The HARDWARE RESULTS of a 64x64 ANALOG INPUT ARRAY for a 3-DIMENSIONAL NEURAL NETWORK PROCESSOR

T.A. Duong, T. Thomas, T. Daud, A. Thakoor, and S. Suddarth†

Center for Space Microelectronics Technology
Jet Propulsion Laboratory, California Institute of Technology
Pasadena, CA 91109
†Ballistic Missile Defense Organization, Washington, DC 20301

I. INTRODUCTION:

Algorithms for the solution of spatio-temporal recognition and classification problems are known to require intense image data computation. To develop a viable hardware solution for such a task, our work at JPL during the past several years has focused on the development of a 3-dimensional artificial neural network (3DANN). It consists of a set of low power analog chips packaged as a 3-dimensional "sugar cube" mated to an infrared (IR) imager to conform to highly parallel processing at very high speeds. For example, an inner-product processing (multiply-accumulate) of a 64x64 image segment with 64 stored templates can be achieved with an unprecedented speed of 10^{12} operations per second (Tera-ops) consuming no more than 2 to 2.5 watts of power [1,2,3].

To obviate the restrictions of a single IR sensor providing data to the cube, the challenge of delivering highly parallel image data input (64x64 bytes of data in parallel every 250 nanoseconds) has been met by the design of an innovative chip architecture (termed the Column Loading Input Chip or CLIC). The CLIC replaces the imager on the cube and can be connected to an image frame grabber or a high-speed data formatter through the sugar cube motherboard. This scheme of 3DANN modification, termed 3DANN-M, has been developed and is pictured in Figure 1.

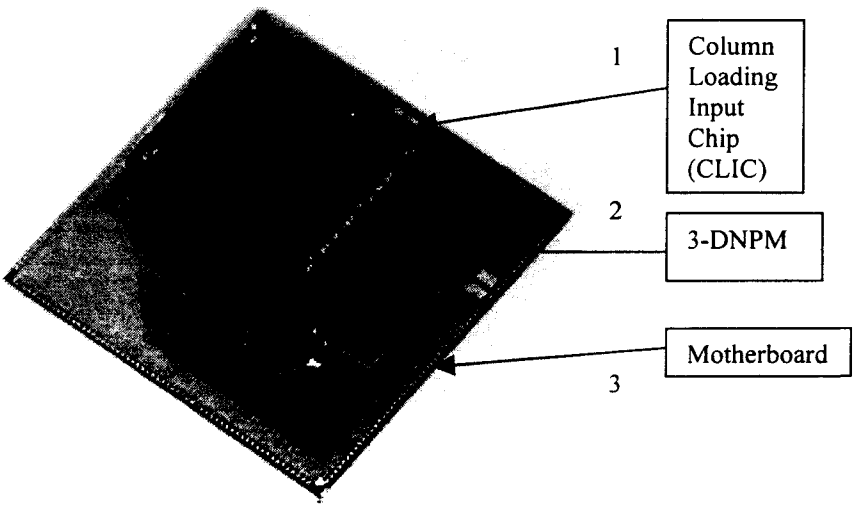


Figure 1: 3-DANN-M. This photo shows the CLIC on top of the 3-DNPM cube sitting on the motherboard.

As described in [4], the 3-DANN-M consists of a 64x64 analog input array called the Column Loading Input Chip (CLIC) and a 3-Dimensional Neural processing Module (3-DNPM). Referencing Figure 1:

1) The CLIC, mated on top of the cube, receives a 64x64-byte sub-image and converts it into a 64x64 array of parallel analog inputs for the 3-DNPM cube. Every 250ns, a new 64-byte row/column from the rastered image is imported into the CLIC forming a new input array to the 3-DNPM. Thus, an image of any size and type can be rastered by repeatedly moving the sub-image window one row or column at a time.
2) The 3-DNPM cube receives its input from the CLIC and performs sixty-four fully parallel inner products with sixty-four 64x64-byte weight templates.

   The 3-DNPM consists of sixty-four stacked chips forming a cube. Each chip contains sixty-four analog inputs, a 64x64 8-bit synaptic weight array, and sixty-four current summation outputs. Thus each chip is capable of performing a fully parallel multiplication between sixty-four analog inputs and sixty-four columns of the synaptic weight array (each column contains sixty-four 8-bit synaptic weights). In addition, each corresponding column across all sixty-four chips forms a template (e.g. the first columns of the sixty-four chips form template # 1, etc...). Furthermore, the sixty-four current summation outputs are connected with their counterparts from the other chips to provide sixty-four total current outputs from the 3-DNPM cube which are sent out via the motherboard for further processing.
3) The motherboard on the bottom serves as the I/O interface for the CLIC and the 3-DNPM cube to the real world.

Every 250ns, the system is able to perform sixty-four fully parallel inner products of a 64x64 byte sub-window of a searched image with sixty-four 64x64 byte preloaded templates. Finally, the sixty-four total current outputs representing the sixty-four inner products are available for validation or further processing.

In this paper, we will discuss the CLIC test results which enable us to build a whole system capable of using the 3-DNPM to solve spatio-temporal pattern recognition and classification problems in real-time. It may be noted that the NPM chip architecture has been reported earlier and hence is not repeated here.

The basic functionality of the CLIC is to receive a digital sub-image (64x64 bytes) and convert it into an analog voltage array. The sub-image is rastered in the original image one column/row at a time every 250 ns. During this time interval, the sub-image is transferred to the CLIC, is converted by the Digital-to-Analog Converter (DAC) array, and processed through the 3-DNPM to be obtained as sixty-four current outputs. The sub-image can be rastered down/up/right one row without losing the overhead time for reloading an entirely new sub-image. The CLIC interfaces to the 3-DNPM cube via a 64x64 local voltage output array which is available on 4,096 metal3 pads (pads are 66x66 μm^2). Each DAC cell, including memory and directional shift registers, is 101.6x101.6 μm^2 and the complete design is done in a 0.8μm HP CMOS process using a 12.1x12.3 mm^2 die size.

II. TECHNICAL APPROACH:

The CLIC architecture shown in Figure 2 contains:

1. A 64-bit digital input bus that is run with a 32-MHz clock to load sixty-four bytes within 250 ns.
2. Three independent controls and non-overlapped select lines that enable a column to shift up or down, or a row to shift right.
3. Three sets of 64-byte shift registers (shift-up, shift-down, and shift-right) which are formed with eight 8-byte parallel-in/8-byte parallel-out registers so that sixty-four bytes can be simultaneously shifted in one of three directions.
4. MDACs arranged in a 64x64 array. Each MDAC has an 8-bit parallel-in/parallel-out SRAM-SHIFT and 8-bit fully parallel DAC. The 8-bit SRAM-SHIFT can hold 8-bits of information like a normal SRAM, but it also is able to transfer 8-bits as an 8-bit parallel in/parallel out shift register (see the design section). The DAC receives the 8-bit complementary data in parallel and converts it into an analog voltage using a reference of 3.5V (e.g. the data with all bits ON will set all bits OFF in the DAC in which case the voltage output remains at 3.5 volts, etc...).

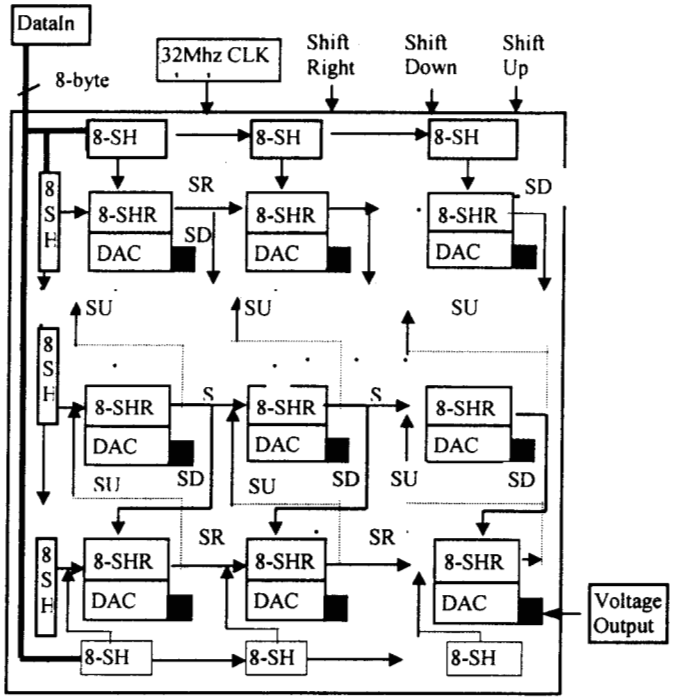


Figure 2: Architecture of the CLIC input chip. The chip contains 64x64 MDACs that can shift a whole array right or down/up. The right most column or the bottom row will be shifted out when shift-right or shift-down is selected, respectively. The SRAM holds an 8-bit byte for each pixel in the sub-image. The solid squares are the analog voltage outputs.

a) SRAM-SHIFT design:

The SRAM-SHIFT (see Figure 3) is required to hold 8 bits in an SRAM and be able to shift-in and shift-out in a fully parallel fashion. The area constraint for an individual cell (101.6x101.6 μm^2) prohibited the use of a standard latch in this design.

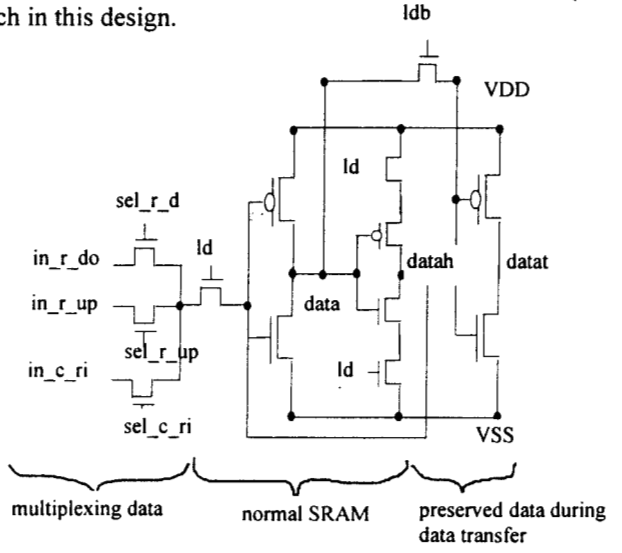


Figure 3: The SRAM-SHIFT cell. The SRAM-SHIFT contains three blocks: data multiplexor, SRAM, and preserved data for transferring without any distortion. The combination of these three blocks provides us with a robust design to hold and shift data within the given space constraints of the 3-DANN-M design.

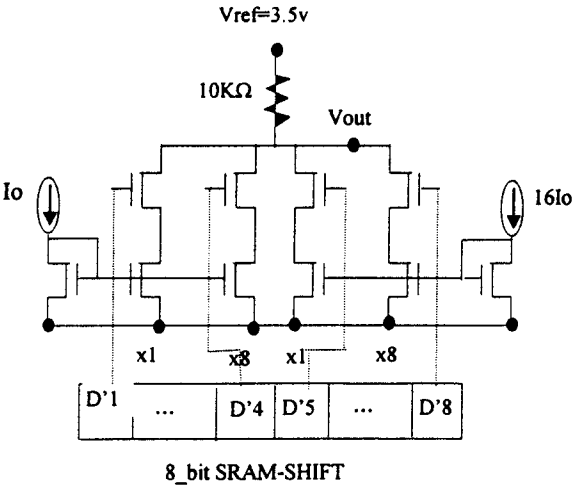
b) DAC converter:

The data is stored in the SRAM-SHIFT cell. The complementary data is used to convert to an analog voltage (see Figure 5). The conversion is performed as follows:

$$Vout = 3.5 - 10000\{\sum_{i=1}^{4} I_0(1-D_i)2^{i-1} + \sum_{i=5}^{8} 16I_0(1-Di)2^{i-1}\} \quad (1)$$

The global currents I_0 and $16I_0$ are injected from outside of the chip so that matching between them is well controlled. A 10KΩ N well resistor is provided for each DAC (see Figure 4).

When all the data bits are ON, the second term on the right side of equation (1) is zero and the voltage output is 3.5 V. When all the data bits are OFF, and 255 units of current source I_0 are drawn from 3.5 V, the voltage level is lowered to the desired value (nominal 2.5 V). Because the conversion is done locally, the parasitic capacitance is small and the speed is enhanced.



* D'i denotes the complementary output of bit i from SRAM-SHIFT.

Figure 4: The digital to analog converter circuit diagram. In this figure, the complementary digital output (D'i) is first converted into current, and then converted to an analog voltage through a 10KΩ resistor. E.g., when the digital value is zero, all bits in the DAC are "ON", and the output voltage is pulled down to 2.5 V.

III. SIMULATION AND TEST RESULTS:

There are two critical aspects of the 8-bit DACs used in the CLIC array: settling time and linearity. In this section we provide both simulation and hardware test results for each criterion.

a) DAC Settling Time:

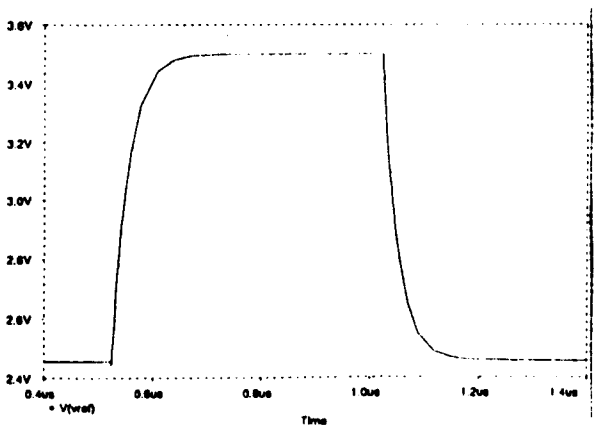
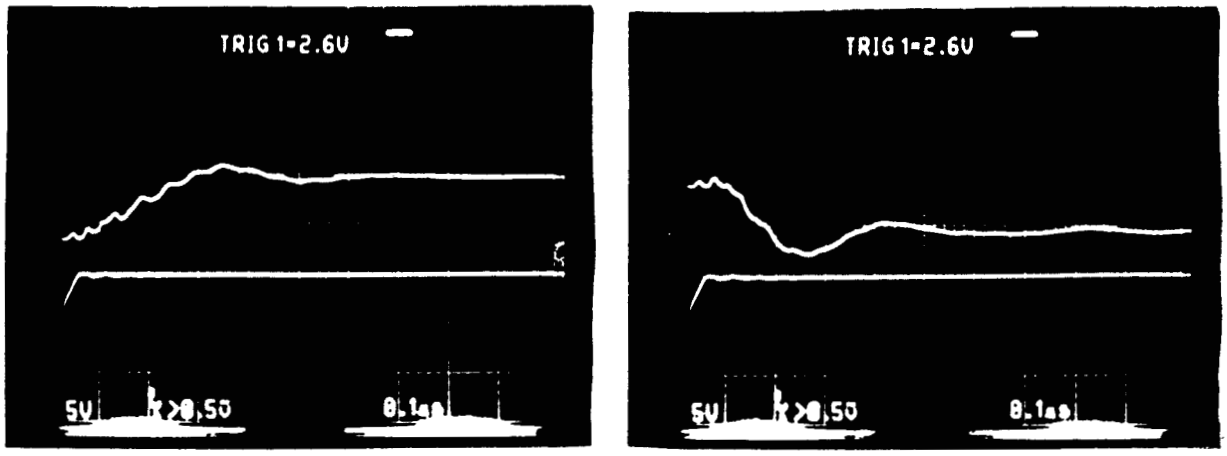


Figure 5: The simulation result of the DAC conversion time. About 150 ns are required to convert the digital value to the output analog voltage with 8-bit precision.



(a) (b)

Figure 6: The hardware test results showing the conversion time of the DAC. (a) Shows the rise time to be about 300ns, and (b) shows the fall time to be around 350ns for a full-scale digital transition.

In figure 5, the simulation shows that the settling time for a DAC is 150 ns after the data is shifted to the SRAM-SHIFT. The photographs in Figure 6, however, show that the actual rise and fall times are 300ns and 350ns respectively. The discrepancy is due in part to the extra delay caused by capacitive loading from the oscilloscope (20pF), that in simulation introduced an extra 150ns of delay. In addition, our test setup may have introduced further delays as well as second order effects caused by inductive loads (as shown in Figure 6). Given the latter considerations, our test data seem to be in line with simulation results.

b) DAC Linearity:

Figure 7 shows the DC simulation results for the 255 possible 8-bit combinations:

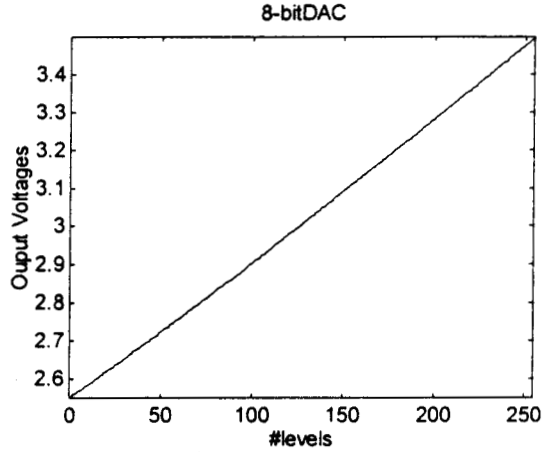


Figure 7: The simulation results of the 8-bit Digital to Analog Converter (8-bit DAC). The x-axis represents 8-bit digital values and the y-axis represents the output analog voltages.

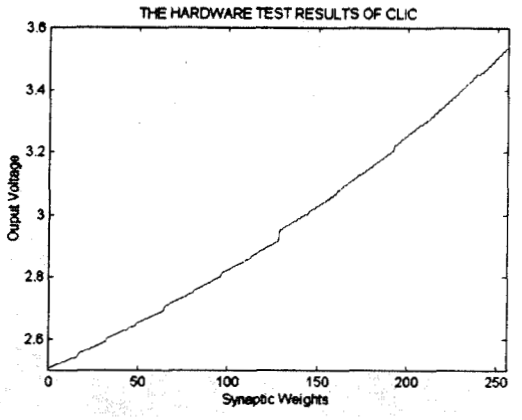


Figure 8: The test results of the 8-bit Digital to Analog Converter (8-bit DAC). The x-axis represents 8-bit digital values and the y-axis represents the output analog voltages.

As shown in Figures 7 and 8, the test and simulation results for DAC linearity are fairly close. The test results may not give us an 8-bit weight resolution as per the simulation, however in our application the hardware may be sufficiently accurate for the network to classify or recognize objects nonetheless.

IV. CONCLUSION:

In this paper it is demonstrated that the CLIC has the capability to override the bottleneck of providing a 64x64 parallel input array so that the 3-DNPM cube can be exploited at full speed. With the combination of the CLIC and the 3-DNPM cube, the massively parallel processing power of the cube architecture can be made available to solve spatio-temporal problems in real time.

Acknowledgments:

The research described herein was performed by the Center for Space Microelectronics Technology, Jet Propulsion Laboratory, California Institute of Technology and was jointly sponsored by the Ballistic Missile Defense Organization/Innovative Science and Technology Office (BMDO/IST) and the National Aeronautics and Space Administration (NASA). The authors would like to thank Dr. Suraphol Udomkesmalee for his support and Mr. Carlos Villalpando at JPL for technical assistance. We would also like to thank Mr. David Escobar of Irvine Sensors Corporation for providing us with the 3-DANN-M photograph.

References:

1. T.A.Duong, et al. "Analog 3-D Neuroprocessor for Fast Frame Focal Plane Image Processing," *Simulation Journal*, Vol. 65, No. 1, pp. 11-25, July 1995.
2. T.A.Duong, et al. "Low Power Analog Neurosynapse Chips for a 3-D 'Sugarcube' Neuroprocessor," *Proc. Of IEEE Int'l Conf. On Neural Networks (ICNN/WCNN)*, Vol. 3, pp. 1907-1911, June 28-July 2, 1994, Orlando, Florida.
3. T.A.Duong, et al. "Room and Low Temperature Performance of High Speed Neural Network Circuits," Accepted for Symposium on Low Temperature Electronics and High Temperature Superconductivity for the Electrochemical Society Meeting, May 2-4, 1997, Montreal, Canada.
4. T.A.Duong, et al. "64x64 Analog Input Array for 3-Dimensional Neural Network Processor," *Neural Networks and Their Applications Neurap'97* in Marseilles, France, pp. 49-53 March 12-13, 1997.