Python Productivity for Zynq

Louis Liu
Senior Application Engineer
weli@xilinx.com
Python Productivity for Zynq

Targeting the data center, artificial intelligence, machine learning, data science

New users are not hardware designers, or embedded systems designers

Enable more people to program Xilinx processing platforms, more productively
Productivity Languages & Hardware Overlays

Zynq / Zynq UltraScale+

Users:
Apps Programmers:
Programmers(embedded):
Device driver writers:
Kernel developers:

Embedded Applications Programmers
Python
C/C++
Overlays

Small group of experts create APSoC overlays and C API/drivers
Many more users build applications in C/Python
Python is increasingly the Language of Choice

Top Programming Languages, IEEE Spectrum, July’18

Python is listed as an embedded language for the first time


To date

https://stackoverflow.blog/2017/09/06/incredible-growth-python/

Python is the fastest growing language: driven by data science, AI, ML and academia
Python productivity for Zynq

Jupyter notebooks/lab, browser-based interface

PYNQ enables Jupyter on Zynq and ZU+

Hardware drivers wrapped in Python libraries

Jupyter web server

IPython kernel

Ubuntu-based Linux

Overlays/designs

PS: ARM

PL: Fabric

Optimized for developer productivity

- All the Linux libraries and drivers you expect
- Pre-built SD image
- Ubuntu/Debian ecosystem & community
- 110,000,000 Google hits
Jupyter Notebooks to JupyterLab IDE

Code editor

Terminal

Jupyter notebooks

Visualization

2017 ACM
Software System Award

Jupyter ... Julia, Python, R

Default engine of data science

Taught to 1,000+ Berkeley students every semester

2+ million notebooks on GitHub

Next-gen browser IDE

Includes Jupyter Notebooks
PYNQ’s Ubuntu-based Linux

PYNQ uses Ubuntu’s:
• Root file system (RFS)
• Package manager (apt-get)
• Repositories

PYNQ bundles:
• Development tools
• Cross-compilers
• Latest Python packages

PYNQ uses the PetaLinux build flow and board support package:
• Access to all Xilinx kernel patches
• Works with any Xilinx supported board
• Configured with additional drivers for PS-PL interfaces
Ubuntu-based Linux versus embedded Linux

**Ubuntu-based Linux**

- Optimized for developer productivity
  - All the Linux libraries and drivers you expect
  - Pre-built SD card image
  - Ubuntu/Debian ecosystem & community
  - 145,000,000 Google hits

**Embedded Linux**

- Optimized for deployment efficiency
  - Selective Linux libraries and drivers
  - Commonly delivered in flash memory on board
  - PetaLinux ecosystem:
  - 143,000 Google hits

3 orders of magnitude difference
Hundreds Overlays in PYNQ Community
- Need your contributions too
PYNQ provides Linux Drivers for PS-PL Interfaces … wrapped in Python Libraries
How MMIO works

```python
self.mmap_file = os.open('/dev/mem',
                       os.O_RDWR | os.O_SYNC)
```

© Copyright 2019 Xilinx
is a Framework

- The key of productivity is the Unified

- **Apps**
  - Jupyter/IPython
  - PYNQ notebooks
  - matplotlib
  - numpy
  - scikit-learn
  - opencv

- **APIs**
  - Python
  - PYNQ libs
  - video
  - audio
  - axi_gpio
  - dma
  - pynqmicroblaze
  - XLINK
  - Overlay
  - PL
  - GPIO
  - Interrupt
  - MMIO
  - libsdss.so

- **Drivers**
  - Linux kernel
  - xdevcfg
  - sysgpio
  - uio
  - devmem
  - xlnk
  - axi_intc

- **Bitstreams**
  - FPGA
  - User designs
  - PYNQ IPs
  - PYNQ overlays

- **Hardware**
  - VIVADO
  - ARM
  - SDSoc Environment

- **Software**
  - Jupyter/IPython
  - Python
  - matplotlib
  - numpy
  - scikit-learn
  - opencv
  - VIDEO
  - AUDIO
  - PYNQ notebooks
  - PYNQ/IPs
  - PYNQ overlays
Software-style packaging & distribution of designs
Enabled by new hybrid libraries

Download a design from GitHub with a single Python command:

```
pip install git+https://github.com/Xilinx/pynqDL.git
```
Loading a design into Zynq using PYNQ

```python
from pynq import Overlay
resizer = Overlay('./resizer.bit')
```

PYNQ automatically configures many design parameters based on data parsed from hybrid library.
Realtime and Graphic analysis
How Python helps, really a lot..
Ecosystem Advantage: there’s a Library for that...

Standard Python comes with comprehensive libraries for common operations (web, regex, os, etc). In addition to this ‘batteries included’ strategy, there is a massive external ecosystem …

CPython is written in C ... and most popular C/C++ frameworks have Python libraries
Base Python libraries used for all Use Case

- NumPy: A Matlab™ like framework for numerical computing.
- Matplotlib: 2D plotting library for static and interactive data visualizations.
- Pandas: Data wrangling for easy-to-use data ingestion, transformation, and export functions.

Acquire, Transform, Organize, Display
What is NumPy

> NumPy is the fundamental package for scientific computing with Python. It contains among other things:
  >> a powerful N-dimensional array object
  >> sophisticated (broadcasting) functions
  >> tools for integrating C/C++ and Fortran code
  >> useful linear algebra, Fourier transform

> NumPy can also be used as:
  >> An efficient multi-dimensional container of generic data.
  >> Arbitrary data-types can be defined.
  >> This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.
How Numpy interacts with Programmable Logic?

**NumPy**

- Array, Matrix
- Virtual memory
- Contiguous Array in DDR Memory
- Shared Object
- Physical memory

**PYNQ**

**Xlnk**

**Hard Real Time**

- Offload CPU and OS
- Dedicated “Accelerator”

**Contiguous Array in DDR Memory**

**Physical memory**

**Virtual memory**

© Copyright 2019 Xilinx
Provide a Numpy Array (or Arrays) with collected data
Or Pandas structure

This notebook makes the Exploratory Data Analysis:

We acquire the Data
We display the Data
We process the Data

```python
# It is the Numpy Array

<table>
<thead>
<tr>
<th>ID</th>
<th>cycle</th>
<th>Setup1</th>
<th>Setup2</th>
<th>la_mA</th>
<th>lb_mA</th>
<th>Total_Current</th>
<th>s4</th>
<th>s5</th>
<th>s6</th>
<th>s7</th>
<th>s8</th>
<th>s9</th>
<th>Acc</th>
<th>Freq</th>
<th>ttf</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>-0.0007</td>
<td>-0.0004</td>
<td>100</td>
<td>518.67</td>
<td>641.82</td>
<td>1598.70</td>
<td>1400.60</td>
<td>14.62</td>
<td>21.61</td>
<td>554.36</td>
<td>2388.05</td>
<td>9046.19</td>
<td>1.3</td>
<td>47.47</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.0019</td>
<td>-0.0003</td>
<td>100</td>
<td>518.67</td>
<td>642.15</td>
<td>1591.82</td>
<td>1403.14</td>
<td>14.62</td>
<td>21.61</td>
<td>553.75</td>
<td>2388.04</td>
<td>9044.07</td>
<td>1.3</td>
<td>47.49</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>-0.0043</td>
<td>0.0003</td>
<td>100</td>
<td>518.67</td>
<td>642.35</td>
<td>1587.99</td>
<td>1404.20</td>
<td>14.62</td>
<td>21.61</td>
<td>554.26</td>
<td>2388.09</td>
<td>9052.94</td>
<td>1.3</td>
<td>47.27</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.0007</td>
<td>0.0000</td>
<td>100</td>
<td>518.67</td>
<td>642.35</td>
<td>1582.79</td>
<td>1401.87</td>
<td>14.62</td>
<td>21.61</td>
<td>554.45</td>
<td>2388.11</td>
<td>9049.48</td>
<td>1.3</td>
<td>47.13</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>-0.0019</td>
<td>-0.0002</td>
<td>100</td>
<td>518.67</td>
<td>642.37</td>
<td>1582.85</td>
<td>1400.22</td>
<td>14.62</td>
<td>21.61</td>
<td>554.00</td>
<td>2388.09</td>
<td>9055.15</td>
<td>1.3</td>
<td>47.28</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>-0.0043</td>
<td>-0.0001</td>
<td>100</td>
<td>518.67</td>
<td>642.10</td>
<td>1584.47</td>
<td>1398.37</td>
<td>14.62</td>
<td>21.61</td>
<td>554.67</td>
<td>2388.02</td>
<td>9049.68</td>
<td>1.3</td>
<td>47.16</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>0.0010</td>
<td>0.0000</td>
<td>100</td>
<td>518.67</td>
<td>642.40</td>
<td>1592.32</td>
<td>1397.77</td>
<td>14.62</td>
<td>21.61</td>
<td>554.34</td>
<td>2388.02</td>
<td>9059.13</td>
<td>1.3</td>
<td>47.35</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>-0.0034</td>
<td>0.0003</td>
<td>100</td>
<td>518.67</td>
<td>642.56</td>
<td>1592.96</td>
<td>1400.97</td>
<td>14.62</td>
<td>21.61</td>
<td>553.85</td>
<td>2388.00</td>
<td>9040.80</td>
<td>1.3</td>
<td>47.24</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>0.0008</td>
<td>0.0001</td>
<td>100</td>
<td>518.67</td>
<td>642.12</td>
<td>1590.96</td>
<td>1394.80</td>
<td>14.62</td>
<td>21.61</td>
<td>553.69</td>
<td>2388.05</td>
<td>9048.46</td>
<td>1.3</td>
<td>47.29</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>-0.0033</td>
<td>0.0001</td>
<td>100</td>
<td>518.67</td>
<td>641.71</td>
<td>1591.24</td>
<td>1400.46</td>
<td>14.62</td>
<td>21.61</td>
<td>553.59</td>
<td>2388.05</td>
<td>9051.70</td>
<td>1.3</td>
<td>47.03</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>0.0018</td>
<td>-0.0003</td>
<td>100</td>
<td>518.67</td>
<td>642.20</td>
<td>1581.75</td>
<td>1400.64</td>
<td>14.62</td>
<td>21.61</td>
<td>554.54</td>
<td>2388.05</td>
<td>9049.61</td>
<td>1.3</td>
<td>47.15</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>0.0016</td>
<td>0.0002</td>
<td>100</td>
<td>518.67</td>
<td>642.06</td>
<td>1583.41</td>
<td>1400.15</td>
<td>14.62</td>
<td>21.61</td>
<td>554.52</td>
<td>2388.09</td>
<td>9049.37</td>
<td>1.3</td>
<td>47.18</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>-0.0019</td>
<td>0.0004</td>
<td>100</td>
<td>518.67</td>
<td>643.07</td>
<td>1582.19</td>
<td>1400.83</td>
<td>14.62</td>
<td>21.61</td>
<td>553.44</td>
<td>2388.12</td>
<td>9046.62</td>
<td>1.3</td>
<td>47.38</td>
</tr>
</tbody>
</table>
```

Your Job is done
Customer can take from it
Connect other Python Libraries -

- pandas
- scikit-learn
- NumPy
- Bullet Physics Library
- ROS.org
- SimPy

Quick Prototyping

Embedded Acceleration

RTL Acceleration

OVERLAYS

- Data analysis
- Statistical machine learning
- Sensors
- Robots
- Discrete event simulation

© Copyright 2019 Xilinx
Edge-to-cloud co-design

Common JupyterLab tooling at edge and cloud

PYNQ enables ML experts and radio engineers to focus on their ‘value-add’

Edge-to-cloud co-design trade-offs:
- Maximize on-chip processing
- Minimize edge-to-cloud data exchange
- Exploit scalability of cloud processing
- Aggregate intelligence between and across multiple edge nodes
- Co-optimize the above for best system performance
AWS IoT Greengrass (Base on MQTT)
RF_QPSK Demo
Student contest designs base on PYNQ
More and more are on the way
PYNQ-Z2 Board - design for starter

- New PYNQ reference platform
- New stereo audio with on-board codec
- New Raspberry Pi connector
- Open source design
- Z2 manufactured in Taiwan by TUL
- Distributed globally by Premier Farnell
- Also Newegg in US
- Academic discounts & donations available

$119 to everyone in US
PYNQ on other Boards
Next steps: scaling across platforms and domains
Efficient porting PYNQ to any Zynq-based platform

1. PetaLinux build tools/BSP
   - Available for any Xilinx-supported board;
   - Instructions for building for custom boards available

2. PYNQ Software Core (board independent)
   - Pre-built images available
   - Expected to take less than 1 day to port

3. Target-specific PYNQ components
   - Board-specific porting
   - Common interfaces available for re-use
http://www.pynq.io
New open source HLS book

Parallel Programming for FPGAs is an open-source book aimed at teaching hardware and software developers how to efficiently program FPGAs using high-level synthesis.

Reply “pp4fpgas” in wechat console

http://kastner.ucsd.edu/hlsbook/
Adaptable. Intelligent.