UNIVERSITY
OF TRENTO

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

38050 Povo – Trento (Italy), Via Sommarive 14
http://www.dit.unitn.it

SoftTOTEM: AN FPGA IMPLEMENTATION OF THE TOTEM PARALLEL PROCESSOR

Stephanie McBader, Luca Clementel, Alvise Sartori,
Andrea Boni and Peter Lee

2002

Technical Report # DIT-02-0033

Also: to be published in Proceedings Field Programmable logic and Applications 2002

SoftTOTEM: An FPGA Implementation of the TOTEM Parallel Processor

Stephanie McBader[1], Luca Clementel[1], Alvise Sartori[1], Andrea Boni[2], Peter Lee[3]

[1] NeuriCam S.p.A, Via S M Maddalena 12, 38100 Trento (TN), Italy
[mcbader , clementel , sartori]@neuricam.com
[2] University of Trento, Via Mesiano, 77, I-38050 Povo (TN), Italy
andrea.boni@ing.unitn.it
[3] University of Kent at Canterbury, Canterbury, Kent, CT2 7NT, UK
P.Lee@ukc.ac.uk

Abstract. TOTEM is digital VLSI parallel processor ideally suitable for vector-matrix multiplication. As such, it provides the core computational engine for digital signal processing and artificial neural network algorithms. It has been implemented in the past as a full-custom IP core, achieving high data throughput at clock frequencies of up to 30 MHz. This paper presents the 'soft' implementation of the TOTEM neural chip, and compares its cost and performance to previous 'hard' implementations.

1. Introduction

NeuriCam is an innovation-oriented company operating in the design and fabless production of VLSI circuits and modules for machine vision and embedded computing. One of its main line of products is that of the TOTEM neural processors [1,2], which integrate parallel processing units to achieve high performance in object recognition and classification. One TOTEM chip flexibly implements multi-layer perceptrons with up to tens of layers, over 8000 connections and a width of up to 32 neurons. An example of this is a 220-32-32 topology where the input layer is formed with 220 inputs, representing the input object to be classified. These connect to a hidden layer of 32 nodes, which in turn produces 32 outputs representing the class to which the input object belongs. Larger networks can be constructed by paralleling 2 to 16 chips, for a maximum network width of 255 neurons.

The versatile memory-like interface makes TOTEM very easy to integrate with microcontrollers to build recognition systems [3], filters or signal compressors [4], and fuzzy controllers based on neural networks [5]. It can also be used to implement complex computations required in DSP operations [6].

Previous implementations of TOTEM have been realised in full-custom VLSI design. This, of course, resulted in highly powerful chips of optimum layouts at minimal power consumption. However, rapid advances in process technologies require that the full-custom layout is manually modified to match newer, smaller transistor

dimensions. In fact, any modification or enhancement on the TOTEM architecture resulted in several man-months of effort and a new silicon run.

The alternative, 'fast-track' approach to this problem necessitates the existence of a 'soft' core, designed at a high level of abstraction. This paper presents the merits of such approach, and reports the obtained performance and incurred costs from porting the TOTEM architecture into its VHDL equivalent, SoftTOTEM, and implementing it on a Xilinx FPGA.

2. Architecture Overview

The TOTEM SIMD architecture is based on a pipelined digital data stream, feeding 32 fixed-point fully-parallel multiply-and-accumulate processors (MACs), as shown in figure 1. The architecture is optimised for the execution of the MAC operation:

```
Acc(n+1) = Acc(n) + DataIn*Weight(n)
```

Where 16-bit `DataIn` is received from a common broadcast bus, in 2's complements format. Weights or filter parameters are stored locally to each neuron. Weight blocks are organised as 256x10-bits and are closely coupled to the MAC units. The 34-bit output of the accumulator is loaded into an output register before passing through a 34-bit input/16-bit output barrel shifter for scaling of results to the 16-bit interface.

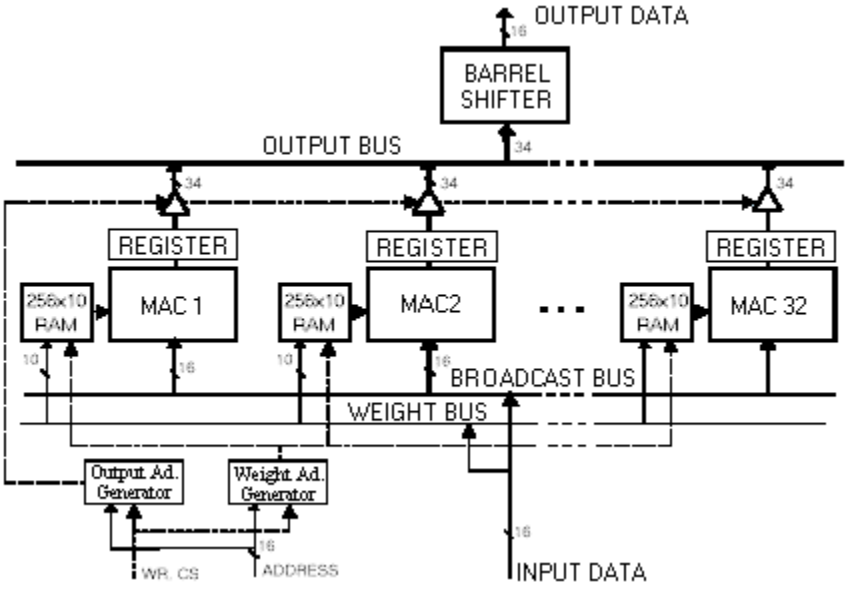

Fig. 1. TOTEM Architecture

The TOTEM chip has a very simple memory-like interface, comprising input address and data busses, output data bus, and control busses for co-processor operation in microprocessor systems. A simple controller decodes input instructions and issues all the internal control signals required to operate the parallel processors.

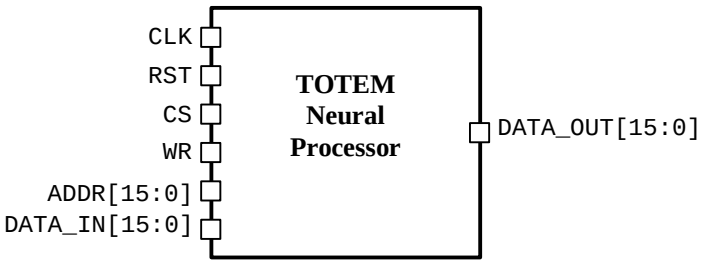

Fig. 2. TOTEM Interface

Instruction	Description
`setbarrel`	Set Barrel Shifter Window
`outneuron`	Select Output Neuron
`writewmem`	Write to Weight Memory
`startaddr`	Set Memory Start Address
`lregister`	Load Result Register
`clearaccu`	Clear Accumulator
`increment`	Increment Memory Address
`calculate`	Compute MAC Operation
`calcincrm`	Calculate then Increment Memory Address

Table 1. TOTEM Instruction Set

3. SoftTOTEM Implementation

SoftTOTEM was implemented on a Xilinx XCV600 FPGA using VHDL for porting the full-custom design of TOTEM. Weight memories were instantiated as Xilinx-specific RAM Blocks, while the multiplier component of the MAC unit was implemented first as a Xilinx core, and then as a soft IP core using the Pezaris algorithm for multiplication.

3.1 Test Hardware

Figure 3 illustrates the prototyping platform used to test SoftTOTEM. The soft neural processor communicates with the host PC using a standard parallel port working in

EPP mode. The clock oscillator available on the prototype board is not hard-wired, therefore providing flexibility to test the soft implementation over a range of clock frequencies. External I/O pins are available for debugging the implementation and monitoring communication with the host.

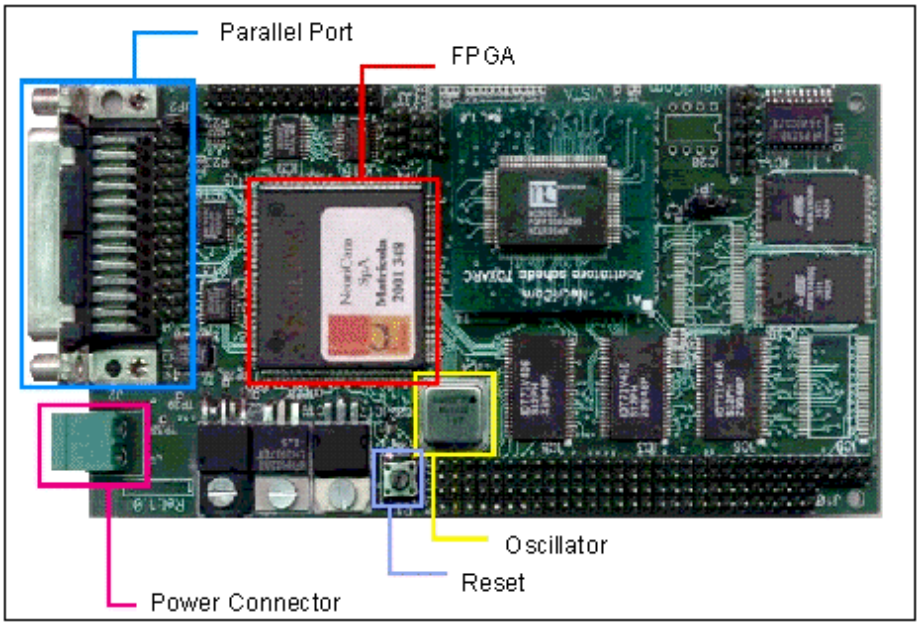

Fig. 3. Prototyping Platform

However as it will be seen later, the Virtex 600 FPGA available on this board could not accommodate all 32 neurons due to the fact that it only contains 24 RAM Blocks when 32 are needed. Because of this, several memory-neuron configurations were implemented:

- SoftTOTEM with only 16 neurons
- SoftTOTEM with 32 neurons, weight memory organised as 128x10-bits, where two neurons share a single RAM Block of 256x10-bits
- SoftTOTEM with 32 neurons and 256x10-bit weight memories, implemented on another prototyping platform with a Xilinx XCV2000-E FPGA.

3.2 Test Software

SoftTOTEM was evaluated using the same approach that was used to test TOTEM chips. A test program generates random test vectors, and as it sends instructions to the hardware, it passes a copy to a TOTEM emulator, which computes the expected output in software. Once the test vectors have all been consumed, the output of the 32 neurons is read and compared to those of the emulated neural array. Figure 4 illustrates an example test of SoftTOTEM.

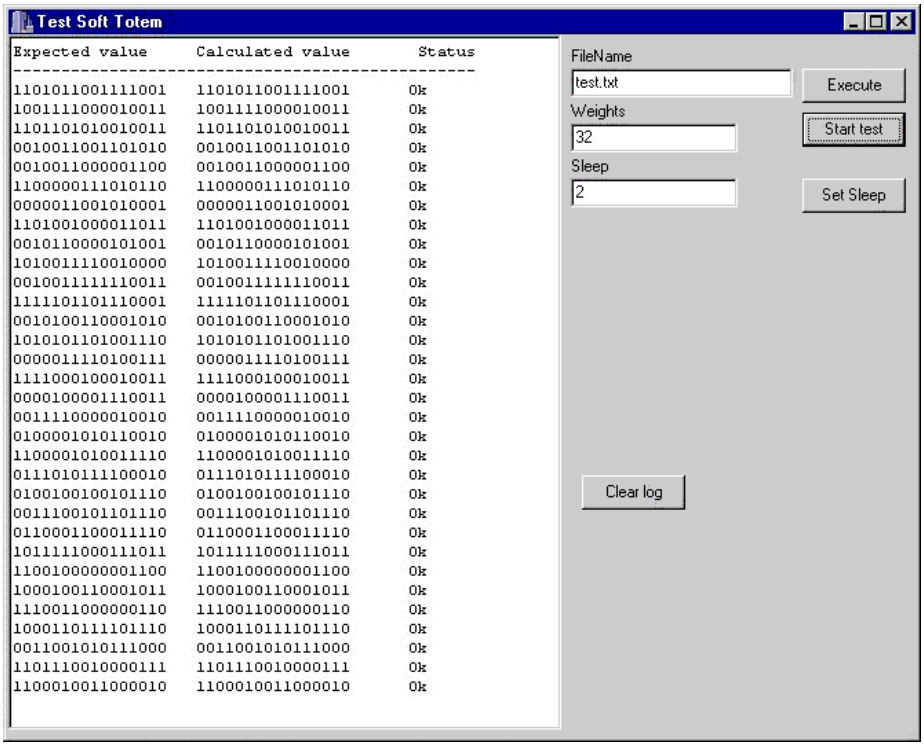

Fig. 4. Testing SoftTOTEM

4. Soft vs. Hard Implementations: Comparison

Synthesis and implementation results have shown that it should be possible to implement a complete SoftTOTEM chip, comprising 32 neurons, 256x10-bit weight memories as well as interfacing and control, on a single Virtex 600-E FPGA, because it contains sufficient RAM blocks on-chip, and the logic would take up 95% of the device.

Table 2 highlights the main characteristics of both neural processor implementations. It can be seen that SoftTOTEM in fact outperforms its full-custom equivalent, at the cost of area and higher power consumption. This cost, however, is not a threatening one – products that are based on the TOTEM chip normally require an FPGA on-board to provide control for the neural array, as well as communication with the external system [7].

Characteristics	TOTEM	SoftTOTEM
Number of Processors	32	32
On-Chip RAM	80 K	80 K
Number of Connections	8192	8192
Max. Network Width	32	32
Processing Power	960 MOPS	3880 MOPS
Max. Clock Frequency	30 MHz (TOTEM limitation)	40 MHz (EPP limitation)
Area	747,166 transistors 186,792 gates? (Exc. Control)	718,076 gates 6,570 Slices (Inc. Control)
Power Consumption	1W @ 5V	2.455 W @ 1.8 V
Man-Months: Complete Design	6-12 Months	3 Months
Man-Months: Modification	4~6 months	< 1 week
Implementation	Min. 3 months (silicon run)	A few hours
Price/Chip	$42.4 (Exc. NRE)	$378 XCV600-E (Feb. 2002)

Table 2. TOTEM vs. SoftTOTEM

5. Conclusion & Further Work

The SoftTOTEM design and implementation illustrates a typical example of the merit of using field programmable logic as opposed to full-custom VLSI design. Fast development times and reduced efforts are not the only advantages of the 'soft' approach to implementing neural processing chips. It has been demonstrated that the soft implementation outperforms its full-custom rival, as it benefits directly from improvements in FPGA technology. The generic nature of the FPGA building blocks enabled the integration of both the arithmetic core of the neural processor and its logic control on a single device. The programmability of FPGAs opens up opportunities for exploring the architecture and implementing a choice of interfacing protocols at negligible costs. Moreover, the use of VHDL permits design re-use and the illustrates the merits of design portability.

Future work would concentrate on the advantages of integrating a host RISC processor into the FPGA, so as to complete a stand-alone system based on the neural approach to problem solving in a variety of applications. This level of integration has become even simpler than ever with the introduction of hard processing cores and multipliers into FPGA architectures [8]. High-density FPGAs make it possible to

implement even larger neural networks by incorporating more than one instance of SoftTOTEM on a single chip.

It can also be said that porting the arithmetic core into a soft implementation & incorporating it with the controller on a single FPGA is both a cost-effective and worthwhile solution. ASIC production, even though generally considered cost-effective in moderate volumes, could be equally if not more expensive than FPGA solutions once non-recurring engineering (NRE) charges incurred during the development phase are taken into account. NRE/FPGA-to-ASIC conversion fees add a considerable cost to ASIC production; this overhead could be in the range of tens of thousands of dollars per customer [9].

References

1. R. Battiti, P. Lee, A. Sartori, G. Tecchiolli, "TOTEM: a Digital Processor for Neural Networks and Reactive Tabu Search", MICRONEURO 94.
2. NeuriCam's NC3003 Datasheet: TOTEM Digital Processor for Neural Networks, Rel. 12/99
3. NeuriCam's Number Plate Recognition System (www.neuricam.com)
4. R. Battiti, A. Sartori, G. Tecchiolli, P. Tonella, A. Zorat, "Neural Compression: an Integrated Application to EEG Signals", IWANT95.

A. Zorat, A. Satori, G. Tecchiolli, L. Koczky, "A Flexible VLSI Processor for Fast Neural Network and Fuzzy Control Implementation", LIZUKA96.

5. NeuriCam's Application Note AN005, "Using the NC3001 for DSP Applications: computing the DFT of a 256x256 image".
6. NeuriCam's Parallel Signal Processing Boards: TOTEM PCI Technical Reference Manual (Rel. 12/99)
7. Xilinx: The Programmable Logic Data Book 2001.
8. Peter Alfke, "Evolution, Revolution and Convolution. Recent Progress in Field-Programmable Logic", FPL2001.
9. Xilinx: Spartan ASIC Alternatives.
10. S. Dusini *et al*, "The Neurochip Totem in the Higgs Search", ABANO96.
11. L. Ricci, G Tecchiolli, "A Neural System for Frequency Control of Tunable Laser Sources", IEEE IMTC'97.
12. F. De Nittis, G Tecchiolli, A. Zorat, "Consumer Loan Classification Using Artificial Neural Networks", EIS'98.