An FPGA Implementation of a Memristive System Based on Wave Digital Principles

E. Solan, B. Janssen, K. Ochs, M. Hübner
Contents

1 Motivation

2 Memristive Emulator

3 Wave Digital Emulation

4 FPGA Implementation

5 Conclusion and Outlook

6 Acknowledgment
Contents

1 Motivation

2 Memristive Emulator

3 Wave Digital Emulation

4 FPGA Implementation

5 Conclusion and Outlook

6 Acknowledgment
Motivation

Why Emulators?

- memristive devices in neuromorphic circuits
- simulation models for reproducible analyses
- huge number of synapses in the brain
- simulations of neuromorphic circuits are very time-consuming

Real-time capable emulators!
Contents

1 Motivation

2 Memristive Emulator

3 Wave Digital Emulation

4 FPGA Implementation

5 Conclusion and Outlook

6 Acknowledgment
Memristive Emulator

Hardware Emulator

Advantages
- efficiency
- common components: suitable for fabrication
- integrability - electrical circuits

Limitations
- fixed functionality
- coupled active components - stability problems
- restricted for a certain application

Reconfigurable emulators based on wave digital algorithms combines advantages of hardware and software emulators.
Contents

1 Motivation

2 Memristive Emulator

3 Wave Digital Emulation

4 FPGA Implementation

5 Conclusion and Outlook

6 Acknowledgment
Wave Digital Emulation

Wave Digital Principle

Reference circuit:

\[ u(t) = u(t_0) + \frac{1}{C} \int_{t_0}^{t} i(\tau) d\tau \]

Differential equation:

\[ t_k = t_0 + kT \]

Difference equation:

\[ u(t_k) \approx u(t_{k-1}) + \frac{T}{2C} [i(t_k) + i(t_{k-1})] \]

Algorithmic model:

\[ a = u + Ri \]

\[ b = u - Ri \]

Wave digital algorithm:

\[ b(t_k) \approx a(t_{k-1}) \]

Implicit

Explicit
Wave Digital Emulation of Memristive Systems

Memristive Reflection Coefficient

\[ b(t_k) = \rho(M) a(t_k) \]

\[ \rho(M) = \frac{M - R}{M + R} \]

\[ \hat{R}(t_k) \]

Transformation Unit

Processing Unit

Reflection Coefficient
Contents

1 Motivation
2 Memristive Emulator
3 Wave Digital Emulation
4 FPGA Implementation
5 Conclusion and Outlook
6 Acknowledgment
HP-Ion Drift Model

Electrical Circuit and Wave Flow Diagram

HP-ion drift model

\[ u(t) = M(z) i(t) \]
\[ \dot{z}(t) = k w(z) i(t) \]
FPGA Implementation

Implementation

Results

\begin{figure}
\centering
\begin{tikzpicture}
  \node[draw, thick, rectangle] (adc) at (0,0) {ADC};
  \node[draw, thick, rectangle] (fpga) at (2,0) {FPGA};
  \node[draw, thick, rectangle] (dac) at (4,0) {DAC};
  \draw[->, thick] (adc) -- (fpga);
  \draw[->, thick] (fpga) -- (dac);
\end{tikzpicture}
\end{figure}

\begin{figure}
\centering
\begin{tikzpicture}
  \begin{axis}[
    xlabel={$u \text{ in V}$},
    ylabel={$i \text{ in mA}$},
    xmin=-1, xmax=1,
    ymin=-4, ymax=4,
    xtick={-1,0,1},
    ytick={-4,0,4},
    legend pos=north west,
    \]
    \addplot[blue, thick] coordinates {(-1,0) (-0.5,4) (0,0) (0.5,-4) (1,0)};\node at (axis cs:-0.5,4) {$f = 0.5 \text{ Hz}$};\node at (axis cs:0.5,-4) {$f = 1 \text{ Hz}$};\end{axis}
\end{tikzpicture}
\end{figure}
Contents

1 Motivation

2 Memristive Emulator

3 Wave Digital Emulation

4 FPGA Implementation

5 Conclusion and Outlook

6 Acknowledgment
Conclusion

- memristive emulators for neuromorphic circuits
- wave digital method
  - preserving passivity of the analog counterpart
  - convenient stability investigations
  - efficient, robust and platform-independent algorithmic model
- FPGA implementation
  - HP-ion drift model
  - hysteresis as expected
  - reconfigurable

Outlook

- ASIC Implementation
  - double barrier memristive device
- emulation of dedicated neuromorphic circuits
## Contents

1. Motivation
2. Memristive Emulator
3. Wave Digital Emulation
4. FPGA Implementation
5. Conclusion and Outlook
6. Acknowledgment
The financial support by the German Research Foundation (Deutsche Forschungsgemeinschaft - DFG) through FOR 2093 is gratefully acknowledged.