Implementing Efficient Split-Radix FFTs in FPGAs

- Radix-2 and Radix-4 FFTs are common
- Many applications benefit from other lengths
- OFDM Transceiver, Digital Video Broadcasts, and software defined radios often require FFTs that aren’t a radix-2 or radix-4 length
- Split-radix simplifies the logic for these applications
- Common Split-radix algorithms are Cooley-Tukey, Kolbe-Parks, and Good-Thomas
Split-Radix FFTs

- Split-radix refers to combinations of two (or more) FFT engines
- Split-radix FFTs have a similar structure to 2D FFTs
- Split-radix FFTs provide bin spacing that produce better results for many applications
- Two split-radix approaches employed by DE:
  - Serial (traditional) – lower performance, higher memory requirements by using serial versions of both FFTs
  - Parallel – higher performance by placing larger radix FFTs in parallel and using a parallel version of the smaller radix FFT
- Parallel version mainly used when combining a larger FFT with a 3 or 5 point FFT, since it is feasible to use 3 or 5 large FFTs in a single device
Traditional Serial Split-Radix Approach

*Continuous data FFTs require enough memory to store two full copies of the data for each re-order stage.
Serial 768-Point Split-Radix FFT

0, 1, 2, …767

Data Re-order

P-Point FFT
(256-Point)

Data Re-order

Q-Point FFT
(3-Point)

Data Re-order

0, 1, 2, …767
Serial 768-Point Split-Radix FFT (cont.)

- Single engine of each radix (256-point FFT followed by 3-point FFT)
- Lower device utilization, with performance suitable for most applications
- High memory requirements for data re-ordering
- Speeds up to continuous data, slower data rates require less logic
- Same structure (with external memory) used for ultra-long FFTs
Parallel 768-Point Split-Radix FFT

\[ P\text{-Point} \]
256 FFT

\[ Q\text{-Point} \]
3 FFT (Parallel)

\[ 3 \text{ Point Re-order} \]

0, 3, 6, … 765

1, 4, 7, … 766

2, 5, 8, … 767

Col 0

Col 1

Col 2

A0

A2

A1

0, 1, 2, … 255

256, 257, … 511

512, 513, … 767
Parallel 768-Point Split-Radix FFT Data Flow

3 x 256-pt on columns with rotate

256 x 3-pt FFT on rows with rotate
Parallel 768-Point Split-Radix FFT (cont.)

- Combines 256-point FFT with 3-point FFT
  - 3 x 256-point FFT executions
  - 256 x 3-point FFT executions
- Eliminates the need for intermediate memory
- Higher resource (logic) usage as more computations are performed in parallel
- Very high performance – perform a new 768-point FFT every 256 clock cycles (1.7μS @ 150 MHz)
# Virtex II Performance

<table>
<thead>
<tr>
<th>Type</th>
<th>Number of Butterflies</th>
<th>Latency (uS)</th>
<th>FFT Rate (uS)</th>
<th>Sizes</th>
<th>Block RAM</th>
<th>Multipliers</th>
<th>Power (mW)</th>
<th>Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial</td>
<td>1</td>
<td>20.67</td>
<td>20.42</td>
<td>3,562</td>
<td>12</td>
<td>4</td>
<td>756</td>
<td>290</td>
</tr>
<tr>
<td>Serial</td>
<td>4</td>
<td>15.67</td>
<td>5.12</td>
<td>5,224</td>
<td>18</td>
<td>16</td>
<td>969</td>
<td>500</td>
</tr>
<tr>
<td>Parallel</td>
<td>3</td>
<td>6.96</td>
<td>6.83</td>
<td>4,180</td>
<td>24</td>
<td>12</td>
<td>974</td>
<td>350</td>
</tr>
<tr>
<td>Parallel</td>
<td>6</td>
<td>3.54</td>
<td>3.41</td>
<td>6,260</td>
<td>45</td>
<td>24</td>
<td>1,331</td>
<td>700</td>
</tr>
<tr>
<td>Parallel</td>
<td>12</td>
<td>1.84</td>
<td>1.70</td>
<td>11,123</td>
<td>75</td>
<td>48</td>
<td>1,840</td>
<td>1,400</td>
</tr>
</tbody>
</table>

- **Latency:** Time from last point in to first out
- **FFT rate:** Rate to input FFT data sets
- **Power:** Estimate via Xilinx XPower
- **Cost:** Based on single piece XC2V3000-6 from Partminer.com
Other Dillon Engineering Resources

- ParaCore Architect (parameterized core builder)
- DSP Algorithms
  - Ultra-long FFTs (2k x 2k = 4M points)
  - 2D FFTs for image processing
  - Fixed or floating-point FFTs
  - Floating point math library
- System level DSP
  - OFDM Transceivers
  - Radar Processing on single FPGA
  - Image Compression/Processing
- FPGA-based DSP development platforms
- Hardware/Software SOC
  - High speed Ethernet Appliances
  - Linux Based SOC in FPGA
  - MicroBlaze application