19.7 A Scalable Pipelined Time-Domain DTW Engine for Time-Series Classification Using Multibit Time Flip-Flops with 140Giga-Cell-Updates/s Throughput

Zhengyu Chen, Jie Gu

Northwestern University, Evanston, IL

Dynamic time warping (DTW), a variant of the dynamic programming algorithm, is widely used for time series classification [1]. Its strong capability for distance measurement for variable-speed temporal sequences makes DTW a popular method for time-series classification in broad applications, such as ECG diagnosis, motion detection, DNA sequencing, etc. [1]. Several efforts have proposed for accelerating the operation of DTW, including a recent demonstration of time-based design in DNA sequencing [2]. However, the demonstration was confined to single-bit operations, a fixed sequence length and low throughput due to non-pipelined operation and a large single-bit delay. To overcome such challenges, this work presents a general-purpose DTW engine for time-series classification using time-domain computing. A pipelined operation is enabled by a time flip-flop (TFF) leading to order-of-magnitude improvements in throughput and a scalable processing capability for time series. Compared with recent time-domain designs, which do not have time-domain memory elements, this work realizes a time-domain pipelined architecture [3].

Fig. 19.7.1 shows the basic principle of DTW, which detects similarities among temporal signals having variable speed. As shown in Fig. 19.7.1, for two time series A and B, $D_{i,j}$ can be formulated as the summation of absolute difference $|A_i - B_j|$ and the minimum value of its three ancestor nodes $\min(D_{i-1,j}, D_{i,j-1}, D_{i-1,j-1})$ where A_i and B_j denotes the i^{th} and j^{th} elements of A, B, and $D_{i,j}$ denotes the DTW value at node (i, j). A "warping path" is produced in order to align the two signals in time, as highlighted in Fig. 19.7.1. Time-domain design has shown significant advantages in performing the warping and comparison operation due to its efficiency in key operations, such as minimum (MIN) and absolute (ABS), which only require a few logic gates. The lower side of Fig. 19.7.1 shows the time-domain principle of DTW, where digital inputs are converted into quantized pulse widths and further processed in the time domain. The pulse $T(D_{i,j})$ can be obtained by the accumulation of the pulses of $|A_i - B_j|$ and $\min(D_{i-1,j}, D_{i,j-1}, D_{i-1,j-1})$, both of which are generated by the time-domain MIN module and ABS module. Both accumulation and pipelined operation are realized by the implemented TFF.

Fig. 19.7.2 shows the ring-based multi-bit TFF. A 33-stage tristate inverter chain serves as the storage unit. During the reset phase, the rstb signal is used to reset voltages at the internal nodes of the TFF. During the write phase, input pulses are sent to the ring, which allows propagation of "0" through the ring with a duration of input pulses. Multiple input pulses can be repeatedly sent to the TFF and will be accumulated through propagation in the ring. During the readout phase, the stored pulse is sent from the output pin of the ring with pulse width equivalent to summation of the stored values. When the ring is filled, a carry signal rises and the ring will rotate back with remainder values stored inside. The "rotation" operation enables scalable operation in multi-bit groups. Simulations show the TFF can retain data for over 100ns, with less than 0.15b loss due to leakage. In addition, a minimum-pulse-generator circuit is used to create a removable offset to keep the pulse from being too narrow to be propagated. As opposed to conventional flip-flops, each TFF can process multibit signals. In this design, each TFF can store a 6b time domain signal and two TFFs are used to construct a 10b time-domain value separated into MSB and LSB units, leading to a wide TFF module (WTFF), as shown in Fig. 19.7.4. In the WTFF, once the LSB TFF is full, a carry signal is sent to a pulse generator to generate an extra pulse to be stored in the MSB TFF extending the operation to 10b. Fig. 19.7.2 also shows the MIN and ABS modules used in this work, consisting of only simple digital gates, e.g. NAND, rendering a 6× reduction compared with an equivalent digital implementation. A 4b digital-to-time converter (DTC) is implemented inside ABS to convert input digital values into time-domain pulses. The DTC consists of an inverter-based delay chain and multiplexers.

Figure 19.7.3 shows the pipelined DTW engine with 20×20 DTW unit cells and scalable operation to construct longer time series. The DTW matrix contains a group of DTW unit cells with a diagonal pipeline structure. The unit cell, depicted in Fig. 19.7.4, contains 2 WTFF modules, an ABS module, and a MIN module. The second WTFF module (marked in white) in the unit cell is used to copy the data from last pipeline stage, because the data stored in node $(i-1, j-1)$ is one pipeline stage earlier than the nodes $(i-1, j)$ and $(i, j-1)$. Due to the use of the TFF, in every clock cycle, the data pulses are propagated along the diagonal direction of the matrix. Fig. 19.7.3 also shows the data streaming flow for pipelined operation, where incoming digital data is fed from an on-chip register file (RF) and clock management unit. Each data item is piped through the DTW matrix as inputs to ABS modules both vertically and horizontally. The pipelined operation allows fixed dimensions of the DTW engine to be unfolded for longer data sequences, as shown in Fig. 19.7.3, ultimately limited by internal register storage capacity – 10b in this implementation. All output pulses from the bottom and right boundaries are decoded by shared time-to-digital converters (TDCs) every clock cycle and re-sent back for processing by subsequent sections. To speed up the operation for a simple data sequence, e.g. a DNA sequence, a non-pipelined mode is also possible by bypassing the TFF modules and allowing signal edges to directly propagate through the matrix. In addition, a 2b tunable delay cell is implemented in each unit cell to tune the output pulse width, compensating for process variations. A special DTW matrix calibration scheme is introduced to calibrate the DTW matrix. With calibration, the maximum DTW distance error drops from 5b to 1.5b.

Figure 19.7.4 shows the test chip implementation of the DTW engine in a 65nm 1V CMOS process. Two sets of TDCs, based on Vernier delay chains, are placed at the right and bottom sides to decode time-domain signals at the boundaries. A single-bit resolution of 40ps is used in the DTW design, while a resolution of 20ps is used in the TDC to reduce quantization errors at the boundary of operation. All the input and output data can be scanned in and out through a scan chain for verification.

Figure 19.7.5 shows measurement results. UCR time-series classification databases were used to test the architecture [4]. Five databases from four typical applications were selected. The DTW engine is configured in unfolded mode to adapt to the variable-input series length. The measured error rate for classification by the DTW engine is only 1.5% higher than ideal DTW operation (floating point results in software) mainly due to quantization (0.5%) and process variation (1%). 100 sets of DNA sequence data from the human genome database (GDB) were also tested for comparison between ideal DTW operation and measurement results [2]. The measured distance closely tracks the ideal results, having an error within 2.6%. The measured waveform (Fig. 19.7.6) confirms the expected output pulse at a frequency of 110MHz in pipelined mode and 3.1ns processing time in DNA-sequencing mode. The linearity of the implemented TFF is measured through on-chip TDC showing the proper storage of a time signal within 0.5b under the quantization limitation of the TDC. The chip was also verified at different supply voltages in pipelined mode down to 0.7V, with a 2.3% increase in error rate compared with ideal DTW operation on the UCR database. Fig. 19.7.6 shows the comparison with prior work. Throughput of 140 giga-cell-updates-per-second (GCUPS) for DNA sequencing is achieved with 9× improvement over previous work [2]. A significantly higher throughput per area (GCUPS/mm^2) is observed compared with prior CPU, GPU and ASIC implementations. Compared with non-pipelined operation, the pipelined design shows 7× improvement in throughput for general DTW applications. In addition, the use of time flip-flops enables a pipelined architecture for time-domain design [3]. The die micrograph is shown in Fig. 19.7.7.

References:

[1] Hui Ding, et al., "Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Measures," *Proceedings of VLDB*, pp. 1287-1300, 2008.

[2] Advait Madhavan, et al., "A 4-mm2 180-nm-CMOS 15-Giga-Cell-Updates-per-Second DNA Sequence Alignment Engine Based on Asynchronous Race Conditions," *CICC*, 2017.

[3] A. Amravati, et al., "A 55nm Time-Domain Mixed-Signal Neuromorphic Accelerator with Stochastic Synapses and Embedded Reinforcement Learning for Autonomous Micro-Robots," *ISSCC*, pp.124-126, 2018.

[4] UCR Archive, http://www.cs.ucr.edu/~eamonn/time_series_data

[5] M. Farrar, "Striped Smith–Waterman Speeds Database Searches Six Times Over Other SIMD Implementations," *Bioinformatics,* vol. 23, no. 2, pp. 156-161, 2007.

[6] Y. Liu, A. Wirawan, and B. Schmidt, "Cudasw++ 3.0: Accelerating Smith-Waterman Protein Database Search by Coupling CPU and GPU SIMD Instructions," *BMC Bioinformatics*, vol. 14, no. 1, 2013.

[7] N. Neves, et al., "Multicore SIMD ASIP for Next-Generation Sequencing and Alignment Biochip Platforms," *IEEE Trans. VLSI*, vol 23, no. 7, pp. 1287-1300, 2015.

 978-1-5386-8531-0/19/$31.00

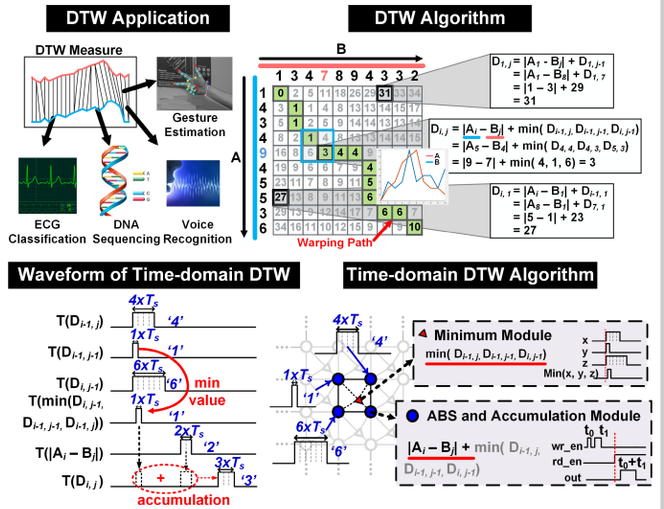


Figure 19.7.1: Application of DTW; concept of DTW algorithm; and, time-domain DTW implementation.

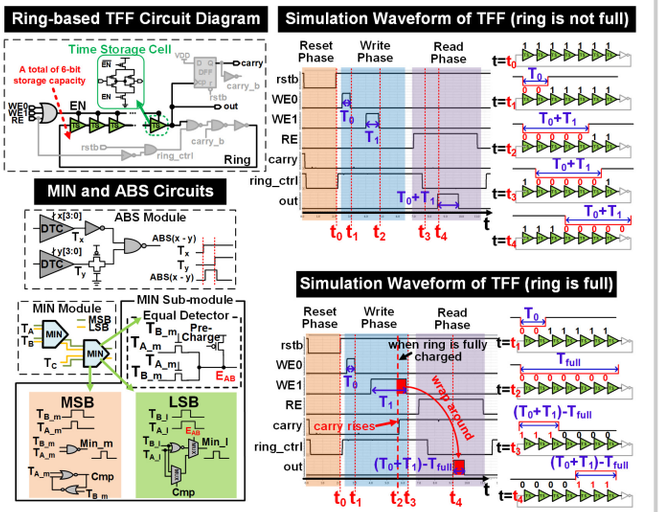


Figure 19.7.2: Circuit diagram and proposed ring-based time flip-flop, and other time-domain circuits.

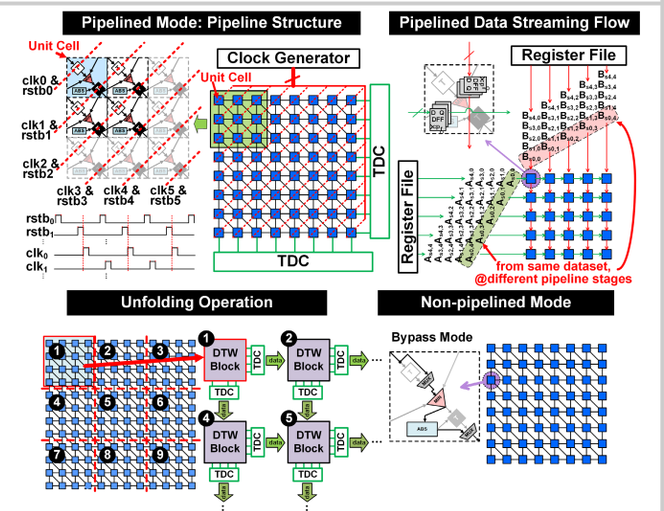


Figure 19.7.3: Pipelined mode of DTW engine; pipelined data streaming flow, unfolding for long sequences; and, non-pipelined mode of DTW engine.

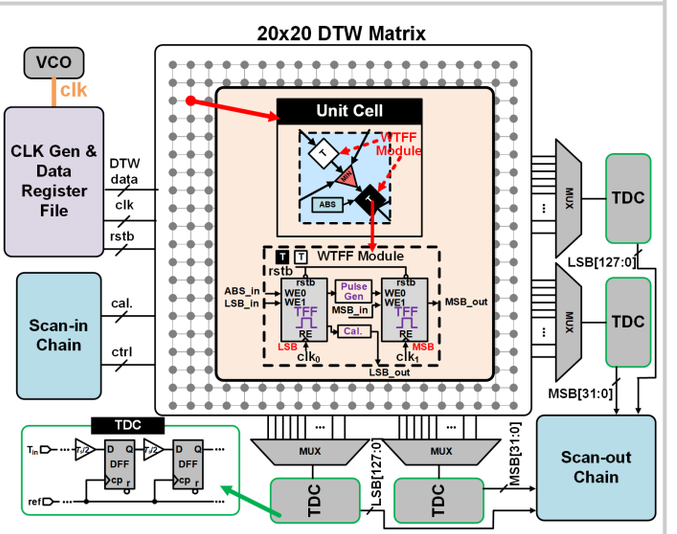


Figure 19.7.4: Overall test chip block diagram.

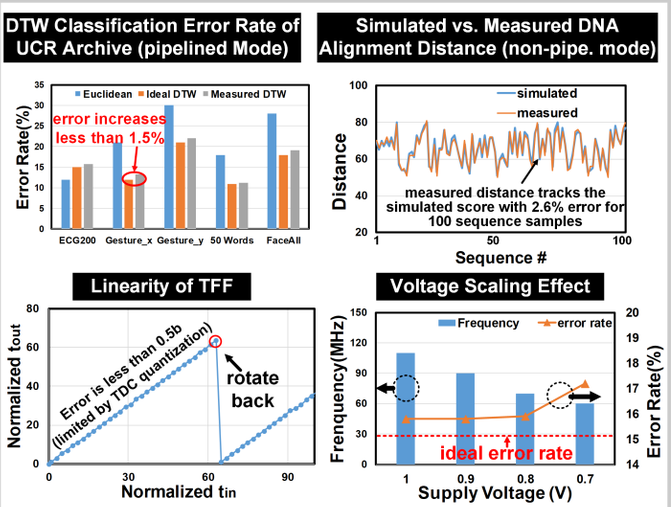
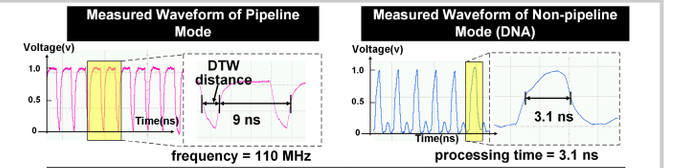


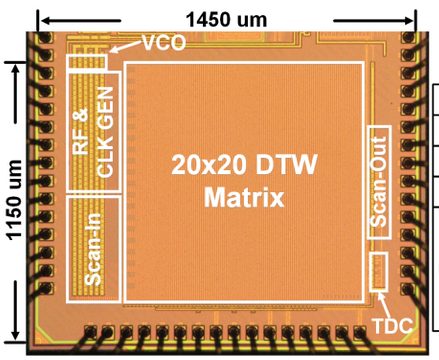
Figure 19.7.5: Measurement results of DTW classification databases, DNA sequencing; linearity of TFF; and, voltage scaling impact on the DTW engine.



	[5] Bioinformatics 2007	[6] Bioinformatics 2013	[7] TVLSI 2015	[2] CICC 2017	This work
Architecture	CPU	GPU	ASIC/ CPU	Time-domain ASIC	Time-domain ASIC
Process (nm)	65	28	90	180	65
Area (mm²)	143	300	6.4	4	1.67
Number of bit	floating point	floating point	32	1	4 (input) 10 (internal)
Power (mW)	9.5x10⁴	2x10⁵	2732	70	136 (pipeline) 35 (non-pipe.)
Clock period (GHz)	2	1	0.6	0.01	0.11 (pipeline)
Throughput for DNA sequencing (GCUPS)	3	119	9	15	140 *
Throughput per Area (GCUPS/mm²)	0.02	0.4	1.4	3.75	84
Throughput for general DTW App. (GCUPS)	-	-	-	-	71 (pipeline) 10 (non-pipe)
Error rate	-	-	-	2.9%	1.5~2.6%

* In DNA application, single bit non-pipeline mode is utilized for fair comparison with prior work.

Figure 19.7.6: Measurement waveform and performance comparison with prior work.



Technology			65nm
Die Area			1.67 mm^2
Frequency			110 MHz
Nom Vdd			1V
Power	DTW Core	Pipelined	111 mW
		Non-pipe.	20 mW
	RF& CLK GEN	Pipelined	25 mW
		Non-pipe.	15 mW

Figure 19.7.7: Die micrograph and specifications.

978-1-5386-8531-0/19/$31.00 ©2019 IEEE