29.6 A 1.5μW End-to-End Keyword Spotting SoC with Content-Adaptive Frame Sub-Sampling and Fast-Settling Analog Frontend

Ji-Hwan Seol[1,2], Heejin Yang[1], Rohit Rothe[1], Zichen Fan[1], Qirui Zhang[1], Hun-Seok Kim[1], David Blaauw[1], Dennis Sylvester[1]

[1]University of Michigan, Ann Arbor, MI
[2]Samsung Electronics, Hwasung, Korea

Keyword spotting (KWS) has become essential as a wake-up mechanism for edge IoT devices. While recent advances in deep learning have improved KWS accuracy [1], reducing system power consumption remains a challenge. A typical KWS signal chain consists of an analog frontend (AFE), feature extractor (FE), and neural network classifier (NN). To reduce total KWS power, all three blocks must be carefully co-optimized. Recent KWS work reported 0.51μW consumption for the FE and NN, but it only supports two keywords and lacks an AFE, whose power often dominates [2]. A KWS system including an AFE was also proposed but consumes 16μW [3]. This work proposes a fully integrated keyword spotting system that employs the skip RNN algorithm [4] to simultaneously reduce the power consumption of the AFE, FE, and NN by adaptively sub-sampling (i.e., skipping) input frames based on the signal content. The skip RNN continually decides whether the RNN state is to be updated or skipped for one or more 16ms-long frames based on its input content history, reducing NN operation and hence power. We further propose a scheme to use the NN skip decision to dynamically turn off the AFE and FE, which dominate the KWS power (combined >65%) for one or more consecutive frames, achieving 3× power reduction. The proposed AFE features a programmable switched capacitor resistor and two-step switching frequency control, demonstrating less than 1ms settling time. The FE and NN employ computational sprinting with efficient scheduling to reduce their operation time and static current. The proposed KWS system consumes 1.5μW (reduced from 4.47μW with an average of 76% frame skipping) in 28nm CMOS, while achieving 92.8% accuracy on five-word GSCD KWS.

Figure 29.6.1 (top) shows the operating principle of the proposed KWS system. The AFE generates an audio frame (16ms) from the input audio waveform, which is then converted by the FE into a log-Mel feature vector, fed to the RNN. Unlike a conventional RNN, where every incoming frame is processed and updates its state, the skip RNN is augmented with a skip policy module that adaptively determines whether the RNN state is updated or skipped. The skip policy module also produces an N_{SKIP} indicating how many frames to skip. The skip RNN is end-to-end trained with the skip policy module by having it observe a sequence of frames to simultaneously learn both keyword classification and skip control. Following a decision from the skip policy module, we power down the system including the AFE and FE for N_{SKIP} frames. Power down is controlled by the skip control FSM, which is always on. Fig. 29.6.1 (middle) shows the overall block diagram of the proposed KWS system, comprised of an AFE, FE, NN, and skip control FSM. The skip control FSM provides power gate (PG) signals for each of the other three blocks, as well as each block's clock, start, and isolation control signal. Fig. 29.6.1 (bottom) shows the detailed operation of the proposed system. KWS begins with PG_ON_{AFE} that turn on the AFE PG, followed by $START_{AFE}$ to enable the AFE control logic (not shown). The FSM inserts a programmable timing margin (up to 250μs with 15.6μs step) between AFE PG_ON and START for proper supply stabilization. After the amplifier settling time (T_{SET}), an ADC starts to sample the incoming signal. Upon conversion of one audio frame, the FE and the RNN process the data. The AFE therefore remains on for an additional T_{FE} + T_{NN} to avoid losing any data from the next frame while the RNN processes the current frame. When re-enabling the AFE after one or more skipped frame, the FSM *pre-starts* the AFE for T_{SET} to settle it before the ADC is again enabled to digitize the new frame. In addition to T_{FRAME}, the AFE also consumes power during T_{SET}, T_{FE}, and T_{NN}, which represents an overhead (Fig. 29.6.2, top left). Therefore, to reduce AFE energy overhead, our design focuses on achieving fast settling time for the AFE, and short operation times for the FE and RNN.

Figure 29.6.2 shows the proposed fast settling AFE. A capacitive feedback (CF) amplifier is attractive for its low power consumption, but has a high-pass corner (f_{HP}) that is inherently low, resulting in a long settling time (e.g., 37ms when f_{HP}=50Hz), which is comparable to T_{FRAME}. We therefore designed the first stage LNA with a DC-coupled Gm-ratio structure [5], while the second stage PGA uses a CF structure. Using a DC-coupled structure for the first stage reduces settling time by 54% due to its lack of a high-pass corner. To improve the settling time of the CF PGA, its f_{HP} must be set as high as possible, without filtering out voice content that would degrade KWS accuracy. Based on software KWS simulations (Fig. 29.6.2, top), f_{HP} can be increased to 140Hz (compared to a conventional value of <50Hz) but must be carefully controlled across PVT conditions to avoid inadvertent excursions into higher frequencies that would impact KWS accuracy. Traditionally, the corner in a CF amp is set using a "pseudo-resistor", but its high PVT sensitivity (>10×) cannot achieve the required corner accuracy. Instead, we employ a switched capacitor resistor with equivalent resistance set by its capacitance and the switching clock ($\Phi_{0,1}$), which are insensitive to PVT. Further, we also adopt a two-step frequency control, where $\Phi_{0,1}$ operates momentarily at a higher frequency (F_{FAST}) to achieve fast settling before switching to its final value (F_{NOM}) as shown in Fig. 29.6.2 (top right), reducing the AFE settling time by 6×. T_{SET_FAST}, T_{SET_NOM}, and F_{FAST} are programmable. By combining all techniques (Gm first stage, increased f_{HP}, and two-step frequency control), settling time is reduced from 37 ms to <1 ms, yielding a 1.1μW average power reduction. The ADC is an 8b synchronous SAR architecture with 16kS/s sampling frequency. The amplifiers, the ADCs capacitor DAC, and the ADC comparator operate using 1.4V, while the AFE FSM and the SAR logic operate at 0.65V. When PG_ON_{AFE} goes low, all AFE blocks except the capacitor DAC are power gated. Fig. 29.6.2 (bottom right) details the measured AFE performance.

The FE, shown in Fig. 29.6.3 (top), consists of windowing, FFT, Mel filter, and log units. A 16ms 256-point 8b non-overlapping audio frame is multiplied by the Hanning window and sent to the FFT module and further processed by power calculation, Mel filter, and log module to generate an 8b 26-dimension feature vector. Because the power-dominating FFT must process non-consecutive frames, it cannot use a conventional pipelined architecture. We therefore employ a memory-based fully-folded structure with a single butterfly unit (BFLY, Fig. 29.6.3, bottom left). Instead of a single-bank dual-port SRAM, we use four banks of a single-port SRAM, which is simpler and smaller than a dual-port SRAM, reducing memory power by 80% [6]. One BFLY calculation requires 2 cycles, where one additional cycle resamples the BFLY input data to reduce glitch power, saving 22% BFLY power. The architecture requires 1152 cycles to finish one FFT, and overlaps the window, FFT power, Mel, and log operations to reduce overall operating time. The FE sprints to complete its computation quickly (290μs) to reduce AFE power overhead and is then power gated to minimize its leakage power. The FE sprinting frequency is set at 4MHz, which marks the point of diminishing returns as seen in Fig. 29.6.3 (bottom).

The FE feeds the 26-dimension features to the 64 hidden nodes of the RNN. Each RNN cell is designed with a gated recurrent unit (GRU) (Fig. 29.6.4 bottom left). The 64 hidden states of the RNN are processed by the fully connected (FC) layer with eight output nodes. Seven of these FC nodes generate class outputs (5 keywords, one non-keywords, one noise). The last FC node produces the skip score for the skip policy module, which accumulates the skip score. When the accumulator overflows, it latches a counter to produce N_{SKIP} (Fig. 29.6.4, top left). The RNN accelerator consists of a MAC array, WMEM, functional units (FU), register bank, and a NN FSM. The MAC array is designed with 64 parallel MACs. We adopt an output stationary dataflow for the MAC array, where the partial sums for the 64 GRU gate vectors are accumulated in the 64 MACs until the processing of each GRU gate vector is completed. For the FC layer, 8 MACs process the 8 FC output nodes in parallel. All activations are scaled and quantized to 12b. The always-on WMEM stores the 8b weights for the RNN model and is custom designed using latches with high V_{th} transistors, achieving 20× leakage power reduction compared to the foundry SRAM (Fig. 29.6.4 middle). After each frame, the RNN hidden states are stored in WMEM before the NN block is power gated. The operation of the MAC array, FU, and the register bank are scheduled to overlap their operations reducing a single frame processing time to 495 cycles (Fig. 29.6.4 bottom). The RNN sprints with a 1MHz clock to reduce the leakage power and optimize overall energy efficiency.

The proposed chip is fabricated in 28nm CMOS (Fig. 29.6.7). We trained the model with different skip ratios and tested KWS accuracy (Fig. 29.6.5 top left), where the GCSD dataset with 5 keywords task is used for the training and testing. Skip ratio can be changed using a single hyperparameter of the loss function [4]. The tested accuracy drop is less than 1% at the skip ratio used for the deployed model (=0.76). Fig. 29.6.5 (top right) compares the measured power consumption of each block of the proposed KWS system when the skip ratio is 0 and 0.76. Measured power reduces from 4.47μW to 1.48μW, achieving 3× power reduction. Fig. 29.6.5 (bottom left) shows the measured amplifier output and frame enable control for an example audio signal, showing the adaptive enable/skip pattern according to input data. Fig. 29.6.5 (bottom right) shows the measured dynamic power of the AFE, FE, and NN. Fig. 29.6.6 provides a comparison table of recently published KWS chips. The proposed design is the lowest power KWS system that fully integrates AFE, FE, and NN.

Acknowledgement:
We gratefully acknowledge TSMC University Shuttle Program for chip fabrication.

References:

[1] M. Price et al., "A Scalable Speech Recognizer with Deep-Neural-Network Acoustic Models and Voice-Activated Power Gating," *ISSCC*, pp. 244–245, 2017.
[2] W. Shan et al., "A 510nW 0.41V Low-Memory Low-Computation Keyword-Spotting Chip Using Serial FFT-Based MFCC and Binarized Depthwise Separable Convolutional Neural Network in 28nm CMOS," *ISSCC*, pp. 230-231, 2020.
[3] J. Giraldo et al., "18μW SoC for Near-Microphone Keyword Spotting and Speaker Verification," *IEEE Symp. VLSI Circuits*, pp. 52-53, 2019.
[4] V. Campos et al., "Skip RNN: Learning to Skip State Updates in Recurrent Neural Networks," *ICLR*, 2018.

2023 IEEE International Solid- State Circuits Conference (ISSCC) | 978-1-6654-9016-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISSCC42615.2023.10067808

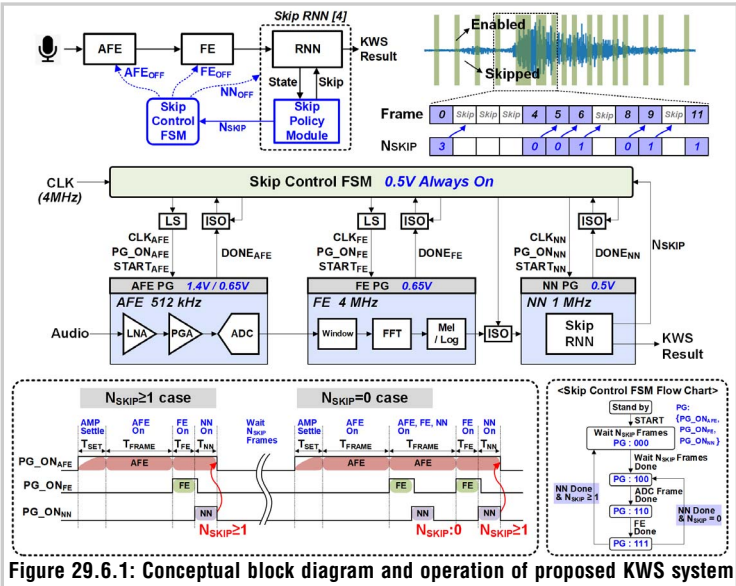

Figure 29.6.1: Conceptual block diagram and operation of proposed KWS system (top), detailed block diagram of proposed KWS SoC (middle), and detailed operation of proposed system (bottom).

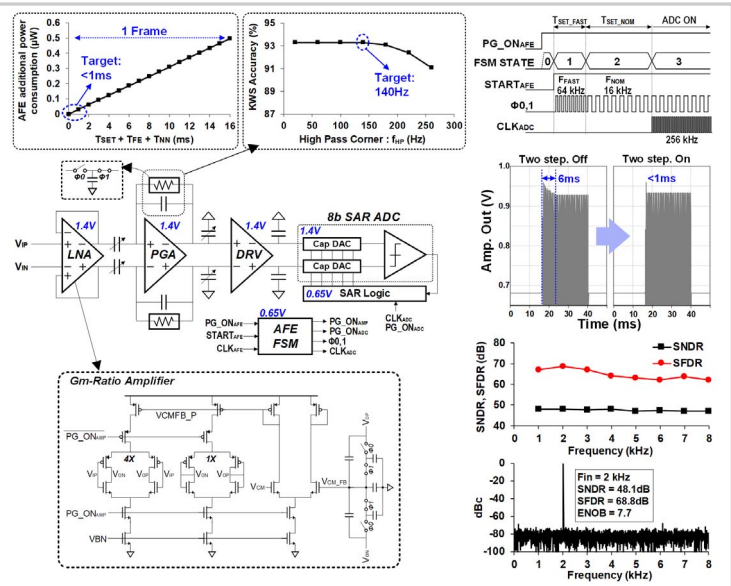

Figure 29.6.2: Proposed fast settling AFE with switched capacitor resistor and two step switching frequency control.

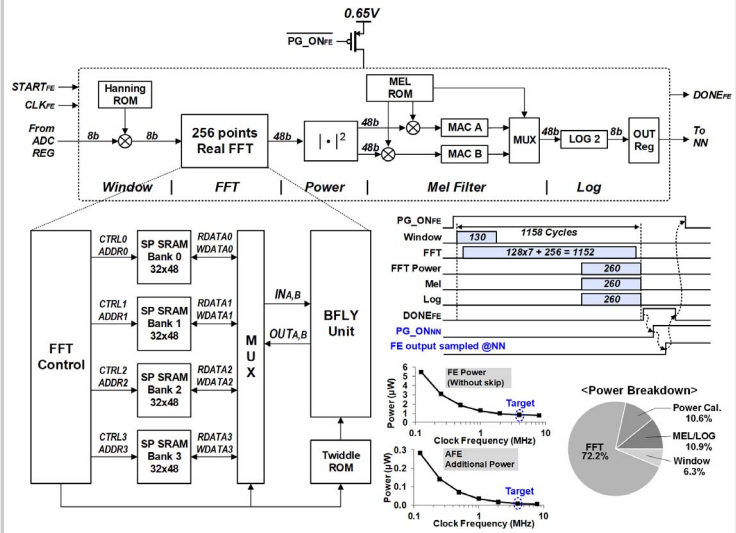

Figure 29.6.3: Block diagram of proposed FE (top), detailed structure of FFT module (bottom left), timing diagram of FE operation (middle right), and simulated power reduction from sprinting and simulated power breakdown (bottom right).

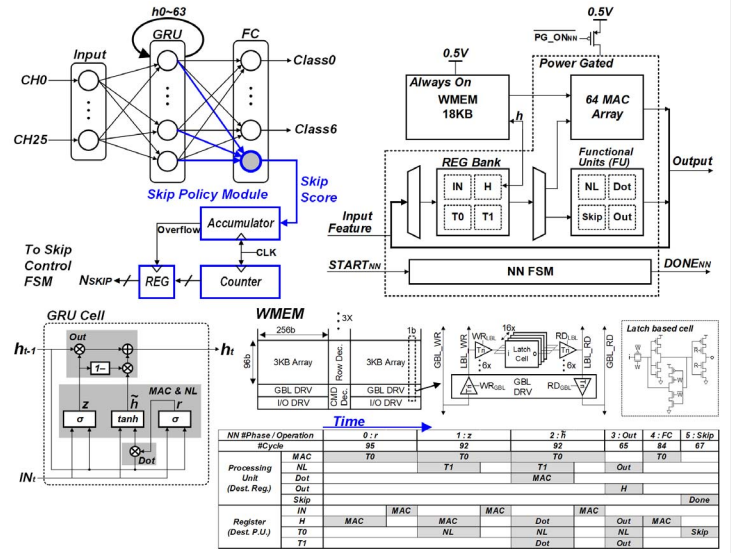

Figure 29.6.4: Conceptual block diagram of proposed RNN accelerator (top left), detailed block diagram of RNN accelerator (top right), proposed latch cell-based WMEM (middle), and RNN scheduling table (bottom).

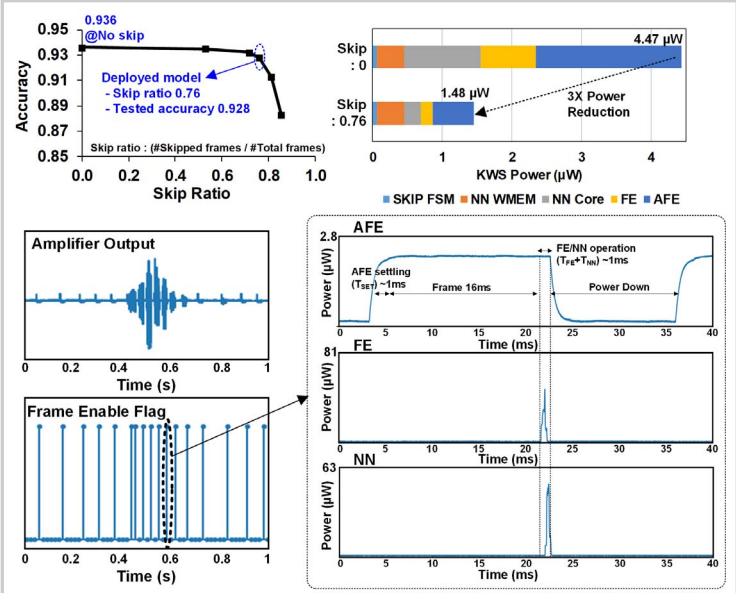

Figure 29.6.5: Measurement results of proposed KWS SoC.

		This work	Kim ISSCC 2022	Shan ISSCC 2020	Giraldo VLSI 2019	Giraldo ESSCIRC 2018
Tech (nm)		28	65	28	65	65
Area (mm^2)		0.8	2.03	0.23	2.56	1.04
Algorithm		Skip RNN	RNN	DSCNN	RNN	RNN
Memory (KB)		18	27	2	105	32
Latency (ms)		16	12.4	64	16	16
Analog Frontend		Yes	Yes	No	Yes	No
Feature Extraction		Yes	Yes	Yes	Yes	No
Neural Network		Yes	Yes	Yes	Yes	Yes
Dataset		GSCD	GSCD	GSCD	GSCD	N/A
# Classes (Keywords)		7 (5)	12 (10)	3 (2)	12 (10)	4 (4)
Accuracy (%)		92.8	86.0	94.6	90.9	91.2
Power (µW)	AFE	0.59	13	-	1.8	-
	FE	0.18		0.34	7.1	-
	NN	0.71	10	0.17	3.4	5
	Total	1.48	23	-	16.1	-

Figure 29.6.6: Comparison table with recently published KWS chips.

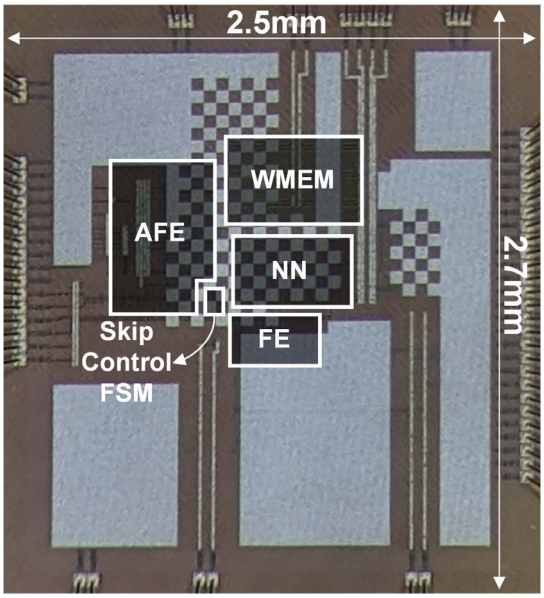

Figure 29.6.7: Chip photograph.

Additional References:
[5] J. Lim, et al., "A 0.19×0.17 mm² Wireless Neural Recording IC for Motor Prediction with Near-Infrared-Based Power and Data Telemetry," *ISSCC*, pp. 416-417, Feb. 2020.
[6] J. Kwong et al., "An Energy-Efficient Biomedical Signal Processing Platform," *JSSC,* vol. 46, no. 7, July 2011.

978-1-6654-9016-0/23/$31.00 ©2023 IEEE