Motivation

Speech Recognition can be broken down into the following steps:
1) Feature extraction
2) Phoneme classification
3) Further processing to written language

Phoneme extraction step is often done with specialized Recurrent Neural Networks (RNNs):
- Long Short-Term Memory (LSTM)
- Gated Recurrent Units (GRUs)

Theory – LSTM Accelerators

Long Short-Term Memory (LSTM) [2]:
- Feedback of hidden states \( h_t \) with \( \text{N}_h \) elements for short-term memory
- Internal cell state \( c_t \) with \( \text{N}_c \) elements for long-term memory
- Input, forget and output gates for controlling the information flow and to the cell state

The component in LSTM is:

\[
\begin{align*}
  i_t &= \sigma(W_{ax}x_t + W_{ah}h_{t-1} + w_{ai}c_{t-1} + b_i) \\
  f_t &= \sigma(W_{cx}x_t + W_{ch}h_{t-1} + w_{cf}c_{t-1} + b_f) \\
  c_t &= \tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \\
  o_t &= \sigma(W_{ox}x_t + W_{oh}h_{t-1} + w_{co}c_{t-1} + b_o) \\
  h_t &= o_t \odot \tanh(c_t)
\end{align*}
\]

\( W_{ax}, W_{ah}, W_{cx}, W_{ch}, W_{ox}, W_{oh} \): weight matrices
\( b_i, b_f, b_c, b_o \): bias vector
\( i_t, f_t, o_t \): target state/gate

HW Acceleration Implementations:
- States local: BW-starved, minimal reuse
- Weights local: Limited in chip size (memory)

Our Approach:
- Weights local
- Distributed tiles on different accelerators

Main Computational Load in LSTMs:
- Matrix-Vector Multiplication

Systolic Array – Vau da Muntanialas

Vau da Muntanialas: An Energy-Efficient Systolic Array of LSTM Accelerators
Gianna Paulin¹, Lukas Cavigelli¹, Francesco Conti¹,², Luca Benini¹,²
¹ETH Zurich, Zurich, Switzerland
²University of Bologna, Bologna, Italy

Architecture / Datapath
- Muntaniala is a successor of Chipmunk [1]
- New interfaces → PCB friendly
- Performs the accumulated matrix-vector multiplications sequentially on \( \text{N}_{h} \times \text{N}_{h} \) LSTM Units

Additional Specifications
- UMC 65nm CMOS HVT std cells
- On one ASIC: \( \text{N}_{h} \times \text{N}_{h} \) = 96
- Total Memory: 833kB SRAM (12 banks)
- I/O pins:
  - 32 logic pins
  - 8 power pins
- Precision:
  - 8-bit external Fixed Point
  - 16-bit internal Fixed Point

Systolic Design

The Systolic accelerator Muntaniala can be scaled in two ways:

- **2-D scaling:** LSTM cell size
  - Quadratically as matrix-vector multiplication problem
  - \( \text{N}_{h} \times \text{N}_{h} \) dies \( \rightarrow \) \( \text{N}_{h} \times \text{N}_{h} \)

- **3-D scaling:** Multi-Layer
  - Connect multiple \( \text{N}_{h} \times \text{N}_{h} \) Layers
  - Connect hidden states of one layer as input state to next layer
  - \( \text{N}_{h} \times \text{N}_{h} \) dies \( \rightarrow \) \( \text{N}_{h} \times \text{N}_{h} \) Layers of each \( \text{N}_{h} = \text{N}_{h} \times \text{N}_{h} \)

Systolic Array – Vau da Muntanialas

Vau da Muntanialas – Systolic Demonstrator
We have build a system for demonstrating the systolic property:

1) PCB Vau da Muntanialas
- 2x systolic array of Muntanialas
- 1 Layer of the size \( \text{N}_{h} = 2 \times \text{N}_{h} \) Muntaniala = 192
- DCDC Converters
- FPGA Mezzanine Card (FMC) connector

2) FPGA Board: Zedboard
- Controller implementation on the FPGA
- Forwarding of the parameters and features from the ARM SoC to the Muntaniala accelerator through the FMC connector
- Parameters and features read from SD-card
- Record audio from microphone (micro feature extraction on ARM SoC)

Bibliography


