Open Source Building Blocks

Rob Oshana
Vice President, Software Engineering R&D
NXP Microcontroller
robert.oshana@nxp.com

January 2019
Software enablement for mass market

- 25,000 unique customers
- MCU → in between → MPU, the entire gamut
- MPU: Linux Yocto then optimize for verticals, additional distros for OOBE
- MCU: SDK & tools
- Software scalability/reuse across platforms critical
- Open source is key to enabling this
- Landscape changing quickly, need to react quickly
- Leverage open source and Linaro as an enabler and accelerator for mass market business
- Let the community do the work!

Significant revenue from legacy devices are supported entirely by the community
Example: The AI / IoT Era – a new landscape

"The 4th Tectonic Shift in Computing"

- Edge Processing
- Machine Learning
- Security
- Connected Data

Units Shipped (Millions)

- Mainframe
- Mini-Computer
- PCs
- Cell Phones
- AI / IoT

1. Source: Jefferies
ML requires Edge Computing Infrastructure (Cloud/Gateway/Endpoint)

Endpoints
- i.MX / Kinetis® / LPC Processors / Connectivity

Edge Gateways
- Layerscape, i.MX Processors
- HOME GATEWAY
- ETHERNET SWITCH
- WIRELESS ROUTER
- INDUSTRIAL CONTROLLER

Cloud Infrastructure
- NXP Cloud Software Platform built on top of Azure/AWS/GCP & partners

Customer Solutions
- SMART CITY
- INDUSTRY 4.0
- CONNECTED VEHICLE
- SMART HOME
- SMART RETAIL

Application Development Environment
- Vision, Voice libraries, toolkits
- Distributed AI & Machine Learning Platform
- Cloud-based Build, Deploy, Test Environment

Secure Application Management

Secure Device Management

Customer Solution
- App
- App
- App

Application Framework
- Linux Platform
- Secure Device Mgmt.

Layerscape, i.MX Platforms
Create a scalable software approach leveraging OSS

1. Establish and maintain a base of upstreamed software

2. Architect software to support open source components

3. Leverage open source components as needed and focus on integration, optimization, and differentiation, become more of an open source software integrator
1. Establish a base of upstreamed enablement
NXP Upstream Work

- ~350 contributors to open source project in the last decade
- Currently 23 maintainers, maintaining 34 modules in kernel, and 25 maintainers maintaining 96 boards in Uboot
- Upstreamed 13800+ patches, ~1.4Million LoC in Kernel/Uboot
  - 8000+ patches, 630k LoC in Linux Kernel
  - 5800+ patches, 740K LoC in Uboot
- 90+ SoCs and 380+ boards support in kernel
- Contributing on other OpenSource Projects (ATF, OPTEE, GStreamer, Android)

<table>
<thead>
<tr>
<th>Other Open Source Project</th>
<th>Patch</th>
<th>LoC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATF</td>
<td>41</td>
<td>16616</td>
</tr>
<tr>
<td>OPTEE</td>
<td>100</td>
<td>8849</td>
</tr>
<tr>
<td>Android</td>
<td>~60</td>
<td>~400</td>
</tr>
<tr>
<td>GStreamer</td>
<td>74</td>
<td>7014</td>
</tr>
</tbody>
</table>
2. Architect software to support open source components
Architect Software to support quick open source integration
Complement commercial with open source components

Development Effort

Complexity

Open Source

Google: Blocky

Micropython

Zephyr

Green Hills

IAR Systems

CFG

IDE

SDK

MCUXpresso C/C++

Commercial

LittlevGL

LLVM
Open Source MPU Platform

- Scalable, portable open source platform
- Based on Zephyr platform
- Focused on ease of use and “plug and play” compatible software components
- Rapid iterative prototyping with Micropython
- Seamless compatibility with internal tools
- Software frameworks enable power management, sensor streaming, etc
- Compact and performant AI/ML with TF Lite
- High quality HMI graphics and UI
3. Leverage open source components
## MCU → Crossover → MPU

### MCU
- **8/16/32-bit core**
- **On-chip SRAM**
- **Embedded flash**
- **Mixed signal analog**

### Wired Connectivity
- USB 1.0/2.0
- UART
- SDIO

### i.MX RT
- **32-bit core(s)**
- **On-chip SRAM**
- **Next-gen non-volatile memory**
- **Embedded secure sub-system**
- **2D graphics**
- **Audio Codec**
- **Embedded DSP**
- **Bluetooth, ZigBee ...**

### i.MX
- **64-bit cores**
- **On-chip SRAM**
- **3D + 2D graphics**
- **4K HDR Display**
- **Audio / Video Codec**
- **Embedded DSP**

### 32-bit cores
- **Embedded secure sub-system**
- **Mixed Signal Analog**
- **Audio Codec**
- **Embedded DSP**

### High-speed connectivity
- **USB 3.0, TSN, QuadSPI, ...**

---

- **Millions** of transistors
- Bulk silicon & flash technologies
- Older technology nodes

- **Billions** of transistors
- High-K Metal Gate, FD-SOI, FinFET
- 40/28nm, 16/14nm & smaller nodes
Breaking the ‘GHz’ Barrier – i.MX RT1100 MCUs
CoreMark® > 5000

- GHz Performance
- Auto & Industrial Grade
- Secure Boot, PUF
- On-the-fly Crypto Tamper Detect
- Low Power 28 FD-SOI
- TSN, Hi-Perf Analog

- Cortex-M7: Up to 1GHz
- Cortex-M4: Up to 400MHz
- Secure Resource Controller

- Overdrive
- Voltage
- Underdrive

1 GHz
Extreme Real-Time Response with i.MX RT & Zephyr OS

High performance with secure, open-source, comprehensive RTOS (e.g. Zephyr) will enable IoT systems of the future.

- A7 @ 600MHz (Linux)
- i.MX RT @ 600MHz (Zephyr RTOS)
- i.MX RT @ 1GHz (est.)

Faster / higher is better.
Example: Machine Learning at the Edge
Enabling Machine Learning Revolution
System cost vs. End-user Experience - One size does not fit all

**Training in the cloud**

- **TensorFlow**
- **Caffe2**
- **ONNX**
- **Keras**
- and others

**Low-end Edge Compute**
- i.MX 6
  - 1-2x Arm 32-bit
- Crossover & High. Perf. MCUs
  - Arm 32-bit MCU + ML DSP

**Mid-end Edge Compute**
- i.MX 7, 8
  - 1-4x Arm 32/64-bit Performance GPUs
  - Integrated DSP

**High-end Edge Compute**
- i.MX 8, Layerscape
  - Multi-core Arm 64-bit
  - High performance
    - GPUs
    - DSP
    - Vector processing

Scalable & optimized inference engines across Embedded Processing continuum

**Trained Models Inference Engines**
Inference Performance Across CPU & GPU Complexes

### Inference Speed
- **Faster**
- **Slower**

### Inference Performance Across CPU & GPU Complexes

<table>
<thead>
<tr>
<th>Model Profiled</th>
<th>Core</th>
<th>MCUs</th>
<th>CPUs</th>
<th>GPUs</th>
<th>Accelerators</th>
</tr>
</thead>
<tbody>
<tr>
<td>ModelTooLarge</td>
<td>M7</td>
<td>A7</td>
<td>A9</td>
<td>A35</td>
<td>AS3</td>
</tr>
<tr>
<td>MobileNetV1.0</td>
<td>1020</td>
<td>0.212</td>
<td>0.129</td>
<td>0.085</td>
<td>0.045</td>
</tr>
<tr>
<td>MobileNetV1.1</td>
<td>1220</td>
<td>0.274</td>
<td>0.154</td>
<td>0.113</td>
<td>0.073</td>
</tr>
<tr>
<td>MobileNetV1.2</td>
<td>1420</td>
<td>0.319</td>
<td>0.192</td>
<td>0.143</td>
<td>0.101</td>
</tr>
<tr>
<td>MobileNetV1.3</td>
<td>1620</td>
<td>0.353</td>
<td>0.225</td>
<td>0.174</td>
<td>0.129</td>
</tr>
<tr>
<td>MobileNetV1.4</td>
<td>1820</td>
<td>0.387</td>
<td>0.258</td>
<td>0.219</td>
<td>0.178</td>
</tr>
<tr>
<td>MobileNetV1.5</td>
<td>2020</td>
<td>0.421</td>
<td>0.291</td>
<td>0.255</td>
<td>0.220</td>
</tr>
<tr>
<td>MobileNetV1.6</td>
<td>2220</td>
<td>0.454</td>
<td>0.324</td>
<td>0.299</td>
<td>0.260</td>
</tr>
<tr>
<td>MobileNetV1.7</td>
<td>2420</td>
<td>0.488</td>
<td>0.357</td>
<td>0.339</td>
<td>0.305</td>
</tr>
</tbody>
</table>

### Work in Progress
- Proprietary models (Work in Progress)

### Preliminary Results
- **Preliminary** Results from Au-Zone Technologies, Inc.

**As of Mar 8, 2019**

**Float32 Models**

**Artificial Intelligence**

**NXP**
Key building blocks for ML

“Hidden Technical Debt” (google)

ML is a combination of tools, platforms, frameworks, libraries, languages, and hardware technology.
Build a Lego brick ML castle!
Cloud ML Development (think Matlab)

26262, Security

GUI, Cookbooks, and Developer Experience

OpenCL, LLVM, metal

ML Components (OSS)

Processing elements

Application (App)
TF runtime
Arm NN
ACL
Opt Libraries (Opt Lbs)
Framework (Frmk)
Optimizer (Opt)
Libraries (Libs)

A core
R core
M core
GPU
DSP
NNA

Cloud development and deployment

Cross-cutting concerns

HW back ends
MCU ML Acceleration (Co-Processors & DSP) Extensions

### LPC5500 MCU:
Dual-core Cortex-M33 with PowerQuad co-processor and 640KB Flash + 320KB SRAM

### i.MX RT600 Crossover
- Cortex-M33
- 128KB TCM Up to 4.5 MB RAM
- Arm Co-processor Interface
- Security
- PowerQuad
- NXP co-processors
- Tensilica HiFi 4
- DSP core with TIE

### Extensions
- **TensorFlow Lite**
- **Arm CMSIS-NN**
- **eIQ-CML with OpenCV for classical ML algorithms**

- **Arm v8-M feature** - tightly couples co-processor to the M33 core
- **Faster execution than instruction extensions**
- **Maintains ecosystem and toolchain compatibility**

- **NXP-designed accelerator for DSP Functions**
- **5-10x faster for Matrix, FIR, Convolution, Correlation ops**
- **15x faster running CMSIS-DSP library for FFT / IFFT**

- **DSP Tensilica HiFi 4 @ 600MHz for Natural Language**
- **Tensilica Instruction Extension (TIE) for Sigmoid transfer function – accelerating wake-word**
MPU ML Acceleration → iMX8M w/ A/M/GPU

- OpenCL optimized libraries for ML or
- Application processing in parallel with ML

- OpenCV support accelerated on NEON
  - TensorFlow and Caffe
  - Classical machine learning algorithms
  - Cloud deployment using Docker

- Other open source options
  - Arm NN w/NEON acceleration using Arm Compute Library (ACL)
  - Android NN
  - TensorFlow, TF Lite (direct deployment)
    --ACL for image segmentation, feature detection/ extraction, image processing, etc.

- Sensor integration (e.g. anomaly detection)
  - M4 manages sensor reading/fusion, feature extraction
  - Then use RPmsg/openAMP to send data to the A53 for inferencing
  - Leverage sensor integration libraries
Next i.MX 8M Applications Processor with Machine Learning

- **Machine Learning**
- **Voice & Natural Language**
- **Audio & Video**
- **3D Graphics**

### Key Features:
- **LPDDR4-4000**
- **3D / 2D Graphics (GC7000UL & GC520)**
- **4x Cortex-A53 up to 1.8GHz**
- **HiFi 4 DSP Audio sub-system**
- **HDMI 2.0**
- **Cortex-M7**
- **TSN CAN-FD**
- **VPU**
- **NNA 2TOPS/s**
- **Image Processing**
- **3D / 2D Graphics (GC7000UL & GC520)**
- **4x Cortex-A53 up to 1.8GHz**
- **HiFi 4 DSP Audio sub-system**
- **HDMI 2.0**
- **Cortex-M7**
- **TSN CAN-FD**
- **VPU**
- **NNA 2TOPS/s**
- **Image Processing**
Google's Raspberry Pi-like Coral board lands: Turbo-charged AI on a tiny computer

Google's own developer board helps developers bring machine-learning models to the network edge.

Includes a full system
SOC + ML + Connectivity all on the board running a derivative of Debian Linux we call Mendel, so you can run your favourite Linux tools with this board.

Supports TFLite
No need to build models from the ground up. TensorFlow Lite models can be compiled to run on the Coral Dev Board.

Scale from prototype to production
Considers your manufacturing needs. The SOM can be removed from the baseboard, ordered in bulk, and integrated into your hardware.

"Build intelligent ideas with our platform for local AI"

https://coral.withgoogle.com/

Create your own castle
So in summary…

- Software support of mass market with multiple MCU and MPU architectures requires discipline
  1. Establish and maintain a base of upstreamed software
  2. Architect software to support open source components
  3. Leverage open source components as needed and focus on integration, optimization, and differentiation, become more of an open source software integrator

- Markets and technologies changing rapidly

- Benefits are there in terms of R&D efficiency and scalability/reuse
THANK YOU!