

# Accelerating Subsurface Offset Gathers for 3D Seismic Applications

This is a summary of a paper presented at the 77th Society of Exploration Geophysicists (SEG) Meeting, San Antonio, USA, September 2007.

## About the Authors

Oliver Pell is with Maxeler Technologies.

Bob Clapp is with the Center for Computational Earth & Environmental Science (CEES) at Stanford University.

## Summary

The cost of constructing subsurface offset gathers can be a dominant cost in some migration methods. With the increased use of wide azimuth geometries and the resulting 3D angle gathers, over 90% of CPU cycles can be spent in constructing the subsurface offset gathers.

FPGA-based hardware accelerators can offer a solution. FPGAs are custom silicon chips that can be configured to implement highly parallel arithmetic circuits. Even though they run at relatively low clock frequencies, with massive parallelism they can provide order-of-magnitude greater performance than conventional CPUs.

Subsurface offset volumes at a given depth  $z$  and location  $x$  can be created by applying:

$$I(\mathbf{h}, \mathbf{x}, \mathbf{z}) = \sum_s \sum_w \mathbf{S}(\mathbf{x} - \mathbf{h}, \mathbf{z}, w, s) \mathbf{G}(\mathbf{x} + \mathbf{h}, \mathbf{z}, w, s)^*$$

where  $S$  is the source wavefield,  $G$  is the receiver wavefield,  $w$  is the temporal frequency and  $s$  is the shot index.

The arithmetic capabilities of the FPGA are substantially in excess of that required, so acceleration is limited by the rate at which data can be streamed across the PCI Express bus. By reducing the number of bits stored for each value from 32 to 16, the performance of the operation can be increased, with negligible degradation in the output image.

To test the applicability of this approach we compare the result of constructing angle gathers for the 2D Marmousi synthetic data set, as shown in [Figure 1](#). The top-left panel shows an angle gather constructed from the CPU implementation of the imaging condition. The remaining panels show various FPGA implementations with different bit widths. Note that they are visually kinematically identical.

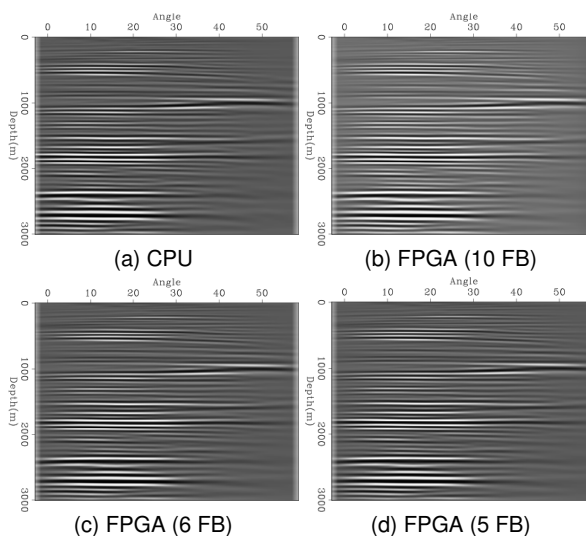


Figure 1: The same angle gather obtained by different implementations. FPGA implementations show the number of fraction bits (FB). Note that they are visually kinematically identical.

To test the speedup offered by the FPGA, we ran a larger 3D problem. Specifically the cost of constructing 41 subsurface offset gathers from 400 in-line cmps, 400 crossline cmps and 200 frequencies. We compare our FPGA implementation to a 2.8GHz AMD Opteron-based PC with 12 GB of RAM. The software was written in C and compiled using gcc and the Intel C compiler. The FPGA accelerator was implemented on a Maxeler MAX1 FPGA platform equipped with a Xilinx Virtex-4 FX100 FPGA. The accelerator circuit consumes 58% of the logic resources of the device and runs at 125MHz.

[Table 1](#) shows the resulting run-times for the subsurface offset computation at a single depth and shot, comparing software and FPGA. Using 16-bit data, the FPGA is 35-42x faster than software.

ny	$T_{sw}$	$T_{fpga32}$	Speed-up	$T_{fpga16}$	Speed-up
1	0.041	0.002	<b>21x</b>	0.001	<b>41x</b>
50	1.48	0.073	<b>20x</b>	0.042	<b>35x</b>
100	2.76	0.149	<b>19x</b>	0.075	<b>37x</b>
200	6.40	0.311	<b>21x</b>	0.150	<b>42x</b>

Table 1: Performance comparison of FPGA and 2.8GHz AMD Opteron.  $T_{sw}$  is time in seconds for the software version.  $T_{fpga32}$  is for the FPGA processing 32-bit data,  $T_{fpga16}$  is for the FPGA processing 16-bit data.