


Low-Power Booth Multiplication without Dynamic Range Detection in FFTs for FMCW Radar Signal Processing

Oğuz Meteer  *
 Email: o.meteer@utwente.nl

Marco J.G. Bekooij*[†]
 Email: marco.bekooij@nxp.com

*Department of Computer Architectures
 for Embedded Systems
 University of Twente, Enschede, The Netherlands

[†]Department of Embedded Software
 and Signal Processing
 NXP Semiconductors, Eindhoven, The Netherlands

Abstract—Multipliers in DSP applications usually consume a significant amount of power. Studies have shown that power efficiency of the often used Booth multiplier can be improved by dynamically swapping input operands based on the dynamic range of both inputs. However, this requires dynamic range detection logic, which increases the area and delay. Also, no studies have proposed multipliers with dynamic range detection that are larger than 16x16 bits. So far, no applications have been identified where statically swapping multiplier inputs leads to increased power-efficiency.

In this paper we show that statically swapping Booth multiplier inputs can greatly improve the power-efficiency of FFTs. In the case of automotive FMCW radar systems, where multidimensional FFTs are calculated, the intermediate and output values of FFTs are typically sparse. Therefore these values are always given to the partial product generating input of the Booth multipliers, instead of being given the twiddle factors.

RTL power-simulation results of a radix-2 FFT implementation using Booth multipliers with swapped inputs show an up to 28.69% decrease in power usage compared to a typical implementation without swapped inputs.

Index Terms—Low-power design, Signal processing systems, FFT, Automotive, FMCW

I. INTRODUCTION

Multiplication is a fundamental operation in digital signal processing (DSP) applications, where the modified Booth multiplier [8] is often used compared to array multipliers, because they generate fewer partial products, leading to a lower delay and therefore a higher maximum clock frequency. Their power usage is significantly influenced by the dynamic range of their inputs. Several studies have proposed Booth multiplier implementations with dynamic range detection hardware that swap the multiplicand and multiplier if it can decrease the switching activity. However, compared to conventional Booth multipliers, these proposed implementations are slower, larger, and only are up to 16x16 bits in size. In contrast, no applications have been identified where *statically* swapping the multiplier inputs leads to lower power usage.

In this paper we show that *statically* swapping the multiplier inputs can significantly increase the power-efficiency in Fast

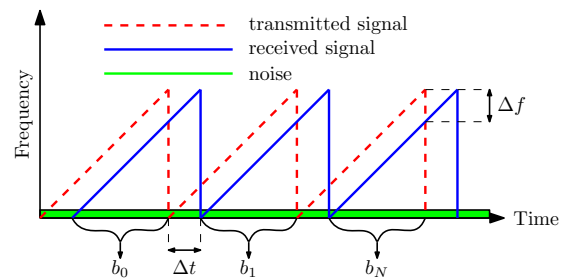


Fig. 1: Basic working principle of FMCW radar.

Fourier Transforms (FFT). We present a real-life, Frequency-Modulated Continuous-Wave (FMCW) radar signal processing use case involving multidimensional FFTs. The properties of the system allow us to always feed the twiddle factors of the FFT to the multiplicand input, and the input data to the multiplier input, saving power without dynamic range detection. In our gate level power simulations, our proposed FFT implementation with swapped Booth multiplier inputs uses up to 28.69% less power processing a real radar frame.

The paper is organized as follows. In Section II we explain the basic principles of FMCW radar and its properties. Then, in Section III we give a brief overview of the Discrete Fourier Transform (DFT) and FFT, and their processing gain. Section IV describes the modified Booth multiplier. In Section V, we list related work that achieve power reduction in Booth multipliers using dynamic range detection of operands. Section VI describes the basic idea of our proposed solution. Next, in Section VII, we evaluate both implementations, perform static power analysis, and list the power usage, area, and maximum frequencies. Section VIII addresses future work. Finally, we conclude the paper in Section IX.

II. FREQUENCY-MODULATED CONTINUOUS-WAVE RADAR

The basic principle of radar systems is to transmit a signal, then receive it after they reflect off of objects, and finally measure the time difference between transmitting and receiving the

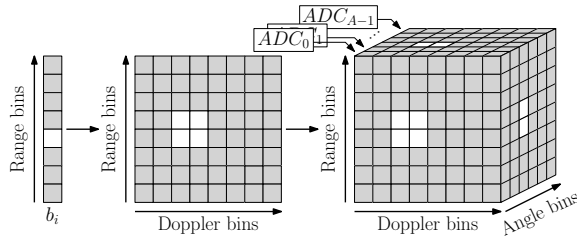


Fig. 2: Radar cube with range, Doppler, and angle information.

signal. This time difference, denoted as Δt , is proportional to the distance between the radar and objects that the signal reflects off of.

The type of signal that an FMCW radar system transmits is a *chirp*, which is a signal that increases in frequency over time. This allows for not only measuring the time difference Δt , but also the frequency difference Δf . Demodulation results in a sinusoidal signal called the *beat signal* b_i , which is obtained by mixing the transmitted and received signals. The beat signal contains frequencies that correspond with the reflections of objects and their distances R , which are extracted by using the FFT. Fig. 1 shows the basic workings of an FMCW radar system.

The proportion of the power of the transmitted and received signals is [4]:

$$P_{rx} \propto \frac{1}{R^4} P_{tx} \quad (1)$$

where P_{rx} is the power of the received signal, P_{tx} is the peak transmission power, and R is the distance to an object. We can see that, while P_{rx} has a high dynamic range, the power of the received signal is usually much smaller compared to the power of the transmitted signal.

A. Radar Data Cube

By applying a 3D FFT on the beat signals we can obtain a *Radar data cube* as shown in Fig. 2, which contains range, Doppler, and angle information. First we apply the FFT to all N beat signals b_i . This results in N columns containing the ranges of objects. Then we concatenate all columns and apply the FFT to all N rows, which results in the range-Doppler map. Finally, angle information is obtained by computing the range-Doppler map for every receiver antenna, and applying the FFT in the Z dimension.

We note that beat signals typically do not have a small amplitude as they contain sums of sinusoidal signals. Therefore the range FFTs are likely to process large signals, for example due to continuous strong reflections like a bumper or the radar dome. However, due to FFTs concentrating signal power in very few frequency bins, the outputs of range FFTs contain mostly noise and very small signals, even with large amplitude inputs. This is indicated in Fig. 2 with a grey color for noise and very weak signals, and white for strong signals. An additional property is that this not only applies to the output, but also applies to internal values of FFTs as well.

Since the output of the range FFTs mostly have very low amplitude, the Doppler and angle FFTs also process mostly very small signals. This effect is larger for range bins that correspond to targets that are increasingly farther away.

III. DISCRETE FOURIER TRANSFORM

For N -periodic discrete signals, the DFT extracts the discrete frequency components and their respective amplitudes. It can be seen as applying a band-pass filter for each output [7] which represents a bin for a specific frequency range. It is defined as:

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot W_N^{kn} \quad (2)$$

where $x[n]$ is the n th complex input sample, $X[k]$ is the k th transformed sample, and $W_N^{kn} = e^{-j \frac{2\pi kn}{N}}$ is called the *twiddle factor* [2] and is the principal N th complex root of unity.

A. Fast Fourier Transform

The most famous FFT uses the Cooley-Tukey algorithm [1] which reduces the complexity from $O(N^2)$ to $O(N \log_2(N))$. This is done by exploiting the symmetry of the twiddle factors and recursively decomposing the DFT into smaller parts. Given two input samples $x[i]$ and $x[j]$, the radix-2 FFT is defined as:

$$\begin{aligned} X[i] &= x[i] + x[j] \cdot W \\ X[j] &= x[i] - x[j] \cdot W \end{aligned} \quad (3)$$

where $X[i]$ and $X[j]$ are the transformed samples.

B. Processing Gain

The magnitude of a bin containing the signal is proportional to the FFT length N , whereas the magnitude of noise is proportional to \sqrt{N} . This interesting property, known as the *processing gain*, is defined as [7]:

$$SNR_N = SNR_{N'} + 10 \cdot \log_{10} \left(\frac{N}{N'} \right) \quad (4)$$

where N and N' are FFT lengths. To increase the SNR, we simply choose a bigger FFT length, i.e. $N > N'$.

When processing beat signals, far away objects have a higher frequency and as such are placed into higher bins. As a consequence of (1), their magnitudes are therefore much smaller, and can even be smaller than noise. The processing gain aids in extracting those small signals.

However, this comes at a cost as it requires more input samples and more operations to be executed. Also, the data path needs to be wide enough to store the larger values to not lose accuracy, increasing the number of logic gates and area of the implementation.

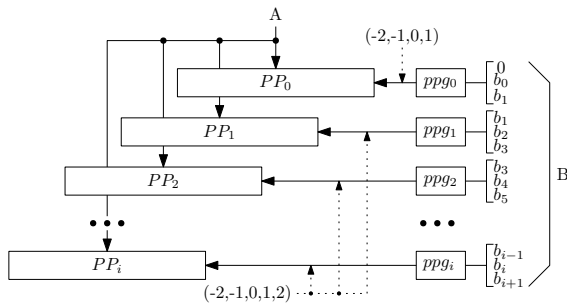


Fig. 3: Partial product generation in a radix-4 Booth multiplier.

TABLE I: Radix-4 Booth encoding

b_{2i+1}	b_{2i}	b_{2i-1}	P
0	0	0	+0
0	0	1	+A
0	1	0	+A
0	1	1	+2A
1	0	0	-2A
1	0	1	-A
1	1	0	-A
1	1	1	-0

IV. BOOTH MULTIPLIER

The modified Booth multiplier [8] with inputs A (multiplier) and B (multiplier), uses a radix-4 scheme to generate a partial product per groups of bits of B . The size of the group depends on the used radix, and in the radix-4 Booth multiplier, a partial product is generated per overlapping group of three bits. Table I lists the generated partial product based on these three bits of B (listed as b_{2i-1} , b_{2i} and b_{2i+1}). Fig. 3 shows how these partial products are generated. The first bit of the first group is always a zero. The LSB of each consecutive group is the MSB of the previous group. If the last group consists of less than three bits, then the MSB of B is sign-extended. Also, since the first bit of the first group is a zero, the first partial product cannot generate $2A$. In Table I, we see that three consecutive ones or zeroes produce a zero as a partial product, lowering the switching activity.

V. RELATED WORK

Multiple studies have looked into methods of determining whether A or B has a lower dynamic range (i.e. more groups that produce a zero partial product), so that it can be used as the B input.

Shen and Chen proposed a 16x16 bit radix-4 Booth multiplier, where they employed dynamic range determination units that partition the input into three groups of five bits, instead of groups of three bits [10]. This was done because it simplifies the implementation to a three input comparator. The downside of this approach is that it cannot detect all situations in which it would be beneficial to switch inputs, since the two extra bits that being compared in each group are unrelated to the main partial product generated in that group.

Park, Kim, and Lee proposed a novel data partitioning method where a 16x16 bit multiplier is split up into four 8x8

bit multipliers [3]. Since each multiplier deals with 8 bits of data, their dynamic range detection unit checks three groups of three bits. The benefit of their method is that each group only contains the exact bits that generate a partial product, resulting in accurate dynamic range detection. Also, the use of four multipliers means that there are more opportunities to switch the two inputs therefore having the potential to decrease the switching activity even further.

Kuang and Wang proposed a low-power, configurable Booth multiplier that can perform a single 16 bit, single 8 bit, or dual parallel 8 bit multiplication [5]. They use a novel dynamic range detector that not only increases the probability that partial products generate zeroes, but also attempts to avoid redundant switching activity in ranges that do not influence the result.

As an alternative to dynamic range detection, Meterer and Bekooij have shown that using radix-2 FFT units with custom sign-magnitude multipliers can lead to a reduction in power usage of up to 46.45% in automotive FMCW radar applications [9]. The downside is that they use a custom multiplier and their design is larger and has a 6.67% lower maximum clock frequency compared to a design using the Booth modifier proposed by Kuang, Wang, and Guo [6].

What previous studies have in common is that compared to conventional Booth multipliers, the proposed ones are larger and slower. Also, the multipliers with dynamic range detection only have a size of up to 16x16 bits. However, in our application, the processing gain of the FFT is used to detect signals far below the noise level, so a large dynamic range and data width is required, which makes previous attempts with dynamic range detection not suitable.

VI. BASIC IDEA

To decrease the dynamic power usage, we note the following observations. First, we want to extract signals far below the noise floor by using the processing gain of the FFT. Therefore we need a wide data path, which surpasses the size of state of the art multipliers with dynamic range detection.

Second, the amount of targets to detect compared to the number of bins is relatively small. Under these circumstances, the signal powers of the targets are quickly concentrated in a few bins, producing sparse outputs. This is due to the FFT essentially applying a band-pass filter to each bin [7]. Also, almost all the intermediate values have small amplitudes and low dynamic range as well [9].

Third, Booth multipliers are sensitive to the dynamic range of the partial product generating input, and a majority of values being processed have a very small dynamic range. We therefore propose a static setup where the twiddle factors are connected to the multiplicand input and the signal data is connected to the partial product generating inputs of the Booth multipliers. Thus, the power-efficiency is improved without using dynamic range detection hardware, and without a penalty to area or critical path length. Also, our proposed setup does not require any custom multipliers as used in [9].

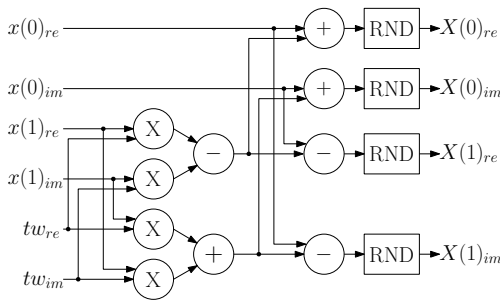


Fig. 4: Basic radix-2 FFT butterfly unit.

VII. EVALUATION

In this section we evaluate the reference and our proposed implementations using synthetic data and a full frame from an actual FMCW radar. Both designs implement a radix-2 FFT butterfly with a 32-bit wide data path and use the Booth multiplier as proposed by Kuang, Wang, and Guo [6]. Fig. 4 shows the design used for both implementations. The subtraction and addition of the complex multiplication use the full 64-bit outputs of the multipliers, and the final additions and subtractions are done with 32 bits, where the rounding hardware (RND) perform unbiased rounding. We show that our proposed design uses significantly less power.

Both implementations were synthesized using the TSMC 40nm LP standard cell library with a typical-typical corner and $V_{dd} = 1.1$ V. Synopsys tools were used for synthesis with *high* synthesis and mapping effort. As a reference, a software implementation of the FFT was written that uses Q2.30 fixed-point numbers, 12-bit input samples, and \sqrt{N} division (implemented as dividing by two each two stages of the FFT). Correctness of our reference software implementation was verified using the built-in FFT implementation in Matlab.

Two types of signals were used to test all implementations. First, two synthetic signals were generated, and a 1024-bin FFT was applied that uses a single radix-2 butterfly unit sequentially. The two signals contain two bits of normally distributed noise:

- 1) *Weak*: sine wave with amplitude $\frac{1}{4096} \approx 2.44 \cdot 10^{-4}$.
- 2) *Strong*: sine wave with amplitude $\frac{4000}{4096} \approx 9.7 \cdot 10^{-1}$.

Second, capturing a scene with several targets at different distances with an actual FMCW radar, we applied a 2D FFT to obtain a range-Dopple frame. The radar was configured to generate 512 chirps with 1024 samples per chirp, so 512 1024-bin FFTs were applied to obtain the range FFT. The input is a real-valued signal, meaning the upper half of the range FFT is symmetrical to the lower half, and was discarded. Therefore, to obtain the Doppler results, 512 512-bin FFTs were performed.

A. Results

The synthesis results are shown in Table II. The area and maximum clock frequency results for both the reference our proposed implementations are the same. This is expected since both implementations are structurally the same, and only the multiplier inputs are swapped.

TABLE II: Synthesis results for circuit area and maximum frequency.

Implementations	Max. Freq.	Area @	Area @
	(MHz) (Δ)	Max. Freq. (μm^2) (Δ)	344.8 MHz (μm^2) (Δ)
Reference	476.2 (-)	48113 (-)	32663 (-)
Proposed	476.2 (+0.0%)	48113 (+0.0%)	32663 (+0.0%)

The power results for both butterfly implementations processing synthetic data and a real radar frame are shown in Fig. 5a and Fig. 5b respectively. Compared to the reference implementation, our proposed implementation shows a significant decrease in power usage of up to 43.03% for synthetic data, and an up to 28.69% decrease for a real radar frame. Since the majority of the intermediate and output data of the FFT has low dynamic range, most partial products of the Booth multipliers generate zeroes. This not only decreases the switching activity in the multipliers, but also permeate less glitches throughout the adders and subtracters, improving the power efficiency even further.

VIII. FUTURE WORK

The results show that choosing the right operands for the inputs of the multipliers has a significant influence on the power-efficiency. Statically swapping the operands is a trivial operation, because we use the exact same multiplier hardware and the functional multiplication results do not change. In future work, it would be fruitful to find other applications, or even change existing algorithms where statically swapping inputs leads to improved power efficiency.

IX. CONCLUSION

In this paper we have proposed a modification to FFT butterfly units that use the modified Booth multiplier. We propose to statically swap the inputs to the multipliers when used in the context of FMCW radar systems.

Evaluation shows that our proposed implementation decreases the power usage up to 28.69% with a real radar frame. Our proposed modification has no penalty to the area or maximum clock speed.

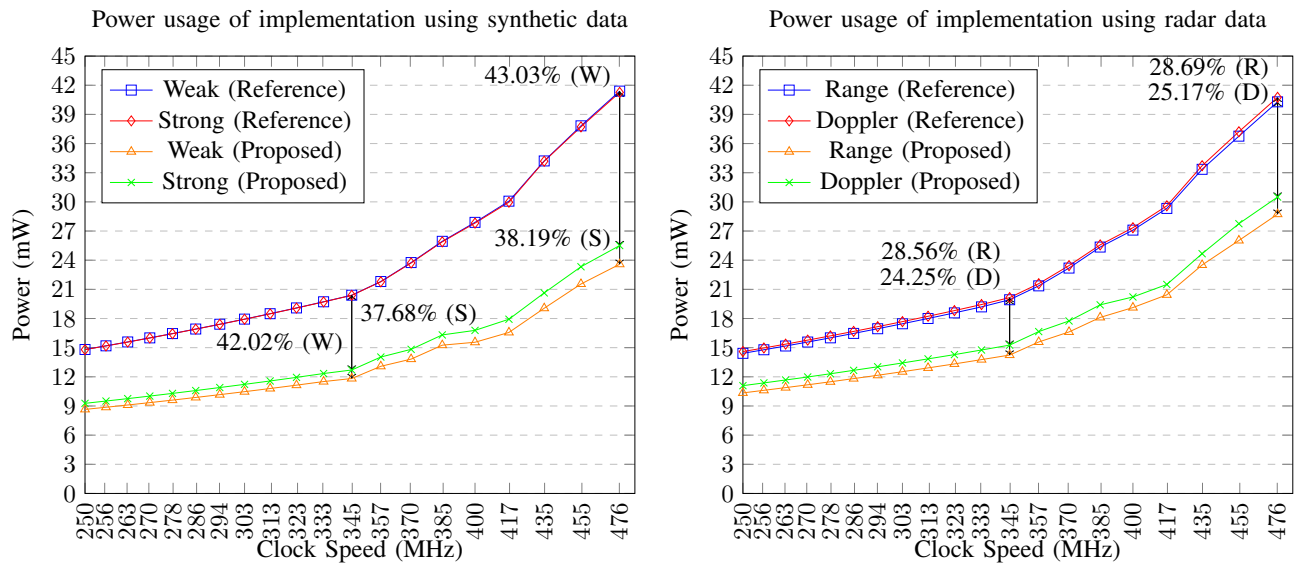
The results clearly show that in the context of FMCW radar signal processing, the FFT can be significantly more power efficient by simply swapping the multiplier inputs permanently.

ACKNOWLEDGMENT

This work is part of the research program Perspectief ZERO with project number P15-06 Project 3, which is (partly) financed by the Dutch Research Council (NWO).

REFERENCES

[1] James Cooley and John Tukey. "An Algorithm for the Machine Calculation of Complex Fourier Series". In: *Mathematics of Computation* 19.90 (1965), pp. 297–301.



(a) Power usage with synthetic data. The (W) and (S) indicators denote the power usage decrease percentages for weak and strong signals respectively.

(b) Power usage with radar data. The (R) and (D) indicators denote the power usage decrease percentages for range and Doppler signals respectively.

Fig. 5: Power usage of the reference and proposed designs with synthetic and radar data, synthesized for a range of clock speeds.

[2] W. M. Gentleman and G. Sande. "Fast Fourier Transforms: For Fun and Profit". In: *Proc. Fall Joint Computer Conference*. AFIPS '66 (Fall). San Francisco, California: ACM, 1966, pp. 563–578.

[3] Jongsu Park, San Kim, and Yong-Surk Lee. "A low-power Booth multiplier using novel data partition method". In: *Proceedings of 2004 IEEE Asia-Pacific Conference on Advanced System Integrated Circuits*. 2004, pp. 54–57.

[4] S. Kingsley and S. Quegan. "Understanding Radar Systems". In: *Electromagnetics and Radar*. SciTech Publishing, 1999, p. 11. ISBN: 9781891121050.

[5] S. Kuang and J. Wang. "Design of Power-Efficient Configurable Booth Multiplier". In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 57.3 (2010), pp. 568–580.

[6] S. Kuang, J. Wang, and C. Guo. "Modified Booth Multipliers With a Regular Partial Product Array". In: *IEEE Transactions on Circuits and Systems II: Express Briefs* 56.5 (2009), pp. 404–408.

[7] Richard G Lyons. *Understanding digital signal processing, 3/E*. Pearson Education India, 2011.

[8] O. L. Macsorley. "High-Speed Arithmetic in Binary Computers". In: *Proc. IRE* 49.1 (Jan. 1961), pp. 67–91. ISSN: 0096-8390.

[9] Oğuz Meteer and Marco J. G. Bekooij. "Low-Power Sign-Magnitude FFT Design for FMCW Radar Signal Processing". In: *Workshop on Design and Architectures for Signal and Image Processing (14th Edition)*. DASIP '21. Budapest, Hungary: Association for Computing Machinery, 2021, pp. 52–59. ISBN: 9781450389013. DOI: 10.1145/3441110.3441145. URL: <https://doi.org/10.1145/3441110.3441145>.

[10] Nan-Ying Shen and O. T. -. Chen. "Low-power multipliers by minimizing switching activities of partial products". In: *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No.02CH37353)*. Vol. 4. 2002, pp. IV–IV.