

Scalability of FPGA to Neurons

Pages suppressed are under CDA.

Colin James III

Ersatz Systems Machine Cognition, LLC

Colorado Springs CO

Table of Contents

U = Unpublished (0); P = Published (19)

Introduction

P Verifiable statistics required

Number of Neurons

P Number of neurons downgraded

Number of Synapses

P Number of synapses downgraded

Number of Transistors

P Manufacturer claims largest FPGA's

Number of Transistors per Neuron

P Predictions

Software and Hardware Performance

P Software performance

P Hardware performance

Scalability of FPGA

P Current status

P Predictions

P Other factors

Appendix

P Source code for software test

Introduction

Published Verifiable statistics are required for:

Number of human neurons

Number of synapses connected to neurons

Number of transistors in Field Programmable Gate Arrays (FPGAs),
pathways connecting transistors in FPGAs, and look up tables (LUTs)

Speed of execution for software and hardware of logical circuitry

Speed of execution for software and hardware of look up tables (LUTs)

Estimation of FPGA resources need to implement neurons

Number of Neurons

Published Number of neurons downgraded

The number of human neurons bandied about is 100-150 billion
($1-1.5 * 10^{11}$).

There is no verifiable source or evidence for that arbitrary range.

Professor Susanna Herculano-Houzel of Brazil (2005 et seq) asserts 86 billion ($8.6 * 10^{10}$) neurons, a downgrade of 15%.

This is based on “isotrophic fractionator”, a scaled count of a sample of neurons from a soup made from a deceased human brain.

The number of glia(l) cells that support neurons with oxygen, is also 81 billion ($8.1 * 10^{10}$) or set at roughly a 1:1 ratio with neurons.

Number of Synapses

Published Number of synapses downgraded

The unverifiable statistic in biology is that the number of synapses per neuron is 2,000 to 7,000 ($2 \cdot 10^3$ to $7 \cdot 10^3$) or ratio 1/2000 - 1/7000.

Such an estimate admits a margin of error of over 55%.

From Reep et al, 2008, in “Quantification of synaptic density ...”, the rat cortex contains the ratio of axon to synapse of 0.50 or 1:2.

There is one axon (output) per neuron; hence there are two synapses per neuron, not 2,000 to 7,000 per neuron.

(Thus is the state of non reproducible medical statistics.)

Number of Transistors

Published Manufacturer claims largest FPGA's

The Xilinx Vertix-7 FPGA has 6.8 billion transistors ($6.8 * 10^9$).

The Xilinx Vertix-7 FPGA has 10,000 data paths between transistors (10^4).

The Xilinx Vertix-7 FPGA has six LUTs, configurable in various ways as 32- and 64-bit units.

Number of Transistors per Neuron

Published Calculations

For hardware circuits, the logical connectives AND and OR each require 4-transistors for an AND-OR gate (AO) of 8-transistors as a neuron.

The 6.8 billion transistors in a Vertix-7 FPGA then accommodates about 850 million such AO gates ($8.5 * 10^8$) as neurons.

For a hardware LUT, a 4-input LUT (4-LUT) is fracturable into two 2-input LUTs (2-LUT).

The Vertix-7 FPGA has $8 * 6$ -LUTs, hence accommodating 24 2-LUTs.

With 24 2-LUTs, this means 24 concurrent look ups are possible that map into 6.8 billion transistor outputs for that number of neurons.

Software and Hardware Performance

Published Software performance

Software circuit emulation is at the rate of 2,981,959 completions per second ($2.98 * 10^6$).

Software LUT emulation is at the rate of over 11.1 million completions per second ($1.11 * 10^7$).

Software source code for the tests appears in the Appendix.

Published Hardware performance

FPGA performance is faster than software performance by a factor of about 1000 (10^3).

Hardware circuit performance is estimated to be at about 3 billion completions per second ($2.98 * 10^9$).

Hardware LUT performance is estimated to be at about 11 billion completions per second ($1.1 * 10^{10}$).

Scalability of FPGA

Published Current status

Given 850 million neurons ($8.5 * 10^8$) and the hardware rate of $2.98 * 10^9$ circuit completions per second, then the ratio of circuit completion per second per neuron is about 3.506 with inverse of 0.285 or about **0.29 neurons per hardware circuit completion per second.**

Given 86 billion neurons ($8.6 * 10^{10}$) and the hardware rate of $1.11 * 10^{10}$ completions per second, then the ratio of LUT completions per second per neuron is about 0.1290 with inverse of 7.75 or about **8 neurons per hardware LUT completion per second.**

Published Predictions

If the number of neurons is scaled down to 30 billion (by a factor of about 1/3 less than 86 billion above) then these estimates follow.

The ratio of circuit completion per second per neuron is $30/86 * 0.209$ or **0.10 neurons per hardware circuit completion per second.**

The ratio of LUT completion per second per neuron is $30/86 * 7.75$ or about **3 neurons per hardware LUT completion per second.**

Question: Can one FPGA with 6.8 billion transistors, 10,000 pathways between transistors, and 24 2-LUTs map 30 billion neurons?

Answer: Maybe. $24 \text{ 2-LUTs} * 6.8 \text{ billion transistors} = 1.6 \text{ trillion indices}$ ($1.6 * 10^{11}$), and $(10^4 \text{ paths})^{24 \text{ LUTs}} = 10^{96} \text{ paths}$

Published Other factors

The power consumed by the human brain is about 30W; hence significant bodily functions support brain operation. (Thomas Edison took 10-minutes naps all the time.)

Synapse voltage is on the order of millivolts.

Synapse chemistry is at the molecular level, and not necessarily fast.

Neurons are supposed to lose 1 bit per second; hence neurons are not necessarily fault tolerant. (Why did I open the refrigerator?)

The human brain operates much slower than computers.

Appendix

Published Source code for software test

```
! Copyright 2012, Colin James III All rights reserved.
! Circuit result in seconds for 100,000,000 ( 10 ^ 8):
!   330.535 seconds
! LUT results in seconds for 1,000,000,000 ( 10 ^ 9):
!   ( 1, 1) = 89.045, 90.106; ( 2, 4) = 88.924; ( 4, 1) = 90.028; ( 4, 4) = 90.293, 89.638
LIBRARY "HEXLIB.TRC"
DECLARE DEF And, Or
OPTION BASE 1
DIM lut( 4, 4)
MAT lut = 3
LET start = time
PRINT start;
! FOR i = 100000000 TO 1 STEP -1 ! 10 ^ 8, 100 million; circuit test
FOR i = 1000000000 TO 1 STEP -1 ! 10 ^ 9, 1 billion, LUT test
    ! LET z = Or( And( 10, 9), 9) ! This produces test results for the circuit.
    LET z = lut( 1, 1) ! This produces test results for the LUT
NEXT I
LET end = time
PRINT end; "elapsed time"; end – start
GET KEY keep_output_screen_displayed
END
```