BitShift-RoPE: Zero-FLOP Relative Positional Encoding for Spiking Neural Network Transformers

Seung-Kyu Hong, Sangheum Hwang, Hyuk-Yoon Kwon

Graduate School of Data Science, Seoul National University of Science and Technology

NeurIPS 2026

Spiking neural networks meet Transformers

BIGBASE
Think the biggest in the world!

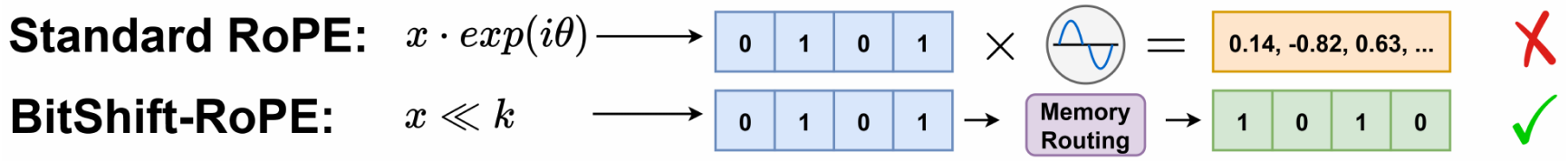
- **Spiking Neural Networks (SNNs) communicate with binary spikes**
  - Floating-point multiply-adds are replaced by sparse synaptic operations (SOPs), the source of the energy advantage
- **Spiking self-attention (SSA) on binary queries and keys**
  - Both operands are 0 or 1, so the dot product is a bitwise AND that counts coincident ones
- **Self-attention is permutation-invariant, so a positional encoding (PE) is required**
  - Spikformer, Spikingformer, and QKFormer rely on convolution to learn position implicitly
  - This limits long-range dependencies and shows how hard it is to inject position without breaking spike binarity

$$\mathrm{SSA} = S_q \cdot S_k^{\top}, \qquad S_q,\, S_k \in \{0,1\}^{N \times D}$$

Eq. (2): spiking self-attention on binary spike tensors

Existing PEs break the spike domain

- **Pre-spike PE (applied before the LIF neuron)**
  - Extra floating-point work, and the LIF non-linearity breaks the relative-distance identity of attention
- **RoPE**
  - Trigonometric rotation turns binary spikes into floating-point tensors
- **CPG-PE (post-spike, absolute)**
  - Stays binary, but is keyed on absolute coordinates and aliases on long sequences
- **Gray-PE and Log-PE (post-spike, relative)**
  - Multi-bit integer outputs replace AND with XNOR; XNOR(0,0) = 1, so the accumulation becomes dense



RoPE multiplies spikes by a rotation and leaves the binary domain. BitShift-RoPE permutes the 0/1 array instead.

Three constraints for an SNN positional encoding

BIGBASE
Think the biggest in the world!

- **C1** No additional floating-point operations are introduced during positional encoding
- **C2** The output of the positional encoding remains binary, so attention does not require a multiplier
- **C3** Relative distance in the attention map is expressible as an exact mathematical formula

Method	C1 no added FP ops	C2 binary output	C3 relative distance
Pre-spike PE (Spikformer, SDT, QKFormer)	✗	✓	✗
CPG-PE (post-spike, absolute)	✓	✓	✗ (aliasing)
Gray-PE / Log-PE (post-spike, relative)	✗	✗ (multi-bit integers)	✓
BitShift-RoPE (Ours)	✓	✓	✓

BitShift-RoPE is, to our knowledge, the first SNN positional encoding that satisfies all three simultaneously.

Key idea: replace the rotation with a cyclic shift

BIGBASE
Think the biggest in the world!

- **RoPE encodes position with a rotation matrix R and recovers the relative distance in the dot product**
- **BitShift-RoPE replaces R with a cyclic shift matrix S over the spike-array indices**
  - S is orthogonal, so the absolute positions cancel exactly in the attention product
- **The shift only relocates 0/1 values**
  - Q and K stay binary, and attention remains a sparse bitwise AND

$$(X_q R_m)\cdot(X_k R_n)^\top = X_q R_m R_n^\top X_k^\top = X_q R_{m-n} X_k^\top \qquad S^\top = S^{-1},\quad (S^n)^\top = S^{-n}$$

RoPE (ANN): Eq. (4)

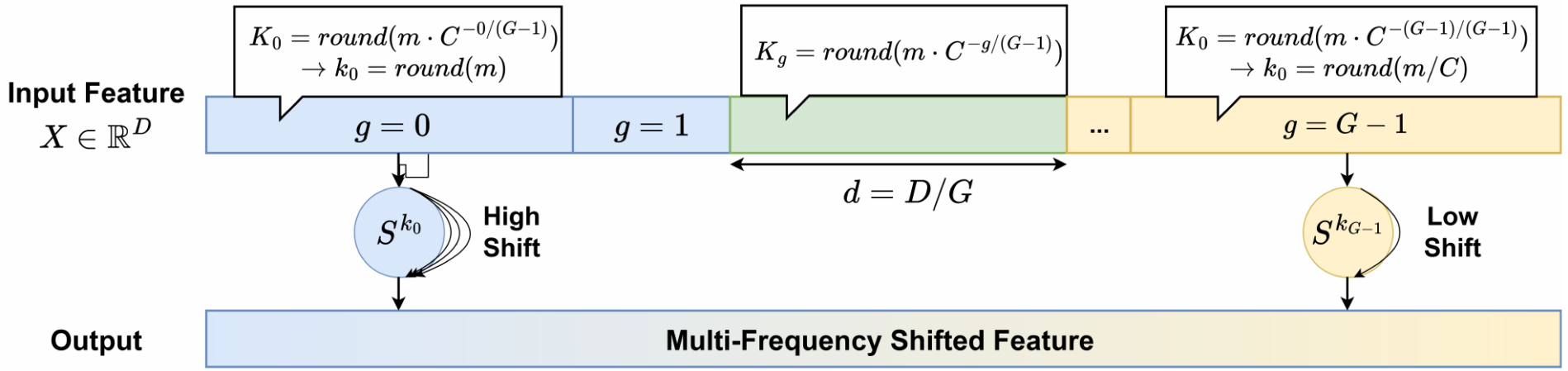
$$\mathrm{Attn}(m,n) = (QS^m)\cdot(KS^n)^\top = QS^m S^{-n} K^\top = QS^{m-n}K^\top$$

BitShift-RoPE: Eqs. (5) and (6)

Multi-frequency shift mapping

- **The feature dimension D is split into G groups, each with its own integer shift count**
  - g = 0 shifts by exactly m and captures fine ordering; the last group shifts slowly and captures global context
  - Derived from RoPE's exponential-decay schedule with a single decay constant C (G = 4, C = 64)
- **Rounding error is bounded by one shift step per group, independently of absolute position**

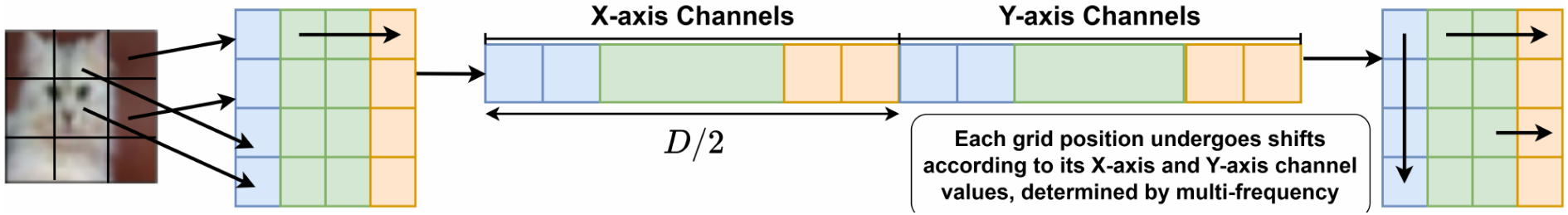
$$k_g(m) = \text{round}\left(m \cdot C^{-g/(G-1)}\right), \quad g \in \{0, \ldots, G-1\} \qquad |k_g(m) - k_g(n) - k_g(m-n)| \leq 1$$



2D extension for vision

- **Channel groups are split into X-axis and Y-axis halves, Q = [Qx ‖ Qy]**
  - Shifts on disjoint channel blocks commute, so the split is exact
  - Keying on (mx, my) separately keeps displacements such as (1,3), (2,2), (3,1) distinguishable, which a sum mx + my would not
- **The permutation preserves binarity, so the 2D variant keeps the same cost profile**

$$\mathrm{SSA}(\mathbf{m},\mathbf{n}) = Q_x S^{\,k_{x,g'}(m_x) - k_{x,g'}(n_x)} K_x^{\top} + Q_y S^{\,k_{y,g'}(m_y) - k_{y,g'}(n_y)} K_y^{\top}$$



Theory

- **Proposition 1 (DFT diagonalization of the cyclic shift)**
  - An m-step shift is a phase rotation at uniformly spaced DFT modes, the same form as RoPE
  - The correspondence is structural, not pointwise: the frequency schedule differs, while the shift-difference identity and binarity are preserved exactly

$$S^m = F^{-1}\Lambda^m F, \quad \Lambda^m_{r,r} = e^{-im\theta_r}, \ \theta_r = \frac{2\pi r}{D}$$

Eq. (7)

- **Corollary 2 (bounded, non-accumulating rounding error)**
  - The bound is 0 for g = 0 and reaches π/2 at the Nyquist mode, so it establishes non-accumulation with sequence length rather than uniformly small error

$$\epsilon_r \leq \frac{1}{2}\theta_r = \frac{\pi r}{d}$$

Corollary 2, per DFT mode r over a group of width d

- **The joint shift code is collision-free only within its period (512 for G = 4, C = 64, d = 8)**

Implementation: one gather, no arithmetic

BIGBASE
Think the biggest in the world!

- **The index table depends only on (G, C, N)**
  - Built once at initialization with pow, einsum, and round, then registered as a buffer
- **Forward path: index expansion + a single torch.gather along the feature dimension**
  - Operator tracing shows aten::gather only, with no online index construction and no floating-point rotation
  - “Memory-pointer routing” is a plausible hardware realization of the same deterministic permutation

Path	Index construction	Forward-path operators	FP positional arithmetic
Cached length (all experiments)	once, at init	expand + aten::gather	none
Uncached length	during forward	index construction + aten::gather	pow, mul, einsum, round, remainder

Zero-FLOP is an architectural arithmetic property, not zero runtime: the gather carries addressing, memory traffic, and index storage.
On the isolated attention block its software overhead is 7.7% at N = 64 and 3.9% at N = 1024, since the gather is $O(N \cdot D)$ while attention is $O(N^2 \cdot D)$.

Experimental setup

BIGBASE
Think the biggest in the world!

- **Three domains, four SNN backbones**
  - Time series: Metr-LA, PEMS-BAY, Solar, Electricity at horizons {6, 24, 48, 96} on Spikformer, Spikingformer, QKFormer
  - Text: MR, SST-2, Subj, SST-5 on SpikeBERT
  - Image: CIFAR-10/100, Tiny-ImageNet, CIFAR10-DVS, DVS128-Gesture on Spikformer, QKFormer
  - Dense prediction: Oxford-IIIT Pet trimap segmentation with a linear per-token decoder
- **Baselines: Conv-PE, CPG-PE, Gray-PE, Log-PE, plus floating-point references**
- **Protocol**
  - T = 4, G = 4, C = 64 everywhere; three independent seeds, mean ± s.d.
  - Paired t-test over the 16 dataset × horizon conditions (time series); Welch two-sided t-test per benchmark (text, image)

Results: time series and text

BIGBASE
Think the biggest in the world!

Time series, average R^2 over 16 dataset × horizon conditions

PE	Spikformer	Spikingformer	QKFormer
FP ref. (Transformer w/RoPE)	.756	–	–
Conv-PE	.733	.654	.688
CPG-PE	.744	.682	.708
Gray-PE	.748	.687	.729
Log-PE	.750	.697	.733
BitShift-RoPE (Ours)	**.758**	**.712**	**.748**

Paired improvement over the strongest baseline: +0.0075 ($p = 0.025$), +0.0146 ($p = 0.042$), +0.0148 ($p = 0.002$).
Margins are small relative to seed variation, but 44 of 48 cells favour BitShift-RoPE over Log-PE.

Text classification, SpikeBERT accuracy (%)

PE	MR	SST-2	Subj	SST-5	Avg.
BERT (FP reference)	86.39	92.01	95.43	49.87	80.93
w/o PE	71.84	80.17	86.60	35.20	68.45
CPG-PE	72.73	81.77	86.76	35.34	69.15
Gray-PE	70.71	81.08	80.00	34.39	66.55
Log-PE	71.09	81.42	80.80	34.43	66.94
BitShift-RoPE (Ours)	**73.81**	**82.00**	**87.90**	**35.93**	**69.91**

Significant over CPG-PE on MR ($p = 0.033$) and Subj ($p = 0.006$); not on SST-2 or SST-5.
Gray-PE and Log-PE fall below the no-PE baseline. Zero extra parameters (109.8M).

Results: image and dense prediction

BIGBASE
Think the biggest in the world!

Image classification accuracy (%), mean over three seeds

Backbone / PE	CIFAR10	CIFAR100	Tiny-IN	C10-DVS	DVS128	Avg.
Spikformer / Conv-PE	94.61	77.55	55.28	80.3	87.59	79.01
Spikformer / CPG-PE	94.94	77.51	54.76	80.8	88.97	79.40
Spikformer / Gray-PE 2D	93.74	77.53	55.45	80.1	88.36	79.04
Spikformer / BitShift 2D	**95.13**	**77.68**	**56.31**	**81.2**	**93.79**	**80.82**
QKFormer / Conv-PE	96.32	80.90	58.64	85.1	98.26	83.84
QKFormer / CPG-PE	96.21	80.65	56.75	82.0	97.26	82.57
QKFormer / Gray-PE 2D	96.03	80.70	57.48	81.4	97.91	82.70
QKFormer / BitShift 2D	**96.36**	**81.62**	**59.05**	**85.2**	**98.61**	**84.17**

Resolvable on CIFAR100 and Tiny-ImageNet for both backbones and on CIFAR10, DVS128-Gesture for Spikformer.
CIFAR10-DVS is quantized to 0.1% (1,000 samples), so no claim is made there.

Oxford-IIIT Pet trimap segmentation (128 px)

PE	Params	mIoU	Boundary IoU
No PE	3.61 M	0.638	0.335
Conv-PE	4.20 M	0.665	0.376
CPG-PE	3.67 M	0.660	0.363
Gray-PE 2D	3.61 M	0.502	0.261
BitShift 2D (Ours)	**3.61 M**	**0.673**	**0.381**

Gains over the no-PE floor of +0.035 mIoU ($p = 0.006$) and +0.046 boundary IoU ($p = 0.011$) must come from the encoder, since the decoder is linear.
Comparable to Conv-PE ($p = 0.13$) with 0.59 M fewer parameters.

Efficiency: zero parameter and FLOP overhead

BIGBASE
Think the biggest in the world!

Method (Spikformer, Electricity)	Params (M)	FLOPs (M)	SOPs (M)	XNOR-OPs (M)	Energy (mJ)	Latency (ms)	Throughput (/s)
Vanilla Spikformer	1.67	24.70	176.80	0	0.216	4.03 ± 0.02	248.0
CPG-PE	1.71	26.59	180.41	0	0.224	4.41 ± 0.81	226.5
Gray-PE	1.98	31.54	207.56	75.87	0.260	10.92 ± 0.92	91.6
Log-PE	1.98	31.54	207.35	57.80	0.260	10.96 ± 0.72	91.3
BitShift-RoPE (Ours)	**1.67**	**24.70**	**176.80**	**0**	**0.216**	**4.06 ± 0.02**	**246.3**

L = 168, D = 256, T = 4. Energy is a Horowitz operation-count estimate (45 nm CMOS), not a silicon measurement. Latency is wall-clock on GPU under PyTorch.

- **Parameter and FLOP parity follow from the design**
  - The shift schedule is deterministic in (G, C, N), and the index table is held as a buffer, so the forward path adds no arithmetic operator of any type
- **Gray-PE and Log-PE run 2.7× slower because they lose sparsity, not because XNOR is expensive**
  - XNOR(q,k) = 1 whenever the operands agree, including 0–0, so nearly every term is non-zero and the accumulation becomes dense
- **BitShift-RoPE keeps the operands sparse and binary and stays within 1% of the vanilla backbone**

Limitations and conclusion

BIGBASE
Think the biggest in the world!

➢ Limitations

- Index table grows as O(N·G); negligible here but relevant for $N \gg 10^4$
- A single fixed decay constant C = 64 for all domains; unlikely to be Pareto-optimal for any one
- The joint shift code is alias-free only within its period, so the span must be checked against the sequence length or patch grid
- Energy figures are Horowitz estimates pending validation on neuromorphic silicon

➢ Conclusion

- Relative position is injected by permuting binary queries and keys, with no floating-point arithmetic on the encoding path
- Q and K stay binary, so attention remains a sparse bitwise AND at the vanilla backbone's cost
- Highest average on every backbone and modality across time series, text, image, and dense prediction, at zero additional parameters

Thank you

BIGBASE
Think the biggest in the world!

Code: github.com/bigbases/Bit_Shift_RoPE

Seung-Kyu Hong · hjh9681@seoultech.ac.kr
BIGBASE Lab, Seoul National University of Science and Technology