

## Synthesis for 1 Million Gate FPGAs: Synplicity Support for Xilinx<sup>®</sup> Virtex<sup>™</sup> Series

Virtex Architectural Highlights

### Elements

Basic element = Logic Cell (LC). LC contains a four input look-up table (LUT), and a D-type flip-flop (DFF)

Two LCs = Slice. A Slice can contain a five function multiplexer (MUXF5), which combines two LUTs, and a six function multiplexer (MUXF6)

Four LCs = Configurable Logic Block (CLB). CLB includes four programmable registers.

**Memory:** up to 131k bits on board memory

**Clocks:** 4 global clock buffers with digital Delay-Locked Loop driving four global clock nets

**Primitives:** Unified library of standard functions

**Cores:** Library of cores, including new 64-bit PCI core

### Overview

In 1999, FPGA technology is moving to even greater levels of density and functionality. New devices contain over a million logic gates and support designs with clock speeds that exceed 100 MHz. Price per gate continue to fall dramatically. The designer can now look to FPGAs not only for prototyping ASICs in the small to medium range of 50K to 200K gates, but also as an alternative to standard-cell ASICs. The fast time to market and the flexibility in development that typifies the FPGA now has the additional benefit of very large capacity and high clock speeds.

To work effectively with million gate FPGAs, designers require tools that free their design creativity while making the most efficient use of the resources on the chip. Synplify<sup>®</sup>, from Synplicity<sup>®</sup>, is the synthesis tool of choice for today's FPGA designers. Synplify continues to provide designers with the power required to handle these new devices. One of the first generation of million gate FPGAs is the Virtex series from Xilinx. The top-end members of this series contain not just a million logic gates, but also has functions that had required separate supporting chips, occupying board space and complicating overall system design. There is no longer a requirement for external phase lock loops, external voltage translation buffers or even, for many applications, external memory. Synplify 5.1 and later versions incorporate an advanced Synplify-Virtex mapper, developed in close cooperation with Xilinx. The result is a synthesis tool that recognizes and automatically exploits the features of the Virtex architecture, while providing

designers with the ability to exercise detailed, hands-on control. Synplify also includes support for Xilinx's Real-PCI™ 64/66 solution, a 64-bit 66 MHz PCI core for producing dense designs with high levels of performance.

### Synplify: Simply Better Synthesis® for Virtex

Synplify provides a comprehensive synthesis and HDL debugging environment for high-density high performance programmable logic design. The technology within Synplify doesn't force the designer into creating hierarchy purely for synthesis. Instead, B.E.S.T.™ (Behavior Extracting Synthesis Technology™) preserves abstract data created in the RTL input. throughout the synthesis process. This dramatically improves performance and reduces the area required since optimization can operate globally.

DST™ (Direct Synthesis Technology) maps the design directly onto the cell structure of the target device, using device and architecture specific knowledge. By taking advantage of device features, DST further improves the design's performance and reduces the area it uses.

The user interface is designed to make Synplify intuitive and easy to use. Behind the interface are intelligent default values based on knowledge of the target device. However, the more experienced user can take control whenever necessary, overriding defaults, setting values or undertaking detailed hand tuning.

When hand tuning is required, the SCOPE™ graphical constraints editor allows the definition of synthesis and place and route constraints within a single, easy to use spread sheet.

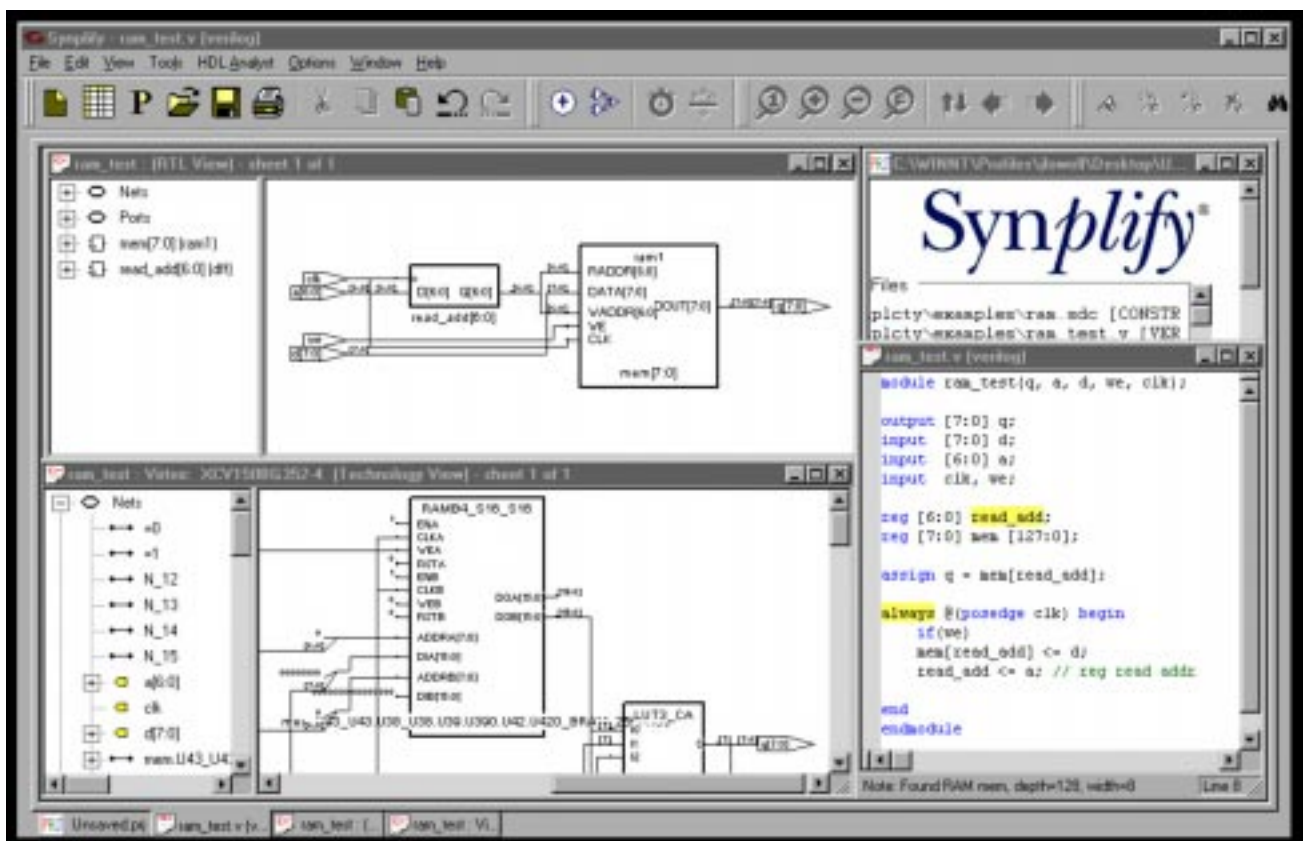


Figure 1: Automatic RAM inferencing

## Synplify-Virtex Mapping Algorithm

### **Improvements delivered in v.5.1**

The enhanced Synplify-Virtex mapping algorithm provides improved language recognition and better implementation of functions within the Virtex architecture. Some of the architecture-specific mapping techniques include:

### **Improved counter inferencing**

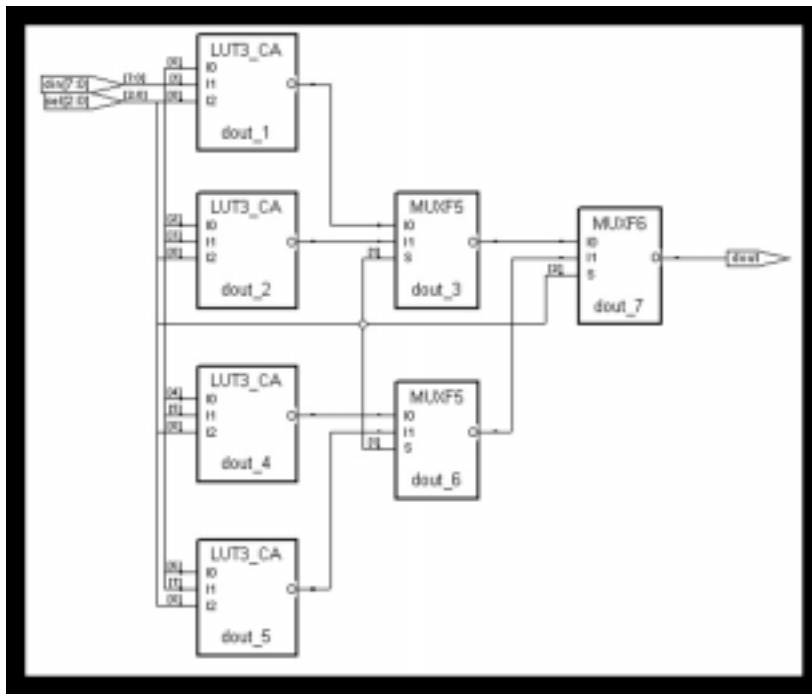
Synplify selects the appropriate counter architecture by using timing-based criteria to make trade-offs between area and performance. Synplify automatically extracts up or down counters and loadable counters, and implements smaller loadable counters.

### **Flip-flops**

The Virtex mapper improves I/O performance by automatically duplicating flip-flops. It also optimizes the clock-to-output delay for registered tri-state outputs when the enable is also registered.

### **Wide functions**

Synplify implements wide functions by combining the resources of the Virtex CLBs. It maps an 8-to-1 multiplexer by using only the F5 multiplexer and F6 multiplexer resources contained within one CLB. This produces a dense implementation, which, since there is only one level of logic, is also fast.



**Figure 2: HDL Analyst Technology View 8-to-1 MUX**

```

/** Description of an 8-to-1 mux */
/** Depending on the value of sel, */
/** dout gets on of the 8 bits of din */

```

```

module mux8x1 (din, sel, dout);
input [7:0] din;
input [2:0] sel;
output dout;
assign dout=din[sel];
endmodule

```

### Synplify-Virtex User Control and IP Support

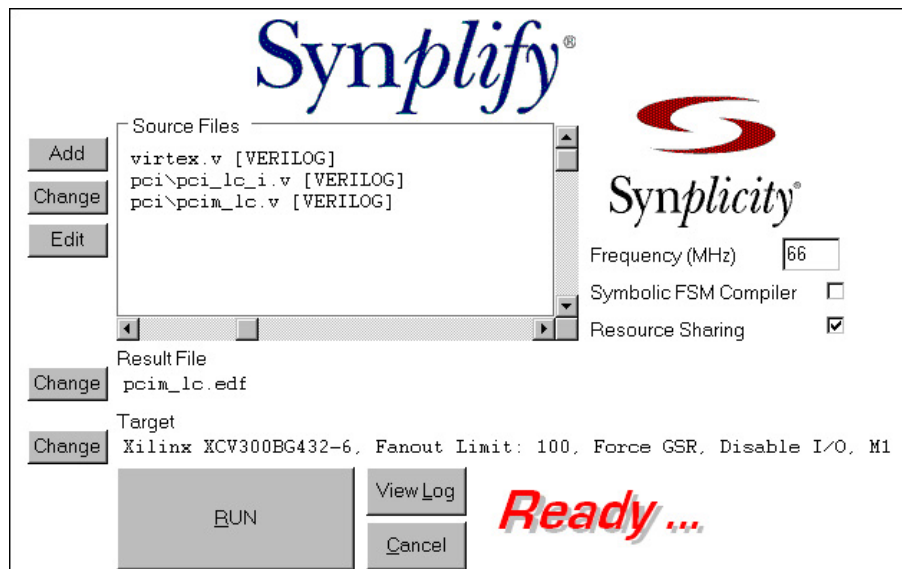
The designer can exercise control over how Synplify maps certain Virtex architectural features and provides state-of-the-art support for IP. Some of the control and IP support includes:

#### **Tri-state to multiplexer**

A unique Synplicity attribute gives users control over tri-state to pmux conversion. When nets are driven purely by tri-state buffers, the user has the option of specifying that these should be converted to a faster multiplexer implementation.

#### **Block RAMs**

Synplify infers from VHDL or Verilog source code synchronous RAMs and maps them to Virtex Block



**Figure 3: Real-PCI 64/66 Support**

SelectRAM+ (the dedicated fast, discrete and large blocks of RAM in Virtex devices). The Synplify-Virtex mapper uses the dedicated RAM resources to produce the best implementation for the user's design.

#### **The Real-PCI 64/66**

The Real-PCI 64/66 is a 64-bit 66MHz PCI core for the Virtex series. Synplify supports the core and integrates it economically into designs.

The designer can build PCI products quickly, with predictable results.

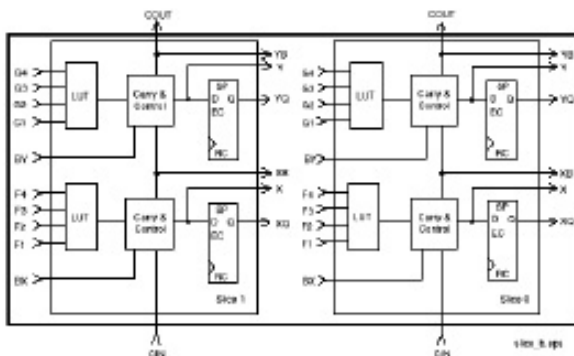
These and other functions make the new Synplify-Virtex mapper a powerful tool to implement the Virtex series.

### Summary

Synplicity's Synplify v. 5.1 and the new Synplify-Virtex mapper target all members of the Virtex series from the Xilinx XCV50™ device, with 50,000 system gates, to the Xilinx XCV1000™ one-million gate FPGA. The mapper produces results that efficiently use chip real estate while providing speeds of up to 160 MHz. As further devices are developed, reaching 2 million or more gates, Synplify will remain the synthesis tool that allows the designer to concentrate on creativity.

### The Virtex Architecture

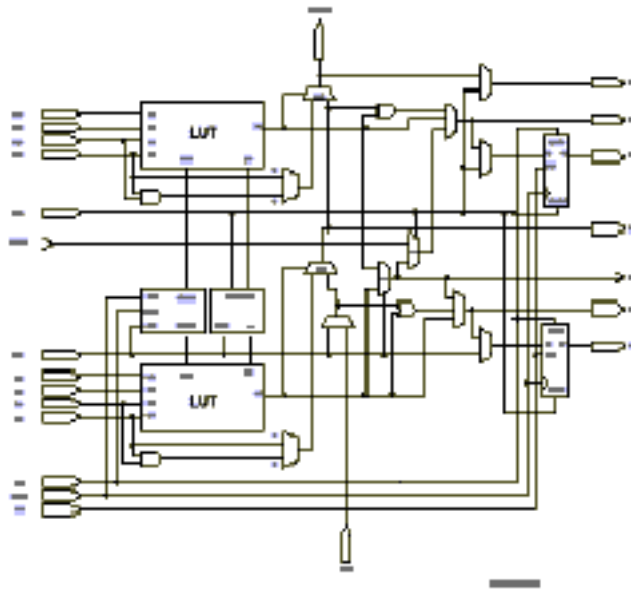
The Virtex series architecture features memory support, dedicated carry and cascade logic, internal tri-state bussing capability and built-in clock management resources.



**Figure A1**

The basis of the architecture is a new configurable logic block (CLB) as shown in Figure A1. A CLB contains four logic cells (LCs). Within each LC is a four input look-up table (LUT), a D-type flip-flop (DFF) with set/reset, and carry and control logic. The DFF may also be used as a latch and the LUT can operate as a function generator and as a 16 by 1-bit synchronous RAM.

A pair of LCs is called a Slice (Figure A2). This can contain a five function multiplexer (MUXF5), which combines two LUTs, and a six function multiplexer (MUXF6) which maps any six input functions in one CLB. The two LUTs within a Slice may be combined to create a 16 by 2-bit



**Figure A2**

synchronous RAM, a 32 by 1-bit synchronous RAM or a 16 by 1-bit dual-port synchronous RAM.

Within the CLB are four programmable registers. These are configurable as latches, synchronous flip-flops or asynchronous flip-flops. Each register has a dedicated Enable control and a Set and Preset control. Each slice shares a common clock enable, programmable on a per slice basis, and two storage elements share a clock pin. There is a global set/reset line.

The architecture includes three levels of support for memory:

- the distributed memory of the LUTs within the CLBs (Distributed SelectRAM+ Memory)
- dedicated 4 k bit blocks of memory (Block SelectRAM+ Memory) that can be configured from 4 k by 1-bit to 256 by 16-bit
- an interface, running at over 133 MHz, for access to off chip memory

Each Virtex device has 4 global clock buffers driving four primary global nets, each with an associated fully digital Delay-Locked Loop to eliminate skew and control multiple clock domains.

Virtex FPGAs are supported by a unified library of standard functions that contains more than 400 primitives and macros, ranging from 2-input AND gates to 16-bit accumulators, and includes arithmetic functions, comparators, counters, data registers, decoders, encoders, I/O functions, latches, Boolean functions, multiplexers, shift registers, and barrel shifters.

There are cores for a range of functions, with further cores under development. An important core is The Real 64/66 PCI core. This conforms to the new PCI standard, with data rates of 66MHz, making it ideal for graphics and related applications. The core only occupies 3 percent of the largest Virtex device (the XCV1000), leaving 97 percent free for other custom logic implementations.



Copyright © 1999 Synplicity, Inc. All rights reserved. Specifications subject to change without notice. *Synplicity*, the *Synplicity* 'S' logo, *HDL Analyst*, and *Synplify* are registered trademarks; and *Certify*, B.E.S.T. (Behavior Extracting Synthesis Technology), and DST (Direct Synthesis Technology) are trademarks of Synplicity, Inc. All other names mentioned herein are trademarks or registered trademarks of their respective owners.