Transistor Level Implementation of Cyclic Combinational Circuits

Vinay Kumar¹ and Anup Dandapat²
¹vinay.kumar@nitm.ac.in, ²anup.dandapat@gmail.com
¹,²Dept. of ECE, NITMeghalaya, Shillong, India

Abstract—Combinational Circuits are defined as the circuit whose output depends on present inputs only and are memory less. These circuits are generally acyclic (feed-forward) but cyclic circuits can be combinational where cycles sometimes occur in designs synthesized from high-level descriptions. Feedback in such cases is carefully contrived when functional units are connected in a cyclic topology. Deliberate incorporation of such cycles or feedbacks in conventional combinational circuit eventually results in less number of transistor counts leading to improved speed and power performance. So a Cyclic Combinational Circuit is defined as the circuit whose output depends on present inputs only, but at the same time contains one or more topological feedback paths. In this paper, we argue the case for radically rethinking the concept of combinational circuit design. We should no longer think of combinational logic as acyclic in theory or in practice since nearly all combinational circuits are best designed with cycles. We propose a methodology and demonstrate the same with a test case for the synthesis of some popular combinational circuits like 2-Bit Magnitude Comparator. A feedback was introduced in the substitution or minimization phase optimizing a multilevel network description for area or transistor count and specially static and dynamic power consumption. The simulations were carried out using the standard Cadence Virtuoso Suite. Further the optimized cyclic combinational circuit is implemented in ASIC platform using Cadence Layout XLSuite and design verification was done using ASSURA DRC and LVS Verification Tool. The test runs confirmed the transistor level functioning of the optimized layout. In simulations, transistor count was minimized significantly leading to 34%, 14% and 26% savings in the area, dynamic and static power respectively as compared to the conventional implementation technique.

Index Terms—Cyclic Combinational Circuit (C.C.C), Cycles, Loops, Feedback, Combinational Circuit, Acyclic.

I. INTRODUCTION

By general perception, digital circuits are classified into combinational and sequential circuits. However, combinational circuits can have cycles [1-6]. Hence, now we can more precisely define the combinational and sequential circuits. A combinational circuit has output values that depend only on the current values applied to the inputs irrespective of presence or absence of feedbacks or cycles whereas a sequential circuit has output values that depend on the entire sequence of past and current values applied to the inputs [1]. So
Cyclic Combinational Circuit (C.C.C) is a part of combinational circuit that contains feedback topology but instead of feedback its output values depends only on the current input values. Many efforts [1][3][7-9] have so far been reported about the minimal logic design capability of C.C.C. Here the objective of this paper is same. Feedback is introduced in the substitution or minimization phase in such a way that the circuit remains combinational as well as optimizing a multilevel network description for area or transistor count and specially static and dynamic power consumption. In this particular work a methodology has been introduced for defining cycle in combinational circuit to forms its equivalent C.C.C. for achieving the goal of minimal logic design [1].

II. METHODOLOGY OF CYCLIC COMBINATIONAL CIRCUITS

There are several output functions in a combinational logic design. Among these outputs we have to choose the appropriate output functions as the dependencies (feedback from the outputs and will be treated as the secondary inputs for another output functions) that provide minimal (less no of literals in the algebraic expression) logic design capability as well as retaining the circuit in its combinational nature. The cyclic combinational technique consists of three steps:

(i) First express each output node function in terms of all remaining output nodes and check for combinationality. If combinational, add the dependencies to the final output expression. If it is not go to step (ii).
(ii) Select a current branch for combinationality, if it is not combinational discard the dependency. If it is combinational set the dependency to the final output expression. In this way check all the branches for combinationality until the overall circuit becomes combinational in nature and go to step (iii).
(iii) In step (ii) there may be several combinations of dependencies but we have to choose the appropriate one which provides fewer literals in the algebraic expression.

The pictorial representation of the above three steps has been shown in Fig. 1.

![Fig. 1- Steps for finding dependencies in cyclic combinational circuits](image_url)

III. PRIOR WORK

Switching relay circuits were analyzed [10] by Claude Shannon for the first time in 1938 and the synthesis of such circuits were done in 1949[11]. The circuits of Shannon often had cyclic topologies. Since relays are directionless, cycles do not pose any problem. In 1953, Shannon again proposed a paper titled, “Realization of All 16 Switching Functions of Two Variables Requires 18 Contacts”, and he proved that this circuit is optimal than the general acyclic conventional one [12]. McCaw presented a thesis for his Engineer's Degree titled, “Loops in Directed Combinational Switching Networks” in 1963 [2]. He started with an example of cyclic circuit which is shown in Fig. 2 consisting of two AND gates and two OR gates. In 1970 and 1971, Kautz[8] and Huffman[7] respectively described that cycle or feedback is a very useful technique for minimal logic design. In 1977, Rivest presented a general version of the circuits [9], as well as the argument
for its optimality. R. Brayton et. al. proposed a paper titled, “MIS: A multiple-level logic optimization system” in 1987. L. Stok, proposed a paper titled, “False loops through resource sharing”[5] in 1992. In this paper he described the false path in a cyclic combinational circuit. The analysis of cyclic combinational circuits was first formulated by Malik [4], based on ternary-valued in 1994. Subsequent efforts [13] [14] [15] were built upon this formulation and bore much the same foundation. However, the quest for solutions to analyze combinationality remained because the analysis at the functional level was left open. In 1996, T. R. Shiplett et. al. proposed a paper titled “Constructive analysis of cyclic circuits”[14]. T. Shiplett presented his Phd thesis titled “Formal Analysis of Cyclic Circuits” in 1996 in which he analysed the feedbacks present in the cyclic circuits[16]. In 1999, K. Namjoshi et. al. presented a paper titled “Efficient Analysis Of Cyclic Definitions” in which he modified the definition of cyclic circuits[17]. Marc D. Riedel et. al. proposed a paper titled “The Synthesis of Cyclic Combinational Circuits”[18] in 2003. In this paper, they explored the topic of cyclic combinational circuit synthesis and demonstrated that these are not isolated examples. They described a general methodology for the synthesis of multilevel combinational circuits with cyclic topologies. Basically, in this paper they introduce feedback in the substitution or minimization phase for optimizing area (as measured by the literal count). In 2004, Jie-Hong et. al. proposed a paper titled “On Breakable Cyclic Definitions” where they focused on the analysis of combinationality in cyclic combinational circuit [19]. In 2006, O. Neiroukh et. al. proposed an algorithm to characterize exactly all combinational behavior of a cyclic circuit[20] and in 2008 they shifted their work to transform a cyclic circuit to its equivalent acyclic one[21]. In 2010, Biswarup Mukherjee et. al. [22] presented a paper titled “Design of Combinational Circuits by Cyclic Combinational Method for Low Power VLSI” in which a gate level design of 2-bit cyclic comparator was done using conventional as well cyclic combinational technique. Mark D. Riedel et. al. proposed a paper titled, “Cyclic Boolean Circuits” in 2012[23] in which he explored the practical implications of cyclic Boolean circuits. He also described an efficient approach for analyzing cyclic circuits and provided a general framework for synthesizing such circuits. Till now all the cyclic combinational circuits have been implemented at gate level in literature. In this paper we present a transistor level design of cyclic combinational with help of 2-Bit Comparator.

Fig.2-A cyclic combinational circuit due to McCaw

IV. CASE STUDY FOR THE PROPOSED WORK

A magnitude comparators a combinational circuit that compares two numbers, A and B and determines their relative magnitude. For a case study, we have applied cyclic combinational technique to get the minimized expression for 2-bit magnitude comparator.

A. 2-Bit Cyclic Comparator

The outcome of the comparison is specified by three binary variables that indicate whether A>B, A=B, or A<B. The circuit for comparing two n-bit numbers has $2^{2n}$ entries in the truth table. For simplicity a two bit comparator is taken for example. A two bit comparator has four input lines namely $A_1$ and $A_0$ representing A number and $B_1$ and $B_0$ representing number B. The truth table of a 2-bit comparator is shown below in Table - 1.

With the help of Quine-McKuskey method the output functions can be represented as

\[
E (A=B) = (A_1 \lor B_1).(A_0 \lor B_0) \quad (1)
\]

\[
G(A>B) = A_1.B_1' + (A_1 \lor B_1).(A_0 \lor B_0') \quad (2)
\]

\[
L(A<B) = A_1'.B_1 + (A_1 \lor B_1).(A_0' \lor B_0) \quad (3)
\]
TABLE I- TRUTH TABLE OF A 2-BIT COMPARATOR

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_1$, $A_0$, $B_1$, $B_0$</td>
<td>$A&gt;B$, $A=B$, $A&lt;B$</td>
</tr>
<tr>
<td>0, 0, 0, 0</td>
<td>0, 1, 0</td>
</tr>
<tr>
<td>0, 0, 1, 0</td>
<td>0, 0, 1</td>
</tr>
<tr>
<td>0, 1, 0, 1</td>
<td>0, 0, 1</td>
</tr>
<tr>
<td>0, 1, 1, 0</td>
<td>0, 0, 1</td>
</tr>
<tr>
<td>1, 0, 0, 1</td>
<td>0, 1, 0</td>
</tr>
<tr>
<td>1, 0, 1, 0</td>
<td>0, 1, 0</td>
</tr>
<tr>
<td>1, 1, 0, 1</td>
<td>0, 1, 0</td>
</tr>
<tr>
<td>1, 1, 1, 0</td>
<td>0, 1, 0</td>
</tr>
</tbody>
</table>

The expression shows acyclic representations of 2-bit comparator. Now the same outputs are represented in form of dependency with each other in order to make these functions cyclic. So the dependency graph of 2-bit comparator can be drawn as shown in Fig. 3 below with help of methodology explained in section 2.

From this graph it can be determined that the output $A>B$ is depending over $A<B$, whereas $A=B$ is depending over $A>B$ and $A<B$ is depending over $A=B$. The minimized expression for cyclic comparator is derived with the help of Quine-Mckulskey Method and are expressed below:

$$E(A\equiv B) = G'(A_0+B_0').(A_1+B_1')$$

$$G(A>B) = A_1.B_1' + A_0.B_0'.L'$$

$$L(A<B) = A_1'.B_1 + E'.B_0(A_1' + B_1)$$

V. RESULTS AND DISCUSSION

The expressions for 2-bit conventional comparator given in section IV is designed at transistor level using gpdk 45 nm technology in cadence tools. The circuit consists of 31 P-type and N-type MOS with channel length 45 nm and width 120 nm. Schematic diagram of 2-bit cyclic comparator using conventional combinational logic in virtuoso schematic editor is shown in Fig. 4.

The circuit shown in Fig. 4 has been simulated with 1 V power supply. The circuit is applied with different input combinations for 0ns to 160ns. The input & output waveform is shown below in Fig. 5.

The complete layout for the conventional 2-Bit Comparator has been done using Cadence Layout XL Suite which is shown in Fig. 6. Design verification was done using ASSURA Design Rule Check(DRC). Chip level extraction was done after checking Layout Versus Schematic(LVS). Post layout simulation confirmed that proposed design will work after chip fabrication.

The expressions for 2-bit cyclic comparator given in section IV is designed at transistor level using gpdk 45 nm technology in cadence tools. The circuit consists of 23 P-type and N-type MOS with channel length 45 nm and width 120 nm. Schematic diagram of 2-bit cyclic comparator using conventional combinational logic in virtuoso schematic editor is shown in Fig. 7.

The circuit shown in Fig. 7 has been simulated with 1 V power supply. The circuit is applied with different input combinations for 0ns to 160ns. The input & output waveform is shown below in Fig. 8.
Fig. 4- MOS Level Design of 2 Bit Conventional Comparator

Fig. 5- Output Waveform for 2-Bit Conventional Comparator

Fig. 6- Layout of 2-Bit Conventional Comparator
Fig. 7- MOS Level Design of 2 Bit Cyclic Comparator

Fig. 8- Output waveform for 2-Bit Cyclic Comparator

Fig. 9- Layout of 2-Bit Cyclic Comparator
The complete layout for the cyclic 2-Bit Comparator has been done using Cadence Layout XL Suite which is shown in Fig. 9. Design verification was done using ASSURA Design Rule Check (DRC). Chip level extraction was done after checking Layout Versus Schematic (LVS). Post layout simulation confirmed that proposed design will work after chip fabrication. After simulating the design of 2-Bit comparator using conventional and cyclic technique, we have found that the average delay for cyclic comparator is less as compared to the conventional one. Static and dynamic power consumption has also reduced for cyclic comparator. There is a saving of 14% and 26% in the dynamic and static power respectively. Layout area of cyclic comparator is 34% less than that of conventional one. The detailed comparison between conventional and cyclic comparator is shown Table-2.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Conventional 2-Bit Comparator</th>
<th>Cyclic 2-Bit Comparator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression</td>
<td>( E(A=B) = (A_1 \oplus B_1)(A_0 \oplus B_0) )</td>
<td>( E(A=B) = G'(A_0 + B_0')(A_1 + B_1') )</td>
</tr>
<tr>
<td></td>
<td>( G(A&gt;B) = A_1.B_1' + (A_1 \oplus B_1)(A_0 \oplus B_0') )</td>
<td>( G(A&gt;B) = A_1.B_1' + A_0.B_0'.L' )</td>
</tr>
<tr>
<td></td>
<td>( L(A&lt;B) = A_1'.B_1 + (A_1 \oplus B_1)(A_0' \oplus B_0) )</td>
<td>( L(A&lt;B) = A_1'.B_1 + E'.B_0(A_1' + B_1) )</td>
</tr>
<tr>
<td>No. Of Transistors</td>
<td>PMOS = 31, NMOS = 31</td>
<td>PMOS = 23, NMOS = 23</td>
</tr>
<tr>
<td>Average Propagation Delay</td>
<td>( G=100\text{ps} )</td>
<td>( G=80\text{ps} )</td>
</tr>
<tr>
<td></td>
<td>( L=120\text{ps} )</td>
<td>( L=90\text{ps} )</td>
</tr>
<tr>
<td></td>
<td>( E=135\text{ps} )</td>
<td>( E=110\text{ps} )</td>
</tr>
<tr>
<td>Static Power</td>
<td>( G=17.04\text{pw} )</td>
<td>( G=15.58\text{pw} )</td>
</tr>
<tr>
<td></td>
<td>( L=17.06\text{pw} )</td>
<td>( L=12.31\text{pw} )</td>
</tr>
<tr>
<td></td>
<td>( E=35.83\text{pw} )</td>
<td>( E=23.97\text{pw} )</td>
</tr>
<tr>
<td>Dynamic Power</td>
<td>( G=69.4\text{nw} )</td>
<td>( G=54.23\text{nw} )</td>
</tr>
<tr>
<td></td>
<td>( L=45.3\text{nw} )</td>
<td>( L=37.06\text{nw} )</td>
</tr>
<tr>
<td></td>
<td>( E=91.11\text{pw} )</td>
<td>( E=85.83\text{nw} )</td>
</tr>
<tr>
<td>Layout Area</td>
<td>53.36(\mu\text{m}^2)</td>
<td>35.2(\mu\text{m}^2)</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

In this paper a popular combinational logic circuit like magnitude comparator was chosen as target circuit for case study of cyclic combinational design methodologies. The motivation behind choosing this circuit is that there is moderate amount of redundancy and function complexity. The effectiveness of cyclic combinational circuits is best viewed when there are more than two output functions based on same input variables. This is exactly the case with the 2-bit magnitude comparator with four common input variables and three output functions based on those four input variables. The work aimed to optimize the transistor count and different power consumption parameters with speed of operation being secondary concern. Finally to implement the circuit on ASIC platform, the circuit and layout is designed and simulated using the industry standard Cadence Virtuoso EDA Design Suite to perform simulation, testing, layout, DRC, LVS and finally chip level extraction to make the design fabrication ready and ascertain the chip performance with parasitic components due to layout. The tools used for the above purpose are Virtuoso Schematic Editor, Virtuoso Analog Design Environment, Virtuoso Spectre Circuit Simulator, Virtuoso Layout Suite, Cadence Physical Verification System and Cadence QRC Extraction. The paper conclusively demonstrated that proper analysis and application of Cyclic combinational reduction methodologies can be applied to practical circuits like Binary Magnitude Comparator to achieve appreciable optimization in vital design parameters not only theoretically but also on transistor level for real time implementation. Cyclic combinational reduction methodologies can be applied to achieve large amount of optimization that no pure combinational reduction technique can achieve without compromising functionalities.

REFERENCES