,B014,B015,B016,B017,
B10,B11,B12,B13,B14,B15,B16,B17,B18,B19,B110,B111,B112,B113,B114,B115,B116,B117,
B20,B21,B22,B23,B24,B25,B26,B27,B28,B29,B210,B211,B212,B213,B214,B215,B216,B217,
B30,B31,B32,B33,B34,B35,B36,B37,B38,B39,B310,B311,B312,B313,B314,B315,B316,B317,
SRIA0,SRIA1,SRIA2,SRIA3,SRIA4,SRIA5,SRIA6,SRIA7,SRIA8,
SRIA9,SRIA10,SRIA11,SRIA12,SRIA13,SRIA14,SRIA15,SRIA16,SRIA17,
SRIB0,SRIB1,SRIB2,SRIB3,SRIB4,SRIB5,SRIB6,SRIB7,SRIB8,
SRIB9,SRIB10,SRIB11,SRIB12,SRIB13,SRIB14,SRIB15,SRIB16,SRIB17,
SROA0,SROA1,SROA2,SROA3,SROA4,SROA5,SROA6,SROA7,SROA8,
SROA9,SROA10,SROA11,SROA12,SROA13,SROA14,SROA15,SROA16,SROA17,
SROB0,SROB1,SROB2,SROB3,SROB4,SROB5,SROB6,SROB7,SROB8,
SROB9,SROB10,SROB11,SROB12,SROB13,SROB14,SROB15,SROB16,SROB17,
SUM0,SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,SUM7,SUM8,
SUM9,SUM10,SUM11,SUM12,SUM13,SUM14,SUM15,SUM16,SUM17,
SUM18,SUM19,SUM20,SUM21,SUM22,SUM23,SUM24,SUM25,SUM26,
SUM27,SUM28,SUM29,SUM30,SUM31,SUM32,SUM33,SUM34,SUM35,
SUM36,SUM37) is black-box
{
property REG_INPUTA0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA0_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA0_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTA1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA1_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA1_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTA2_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA2_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA2_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTA3_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA3_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA3_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB0_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB0_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB1_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB1_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB2_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB2_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB2_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB3_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB3_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB3_RST {"RST0","RST1","RST2","RST3"};
```

```
property REG_PIPELINE0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE0_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE0_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE1_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE1_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE2_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE2_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE2_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE3_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE3_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE3_RST {"RST0","RST1","RST2","RST3"};
property REG_OUTPUT_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_OUTPUT_CE {"CE0","CE1","CE2","CE3"};
property REG_OUTPUT_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB1_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB1_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB1_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB1_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB1_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB1_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB3_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB3_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB3_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB3_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB3_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB3_1_RST {"RST0","RST1","RST2","RST3"};
property SHIFT_IN_A0 {"FALSE", "TRUE"};
property SHIFT_IN_B0 {"FALSE", "TRUE"};
property SHIFT_IN_A1 {"FALSE", "TRUE"};
property SHIFT_IN_B1 {"FALSE", "TRUE"};
property SHIFT_IN_A2 {"FALSE", "TRUE"};
property SHIFT_IN_B2 {"FALSE", "TRUE"};
property SHIFT_IN_A3 {"FALSE", "TRUE"};
property SHIFT_IN_B3 {"FALSE", "TRUE"};
property GSR {"ENABLED","DISABLED"};
```

MULT9X9 Primitive

```
MULT9X9(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,
A0,A1,A2,A3,A4,A5,A6,A7,A8,B0,B1,B2,B3,B4,B5,B6,B7,B8,
SRIA0,SRIA1,SRIA2,SRIA3,SRIA4,SRIA5,SRIA6,SRIA7,SRIA8,
SRIB0,SRIB1,SRIB2,SRIB3,SRIB4,SRIB5,SRIB6,SRIB7,SRIB8,
SROA0,SROA1,SROA2,SROA3,SROA4,SROA5,SROA6,SROA7,SROA8,
SROB0,SROB1,SROB2,SROB3,SROB4,SROB5,SROB6,SROB7,SROB8,
P0,P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11,P12,P13,P14,P15,P16,P17) is black-box
{
property REG_INPUTA_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB_RST {"RST0","RST1","RST2","RST3"};
```

```
property REG_PIPELINE_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE_RST {"RST0","RST1","RST2","RST3"};
property REG_OUTPUT_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_OUTPUT_CE {"CE0","CE1","CE2","CE3"};
property REG_OUTPUT_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_1_RST {"RST0","RST1","RST2","RST3"};
property SHIFT_IN_A {"FALSE", "TRUE"};
property SHIFT_IN_B {"FALSE", "TRUE"};
property GSR {"ENABLED","DISABLED"};
```

MULT9X9MAC Primitive

```
MULT9X9MAC(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,ADDNSUB,ACCUMS-
LOAD,
A0,A1,A2,A3,A4,A5,A6,A7,A8,
B0,B1,B2,B3,B4,B5,B6,B7,B8,
SRIA0,SRIA1,SRIA2,SRIA3,SRIA4,SRIA5,SRIA6,SRIA7,SRIA8,
SRIB0,SRIB1,SRIB2,SRIB3,SRIB4,SRIB5,SRIB6,SRIB7,SRIB8,
SROA0,SROA1,SROA2,SROA3,SROA4,SROA5,SROA6,SROA7,SROA8,
SROB0,SROB1,SROB2,SROB3,SROB4,SROB5,SROB6,SROB7,SROB8,
ACCUM0,ACCUM1,ACCUM2,ACCUM3,ACCUM4,ACCUM5,ACCUM6,ACCUM7,ACCUM8,
ACCUM9,ACCUM10,ACCUM11,ACCUM12,ACCUM13,ACCUM14,ACCUM15,ACCUM16,ACCUM17,
ACCUM18,ACCUM19,ACCUM20,ACCUM21,ACCUM22,ACCUM23,ACCUM24,ACCUM25,ACCUM26,
ACCUM27,ACCUM28,ACCUM29,ACCUM30,ACCUM31,ACCUM32,ACCUM33,OVERFLOW) is black-box
{
property REG_INPUTA_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE_RST {"RST0","RST1","RST2","RST3"};
property REG_OUTPUT_CLK {"CLK0","CLK1","CLK2","CLK3"};
property REG_OUTPUT_CE {"CE0","CE1","CE2","CE3"};
property REG_OUTPUT_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ACCUMSLOAD_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ACCUMSLOAD_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ACCUMSLOAD_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ACCUMSLOAD_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
```

```
property REG_ACCUMSLOAD_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ACCUMSLOAD_1_RST {"RST0","RST1","RST2","RST3"};
property SHIFT_IN_A {"FALSE", "TRUE"};
property SHIFT_IN_B {"FALSE", "TRUE"};
property GSR {"ENABLED","DISABLED"};
```

MULT9X9ADDSUB Primitive

```
MULT9X9ADDSUB(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,ADDNSUB,
A00,A01,A02,A03,A04,A05,A06,A07,A08,
A10,A11,A12,A13,A14,A15,A16,A17,A18,
B00,B01,B02,B03,B04,B05,B06,B07,B08,
B10,B11,B12,B13,B14,B15,B16,B17,B18,
SRIA0,SRIA1,SRIA2,SRIA3,SRIA4,SRIA5,SRIA6,SRIA7,SRIA8,
SRIB0,SRIB1,SRIB2,SRIB3,SRIB4,SRIB5,SRIB6,SRIB7,SRIB8,
SROA0,SROA1,SROA2,SROA3,SROA4,SROA5,SROA6,SROA7,SROA8,
SROB0,SROB1,SROB2,SROB3,SROB4,SROB5,SROB6,SROB7,SROB8,
SUM0,SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,SUM7,SUM8,
SUM9,SUM10,SUM11,SUM12,SUM13,SUM14,SUM15,SUM16,SUM17,SUM18) is black-box
{
property REG_INPUTA0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA0_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA0_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTA1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA1_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA1_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB0_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB0_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB1_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB1_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE0_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE0_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE1_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE1_RST {"RST0","RST1","RST2","RST3"};
property REG_OUTPUT_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_OUTPUT_CE {"CE0","CE1","CE2","CE3"};
property REG_OUTPUT_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB_1_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_1_RST {"RST0","RST1","RST2","RST3"};
property SHIFT_IN_A0 {"FALSE", "TRUE"};
property SHIFT_IN_B0 {"FALSE", "TRUE"};
property SHIFT_IN_A1 {"FALSE", "TRUE"};
property SHIFT_IN_B1 {"FALSE", "TRUE"};
property GSR {"ENABLED","DISABLED"};
```

MULT9X9ADDSUBSUM Primitive

```
MULT9X9ADDSUBSUM(CE0,CE1,CE2,CE3,CLK0,CLK1,CLK2,CLK3,RST0,RST1,RST2,RST3,SIGNEDAB,ADDNSUB1,AD
DNSUB3,
A00,A01,A02,A03,A04,A05,A06,A07,A08,
A10,A11,A12,A13,A14,A15,A16,A17,A18,
A20,A21,A22,A23,A24,A25,A26,A27,A28,
A30,A31,A32,A33,A34,A35,A36,A37,A38,
B00,B01,B02,B03,B04,B05,B06,B07,B08,
B10,B11,B12,B13,B14,B15,B16,B17,B18,
B20,B21,B22,B23,B24,B25,B26,B27,B28,
B30,B31,B32,B33,B34,B35,B36,B37,B38,
SRIA0,SRIA1,SRIA2,SRIA3,SRIA4,SRIA5,SRIA6,SRIA7,SRIA8,
SRIB0,SRIB1,SRIB2,SRIB3,SRIB4,SRIB5,SRIB6,SRIB7,SRIB8,
SROA0,SROA1,SROA2,SROA3,SROA4,SROA5,SROA6,SROA7,SROA8,
SROB0,SROB1,SROB2,SROB3,SROB4,SROB5,SROB6,SROB7,SROB8,
SUM0,SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,SUM7,SUM8,
SUM9,SUM10,SUM11,SUM12,SUM13,SUM14,SUM15,SUM16,SUM17,
SUM18,SUM19) is black-box
{
property REG_INPUTA0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA0_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA0_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTA1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA1_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA1_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTA2_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA2_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA2_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTA3_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTA3_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTA3_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB0_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB0_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB1_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB1_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB2_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB2_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB2_RST {"RST0","RST1","RST2","RST3"};
property REG_INPUTB3_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_INPUTB3_CE {"CE0","CE1","CE2","CE3"};
property REG_INPUTB3_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE0_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE0_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE1_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE1_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE2_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE2_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE2_RST {"RST0","RST1","RST2","RST3"};
property REG_PIPELINE3_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_PIPELINE3_CE {"CE0","CE1","CE2","CE3"};
property REG_PIPELINE3_RST {"RST0","RST1","RST2","RST3"};
property REG_OUTPUT_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_OUTPUT_CE {"CE0","CE1","CE2","CE3"};
property REG_OUTPUT_RST {"RST0","RST1","RST2","RST3"};
```

```
property REG_SIGNEDAB_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_0_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_0_RST {"RST0","RST1","RST2","RST3"};
property REG_SIGNEDAB_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_SIGNEDAB_1_CE {"CE0","CE1","CE2","CE3"};
property REG_SIGNEDAB_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB1_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB1_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB1_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB1_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB1_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB1_1_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB3_0_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB3_0_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB3_0_RST {"RST0","RST1","RST2","RST3"};
property REG_ADDNSUB3_1_CLK {"NONE","CLK0","CLK1","CLK2","CLK3"};
property REG_ADDNSUB3_1_CE {"CE0","CE1","CE2","CE3"};
property REG_ADDNSUB3_1_RST {"RST0","RST1","RST2","RST3"};
property SHIFT_IN_A0 {"FALSE", "TRUE"};
property SHIFT_IN_B0 {"FALSE", "TRUE"};
property SHIFT_IN_A1 {"FALSE", "TRUE"};
property SHIFT_IN_B1 {"FALSE", "TRUE"};
property SHIFT_IN_A2 {"FALSE", "TRUE"};
property SHIFT_IN_B2 {"FALSE", "TRUE"};
property SHIFT_IN_A3 {"FALSE", "TRUE"};
property SHIFT_IN_B3 {"FALSE", "TRUE"};
property GSR {"ENABLED","DISABLED"};
```