CHAPTER 6

FPGA IMPLEMENTATION

6.1 INTRODUCTION

Xilinx FPGAs is a very effective platform to realize filters in hardware. Content-Addressable Memories (CAMs) that do exact matching can be used to implement best matching prefix. A scheme in McAuley et al (1995) uses a separate CAM for each possible prefix length. This can require 32 CAMs for IPv4 and 128 CAMs for IPv6, which is expensive. It is possible to obtain CAMs that allow, “don’t care” bits in CAM entries to be masked out. Such designs only require a single CAM. However, the largest such CAMs today only allow around 8000 prefixes. While such a CAM may be perfectly adequate for an enterprise router, it is inadequate for a backbone router. Finally, CAM designs have not historically kept pace with improvements in RAM memory. Thus any CAM solution runs the risk of being made obsolete in a few years by software running on faster processors and memory. So in this work a DRAM based Xilinx series is used.

This series provides hard macros for network access, has ample memory resources to store filter rules, and provides embedded processors to manage and maintain the built network devices. This pipelined packet-filter architecture on Spartan 2E FPGAs is able to filter Gigabit Ethernet traffic. On a small Spartan 2E xc2s50e-6ft256, (Xilinx Corporation, 2005, 2006). Thirty two complex filter rules can be applied to decide the acceptance (or dropping) of packets by applying linear search. The filter causes a latency of only 2,300 nsec, which is faster than software solutions. In this work, the implementation of a packet filter for Gigabit Ethernet being realized on Xilinx FPGAs is presented.
The complexity of Xilinx FPGAs and their ability to be reconfigured in the field turns out be valuable assets that facilitate the implementation of flexible network applications.

A pipelined architecture on a Xilinx Spartan 2E xc2s50e-6ft256 device is presented in this work, that is able to filter network traffic on layers 2 and 3 at a throughput rate of 1 Gbps. This proof-of-concept on the rather small xc2s50e-6ft256 device is limited to handle 32 filter rules. Moreover, it supports 32 entries of routing information for routing packets to their final destination. The approach is scalable for a medium number of filter rules on larger FPGA devices. The architecture is tuned to utilize Xilinx FPGA resources like distributed RAM, Block RAM and of course the integrated hard-macro Media Access Controller (MAC) for Ethernet access. It does not use ternary memory but instead utilizes the enormous bandwidth to the on-chip RAM resources.

6.2 OVERVIEW OF SPARTAN II E

The Xilinx Spartan II E FPGA circuit board provides an inexpensive and expandable platform on which to design and implement digital circuits of all kinds. Spartan board features include:

- 1.5A power regulators (2.5V and 3.3V).
- A socketed 50MHz oscillator.
- An EPP-capable parallel port for JTAG based FPGA programming and user data transfers.
- A 5-wire Rs-232 serial port.
A status LED and pushbutton for basic I/O.

Six 100-mil spaced, right-angle DIP socket 40-pin expansion connectors.

This board has been designed specifically to work with the Xilinx ISE CAD tools, including the free WebPack tools available from the Xilinx website. A power supply and programming cable, so designs can be implemented immediately without the need for any additional hardware.

6.2.1 Configurable logic block

Figure 6.1 Spartan II CLB slice
Figure 6.1 shows the basic building block of the Spartan-II CLB is the logic cell (LC). An LC includes a 4-input function generator, carry logic, and storage element. Output from the function generator in each LC drives the CLB output and the D input of the flip-flop. Each Spartan-II CLB contains four LCs, organized in two similar slices; a single slice is shown in Figure 6.1. In addition to the four basic LCs, the Spartan-II CLB contains logic that combines function generators to provide functions of five or six inputs.

6.2.2 Lookup tables

Spartan-II function generators are implemented as 4-input lookup tables (LUTs). In addition to operating as a function generator, each LUT can provide a 16 x 1-bit synchronous RAM. Furthermore, the two LUTs within a slice can be combined to create a 16 x 2-bit or 32 x 1-bit synchronous RAM, or a 16 x 1-bit dual-port synchronous RAM. The Spartan-II LUT can also provide a 16-bit shift register that is ideal for capturing high-speed or burst-mode data. This mode can also be used to store data in applications such as Digital Signal Processing.

6.2.3 Storage elements

Storage elements in the Spartan-II slice can be configured either as edge-triggered D-type flip-flops or as level-sensitive latches. The D inputs can be driven either by function generators within the slice or directly from slice inputs, bypassing the function generators. In addition to Clock and Clock Enable signals, each slice has synchronous set and reset signals (SR and BY). SR forces a storage element into the initialization state specified for it in the configuration. BY forces it into the opposite state. Alternatively, these signals may be configured to operate asynchronously. All control signals are independently invertible, and are shared by the two flip-flops within the slice.
6.2.4 Additional logic

The F5 multiplexer in each slice combines the function generator outputs. This combination provides a function generator that can implement any 5-input function, a 4:1 multiplexer, or selected functions of up to nine inputs. Similarly, the F6 multiplexer combines the outputs of all four function generators in the CLB by selecting one of the F5-multiplexer outputs. This permits the implementation of any 6-input function, an 8:1 multiplexer, or selected functions of up to 19 inputs. Each CLB has four direct feedthrough paths, one per LC.

These paths provide extra data input lines or additional local routing that does not consume logic resources.

6.2.5 Arithmetic logic

Dedicated carry logic provides fast arithmetic carry capability for high-speed arithmetic functions. The Spartan-II CLB supports two separate carry chains, one per slice. The height of the carry chains is two bits per CLB. The arithmetic logic includes an XOR gate that allows a 1-bit full adder to be implemented within an LC. In addition, a dedicated AND gate improves the efficiency of multiplier implementation. The dedicated carry path can also be used to cascade function generators for implementing wide logic functions.

6.2.6 BUFTs

Each Spartan-II CLB contains two 3-state drivers (BUFTs) that can drive on-chip busses. Each Spartan-II BUFT has an independent 3-state control pin and an independent input pin.
6.2.7 Block RAM

The structure of Block RAM is given in Table 6.1. Spartan-II FPGAs incorporate several large block RAM memories. These complements the distributed RAM Look-Up Tables (LUTs) that provide shallow memory structures implemented in CLBs. Block RAM memory blocks are organized in columns. All Spartan-II devices contain two such columns, one along each vertical edge. These columns extend the full height of the chip. Each memory block is four CLBs high, and consequently, a Spartan-II device eight CLBs high will contain two memory blocks per column, and a total of four blocks.

Table 6.1 Spartan II Block RAM Amounts

<table>
<thead>
<tr>
<th>Spartan II devices</th>
<th>No. of blocks</th>
<th>Total Block RAM bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>XC 2S 15</td>
<td>4</td>
<td>16K</td>
</tr>
<tr>
<td>XC 2S 30</td>
<td>6</td>
<td>24K</td>
</tr>
<tr>
<td>XC 2S 50</td>
<td>8</td>
<td>32 K</td>
</tr>
<tr>
<td>XC 2S 100</td>
<td>10</td>
<td>40K</td>
</tr>
<tr>
<td>XC 2S 200</td>
<td>14</td>
<td>56K</td>
</tr>
</tbody>
</table>

Each block RAM cell is a fully synchronous dual-ported 4096-bit RAM with independent control signals for each port. The data widths of the two ports can be configured independently, providing built-in bus-width conversion. The Spartan-II block RAM also includes dedicated routing to provide an efficient interface with both CLBs and other block RAMs.
6.3 ROUTER GENERAL ARCHITECTURE

Different kinds of memory reside in the Cisco 1600 Series Router, and each of them is handled in a different way and for different purposes.

6.3.1 RAM

![Figure 6.2 Memory details](image)

Figure 6.2 illustrates the memory, which is logically divided into Main processor memory and Shared Input/Output (I/O) memory.

6.3.1.1 Main processor memory

It is used to store routing tables, fast switching cache, running configuration, and so on. It can take unused shared I/O memory, if needed.
6.3.1.2 Shared I/O memory

It is used for temporary storage of packets in system buffers at the time of process switching, and interface buffers during fast switching. Cisco 1600 Series Routers running Cisco IOS software versions prior to the integration of CSCdk40685 (registered customers only) have a fixed I/O memory of 512 KB. After CSCdk40685, if the router has enough memory, it allocates 25% to I/O memory. If not, I/O memory remains at 512 KB.

Physically, DRAM is a combination of 2 MB on-board non-parity chips and one Single In-line Memory Module (SIMM) (72-pin, 60 ns, with or without parity). If SIMM is non-parity, total DRAM can be up to 18 MB. If SIMM is with parity, total DRAM can be up to 16 MB (on-board 2 MB will be disabled).

6.4 THE WORKING MODEL OF XILINX SPARTAN II E

A prototype of the described gateway system on a Xilinx ML403 evaluation board has been implemented in this work. It filters incoming packets and sends them back to the source of the packet. It can handle all sizes of Ethernet packets and is not limited to maximum number of packets per second. The Xilinx Spartan II E allows using its configurable logic blocks (CLBs) in a distributed-RAM mode. This means, that the DRAM of the CLBs can be used to provide memory elements instead of lookup-tables (LUTs). These LUTs usually hold data representing combinational logic. Each LUT can implement a 16×1-bit synchronous RAM. Multiple LUTs can be combined to provide larger single ported and dual-ported memory elements. Because distributed RAM can be sized very fine-grained, it is a good choice for memory elements with unusual aspect ratios like our 16 × 31 1-bit memory. The
external DRAM which combined with the Spartan xc2s50e-6ft256 devices is shown in Figure 6.3.

![Figure 6.3 Working model of FPGA](image)

6.5 FPGA UTILIZATION

After completing the full module the program has been run into the selected device and the devices utilization summary is given below.

### 6.5.1 Device utilization summary

<table>
<thead>
<tr>
<th>Component</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selected device: 2s50eft256-6</td>
<td></td>
</tr>
<tr>
<td>Number of slices:</td>
<td>91 out of 768</td>
</tr>
<tr>
<td></td>
<td>11%</td>
</tr>
<tr>
<td>Number of slice Flip-Flops:</td>
<td>18 out of 1536</td>
</tr>
<tr>
<td></td>
<td>1%</td>
</tr>
<tr>
<td>Number of 4 input LUTs:</td>
<td>175 out of 1536</td>
</tr>
<tr>
<td></td>
<td>11%</td>
</tr>
<tr>
<td>Number of bonded IOBs:</td>
<td>17 out of 182</td>
</tr>
<tr>
<td></td>
<td>9%</td>
</tr>
<tr>
<td>Number of GCLKs:</td>
<td>1 out of 4</td>
</tr>
<tr>
<td></td>
<td>25%</td>
</tr>
</tbody>
</table>
The used Xilinx Spartan 2E FX-12 FPGA is utilized to 54% (5,913) of the Slice Flip Flops. About 46% of the FPGA LUTs are used. A rough estimation shows that the FPGA is approximately utilized to 50%. The realized systems comprehend approximately 50% of all necessary components in comparison to a system including both inbound and outbound filter. The Spartan 2E FX-20 is the FPGA next in size providing 82% more LUTs than the one be used. With this FPGA it will be possible to implement the complete filter (not considering the IPsec modules).

The prototype is able to run at a maximum frequency of 140 MHz. This is above the required 125 MHz for Gigabit Ethernet networking. The system causes a packet latency of 283 clock cycles (2,264 nsec at 125 MHz) for every packet, independent from the packets protocol, number of filter rules, or the action of the matching filter rule.

6.6 CONCLUSION

In this research a Gigabit packet filter tailored for implementation on FPGAs is presented. This solution stores the filter rules in distributed RAM on the FPGA and uses the linear search algorithm to find a matching rule. It exploits the fact that internal RAM resources on FPGAs can be connected with high data widths and do not face the limitation of restricted pin counts as when using external memories. A prototype on the Xilinx ML403 board, achieving one Gigabit throughput and a latency of only 2,264 nsec, which is a magnitude faster than comparable software approaches has been implemented. Furthermore these results show that FPGAs are not only useful as prototyping platforms but are also well suited for Gigabit network filtering in real-life applications. This is in particular true when the specifics of FPGAs are taken into account when choosing the system architecture.