FPGA Design in the cloud – Basics to Optimization

Presented By

PLUNIFY

Name: Kirvy Teo
Title: Founder, COO
Date: 2nd October 2018
Agenda

> About Plunify
  >> Xilinx and Plunify Partners in the Cloud

> First-time Cloud Users

> Tools for FPGA Design in the Cloud
  >> AI Lab
  >> FPGA Expansion Pack

> Optimization in the Cloud
  >> What is InTime
  >> Methodology
  >> Results / Case Study

> Q&A
About Plunify

> Started in 2009
> Founders: Harnhua Ng and Kirvy Teo
> Goal: Optimize FPGA design performance
Xilinx & Plunify Cloud Partnership

> Drive effective use of the cloud for FPGA design and performance optimization

> Link Xilinx and Plunify tools and flows
  >> Fully-featured licenses. This includes Vivado, SDAccel and SDSoC.
  >> On-demand
  >> Unlimited quantities

100 hours x 1 copy of Vivado == 1 hour x 100 copies of Vivado

Considerations for first-time cloud users

1. **What types of workloads are best?**
   - Supermarket (on-demand) versus Fridge (on-premise)
   - Interactive or batch mode
   - **Suggestion:** Use only for on-demand

2. **Understand costs involved**
   - Servers (CPU / RAM capacity, Spot/On-demand)
   - Storage (IO speed / Capacity)
   - Network Bandwidth *(Hidden! – 30% of your costs. Netlists can be huge)*
   - People (You will need experts, DevOps for maintenance)
Considerations for first-time cloud users

3. Do-it-all-yourself or use a managed service?
   ▶️ Burden of managing the cloud falls on you
   ▶️ Or use a 3rd-party solution like Plunify Cloud

4. What may cause unnecessary grief or easily get overlooked
   ▶️ User Experience
     – Required operations: Users authentication, start servers, mount tool volumes, transfer project files, run builds, retrieve results etc. involve many steps
   ▶️ Automation
     – Convert build scripts, backup, testing
   ▶️ Policies & license management
     – Where to host your licenses, how to secure data transfers, data retention
Cloud Basics:
How to build your design (easily)
Plunify Cloud Platform & Tools

- Seamless and fuss-free FPGA design experience with the cloud
- Provides automation, ease-of-use and cloud maintenance
FPGA Expansion Pack

Fully integrated with your Vivado tools.

FPGA Expansion Pack

GUI Mode

Tcl Mode (Project Mode)

```tcl
set design_list [list "A/A.xpr" "B/B.xpr" "C/C.xpr" "D/D.xpr" "E/E.xpr"]
foreach each_design $design_list {
    open_project "$each_design"
    reset_run synth_1
    launch_runs synth_1 impl_1
    wait_on_run impl_1
    close_project
}
```

```tcl
foreach each_design $design_list {
    fcp::runCloudCompile -project "$each_design" -serverclass 3
}
```

https://www.youtube.com/watch?time_continue=31&v=7dN9iRozzT8
## Why use FPGA Expansion Pack?

<table>
<thead>
<tr>
<th>Features</th>
<th>Benefits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Launch builds directly into the cloud.</td>
<td>Reduce compile time with higher performance servers.</td>
</tr>
<tr>
<td>Instant access to additional cloud resources</td>
<td>On-demand scalability.</td>
</tr>
<tr>
<td>Use F1 instances</td>
<td></td>
</tr>
<tr>
<td>Integrated with Vivado</td>
<td>No learning curve.</td>
</tr>
<tr>
<td></td>
<td>Familiar and easy to use.</td>
</tr>
<tr>
<td>Data transmission and storage security.</td>
<td>No cloud knowledge or management required</td>
</tr>
<tr>
<td>Cloud infrastructure for the tool version, licenses, billing and usage.</td>
<td></td>
</tr>
<tr>
<td>Optimization in the cloud (with InTime)</td>
<td>Better Quality of Results</td>
</tr>
<tr>
<td>Get results back in 1-7 days.*</td>
<td>Timing Optimization</td>
</tr>
</tbody>
</table>

**Use cases:** Run regressions, accessing F1 instances, resource crunch
AI Lab

- Web Browser Interface
- Launch a virtual desktop pre-loaded with FPGA tools on the cloud
- Access remote desktop with a browser and WiFi connection.
- No tool installation required. Instant setup.

AI Lab – Vivado on Chromebook (And Ultra96)

https://www.youtube.com/watch?v=xufZF--zzO0
Why use AI Lab

<table>
<thead>
<tr>
<th>Features</th>
<th>Benefits</th>
</tr>
</thead>
<tbody>
<tr>
<td>No installation and setup.</td>
<td>Ease of Use. No downloads required.</td>
</tr>
<tr>
<td>Works on any computer with a browser</td>
<td>Greater access to Xilinx tools. No restrictions on OS.</td>
</tr>
<tr>
<td>Converts IT to an operating expenditure.</td>
<td>More accurate forecasts.</td>
</tr>
<tr>
<td>Eliminates capital expenditure and on-premise</td>
<td>Scales based on actual demand.</td>
</tr>
<tr>
<td>maintenance.</td>
<td>No maintenance required.</td>
</tr>
</tbody>
</table>

Use cases: Run an evaluation, test a new or ancient version of the tools, training / education
InTime – Timing Closure and Optimization

> Machine Learning Optimizes FPGA
  >> timing and performance.

> Identifies good settings for synthesis and place-and-route.
  >> Actively learns and improves from many builds

> Integrates UltraFAST timing closure techniques recommended by
  >> Xilinx FAE in a box
  >> 50% better results from the FPGA tools.
Why use InTime on the cloud

- Run compilations concurrently ➡ Significantly reduces turnaround time
- More results (good & bad) ➡ Converge much faster.
Why InTime works (or not)?

- **Works well for congested designs**
  - Solving issues manually takes a long time
  - Finding the right synthesis/placement parameters makes a huge difference in results
  - Also good for multi-die devices

- **Doesn’t work well for heavily floor planned or constrained designs**
  - No freedom or room to optimize
Optimization Methodology

> **Phase 1: Learning Recipes**
  >> Run builds in parallel to save time.
  >> Minimize WNS to continue to Phase 2

> **Phase 2: Last-Mile Recipes**
  >> Works better with good results from Phase 1
  >> Placement (InTime brought “seed-like” effects back!)
  >> Iterative optimization

> **When should you give up?**
  >> Rule of thumb: At least 50% off the best WNS from “Hotstart”

> **Tip!**
  >> Use post-place timing to estimate final results and save 50% runtime(!)

InTime Case Study (xcvu190)

- **Requirements:** Go below -300ps.
- **Server Type:** 4 CPU, 31 Gb RAM
- **Average Cloud Hours / Project:** 957 hours

<table>
<thead>
<tr>
<th>Projects</th>
<th>TNS (ns)</th>
<th>WNS (ns)</th>
<th>WHS (ns)</th>
<th>CLB Utilization (%)</th>
<th>DSP</th>
<th>LUT</th>
<th>FF</th>
<th>Mem Blocks (BRAM)</th>
<th>Compilations</th>
<th>TNS (ns)</th>
<th>WNS (ns)</th>
<th>TNS Improvements (%)</th>
<th>WNS Improvements (%)</th>
<th>WHS Improvements (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>-33137.63</td>
<td>-1.38</td>
<td>0.016</td>
<td>88.72</td>
<td>12</td>
<td>63569</td>
<td>65715</td>
<td>1508</td>
<td>127</td>
<td>-1536.53</td>
<td>-0.198</td>
<td>95.36%</td>
<td>85.65%</td>
<td>0.00%</td>
</tr>
<tr>
<td>B1</td>
<td>-99541</td>
<td>-2.689</td>
<td>0.016</td>
<td>86.86</td>
<td>12</td>
<td>63849</td>
<td>65702</td>
<td>1508</td>
<td>37</td>
<td>-276.021</td>
<td>-0.183</td>
<td>99.27%</td>
<td>93.19%</td>
<td>0.00%</td>
</tr>
<tr>
<td>C1</td>
<td>-25112.76</td>
<td>-0.943</td>
<td>0.016</td>
<td>74.26</td>
<td>12</td>
<td>50856</td>
<td>49593</td>
<td>1222</td>
<td>37</td>
<td>-0.633</td>
<td>-0.062</td>
<td>100.00%</td>
<td>92.55%</td>
<td>18.75%</td>
</tr>
<tr>
<td>D1</td>
<td>-4789.12</td>
<td>-0.803</td>
<td>0.016</td>
<td>69.98</td>
<td>12</td>
<td>45667</td>
<td>49908</td>
<td>819</td>
<td>37</td>
<td>-0.002</td>
<td>-0.002</td>
<td>100.00%</td>
<td>99.75%</td>
<td>0.00%</td>
</tr>
<tr>
<td>E1</td>
<td>-123092.5</td>
<td>-9.446</td>
<td>-0.306</td>
<td>60.12</td>
<td>12</td>
<td>41417</td>
<td>39597</td>
<td>976</td>
<td>37</td>
<td>-302.274</td>
<td>-0.283</td>
<td>99.75%</td>
<td>97.00%</td>
<td>109.16%</td>
</tr>
<tr>
<td>F1</td>
<td>-117562.5</td>
<td>-3.299</td>
<td>0.016</td>
<td>58.92</td>
<td>12</td>
<td>40950</td>
<td>43404</td>
<td>554.5</td>
<td>238</td>
<td>-1502.151</td>
<td>-0.27</td>
<td>98.72%</td>
<td>91.82%</td>
<td>0.00%</td>
</tr>
<tr>
<td>G1</td>
<td>-150163</td>
<td>-2.125</td>
<td>0.03</td>
<td>90.84</td>
<td>0</td>
<td>70174</td>
<td>66546</td>
<td>718.5</td>
<td>578</td>
<td>-495.731</td>
<td>-0.269</td>
<td>99.57%</td>
<td>87.34%</td>
<td>0.00%</td>
</tr>
<tr>
<td>A2</td>
<td>-11609.56</td>
<td>-0.731</td>
<td>0.016</td>
<td>89.79</td>
<td>12</td>
<td>63567</td>
<td>65798</td>
<td>1508</td>
<td>256</td>
<td>-1319.127</td>
<td>-0.223</td>
<td>88.64%</td>
<td>68.81%</td>
<td>0.00%</td>
</tr>
<tr>
<td>B2</td>
<td>-6644.22</td>
<td>-0.519</td>
<td>0.016</td>
<td>91.74</td>
<td>12</td>
<td>63537</td>
<td>65793</td>
<td>1508</td>
<td>37</td>
<td>-2306.747</td>
<td>-0.266</td>
<td>65.28%</td>
<td>48.75%</td>
<td>25.00%</td>
</tr>
<tr>
<td>C2</td>
<td>-10229.04</td>
<td>-1.203</td>
<td>0.016</td>
<td>75.71</td>
<td>12</td>
<td>50888</td>
<td>49502</td>
<td>1222</td>
<td>131</td>
<td>-770.995</td>
<td>-0.187</td>
<td>92.46%</td>
<td>84.46%</td>
<td>0.00%</td>
</tr>
<tr>
<td>D2</td>
<td>-7362.09</td>
<td>-1.088</td>
<td>0.016</td>
<td>66.41</td>
<td>12</td>
<td>45693</td>
<td>49864</td>
<td>819</td>
<td>37</td>
<td>-2.137</td>
<td>-0.127</td>
<td>99.97%</td>
<td>88.33%</td>
<td>0.00%</td>
</tr>
<tr>
<td>E2</td>
<td>-32691.9</td>
<td>-9.515</td>
<td>0.016</td>
<td>59.11</td>
<td>12</td>
<td>413997</td>
<td>396108</td>
<td>976</td>
<td>37</td>
<td>-426.359</td>
<td>-0.249</td>
<td>98.70%</td>
<td>97.41%</td>
<td>0.00%</td>
</tr>
<tr>
<td>F2</td>
<td>-5906.046</td>
<td>-1.13</td>
<td>0.016</td>
<td>58.18</td>
<td>12</td>
<td>40571</td>
<td>43391</td>
<td>564.5</td>
<td>105</td>
<td>-38.762</td>
<td>-0.167</td>
<td>99.34%</td>
<td>85.85%</td>
<td>0.00%</td>
</tr>
<tr>
<td>G2</td>
<td>-13299.1</td>
<td>-2.393</td>
<td>0.014</td>
<td>90.21</td>
<td>0</td>
<td>73354</td>
<td>67075</td>
<td>734.5</td>
<td>158</td>
<td>-5531.66</td>
<td>-1.077</td>
<td>58.40%</td>
<td>54.99%</td>
<td>-28.57%</td>
</tr>
<tr>
<td>F1</td>
<td>-5641.436</td>
<td>-0.725</td>
<td>0.016</td>
<td>80.15</td>
<td>12</td>
<td>572470</td>
<td>600091</td>
<td>1154</td>
<td>37</td>
<td>-1124.917</td>
<td>-0.257</td>
<td>80.74%</td>
<td>67.31%</td>
<td>0.00%</td>
</tr>
<tr>
<td>F2</td>
<td>-5018.338</td>
<td>-0.554</td>
<td>0.016</td>
<td>81.64</td>
<td>12</td>
<td>572725</td>
<td>600094</td>
<td>1154</td>
<td>37</td>
<td>-1273.956</td>
<td>-0.293</td>
<td>74.61%</td>
<td>47.11%</td>
<td>0.00%</td>
</tr>
</tbody>
</table>
Case Study Takeaways

9 of the 14 projects required fewer than 40 compilations

> Not every design require machine learning. E.g. you can start with incremental compile.

WNS improvements can be more than 90%

> FMax improvements up to 79.7%

Longest project took about 6 days to meet optimization targets

> Actual Wait Time: 1.32 to 6.24 days
Plunify Cloud is a managed cloud platform solution

- Cloud automation, ease-of-use and tools license and maintenance.

AI Lab - virtual desktop pre-loaded with FPGA tools

- Access with a web browser.

FPGA Expansion Pack enable cloud compile from Vivado

- Launch directly from Vivado or Tcl. No cloud setup.

InTime enables timing optimization in the cloud

- Vivado is capable of massive performance improvements with the right settings
Contact Us

Sign up for an InTime evaluation: https://www.plunify.com/en/free-evaluation/

Sign up for Plunify Cloud https://cloud.plunify.com/register

More information: http://www.plunify.com

kirvy@plunify.com / skype: kirvyteo