Computers and Electrical Engineering 71 (2018) 173–190

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

High-speed FPGA implementation of orthogonal matching pursuit for compressive sensing signal reconstruction

Önder Polat*, Sema Koç Kayhan

Department of Electrical and Electronics Engineering, Gaziantep University, Gaziantep, Turkey

ARTICLE INFO

Keywords:
Compressive sensing
Field-programmable gate array (FPGA)
Orthogonal matching pursuit (OMP)
Sparse signal recovery

ABSTRACT

In this paper, a reconfigurable, high-speed, low-power, and vendor-independent field-programmable gate array (FPGA) implementation of the orthogonal matching pursuit (OMP) algorithm is presented. Unlike existing previous work, the proposed architecture is reconfigurable, that is, the architecture can accept different signal sizes, different sparsity levels, and sampling matrix sizes. Furthermore, the Goldschmidt algorithm is used to implement the fixed point division unit that achieves state-of-the-art performance. To the best of our knowledge, the presented architecture demonstrates the fastest reconstruction times for the standard OMP algorithm on FPGA for high sparsity levels. The proposed design is able to recover a 128-length signal with $K = 5$ in 7.75 μs and 256-length signals with $K = 8$ and $K = 12$ in 23.27 μs and 39.56 μs respectively. The results also show that our design outperforms many other existing FPGA implementations in terms of power consumption.

1. Introduction

Compressive Sensing (CS) [1,2] is a recently developed theory that makes it possible to reconstruct sparse signals from fewer number of samples than the Nyquist-Shannon sampling theorem demands. CS has many applications in real-world signal processing systems. Electrocardiography (ECG) monitor for ambulatory use [3], magnetic resonance imaging [4], and synthetic aperture radar imaging [5] are prominent examples. A CS application can basically be considered as a two-step process; compressively sensing the signal and reconstructing it. To recover a signal from fewer samples in a reasonable time, reconstruction algorithms require a lot of computational power and resources. The iterative nature of the CS algorithms makes them computationally expensive to implement and use on software platforms. Thus, to accelerate the reconstruction process, custom hardware implementations of the orthogonal matching pursuit (OMP) and other CS recovery algorithms are proposed by the research community.

Hardware implementations in literature aim to achieve optimum performance in timing and resource utilization for the reconstruction process. The proposed architectures vary in specifications, algorithms used, and the target device. The size of the measurement matrix and the sparsity of the signal are two important parameters which determine the performance of the circuit. Designs proposed in [6] and [7] implement the OMP algorithm and use Cholesky Decomposition (CD) for the Least Squares (LS) solution. Both designs use fixed point notation for efficiency and are implemented on Virtex-5 and Virtex-6 FPGAs respectively. The construction in [7] is implemented using High Level Synthesis (HLS) tools instead of manual HDL (Hardware Description Language) and it has better performance in terms of latency and signal quality in comparison to other OMP implementations. Designs proposed

Reviews processed and recommended for publication to the Editor-in-Chief by Area Editor Dr. E. Cabal-Yepez.
* Corresponding author.
E-mail addresses: onderp@gantep.edu.tr (Ö. Polat), skoc@gantep.edu.tr (S.K. Kayhan).

https://doi.org/10.1016/j.compeleceng.2018.07.017
Received 11 December 2017; Received in revised form 10 June 2018; Accepted 24 July 2018

Table 1
Hardware implementations of OMP.

Technology	Φ size	Sparsity	LS	Data Format
Virtex-5[6]	32 x 128	5	CD	32-bit FP
Virtex-5[12]	32 x 128	5	CD	16-bit FP
Virtex-5[9] [8]	64 x 256	8	QRD	24-bit FP
Virtex-5[9] [8]	32 x 128	8	QRD	24-bit FP
Virtex-6[10]	256 x 1024	36	QRD	25/18-bit FP
Virtex-5[10]	32 x 128	5	QRD	25/18-bit FP
Virtex-6[7]	256 x 1024	36	CD	18-bit FP
90nm CMOS[11]	64 x 256	12	QRD	8/18-bit FP

in [8–10] incorporate QR decomposition for the least squares solution. The structures in [8,9] are designed for 128 and 256-length input signals. The latter is the low complexity version of the first, and it sacrifices signal quality to reduce the latency of the circuit. In [10], another architecture is proposed for both the OMP and the Approximate Message Passing (AMP) algorithms. The OMP implementation supports sparsity of 36 and input length of 1024. The study in [11] is an ASIC implementation on 90nm CMOS and uses QR decomposition with matrix-inversion-bypass (MIB) to reduce the computational complexity. The implemented algorithm is the projection-based atom selection orthogonal matching pursuit (POMP) algorithm, a variant of the OMP algorithm. Quality of the reconstructed signal is higher compared to the OMP implementations. Notable architectures that implement OMP are listed in Table 1 along with their specifications.

Most implementations employ the OMP algorithm due to its high performance for high sparsity signals, and its flexible nature indicated by the available extensions. Most of the existing implementations are designed to support a fixed size sampling matrix. The size of the sampling matrix and the input signal determines the number of multipliers in the inner product unit, the number of utilized RAM blocks, the number of adders, layers in the adder tree, and many more structural parameters. In this paper, a scalable hardware implementation of OMP is proposed for CS signal reconstruction. The proposed architecture is designed using Verilog HDL and synthesized for Xilinx Virtex-5-6-7 FPGAs, and Altera Stratix and Cyclone FPGAs. The architecture is fully parametrized and reconfigurable depending on the size of measurement matrix and the signal sparsity. The size of the measurement matrix and sparsity of the input signal are provided as parameters in the HDL design for flexible implementation. Matrix inversion makes use of the Goldschmidt algorithm for the reciprocal operation, which improves the critical path of the proposed design as discussed in Section 3. Compared to the existing hardware implementations of the standard OMP algorithm, the results indicate that the proposed architecture demonstrates faster reconstruction times for high sparsity signals.

The rest of this paper is organized as follows: Section 2 provides a description of the OMP algorithm. Section 3 describes the proposed hardware architecture for the OMP algorithm in detail. Experimental results of the proposed hardware, and a comparison to the existing works are given in Section 4. Finally, the paper is concluded in Section 5.

2. Compressive sensing and orthogonal matching pursuit algorithm

CS theory dictates that a sparse signal in time domain that is sampled by using a random Gaussian matrix can be reconstructed by finding the sparsest solution. Considering an $M \times N$ dimensional random matrix $\mathbf{\Phi}$, an N dimensional signal $\mathbf{x}$ is sampled (or sensed) by forming the measurement vector $\mathbf{y}$ such as;

$$\mathbf{y} = \mathbf{\Phi x} \tag{1}$$

$\mathbf{y}$ is an M dimensional vector such that $M \ll N$. After the sensing part, signal $\mathbf{x}$ is reconstructed by solving (1) for the optimum sparsest solution.

OMP is an iterative greedy algorithm [13], in each iteration; the algorithm selects a column (atom) of which is most correlated with the current residual. Once the atom is selected, the signal is orthogonally projected to the span of the selected atoms, a new residue used for the next iteration is computed. This process is repeated for K times to find K columns of $\mathbf{\Phi}$ and the estimated signal $\tilde{\mathbf{x}}$ is computed by solving an overdetermined least square problem. The OMP algorithm is shown in Algorithm 1.

3. Proposed architecture

The proposed structure for the implementation of the OMP algorithm, shown in Fig. 1, mainly consists of three stages: (1) Optimization Unit which finds the strongly correlated vectors, (2) LS solution to compute the new estimate of the signal and residual update, and (3) Control and Data Acquisition. The proposed OMP architecture is given in the following sections.

3.1. Optimization

Optimization step involves inner product calculation between the columns of $\mathbf{\Phi}$ and the residual. Purpose is to find the index of the column vector that is most relevant with the residual vector. At the start of the algorithm, the residual is initialized to the measurement vector. After the first index is found, the respective column of $\mathbf{\Phi}$ is saved in a matrix of selected columns ($\tilde{\mathbf{\Phi}}$) and used in