Troncoso Romero et al. EURASIP Journal on Advances in Signal Processing 2014, 2014:122
http://asp.eurasipjournals.com/content/2014/1/122

EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

On the inclusion of prime factors to calculate the theoretical lower bounds in multiplierless single constant multiplications

David E Troncoso Romero[1*], Uwe Meyer-Baese[2] and Gordana Jovanovic Dolecek[1]

Abstract

This paper presents an extension to the theoretical lower bounds for the number of adders and for the adder depth in multiplierless single constant multiplications (SCM). It is shown that the number of prime factors of the constants is key information to extend the current lower bounds in certain cases that have not yet been exposed. Additionally, the hidden theoretical lower bound for the number of adders required to preserve the minimum adder depth is revealed.

Keywords: Multiplierless; Single constant multiplication (SCM); Canonic signed digit (CSD), Directed acyclic graph (DAG); Common subexpression elimination (CSE); Minimized adder graph (MAG)

1 Introduction

Abundant research has been realized since the past two decades to solve single constant and multiple constant multiplication problems (SCM and MCM, respectively), where the hardware requirements can be reduced by exploiting the constant coefficient characteristics known *a priori* [1-18]. In these cases, multiplications are performed without using general multipliers and the unique arithmetic operations are additions and subtractions. In addition to these operations, only scaling by powers of two is allowed. These powers of two are implemented using hardwired shifts and therefore are considered with no cost.

Even though MCM and SCM problems are considered closely related, the latter constitutes the fundamental basis for all the constant multiplication problems and is usually solved by specialized, fine-tuned algorithms [1,3-5,9-11,14]. The usual metric to minimize in the SCM algorithms has been the number of arithmetic operations needed to implement the constant multiplier. However, it has been reported that the number of sequentially connected arithmetic operations forming a critical path has the main negative impact in performance and power consumption [3,7-9,15-18]. This, currently, has led to substantial research activity targeting both, application-specific integrated circuits (ASICs) [15,16] and field-programmable gate arrays (FPGAs) [17,18], where the minimization of the number of arithmetic operations subject to a minimum critical path is the ultimate goal.

Theoretical lower bounds for these two metrics in SCM, MCM, and other related problems have been derived in [12] based on a simple number that can be calculated in advance from the involved constants, namely, their minimum number of signed digits (MNSD). In [12], the theoretical lower bounds were first obtained for the simplest case of constant multiplications, namely, SCM, and then deduced for cases involving multiple constants. Nevertheless, the compromise between the number of arithmetic operations and the critical path was only mentioned and the theoretical lower bound for such compromise still remained hidden. It has been pointed out in literature that using the lowest critical path often results in higher performance and lower power consumption at expenses of increasing the number of arithmetic operations. Similarly, the minimization of the number of arithmetic operations may result in higher area saving at expenses of an increased critical path [3,7-9,15,16].

This paper introduces the extension to the theoretical lower bounds given in [12] for the SCM case. This extension shows that these bounds can be increased in

* Correspondence: dtroncoso@inaoep.mx
[1]Department of Electronics, National Institute of Astrophysics, Optics and Electronics, Tonantzintla, Puebla 72840, Mexico
Full list of author information is available at the end of the article

certain cases which have not been exposed in [12] by including the number of prime factors of the constants. Moreover, the theoretical lower bound for the number of arithmetic operations subject to the minimum critical path is revealed. With this, the preliminary estimation of how many extra arithmetic operations are needed to preserve the lowest critical path in a SCM block becomes possible.

This paper is organized as follows: Section 2 presents the key observations from [5] used in [12] as a starting point to develop the current theoretical lower bounds. Section 3 develops an analysis of these SCM lower bounds. From that analysis, in Section 4, we present a detailed development of new theorems that allow the extension of these lower bounds. Section 5 presents the comparison between the proposed lower bounds and the lower bounds from [12]. Finally, concluding remarks and future works are highlighted in Section 6.

2 Key observations for the current lower bounds

The main operation in SCM blocks, called *A-operation* in [6], is defined as

$$w = A_x(u, v) = |2^{e_1}u + (-1)^{e_4}2^{e_2}v|2^{-e_3} \quad (1)$$

where $e_1 \geq 0$, $e_2 \geq 0$ are integers denoting left shifts, $e_3 \geq 0$ is an integer indicating right shift, $e_4 \in \{0, 1\}$ chooses the addition or subtraction operation to be performed, $x = \{e_1, e_2, e_3, e_4\}$ is the parameter set or *A-configuration* of $A_x(u,v)$ and u, v, and w are positive and odd constants. As additions and subtractions have a similar complexity when it comes to hardware implementation, they are usually referred without distinction as *A*-operations. An SCM block is designed as a network of *A*-operations represented using directed acyclic graphs (DAGs) with the following characteristics [5,12,13]:

- Shifts are assumed to be free. Additionally, the sign of the constants formed in the DAG is assumed to be adjusted at some part of the design. Therefore, only positive and odd integers are considered. These constants are known as *fundamentals*.
- For a graph with n *A*-operations, there are $n + 1$ vertices and n fundamentals. Every fundamental is obtained as a result of an *A*-operation.
- Each vertex has an in-degree two except for the input vertex which has in-degree zero.
- A vertex with in-degree two corresponds to an *A*-operation.
- Each vertex has out-degree larger than or equal to one except for the output vertex which has out-degree zero.
- The constant resulting from the last *A*-operation is known as output fundamental (OF), whereas the constants resulting from previous *A*-operations are non-output fundamentals (NOFs). In a DAG that does not have the minimum number of *A*-operations, the constants resulting from the extra *A*-operations are referred as non-essential fundamentals (NEFs).

The number of *A*-operations, N_A, is frequently called *adder cost*. However, this value will be referred here as *A-cost* for consistency. The number of cascaded *A*-operations, N_d, where the output of an *A*-operation is at least one input of another *A*-operation, is frequently called *logic depth* or *adder depth* and it will be referred here as *A-depth*.

The following observations from [5] relate the number of nonzero digits of a constant with its *A*-depth:

- **Observation A:** *The sum of two coefficients with k_1 and k_2 nonzero digits respectively, has at most $k_1 + k_2$ nonzero digits.*
- **Observation B:** *A multiplier graph with A-depth equal to N_d can generate coefficients with at most 2^{Nd} nonzero digits.*

On the other hand, a *multiplicative graph* is the graph obtained by cascading two subgraphs such that the resulting OF is a product of the OFs of the two subgraphs (classification of graph structures is detailed in [5] and extended in [13]). An articulation point is the point where the output of the first subgraph is joined with the input of the second subgraph. Since in DAGs the *A*-operations become nodes, the *A*-operation whose output is an articulation point can be also referred as an articulation point.

- **Observation C:** *If a graph has an articulation point, the graph is multiplicative.*

Proofs and more details of these observations can be found in [5].

3 Analysis of the current lower bounds

Consider a given constant c whose MNSD, i.e., the number of digits obtained by representing c in canonic signed digit (CSD), is $S(c)$. The lower bounds L_A and L_d for the *A*-cost and the *A*-depth of the graph for c, respectively, are given in [12] as

$$L_A = L_d = \lceil \log_2 S(c) \rceil \quad (2)$$

where $\lceil x \rceil$ is the nearest integer greater than or equal to x. Taking into account that the MNSD can be expressed as $2^{p-1} < S(c) \leq 2^p$ for all $p \geq 1$ and p integer, we can note that all values $S(c)$ in the range $(2^{p-1}, 2^p]$ yield the same L_A and L_d according to (2), with

$$p = \lceil \log_2 S(c) \rceil \quad (3)$$

We will refer to the range $(2^{p-1}, 2^p]$ as *MNSD-range*. Note that the function $S(c)$ is the unique information taken from the constants and used in [12] to derive the current lower bounds for constant multiplication problems.

An important characteristic given in [12] and related to the graph structure presented in Figure 1, which will be called here the *completely multiplicative* (CM) graph, is that if $S(c) = 2^p$ (the highest MNSD in the MNSD-range), the lower bound $L_A = p$ will be obtained only with a CM graph.

Is always possible to find a solution with the lower bound L_d because the coefficient c can be implemented by summing all its non-zero terms using A-operations arranged in binary tree [12]. Nevertheless, the lower bounds given in (2) do not make any distinction among the constants that have their MNSD included into the MNSD-range.

4 Extension of the current lower bounds

Let us start with the *CM-based graph* of Figure 2, which consists of the cascade of a subgraph H with $p - l$ A-operations and a CM subgraph with l A-operations. This graph is exploited in [14] to obtain optimal multiplications by rational constants with periodic binary representations, and it has the following characteristics:

1) At the output of every A-operation of the CM subgraph, the MNSD of the resulting constant is at most twice the MNSD of the constant at its inputs, according to Observation A. If the MNSD at the output of the subgraph H, denoted by n_H, is the highest possible in H, the maximum resulting MNSD at the overall graph is $2^l \times n_H$.
2) If H has the minimum A-depth, the overall graph has the minimum A-depth because the CM subgraph has also the minimum A-depth. The same holds for the A-cost.
3) If H is non-multiplicative, the CM-based graph has l articulation points formed with the minimum A-cost and A-depth and clustered in the higher depth levels. Thus, this graph has the highest MNSD in its first articulation point in comparison to any other graph with l articulation points. Moreover, it can have the maximum MNSD with respect to other graphs with l articulation points.

Due to their aforementioned characteristics, CM-based graphs will be analyzed in the following to show that the graph of a constant whose MNSD is given in specific intervals into the MNSD-range requires certain multiplicative characteristics to preserve the lower bound L_A. It shall be considered henceforth that the MNSD is given in the MNSD-range $(2^{p-1}, 2^p]$ with $p > 1$, since $L_A = L_d = 1$ for $S(c) = 2$ (i.e., when $p = 1$).

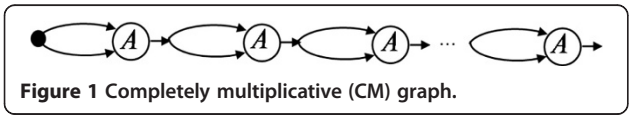

Figure 1 Completely multiplicative (CM) graph.

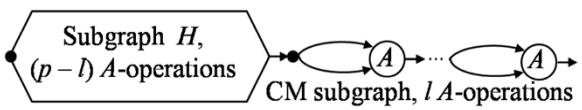

Figure 2 CM-based graph with p A-operations.

Theorem 1 *A constant c whose MNSD is* $S(c) > 2^{p-1} + 1$ *cannot be obtained with a non-multiplicative graph with p A-operations.*

Proof Let us review the simplest case $p = 2$. There are two possible graphs, one of them additive and the other one multiplicative, as shown in Figure 3 (see also Cost-2 Graphs No. 1 and 2 of Appendix A of [13]). The highest MNSD of any NOF in both graphs is equal to 2. The last A-operation of the non-multiplicative graph only can add one nonzero digit to a constant, thus yielding OFs whose highest MNSD is $2^{p-1} + 1 = 3$ according to Observation A. The highest MNSD of any OF in the multiplicative graph is equal to 4 because both inputs to the last A-operation can have up to 2 nonzero digits each. Thus, the case $p = 2$ is a base for the following inductive hypothesis: the highest MNSD of any OF in a non-multiplicative graph with p A-operations is equal to $2^{p-1} + 1$ for all $p > 1$. Assuming it as a true statement, let us construct a non-multiplicative graph with $p + 1$ A-operations, with the aim of obtaining an OF whose MNSD is $n = 2^p + 2$, i.e., the lowest MNSD that contradicts the hypothesis.

A non-multiplicative graph with p operations cannot have the inputs of its last A-operation (the A-operation placed at the p-th depth level) coming from the same position because an articulation point is generated and the graph becomes multiplicative (see Observation C). For the sake of clarity, let us identify the two inputs of the last A-operation as a_1 and a_2. Let us assume, without loss of generality, that a_1 comes from the output of a subgraph G_1, as shown in Figure 4. This subgraph can be either non-multiplicative or multiplicative (Figure 4a,b, respectively). In the following, we will review each of these two cases, along with the point where a_2 can come from.

If G_1 is non-multiplicative (Figure 4a), the highest MNSD of any of its OFs is $n_1 = 2^{p-1} + 1$ (this follows from the inductive hypothesis). For the overall graph, the only way to obtain an OF whose MNSD is $n = 2^p + 2$ is having a_2 coming from an A-operation that produces constants with an MNSD at least equal to $n - n_1 = 2^{p-1} + 1$. From Observation B, we have that this MNSD can be

Figure 3 The two graphs with $p = 2$ *A*-operations: (a) additive graph and (b) multiplicative graph.

obtained only from the p-th A-operation, which means that a_1 and a_2 would be coming from the same A-operation, generating an articulation point. If a_2 comes from the $(p-1)$-th A-operation, the articulation point is avoided and the whole graph can still be non-multiplicative. However, the highest MNSD of a constant generated from the $(p-1)$-th A-operation is equal to 2^{p-1} (see Observation B) and thus the highest MNSD of the OF in the overall graph is equal to $n_1 + 2^{p-1} = 2^{p-1} + 1 + 2^{p-1} = 2^p + 1$.

If G_1 is multiplicative (Figure 4b), it can have l articulation points where $1 \le l \le p-2$. With l articulation points and p operations, the highest MNSD of any OF produced by G_1 is $n_1 = 2^l \times (2^{p-l-1} + 1)$. This follows from the assumption that G_1 is a CM-based graph and from the inductive hypothesis. In order to obtain an OF whose MNSD is $n = 2^p + 2$, a_2 must come from an A-operation that produces constants with an MNSD at least equal to $n - n_1 = 2^{p-1} - 2^l + 2$. Moreover, such A-operation must be placed at a depth level less than the depth level where the first articulation point is placed; otherwise this articulation point will make the whole graph multiplicative. Since G_1 has its first articulation point at a depth level equal to $p-l$, a_2 must come from an A-operation placed in a depth level that does not exceed $p-l-1$. The highest MNSD of any NOF obtained from an A-operation placed in the $(p-l-1)$-th depth level is 2^{p-l-1} (from Observation B). Hence, this value should be equal or greater than $n - n_1 = 2^{p-1} - 2^l + 2$ in order to obtain an OF

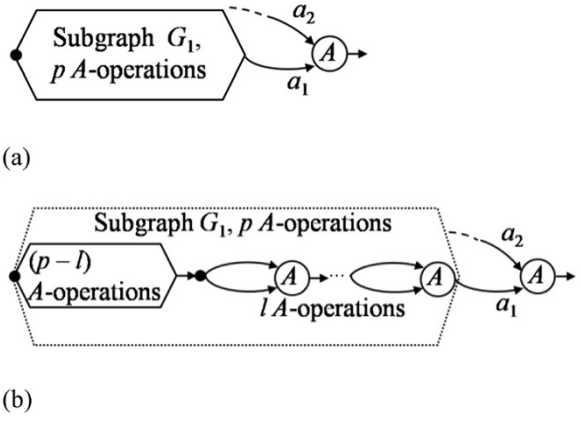

Figure 4 A graph composed by a subgraph G_1 plus a last *A*-operation. (a) G_1 is non-multiplicative graph and **(b)** G_1 is a CM-based multiplicative graph.

in the overall graph whose MNSD is $n = 2^p + 2$. However, it can be shown that $2^{p-l-1} < 2^{p-1} - 2^l + 2$ holds. Thus, the highest MNSD of any OF in the overall graph is equal to $n_1 + 2^{p-l-1} = 2^l \times (2^{p-l-1} + 1) + 2^{p-l-1} \le 2^p + 1$.

As a result, we have that the inductive hypothesis for a graph with p A-operations implies that it holds for $p+1$ A-operations. Therefore, considering this implication along with the base case with $p = 2$, we prove Theorem 1 by induction on p. ■

Theorem 2 *A constant c whose MNSD is $S(c) > 2^{p-1} + 2^{q-1}$ with $q = \{1, 2, \ldots, (p-1)\}$ can be obtained by using a graph with p A-operations only if at least q of these operations are articulation points.*

Proof Consider that the overall graph is a CM-based graph with l articulation points, where $l < p$. The highest MNSD of any OF in the overall graph is $n = 2^l \times (2^{p-l-1} + 1) = 2^{p-1} + 2^l$ (from Theorem 1). Therefore, l has to be at least equal to q in order to achieve $n > 2^{p-1} + 2^{q-1}$. Note that Theorem 1 corresponds to the case $q = 1$. ■

Theorem 3 *A constant c whose MNSD is $S(c) > 3 \times 2^{p-2} + 1$ cannot be obtained by using a non-multiplicative graph with only $p+1$ A-operations and A-depth preserved equal to p.*

Proof For the simplest case $p = 2$, there is only one graph with $p + 1 = 3$ A-operations and A-depth preserved equal to $p = 2$. This graph is shown in Figure 5 (see also Cost-3 Graph No. 7 of Appendix A of [13]). The highest MNSD of the OF in this graph is $3 \times 2^{p-2} + 1 = 4$.

Now, we review the case $p \ge 3$. Since the A-depth must be preserved equal to p, only p of these operations can be sequentially connected. Consider that a graph G_1 with A-cost and A-depth equal to $p-1$ is connected to one of the inputs of the last A-operation, as shown in Figure 6. In order to avoid an articulation point, the other input must come from a different position. Thus, this input is connected to the remaining A-operation, which should be placed in the $(p-1)$-th depth level to obtain the highest possible MNSD (from Observation B), and whose inputs are identified as a_1 and a_2. Consider that a_1 comes from the A-operation placed in the $(p-2)$-th depth level. Let us review the cases when G_1 is either non-multiplicative (Figure 6a) or multiplicative (Figure 6b), along with the point where a_2 can come from.

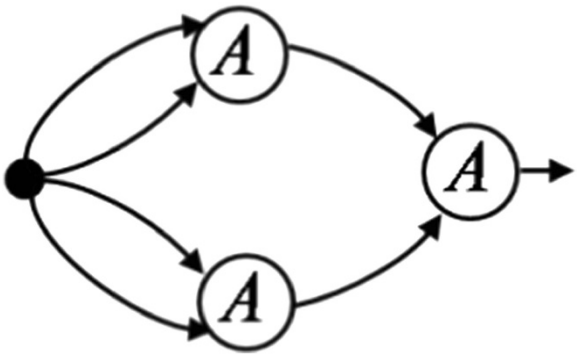

Figure 5 The unique graph with *A*-depth equal to $p = 2$ and $p + 1 = 3$ *A*-operations.

If G_1 is non-multiplicative (Figure 6a), the highest MNSD of its OFs is $n_1 = 2^{p-2} + 1$ (from Theorem 1) and a_1 with a_2 can come from the A-operation placed in the $(p-2)$-th level. Thus, the highest possible MNSD of the OF in the last A-operation is $n = n_1 + n_{a1} + n_{a2}$, where n_{a1} and n_{a2} are the respective highest MNSD in a_1 and a_2. The highest MNSD of G_1 is $n_1 = 2^{p-2} + 1$ and $n_{a1} = n_{a2} = 2^{p-2}$. Therefore, we have $n = (2^{p-2} + 1) + 2^{p-2} + 2^{p-2} = 2^{p-2} + 1 + 2^{p-1} = 3 \times 2^{p-2} + 1$.

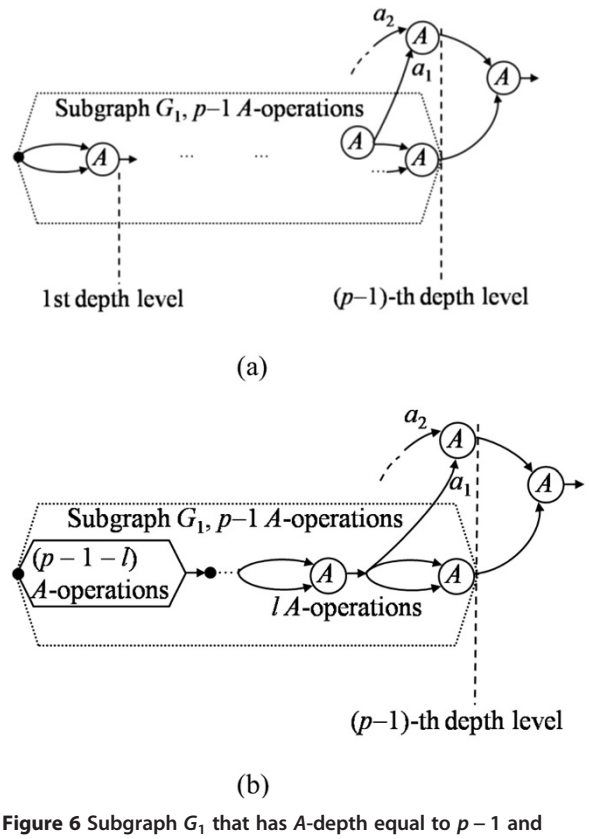

Figure 6 Subgraph G_1 that has *A*-depth equal to $p - 1$ and $p - 1$ *A*-operations, plus two extra *A*-operations. (a) G_1 is non-multiplicative graph and **(b)** G_1 is a CM-based multiplicative graph.

If G_1 is multiplicative (Figure 6b), consider that it is a CM-based graph with l articulation points, where $1 \le l \le p-2$. The highest MNSD of its OFs is $n_1 = 2^l \times (2^{p-l-2} + 1)$. Similarly, the highest MNSD in a_1 is $n_{a1} = 2^{l-1} \times (2^{p-l-2} + 1)$. To avoid an articulation point, a_2 must come from a depth level that does not exceed $p - l - 2$, thus $n_{a2} = 2^{p-l-2}$. The highest possible MNSD of the OFs in the last A-operation is $n = n_1 + n_{a1} + n_{a2} = 2^{p-2} + 2^{p-3} + 2^l + 2^{l-1} + 2^{p-l-2}$, which can be shown to be less or equal to $3 \times 2^{p-2} + 1$. ■

Theorem 4 *A constant c whose MNSD is $S(c) > 3 \times 2^{p-2} + 2^{q-1}$ with $q = \{1, 2, \ldots, (p-2)\}$ can be obtained by using a non-multiplicative graph whose A-depth is equal to p only if at least $p + q + 1$ A-operations are used.*

Proof Consider $p + r$ A-operations, with $r < p$. Since the A-depth is equal to p, only p of these operations can be sequentially connected. Let us assume that a graph G_1 with A-cost and A-depth equal to $p - 1$ is connected to one of the inputs of the last A-operation, whereas the other input comes from a different position to avoid an articulation point. Consider that this input comes from a graph G_2, formed with $r - 1$ of the remaining A-operations. To obtain the highest MNSD in G_2, we can consider that G_2 is a CM graph with its last A-operation placed at the $(p-1)$-th depth level. Let us assume that the last of the remaining A-operations is connected to the input of G_2, with its inputs being a_1 and a_2. This is shown in Figure 7. The highest MNSD of any OF from G_2 is $n_2 = 2^{r-1} \times (n_{a1} + n_{a2})$, where n_{a1} and n_{a2} are the respective highest MNSD in a_1 and a_2.

If G_1 is non-multiplicative (Figure 7a), the highest MNSD of its OFs is $n_1 = 2^{p-2} + 1$ (from Theorem 1). Both inputs a_1 and a_2 can come from the same point because G_1 is non-multiplicative, and the highest depth level where this point can be placed is $p - r - 1$; thus, $n_{a1} = n_{a2} = 2^{p-r-1}$. Hence, the highest MNSD of any OF in the overall graph is $n = n_1 + n_2 = (2^{p-2} + 1) + 2^{r-1} \times (n_{a1} + n_{a2}) = (2^{p-2} + 1) + 2^{r-1} \times (2^{p-r}) = (2^{p-2} + 1) + 2^{p-1} = 3 \times 2^{p-2} + 1$, which clearly is less or equal than $3 \times 2^{p-2} + 2^{q-1}$ even for the smallest $q = 1$.

If G_1 is multiplicative, consider that it is a CM-based graph with l articulation points, where $1 \le l \le p-2$. The highest MNSD of its OFs is $n_1 = 2^l \times (2^{p-l-2} + 1)$. We can have two cases, depending on whether r is greater than l or not. Let us see each one of these cases.

When $r \le l$ (Figure 7b), only either a_1 or a_2 can come from the A-operation placed in the $(p - l - 1)$-th depth level but not both, because an articulation point would be generated. If a_1 comes from this A-operation, a_2 can come from the immediate lower depth level. Thus, $n_{a1} = (2^{p-l-2} + 1)$ and $n_{a2} = 2^{p-l-2}$. Hence, the highest MNSD of any OF in the overall graph is $n = n_1 + n_2 =$

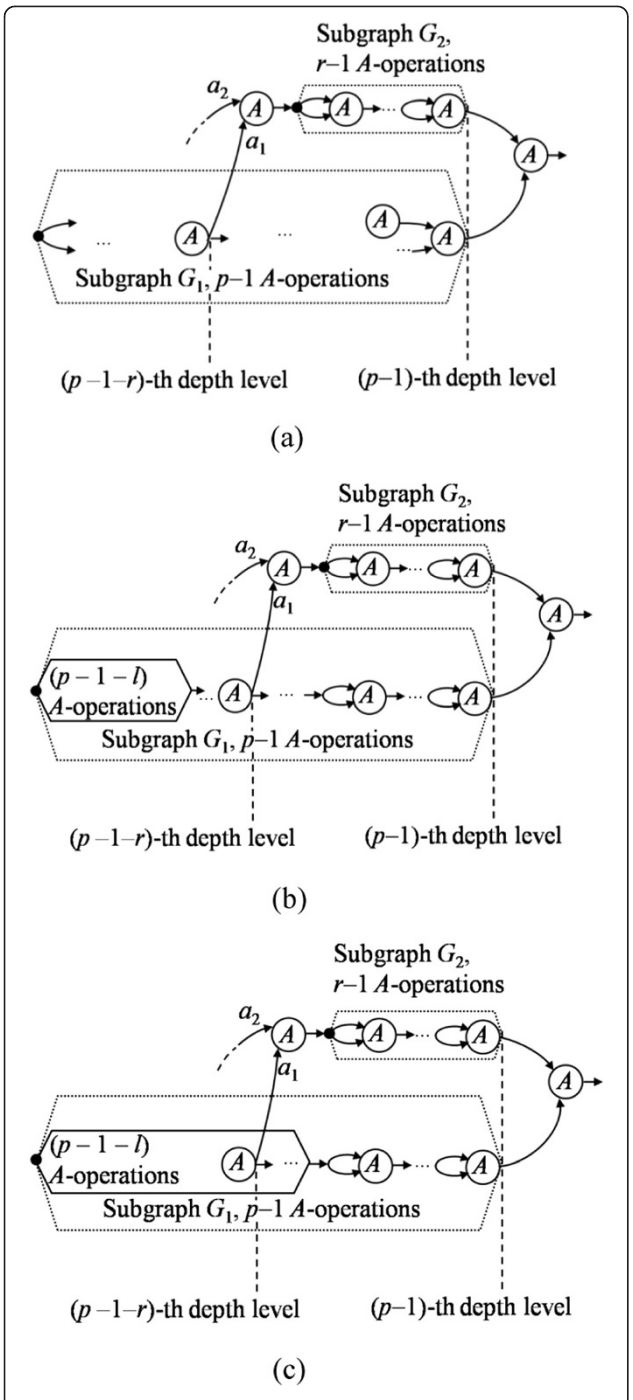

Figure 7 Subgraph G_1, CM subgraph G_2, and two extra A-operations. (a) G_1 is non-multiplicative graph. **(b)** G_1 is a CM-based multiplicative graph, with $r \leq l$ and **(c)** G_1 is a CM-based multiplicative graph, with $r > l$.

$(2^{p-2} + 2^l) + 2^{r-1} \times (n_{a1} + n_{a2}) = (2^{p-2} + 2^l) + 2^{r-1} \times [(2^{p-l-2} + 1) + 2^{p-l-2}] = 2^{p-2} + 2^l + 2^{r-1} \times (2^{p-l-1} + 1)$, which can be shown to be less or equal to $3 \times 2^{p-2} + 2^{r-1}$. Thus, r must be at least equal to $q + 1$ in order to achieve a value $n > 3 \times 2^{p-2} + 2^{q-1}$.

When $r > l$ (Figure 7c), a_1 and a_2 can come from the same point because the subgraph H in G_1 is non-multiplicative, thus $n_{a1} = n_{a2} = 2^{p-r-1}$. Hence, the highest MNSD of any OF in the overall graph is $n = n_1 + n_2 = (2^{p-2} + 2^l) + 2^{r-1} \times (n_{a1} + n_{a2}) = (2^{p-2} + 2^l) + 2^{r-1} \times (2^{p-r}) = 2^{p-2} + 2^l + 2^{p-1} = 3 \times 2^{p-2} + 2^l$. Since l is at most equal to $r - 1$, we have that the highest value for n is $n = 3 \times 2^{p-2} + 2^{r-1}$. Thus, $n > 3 \times 2^{p-2} + 2^{q-1}$ holds only if r is at least equal to $q + 1$. ■

Now, let us introduce the number of prime factors of the constant c, denoted as $\Omega(c)$. Note that, along with $S(c)$, $\Omega(c)$ is a function that can be known *a priori* from the constants. Generally speaking, obtaining the prime factors of c is a challenging problem for very large constants (over 110 digits) and it is one of the bases for the Rivest-Shamir-Adleman (RSA) encryption scheme [19]. However, for constants up to 32 bits, which cover the DSP problem sizes of the most practical importance [14], the prime factors of c can be obtained straightforwardly even with the simple sieve approach, such as the one used in the MATLAB function *'factor'*.

The following theorems present the relations between the values $\Omega(c)$ and $S(c)$. If the required multiplicative characteristics for the corresponding graph of the constant c, highlighted in the previous theorems, cannot be accomplished due to the value $\Omega(c)$, the lower bounds for the A-cost and the A-depth are affected.

Theorem 5 *A constant c whose MNSD is* $S(c) > 2^{p-1} + 2^{\Omega(c)-1}$ *only can be obtained by using a graph with at least* $p + 1$ *A-operations.*

Proof From Theorem 2, it is known that at least $\Omega(c)$ articulation points are required if only p A-operations can be used. However, since $\Omega(c)$ is the number of prime factors, only up to $\Omega(c) - 1$ articulation points are allowed. Thus, the only solution is using an additional A-operation. By adding one more A-operation, the overall non-multiplicative graph can give an OF whose MNSD is up to $2^p + 1$, which covers the MNSD-range regardless of the value $\Omega(c)$. Therefore, at least $p + 1$ A-operations are required. ■

Theorem 6 *A constant c whose MNSD is* $S(c) > 3 \times 2^{p-2} + 2^{\Omega(c)+q-2}$ *with* $q = \{1, 2, \ldots, (p - \Omega(c) - 1)\}$ *only can be obtained by using a graph with A-depth at least equal to* $p + 1$ *if up to* $p + q$ *A-operations are used, or with at least* $p + q + 1$ *A-operations if the minimum A-depth equal to p is preserved.*

Proof Let us consider a CM-based graph with $p + r$ A-operations and A-depth equal to $p + s$, where $r < p$ and $s \geq 0$. Since at most $\Omega(c) - 1$ articulation points can be used, its CM subgraph has $\Omega(c) - 1$ A-operations, whereas

its non-multiplicative subgraph H has $p + r - \Omega(c) + 1$ A-operations and an A-depth equal to $p + s - \Omega(c) + 1$. According to Theorem 4, the highest MNSD of any OF generated in the subgraph H is $3 \times 2^{p+s-\Omega(c)-1} + 2^{r-s-1}$ and the highest MNSD at the output of the overall graph is $n = 2^{\Omega(c)-1} \times (3 \times 2^{p+s-\Omega(c)-1} + 2^{r-s-1}) = 3 \times 2^{p+s-2} + 2^{\Omega(c)+r-s-2}$. If the A-depth is preserved equal to p, we have $s = 0$. In such case r needs to be at least equal to $q + 1$ in order to achieve $n > 3 \times 2^{p-2} + 2^{\Omega(c)+q-2}$. On the other hand, if $r \le q$ holds, the value s should be at least equal to 1 in order to achieve $n > 3 \times 2^{p-2} + 2^{\Omega(c)+q-2}$, implying that the A-depth of the overall graph is at least equal to $p + 1$. ■

From Theorem 5 and substituting p using (3), the lower bound L_A can be expressed as

$$L_A = \begin{cases} \lceil \log_2 S(c) \rceil; \text{ if } \Omega(c) \ge \log_2\left\{S(c) - 2^{\lceil \log_2 S(c) \rceil - 1}\right\} + 1, \\ \lceil \log_2 S(c) \rceil + 1; \text{ otherwise.} \end{cases} \tag{4}$$

Note that if $L_A = \lceil \log_2 S(c) \rceil$ holds, L_d will have the same value as L_A without requiring any additional A-operation (as given by (2)). From (4), we can see that this only can be accomplished if $\Omega(c) \ge \log_2\{S(c) - 2^{\lceil \log_2 S(c) \rceil - 1}\} + 1$ holds. The argument $S(c) - 2^{\lceil \log_2 S(c) \rceil - 1}$ in the $\log_2\{x\}$ operation is always greater than zero and therefore the condition can always be evaluated.

From Theorem 6, we have that if the A-cost is given as $N_A \ge \lceil \log_2 S(c) \rceil + q + 1$ and if $\log_2\{S(c) - 3 \times 2^{\lceil \log_2 S(c) \rceil - 2}\} + 2 - \Omega(c) > q$ holds, the A-depth can be kept equal to $\lceil \log_2 S(c) \rceil$. By noticing that any real number a is greater than an integer b if this integer is given as $b = \lceil a \rceil - 1$ and applying this observation to q, we obtain $q = \lceil \log_2 \left\{S(c) - 3 \times 2^{\lceil \log_2 S(c) \rceil - 2}\right\} + 2 - \Omega(c) \rceil + 1$. In this equality, the $\log_2\{x\}$ operation does not yield finite values when $S(c) \le 3 \times 2^{\lceil \log_2 S(c) \rceil - 2}$ holds. However, according to Theorem 3, the A-depth can be kept equal to $\lceil \log_2 S(c) \rceil$ by using only one additional A-operation if the condition $S(c) \le 3 \times 2^{\lceil \log_2 S(c) \rceil - 2} + 1$ holds. Therefore,

$$L_d = \begin{cases} \lceil \log_2 S(c) \rceil; & \text{if } N_A \ge \lceil \log_2 S(c) \rceil + L_{EA}, \\ \lceil \log_2 S(c) \rceil + 1; & \text{otherwise,} \end{cases} \tag{5}$$

where L_{EA}, the lower bound for the number of extra A-operations required to generate NEFs and preserve $L_d = \lceil \log_2 S(c) \rceil$, is given as

$$L_{EA} = \begin{cases} 0; & \text{if } \Omega(c) \ge \log_2\left\{S(c) - 2^{\lceil \log_2 S(c) \rceil - 1}\right\} + 1, \\ 1; & \text{if } S(c) \le 3 \times 2^{\lceil \log_2 S(c) \rceil - 2} + 1, \\ \lceil \log_2\left\{S(c) - 3 \times 2^{\lceil \log_2 S(c) \rceil - 2}\right\} + 2 - \Omega(c) \rceil; & \text{otherwise.} \end{cases} \tag{6}$$

It is worth highlighting that, whereas the lower bound L_A only depends on the relation between $\Omega(c)$ and $S(c)$, the lower bound L_d depends on the A-cost N_A. As pointed out in [12], for many problem instances, there is a tradeoff between the A-cost and the A-depth. From (5), we have that such tradeoff is the minimum theoretical value L_{EA}, given in (6).

5 Comparison of proposed and current lower bounds

Let us review a simple example to illustrate how the proposed lower bounds given in (4) and (5)-(6) are more precise with respect to the lower bounds from [12], expressed in (2). To this end, consider the 14-bit integer constant 11,467, whose binary representation is 10110011001011. Its CSD representation is $10\bar{1}0\bar{1}010\bar{1}010\bar{1}0\bar{1}$, which can be found from the binary representation by iteratively replacing every string having $n \ge 2$ consecutive digits '1' (e.g., 1111, where $n = 4$) with the string having $n - 1$ digits '0' between a digit '1' and a digit '$\bar{1}$', (e.g., $1000\bar{1}$, where $n = 4$; see [20] for more details). A MATLAB routine to find the CSD representation of an integer is given in [21].

Recall that the value $S(c)$ represents the number of non-zero digits of a minimum signed digit (MSD) representation (such as CSD) of the constant c. Hence, it is clear from the CSD representation of the constant c = 11,467 that $S(c) = 8$, i.e., there are eight non-zero digits in that representation. Note in passing that, in this particular example, the number of non-zero digits is the same for both, CSD and binary representations. However, this not always occurs. Additionally, we can obtain $\Omega(c)$, the number of prime factors of the constant c, using the MATLAB command *'length(factor(c))'*. For the constant c = 11,467, we have $\Omega(c) = 1$, which indicates that 11,467 is a prime number.

The constant 11,467 is known to have an optimal A-cost that does not exceed 4, as mentioned in [20]. A graph with four A-operations that implements the multiplier by 11,467, based on the 'Leapfrog' Cost-4 Graph No. 10 of Appendix A of [13], is shown in Figure 8.

Let us denote the current lower bounds from [12] as $L_{A,c}$ for the A-cost and $L_{d,c}$ for the A-depth, which can be obtained using (2). Similarly, we will denote the new lower bounds as $L_{A,\text{new}}$ for the A-cost and $L_{d,\text{new}}$ for the A-depth, which can be obtained using (4) and (5)-(6), respectively. All these lower bounds will be calculated by substituting c = 11,467.

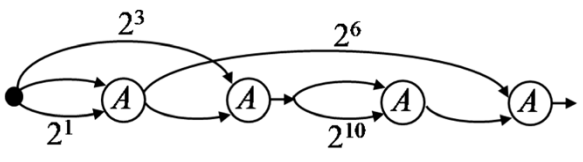

Figure 8 Graph with four *A*-operations to implement the multiplier by 11,467.

The value $S(c) = 8$ replaced in (2) yields $L_{A,c} = \lceil \log_2 S(c) \rceil = \lceil \log_2(8) \rceil = 3$. However, besides of $S(c) = 8$, we also have $\Omega(c) = 1$ as additional information for the new lower bounds. Using $S(c)$ and $\Omega(c)$ in (4), we have that the condition $\Omega(c) \geq \log_2\left\{S(c) - 2^{\lceil \log_2 S(c) \rceil - 1}\right\} + 1$ does not hold (in other words, $1 \geq \log_2\left(8 - 2^{\lceil \log_2(8) \rceil - 1}\right) + 1$ is a false statement). Therefore, according to (4), the new lower bound for the A-cost of this constant is $L_{A,\text{new}} = \lceil \log_2 S(c) \rceil + 1 = \lceil \log_2(8) \rceil + 1 = 4$. This new lower bound reveals that the solution with A-cost equal to 4, whose graph is presented in Figure 8, is in fact optimal in terms of the A-cost.

On the other hand, recall from [12] that it is always possible to find a solution with the lower bound $L_{d,c}$ given in (2), which in this case is $L_{d,c} = \lceil \log_2 S(c) \rceil = \lceil \log_2(8) \rceil = 3$. Figure 9 shows a graph that implements the multiplier by 11,467 with the minimum A-depth equal to 3 and five A-operations, based on the 'Leapfrog' Cost-5 Graph No. 14 of Appendix A of [13].

Since we have $L_{A,c} = L_{d,c} = 3$, with the current lower bounds one can assume that the three A-operations necessary to meet the minimum A-depth might be sufficient to implement the constant multiplier by 11,467. However, from (5) we have that the lower bound $L_{d,\text{new}} = \lceil \log_2 S(c) \rceil = \lceil \log_2(8) \rceil = 3$ can be preserved if at least L_{EA} extra A-operations are used in addition to the three A-operations dictated by the bound $L_{d,new} = 3$. The value for L_{EA} is obtained using (6), where the conditions $\Omega(c) \geq \log_2\left\{S(c) - 2^{\lceil \log_2 S(c) \rceil - 1}\right\} + 1$ and $S(c) \leq 3 \times 2^{\lceil \log_2 S(c) \rceil - 2} + 1$ do not hold (in other words, $1 \geq \log_2\left\{8 - 2^{\lceil \log_2(8) \rceil - 1}\right\} + 1$ and $8 \leq 3 \times 2^{\lceil \log_2(8) \rceil - 2} + 1$ are false statements), leading to $L_{EA} = \lceil \log_2\left\{S(c) - 3 \times 2^{\lceil \log_2 S(c) \rceil - 2}\right\} + 2 - \Omega(c) \rceil = \lceil \log_2\left\{8 - 3 \times 2^{\lceil \log_2(8) \rceil - 2}\right\} + 2 - 1 \rceil = 2$. This means that three A-operations are actually not sufficient to preserve the minimum A-depth and at least five A-operations are required. Thus, the solution with A-cost equal to 5 and A-depth equal to 3, whose graph is presented in Figure 9, is in fact optimal in terms of the A-cost subject to the minimum A-depth.

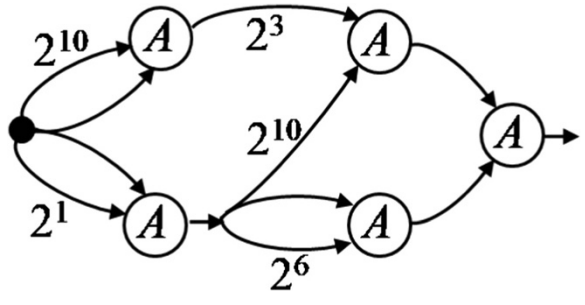

Figure 9 Graph to implement the multiplier by 11,467 with five *A*-operations and the minimum *A*-depth equal to 3.

It is clear that the proposed lower bounds are more informative in comparison to the lower bounds from [12]. Similar examples arise with the integer constants 11,093 and 13,003, among others. Finally, Table 1 shows the percentage of constants with improved lower bounds among all the 14-bit odd integer constants and among 10,000 randomly generated constants that can be expressed with B-bits, where $14 < B \leq 32$.

6 Conclusions

This paper has presented an extension of the current theoretical lower bounds for the A-cost and the A-depth in single constant multiplication (SCM) blocks constructed with shifts and A-operations (additions and subtractions). The insight of this work has been the introduction of the number of prime factors of the constant as key information to such extension.

Additionally, a new lower bound has been revealed, namely, the theoretical minimum number of A-operations required to preserve the minimum A-depth, denoted as L_{EA}. This new lower bound is essential because preserving the minimum A-depth has been currently an important objective to design systems with high speed and low power. By knowing L_{EA} it is possible to have an early estimation of how many extra adders would be needed for a given implementation with the lowest A-depth, which is translated to extra chip area. Since for current and future semiconductor technologies leakage power consumption is closely related to chip area and it has a significant impact, the value L_{EA} might provide valuable information to decide in an early stage of design if it is better to pursue a minimization of the A-cost subject to the minimum A-depth or minimize the A-cost subject to a given A-depth greater than the minimum.

As a future work, the current theoretical lower bounds for multiple constant multiplication (MCM) cases can be also extended. The proposed theorems and the extended lower bounds can be used as a basis, and the inclusion of the number of prime factors of the involved constants might, of course, play an important role as it did in this work. The theoretical lower bound for the A-cost required when the minimum A-depth is preserved, as well as the lower bound for the A-depth necessary to obtain the minimum A-cost, can be revealed. Additionally, the formulation of optimal and a suboptimal SCM and MCM algorithms to minimize the A-cost subject to the minimum A-depth can be developed, taking as a

Table 1 Percentage of constants with improved lower bounds

Word length	Total	$L_{EA} = 1$	$L_{EA} = 2$	$L_{EA} = 3$
($B = 14$)-bits	19.6%	19.4%	0.2%	0%
($14 < B \leq 32$)-bits	33.36%	32.08%	1.14%	0.14%

basis the theorems developed in this work, where the prime factors of the involved constants can be used as input information.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
This work has been supported by CONACYT Project No. 179587.

Author details
[1]Department of Electronics, National Institute of Astrophysics, Optics and Electronics, Tonantzintla, Puebla 72840, Mexico. [2]Electrical and Computer Engineering Department, Florida State University, Tallahassee, FL 32310, USA.

Received: 3 May 2014 Accepted: 25 July 2014
Published: 5 August 2014

References
1. AG Dempster, MD Macleod, Constant integer multiplication using minimum adders. IEEE Proc. Circuits Devices Syst. **141**(5), 407–413 (1994)
2. AG Dempster, MD Macleod, Use of minimum-adder multiplier blocks in FIR digital filters. IEEE Trans. Circ. Syst. II **42**(9), 569–577 (1995)
3. AG Dempster, SS Dimirsoy, I Kale, *Designing multiplier blocks with low logic depth, in Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS 2002)*, vol. 5 (IEEE, Piscataway, 2002), pp. 773–776
4. O Gustafsson, AG Dempster, L Wanhammar, Extended results for minimum-adder constant integer multipliers, in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS 2002) vol. 1* (IEEE, Piscataway, 2002), pp. 73–76
5. O Gustafsson, AG Dempster, K Johansson, MD Macleod, L Wanhammar, Simplified design of constant coefficient multipliers. Circ. Syst. Signal Process **25**(2), 225–251 (2006)
6. Y Voronenko, M Püschel, Multiplierless multiple constant multiplication. ACM Trans. Algorithms **3**(2), 1–38 (2007)
7. M Faust, C Chip-Hong, Minimal logic depth adder tree optimization for multiple constant multiplication, in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS 2010)* (Paris, 2010), pp. 457–460
8. K Johansson, O Gustafsson, LS DeBrunner, L Wanhammar, Minimum adder depth multiple constant multiplication algorithm for low power FIR filters, in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS 2011)* (Rio de Janeiro, 2011), pp. 1439–1442
9. AG Dempster, MD Macleod, Using all signed-digit representations to design single integer multipliers using subexpression elimination, in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS 2004) vol. **3*** (IEEE, Piscataway, 2004), pp. 165–168
10. J Thong, N Nicolici, Time-efficient single constant multiplication based on overlapping digit patterns. IEEE Trans. VLSI Syst. **17**(9), 1353–1357 (2009)
11. J Thong, N Nicolici, An optimal and practical approach to single constant multiplication. IEEE Trans. Comput. Aided Des. **30**(9), 1373–1386 (2011)
12. O Gustafsson, Lower bounds for constant multiplication problems. IEEE Trans. Circ. Syst. II **54**(11), 974–978 (2007)
13. U Meyer-Baese, O Gustafsson, A Dempster, The canonical minimised adder graph representation, in *Proceedings of the SPIE 2008, Independent Component Analyses, Wavelets, Unsupervised Nano-Biomimetic Sensors, and Neural Networks VI, vol. **6979*** (SPIE, 2008), pp. 1–12
14. F De-Dinechin, Multiplication by rational constants. IEEE Trans. Circ. Syst. II **59**(2), 98–102 (2012)
15. L Aksoy, E Costa, P Flores, J Monteiro, Finding the optimal tradeoff between area and delay in multiple constant multiplications. Elsevier J. Microprocessors Microsystems **35**(8), 729–741 (2011)
16. Y Pan, PK Meher, Bit-level optimization of adder trees for multiple constant multiplications for efficient FIR filter implementation. IEEE Trans. Circ. Syst. I **61**(2), 455–462 (2014)
17. S Mirzaei, R Kastner, A Hosangadi, Layout aware optimization of high speed fixed coefficient FIR filters for FPGAs. Int. J. Reconfigurable Comput. **4**(1), 1–17 (2010)
18. K Martin, H Martin, W Jens, Z Peter, M-B Uwe, Multiple constant multiplication with ternary adders, in *Proceedings of the 23rd International Conference on Field Programmable Logic and Applications FPL 2013* (Porto, 2013), pp. 1–8
19. S Hung-Min, W Mu-En, T Wei-Chi, MJ Hinek, Dual RSA and its security analysis. IEEE Trans. Inf. Theory **53**(8), 2922–2933 (2007)
20. U Meyer-Baese, *Digital Signal Processing with Field Programmable Gate Arrays*, 3rd edn. (Springer, New York, 2007). p. 85
21. MATLAB Central, *File Exchange*. http://www.mathworks.co.jp/matlabcentral/fileexchange/9730-canonical-signeddigits/content/csdigit.m. Accessed 07 Jul 2014

doi:10.1186/1687-6180-2014-122
Cite this article as: Troncoso Romero *et al.*: On the inclusion of prime factors to calculate the theoretical lower bounds in multiplierless single constant multiplications. *EURASIP Journal on Advances in Signal Processing* 2014 **2014**:122.