FPGA Implementation of Back Projection Algorithm for Radar Imaging

Nieves Crasto, T Kishore Kumar
Department of Electronics and Communication Engineering
National Institute of Technology
Warangal, India
Email: nievescrasto@gmail.com

D. Anuradha, Paramita Barua, Sirisha Nemani
Electronics and Radar Development Establishment
DRDO
Bangalore, India

Abstract—The growing security concerns worldwide have lead to the development of wideband radar systems to estimate the position and shape of objects behind barriers. For an accurate reconstruction of the scene of interest, a huge amount of measurements are required. Thus wideband signals with large array apertures are required to generate high resolution images. This paper describes how the raw data is processed using back-projection algorithm in order to reconstruct a 2-D image with the help of FPGA. It also deals with the practical advantages of back projection algorithm compared to other beamforming algorithms. The raw data is generated using stepped frequency continuous wave radar.

I. INTRODUCTION

Electromagnetic waves have the ability to penetrate man-made building materials and to image targets behind opaque structures. These attributes are highly desirable for a range of organizations, including the police, fire fighters, defense forces, etc. There are many challenges facing the development of a successful radar imaging system. The system should be reliable, portable, light in weight and small in size. It should also have high down-range and cross-range resolution.

II. STEPPED FREQUENCY CONTINUOUS WAVE RADAR (SFCW)

Radars used for detecting objects through a barrier may be implemented as either time-domain (short-pulse) or frequency-domain (stepped-frequency) systems. In this paper, we have used stepped frequency continuous wave (Fig.1) radar because it preserves the phase information which makes it easier to align the received echo signals. Further stepped-frequency implementation of a wideband signal allows changing the emitted power over the signal bandwidth. We can therefore compensate for frequency-dependent power attenuation due to the barrier. But the main advantage of SFCW is its instantaneous narrow bandwidth which makes it possible to use lower A/D conversion sampling rates, lower peak power sources, and slower computers to process smaller sets of data.

III. RADAR IMAGING METHODS

As it is shown on Fig.2, SAR scanning causes one point in space domain $S(X,Z)$ to be represented in B-scan $A(X,t)$, as a hyperbola. In order to transform $A(X,t)$ domain back to the $S(X,Z)$ domain, some migration algorithm has to be used. The signal received in the given time can be reflected from all points that lie on the locations where TOA (Time Of Arrival is the time taken by the wave to travel from transmitter to target and back to receiver) is constant. The points that have the same TOA are on a hyperbola $H$ with focuses at transmitter and receiver positions[3]. By other words, the task of imaging is to transform time domain data back into depth domain, where depth means the coordination from antenna to the target (X-axes, looking direction). Such transformation is often called migration.

When imaging a target using the conventional SAR technique, the radar sensor physically moves down a linear track acquiring range profile data at known locations. This is too slow to be used in practical situations. Therefore rather than physically moving the radar down a rail, it is better to employ an antenna array. In this paper, we have considered an antenna array system of 4 transmitters and 4 receivers.

There are several migration algorithms which can be used for through the barrier imaging [3]. These are mainly divided into two categories; one based on geometric approach (back-projection algorithm) and the other on solving wave equation (Kirchoffs migration and F-k migration [5]). Wave equation based migration involves the use of calculus and it is difficult to implement these differential equations using FPGA.

Thus we have chosen back-projection algorithm as a method of imaging because it enables us to determine the intensity values at desired pixel locations and it is more suitable for parallel hardware implementation.
A. Back Projection Algorithm

The fundamental aspect of back projection is that it is a matched-filter implementation of time-domain correlation. The basic principle of back projection is to correlate data collected at each antenna element as a function of round-trip delay time which is the time to travel from the transmitter to the pixel and then back to a receiver (shown in Fig.3). Back projection algorithm time shifts the received echo signal at each antenna position so as to align it to a particular pixel element in the image map. Then these time shifted echo signals are coherently summed up to generate the pixel’s intensity value. Following this, all the recorded amplitudes from each receiving antenna are added together on the spatial grid. At the target locations the signal amplitudes will add up coherently and display a bright spot on the image [1].

The (x_{T}(m), y_{T}(m), z_{T}(m)) is the coordinate of the mth transmitter and the (x_{R}(n), y_{R}(n), z_{R}(n)) is the coordinate of the nth receiver. The result of this procedure is a 2-D radar image, which provides both the range and the position of potential targets behind the barrier.

1) Timing Requirement: If the received I/Q data is read in a serial manner then the time to generate one frame is given by

\[ \text{Time}_{\text{frame}} = M \times N \times K \times P \times \text{clock}_{\text{period}} \] (5)

location until all the N receive locations have been used sequentially;
5) The above data measurement process is then repeated by the sequential use of M transmit locations.

Mathematically, in free space, the back projected signal at pixel (p,q) in the image is given by[2]

\[ I(p,q) = \sum_{m=1}^{M} \sum_{n=1}^{N} \sum_{k=1}^{K} S_{mn}(\omega_{k}) \ast \exp(j\omega_{k}\tau_{mn}(p,q)) \] (1)

\[ \tau_{mn}(p,q) = \frac{R_{Tn}(m) + R_{Rn}(n)}{c} \] (2)

\[ R_{Tn} = \sqrt{(x_{T}(m) - p)^2 + (y_{T}(m) - q)^2 + (z_{T}(m) - 0)^2} \] (3)

\[ R_{Rn} = \sqrt{(x_{R}(n) - p)^2 + (y_{R}(n) - q)^2 + (z_{R}(n) - 0)^2} \] (4)

Where
- \( c \) is the speed of light in free space (m/s);
- \( S_{mn}(\omega_{k}) \) is the measured complex amplitude of the return when the transmitter is at the mth location and the receiver is at the nth location;
- \( \tau_{mn}(p,q) \) is the focusing delay applied to the output of the nth receiver when the transmitter is at the mth location;

The \((x_{T}(m), y_{T}(m), z_{T}(m))\) is the coordinate of the mth transmitter and the \((x_{R}(n), y_{R}(n), z_{R}(n))\) is the coordinate of the nth receiver. The result of this procedure is a 2-D radar image, which provides both the range and the position of potential targets behind the barrier.

1) Timing Requirement: If the received I/Q data is read in a serial manner then the time to generate one frame is given by

1) Divide the whole region into small pixels (p×q);
2) Let the transmitter located at the mth position \((x_{T}(m), y_{T}(m), z_{T}(m))\) illuminate the scene with a sequence of K monochromatic signals spanning the desired bandwidth (stepped frequency signal consisting of K frequency steps);
3) The complex amplitudes of the returns are measured and stored by the receiver at nth position \((x_{R}(n), y_{R}(n), z_{R}(n))\);
4) The process is repeated with the transmitter at the mth
For a clock period of 140 MHz, $\text{Time}_{\text{frame}} = 1.75s$. This is very slow for real time applications. In order to increase this speed, the received data is stored in 4 dual port Block RAMs and the data is read in parallel. Using this logic the speed reduces by a factor of 8 and gives $\text{Time}_{\text{frame}} = 0.218s$, i.e. approximately 5 frames/s.

IV. FPGA IMPLEMENTATION

The back-projection algorithm has been implemented using Xilinx Virtex5 FX100T FF1738-2C device. The entire VHDL code has been divided into the following sub-modules as showed in Fig.4.

1) Time Delay
2) $\omega$ values (calculates the phase angle in radians)
3) Radian (calculates sine and cosine values)
4) Complex Multiply (multiplies the received I/Q data with sine and cosine values)
5) Sum (generates real and imaginary value of a pixel)
6) Abs (calculates the intensity of the pixel)
7) Mem Write (stores the generated pixels values to be sent for further processing)

The device utilization is summarized in Table II.

### TABLE II

**DEVICE UTILIZATION SUMMARY: IMPLEMENTATION OF BACK PROJECTION ALGORITHM**

<table>
<thead>
<tr>
<th>Logic Utilization</th>
<th>Used</th>
<th>Available</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slice Registers</td>
<td>41940</td>
<td>64000</td>
<td>65%</td>
</tr>
<tr>
<td>Number of Slice LUTs</td>
<td>43258</td>
<td>64000</td>
<td>67%</td>
</tr>
<tr>
<td>Number of fully used LUT-FF pairs</td>
<td>38564</td>
<td>46634</td>
<td>82%</td>
</tr>
<tr>
<td>Number of bonded IOBs</td>
<td>305</td>
<td>680</td>
<td>44%</td>
</tr>
<tr>
<td>Number of BlockRAM/FIFO</td>
<td>176</td>
<td>228</td>
<td>77%</td>
</tr>
<tr>
<td>Number of BUFG/BUFGCTRLs</td>
<td>2</td>
<td>32</td>
<td>6%</td>
</tr>
<tr>
<td>Number of DCM ADVs</td>
<td>1</td>
<td>12</td>
<td>8%</td>
</tr>
<tr>
<td>Number of DSP48Es</td>
<td>214</td>
<td>256</td>
<td>83%</td>
</tr>
</tbody>
</table>

V. EXPERIMENTAL SETUP

For the experimental setup, a metal box at a position of (0.8m,0.0m) is considered as the target. The barrier is located at (0.0m,0.0m). The received echo signal (Fig.5) is processed using back-projection algorithm with the help of MATLAB. This takes around 10s on a Pentium Dual Core Processor running at 3 GHz. Fig.6 shows the image generated using back-projection algorithm with the help of Virtex-5 FPGA which takes around 0.2s. Thus the FPGA implementation of back projection runs approximately 50 times faster and produces an image indicating the true position of the target.

VI. CONCLUSION

In this paper, we conclude that the FPGA implementation of back projection algorithm using stepped frequency continuous wave radar signal can generate high quality images at rate of 5 frames/s. A further improvement in the frame rate can be achieved by splitting the image into sub-images and having separate FPGAs to process these sub-images which can later be clubbed together to generate the final image.

REFERENCES

