Case study

EMBEDDED AI ON L31 CORE

NEURAL NETWORKS EMPOWERED BY CUSTOM INSTRUCTIONS

DOMAIN	Neural networks, AI handwriting recognition (MNIST)
PRODUCT	Codasip L31™ processor, Codasip Studio™
RESULT	Optimized custom L31 core for IoT/edge applications
PUBLISHED	August 2022

1 The Context

Over the last few years there has been an important shift from cloud-level to device-level AI processing. The ability to run AI/ML tasks becomes a must-have when selecting an SoC or MCU for IoT and IIoT applications.

Embedded devices are typically resource-constrained, making it difficult to run AI algorithms on embedded platforms. The Codasip Application Engineering team looked at what could make it easier from a software and hardware point of view. They used the **Codasip L31 RISC-V core** and **Codasip Studio** to explore and customize the design.

2 The Scope of the project

The Codasip Application Engineering team used TensorFlow Lite for Microcontrollers (TFLite-Micro) as a dedicated AI framework and compared the performance of the Codasip L31 processor core with both standard and custom extensions. This project highlighted the benefits of custom instructions for neural networks.

3 The Development

The team used TensorFlow Lite Micro with a Codasip L31 RISC-V core to implement a convolutional neural network for image classification. The neural network architecture contains two convolutional and pooling layers, at least one fully-connected layer, vectorized nonlinear functions, data resize and normalization operations (Figure 1).

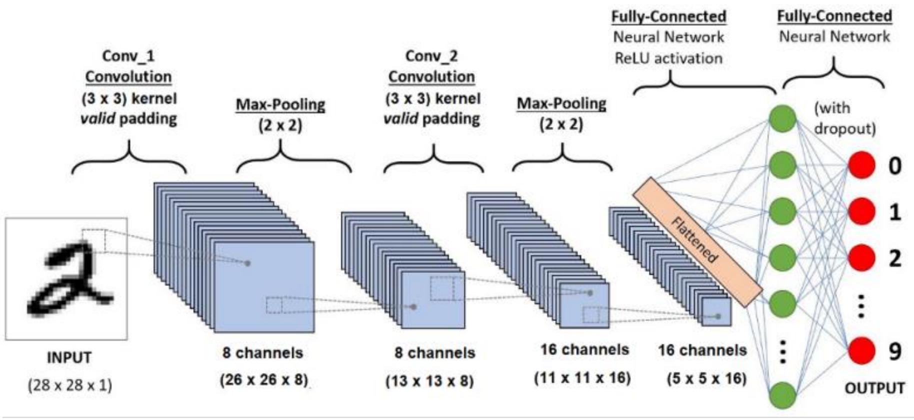

Figure 1 Convolutional neural network architecture

The team took the well-known "MNIST handwritten digits classification" benchmark and used the **Codasip Studio profiler** (Figure 2) to analyze the image classification task. Codasip Studio makes it easy to see which are critical algorithms and where to optimize, identifying hot spots.

Symbol	Address	Instructions	Instructions Percent	Cycles	Cycles Percent
tflite::reference_integer_ops::ConvPerChannel	36fa6	6572379	86.3 %	9340321	83.9 %
tflite::reference_integer_ops::MaxPool	45e60	412255	5.4 %	710898	6.4 %
tflite::reference_integer_ops::FullyConnected	3e388	158370	2.1 %	236154	2.1 %

Figure 2 Codasip Studio Profiler

As would be expected, ~84% of the cycles were used on the image convolution function. The convolution is implemented by deeply nested for-loops.

```
752256  9.880%  for (int in_channel = 0; in_channel < input_depth; ++in_channel) {

374670  4.921%    int8 filter_val = filter_data[filter_offset + in_channel];   lb x10, 0x0 ( x28 )
188064  2.470%    int8 input_val = input_data[value_offset + in_channel];      lb x18, 0x0 ( x26 )

727942  9.561%    acc += filter_val * (input_val + input_offset);              mul x10, x10, x18
                }                                                              c.add x11, x10
```

Figure 3 Profiler identifying 'hot spots' in deepest for-loop

In this case of TFLite convolution, most time is spent for multiply + accumulate operation (mul followed by c.add) and the consequent (vector) loads from the memory (lb instructions after the for-statement). Merging multiplication and addition as well as loading bytes with an immediate address increment were promising ideas for creating RISC-V custom instructions.

4 The Result

Adding two simple custom instructions to improve the arithmetic and vector loads led to a custom L31 core with better performance and power consumption than the standard L31.

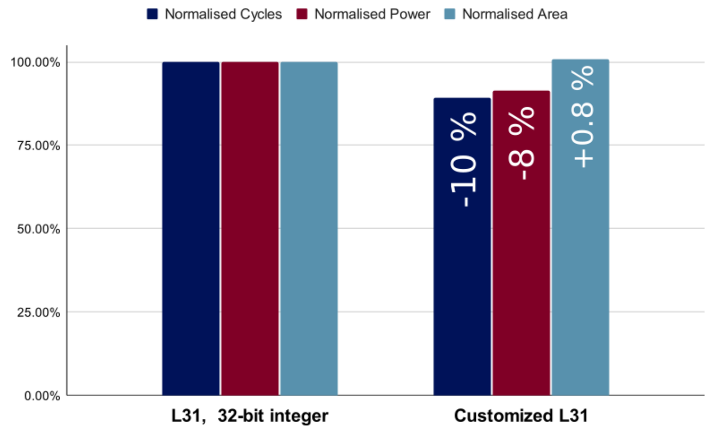

Figure 4 The benefits of adding custom instructions

The number of clock cycles required for image classification were reduced by more than 10% and the power consumption reduced by more than 8%. All of this was achieved with almost no additional cost in area (<1%).

Note: AI & ML applications vary in their computational requirements. The custom instructions example provided above is given for illustration purposes only and does not pretend to be a complete and optimized solution. Other custom instructions might result in further PPA improvements.

About Codasip

Codasip delivers leading-edge RISC-V processor IP and high-level processor design tools, providing IC designers with all the advantages of the RISC-V open ISA, along with the unique ability to customize the processor IP. As a founding member of RISC-V International and a long-term supplier of LLVM and GNU-based processor solutions, Codasip is committed to open standards for embedded and application processors. Formed in 2014 and headquartered in Munich, Germany, Codasip currently has R&D centers in Europe and sales representatives worldwide. For more information about our products and services, visit www.codasip.com.