CHAPTER 2

LITERATURE REVIEW

2.1 REVERSIBLE GATES

Reversible computing has known significant developments in the last
decade. Logic synthesis methods have been developed for reversible circuits [13,
14]. In 1973 famous International Business Machines (IBM) researcher Bennett
[15], incidentally invented quantum cryptography and quantum teleportation. Till
today, Bennett is an active and widely respected leader in the growing fields of
quantum information and quantum communication. A logic gate L is reversible if,
for any output y, there is a unique input x such that applying \( L(x) = y \). If a gate L is
reversible, there is an inverse gate \( L' \) which maps y to x for which \( L(y) = x \). From
common logic gates, NOT is reversible, as the inputs are 0 or 1 and the outputs are
also 1 or 0. But the common AND gate is not reversible however. The inputs 00, 01
and 10 all get mapped to the output 0. Ordinarily, in a classical computer, the logic
gates other than the NOT gate are not reversible. Thus, for instance, for an AND
gate one cannot generally recover the two input bits from the output bit; the case
both input bits are 1 is the exception. Any unitary operation is reversible, hence
quantum networks effecting elementary arithmetic operations such as addition,
multiplication and exponentiation cannot be directly deduced from their classical
boolean. In literature there are various reversible gates available [16-20, 26] as
shown in Figure 2.1.

In 1980, Toffoli [18], showed that there exists a reversible gate which
could play a role of a universal gate for reversible circuits. In 1985, Feynman [17]
proposed the idea that every linear reversible function can be built by composing
only 2*2 Feynman gate and inverters. With b=0, Feynman gate is used as a fan-out
gate or copying gate. Also, Fredkin gate is a fundamental concept in reversible and quantum computing. In 1988, Margolus [20] proposed Margolus gate for reversible computation.

Figure 2.1 Various gates available in literature (a) Feynman gate (b) Fredkin gate (c) Toffoli gate (d) Kerntopf gate (e) R1 gate

2.2 CMOS IMPLEMENTATION

J. M. Wang, et al. (1994) proposed the new designs of EX-OR and EX-NOR functions at transistor level [21]. The EX-OR functionality is important part of reversible gate, it has been implemented using a four transistor EX-OR function design [21, 22], which proves to be better technique than static CMOS. More importantly, physical reversible combinational devices have been implemented using CMOS transmission gates [22, 25]. The power consumption of these devices is 10-15 times lower as compared to their conventional counterparts. With an ever-increasing quest for greater computing power on battery operated mobile devices,
design emphasis has shifted from optimizing delay, time, area, and size to minimize power dissipation while still maintaining the high performance.

2.3 REVERSIBLE LOGIC AND QUANTUM GATES

Reversible logic has received considerable attention in quantum computing [27]. Several such gates have been proposed over the past decades. Among them are the controlled-not (CNOT) proposed by Feynman [17], Toffoli [18], and Fredkin [19]. In addition, in literature a synthesis method for large functions that relies on gate libraries is also presented, which describes transformation rules for CNOT based circuits [28]. Some achievements of quantum information science can be described as generalizations or extensions of the classical theory that apply when information is represented as an analogue state in terms of classical bits. A classical computer is built using circuit having wires and logic gates. Similarly, quantum computer is built from a circuit having wires and quantum gates to transmit and manipulate quantum information. To perform logic computing, a universal complete logic set and a set of boolean logic gates are required that can perform AND, OR, NOT and FANIN and FAN OUT operations [29].

W. Hung et al. (2004) proposed the quantum logic synthesis by symbolic reachability analysis where primary inputs are purely binary [37]. The quantum cost analysis is also done for various reversible gates available in literature.

V. V. Shende et al. (2005) gave the concept of synthesis of quantum logic circuits [35]. In this work the synthesis of quantum circuits using functional blocks was proposed, that realize quantum logic gates physically for synthesis of quantum circuits.

D. M. Miller et al. (2006) proposed the concept of quantum multiple valued circuits [34]. Miller proposed a simple heuristic algorithm that exploits the bidirectional synthesis possibility inherent in the reversibility of the specification. The primitive reversible gates considered here are multiple valued logic (MVL) which are extensions of the well-known binary toffoli gates.
M. Saeedi (2007) proposed CNOT based quantum gates synthesis which uses optimized algorithms and can be even extended for other reversible gates also in future [80].

M. S. Islam et al. (2009) proposed the design of multiplier using reversible gates with reduced quantum cost [57]. The quantum cost analysis is done by validating the quantum circuit for the particular reversible gate.

The nature of quantum particles and quantum environment allows quantum elementary particles to have in general more than two states [30, 31]. Depending on the quantity that is being measured, the particles can have two, three, high natural number or even infinity quantum states. For instance, such measured quantity can be the spin, the energy or the position in a molecule. Depending on the technology used, these particles can be atoms, molecules, photons or electrons. However so far, most of the quantum logic synthesis methods implicitly assume that the underlying mechanisms of quantum computing are quantum binary and reversible logic [32-37]. This assumption is correct for classical technology where the radix-2 logic is easiest to implement. In the proposed work the quantum simulation of 5-input and 5-output reversible NTG is carried out which has testability property and no such designs are available in literature so far.

2.4 COMBINATIONAL AND SEQUENTIAL CIRCUIT IMPLEMENTATION

A wide range of design methodologies at different abstraction levels have been developed for reversible circuits. Synthesis of reversible circuits differs significantly from the traditional logic synthesis. As far as the number of gates realization is concerned from a two input reversible gate the total 4! gates can be realized which is 24, similarly using three input reversible gate the total 8! i.e. 40320 gates can be realized. There exist some different types of adders presented in literature [43-45]. Various sequential elements at logic level have been proposed [38]. The flip flop being a sequential element is designed efficiently using the reversible logic [39]. The designing of sequential circuits design is not that easy
because the care should be taken to avoid the direct feedback from output to input of the circuit while designing a sequential circuit [40]. The testability is also introduced along with reversibility which is used to realize the desired design [41]. Moreover there are some combinations for each gate to work appropriate. In [41, 42] testability in introduced, but number of garbage outputs, constant inputs and logical complexity can be improved further using proposed gate discussed in this research work.

The important algorithms proposed in literature for fast multiplication are booth multiplier, array multiplier and wallace tree multiplier [49]. The low power and high speed multipliers can be implemented with different logic styles. Since the multipliers have a significant impact on the performance of the entire system, many high-performance algorithms and architectures have been proposed to accelerate multiplication [50, 51]. In [52] the well-structured design of booth multiplier is given. Moreover, further optimization can be achieved using reversible logic. In [54 -57] the multipliers are implemented using reversible logic. But, in these papers the design is given either using 3*3 or 4*4 reversible gates which do not incorporate testability feature. Though, in paper [41, 42], the combined block termed as Rgate includes testability which is compared with the proposed gate implemented in this research work. It is also shown that the circuits implemented using proposed gate are advantageous in terms of logical complexity, delay, area and speed.

2.5 STRENGTH REDUCTION TECHNIQUES IN VLSI

Various strength reduction techniques in VLSI are available in literature. Subexpression elimination (SE) is one among them which was introduced by Hartley [58] as a method of utilizing redundancy between coefficients in FIR filters, and other was invented by Bull and Horrocks [59]. By combining subexpressions occurring often in coefficients, the numbers of adders required are reduced. In [60], Hartley introduced two algorithms.

The algorithm of Potkonjak, et al. selects sub expressions based on the estimate of cost savings [61]. In Pasko, et al. algorithm large common subexpressions (with many non-zero digits) are selected first and smaller
subexpressions are assigned later [62]. A comparison suggests that Hartley and Pasko’s methods achieve an adder’s cost as good as or lower than that of Potkonjak, et al. [61]. Subexpression elimination can also be applied to linear transformation [64]. These strength reduction techniques are used for further optimization of VLSI circuits and are implemented in this research work.

2.6 DISTRIBUTED ARITHMETIC AND FIR FILTERS

The digital filters are implemented using signed digit DA [65]. A low power FIR filter realization with differential coefficients and inputs has been implemented in [66] which is a multiplier free technique. Distributed arithmetic is a possible option for low power applications because it allows replacement of costly multiplies with shifts and table lookups [67, 68].

Software radio receivers [69, 70] require mixing, filtering and down sampling of received signals to allow data to be processed at a suitable rate. Part of this process can be achieved in FPGAs using a digital down converter (DDC). The use of software defined radio (SDR) technology is predicted to replace many of the traditional methods for implementing transmitters and receivers by offering a wide range of advantages including adaptability, reconfigurability, and multifunctionality encompassing modes of operation, radio frequency bands, air interfaces, and waveforms [70]. Research in this field is mainly directed towards improving the architecture and the computational efficiency of SDR systems. The most computationally intensive part of an SDR receiver is the channelizer since it operates at the highest sampling rate. SDR receiver has filtering operation also, which comes under its digital down converter (DDC) part.

The key functional units in a digital filter are delay, adder, and multiplier – out of which multiplier dominates the hardware complexity. It is well known that by representing filter coefficients as sum-of-powers-of-two (SOPOT), each multiplication in filtering can be replaced with simple shift-and-add operations [71-75]. The complexity of FIR filter is dominated by the number of adders or subtractors employed in the coefficient of the multipliers used. In this research work
the implementation of FIR filters is presented using a distributed arithmetic scheme with CSD representation of coefficients [84]. It is shown that the filter coefficients coded using CSD are advantageous in terms of area, speed and power as compared to SOPOT representation [72].