Getting started with Cortex-M software development and Arm Development Studio

Pareena Verma & Ronan Synnott, Arm
August 11, 2020
## AI Virtual Tech Talks Series

<table>
<thead>
<tr>
<th>Date</th>
<th>Title</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>Today</td>
<td>Getting started with Arm Cortex-M software development and Arm Development Studio</td>
<td>Arm</td>
</tr>
<tr>
<td>August, 25</td>
<td>Efficient ML across Arm from Cortex-M to Web Assembly</td>
<td>Edge Impulse</td>
</tr>
<tr>
<td>Sept, 8</td>
<td>Running accelerated ML applications on mobile and embedded devices using Arm NN</td>
<td>Arm</td>
</tr>
<tr>
<td>Sept, 22</td>
<td>How to reduce AI bias with synthetic data for edge applications</td>
<td>Dori.AI</td>
</tr>
</tbody>
</table>

Visit: developer.arm.com/solutions/machine-learning-on-arm/ai-virtual-tech-talks
Agenda

➢ Introduction to Arm Cortex-M processor family
  ➢ Highest performance Cortex-M7 processor
  ➢ First Helium capable Cortex-M55 processor

➢ CMSIS-NN and TensorFlow Lite Micro
  ➢ Build and deploy a ML application with TensorFlow Lite and CMSIS-NN kernels

➢ Q & A 1

➢ Arm Development Studio and demonstration
  ➢ Develop and debug your ML application on Arm Cortex-M7 and Cortex-M55 FVPs

➢ Q & A 2
Arm Cortex-M processors
Cortex-M processor portfolio

- Cortex-M7: Maximum performance, control and DSP
- Cortex-M3: Performance efficiency
- Cortex-M4: Mainstream control and DSP
- Cortex-M0: Lowest cost, low power
- Cortex-M0+: Highest energy efficiency
- Cortex-M33: Flexibility, control and DSP
- Cortex-M35P: Tamper resistance, flexibility, control and DSP
- Cortex-M23: Smallest area, lowest power
- Cortex-M55: Balanced performance and efficiency for ML

TrustZone: Tamper resistance, flexibility, control and DSP

Armv6-M
Armv7-M
Armv8-M
Uplift in DSP and ML performance for Cortex-M

Helium MVE extension
- > 150 new scalar and vector instructions
- Support for complex maths
- Low overhead loops

*Existing processors with DSP extensions
Cortex-M7 key features
https://developer.arm.com/ip-products/processors/cortex-m/cortex-m7

• High performance core with DSP capabilities
  • Powerful DSP instructions
  • SP/DP Floating Point Unit
  • Six stage dual-issue pipeline
  • 5.01 CoreMark/MHz

• Flexible memory systems
  • Up to 16MB tightly-coupled memories for real-time determinism
  • Memory Protection Unit (MPU) and up to 64kB caches
  • 64-bit AXI4 memory interface
Cortex-M55 key features
https://developer.arm.com/ip-products/processors/cortex-m/cortex-m55

• High performance core with DSP and vector processing capabilities
  • Helium vector processing technology
    – Configurable as integer-only or integer + floating-point
  • Powerful DSP instructions
  • HP/SP/DP Floating Point Unit
    – Vector HP/SP
  • Four stage pipeline
  • 4.2 CoreMark/MHz

• Flexible memory systems
  • Up to 16MB tightly-coupled memories for real-time determinism
  • Memory Protection Unit (MPU) and up to 64kB caches
  • 64-bit AXI5 memory interface
### Cortex-M55 arithmetic configuration options

#### Helium Configuration

<table>
<thead>
<tr>
<th>FPU Configuration</th>
<th>MVE=0 (No MVE)</th>
<th>MVE=1 (Integer MVE)</th>
<th>MVE=2 (Integer and Floating-point MVE)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPU=0 (No FPU)</td>
<td>• Scalar Integer</td>
<td>• Vector Integer</td>
<td>• N/A</td>
</tr>
<tr>
<td>FPU=1 (With FPU)</td>
<td>• Scalar Integer • Scalar Float</td>
<td>• Vector Integer • Scalar Float</td>
<td>• Vector Integer • Vector Float</td>
</tr>
</tbody>
</table>
CMSIS-NN and TensorFlow Lite Micro
Meet the TensorFlow Family

TensorFlow

- Model Creation
- Model Training
- Inference in cloud
- Computationally Large
- Google TPU

TensorFlow Lite

- TensorFlow models made suitable for Edge devices
- Inference on device only
- No reliance on network connectivity
- E.g. Speech recognition and NLP in 80Mb Smartphones
- Cortex-A Class IoT

TensorFlow Lite Micro

- Bare metal TensorFlow Lite runtime for Arm MCU
- Inference on device only
- No reliance on network connectivity
- Runs on MCU in 10’sK
- Ultra Low Power- always on*
- Ultra Low Power- Cortex-M

*May be woke often by sensors
TensorFlow Lite Micro

• A version of TensorFlow Lite designed to run on Microcontrollers:
  • Less than 16KB binary footprint + Inference Model (typically < 100K)
  • No memory allocation
  • Maintains the TensorFlow Lite API’s
  • Examples to get started on Arm Cortex-M
    – https://github.com/tensorflow/tensorflow/tree/master/tensorflow/lite/micro/examples/

• Several Arm Fixed Virtual Platforms (FVPs) available today for software development
  • Start porting your ML workloads to Arm Cortex-M systems without the need for a physical target
  • Programmer’s view, which gives you a comprehensive model on which to build and test your software
Software Architecture - CMSIS-NN

- Open Source: [https://github.com/ARM-software/CMSIS_5](https://github.com/ARM-software/CMSIS_5)
- An optimized kernel library for Cortex-M
  - Supports TFLite operators
  - Broadly equivalent to the Arm Compute Library for Cortex-A CPUs
- Offline flow creates a binary for Cortex-M based platforms
- Targets the Cortex-M architectures
  - Armv6-M/Armv7-M/Armv8-M and Armv8.1-M with MVE support
  - Runs on earlier versions of the architecture
CMSIS-NN and TensorFlow Lite for Microcontrollers

Access to optimized kernels through TensorFlow Lite micro

- Support for optimized bit exact int8 kernels
- Fallback on reference kernels when optimization is not available
Build TensorFlow Lite micro examples for Cortex-M7 and Cortex-M55 FVPs

• Start with the examples in the TensorFlow repo
  https://github.com/tensorflow/tensorflow/tree/master/tensorflow/lite/micro/examples/

• Add support for your target platform (Cortex-M7 and Cortex-M55 FVP)

• Generate project/binary with ‘TAGS=cmsis-nn armclang’

• Build the binary for a simple application
  (micro_speech_test) that runs an inference on Cortex-M FVP

Docker project in Arm Tool-Solutions repo on Github
https://github.com/ARM-software/Tool-Solutions/tree/master/docker/tensorflow-lite-micro-fvp
TensorFlow Lite Micro examples – micro_speech

Audio detection to detect the words “yes” and “no”

Silence: 0%
Unknown: 1.00%
Yes : 86.00%
No : 12.00%
Arm Development Studio
Arm Development Studio

The most comprehensive embedded C/C++ dedicated software development solution

**Compiler**
Best-in-class, safety-certified code compilation designed alongside Arm IP

**Multi-Core Debugger**
Family of debug tools for silicon bring-up and software development for complex SoCs

**Virtual prototypes**
Architectural exploration and early software development

**Streamline**
Visibility of software performance running in Linux or baremetal
Arm Fast Models and Fixed Virtual Platforms (FVPs)

- Software development platforms for leading edge Arm IP
  - Available as pre-configured FVPs and model portfolio to build custom platforms
  - Earliest access to simulation models for Arm CPU and SystemIP
  - Ideal for software validation and continuous integration environments

- Performance, fidelity and flexibility
  - High-performance models for software developers
  - Accuracy proven against IP validation suites, used in IP validation flows

Earliest architecture support
Accelerate time to market
Supported by experts
AI Virtual Tech Talks Series

Demonstration
Try it for yourself
Get started

• Get a free 30-day evaluation of Arm Development Studio

• Download the examples used in this webinar
  • git clone https://github.com/ARM-software/Tool-Solutions/

• For more information on Helium technology and get access to programmer’s guides
  • https://developer.arm.com/architectures/instruction-sets/simd-isas/helium

• Contact us
  • Arm-Tool-Solutions@arm.com
Q & A
Join our next virtual tech talk: Efficient ML across Arm from Cortex-M to Web Assembly by Edge Impulse

Tuesday 25 August

Register here:
developer.arm.com/solutions/machine-learning-on-arm/ai-virtual-tech-talks
Thank You
Danke
Merci
谢谢
ありがとうございました
Gracias
Kiitos
감사합니다
धन्यवाद
شكرًا
toda