IFERP

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 3, Issue 8, August 2016

# VHDL Simulation of Compressive Sensing Reconstruction

[1]Nimisha P K, [2]Saritha E
[1] P.G.Student ,GCEK,[2] Assistant Professor,ECE Department,GCEK
[1]nimishapk31@gmail.com, [2] sarithae@gcek.ac.in

***Abstract***— **Compressive Sensing (CS) signal reconstruction can be implemented using convex relaxation, non-convex, or local optimization algorithms. Though the re- construction using convex optimization, such as the Iterative Hard Thresholding algorithm is more accurate than matching pursuit algorithms, most researchers focus on matching pursuit algorithms because they are less computationally complex. Orthogonal Matching Pursuit (OMP) is a greedy algorithm, which solves the problem by choosing the most significant variable to reduce the least square error. Simultaneous OMP is an extension of OMP algorithm which contain multiple measurement vector (MMV). In this paper, we present an architecture by using VHDL for the reconstruction of compressively sensed signal using the orthogonal matching pursuit (OMP) and simultaneous OMP.**

***Index Terms***—**Cholesky decomposition, Compressive sensing, Orthogonal matching pursuit, Simultaneous OMP**

## I. INTRODUCTION

Conventional approaches to sampling signals or images follow Shannons sampling theorem: the sampling rate must be at least twice the maximum frequency present in the signal (the so called Nyquist rate). In fact, this principle underlies nearly all signal acquisition protocols used in consumer audio and visual electronics, medical imaging devices, radio receivers, and so on. For some signals, such as images that are not naturally band limited, the sampling rate is dictated not by the Shannon theorem but by the desired temporal or spatial resolution. However, it is common in such systems to use an antialiasing low-pass filter to band limit the signal before sampling, and so the Shannon theorem plays an implicit role. In the field of data conversion, for example, standard analog to digital converter (ADC) technology implements the usual quantized Shannon representation; the signal is uniformly sampled at or above the Nyquist rate.

Compressive sensing technique is a data sampling and compression approach, used by acquiring a compressed signal representation of length M (M<N) for a signal of length N. It is a powerful technique to represent signals at a sub- Nyquist sampling rate, provided the signal is known to be sparse in some domain. It retains the capacity of perfect (or near perfect) reconstruction of the signal from fewer samples than provided by Nyquist rate sampling. CS technique has attracted considerable attention from across a wide array of fields like applied mathematics, statistics and engineering including signal processing areas like MR imaging, speech processing, analog to digital conversion etc. Sparse Model for signals, which represents signals in sparse form in some representation basis. All natural images have sparse representation in certain basis like Wavelets, DCT, Curvelets etc.

Compressive sensing is generally performed by multiplying the original signal with a measurement matrix $\mathbf{\Phi} \in \mathbf{R}^{\mathbf{KXN}}$.

$$\mathbf{y} = \mathbf{\Phi x} \qquad (1)$$

To reconstruct the original signal from compressively sampled signal $\mathbf{y} \in \mathbf{R}^{\mathbf{K}}$ requires the knowledge of the measurement matrix $\Phi$.The reconstruction process lie in finding most effective solution to an underdetermined system of linear equation $\mathbf{y} = \mathbf{\Phi x}$, where measurement matrix $\mathbf{\Phi}$ and measured signal $\mathbf{y}$ are known.

## II. CURRENT STATE OF ART

Various algorithms have been proposed for the reconstruction of signals from the compressively sensed samples. Matching pursuit (MP) [1] is a common approach for sparse signal reconstruction, which greedily computes an approximation to the original signal. MP algorithm iteratively identifies the column of measurement matrix that is most correlated to a current signal estimate, followed by a simple update that computes an improved signal estimate. While each iteration of MP requires very low computational effort, the number of iterations heavily depends on the sparsity level 'm', and consequently, MP is more suitable for signals with high sparsity degrees[1]. Orthogonal Matching Pursuit (OMP) proposed in [2] is a more complex algorithm that incorporates a least- squares (LS) step to compute a signal estimate. In OMP, the LS step signicantly reduces the number of required iterations compared with MP, but it results in a high computational complexity per iteration [3]. This complexity is mainly due to the large number of inner-product computation (IPC), several comparison operations, and matrix inversion. Therefore, the high computational complexity of OMP algorithm is a major concern for its implementation to achieve real-time reconstruction of compressively sensed signals. Several software implementations on general purpose computer and graphic processor unit (GPU) have been proposed in the literature. It is observed that the acceleration achieved by the GPU-based implementation is signicantly better than CPU-based implementation [4]. However, the GPU-based implementation has a major problem of intermittent memory bandwidth between the main memory and the GPU [9], which does not facilitate regular flow of data communication with the host [4]. Several schemes have been proposed to accelerate individual computing stages of OMP algorithm in the hard- ware solutions presented in [6]. We found only a few proposed designs for the complete implementation of OMP algorithm in hardware. Recently, Septimus and Steinberg [4] and Stanislaus and Mohsenin [7] have presented a field programmable gate array (FPGA) implementation of OMP algorithm. However, a close examination of the algorithm and the proposed architectures reveals that those are from MP implementation rather than OMP, as stated by the authors. Septimus and Steinberg [4] have proposed an FPGA based design that involves signicantly higher cycle pe- riod due to a large size inner product (IP) in the critical path. Stanislaus and Mohsenin [7] have also proposed an FPGA implementation of MP algorithm based on QR-decomposition scheme for matrix inversion to reduce the computation complexity. Blach et al.[8] have presented FPGA implementations of reconstruction algorithms based on OMP. In the design presented in [8], the matrix inversion is based on CORDIC divider with high latency and a sequential execution of several parts of matrix multiplication.

## III. METHODS FOR COMPRESSIVE SENSING

### CONVEX RELAXATION

It replaces the l0 norm by l1 norm to reduce the problem to a convex problem. Three main directions under this category, namely the basis pursuit (BP),the basis pursuit denoising (BPDN) and the least absolute shrinkage and selection operator (LASSO). The BP problem can be solved by standard polynomial time algorithms of linear programming (LP) methods. The exact K-sparse signal re-construction by BP algorithm based on RIP. The BPDN and LASSO problems can be solved by efficient quadratic programming (QP) like primal dual interior method.

However, the regularization parameters λ and φ play a crucial role in the performance of these algorithms

## IV. GREEDY PURSUITS

This approach recovers the K-sparse signal by iteratively constructing the support set of the sparse signal (index of non-zero elements in the sparse vector). At each iteration, it updates its support set by appending the index of one or more columns (called atoms) of the matrix φ (often called dictionary) by some greedy principles based on the correlation between current residual of observation vector and the atoms. Few examples of greedy algorithms are Orthogonal Matching Pursuit (OMP) [6], Compressive Sampling Matching Pursuit (CoSaMP), Subspace Pursuit (SP), Iterative Hard Thresholding (IHT), Generalized Orthogonal Match- ing Pursuit (gOMP). MP is a class of iterative algorithm that decomposes a signal into linear expansion functions that form a dictionary. At each iteration it choose dictionary elements in a greedy fashion that best approximate the signal.OMP reconstructs the K-sparse signal in K steps by selecting one atom in each iteration. CoSaMP and SP select a indexed number of atoms (2K in CoSaMP and K in SP, for K-sparse signal) in each iteration while keeping the provision of rejecting a previously selected atom. IHT uses gradient descent followed by a hard thresholding that sets all but the K largest (in magnitude) elements in a vector to zero.

## V. PROPOSED SYSTEM

CS is based on the fact that the information from a signal may be captured by a small set of nonadaptive linear measurements when the signal is sparse in some basis [1].

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 3, Issue 8, August 2016

An m-sparse signal vector consists of at most m nonzero scalar components. A signal vector $x \in R^N$ acquired via linear measurements is given by

$$y = \phi x + n \qquad (2)$$

where $\phi \in R^{KXN}$ is a rectangular sampling matrix modeling the sampling system, $y \in R^K$ is the measurement vector, and n is a K-point vector that represents the measurement error or noise. The columns of matrix φ denoted $(\phi_1, \phi_2, \ldots\ldots \phi_n)$ are K-point vectors (K< N), also called atoms. The length of measurements vector y is in general assumed to be much smaller than the length of signal vector x.

### *A. OMP ALGORITHM*

It takes the measurement matrix Φ and the measured vector y as inputs and provides an estimate $\tilde{x}$ of the original signal x. This algorithm is iterative. During each iteration, it chooses one of the columns of Φ, which is most strongly correlated with the residual of measurements y, and then it removes the contribution of this column to compute a new residual. It also computes a new estimate of the original signal; after m iterations, the algorithm will generate the final estimate of the original signal.

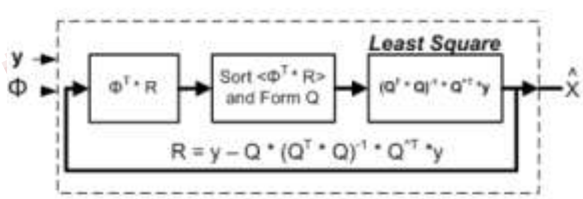


***Figure 1: Block diagram of OMP algorithm***

The OMP algorithm proposed in [7] is given in Algorithm1.

**Algorithm 1:OMP Reconstruction Algorithm**

**Inputs:** $\Phi \in R^{KXN}$ : The sampling matrix
$y \in R^K$ : The measurement vector
m : The sparsity level of the signal
**Output:** $\tilde{x} \in R^N$ : The estimate of the original signal
**Procedure**:

1. Initialize $r_0 = y$, index set $\Lambda_0 = \{\emptyset\}$, set iteration counter i=1.
2. Find index: $\lambda_i = \max|< r_{i-1}, \Phi_j >|$.
3. Update index set: $\Lambda_i = \Lambda_{i-1} \cup \{\lambda_i\}$.
4. Update ϕ: $\tilde{\Phi}_i = [\tilde{\Phi}_{i-i} \phi_{\lambda_i}]$.
5. Solve the Least Square Problem for new estimate $\tilde{x}$ for the original signal x:
$$\tilde{x}_i = \min||y - \tilde{\Phi}_i x||$$
6. Calculate new residual $r_i = y - \tilde{\Phi}_i \tilde{x}_i$
7. Increment the counter and return to step 2 if i < m.
8. Retrieve the final estimate $\tilde{x}$

The optimization problem of step 2 of Algorithm 1 is solved by calculating correlation vector w as follows:

$$w = \phi^T r_{i-1} \qquad (3)$$

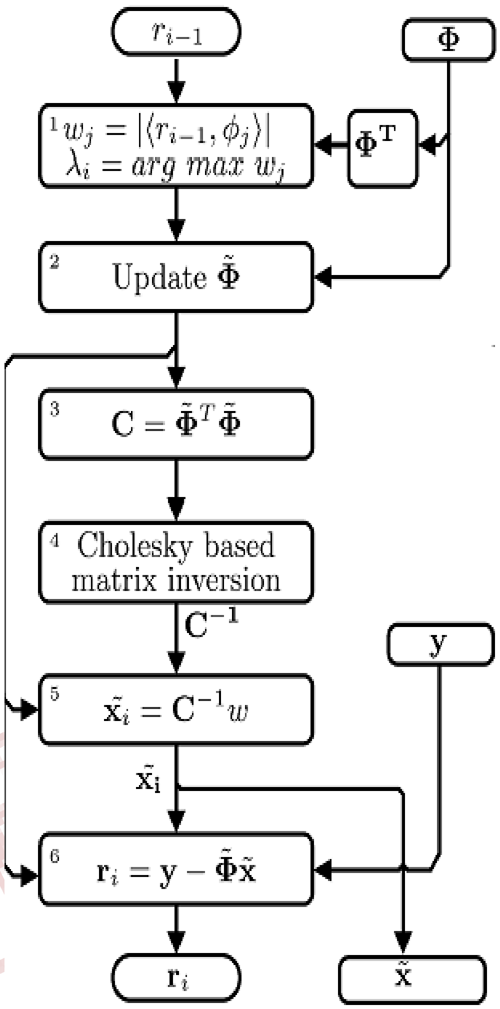


***Figure 2:Flow graph of one iteration of OMP algorithm***

where $r_{i-1}$ is the residual vector of $(i-1)^{th}$ iteration. The index $\lambda_i$ of the component of w having maximum absolute value is identified, and the corresponding column is extracted from ϕ to constitute matrix $\tilde{\phi}$ of such extracted columns. Ac- cording to step 4 of Algorithm 1, an estimate of the reconstructed signal $\tilde{x}_i$ is obtained by solving the following

$$y = \tilde{\phi}\tilde{x} \qquad (4)$$

Where $\tilde{\phi}$ a (K > m) rectangular matrix with K > m. The solution of (3) is obtained by solving the following:

$$w = \tilde{C}x \qquad (5)$$

Where $C = \tilde{\phi}\tilde{\phi}^T$ is a symmetric matrix $\in R^{m\times m}$. Equation (4) can be solved by matrix inversion or by forward / backward substitution .

Various methods can be used to find the inverse of a matrix, such as Cholesky factorization, LU, and QR decomposition methods. We have used the modified Cholesky factorization method for matrix inversion since it does not require square root operations. Based on the modified Cholesky factorization [10], matrix C can be expressed as the product of three matrices as

$$C = LDL^T \qquad (6)$$

The lower triangular matrix L and diagonal matrix D are computed using the following relations:

$$L_{i,j} = \frac{1}{D_{i,j}}\left\{C_{i,j} - \sum_{k=1}^{j-1}(L_{i,k}, L_{j,k}, D_{k,k})\right\} \qquad (7)$$

$$D_{i,i} = C_{i,j} - \sum_{k=1}^{j-1}\left(L_{i,k}{}^2 D_{k,k}\right) \qquad (8)$$

The inverse of matrix C is obtained as follows:

$$C^{-1} = (L^{-1})^T D^{-1} L^{-1} \qquad (9)$$

*The inversion of matrix D is obtained by taking inversion of its diagonal components, while the inversion of matrix L is performed iteratively using the relation* for i > j. In step 6 of Algorithm 1, the residue vector $'r'$ is updated for the next iteration using the relation

$$r_i = y - \tilde{\phi}\tilde{x} \qquad (10)$$

### *B. SIMULTANEOUS ORTHOGONAL MATCHING PURSUIT*

In Simultaneous OMP, the signal model is [2]:
1. K sparse signals $x_k \in R^N$ to be recovered (1≤ k≤ K).
2. A common linear measurement process described by the matrix $\phi \in R^{M\times N}$.

3. K measurement vectors $y_k \in R^M$ gathering the observations of each sparse signal when acquired through ϕ :

$$y_k = \phi x_k$$

To simplify the signal model, we introduce Equation (11) to summarize the K equations $y_k = \phi x_k$ into a single one:

$$Y = \phi X \qquad (11)$$

where Y = $(y_1 \ldots\ldots y_K) \in R^{M\times K}$ and X = $(x_1 \ldots\ldots x_K) \in R^{N\times K}$.

Using this formulation, the support of X, denoted by supp(X) is equal to the joint support $S := \cup_{k\in K}$ supp($x_k$). When a model involves one measurement vector, it is referred to as a single measurement vector (SMV) model while models incorporating K > 1 measurement vectors are multiple measurement vector (MMV) models. The columns of ϕ are often referred to as the atoms. This terminology being typically associated with dictionaries, it is worth emphasizing that the problem of recovering a s-sparse vector x on the basis of the measurement vector $y = \phi x$ is equivalent to finding s columns (or atoms) of the (dictionary) matrix ϕ that fully express y when using the proper linear combination.

The simultaneous orthogonal matching pursuit (SOMP) algorithm, which is described in Algorithm 2, is an extension of OMP to the MMV case and performs a joint support recovery.

Algorithm 2: SOMP Algorithm for Signal Recovery

Require: $Y \in R^{M\times K}, \phi \in R^{M\times N}, s \geq 1$

1. Initialization: $R^{(0)} \leftarrow Y$ $and$ $S_0 \leftarrow \emptyset$
2. $t \leftarrow 0$
3. While t < s do
4. Determine the atom of ϕ to be included in the support : $j_t \leftarrow argmax_{j\in[N]}(\|((R^t)^T)\phi_j\|)_1$
5. Update the support: $S_{t+1} \leftarrow S_t \cup j_t$
6. Projection of each measurement vector onto span $(\phi_{S_{t+1}})$ :
$$Y^{t+1} \leftarrow \phi_{S_{t+1}} {\phi^T}_{S_{t+1}} Y$$
7. Projection of each measurement vector onto span $\left(\phi_{S_{t+1}}\right)^T$:
$$R^{(t+1)} \leftarrow Y - Y^{(t+1)}$$
8. $t \leftarrow t+1$
9. end while
10. return Ss Support at last step

As shown in Algorithm 2, at each iteration t , SOMP adds to the estimated support the index $j_t$ of the atom $\phi_{j_t}$ maximizing the metric $\|(R_t)^T\phi_j\|_1 = \sum_{k=1}^{K} | < \phi_j, {r_k}^{(t)} > |$(steps4and5)where r(t) k denotes the $kth$column of the residual matrix R(t) . Each measurement vector is then projected onto the orthogonal complement of span $(\phi_{S_{t+1}})$, denoted by span $\left(\phi_{S_{t+1}}\right)^T$, during steps 6 and 7. The algorithm terminates when the prescribed number of iterations s has been reached. It is worth noticing that an atom cannot be picked twice as, once chosen, the projection onto span $\left(\phi_{S_{t+1}}\right)^T$ ensures that $< \phi, {r_k}^{t+1} > = 0$ if $\phi \in S_t$.

## V. RESULT AND DISCUSSIONS

In this system, simulation of OMP algorithm and SOMP algorithm for compressive sensing reconstruction is performed using MATLAB R2013a and Modelsim SE 6.1f.

### A. OMP Algorithm in MATLAB

OMP algorithm is first simulated using MATLAB R2013 a.

***Table 1:Signals and their size used in OMP algorithm***

| Signal | Size |
|---|---|
| Sparse signal - x | 1024 X 1 |
| Sampling Matrix- ϕ | 512 X1024 |
| Measurement Vector- y | 512 X 1 |
| Estimate of original signal | 1024 X 1 |
| Sparsity | 102 |

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 3, Issue 8, August 2016

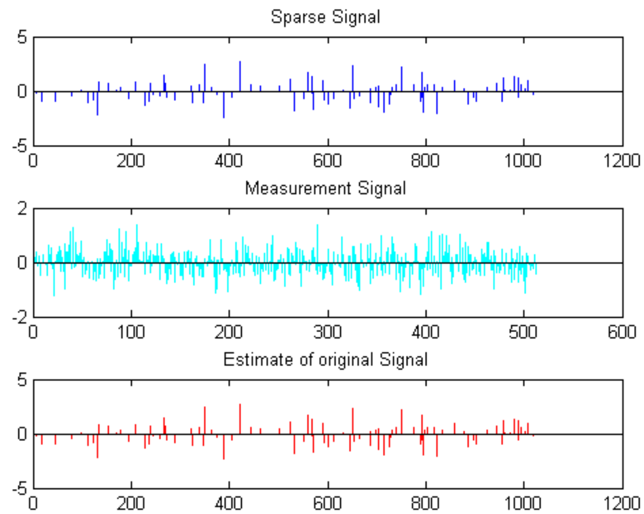


*Figure 3: Result of CS reconstruction using OMP*

### *B. Simulation result of OMP Algorithm*

OMP Algorithm is simulation is one using Modelsim. The MSE and PSNR is computed using Matlab.

*Figure 4Simulation resuly of OMP algorithm*

The mean square error is 1.9271e-08 and PSNR (in dB) is 84.7 dB.

### *C. SOMP Algorithm in MATLAB*

Three signals (speech signal) are simultaneously transmitted and recovered using SOMP algorithm. SOMP algorithm is an extension of OMP algorithm, that is, single measurement vector (SMV) to multiple measurement vector (MMV). Figure 5. shows the simulation result of SOMP algorithm with frame size 512.

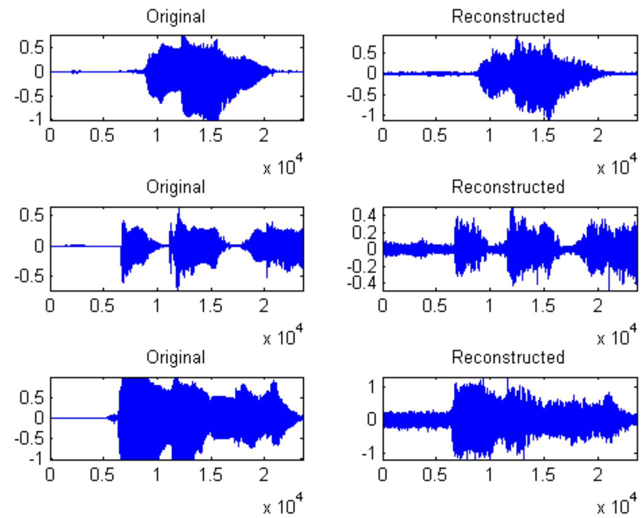


*Figure 5:Result of CS reconstruction using SOM*

*Table 2:PSNR and MSE for different frame rate of speech signal with SOMP algorithm*

| Frame size | MSE | PSNR (in dB) |
|---|---|---|
| 1024 | 1.2051e-07 | 7072 |
| 512 | 9.398e-08 | 72.56 |

### *D. SOMP Algorithm simulation results*

The simulation of SOMP algorithm is done in Modelsim SE. Two random signals (1024×2) are used.

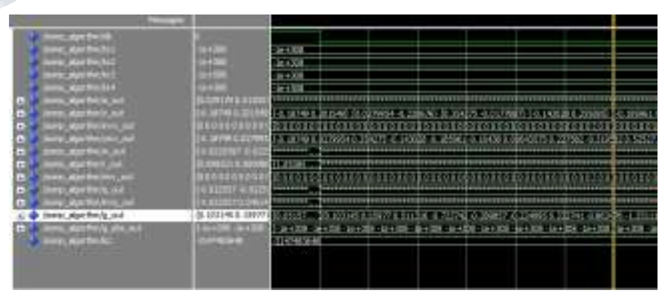

*Figure 6:Simulation result of SOMP algorithm*

The mean square error and PSNR is computed using Matlab.The mean square error of the signal is 4.6656e−05 and the PSNR (in dB) is 51.18dB.

## VI. APPLICATIONS

The fact that a compressible signal can be captured efficiently using a number of incoherent measurements that is proportional to its information level which has implications that are far reaching and concern a number of possible applications:

Data compression: In some situations, the sparse basis may be unknown at the encoder or impractical to implement for data compression. In random sensing, however, a randomly designed φ can be considered a universal encoding strategy, as it need not be designed with regards to the structure of sparse basis. This universality may be particularly helpful for distributed source coding in multi signal settings such as sensor networks.

Channel coding: CS principles (sparsity, randomness, and convex optimization) can be turned around and applied to design fast error correcting codes over the reals to protect from errors during transmission.

Data acquisition: In some important situations the full collection of n discrete time samples of an analog signal may be difficult to obtain (and possibly difficult to subsequently compress). Here, it could be helpful to design physical sampling devices that directly record discrete, low rate incoherent measurements of the incident analog signal.

## VII. CONCLUSION

There are different recovery algorithms for the reconstruction of sparse signal. This paper describes the reconstruction using OMP algorithm. In OMP, a single measurement vector (SMV) model signal is used. It is modeled using Matlab and simulation is done by using Modelsim. The quality of reconstructed data is evaluated by computing PSNR for different data precision. Then this algorithm is extended to multiple measurement vector (MMV) model signal, that is, simultaneous orthogonal matching pursuit (SOMP). This algorithm is analyzed and verified in Matlab by using speech signals. Somp algorithm simulation is modeled using Modelsim and computed the PSNR.

## REFERNCES

[1] Hassan Rabah, Abbes Amira, Basant Kumar Mohanty, Somaya Almaadeed and Pramod Kumar Meher, “FPGA Implementation of Orthogonal Matching Pursuit for Compressive Sensing Reconstruction,” IEEE Trans. VLSI systems, Sept.2014.

[2]Emmanuel J. Cands and Michael B. Wakin, “An Introduction To Compressive Sampling ”, IEEE SIGNAL PROCESSING MAGAZINE , March 2008.

[3] Jean Franois Determe, Jrme Louveaux, Laurent Jacques and Franois Horlin, “On The Exact Recovery Condition of Simultaneous Orthogonal Matching Pursuit ”, IEEE SIGNAL PROCESSING LETTERS, Jan 2016.

[4] Joel A. Tropp, Anna C. Gilbert, Martin J. Strauss, “Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit”, Elsevier Computer Science, Aug.2005.

[5] S.G. Mallat and Z. Zhang, “Matching pursuits with time frequency dictionaries,” IEEE Trans. Signal Process, vol. 41, pp. 3397–3415, Dec 1993.

[6] J. A. Tropp and A. C . Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit, IEEE Trans. Inf. Theory, vol. 53, pp. 4655 - 4666, Dec 2007.

[7] P. Maechler et al., “VLSI design of approximate message passing for signal restoration and compressive sensing,” IEEE J. Emerg. Sel.Topics Circuits Syst, vol. 2, no. 3, pp. 579 - 590, Sep. 2012

[8] A. Septimus and R. Steinberg, “Compressive sampling hardware reconstruction,” Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), pp.3316-3319 May/June 2010.

[9] T. R . Braun, “An evaluation of GPU acceleration for sparse reconstruction ,” Proc. SPIE, Signal Process., Sensor Fusion, Target Recognit. XIX, vol. 53, pp. 105–115, Apr 2010.

[10] A. Happonen, A. Burian, and E. Hemming, “A recongurable processing element implementation for matrix inversion using Cholesky decomposition, World Acad. Sci., Eng., Technol., vol. 3, no. 28, pp. 114-117, Mar. 2004.