

# FPGA Implementation of Efficient Beamformer for On-Board Processing in MEO Satellites

Rakesh Palisetty\*, Luis Manuel Garces Socarras\*, Haythem Chaker\*, Vibhum Singh\*, Geoffrey Eappen\*,  
Wallace Alves Martins\*, Vu Nguyen Ha\*, Juan A. Vásquez-Peralvo\*, Jorge Luis Gonzalez Rios\*,  
Juan Carlos Merlano Duncan\*, Symeon Chatzinotas\*, Björn Ottersten\*, Adem Coskun†, Stephen King†,  
Salvatore D’Addio†, and Piero Angeletti†

*\*Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg*

*†European Space Agency*

Emails: (rakesh.palisetty, luis.garces, haythem.chaker, vibhum.singh, geoffrey.eappen, vu-nguyen.ha, juan.vasquez,  
jorge.gonzalez, juan.duncan, wallace.alvesmartins, symeon.chatzinotas, bjorn.ottersten)@uni.lu,  
(adem.coskun, salvatore.daddio, piero.angeletti)@esa.int, stephen.king@ext.esa.int

**Abstract**—Medium Earth orbit (MEO) constellation is an appealing solution between geostationary equatorial orbit (GEO) and lower Earth orbit (LEO) in terms of latency and number of satellites required. On-board processing of digital beamformer in MEO satellites is an efficient solution for achieving wider bandwidth, increased flexibility, and lower latency. Power constraints, however, make it impractical to digitally create thousands of beams at once. In this paper, area-power efficient digital beamformer architectures are proposed considering key metrics of a typical MEO scenario. The proposed efficient digital beamformer is comprised of a sparse-matrix-based user selection, a 2D discrete Fourier transform (DFT)-based digital beam generation, which is implemented by a fast Fourier transform (FFT) algorithm, and a spatial windowing module for selecting the antenna pattern. Furthermore, architectures of digital beamformer using conventional 2D-FFT approach, fully unrolled 2D-FFT, and an area-power efficient twiddle factor (TF) quantized fully unrolled 2D-FFT are proposed. The spatial windowing architecture concerning  $10 \times 10$  radio frequency chains and sparse matrix architecture for user selection is also proposed. The proposed architectures are implemented targeting Virtex ultrascale FPGA and the area-power utilization is reported. It is noticed that more than 50%-reduction in area and power is achieved with the beamformer incorporating the proposed TF quantized fully unrolled 2D-FFT.

**Index Terms**—beamforming, fast Fourier transform (FFT), FPGA implementation, MEO satellites, power estimation.

## I. INTRODUCTION

Geostationary equatorial orbit (GEO) very high throughput satellite (VHTS) systems require significant capital investment for each satellite when the design requires very narrow beam footprints. Due to the relatively high round-trip times of around 500 ms between users and terrestrial gateways, GEO satellites suffer from higher latency compared to lower Earth orbit (LEO) satellites [1]. On the other hand, in order to provide operational coverage, LEO constellations require hundreds or even thousands of satellites [2]. LEO satellites also have a shorter “user dwell” period because of their high radial speed. Medium Earth orbit (MEO) constellations, which normally require between ten to twenty satellites to provide operational coverage, seem to be an appealing solution that

attains a reasonable trade-off between LEO constellations and GEO systems [3]. Additionally, MEO offers a latency of 180 ms (at an altitude of 8000 km) which is less than the GEO. In the space-segment, the use of digital beamforming technology to provide fully steerable beams when working along with active antennas to improve capacity, control, and flexibility has been a prior theoretical answer to these problems. Nonetheless, in order to make it easier to fit the payload on a MEO satellite platform, complexity considerations must be taken into account while choosing the beamformers.

Analog beamformers are limited in adaptability and have huge volume and power requirements [4]. Array fed reflector concept is practical to generate multiple beams that are fixed. The most favourable way to build a beamforming network for large VHTS direct radiating array antennas is through hybrid beamforming [5]. However, there is a complexity versus flexibility trade-off when hybrid beamforming is used [6]. Digital components used in digital beamformers now cost less and have more capabilities thanks to advances in digital hardware-based design. To implement a fully digital payload with increased flexibility of capacity allocation to accommodate a larger range of user applications, the power consumption must be reduced [7]. These demands can be fulfilled by a digital beamformer that is both area and power efficient without compromising the adaptability and signal integrity.

A brute-force approach to implement digital beamforming is matrix-by-vector multiplication, where each scalar multiplication is a hardware operation that consumes a lot of resources. When using uniformly linear or rectangular arrays, it is straightforward to build codebook-based digital beamforming using the columns of the discrete Fourier transform (DFT) matrix as the beamforming weight-vector (i.e., as the code-words of the codebook) [8]. This makes it possible to effectively implement the DFT using fast Fourier transform (FFT) algorithms [9].<sup>1</sup> Real-time beamformers can be implemented

<sup>1</sup>The DFT computation is implemented via a fast Fourier transform (FFT) algorithm. Thus, from now on we shall use the term FFT-based beamformer.

more efficiently with less circuitry and power than matrix-by-vector multiplication because FFT techniques can reduce the computational cost of the DFT computation to  $\mathcal{O}(N \log N)$ . Two-dimensional (2D) FFT digital beamforming implementation calls for  $N$  output samples every clock cycle [10]. To execute FFT-based digital beamforming, conventional FFT algorithms need  $N$  clock cycles for  $N$  samples, which is not efficient implementation for real-time beamformer. These  $N$  samples can be generated by a completely unrolled FFT per clock cycle [11]. The performance of fully unrolled FFT-based digital beamforming on satellite systems can be significantly enhanced in terms of power reduction, area reduction, and higher throughput, as shown in [11].

For this reason, an area-power efficient digital beamformer consisting of a sparse matrix for DFT selection, a 2D-FFT for digital beamforming, and a spatial windowing module for the number of RF chains of the antenna is proposed. These three modules constitute a real-time beamformer for the on-board processing in MEO satellites. The hardware implementation of the sparse matrix for DFT selection is the initial topic of discussion. Then, we look at the conventional serial FFT complexity and implementation, which rolls one input sample through each cycle of operation. The discussion of a fully unrolled FFT, which performs  $N$  input sample operations per clock cycle, follows. Fully unrolled FFT implementation results in high area and power usage. Therefore, a 4-bit twiddle factor (TF)-quantized FFT is proposed for 2D-FFT design in order to further reduce area and power usage in a fully unrolled FFT. Efficient hardware implementation techniques with respect to conventional FFT and fully unrolled FFT are presented in detail. Furthermore, the hardware implementation of spatial windowing module is described.

The rest of the paper is organized as follows. Section II presents the real-time system model of digital beamformer proposed in this paper that is comprised of a sparse matrix, 2D-FFT, and spatial windowing. Then, the implementation methodology of this real-time efficient beamformer is presented in Section III. Finally, a preliminary assessment of the simulations in MATLAB and field programmable gate arrays (FPGA) implemented area-power consumption of the beamformer for on-board digital beamforming in MEO scenarios is described in Section IV, followed by some concluding remarks in Section V.

## II. PROPOSED BEAMFORMER SYSTEM MODEL

This section introduces the overall digital beamformer architecture of the satellite payload with  $10 \times 10$  radiofrequency (RF) chains. The proposed architecture is illustrated in Fig. 1, which consists of three main components, namely sparse matrix, 2D-FFT, and spatial windowing.

### A. Sparse Beam domain Selection Matrix

In order to illuminate a target coverage area, a set of fixed spatially separated beams  $\mathcal{B}$  create  $B = |\mathcal{B}|$  overlapping cells defined as the spot beams' 3-dB gain contours. This spatial RF design assumption guarantees lowest complexity, for example

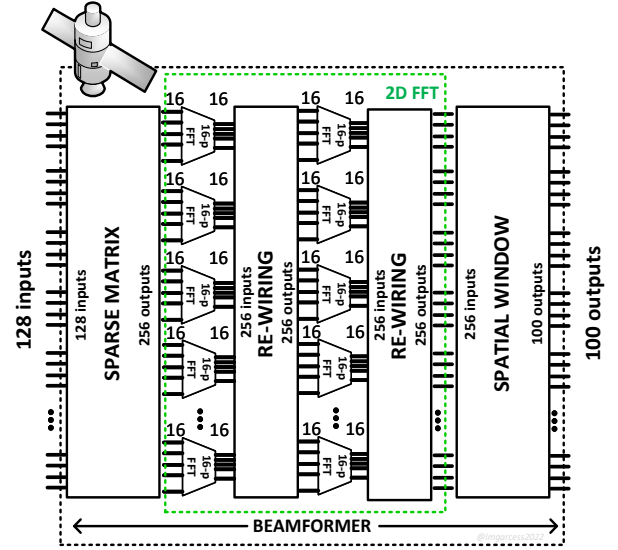


Fig. 1: Proposed architecture of digital beamformer with  $10 \times 10$  RF chains in MEO scenario.

using uniform rectangular array (URA). When all these spot beams are formed using a beamforming matrix  $\mathbf{W}_B$ , these cells are typically non-orthogonal. Orthogonality here refers to the property of the output beamforming matrix  $\mathbf{Y}$  being a unitary matrix, which implies that the beam domain covariance matrix  $\mathbf{X} \in \mathbb{C}^{B \times B}$  is diagonal matrix.

The noise in different orthogonal fixed beams  $\mathcal{B}_\perp$  is uncorrelated (the beam domain noise covariance  $\mathbf{K}_{\mathcal{B}_\perp} \in \mathbb{C}^{|\mathcal{B}_\perp| \times |\mathcal{B}_\perp|}$  is diagonal). Consider a channel state information (CSI)  $m$  indicative for a single antenna receiver. An optimal beamformer in element domain  $\mathbf{Y}$  that maximizes SINR for a single receiver  $m$ , given a set of interfering sources has infinite precision. Its corresponding beam covariance matrix is  $\mathbf{X}_{\mathcal{B},m} = \mathbf{K}_{\mathcal{B}} \mathbf{s}_m^T$ . In the finite case,  $\mathbf{s}_m$  of length  $B$ , is a binary selection vector indicating which beams to activate in order to serve  $m$  by a coefficient 1 if it is inside the corresponding beams' 3dB contour, and 0 elsewhere. Orthogonal fixed beams are created via Butler matrix beamformer in the analog domain or via DFT matrix  $\mathbf{W}$  for digital beamforming, with  $\text{rank}(\mathbf{W}) = N$ , such that  $N = 2^p \geq B$ .

In DSP beam processing, our proposed system model for the MEO scenario employs a  $N \times N = 16 \times 16$  2D-DFT matrix  $\mathbf{W} \in \mathbb{C}^{16 \times 16}$ . We set  $B = N \times N$ , w.l.o.g. As detailed,  $\mathbf{Y}$  is modeled as

$$\mathbf{Y} = \mathbf{W} \mathbf{X} \mathbf{W}^T \quad (1)$$

Given a set of co-multiplexed users scattered over the coverage area, there exists a subset of orthogonal spotbeams  $\mathcal{B}_L = \{m_1, m_2, \dots, m_L\} \subset \mathcal{B}$ , whose union covers all the users. This results in  $\text{rank}(\mathbf{X}_{\mathcal{B}_L}) \leq \min(L, B)$ . The activation of the remaining  $B - L$  spotbeams is power-inefficient. At each cycle, to construct  $\mathbf{Y}$ , it is sufficient to select  $V = \lceil \sqrt{L} \rceil$  DFT column vectors from  $\mathbf{W}$ , such that  $\mathbf{W}^T$  is invertible and  $\text{rank}(\mathbf{W}^T) = V$ .

From the relationship  $\mathbf{X}_{B,m} = \mathbf{K}_B \mathbf{s}_m^T$ , if  $\mathbf{K}_B$  is not invertible, then  $\mathbf{X}_B$  is also invertible. The uncorrelated noise covariance  $\mathbf{K}_{B_L} \in \mathbb{C}^{B \times B}$  of rank  $V$  has few coefficients not close to 0 is said to be sparse. However, it is not diagonal, which means  $\mathbf{X}_B^T$  is invertible. To solve this, we notice that there exists a (selection) matrix  $\mathbf{A}$  such that

$$\mathbf{X} = \mathbf{X}_{B_L} \mathbf{A} \quad (2)$$

and  $\text{rank}(\mathbf{X}^T) = L$ . At each RF resource slot one user from each selected spotbeam is scheduled for transmission. Their corresponding symbols constitute  $\mathbf{X}$ . In section IV-A, we show that  $\mathbf{A}$  too is sparse and that its design depends on the scheduling procedure. In this paper we simplify this procedure using two relevant cases for our implementation, namely, for a number of uniformly separated users in the target coverage area (illuminated by  $\mathcal{B}$ ), and for the case of two adjacent non-orthogonal beams.

### B. 2D-FFT

The two dimensional DFT operation can be expressed as a function of a matrix  $\mathbf{X}$  of complex values that represent the complex samples of each of the beams, assuming that  $\mathbf{X}$  is a square matrix of dimensions  $N$ , and  $\mathbf{W}$  is the DFT matrix of dimension  $N$ . The output of the 2D-DFT operation can be written as in Equation 1. When the  $N$ -point DFT is implemented directly using matrix multiplication, the computational complexity is  $N \times N$  complex multiplications and  $N \times (N-1)$  complex additions, with a total computing cost of  $\mathcal{O}(N^2)$  [11]. Fast Fourier transform (FFT) techniques can be utilized to reduce the DFT computation's computational cost to  $\mathcal{O}(N \log N)$ , that explains the real-time beamformers can be more effectively implemented with less hardware and reduced power than matrix-by-vector multiplication. The efficiency of the OBPs on satellites can be effectively improved in terms of power reduction, mass reduction, and throughput gain using FFT-based digital beamformer. As presented in Fig. 1, the proposed system model for the MEO scenario employs a 2D-FFT since  $10 \times 10$  RF chains require a minimum of  $16 \times 16$  2D-FFT.

### C. Spatial Windowing

In the spatial windowing process,  $10 \times 10$  successive outputs from the 2D-FFT are chosen, and each is connected to an antenna element via an RF chain analog interface. Due to the smaller array aperture, discarding the FFT outputs increases the power supporting for each radiation element in comparison to its maximum utilization. The  $N$  (size of 2D-FFT) beam-pointing directions are maintained by the system, nevertheless, as they rely on the incremental phase shifts produced by the FFT operation, which are preserved in the available  $10 \times 10$  antennas. It might be necessary to design the array with fewer elements to adhere to the physical payload restrictions, such as the available area, mass, and power. To avoid rapid transitions in beam hopping operations, for example, the design can also be employed to form overlapping beams.

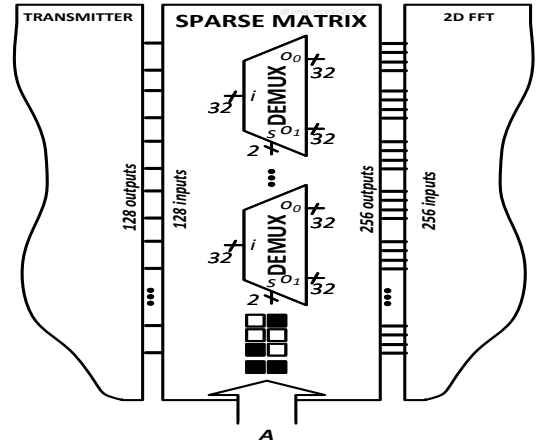


Fig. 2: Simplified sparse matrix implementation general diagram.

## III. IMPLEMENTATION METHODOLOGY OF THE PROPOSED BEAMFORMER

This section discusses the implementation of different FFT architectures selected for technical baseline solutions of efficient digital beamformer.

### A. Architecture of The Sparse Matrix

The sparse matrix architecture is composed of an FFT Selection (Fig. 2), where the selection matrix  $\mathbf{A}$  defines the input beams that will be directed to one of two different outputs. The implementation (Fig. 2) receives the 128 32-bit inputs (representing a complex value) and a 2-bit vector of 128 locations (representing the selection matrix  $\mathbf{A}$ ), delivering 256 32-bit outputs for the 2D-FFT block.

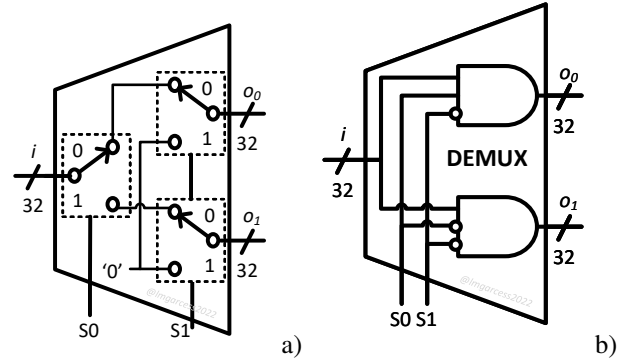


Fig. 3: Simplified sparse matrix implementation a) functional demultiplexer architecture. b) basic architecture.

### B. Architecture of The 2D-FFT

The block is based on a demultiplexer architecture shown in Fig. 3a, where each input is mapped to one of two possible outputs, depending on the selection vector  $\mathbf{s}$  coming from the selection matrix  $\mathbf{A}$ . In this 2-bit vector, the most significant bit ( $S1$ ) works as an enable signal, whereas the least significant

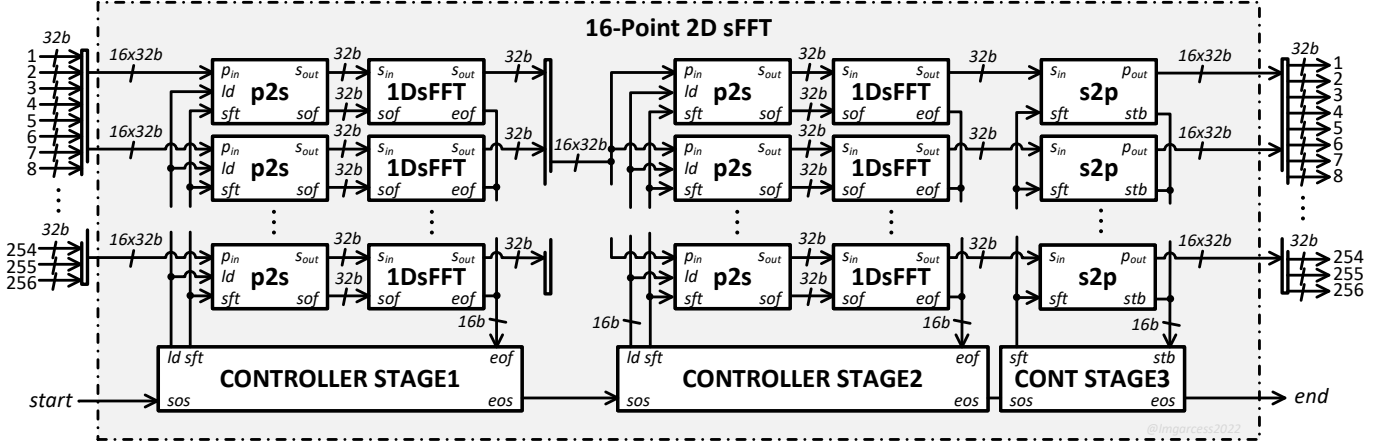


Fig. 4: Proposed architecture of conventional 2D-FFT at an operating frequency of 2 GHz.

bit selects to which output the 32-bit input will be driven ( $o0$  or  $o1$ ). A possible implementation of this logic cell using logic gates is shown in Fig. 3b.

The conventional benchmark 2D-FFT was implemented with the Xilinx IP FFT9.1, then the fully unrolled 2D-FFT using radix-4 algorithm, and the proposed fully unrolled 2D-FFT with radix-4 and 4-bit TF quantization. FFT architectures implemented using the radix-2 generate a computation complexity of multiplications to  $N/2 \times \log_2(N)$  and additions to  $N \times \log_2(N)$ . Similarly, radix-4 has  $\frac{3N}{4} \times \log_4(N)$  and  $N \times \log_2(N)$  as the total multipliers and adders, respectively. Even though the area of the adders is the same in both implementations, radix-4-based FFT requires 25% fewer multipliers than radix-2-based FFT. Hence radix-4 algorithm is considered as the technical baseline solution for implementation of fully unrolled 2D-FFT and 4-bit TF quantized fully unrolled 2D-FFT. Furthermore, the MEO scenario employs a  $16 \times 16$  2D-FFT since the number of RF chains accounts to  $10 \times 10$ . The required bandwidth with each beam is 125 MHz in this typical application.

1) *Conventional 2D-FFT*: The 16-point FFT IP from Xilinx Vivado was employed as the conventional benchmark solution for 2D-FFT implementation. Here the parameters like clock frequency of 125 MHz, and transform length of sixteen with pipelined streaming input/output (I/O) were considered. There were challenges when implementing the 2D-FFT using the conventional FFT algorithm since it is serial in nature and produces one sample per clock cycle with an iteration interval of sixteen. In this regard, two solutions are proposed to implement it. First solution is to operate the conventional 2D-FFT at 2 GHz (16 times the required frequency) to obtain the required frequency of 125 MHz with each beam using the architecture shown in Fig. 4. The second solution is to replicate the previous block sixteen times using  $256 \times 256$  FFT architecture since 256 output samples per clock cycle are needed at once.

Considering the first solution presented in Fig. 4, the in-

coming 256 32-bit data samples in parallel at 125 MHz are sub-categorized into sixteen sets of sixteen data samples and passed on to the sixteen parallel-to-serial (p2s) blocks (i.e., 1 to 16 on first p2s block, then 17 to 32 on the second p2s block, and so on till 241 to 256 on the sixteen p2s block). The CONTROLLER STAGE1 generates the load (ld) and shift (sft) signal to the p2s block. The output samples of the p2s block are generated at a rate of 2 GHz. The data samples are then passed on to the sixteen 1DsFFT (here 1DsFFT refers to one 16-point FFT) simultaneously with the start of the frame (sof) control signal. The 1DsFFT generates samples at the rate of 2 GHz in a serial manner, and the output pattern of first 1DsFFT is 1, 2, ..., 16, second is 17, 18, ..., 32, and so on, until the last 1DsFFT generates 241, 242, ..., 256 is indicated through the end of frame (eof) control signal.

Later, the rewiring must be performed on this data before transmitting the data samples to the second set of 1DsFFT (for column-wise operation). If observed carefully, the generated data samples from the first set of sixteen 1DsFFT are rewired. Another way to express it is that each 1DsFFT's column outputs follow a rewired pattern, with the first column having 1, 17, 33, ..., 241, the second column having 2, 18, 34, and so on through the last column, which has a pattern of 16, 32, and 256. To execute this operation (writing data horizontally and reading data vertically), another set of p2s blocks are employed, which has a total of  $16 \times 16$  registers. The data samples are provided to another set of 1DsFFT for column-wise FFT operation. The CONTROLLER STAGE2 handles the loading and shifting of data to and from p2s and 1DsFFT blocks. The serial output samples are then processed to serial-to-parallel (s2p) block, which generates samples at a rate of 125 MHz with the help of controller CONT STAGE3.

The second solution of implementing the 2D-FFT of  $16 \times 16$  using case1 is to implement the design with  $256 \times 256$  FFTs i.e., instead of running the design at 2 GHz, the hardware mentioned in Fig. 4 is to be duplicated sixteen times. This proposed architecture is presented in Fig. 5, where the archi-

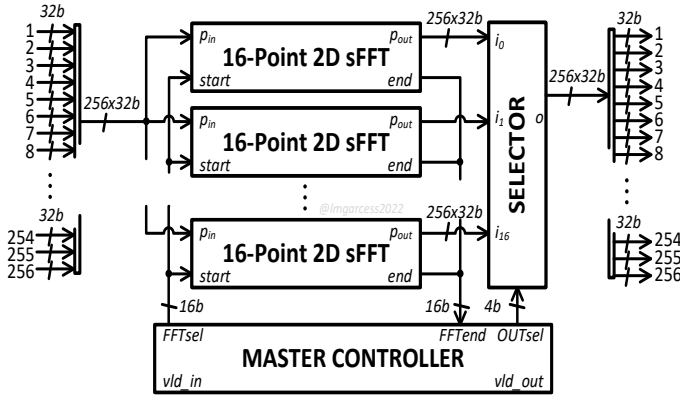


Fig. 5: Proposed architecture of conventional 2D-FFT at an operating frequency of 125 MHz.

ture of Fig. 4 is replicated sixteen times. The MASTER CONTROLLER plays a crucial role in loading of data to each of the 16-point 2D sFFT blocks. There is a total of sixteen 16-point 2D sFFT blocks. The first set of 256 data samples, i.e., 1, 2, ..., 256, is provided to first 16-point 2D sFFT block, then the second set, i.e., 257, 258, ..., 512, is provided to 16-point 2D sFFT block, and so on until the sixteenth 16-point 2D sFFT block receives the data samples 3841, 3842, ..., 4096. Here, the MASTER CONTROLLER generates a start signal (FFTsels) to each of the 16-point 2D sFFT block to load the data. The selector block selects the 256 data samples (data width of  $256 \times 32$  bits) from each of the 16-point 2D sFFT block to generate the required data samples in parallel. Also, to execute the rewiring operation, the number of registers used in rewiring in Fig. 4 is multiplied by sixteen times, i.e.,  $256 \times 16$  registers. The second solution of conventional 2D-FFT presented in Fig. 5 was adopted in this paper for implementing and estimating the hardware resources and power consumption since the first solution is not feasible to implement at an operating frequency of 2 GHz.

2) *Fully unrolled 2D-FFT*: The functional block diagram for implementing 2D-FFT using fully unrolled radix-4 algorithm is presented in Fig. 6. The digital beamformer is implemented using the 2D-FFT procedure. A 2D-FFT is conducted by first row-wise and then column-wise operations. In Fig. 6, a  $16 \times 16$  2D-FFT digital beamforming is carried out. As a result, sixteen 16-point FFTs are needed in total to complete the row-wise operation, and sixteen 16-point FFTs are needed to complete the column-wise operation, as shown in Fig. 6. Each TF in the proposed 4-bit TF quantized 2D-FFT has a width of 4 bits, compared to the conventional and fully unrolled architecture's 16-bit TFs. The 256 inputs can be seen as being passed to the sixteen blocks of the 16-point 1D-FFT in order to perform the column-wise FFT process. The outputs from the sixteen modules of the 16-point 1D-FFT are then reconnected to another sixteen modules of the 16-point 1D-FFT for the row-wise processing by the rewiring block. The dashed lines indicate inputs to the second set of

sixteen 1D-FFT. The rewiring employed in this architecture is based on hardwired methodology. The 1D-FFT fully unrolled

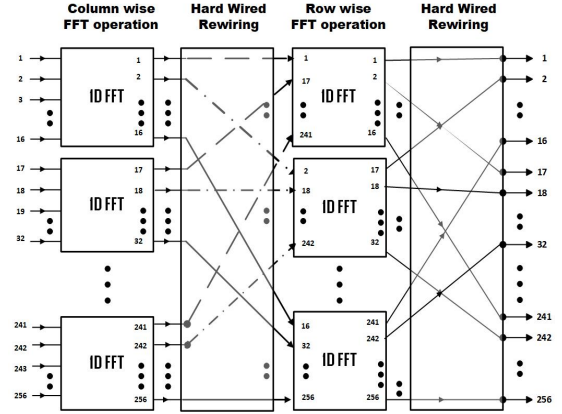


Fig. 6: Architecture of fully unrolled 2D-FFT.

architecture employed in 2D-FFT construction is based on [11].

3) *Fully unrolled 4-bit TF quantized 2D-FFT*: The proposed fully unrolled 4-bit TF quantized 2D-FFT is also based on the radix-4 algorithm and employs the same architecture mentioned in Fig. 6. But the novel methodology that has been proposed in this architecture is that, instead of 16-bit twiddle width, a 4-bit TF width has been incorporated. The advantage of TF quantized FFT, however, is that there will be a lower area utilization and power consumption than with conventional and 16-bit twiddle width FFT. Furthermore, the equivalent signal-to-noise ratio (SNR) for the TF quantized FFT with complex Gaussian distribution input was found to be 24.6 dB, 35.6 dB, and 47.9 dB, respectively. These values are roughly proportional to the theoretical SNR given by  $SNR = 6b + 1.72$ , where  $b$  is the number of bits.

### C. Architecture of Spatial Windowing

The spatial window depends on two factors of design: the number of subarrays of the antenna and/or the overlapping beams. The 256 outputs from the 2D-FFT is be represented

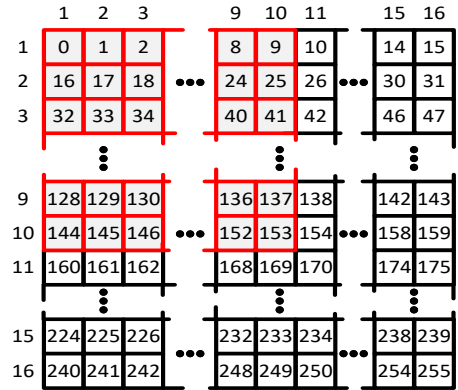


Fig. 7: Spatial Window functional diagram.

as a  $16 \times 16$  2D array, and a part of this window is working as the input of the subarrays of the antenna. Fig. 7 shows the functional diagram of the module for the fixed  $10 \times 10$  subarray. In this case, the module selects  $10 \times 10$  signals (in red) from the  $16 \times 16$  outputs of the 2D-FFT and passes them to the subarrays while the rest of the inputs outside the window (in black) are discarded.

The implementation takes from a total of 256 inputs represented as a row, a set of 100 inputs, considering its position in the  $16 \times 16$  window. Each input received a complex value of 32 bits (16 bits signed real and 16 bits signed imaginary) and is returned to the output without any modification if the signal is inside the window or discarded otherwise.

#### IV. RESULTS AND DISCUSSIONS

This section presents the MATLAB simulation and the hardware results targeting xcvu29p-l2fsga2577e FPGA.

##### A. MATLAB Simulation

In this section, we simplify the spatial multiplexing properties of the quantized DFT vector selection using a lossy multiplexer. We assume a PHY scheduling logic associates a single user per spotbeam to guarantee orthogonality. This can be done by estimating the CSI from  $2Z$  available real samples of finite size  $T = 2^q \leq L, q \leq p$ , to form an indicative CSI matrix  $\mathbf{H}$  corresponding to  $\mathbf{X}$ ; then selecting the  $F \leq V$  DFT vectors contributing to the beams with the  $F$  most significant column sum powers of  $\mathbf{X}$ . Furthermore, we take  $M = L < B$  as a generic case, and  $M = L = 2$  as a simplified case. In the generic case we assume the scheduler realizes ideal input of length  $Z = 128$  similar to the case of URA with half wavelength spacing. Hence we can take  $F = V$ .

Ultimately, the samples of the input  $\mathbf{X}$  designate a set of orthogonal beams  $\mathcal{B}_L$ . In this case, at most  $V$  corresponding DFT vectors per cycle are sufficient to be selected. After the DFT operation<sup>2</sup>, and due to finite precision ( $q \leq p$ ), the power spread of  $\mathbf{Y}$  is limited in the frequency domain<sup>3</sup>. Therefore, we can replace  $\mathbf{H}$  in the delay domain by a sparse matrix using information about the user spatial distribution as handled by the scheduler:  $\hat{\mathbf{H}} \leftarrow \mathbf{S}\mathbf{H} + \epsilon : \epsilon \sim \mathcal{CN}(0, \mathbf{\Omega})$ , where  $\mathbf{\Omega}$  is the delay power spread and  $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{0}_{L \times (B-L)}]^T$  is the spatial-user-adjacency matrix of size  $M \times M$  having  $L < B$  distinct rows with non-zero values. In the beam domain, user multiplexing is equivalent to selecting the  $V$  vectors contributing to the  $L$  beams. The matrix  $\mathbf{A}$  is then sparse at each RF slot [12].

In our digital implementation, the samples in  $\mathbf{X}_{\mathcal{B}_L}$  are coded using  $V$  DFT vectors whose coefficients constitute the transmitted symbol. To illustrate, we take  $L = M = 100$  users

<sup>2</sup>In the infinite case, the DFT matrix  $\mathbf{W}_{\mathcal{B}_L}$  is proven to be the eigenvector matrix of the correlation matrix  $\mathbf{X}_{\mathcal{B}_L}$ . A crucial result from this statement is that the column vectors of  $\mathbf{H}$  become approximately sparse after the DFT operation.

<sup>3</sup>Achieving orthogonality,  $\mathbf{Y}_{\mathcal{B}_L}$  is a unitary matrix if we consider infinitely long repetitions of the cycle period. In the finite-precision case its rows have  $N$  distinct Fourier coefficients corresponding each to the input signal defined by the symbols in  $\mathbf{X}_{\mathcal{B}_L}$  and the 2D DFT operation.

in the generic case. For  $N = 16$ , we show the first 128 entries of the sparse representation of matrix  $\mathbf{S}$ , and matrix  $\mathbf{A}$  in Fig. 8. We take  $q = 3$  (ie  $T = 8 < 9$  to showcase worst case imprecision) at each cycle then build the matrix  $\mathbf{A}$  for DFT selection.

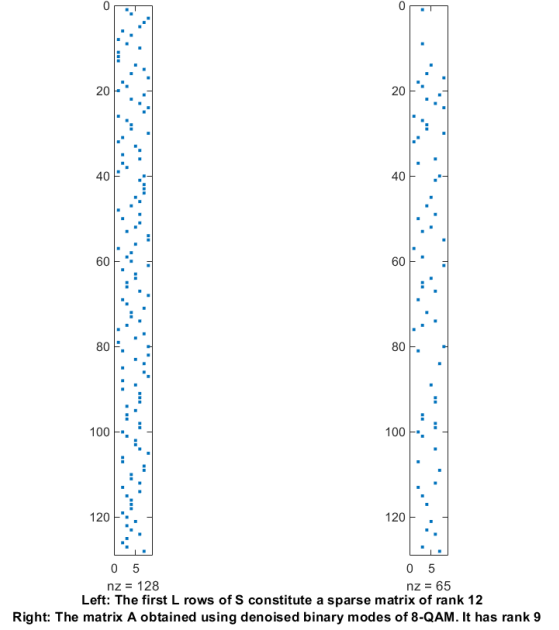


Fig. 8: Left: the  $M$  users (scattered each on the 3dB spotbeam of an  $N \times N$  URA) constitute a sparse matrix  $\mathbf{S}$  of rank 12. Therefore from  $p = 4$ , we can select at most  $F = V = 12 < N = 2^p$  vectors from  $\mathbf{X}$  as input to the 2D DFT operation. Right: a denoised DFT selection matrix  $\mathbf{A}$  for  $q < p$  illustrates both spatial lag and finite precision.

The most area-power efficient cycle has some  $F \leq V$  vectors selected using the selection matrix's first 8-length sequence in Fig. 8 (simplified case). The sparse matrix implementation, in this case, can be exactly represented by a multiplexer as in Fig. 3, for  $\mathcal{B}_L = \{m_1, m_2\}$ , where  $m_1$  and  $m_2$  are two spotbeams with non overlapping 3dB contours.

##### B. FPGA Results

The hardware implementation results are based on the out-of-context (OOC) synthesis, which estimates the area and power consumption independent of FPGA I/Os. The implemented beamformer has a architecture of 128 inputs to sparse matrix, then an input of 256 data samples (these data samples are the output of sparse matrix) to 2D-FFT, and 100 data samples as the output of spatial windowing block. Therefore, a total of  $128 \times 32$  inputs pins and  $100 \times 32$  output pins are required on the FPGA IO. Since it is not possible to implement a design with these many IOs, OOC synthesis was employed. With a maximum operating frequency of 125 MHz, the xcvu29p-l2fsga2577e ultrascale+ FPGA is targeted for implementing the proposed architectures. Here the proposed



architectures refer as case1 with sparse matrix, conventional 2D-FFT, and spatial windowing. Then case2 as sparse matrix, fully unrolled 2D-FFT, and spatial windowing followed by case3 as sparse matrix, 4-bit TF quantized fully unrolled 2D-FFT, and spatial windowing. The hardware results in terms of power and area consumption at an operating frequency of 125 MHz are presented in Table I.

TABLE I: Resource utilization with xcvu29p-l2fsga2577e FPGA

Resources	Dynamic Power (W)	LUTs	FFs	DSPs
Case1	15.119	567,776	1,388,032	6144
Case2	5.618	134,304	128,195	640
Case3	5.565	146,688	128,131	0

The real-time beamformer in a typical MEO scenario employs a bandwidth of 1500 MHz, and hence the results in terms of power consumption in Table I are extrapolated for this bandwidth. Since the hardware results in Table I are at an operating frequency of 125 MHz, the MEO scenario hardware results are extrapolated by a factor of twelve. These extrapolated results are present in Table II.

TABLE II: Extrapolated resource utilization in MEO

Resources	Dynamic Power (W)	LUTs	FFs	DSPs
Case1 MEO	181.428	6,813,312	16,656,384	73,728
Case2 MEO	67.416	1,611,648	1,538,340	7,680
Case3 MEO	66.78	1,760,256	1,537,572	0

It can be noted from Table II that the case1 MEO does not fit into a single FPGA since the number of DSPs is 12,288, the number of FFs is 3,456,000, and the number of LUTs is 1,728,000. On the other hand, case2 MEO and case3 MEO fit in a single FPGA. Also, the high number of DSPs in case2 MEO is a constraint for implementation in a real-time scenario since the transmitter modules such as source encoding, channel encoding, linear precoding and so on consume extra DSPs. Therefore, case3 MEO, which employs sparse matrix, proposed 4-bit TF quantized fully unrolled 2D-FFT, and a spatial windowing is a feasible solution of fully digital beamformer in MEO satellite scenario.

## V. CONCLUSION

This paper addressed the problem of implementing codebook-based digital beamformers for on-board processing in MEO satellites. The paper focused particularly on FPGA implementations that are efficient from both power-consumption and area-occupation viewpoints. The proposed codebook-based digital beamformers rely on efficient FFT realizations to generate codewords that correspond to the columns of the DFT matrix. More specifically, the resulting efficient digital beamformer is composed by a sparse-matrix-based user selection, followed by a 2D-FFT and a spatial windowing module for selecting the appropriate antenna pattern. Different architectures were investigated, including the

use of conventional 2D-FFT, the use of fully unrolled 2D-FFT, and the use of TF quantized fully unrolled 2D-FFT. The implementation results on the Virtex ultrascale plus FPGA board indicated substantial gains, up to 50% in area and power, with the proposed beamformer realization via TF quantized fully unrolled 2D-FFT. The resulting lower power consumption and area utilization indicate that the proposed architecture of digital beamformer is a promising solution for satellites using DRA communication.

## ACKNOWLEDGMENT

This work was supported by European Space Agency under the project number 4000134678/21/UK/AL “EFFICIENT DIGITAL BEAMFORMING TECHNIQUES FOR ON-BOARD DIGITAL PROCESSORS (EGERTON)” and SES S.A. (Opinions, interpretations, recommendations and conclusions presented in this paper are those of the authors and are not necessarily endorsed by the European Space Agency or SES). This work was supported by the Luxembourg National Research Fund (FNR), through the CORE Project (ARMONY): Ground-based distributed beamforming harmonization for the integration of satellite and Terrestrial networks, under Grant FNR16352790

## REFERENCES

- [1] S. H. Blumenthal, “Medium Earth Orbit Ka Band Satellite Communications System,” in *MILCOM 2013 - 2013 IEEE Military Communications Conference*, pp. 273-277, 2013.
- [2] Inigo del Portillo, Bruce G. Cameron, Edward F. Crawley, “A technical comparison of three low Earth orbit satellite constellation systems to provide global broadband,” in *Acta Astronautica*, Volume 159, pp. 123-135, 2019.
- [3] F. Vidal, H. Legay, G. Goussetis, T. Ströber and J. -D. Gayraud, “Benchmark of MEO Multibeam Satellite Adaptive Antenna and Payload Architectures for Broadband Systems,” in *2020 10th Advanced Satellite Multimedia Systems Conference and the 16th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, pp. 1-8, 2020.
- [4] A. Arora, C. G. Tsinos, B. Shankar Mysore R, S. Chatzinotas, and B. Ottersten, “Analog beamforming with antenna selection for large-scale antenna arrays,” in *Proc. IEEE Int. Conf. Acoust., Speech and Sig. Process. (ICASSP)*, Toronto, ON, Canada, pp. 4795-4799, Jun. 2021.
- [5] X. Zhai, X. Chen, J. Xu, and D. W. Kwan Ng, “Hybrid beamforming for massive MIMO over-the-air computation,” *IEEE Trans. Commun.*, vol. 69, no. 4, pp. 2737-2751, Apr. 2021.
- [6] I. Ahmed, et al, “A survey on hybrid beamforming techniques in 5G: Architecture and system model perspectives,” *IEEE Commun. Surv. Tut.*, vol. 20, no. 4, pp. 3060-3097, Fourthquarter. 2018.
- [7] P. Angeletti and M. Lisi, “Digital beam-forming network with reduced complexity and low power consumption for array antennas,” in *Proc. 21st Ka and Broadband Commun. Conf.*, 2015.
- [8] D. Suarez, R. J. Cintra, F. M. Bayer, A. Sengupta, S. Kulasekera, and A. Madanayake, “Multi-beam RF aperture using multiplierless FFT approximation,” *Electronics Lett.*, vol. 50, no. 24, pp. 1788-1790, Nov. 2014.
- [9] E. O. Brigham and R. E. Morrow, “The fast Fourier transform,” in *IEEE Spectrum*, vol. 4, no. 12, pp. 63-70, Dec. 1967.
- [10] Fast Fourier Transform v9.1, (2022), LogiCORE IP Product Guide Vivado Design Suite PG109.
- [11] R. Palisetty et al., “Area-power analysis of FFT based digital beamforming for GEO, MEO, and LEO scenarios,” in *Proc. IEEE Veh. Technol. Conf. (VTC) Spring*, Helsinki, Finland, pp. 1-5, Jun. 2022.
- [12] A. Adhikary, J. Nam, J.-Y. Ahn, and G. Caire, “Joint spatial division and multiplexing: The large-scale array regime,” in *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6441-6463, Oct. 2013.