An Efficient realization Area-Time with Multi Constant Multiplications for An Fine-Grained Critical Path Analysis and Optimization

Eswar Amuru¹, K. Bala²

¹M.Tech, Dept of ECE, Srinivasasa Institute of Technology and Science, Ukkayapalli, kadapa, India
²Associate Professor, Dept of ECE, Srinivasasa Institute of Technology and Science, Ukkayapalli, kadapa, India

ABSTRACT

In this paper, critical path of multiple constant multiplication (MCM) block is analyzed precisely and optimized for high-speed and low-complexity implementation. A delay model based on signal propagation path is proposed for more precise estimation of critical path delay of MCM blocks than the conventional adder depth and the number of cascaded full adders. A dual objective configuration optimization (DOCO) algorithm is developed to optimize the shift-add network configuration to derive high-speed and low-complexity implementation of the MCM block for a given fundamental set along with a corresponding additional fundamental set. A genetic algorithm (GA)-based technique is further proposed to search for optimum additional fundamentals. In the evolution process of GA, the DOCO is applied to each searched additional fundamental set to optimize the configuration of the corresponding shift-add network. Experimental results show that the proposed GA-based technique reduces the critical path delay, area, power consumption, area delay product and power delay product by 32.8%, 4.2%, 5.8%, 38.3%, and 41.0%, respectively, over other existing optimization methods.

Keywords: Efficient Distributed Arithmetic (DA), Finite Impulse Response (FIR), Look-up-table(LUT).

1. INTRODUCTION

Finite impulse response (FIR) filters are of great importance in digital signal processing (DSP) systems since their characteristics in linear-phase and feed-forward implementations make them very useful for building stable high-performance filters. Although both architectures have similar complexity in hardware, the transposed form is generally preferred because of its higher performance and power efficiency. The multiplier block of the digital FIR filter in its transposed where the multiplication of filter coefficients with the filter input is realized, has significant impact on the complexity and performance of the design because a large number of constant multiplications are required. That is often called the multiple constant multiplications (MCM) operation and can be a significant operation and performance bottleneck in many other DSP programs equivalent to fast Fourier transforms, discrete cosine transforms (DCTs), and blunder-correcting codes. Even though field-, delay-, and energy-efficient multiplier architectures, corresponding to Wallace and modified sales space multipliers, had been proposed, the whole flexibility of a multiplier is just not imperative for the steady multiplications, given that filter coefficients are fixed and determined beforehand by the DSP algorithms.

Hence, the multiplication of filter coefficients with the input data is generally implemented under a shift addition architecture where each constant multiplication is realized using addition/subtraction and shift operations in an MCM operation. For the shift-adds implementation of constant multiplications, a straightforward method, generally known as digit based run at the gate level. In this paper, we initially determine the gate-level implementation costs of digit-serial addition, subtraction, and left shift operations used in the shift-adds design of digit-serial MCM operations. Then, we introduce the exact CSE algorithm that formalizes the gate-level area optimization problem as a 0–1 integer linear programming (ILP) problem when constants are defined under a particular number representation. We also present a new optimization model that reduces the 0–1 ILP problem size significantly and, consequently, the runtime of a generic 0–1 ILP solver. Since there are still instances which the exact CSE algorithm cannot handle, we describe the approximate GB algorithm that iteratively finds the “best” partial product which leads to the optimal area in digit-serial MCM design at the gate level.

This paper also introduces a computer-aided design (CAD) tool called SAFIR which generates the hardware descriptions of digit-serial MCM operations.
and FIR filters based on a design architecture and implements these circuits using a commercial logic synthesis tool. In SAFIR, the digit-serial constant multiplications can be implemented under the shift adds architecture, and also can be designed using generic digit serial constant multipliers. Experimental results on a comprehensive set of instances show that the solutions of algorithms introduced in this paper lead to significant improvements in area of digit-serial MCM designs compared to those obtained using the algorithms designed for the MCM problem. The digit-serial FIR filter designs obtained by SAFIR also indicate that the realization of the multiplier block of a digit-serial FIR filter under the shift adds architecture significantly reduces the area of digit-serial FIR filters with respect to those designed using digit-serial constant multipliers Additionally, it is observed that the optimal tradeoff between area and delay in digit-serial FIR filter designs can be explored by changing the digit size d. In the last two a long time, many effective algorithms and architectures had been introduced for the design of low complexity bit-parallel a multiple constant multiplications (MCM) operation which dominates the complexity of many digital signal processing techniques. Then again, little awareness has been given to the digit-serial MCM design that offers substitute low complexity MCM operations albeit at the cost of an elevated lengthen. In this paper, we address the problem of optimizing the gate-level area in digit-serial MCM designs and introduce high level synthesis algorithms, design architectures by using Verilog HDL and FPGA Spartan3.

II. LITERATURE SURVEY


The problem of an efficient hardware implementation of multiplications with one or more constants is encountered in many different digital signal-processing areas, such as image processing or digital filter optimization. In a more general form, this is a problem of common subexpression elimination, and as such it also occurs in compiler optimization and many high-level synthesis tasks. An efficient solution of this problem can yield significant improvements in important design parameters like implementation area or power consumption. In this paper, a new solution of the multiple constant multiplication problem based on the common subexpression elimination technique is presented. The performance of our method is demonstrated primarily on a finite-duration impulse response filter design. The idea is to implement a set of constant multiplications as a set of add-shift operations and to optimize these with respect to the common subexpressions afterwards. We show that the number of add/subtract operations can be reduced significantly this way. The applicability of the presented algorithm to the different high-level synthesis tasks is also indicated. Benchmarks demonstrating the algorithm's efficiency are included as well.


A modified reduced adder graph (MRAG) Algorithm and its hybrid variant are proposed for effective digital filter implementation. A few upgrades are made to exploit absolutely the efficient surest part of the n-dimensional decreased adder graph (RAG-n) algorithm. Simulation results display MRAG is in a position of generating cut back adder rate.


Multiple constant multiplication (MCM) scheme is widely used for implementing transposed direct-form FIR filters. While the research focus of MCM has been on more effective common sub expression elimination, the optimization of adder-trees, which sum up the computed sub-expressions for each coefficient, is largely omitted. In this paper, we have identified the resource minimization problem in the scheduling of adder-tree operations for the MCM block, and presented a mixed integer programming (MIP) based algorithm for more efficient MCM-based implementation of FIR filters. Experimental result shows that up to 15% reduction of area and 11.6% reduction of power (with an average of 8.46% and 5.96% respectively) can be achieved on the top of already optimized adder/subtractor network of the MCM block.

Multiple constant multiplications: efficient and versatile framework and algorithms for exploring common subexpression elimination Author: Potkonjak, M. Srivastava, M.B.; Chandrakasan, A.P.

Many applications in DSP, telecommunications, graphics, and control have computations that either
involve a large number of multiplications of one variable with several constants, or an easily be transformed to that form. A proper optimization of this part of the computation, which we name the more than one consistent multiplication (MCM) quandary, mostly results in a huge development in a couple of key design metrics, such as throughput, subject, and energy. Nevertheless, until now little awareness has been paid to the MCM predicament.

After defining the MCM difficulty, we introduce an potent challenge formula for solving it where first the minimum number of shifts that are needed is computed, after which the quantity of additions is minimized making use of fashioned sub expression removing.

The algorithm for original sub expression removal is founded on an iterative pairwise matching heuristic. The energy of the MCM approach is augmented by using preprocessing the computation structure with a brand new scaling transformation that reduces the number of shifts and additions.

An effective branch and sure algorithm for applying the scaling transformation has additionally been developed. The flexibility of the MCM obstacle formulation enables the appliance of the iterative pair shrewd matching algorithm to a couple of other primary and normal high stage synthesis duties, such as the minimization of the quantity of operations in steady matrix-vector multiplications, linear transforms, and single and more than one polynomial critiques. All functions are illustrated with the aid of a number of benchmarks

Number-Splitting with Shift-and-Add Decomposition for Power and Hardware Optimization in Linear DSP Synthesis

Author: Huy T. Nguyen and Abhijit Chatterjee

Most DSP synthesis tools perform limited architectural transformations to optimize hardware and power. Multiplications are mostly carried out with shift-and-add operations for hardware affectivity.

In this paper, we endorse an optimization that combines a numerical transformation referred to as quantity-splitting with a shift-and-add decomposition scheme. The numerical transformation “globally” changes the regular multipliers and the data drift-graph of the process under design, enabling implementations with fewer shifts and adds.

The decomposition of multiplications into shifts and provides is such that as a lot intermediate computation results (partial merchandise) will also be reused as viable. The total number of operations can be diminished to 30% for two’s complement encoding, yielding big vigour and hardware saving.

Global Optimization of Common Subexpressions for Multiplierless Synthesis of Multiple Constant Multiplications

Author: Yuen-Hong Alvin Ho, Chi-Un Lei, Hing-Kit Kwan and Ngai Wong

In the context of multiple constant multiplication (MCM) design, we advise a novel common sub expression elimination (CSE) algorithm that units the best synthesis of coefficients right into a 0-1 mixed-integer linear programming (MILP) hindrance. A time prolong constraint is included for synthesis. We additionally propose coefficient decompositions that mix all minimal signed digit (MSD) representations and the shifted sum (change) of coefficients. Within the examples we reveal, the proposed answer house additional reduces the quantity of adders/subtractors within the MCM synthesis.

Efficient Bit-Serial Constant Multiplication for FPGAs

Author: Florian Dittmann Bernd Kleijn Johann Achim Rettberg

This paper describes how to realize place-effective synchronous bit-serial constant multiplications, which can be efficiently used for a block of constants (Multiple Constant Multiplication). The architecture combines traditional concepts and new approaches, which leads to the possibility of simultaneous fast multiplications of different input values. Rapid multiplications are a core for updated embedded systems, as they rely on fast input processing which may also be supplied via fashioned transformations. As an example, we have realized the Discrete Cosine Transformation (DCT), the place we can show that our structure points an top of the line alternate-off between discipline and velocity.

III. CSE AND GB ALGORITHM

An algorithm for efficient solution of the multiple constant multiplication problems. Common sub expression elimination (CSE) as a way to tackle the MCM problem was already proposed by various authors primarily as a possible method for the optimization of finite-duration impulse response (FIR) filter area through the reduction of the
multiplier block logic a number of other applications in which the MCM transformation can be successfully applied were proposed. In this work, we will introduce an algorithm able to solve the CSE problem in an efficient way. The idea of CSE can be demonstrated on a FIR filter design. The optimization procedure targets the minimization of the multiplier block area. After expressing the coefficients in a canonical signed digit (CSD) format in order to reduce the total number of nonzero bits (thus also the additions/subtractions necessary), an add shift expansion is performed. The goal of CSE is to identify the bit patterns that are present in the coefficient set more than once. Since it is sufficient to implement the calculation of the multiple identical expressions only once, the resources necessary for these operations can be shared.

A. EXISTING SYSTEM

Multiple constant multiplication (MCM) constitutes a typical fixed-point arithmetic operation in digital signal Processing. It is the focus of a lot of research on high-speed and low power devices in communication systems and signal processing systems. In multiplier less MCM, multipliers are replaced by simpler components such as adders and hard-wired shifts (adders in our paper include also subtractors as their hardware costs are similar). By using the

Negative digits (subtractor in circuit) in their signed-digit representations, coefficients may be synthesized with fewer adders; therefore the area and power consumption of the circuit can be reduced. An example of a multiplier-based and a multiplier less based MCM implementations respectively, wherein 4 multiplications are replaced by 6 adders and 6 hard-wired shifts. Such

Multiplier less MCMs are utilized, for example, in the design of finite-impulse response Filters.

B. PROPOSED SYSTEM

EXACT GB ALGORITHM:

The optimization of gate-level area problem in digit-serial MCM design is an NP-complete problem due to the NP-completeness of the MCM problem. Thus, naturally, there will always be 0–1 ILP problems generated by the exact CSE algorithm that current 0–1 ILP solvers find difficult to handle. Hence, the GB heuristic algorithms, which obtain a good solution using less computational resources, are indispensable. In our approximate algorithm called MINAS-DS, as done in algorithms designed for the MCM problem given in Definition 1, we find the fewest number of intermediate constants such that all the target and intermediate constants are synthesized using a single operation. However, while selecting an intermediate constant for the implementation of the not yet synthesized target constants in each iteration, we favor the one among the possible intermediate constants that can be synthesized using the least hardware and will enable us to implement the not-yet synthesized target constants in a smaller area with the available constants. After the set of target and intermediate constants that realizes the MCM operation is found, each constant is synthesized using an A-operation that yields the minimum area in the digit-serial MCM design. The area of the digit-serial MCM operation is determined as the total gate-level implementation cost of each digit-serial addition, subtraction, and shift operation under the digit size parameter d as described in Section II-D. The preprocessing phase of the algorithm is the same as that of the exact CSE algorithm, and its main part and routines are given.

IV. DESIGN AND IMPLEMENTATION

Multiple Constant Multiplication (MCM) is an arithmetic operation that multiplies a set of fixed-point constants with the same fixed-point variable X. From a circuit point of view, MCM dominates the complexity of the whole category of Linear Time Invariant (LTI) systems, such as, FIR/IIR filters, DSP transforms (DCT, DFT, Walsh, etc), LTI controllers, crypto-systems, etc. To be efficiently implemented, MCM must avoid costly multipliers. The hardware alternative must be multiplier less, i.e., using only additions, subtractions, and shifts. Therefore, the MCM problem is defined as the process of finding the minimum number of addition/subtraction operations. The computational complexity of MCM is conjectured to be NP-hard.
Because of the increasing demand in high-speed and low-power design, MCM problem has been the focus of important researches during these last three decades. As a result, a big number of MCM algorithms have been proposed, mostly based on the acyclic directed graphs or common sub expression elimination or the combination of both together.

Typical DSP algorithms involve large number of multiplications, and multipliers consume significant amount of area and computation time. It is therefore important to reduce the area and time complexity of implementations of multipliers. In some applications such as linear transformations and transposed form finite impulse response (FIR) filters as illustrated in figure of acyclic graph the same variable is multiplied by a set of constant coefficients, which are known a priori. Such structures are referred to as multiple constant multiplications (MCM). Efficient implementation of MCM is important for high-speed, low-complexity and low-power DSP systems. Typically the multiplications in an MCM block are realized by a network of adders (subtractors) and hardwired shifts with sharing of partial products across all the multiplications. The MCM problem is extensively studied and many different algorithms have been proposed by researchers to optimize the area consumption and computation time of the MCM block. All these approaches can be put into two broad categories: i) the common sub expression elimination (CSE) technique proposed by Hartley, and ii) the graph-dependence (GD) algorithms.

The GD algorithms share the same optimal part and use different criteria to determine additional fundamentals. While the earlier GD algorithms, focus on reducing the number of additional fundamentals (i.e., word-level adders), in some recent works, the bit-level information of word-level adders has been taken into consideration to minimize the number of full adders. To optimize the bit-level complexity suggest a two-step design path as shown in Fig 4.3. Besides finding the additional fundamentals as those word-level techniques, the interconnection of DAG, referred to as shift-add network configuration (second step in Fig.), is further optimized to minimize the FA cost.

**Figure: Directed acyclic graph of constants multipliers of 23 and 49**

In this section, the bit-level complexity model for the MCM problem based on operations discussed and the shift-add network configuration optimization technique for reducing hardware complexity is introduced. Prior to that, the definition of operation is reviewed for the convenience of readers.

The notation of operation was first introduced in for a parameterized representation of adds, subtracts and shifts for MCM implementation. The formal definition of operation is as follows: let be integers and . An operation is defined as

\[
A_p(u, v) = 2^{|u|} + (-1)^{|v|} 2^{2|v|} 2^{-r}
\]

(1)
Where and are two previously realized fundamentals. is the parameter set, where and are, respectively, the bit positions through which and are shifted to left, is the bit positions through which and both are shifted to right, and indicates if the shifted fundamentals are added or subtracted. The result of an -operation is a fundamental from the given fundamental set to be realized or an additional fundamental that helps to realize the given fundamentals. We define as the configuration set of an -operation. Since MCM algorithms deal only with multiplications by positive odd integers, certain constraints can be applied to the -operation. In the MCM problem, there are only two kinds of possible -operations: i) one of the input fundamentals, say, is left shifted by one or more bit positions, while the value of the other input fundamental does not change (otherwise the result is not odd). ii) Both input fundamentals and are right shifted with the same amount of bit positions. This constraint can be expressed as:

\[
\begin{aligned}
 & l_1 > 0, l_2 = r = 0 \quad \text{(only } u \text{ is left shifted), or} \\
 & l_1 = l_2 = 0, r > 0 \quad \text{(both } u \text{ and } v \text{ are right shifted)}
\end{aligned}
\] (2)

BIT-LEVEL COMPLEXITY MODEL

Based on the -operation, a complexity model is recently proposed for ripple-carry adders (RCAs) [18]. The complexity model counts the number of full adders needed to realize a word-level adder. In this paper, we use the sign transformation technique of [21] to eliminate half adders in subtract operations. Half adders used for the last bit of word-level adders are counted as full adders. The number of full adders, denoted as \(N_{fa}\), needed to realize an adder in the multiplier block is estimated to be:

\[
N_{fa} = \begin{cases} 
\lfloor \log_2 \left( \frac{f_1}{f_0} \right) \rfloor - l_1 + w_{in} & \text{for } l_1 > 0, l_2 = r = 0 \\
\lfloor \log_2 \left( \frac{f_1}{f_0} \right) \rfloor + w_{in} & \text{for } l_1 = l_2 = 0, r > 0 
\end{cases}
\] (3)

Where \(f_1\) is the fundamental to be realized, and \(w_{in}\) is the word length of the input to the multiplier block. The total number of full adders needed to implement the MCM block, denoted as \(TotalFA\), is \(TotalFA = \sum N_{fa}\).

\[
\begin{aligned}
(a) & \quad (b) \\
(c) & \quad (d)
\end{aligned}
\]

Figure: Different ways to implement coefficient set \(\{3, 13\}\).
Optimization of Shift-Add Network Configuration for Reduction of Hardware Complexity

for a given fundamental set and a corresponding additional fundamental set, several different DAGs with different FA costs can be constructed represent the MC M block. In the following example we show that the shift-add network configuration of the DAG can be optimized to reduce the FA cost. Supposing that an 8-bit input is multiplied by a coefficient set \{3, 13\} using a shift-add network, there are 4 different options to realize the MCM block as shown in below Figure. The FA costs for DAGs of (a), (b), (c), and (d) are 17, 16, 19, and 18, respectively. Different shift-add network configurations result in different FA costs. In [21], a greedy algorithm was proposed to optimize the shift-add network of the DAG to reduce the number of full adders. It could be stated as follows: during the construction of the DAG, each time only the fundamental (no matter if it is a given or additional fundamental) which requires the fewest number of full adders according to (3) is realized. This algorithm is referred to as reduced full adder (RFA) algorithm in this paper.

B. SIGNAL PROPAGATION PATH BASED CRITICAL PATH OPTIMIZATION

While the area complexity is estimated at bit-level, the CPD in our paper is estimated based on the signal propagation path, because in a 1-bit full adder, the delays for different paths may be different. For this reason, a fine-grained signal propagation path based delay model is proposed for the analysis of critical path of MCM blocks. Thereafter, a shift-add network configuration optimization scheme is developed for reducing CPD as well as hardware complexity.

SIGNAL PROPAGATION PATH BASED CRITICAL PATH ANALYSIS

In the bit –level delay model proposed in the computation time of the sum and carry-out of a full adder is assumed to be the same. However, a closer look at the full adder reveals that this is not the true case in real digital circuits. Let us take the full adder in below figure.

Furthermore, the loads driven by the outputs of a full adder impact the delays as well. Thus, instead of assuming the computation time of the sum and carry out to be the same, we treat the delays of different signal paths in a full/half adder individually.

Figure (a) Full adder. (b) Two consecutive additions

V. EXPERIMENTAL RESULTS

In this section, numerical results are presented to demonstrate the effectiveness of the proposed GA-based technique for reducing the CPD as well as hardware complexity of MCM blocks. The efficiency of our proposed approach is compared with several CSE algorithms and GD-based algorithms in terms of area, time complexity, and power consumption. We have taken only positive odd integers as coefficients since even values can be realized by shifting an odd integer to the left and multiplying by negative coefficients can be subtracted instead of being added. An input word-length of 8-bit is assumed for all the design examples. In this experiment, eight commonly referenced benchmark FIR filter coefficient sets are tested. The specifications of all these benchmark filters are listed, where denotes the word-length of the filter coefficients, and # Taps is the number of filter taps. The information regarding types of filters is given in the Type column, where LP and HP, respectively, denote low-pass and high-pass filters. The last two columns list the passband edge and the stopband edge of the benchmark filters.

FIGURE: WAVEFORMS OF MCM
VI. CONCLUDING REMARKS

In this project, the critical path of MCM blocks are analyzed based on the signal path and a fine-grained delay model for CPD estimation is proposed. Based on this precise estimate of path, we proposed an algorithm named DOCO to optimize the shift-add network configuration of MCM blocks for the reduction of CPD and hardware complexity subject to an additional fundamental set. In this order to find the optimum additional fundamentals for a given fundamental set, a GA-based search method is proposed. The DOCO algorithm is adopted in the proposed GA-based technique to optimize the shift-add network configurations. Experimental results show that solutions generated by the proposed GA-based technique outperform existing algorithms in terms of CPD, area, power consumption, ADP and PDP. The CPD, area, power, ADP, and PDP are reduced by 32.8%, 4.2%, 5.8%, 38.3%, and 41.0%, respectively, in average over the existing algorithms.

VII. REFERENCES


Author’s Biography

Mr. Eswar Amuru received his B.Tech degree from the department of Electronics and Communication Engineering, from MADINA Engineering College (Affiliated to JNTU ANANTAPUR). He is pursuing M.Tech in SRINIVASA Institute of Technology & Science (Affiliated to JNTU ANANTAPUR), Ukkayapalli, KADAPA, AP, India. His current research interests are VLSI SYSTEM DESIGN.

Mr. K. Bala is currently working as an associate professor in ECE Department, Srinivasa Institute of Technology and Science, Ukkayapalli, Kadapa, India. He received his M.Tech from Sri Kottam Tulasi Reddy Memorial college of Engineering Kondair, Mahaboobnagar, A.P., India.