Running TF Lite on Microcontrollers without hardware in Renode

Tiny ML Talks, 2020-09-29
Michael Gielda, mgielda@antmicro.com
ANTMICRO

- Founded 2009
- Turning ideas into software-driven products
- Industrial IoT and embedded systems: AI/ML in defense/security, mining, agriculture, autonomous vehicles, robotics, aerospace, industrial automation
- We use, develop, advocate open source
- Member of Linux Foundation, Zephyr Project, CHIPS Alliance, OpenPOWER Foundation, Strategic Founding member of RISC-V International
- Introducing new design methodologies and workflows based on open source
WHAT WE DO

• See our technology showcase on antmicro.com

Antmicro joins the OpenPOWER Foundation - Q&A
SweRV and open tooling
Jetson Nano / Xavier NX with 10Gb Ethernet Controller
seL4 on RISC-V in Renode
Plug-and-play AI acceleration with Thunderbolt
Antmicro's open source SDI-to-MIPI Bridge

Google
SkyWater open PDK release
Renode 1.9: new platforms, RISC-V
Antmicro's TX2 platform in the works
Renode 1.10 release
Android on open Jetson
PolarFire SoC Icicle Kit with
WHAT WE DO

HARDWARE
Proof of Concepts (PoC), demonstrators, prototyping, open source platforms

SOFTWARE & AI
OS porting, building BSPs, build systems, device management, edge & cloud AI

FPGA & ASIC
Custom IP blocks, SiP development, soft SoCs, heterogeneous processing systems

TOOLS
Tools, new software and hardware development and testing methodologies
WHAT WE DO

SOFTWARE & AI
OS porting, building BSPs, build systems, device management, edge & cloud AI

FPGA & ASIC
Custom IP blocks, SiP development, soft SoCs, heterogeneous processing systems

TOOLS
Tools, new software and hardware development and testing methodologies

HARDWARE
Proof of Concepts (PoC), demonstrators, prototyping, open source platforms

Running TF Lite on Microcontrollers without hardware in Renode
TINY ML - BIG OPPORTUNITIES

- Trickling down of ML capabilities into smaller, lower-power devices
- Low energy cost and complexity (no wireless communication)
- Higher responsiveness and privacy
- Autonomy of edge devices
- New ML use cases, like:
  - Gesture recognition
  - Voice/sound recognition
  - Anomaly detection
  - Failure alerts, etc.
TINY ML - BIG CHALLENGES

- Power/performance constraints
- Memory constraints
- Testing software
  - Sourcing hardware
  - Testing at scale
  - Configuration of complex systems of devices
  - Tedious manual testing procedures
  - Repeatability/Determinism
NEED FOR TESTING

- Adding embedded and IoT devices into TensorFlow increases need for reliable testing of constrained devices

- **Renode** - Antmicro’s open source simulation framework is being introduced into TensorFlow’s CI to address those challenges with:
  - Hardware-less CI-driven workflows for IoT
  - Full determinism
  - Plug and play peripheral models
  - Easy setup of development environments
  - Quick configuration of multi-node networks of unique devices
GOOGLE AND ANTMICRO COLLABORATION

- Collaboration started in 2018
- Lots of work around open source ASICs, FPGA, software
- Enabled running TF Lite on FPGA and virtual MCUs in Renode
- First integration with Zephyr and co-marketing with Zephyr, RISC-V
- Demo for soft RISC-V MCU or Digilent Arty board
- Our note on TF Lite blog with demo: https://blog.tensorflow.org/2020/06/running-and-testing-tf-lite-on-microcontrollers.html
A guest post by Michael Gielda of Antmicro

Every day more and more software developers are exploring the worlds of machine learning, embedded systems, and the Internet of Things. Perhaps one of the most exciting advances to come out of the most recent innovations in these fields is the incorporation of ML at the edge and into smaller and smaller devices - often referred to as TinyML.

In "The Future of Machine Learning is Tiny", Pete Warden predicted that machine learning would become increasingly available on tiny, low-power devices. Thanks to the work of the TensorFlow community, the power and flexibility of the framework is now also available on fairly resource-constrained microcontrollers.
GOOGLE AND ANTMICRO COLLABORATION

- Collaboration started in 2018
- Lots of work around open source ASICs, FPGA, software
- Enabled running TF Lite on FPGA and virtual MCUs in Renode
- First integration with Zephyr and co-marketing with Zephyr, RISC-V
- Demo for soft RISC-V MCU or Digilent Arty board
- Our note on TF Lite blog with demo:
  https://blog.tensorflow.org/2020/06/running-and-testing-tf-lite-on-microcontrollers.html
TF Lite demo running in LiteX/VexRiscv soft RISC-V SoC on a Digilent Arty board

antmicro.com/blog/2019/12/tf-lite-in-z...
SO WHAT’S THIS RENODE THING?
Develop your IoT product with Renode:
A BIT OF HISTORY

- Simulation framework developed by Antmicro since 2010
- Started as a response to our internal needs
- Open source since 2015
- Current version: 1.10.1
What is Renode?

Open source hardware-simulation framework for:

• Development of complex software for embedded and IoT systems
• Architectural exploration and research
• Pre-silicon prototyping and HW-SW co-development
What is Renode?

Features in brief:

• Plug-and-play building blocks
• Simulates system on many levels - CPUs, SoCs, peripherals, sensors, wired/wireless connection
• Flexible, deterministic and software-agnostic
• Continuous Integration-oriented
• “Batteries included” - lots of demos and binaries
Simulate your system on many levels
Layer #1: System-on-Chip

- CAN
- Ethernet
- RAM
- I2C
- UART
- SPI
- USB
- LCD
- Flash

CPU
- Cortex-A7 Core
- Cortex-M4 Core
Layer #2: The device
Layer #3: Complex system

- Sensor nodes
- Gateway
### Supported platforms/vendors

<table>
<thead>
<tr>
<th>LEON3</th>
<th>ST</th>
<th>QuickLogic</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORDIC</td>
<td>Microchip</td>
<td>NXP</td>
</tr>
<tr>
<td>Silicon Labs</td>
<td>Texas Instruments</td>
<td>ZYNQ</td>
</tr>
<tr>
<td>SiFive</td>
<td>RISC-V</td>
<td>OpenPOWER</td>
</tr>
</tbody>
</table>
Supported boards
Supported boards
Our focus is not just on SoCs, but also I/O peripherals
- UART, SPI, I2C, RAM, ROM, GPIO, CAN, ETH, I2S, PCIE...
Also sensors:
- Thermometers, humidity meters, accelerometers, microphones, etc.
Various integrations

<table>
<thead>
<tr>
<th>GitLab</th>
<th>Wireshark</th>
<th>Robot</th>
<th>Jenkins</th>
</tr>
</thead>
<tbody>
<tr>
<td>VERILATOR</td>
<td>GDB</td>
<td>Verilator</td>
<td>USB/IP</td>
</tr>
</tbody>
</table>
So, what can you do with it?

- Verilator for HW-SW co-development
  - note with Google on our blog
- PlatformIO integration
  - note on Zephyr Project blog
- CI with Robot Framework, Jenkins, GitLab CI, GitHub Actions
  - e.g. note on Memfault’s blog
- protocol / stack testing: OPC-UA, TSN, 6lowpan, Thread etc.
Pretty neat that with @renodeio on my laptop, I can emulate Linux running on a RISC-V processor including a framebuffer to show that adorable Tux 🦩

[link to repository](https://github.com/renode/renode/...)

1:45 AM - Feb 9, 2020 - Twitter Web App

7 Retweets 30 Likes
RENODE - TESTING & METRICS
Running TF Lite on Microcontrollers without hardware in Renode

TRAVIS CI

- performs the whole “demo” procedure automatically
- a demo of how CI with Renode could work (also possible with GitLab CI, Jenkins, GitHub Actions... - we work with all of them)
<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>LiteX_12C_Zephyr.cs</td>
<td>Reduce verbosity of the demo</td>
<td>3 months ago</td>
</tr>
<tr>
<td>angle.data</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>circle.data</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>litex-vexriscv-tflite.repl</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>litex-vexriscv-tflite.resc</td>
<td>README: Use locally-built binaries when running Renode</td>
<td>8 days ago</td>
</tr>
<tr>
<td>litex-vexriscv-tflite.robot</td>
<td>Remove redundant ADXL345 model</td>
<td>3 months ago</td>
</tr>
</tbody>
</table>
mateusz-holenko README: Use locally-built binaries when running Renode

- LiteX_I2C_Zephyr.cs: Reduce verbosity of the demo
- angle.data: Add Renode section
- circle.data: Add Renode section
- litevexriscv-tf-lite.repl: Add Renode section
- litevexriscv-tf-lite.resc: README: Use locally-built binaries when running Renode
- litevexriscv-tf-lite.robot: Remove redundant ADXL345 model
<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>LiteX_I2C_Zephyr.cs</td>
<td>Reduce verbosity of the demo</td>
<td>3 months ago</td>
</tr>
<tr>
<td>circle.data</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>circle.data</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>litevexriscv-tfLite.repl</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>litevexriscv-tfLite.resc</td>
<td>README: Use locally-built binaries when running Renode</td>
<td>8 days ago</td>
</tr>
<tr>
<td>litevexriscv-tfLite.robot</td>
<td>Remove redundant ADXL345 model</td>
<td>3 months ago</td>
</tr>
<tr>
<td>File Name</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>LiteX_I2C_Zephyr.cs</td>
<td>Reduce verbosity of the demo</td>
<td></td>
</tr>
<tr>
<td>angle.data</td>
<td>Add Renode section</td>
<td></td>
</tr>
<tr>
<td>circle.data</td>
<td>Add Renode section</td>
<td></td>
</tr>
<tr>
<td>litex-vexriscv-tfLite.repl</td>
<td>Add Renode section</td>
<td></td>
</tr>
<tr>
<td>litex-vexriscv-tfLite.resc</td>
<td>README: Use locally-built binaries when running Renode</td>
<td></td>
</tr>
<tr>
<td>litex-vexriscv-tfLite.robot</td>
<td>Remove redundant ADXL345 model</td>
<td></td>
</tr>
</tbody>
</table>
Platform description format

- human readable
- modular
- extendible
- enable new boards / platforms w/o coding
uart: UART.MiV_CoreUART @ sysbus 0x70001000
  clockFrequency: 66000000

cpu: CPU.RiscV @ sysbus
  cpuType: "rv32g"

plic: Interrupts.PlatformLevelInterruptController @ sysbus 0x40000000
  IRQ -> cpu@1
  numberOfSources: 31 //based on release notes
README: Use locally-built binaries when running Renode

- LiteX_I2C_Zephyr.cs
  - Reduce verbosity of the demo
- angle.data
  - Add Renode section
- circle.data
  - Add Renode section
- litex-vexriscv-tlite.repl
  - Add Renode section
- litex-vexriscv-tlite.resc
  - README: Use locally-built binaries when running Renode
- litex-vexriscv-tlite.robot
  - Remove redundant ADXL345 model
<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>LiteX_I2C_Zephyr.cs</td>
<td>Reduce verbosity of the demo</td>
<td>3 months ago</td>
</tr>
<tr>
<td>angle.data</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>circle.data</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>litev-vexriscv-tflite.repl</td>
<td>Add Renode section</td>
<td>9 months ago</td>
</tr>
<tr>
<td>litev-vexriscv-tflite.resc</td>
<td>README: Use locally-built binaries when running Renode</td>
<td>8 days ago</td>
</tr>
<tr>
<td>litev-vexriscv-tflite.robot</td>
<td>Remove redundant ADXL345 model</td>
<td>3 months ago</td>
</tr>
</tbody>
</table>
SPEECH RECOGNITION I²S IN ZEPHIR AND TENSORFLOW LITE

- Speech recognition in systems running LiteX-based soft SoCs with I²S
- Expanded the I²S support in LiteX
- Developed dedicated Zephyr driver
- Created software interface in the TF Lite speech recognition demo
- Read more on the blog
RENODE METRICS ANALYZER

- Collecting execution data from the simulation
- Representing data as graphs
- Currently supported:
  - Executed instructions
  - Memory access
  - Peripheral access
  - Exceptions
- Virtual and real time stamps
TF LITE MACHINE LEARNING ALGORITHM ANALYSIS

- MSc: Analysis of optimization method - quantization
- Experiments with CNN model architectures
- No instrumentation!
- Hard to get that kind of data from real hardware
- Very good for experimenting with e.g. various amounts of available memory

<table>
<thead>
<tr>
<th>Metric</th>
<th>2 pooling layers</th>
<th>1 pooling layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.966887417219</td>
<td>0.940397350993</td>
</tr>
<tr>
<td>Size</td>
<td>19616 bytes</td>
<td>137180 bytes</td>
</tr>
<tr>
<td>Execution time</td>
<td>2425.8 ms</td>
<td>20206.2 ms</td>
</tr>
<tr>
<td>Memory reads</td>
<td>4740 operations</td>
<td>35106 operations</td>
</tr>
<tr>
<td>Memory writes</td>
<td>888 operations</td>
<td>4777 operations</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Metric</th>
<th>Base TensorFlow model</th>
<th>Converted model</th>
<th>Quantized model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model size</td>
<td>96 744 bytes</td>
<td>19 616 bytes</td>
<td>8 896 bytes</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.970198690891</td>
<td>0.970198675497</td>
<td>0.94701986755</td>
</tr>
<tr>
<td>Execution Time</td>
<td>N/A</td>
<td>2280.7 ms</td>
<td>121.9 ms</td>
</tr>
<tr>
<td>Memory Reads</td>
<td>N/A</td>
<td>4496 operations</td>
<td>306 operations</td>
</tr>
<tr>
<td>Memory Writes</td>
<td>N/A</td>
<td>834 operations</td>
<td>78 operations</td>
</tr>
<tr>
<td>Test Description</td>
<td>Status</td>
<td>Last Run</td>
<td>Runtime</td>
</tr>
<tr>
<td>------------------------------------------------------</td>
<td>--------</td>
<td>-------------------</td>
<td>-----------</td>
</tr>
<tr>
<td><strong>LITEX MICROPYTHON TUTORIAL TEST</strong></td>
<td>Pass</td>
<td>2020-09-27 04:20:31+00:00</td>
<td>0:04:21</td>
</tr>
<tr>
<td>A test of the LiteX BuildEnv tutorial for MicroPython on LiteX/VexRiscv. Uses Conda-packaged Renode.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>ZEPHIR TSN/GPTP ON SAM E70</strong></td>
<td>Pass</td>
<td>2020-09-27 04:21:05+00:00</td>
<td>0:10:10</td>
</tr>
<tr>
<td>A Microchip SAM E70 platform running the Zephyr/Renode TSN tutorial.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>RPL UDP IN CONTIKI-NG ON CC2538DK</strong></td>
<td>Pass</td>
<td>2020-09-27 04:19:52+00:00</td>
<td>0:03:32</td>
</tr>
<tr>
<td>Two CC2538DK nodes running Contiki-NG connected with RPL-UDP.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Tested software includes (internally):

- Linux
- FreeRTOS
- Zephyr
- RIOT
- TensorFlow Lite
- Mbed
- NuttX
- Tock
- micropython
- Android
- Bare Metal software
- Contiki
- Wolfboot
- eCos
- Redboot
- getting them published now!
RENODE + TF LITE IN 5 MINUTES: GOOGLE COLAB

- A mostly-Python workspace in the cloud
- Built with Jupyter notebooks
- Allows you to run arbitrary scripts and share the results online
- Fantastic tool for presentation purposes
- Very popular among students and academics
- One click to run everything
- Uses pyrenode - “A very bad lib to talk to Renode over telnet”
- See e.g. https://colab.research.google.com/github/mgielda/renode-notebooks-ipynb/blob/master/unleashed-metrics.ipynb
RENODE + TF LITE IN 5 MINUTES:
GOOGLE COLAB
INTERESTING THINGS TO NOTICE

- Jupyter notebooks can be stored in the percent format as fully usable .py scripts - we use CI to generate the JSON notebook files
- Each instance runs in a separate VM
- It is easy to run with a local Jupyter installation instead
- Can be interactive - e.g. we have a notebook where you can click a button!
- Can be edited online - ideal for experiments
THANK YOU FOR YOUR ATTENTION!