

## CRONOS-2024 AREA 2(PO:KAWAHARA)

## <u>Ultra-Brain Neuromorphic by Material-Device-System Co-Research</u>

Principal Investigator: Mutsumi Kimura

(Professor, Electronics, Information and Communication Engineering, Ryukoku University)

Co-PI: Yasuhiko Nakashima (Nara Institute of Science and Technology (NAIST))



**Devices** 

Cyberspace

Big data

Processed data Super-low energy

## Grand Challenge and Goal:

Material-Device-System Co-Research can realize Ultra-Brain Neuromorphic Systems with Super-Low Power Consumption, Compact, and High Speed.

Dramatic Reduction of Power Consumption for Big Data Analysis and Data Communication

is Core of Carbon Neutral

Created future by this research

Summary: Advanced candidate ↓ • Incorporation of others → QKV attention • Memriscapacitor = Analog memristor + Capacitor + Schottky ⇒ Dynamic behavior

- Extended spiking principle = Simultaneous use of various parameters Stochastic
- Super low-leakage TFT = Analog cache → Scaled dot-product attention
- □ Best assign of reusable or unnecessary data to effective integration of nonand volatile devices

Neuromorphic Transformer

- Basic research [SSI] Prototype Actual verification
- Device=10<sup>5</sup>: Speed<10ms: Power<100pJ/token
- [LSI] System simulation
- Device=10<sup>14</sup>: Speed<10ms: Power<100mJ/token
- Transfer research Prototype Actual verification
- Social implementation IoT verification
- Novelty Originality Co−research → Simple LSI
  Resource O(n²) → O(1) Efficiency180 nm264 → 1540 TOPS/W

## Social Impact:

- All power consumption 1. World Today 2050
- Data communication → 100 power 25PWh → 800 → 29
  Complete solution to future problem of power consumption

Society 5.0 & SDGs & Carbon neutral

Synapse array QKV attention Reusable Extended spiking computing principle Scaled dot-product attention Super low-leakage TF Neuromorphic transformer Neuromorphic transformer by memriscapacitors, spiking principle, and super low-leakage TFTs

**Materials** 

**Analysis** 

Cyberspace

Big data

Neuromorphic systems

Implementation