Implementation of Infomax ICA Algorithm with Analog CMOS Circuits

Ki-Seok Cho, Soo-Young Lee

Brain Science and Research Center
KAIST, Tae-Jon, Korea

ABSTRACT

Independent Component Analysis algorithm based on infomax theory with natural gradient was implemented with a fully-analog CMOS chip. Although one chip consists of 4 inputs and 4 outputs, the chip incorporates fully-modular architecture for multi-chip applications. The fabricated chip demonstrated improved SNRs for unknown speech mixtures.

1. INTRODUCTION

Recently, in neuromorphic system engineering, analog CMOS implementation plays a very important role. Many principles found in the brain information processing provide inspiration to solve very huge and complex problems. Integrated computation in massively-parallel analog CMOS circuits may be the best engineering approach to solve such as applications such as audio and video signal processing [5].

ICA using Herault-Jutten algorithm was realized by Cohen and Andreou. However, the convergence is sensitive to the mixing condition [7]. The improved infomax ICA algorithm incorporates natural gradient, which greatly improves the convergence. It is more practical and effective algorithm for a variety of real-world problems [6].

In this paper, we present a new approach to the analog CMOS implementation of ICA using infomax theory and natural gradient. It does not need an AD/DA converter or a digital memory. And Massively parallel processing can be done with fully connected analog modules.

2. Problem Statement and Assumptions

Measured signal **X** is a mixture of independent sources **S** with an unknown matrix **A**. The goal is to learn the un-mixing matrix **W**, which is the inverse of **A**. After training, **U=WX** becomes the estimate of the recovered sources.

The infomax algorithm is one way to find the un-mixing matrix **W** by means of maximizing the joint entropy **H(y)** of the outputs. Its learning rule is driven as

$$\Delta \mathbf{W} \propto \frac{\partial H(\mathbf{y})}{\partial \mathbf{W}} \mathbf{W}^T \mathbf{W} = \left[\mathbf{I} - \varphi(\mathbf{u})\mathbf{u}^{\mathbf{T}}\right]\mathbf{W} \qquad (2.1)$$

Here, we adopted the natural gradient algorithm developed by Amari [6].

3. 4 x 4 ICA Chip

A block diagram of the whole system is shown in Figure 1(a). It consists of 16 synapse-cells and 4 non-linear function modules.

3.1 Synapse Cell

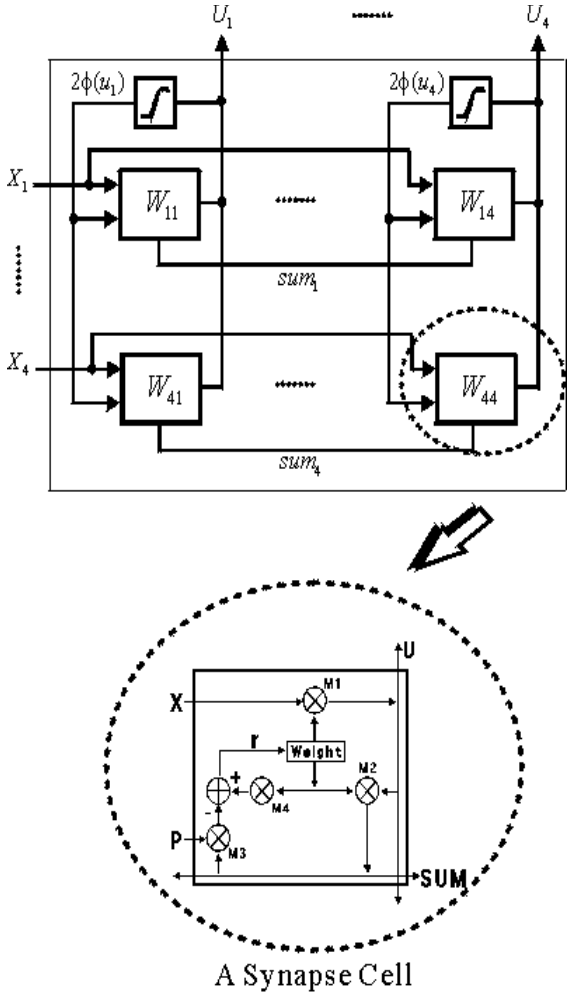

Figure 1: (a) 4 x 4 ICA Chip (b) a Synapse Cell

The rule (2.1) can be written as

$$\mathbf{U} = \mathbf{WX}$$
$$\mathbf{SUM} = \mathbf{U}^{\mathbf{T}}\mathbf{W} \qquad (3.1)$$
$$\Delta\mathbf{W} = \eta[\mathbf{W} - \Phi(\mathbf{U})\mathbf{SUM}]$$

In Figure 1(b), input **X** is multiplied by weight **W** and produces **U**. Each **U** is multiplied by **W** and produces **SUM**. The **SUM** is multiplied by **P** and subtracted from **W** to produce $\Delta\mathbf{W}$. The **P** means the output of a non-linear function circuit. The **r** is a learning rate controlled by weight update circuit. The summation is done by simple wired current summation.

3.2 Multiplier Circuit

As shown in Figure 2, we used the four-quadrant multiplier proposed by Zhenhua [3]. Since we can use high resistive poly-silicon layer for a load, no MOS resistor was used in the design.

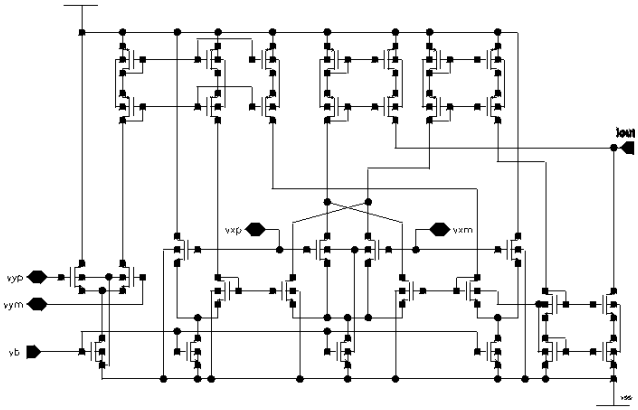

Figure 2: Schematic of 4-Quadrant Analog Multiplier

3.3 Non-linear Function Circuit

In the infomax algorithm, the non-linear function is related to the source distribution. We assume that sources have super-Gaussian distribution. Figure 3 shows the non-linear circuit, of which transfer function is an approximation of the hyperbolic tangent function.

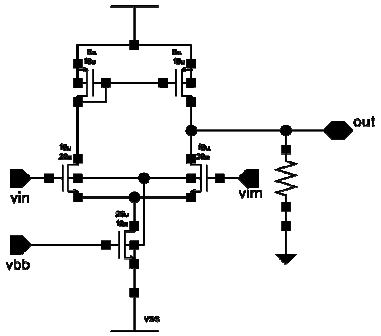

Figure 3: Non-linear Circuit

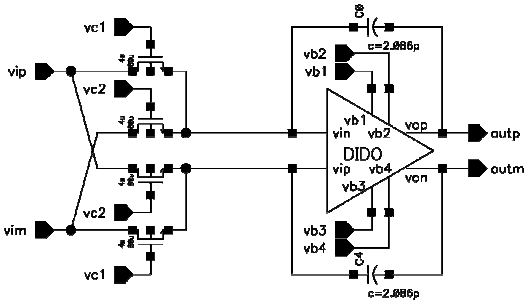

Figure 4: Weight Update Circuit

3.4 Weight Update Circuit

The weight update circuit in Figure 4 has a fully differential MOSFET-C integrator structure for common mode rejection. The operational amplifier uses a fully differential folded cascade configuration. The learning rate can be controlled by the gates of differential NMOS switch [2]. The output of the circuit is

$$V_{out} \equiv outp - outm \approx \frac{\beta(V_{c1} - V_{c2})}{sC}(vip - vim) \qquad (3.2)$$

3.5 Modular Architecture

Due to its modular architecture and wired current summation, as shown in Figure 5, the fabricated chips may be integrated into multi-chip systems for larger number of input and output channels.

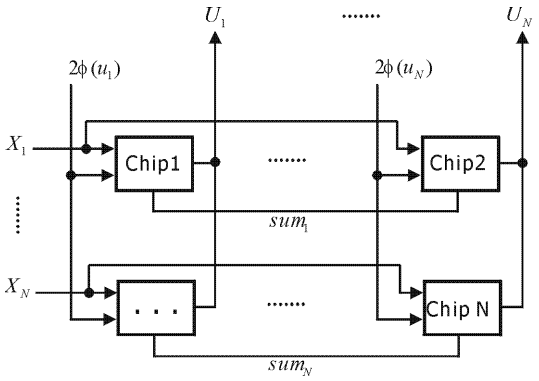

Figure 5: The Modular Structure of the ICA Chip

4. Experimental Results

The proposed ICA chip was fabricated using a 0.6μm, p-well, 2-poly 3-metal AMS CMOS process. Figure 6 shows the die photograph of the chip. The Active Chip Area is 2.8mm x 2.8mm.

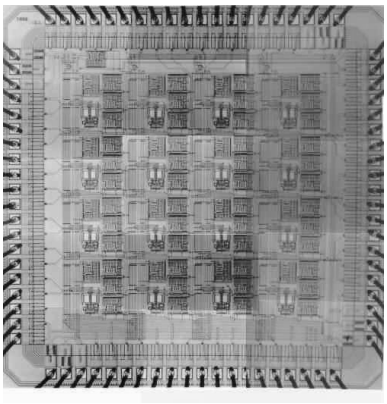

Figure 6: Fabricated ICA Chip

4.1 Multiplier and Non-linear Function Circuit Measurement

The output voltage swings between –2.5 and +2.5 V, while the input range is ±2.5 V. Figure 7 and Figure 8 show the measured characteristics of the multiplier and non-linear function circuits, respectively.

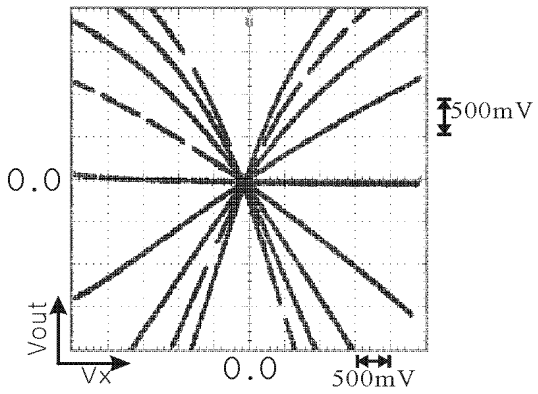

Figure 7: Result of Multiplier

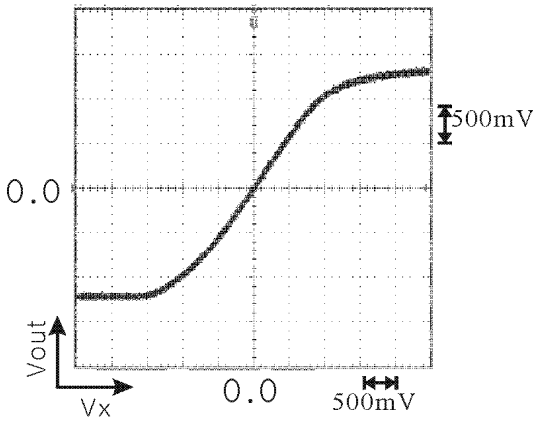

Figure 8: Result of Non-linear Circuit

4.2 Offset Compensation

As Figure 7 shows, the multipliers have input offsets, which may harm the chip performance. Table 1 shows the offset voltages measured by a typical fabricated chip.

Table 1: Offset Voltage for A Chip

NODE	OFFSET (mV)	NODE	OFFSET (mV)
U_1	235	SUM_1	134
U_2	330	SUM_2	171
U_3	77	SUM_3	33
U_4	-310	SUM_4	123

As shown in Figure 7, the multiplier has much less offset on the weight **W**. Therefore, one can approximate as

$$u_i \approx \sum_j W_{ij}(x_j + \Delta x_j) = \sum_j W_{ij} x_j + \sum_j W_{ij} \Delta x_j \quad (4.1)$$

where the second term represents the overall offset. If one reserves the last input channel for compensation, the offset term may be cancelled.

4.3 2 x 2 Network Experimental Results for Speech Signal

To test our chip, we used only 4 synapses with two men's speech signal recorded in 16kHz sampling rate. The other synapses and input channels was used for offset compensation. The used unknown mixing matrix is

$$\mathbf{A} = \begin{bmatrix} 0.8 & 0.6 \\ 0.7 & 0.8 \end{bmatrix} . \quad (4.5)$$

Figure 9 shows original sources, mixed input signals with mixing matrix $\mathbf{A}$, and separated outputs.

The average SNR after convergence is shown in Table 2. As you see in the Table, the SNRs become higher with the ICA chip.

Table 2: SNRs before and after the ICA (in dB)

	Input1	Input2	Output1	Output2
Source1	0.8	2.8	-3.63	**7.5**
Source2	3.9	1.5	**10.1**	-1.5

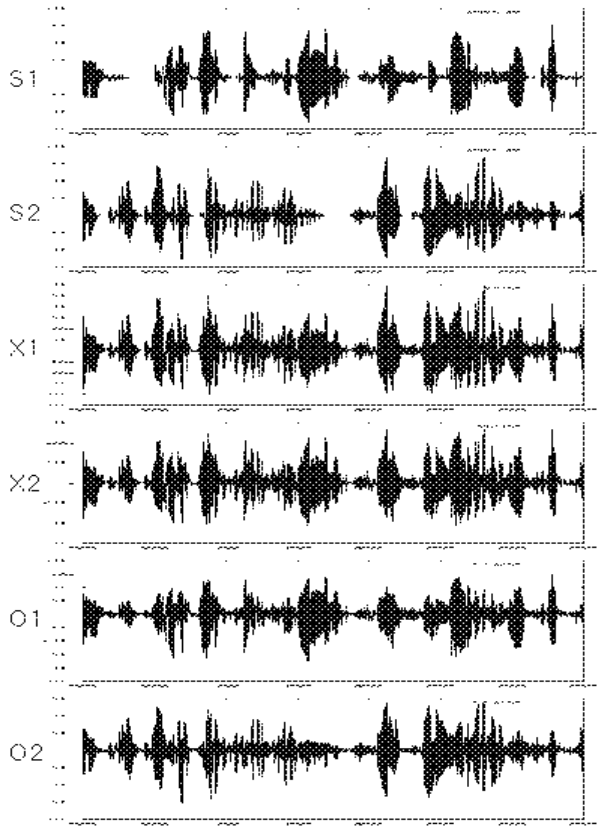

Figure 9: Sources, Mixed Inputs, and separated outputs

5. Discussion and Conclusion

We designed 4 x 4 ICA chips based on Maximize Entropy algorithm. Although each multiplier has input offsets, the big portion of the offset may be compensated with unused synapse weights. By this method we can get a good speech enhancement.

The experiments are still in its early stage. More experimental results will be reported at the conference.

6. Acknowledgements

This research was supported by Korean Ministry of Science and Technology.

7. References

[1] Phillip E. Allen & Douglas R. Holberg (1987), *CMOS Analog Circuit Design*, pp. 494-497. Holt, Rinehart and Winston.

[2] S. H. Lee, B. S Kim, M. K. Song & J. H. Choi (1999), *CMOS Analog/Mixed Mode IC design*, pp. 19-20. Sigma Press (IDEC)

[3] Zhenhua Wang. (1991), A CMOS Four-Quadrant Analog Multiplier with Single-Ended Voltage Output and Improved Temperature Performance, *IEEE Journal of Solid-State Circuits* **26** (9):1293-1301.

[4] Seung-Jin Choi, Andrzej Cichocki & Shun-Ichi Amari (2000), Flexible Independent Component Analysis, *Journal of VLSI Signal Processing – Systems for Signal, Image, and Video Technology,X*, Boston:Kluwer Academic Publishers.

[5] Tor Sverre Lande (1998), *Neuromorphic Systems Engineering-Neural Networks in Silicon*, Preface, Kluwer Academic Publishers.

[6] Te-Won Lee (1998), *Independent Component Analysis-Theory and Application*, pp. 5-64 , Kluwer Academic Publishers.

[7] Marc H. Cohen & Andreas G. Andreou. (1992), Current-Mode Subthreshold MOS Implementation of the Herault-Jutten Autoadaptive Network, *IEEE Journal of Solid-State Circuits* **26** (5):714-727.