Computers and Electrical Engineering 44 (2015) 13–23

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

State assignment for area minimization of sequential circuits based on cuckoo search optimization ☆

Aiman H. El-Maleh [a], Sadiq M. Sait [a,b,*], Abubakar Bala [a]

[a] Computer Engineering Department, KFUPM, Dhahran, Saudi Arabia
[b] Center for Communications and IT Research, Research Institute, KFUPM, Dhahran, Saudi Arabia

ARTICLE INFO

Article history:
Received 5 November 2014
Received in revised form 9 March 2015
Accepted 9 March 2015
Available online 21 April 2015

Keywords:
Cuckoo search
State Assignment
Heuristics
Sequential circuit
Area minimization
Finite state machines

ABSTRACT

A major optimization problem in the synthesis of sequential circuits is State Assignment or State Encoding in Finite State Machines (FSMs). The state assignment of an FSM determines the complexity of its combinational circuit and thus area, delay, testability and power dissipation. Since optimal state assignment is an NP-hard problem and existing deterministic algorithms produce solutions far from best known solutions, we resort to the use of non-deterministic iterative optimization heuristics. This paper proposes the use of cuckoo search optimization (CSO) algorithm for solving the state assignment problem (SAP) of FSMs with the aim of minimizing area of the resulting sequential circuit. Results obtained from the CSO algorithm are compared with those obtained from binary particle swarm optimization (BPSO) algorithm, genetic algorithm (GA), and the well-known deterministic methods of NOVA and JEDI. The results indicate that CSO outperforms deterministic methods as well as other non-deterministic heuristic optimization methods.

1. Introduction

As the density and size of integrated circuits (ICs) keep increasing rapidly, area and power dissipation have become and are still a significant concern in Very Large Scale Integration (VLSI) designs. VLSI systems by nature are mostly sequential circuits and are modeled as finite state machines (FSMs). In FSMs, the behavior of sequential circuits is characterized by using symbolic names to represent states. State assignment is the mapping of the state names of an FSM to a set of binary codes. This mapping has a significant impact on the circuit area and power dissipation [1]. An example of an FSM is given in Table 1, which has 4 states, one input and one output. To understand the example in Table 1, consider the case when *Present State=S0*. If input *X=0*, then *Next State=S0* and *Output=1*, but if *X=1*, then *Next State=S2* and *Output=0*.

Since there are 4 states in the FSM, a 2-bit code is sufficient for encoding each state. Table 2 shows two typical state assignments for the FSM labeled as “Ass. 1” and “Ass. 2”. The number of literals of the Boolean equations that implement the FSM as a multi-level circuit with “Ass. 1” is 6 literals while that with “Ass. 2” is 14 literals. The number of literals is a cost measure that correlates with the number of transistors in the circuit and hence its area.

☆ Reviews processed and approved for publication by the Editor-in-Chief.

* Corresponding author at: Computer Engineering Department, KFUPM-#673, Dhahran-#31261, Saudi Arabia.
E-mail addresses: aimane@kfupm.edu.sa (A.H. El-Maleh), sadiq@kfupm.edu.sa (S.M. Sait), g201201620@kfupm.edu.sa (A. Bala).

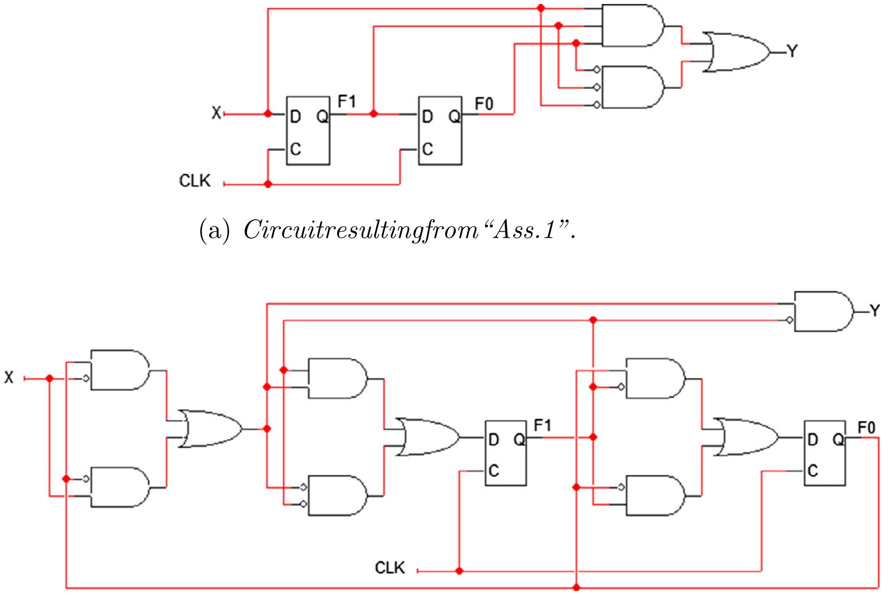
http://dx.doi.org/10.1016/j.compeleceng.2015.03.014
0045-7906/© 2015 Elsevier Ltd. All rights reserved.

Table 1
An example of an FSM.

Present state	Next state		Output	
	$X = 0$	$X = 1$	$X = 0$	$X = 1$
S0	S0	S2	1	0
S1	S0	S2	0	0
S2	S1	S3	0	0
S3	S1	S3	0	1

Table 2
State assignment for “Ass. 1” and “Ass. 2” and their resulting literal count.

State	“Ass. 1”	“Ass. 2”
S0	00	01
S1	01	10
S2	10	11
S3	11	00
Area (no. of literals)	6	14



(a) *Circuitresultingfrom“Ass.1”.*

(b) *Circuitresultingfrom“Ass.2”.*

Fig. 1. Multi-level circuits synthesized based on “Ass. 1” and “Ass. 2” for the FSM example.

The multi-level circuits resulting from synthesizing the FSM example using “Ass. 1” and “Ass. 2” are shown in Fig. 1. This example demonstrates the significant impact of state assignment on the area of a synthesized sequential circuit.

Formally, the state-assignment problem of an FSM is one that maps state symbols to binary codes using the mapping function $f : S \rightarrow B^n$, where n is the code length, $n \geqslant \lceil \log_2 |S| \rceil$, B^n is an n-dimensional Boolean hypercube and $|S|$ is the number of states. To encode S states, using k bits, the number of possible state-assignment combinations is given in Eq. (1).

$$\frac{(2^k)!}{(2^k - |S|)!} \tag{1}$$

As an illustration, if we have an FSM with 10 states, then each state will require 4 bits for distinctive encoding. As a result, the number of possible state assignments for these 10 states as obtained from Eq. (1) is 29,059,430,400. Hence, exhausting all the