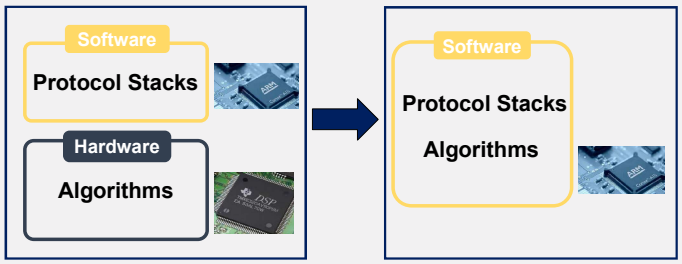
Low Complexity FFT Algorithm for Software-based LTE/5G OFDM Symbol Processing

Bitna Kim, Ji Yoon Han, Juyeop Kim
2019-11-16
College of Engineering
Division of Electronics Engineering

Contents

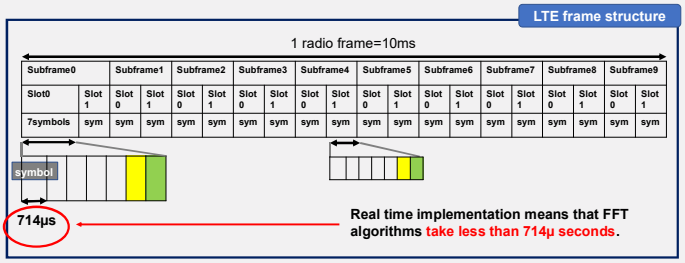
Software Defined Radio (SDR)

- Process through Software on a universal Central Processing Unit(CPU) instead of using Hardware or Digital Signal Processor(DSP)



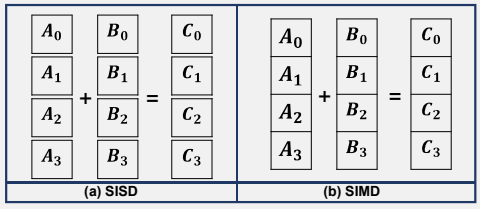
Necessity of Low-complexity FFT algorithm

- Necessary to implement a low-cost FFT algorithm that can perform OFDM symbol in real time



Single Instruction Multiple Data (SIMD)

- SIMD is something that can handle multiple data with one instruction which stands for Single Instruction, Multiple Data.
- Multiple data is a structure that has size of 4byte boundary like 64 bit (for MMX), 128bit(for SSE), 256 bit(for AVX). The data must be aligned on the memory and they can be processed by SIMD instruction at a time.



FFT Algorithm- FFT formula

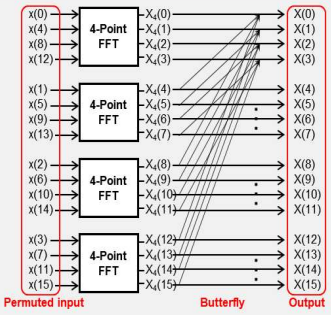
$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}$$

$$= \sum_{n=0}^{\frac{N}{4}-1} x(4n) * W_{N/4}^{nk} + W_{N/2}^{k} * \sum_{n=0}^{\frac{N}{4}-1} x(4n+2) * W_{\frac{N}{4}}^{nk} + W_N^k * \sum_{n=0}^{\frac{N}{4}-1} x(4n+1) * W_{\frac{N}{4}}^{nk} + W_N^k * W_{N/2}^k * \sum_{n=0}^{\frac{N}{4}-1} x(4n+3) * W_{\frac{N}{4}}^{nk}$$

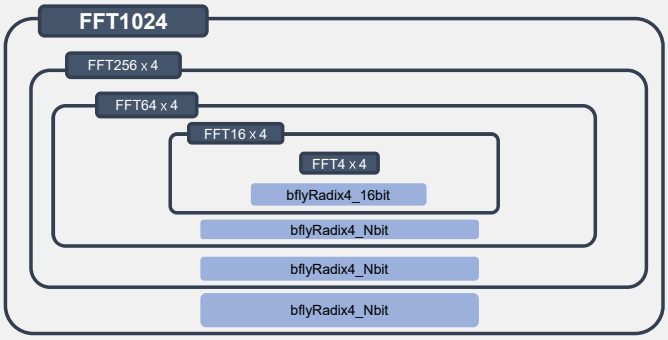
when **N=16** ,

$$1*(x(0) + x(4)*W_4^k + x(8)*W_4^{2k} + x(12)*W_4^{3k})$$
$$+W_{16}^k * (x(1) + x(5)*W_4^k + x(9)*W_4^{2k} + x(13)*W_4^{3k})$$
$$+W_{16}^{2k} * (x(2) + x(6)*W_4^k + x(10)*W_4^{2k} + x(14)*W_4^{3k})$$
$$+W_{16}^{3k} * (x(3) + x(7)*W_4^k + x(11)*W_4^{2k} + x(15)*W_4^{3k})$$

result of N=4



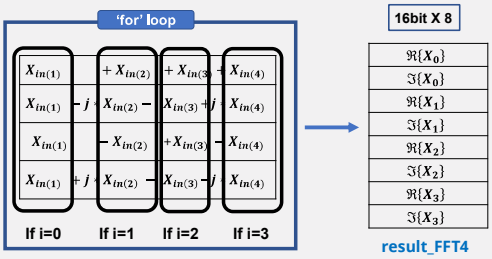
FFT Structure



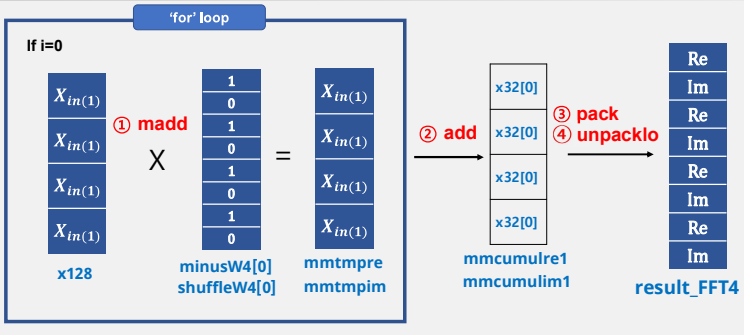
FFT Algorithm- FFT4

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} x_{in(1)} \\ x_{in(2)} \\ x_{in(3)} \\ x_{in(4)} \end{bmatrix} =$$

rxdata input

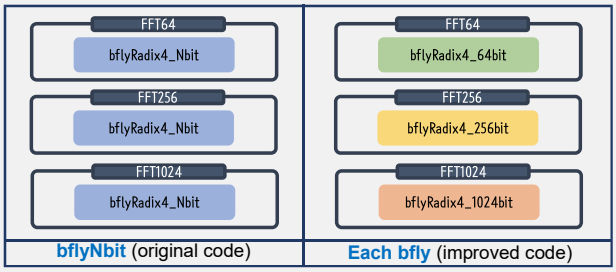


FFT Algorithm- FFT4



Improved FFT Algorithm- (1)

- Changed **bflyNbit** function, which was generalized in functions 64-point FFT, 256-point FFT, and 1024-point FFT according to N.



Improved FFT Algorithm- (2)

- Implemented a new 4-point FFT algorithm to reduce the use of **_mm_madd_epi16** as much as possible. Since 4-point FFT is called repeatedly in 1024-point FFT, the time is reduced.

Bfly Nbit :FFT4

31

→

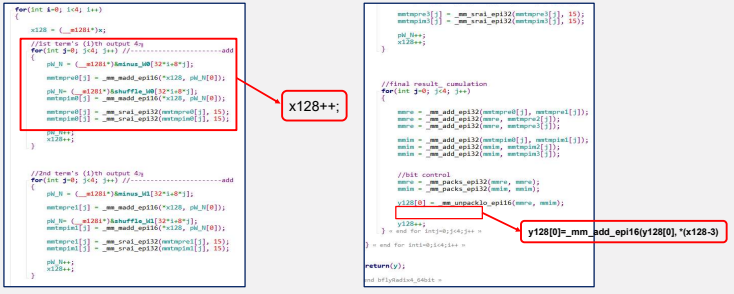
New FFT4 :FFT4

12

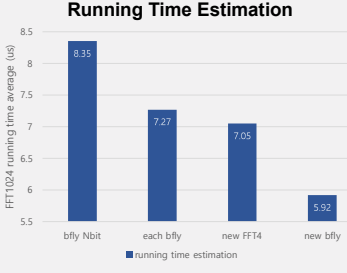
SIMD instruction	Estimation time [ns]
_mm_madd_epi16	0.1304
_mm_add_epi32	0.0887

Improved FFT Algorithm- (3)

- We found some unnecessary calculations in the code and changed them to be calculated only once.



Performance Analysis



Running time estimation				
	Bfly Nbit	Each bfly	New FFT4	New bfly
1024-point FFT (us)	8.35	7.27	7.05	5.92
Reduction Ratio(%)	Ref.	12.86	15.5	29.16

Conclusion

- Design and implement an FFT algorithm that can be processed in real time using the parallel processing function of the CPU.
- Confirmed that our FFT algorithm can achieve the LTE synchronization process in real time.
- By applying this, the 5G synchronization process will also be implemented in software, making it a key technology in V2X.