An FPGA-based Fan Beam Image Reconstruction Module (Extended Abstract)

Luiz Maltar C. B., Felipe M. G. França, Vladimir C. Alves, Cláudio L. Amorim
COPPE-Universidade Federal do Rio de Janeiro, Caixa Postal 68511, Zip Code 21945-970, Rio de Janeiro- RJ- Brazil
maltar@cos.ufrj.br felipe@cos.ufrj.br amorim@cos.ufrj.br castro@lpc.ufrj.br

1 Introduction

Filtered Back-Projection (FBP) is a well-known algorithm for reconstruction of tomographic images from projections [1-3]. Some of FBP's highlights are: (i) allows agile software implementations, and; (ii) production of images of good quality, i. e., relatively free of artifacts.

Our goal is to reconstruct images from fan beam projections collected by detectors set in a linear array. Figure 1 presents a simplified geometry of the data acquisition system. A projection can be seen as a set of line integrals from a x-ray source up to the detectors. Other projections are collected after successive rotations of the source-detector system over the subject by an step angle $\Delta\theta = 2\pi/(N-1)$, where N is the number of projections.

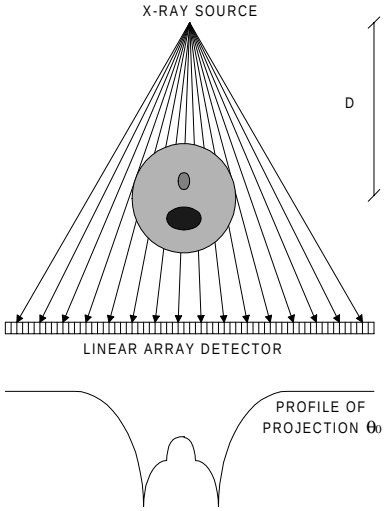

Figure 1: Fan beam geometry.

The Filtered BP algorithm is basically composed by 2 steps,[1]:

(i) Filtering of each projection, $r(sm, \theta_i)$:

$$Q(s_m,\theta_i) = [1/2h(s_m)] \otimes [r(s_m,\theta_i).\frac{D}{\sqrt{D^2+s_m^2}}] \qquad (1)$$

where $h(sm)$ is high-pass filter, sm is the detector position in array detector and D is the distance from the X-ray source to the center of rotation;

(ii) BP processing with linear interpolation of filtered projections is described by the expressions:

$$f(x_k,y_l) \approx \frac{2\pi}{N-1}\sum_{i=0}^{N-1}\frac{1}{U^2(x_k,y_l,\theta_i)}\overline{Q}(s',\theta_i) \qquad (2)$$

$$\overline{Q}(s',\theta_i) = .Q(s_m,\theta_i)(s_{m+1}-s'(x_k,y_l,\theta_i)) + Q(s_{m+1},\theta_i).(s'(x_k,y_l,\theta_i)-s_m) \qquad (3)$$

$$s'(x_k,y_l,\theta_i) = \frac{(x_k\cos\theta_i + y_l\sin\theta_i)}{U(x_k,y_l,\theta_i)} \qquad (4)$$

where $f(x,y)$ is the original image, $\overline{Q}(s',\theta_i)$ represents the linear interpolation of filtered projections $Q(S_m, \theta_i)$ and $Q(S_{m+1}, \theta_i)$ that are closer (from left and right) to the projection of the pixel image position S' over detector array that during the BP processing. So, $S_m = \lfloor S' \rfloor$ and $S_{m+1} = \lceil S' \rceil$ and U is defined as

$$U(x_k,y_l,\theta_i) = 1 + \frac{(x_k\sin\theta_i + y_l\cos\theta_i)}{D} \qquad (5)$$

We have developed a RAM FPGA-based reconfigurable architecture, where each particular task involved in the tomographic processing is served by a specific system configuration that implements a dedicated hardware architecture [6]. In this work, we present a pipelined FPGA-oriented solution to evaluate s' using radix-4-based SRT division scheme.

2 Target Application

We are interested in the reconstruction of 512 x 512 pixel images using about 8 bits per pixel in the BP interpolation. Given these specifications, the evaluation of the s' expression is carried out by using 36 bits to represent the dividend and 18 bits to represent the divisor, thus producing a 18 bit quotient. The value of $sine(\theta i)$ and $cosine(\theta i)$ are known beforehand and are supplied together with the filtered projection.

The values of numerator and denominator (U) in the s' expression can be obtained recursively by the algorithm shown in Figure 2.

We have implemented the SRT division scheme[4,5] in order to calculate s' in redundant radix-4 form that yields 2 bits per stage with quotient digits {-2,-1, 0, 1, 2} (minimal redundancy). The partial remainder is normalized within (-8/3, 8/3) using a divisor normalization inside [0.5,1), the number of bits that has to be examined is equal to 6 in the case of partial remainders and equal to 3 in the case of divisors in order to yield one quotient digit.

The value of U is always positive and, through a simple inspection of Equation 5, it is easy to determine its range. Considering 512 x 512 pixel images, practical values of the distance from center of rotation to the X-ray source, i.e., $D > 512$, the divisor range found is (0.2910, 1.707). This range greatly sim-

plifies the normalization hardware for divisor operands and quotient selection table.

```
for ( ang = 0 ; ang < 360 ;  ang = ang + delta_ang)
 {
   Cos = cossine(ang) ;
   Sin = sine(ang) ;
   N0 = -256. Cos – 256 Sin;
   D0 = 1 - 256. sin + 256. cos  ;
                D          D
   for ( x = 0 ; x < 512; x++)
    {
      N0 = N0 + Cos ;
      D0 = D0 + Sin /D ;
      Num = N0 ;
      U = D0 ;
      for ( y = 0 ; y < 512 ; y++)
       {
           Num = Num + Sin;
           U = U – Cos/D;
           S' = Num/U ;
        }
      }
   }
```

Figure 2: Recursive algorithm to obtain *s'* operands

Our implementation uses replicated hardware (see Figure 3) with a 9 stage pipeline where division steps are evaluated simultaneously aiming at obtaining a 18 bit result at the last stage. This work was developed in a ALTERA 10K50 component, which has 2880 logical elements (having one 4-input LUT and one flip-flop) and 10 blocks of RAM, called EABs. Each EAB can be configured into any of the following four sizes: 256 x 8 bits, 512 x 4 bits, 1024 x 2 bits, and 2048 x 1 bits.

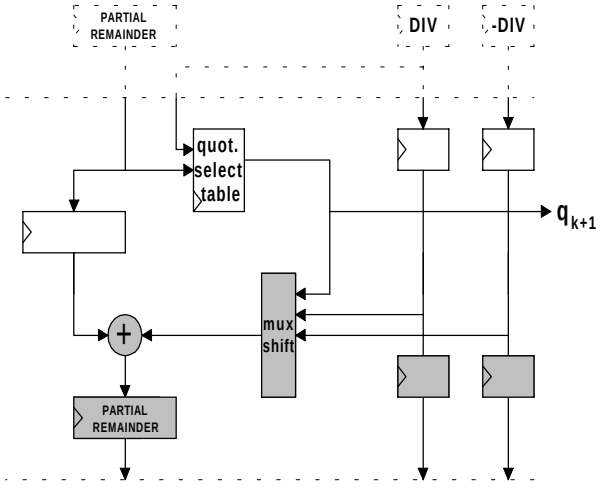

Figure 3: Scheme of a radix-4 i^{th} pipeline stage.

The pipeline division stage takes two cycles to implement the following steps: (i) quotient selection (QS); (ii) divisor multiple formation and partial remainder formation. The QS functions are mapped into ALTERA's EABs configured as 512 x 4 bits size, thus simplifying the circuit design. Although only 3 bits are necessary to represent the quotient, all 4 bits of each EAB are used. The result of the division (at the last stage) is produced in redundant radix-4 form and is converted to conventional 2's complement form by using the "On-the-Fly Conversion" technique [4].

3 Results

We simulated the circuit in MAX PLUS 2, version 9.1. Our preliminary results indicated a total of 2190 LEs and 9 EABs spent in the implementation of the pipelined circuit used to calculate *s'*. This pipeline works at 45 MHz and has the 23 latency cycles.

All adders were implemented as ripple carry, once that the device used has dedicated structures providing fast carry propagation. The pipeline stage time is determined by the cycle time of *s'* operands generation since the adders used in the generation step are wider than the adders in the division step.

4 Conclusions and Future work

We have shown that the complexity of BP reconstruction algorithms from fan beam projections is mainly due to the division operation needed to calculate *s'*. We have introduced a new solution based on a pipelined radix-4 SRT division in contrast to the pipelined BP solution that has been adopted in previous works [2,3]. The latency introduced by our implementation is negligible if compared to the total time taken to reconstruct one image. The use of EABs offered by the chosen FPGA device simplifies the circuit design. Further studies on rounding errors and its impact on the precision of *s'* are needed.

Acknowledgments

The authors want to acknowledge Altera Corporation for providing the ALTERA MAX plus II software. This work was partially supported by CNPq and FINEP, Brazilian research agencies.

References

[1] A.C. Kak and M. Slaney, Principles of Computerized Tomographic Imaging, IEEE PRESS, 1988.

[2] I. Agi, P.J. Hurst and K.W. Current, "An Image Processing IC for Backprojection and Spatial Histogramming in a Pipelined Array," IEEE Journal of Solid-State Circuits , Vol 28, No 3 , March 1993.

[3] L. Lin and V.K. Jain, "Image Processing Using a Universal NonLinear Cell and the Warp wafer," IEEE Trans. On Components, Packaging, and Manufacturing Technology- part B, vol. 17, no. 3, August 1994.

[4] M.D. Ercegovac and T. Lang, Division and Square Root: Digit-Recurrence Algorithms and Implementations. Boston: Kluwer Academic,1994.

[5] S.F. Oberman and M. J. Flynn, "Division Algorithms and Implementations," IEEE Trans. Computers, vol.46, no. 8, August 1997.

[6] L. Maltar C.B., F.M.G. França, V.C. Alves, C.L. Amorim, "Reconfigurable Hardware for Tomographic Processing," proceedings of SBCCI'98 XI Brazilian Symposium on Integrated Circuit Design, Búzios-Rj, Brazil. (IEEE Computer Society PRESS)