

---

## Computational equivalence between CogniMem and a DSP

Pattern recognition is based on the comparison of an incoming vector with other reference vectors. When neurons are trained to recognize complex patterns, or simple patterns in complex environments, the number of reference vectors can quickly reach several thousands or even millions. This short technical note compares the speed performances of a DSP versus the CogniMem parallel neural network to calculate the distances between one vector and one thousand other vectors.

### A CHAIN OF COGNIMEM CHIPS VERSUS A FARM OF DSPS

Traditional computers have known limitations for pattern recognition. With respect to speed, the execution of the instructions is sequential and, therefore, the search time increases with the number of references to compare. Also the memory access through a single bus is a bottleneck. The new processors featuring dual and quad core CPUs are an improvement, but at the expense of simple data access and synchronization protocols. With respect to cost, a DSP is not self sufficient and requires access to a RAM memory and a non-volatile memory for the storage. Finally, with respect to consumption, the DSP must run at very high clock frequency (100 Mhz to 4 GHz) leading to a very high power consumption, which prevents high miniaturization.

CogniMem has a very simple architecture: it is a chain of identical neurons operating in parallel. A neuron is an associative memory which can autonomously compare an incoming pattern with its reference pattern. During the formulation of the global response, all the neurons communicate briefly with one another (for 16 clock cycles) to define which neuron(s) has the smallest distance or closest match.

### COMPUTATIONAL COMPARISON FOR THE L1 DISTANCE

Assuming that V is an incoming vector and N is a neuron with a learned vector in memory, the following sequence of instructions calculates the L1 distance between V and all the neurons N:

```
0: for( n=0; n< Neuron-Number ; n++; MinDist = 0xFFFF)
   {
1:   for( i=0 ; i< Pattern-Length ;i++; L1Dist[n]=0)
2:     { if ( V[i] < N[n][i] ) then L1Dist[n] += N[n][i] - V[i] ; else L1Dist[n] += V[i] - N[n][i] ; }
3:   if (L1Dist[n] < AIF[n]) fire[n] = 1; else fire[n] = 0;
4:   if (L1Dist[n] < MinDist) MinDist = L1Dist[n] ;
5:   }
```

Using the CogniMem chip, all the neurons execute the commands in parallel. This means that the loop in line 0 does not exist.

line 2 for i<Pattern-Length-1	= Write Component command (1 clock cycles)
line 2 for i = Pattern-Length-1 + line 3	= Write Last Component command (2 clock cycles)
line 4	= Read Distance command (17 clock cycles)

The total number of clock cycles to find the smallest the L1 distance between a vector V and 1024 neurons is 275 on the CogniMem chip, equivalent to 10.18 microseconds at 27 Mhz.

As a courtesy, Prof. Pierre Raymond from the Institut Franco-Allemand de Recherches de St Louis (ISL) has developed a program in Assembler optimized for a DSP TIGER SHARK 21160 from Analog Device and executing the L1 distance between 1 vector and N reference vectors. The DSP reaches a number of clock cycles equal to 284,706, or 0.95 milliseconds at 300 Mhz.

The comparative results are presented in the table below:

<i>N=1024</i> <i>Vector length =256 bytes</i>	<b>CogniMem</b>	<b>Tiger DSP SHARK</b>
Clock frequency	27 Mhz	300 Mhz
Clock cycle (ns)	37	3.33
Number of instructions	275	N*278+34
Total cycles	275	284,706
Total time (usec)	10.18	949.02
Ratio		93

CogniMem is 93 times faster than the DSP Tiger SHARK.

### A CONSTANT RECOGNITION TIME

A key feature of CogniMem is that its recognition time remains constant and equal to 10.18 usec. If we increase the number of neurons in the above table to N=2048 (i.e. the capacity of 2 CogniMem chips), the recognition time for the DSP rises to 1.897 milliseconds, or 187 times slower than CogniMem.

### GIGAOPS EQUIVALENCE

With a recognition cycle of 10.18 usec at 27Mhz, the CogniMem chip can deliver 98,231 recognitions per second. The same speed performance on a DSP would require the execution of 278 instructions \* 98,231 times per seconds or 27.3 giga instructions per second.

### CONCLUSION

In 1988, the Darpa published a “Neural Network Study” and following are two conclusive extracts from the book:

“As the number of neurons and interconnects increases with regards to the size of the application, the amount of memory required to store the interconnect values increases. If that memory cannot be stored locally with every processor, then the processor must access memory external to itself – and that slows the overall speed of the simulator”

“If the development of neural network capability in signal processing, speech and vision applications become a focal point of DoD interest, then simulation facilities beyond hardware accelerators are necessary for researchers”

Twenty years later, CogniMem addresses the above concerns and it is not a software nor a simulation, but a real and affordable chip. It is easy to integrate into data mining systems in order to achieve capacities of thousands or millions of interconnects. It is easy to integrate into desktop appliances like a USB key so that researchers and consumers can carry one in their pocket.