Distributed-Lock Relation for Scalable-Delay-Insensitive Optimization in Multiple-Cycle STG Specifications

Pitchayapatchaya Srikram and Arthit Thongtak

***Abstract*— Signal transition graph specification has a potential to describe behavior of hardware system in term of concurrent, sequential and one instance of the same events. One typical idea is for asynchronous control circuits, which is a variety of delay assumption design by means of signal transition graph specification. This paper proposes a distributed lock relation to determine the completion path for multiple-cycle signals. We select the tardy internal-completion signal to be the volunteer signal based on Scalable-Delay-Insensitive (SDI) model. The effectiveness of the proposed methodology is evaluated by cost of area, which is number of internal input signals and literal logic gates.**

***Index Terms*— Scalable-insensitive delay model, signal transition graphs, Asynchronous control circuits, logic synthesis, multiple-cycle signal**

I. Introduction

The recent advancement in asynchronous control circuits design , the absence of clock circuits, there has been considerable in an wire fork that lead to design sophisticated asynchronous circuits under unbounded gate and wire delay assumption . As regards the Quasi-Delay-Insensitive (QDI) model was designed on the basis of wire forks, on which of the propagation delay of each output is poised on delay, called an isochronic fork. Since, the main challenge of design faced in designed isochronic fork with utilizing time information. It has been shown that two paths may travel through n gates before acknowledge another, is called extended isochronic forks or denoted as Q^nDI[1], likewise difference propagation delays, called an asymmetric isochronic forks assumption [2]. The delay assumption of isochronic forks is still not completely understood; However, scalable-delay-insensitive (SDI) model is one of major design considerations that alleviated the ascetic completed signal , this assumes on relative delay ratio between any two component is bounded by K value that guarantee the correct circuit operation [3]. Although considerable a research has been done on asynchronous combinational circuit on data path, rather than describe design in term of asynchronous control circuit in the class of asynchronous sequential circuits. In current practice the design of asynchronous SDI control circuits has been present SDI optimization [4]. As with the SDI optimization, the approach is modified each a concurrent transition model on wholly of Signal Transition Graph (STG), there is concurrent relation of primary-input signal transition and non-primary signal transition, if the underlying STG satisfies its property previously. After SDI optimization, the approach is satisfied STG properties such as persistence and complete state coding, but it is not satisfied safeness and liveness in the event that is contained multiple token. One solution has been presented the determined concurrent transition model whether it can be optimized based on SDI model by using lock relation [5]. The result of above approach indicated that is reduced area of circuits; However, all the previously mention approaches suffer from some limitations for handled some multiple-cycle signal at STG domain.

This paper is introduced a distributed-lock relation in order to simplified multiple-cycle signal. This investigation is taken the form of a case-study of the design of asynchronous control circuit by using the novel SDI optimization. To design such circuit at STG domain with multiple-cycle signal, since each transition on the wire fork needs to be acknowledged explicitly, in the other word, it is a casual relation. This is exemplified in the research undertaken by determination of the volunteer signal, which is a tardy acknowledgement of completion path signal. This proposed methodology is demonstrated through experimental result as an example on optimization and implementation of asynchronous control circuits based on SDI model. The procedure of this method is illustrated in Fig1, which is compared to the previous method. As well as, this approach is not implemented to SDI circuit, if the giving STG has not to satisfy the implementable STG property. The synthesis of circuit from STG is based on S. Park method [6]

The core contribution of this paper can be summarized as follow: In the next section is briefly introduced the related work, an overview of the STG notation used in this paper, the basic of lock relation properties. In addition, this section is presented the distributed lock relation specification in detail and the definition of Scalable-Delay-Insensitive (SDI) model. Then, Section III discusses distributed-lock relation based on design style of SDI model. Section IV is proposed method of SDI implementation and optimization. Then, the experimental results and conclusions are given in Sections V and VI, respectively.

Manuscript received January 18, 2017; revised January 26, 2017.

Pitchayapatchaya Srikram is a Research Student with Department of Computer Engineering,Faculty of Engineering, Chulalongkorn University. (e-mail:Pitchayapatchaya.s@hotmail.com).

Arthit Thongtak is an Assistant Professor of Computer Engineering at Faculty of Engineering, Chulalongkorn University (e-mail:arthit.t@chula.ac.th).

ISBN: 978-988-14047-7-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

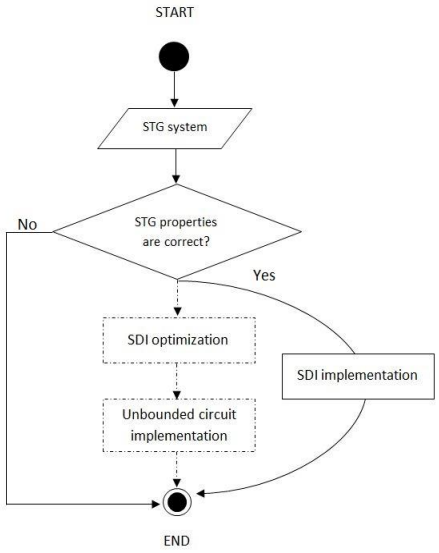

Fig.1 Overall procedure for SDI implementation

II. Related Work

The following is the brief description of overview of signal transition graph, lock relation and Scalable-Delay-Insensitive model definition.

A. Signal Transition Graph (STG) Notation: An Overview

The signal Transition Graph (STG) is formalization of characteristic of asynchronous control circuits in described. [7] This is interpreted as live safe free-choice Petri nets. The signal transition is represented as rising and falling, It seem, event-driven on timing diagrams, if whose transitions are rising and falling immediately, is called single-cycle signal as a^+ and a^-. The transition $a*$ denotes either a^+ or a^-, while $\overline{a*}$ denotes the complement of $a*$. Otherwise, the same of signal is rising and falling various times. This is called multiple-cycle signal as a^+/t and a^-/t. The arcs are represented casual relation i.e. *ai* is trigger signal for *bo*, when *bo* is achieved transition, it means that *bo* is acknowledge signal transition for *ai*. Marking (•, token) is described the enabled transition which $\mu 0$ is initial marking.

However, the STG is implementable, this is be satisfied following properties:

--**Safeness** if, it is a strongly-connected graph and every simple cycle in STG has exactly one marking (token, μ).

--**Liveness**, the initial marking can be reached enabled all the transition in STG.

--**Persistency,** the signal transition must not be disabled by another signal transition.

--**Complete state coding** (CSC), all the signal transition is produce the difference binary code.

--**Simple cycle,** the signal transition is sequenced as t1…titj…t1 in STG.

B. Formal Definition of Lock Relation

Lock relation describe the interleaving signal transitions between couple signal, there is causal relation by live STG which is formalized two tuple G =<V, E>. Previous research has shown that the lock relation is sufficient condition for implementation circuit and verification of STG properties which is implementable circuit [6, 8].

--**Full-lock** if, the two signals *a* and *b* on simple cycle, are interleaved its transition that $a* \to b* \to \overline{a*} \to \overline{b*}$

--**Semi-lock** if, the two signals a and b on simple cycle, are interleaved its transition that $b* \to \overline{a*} \to \overline{b*}$ or $a* \to b* \to \overline{a*}$

--**Associate-lock** if**,** A is minimal set of full-lock relation of two signals as a1 and a2, and the any transition of set A is fully-locked with signal b , this is such that $\exists a1, a2 \in A : a1 \to b* \to a2 \to \overline{b*}$ on simple cycle and also called *transitive-lock relation*. The STG is satisfied lock-relation, this is enable implementing hazard-free circuits.

According to lock relation is interpreted not only single-cycle signal but also multiple-cycle signal in described. It has been is presented solving state coding problem on STG domain by transitive-lock relation on which contains multiple-cycle signal [9]. One is splitting multiple-cycle into virtual single-cycle signal. Meanwhile, this paper is developed multiple-cycle signal specification by a distributed-lock relation.

--**Distributed-lock** if, all candidate transition of multiple- cycle signal *a/t* are full-lock relation with signal b and c, then signal *b* and *c* are fully locked. This is denoted as $a/tL(bLc), t \ni \{1,2,...,n+1\}$

***Proof*:** if the multiple-cycle signal *a/t* is full-lock relation with signal b on the simple cycle such as in (1):

$$\begin{aligned} &\exists a/t : a/t* \to b* \to \overline{a/t*} \to a/(t+1)* \to \overline{b*} \to \overline{a/t*} \\ or,\ &\exists a/t : a/t* \to \overline{a/t*} \to b* \to a/(t+1)* \to \overline{a/t*} \to \overline{b*} \end{aligned} \quad (1)$$

While, the multiple cycle signal *a/t* is full-lock relation with signal *c* on the simple cycle such as in (2):

$$\begin{aligned} &\exists a/t : a/t* \to c* \to \overline{a/t*} \to a/(t+1)* \to \overline{c*} \to \overline{a/t*} \\ or,\ &\exists a/t : a/t* \to \overline{a/t*} \to c* \to a/(t+1)* \to \overline{a/t*} \to \overline{c*} \end{aligned} \quad (2)$$

Thus (1) and (2), there are formalized as in (3):

$$\begin{aligned} &\{[(a/t \to b*) \cup (a/t \to \overline{b*})] \cup \\ &\qquad [(a/t \to c*) \cup (a/t \to \overline{c*})]\} \end{aligned} \quad (3)$$

Then (3), there is summarized as in (4) and the final result of summarization of (4) is formalized as in (5)

$$(a/tLb) \cup (a/tLc) : b \ni \{b*, \overline{b*}\} \text{ and } c \ni \{c*, \overline{c*}\} \quad (4)$$

$$\begin{aligned} &a/tL(bLc), a/t \ni \{a/t*, \overline{a/t*}, ..., \\ &\qquad a/t(n+1)*, \overline{a/t(n+1)*}\}, n \ni \{1,2,...,n+1\} \end{aligned} \quad (5)$$

■

C. Scalable-Delay-Insensitive (SDI) Model

The SDI model [10] refers to an interconnection of two components (C1, C2). There is *t0* signal that causes to *t1* and

ISBN: 978-988-14047-7-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

t2 signal as illustrated in Fig2. The propagation delay paths of two circuit components from *t0* to *t1* denote delay as D1 and another from *t0* to *t2* denote delay as D2, Let De1 and De2 refer to estimated delay for two paths accordingly, and Da1 and Da2 refer to actual delay that occur in two paths.

K is constant of margin for correctness operation circuit based on SDI model, is called the maximum variation ratio. For the present case under consider signal transition *t1* is specified to precede *t2*. Consequently, The K value is given for De1 such that the relation $K \cdot De1 < De2$, then $Da1 < Da2$. On the Contrary, if K value is given for De2 such that the relation $K \cdot De2 < De1$, then $Da2 < Da1$, since signal transition *t2* is preceded *t1*.

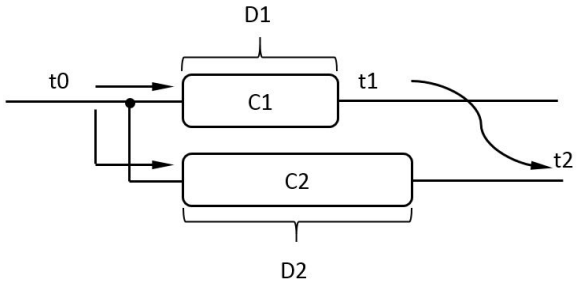

Fig.2 the SDI assumption model

III. Distributed-Lock Relation based on Design Style of SDI Model

In this section, this is approach to implement such circuit based on SDI design style from STG, which hold the multiple-cycle signal. As explained earlier, the isochronic forks is cause simultaneous pair of signal transition, then the conversional asynchronous control circuits confined to the each non-primary input for practicality and ease of implementation on unbounded delay model, which is not only caused by the primary-input signal, but also the other acknowledgement with a subsequent non-primary input signal as example illustrated in Fig3 (a), signal *t0* is trigger signal for signal *t1* and *t2*, both signal *t1* and *t2*, is caused by each other's acknowledgement signal transition (*t1'* and *t2'*) between two paths.

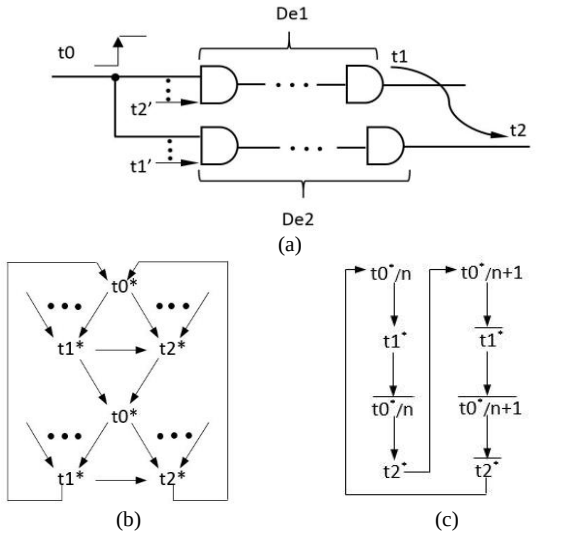

Fig.3 The SDI circuit model (a) and (b) simplify STG which are behaviorally equivalent and corresponding STG with multiple-cycle signal.

In term of STG, this characteristic refers to causal relation, which is interpreted STG as illustrated in the fig.3 (b). There are causal relation among *t0*, *t1* and *t2*. On the single-cycle signal *t0* is caused to accomplished signal *t1* and *t2*. As well as, Even if the circuit operation is multiple-cycle signal of *t0*, it is refers that *t0* can accomplish signal *t1* and *t2* with multiple-cycle signal. As mention above, this behavior is given by corresponding distributed-lock relation, as illustrated in Fig.3(c).

However, on the SDI model refers that any two path in which wire fork, one non-final transition signal is accomplished, the other is also accomplished by K value i.e. *t1* precede *t2*, this means that if *t2* is accomplished, *t1* is also accomplished. On the other hands, *t2* is volunteer successor acknowledge the *t1*. Therefore, *t2* is called volunteer signal. In term of delay, *t2* is tardy internal-completion signal. Therefore, the next section will be discussed the distributed-lock relation which is guarantee the completion path, there can be operation based on SDI model.

IV. SDI Optimization and Implementation

Since the distributed-lock relation can be handled STG, which is obtained the multiple-cycle signal based on SDI model, as described in the previous section this section is presented the method to implement and optimize circuit based on SDI model. A case-study approach was adopted to determine the distributed-lock relation on which the primary input signal is multiple-cycle signal and the full-lock relation set of non-primary input signal. However the terminologies of signal, these is describe on the algorithm as illustrate on Table I.

TABLE I
Terminologies for Signal and Arcs on SDI Design

Notation	Descriptions
$Vip_{(i)}(*/t)$	: set of primary input with multiple-cycle signal, $Vip_{(1)}(*/1), Vip_{(1)}(*/2), \ldots, Vip_{(n+1)}(*/n+1)$
$Vnp_{(i)}$	: set of non-primary input with single-cycle signal, $Vnp_{(1)}, Vnp_{(2)}, \ldots, Vnp_{(n+1)}$
$Vip_{(i)}$	: set of primary input with single-cycle signal, $Vip_{(1)}, Vip_{(2)}, \ldots, Vip_{(n+1)}$
$V_{(i)}$	: set of both primary input signal and non-primary input signal with single-cycle signal, $V_{(1)}, V_{(2)}, \ldots, V_{(n+1)}$
$Vop_{(i)}$	: set of non-primary input on which are candidate of distributed-lock relation, $Vop_{(1)}, Vop_{(2)}, \ldots, Vop_{(n+1)}$
$\bullet\, enp_{(i)}$	: set of incoming marking on which is arc of non-primary input, $[\, {}^{\bullet}enp_{(i)} \mid V_{(i)} \xrightarrow{enp_{(i)}} Vnp_{(i)} \,]$
$enp_{(i)}{}^{\bullet}$	: set of outgoing token on which is arc of non-primary input, $[\, enp_{(i)}{}^{\bullet} \mid Vnp_{(i)} \xrightarrow{enp_{(i)}} V_{(i)} \,]$ $i,t \ni \{1,2,3,..,n+1\}$

As regarding the methodology, which is consists of two steps; First step is to determine a volunteer signal, then established to a partial transition graph as described on the following;

ISBN: 978-988-14047-7-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

A. To Determine a Volunteer Signal

This algorithm is described to determine a volunteer signal on the distributed-lock relation as shown in Algorithm I and given as procedure example as illustrate in Fig.4.

Algorithm I : To Determine a Volunteer Signal

Input : The distributed-lock relation
Output: Volunteer signal (v)

1: **begin**
2: **while** distributed-lock $\ni < V_{ip(*/t)}, V_{np(1)}, V_{np(2)} >$ **do**
3: Let $<Vnp_{(1)}, Vnp_{(2)}> \in$ set of {full-lock on simple cycle $\subseteq \mu 0$}
4: Check : volunteer signal
5: **if** ($\mu 0 \mid \bullet enp_{(1)} \cup enp_{(2)}{}' \bullet \neq \emptyset$)
6: **then** Let volunteer signal (v) := $Vnp_{(2)}$
7: **endif**
8: **return** (volunteer signal (v))
9: **end**

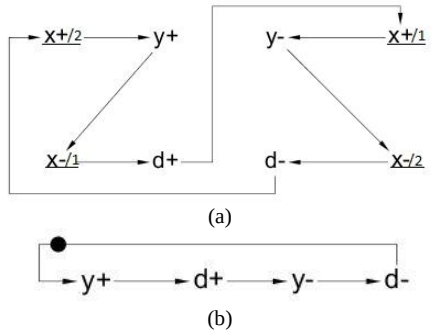

Fig. 4 The STG on which (a) is distributed-lock relation and (b) is to determine volunteer signal

Since the distributed-lock relation set of <*X*/t*, *y**, *d**> as illustrated in Fig.4 (a) then extract full-lock relation set of <*y**, *d**> including the initial marking as illustrated in Fig.4 (b). The propagation timing assumption is given that if initial state is accomplished, then the final state is also accomplished. This can refer that if *y** is accomplish signal, then *d** is also accomplish. Therefore, *y** is a volunteer signal.

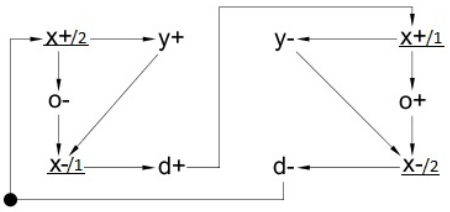

Fig. 5 The STG on which multiple-cycle signal is multiple distributed-lock relation

This algorithm is satisfied for multiple-distributed lock relation on STGs, if the multiple-cycle input signal is not redundant to another set of distributed-lock relation. The case of redundant primary input signal set of distributed-lock relation; However, the case-study of SDI model can guarantee only the pair of non-primary input signal on which caused by the same input signal. Therefore, the designer have to decide on the selecting guarantee path. As illustrated in Fig.5, the signal *X*/t* is distributed-lock relation not only set of *y** and *d** but also the set of *o** and *d**. The guarantee path can be either set of distributed-lock relation <*X*/t*, *y**, *d**> or set of distributed-lock relation <*X*/t*, *o**, *d**>.

B. Established Partial Transition Graph for Each Non-Primary Signal on Distributed-Lock Relation

Algorithm II : Established Partial Transition Graph

Input : The given STG (G)
Output : The partial STG for each non-primary input on distributed-lock relation $g(Vop_{(i)})$

1: **begin**
2: **do** the partial transition graph for each output signal:
$\{g(Vop_{(i)}) \ni \bigcup_{i=1}^{n} Vip_{(i)}\}$
3: Let $g(v) \ni \{G(\bigcup_{i=1}^{n} Vip_{(i)}(*/t) \cup \bigcup_{i=1}^{n} Vip_{(i)} \cup \bigcup_{i=1}^{n} Vnp_{(i)})\}$
4: Let $g(Vnp_{(1)}) \ni \{G(\bigcup_{i=1}^{n} Vip_{(i)}(*/t) \cup \bigcup_{i=1}^{n} Vip_{(i)} \cup (\bigcup_{i=1}^{n} Vnp_{(i)} - v))\}$
5: **return** (the partial transition graph : $g(v), g(Vnp_{(1)})$)
6: **end**

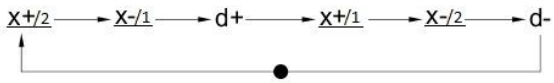

Fig. 6 The partial transition graph for any non-primary input without volunteer signal

Due to each non-primary input signal needs to be acknowledged explicitly; however, the volunteer signal is implied as successor acknowledged path. This is lead it to be negligible in input signal, Therefore, As shown in Algorithm II, which is describe the established partial transition graph for each non-primary input on the distributed-lock relation which is eliminated the volunteer signal of whose other's input signal from the given STG. As given STG example as illustrated in Fig.4 (a), the volunteer signal is *y**. Thus, *y** was eliminate of *d*s' input signal as illustrated in Fig.6.

V. Experiment And Result

As was pointed out introduction to this research paper, our proposed method is significantly to improve area cost of asynchronous control circuit with multiple-cycle signal. In this section has demonstrated a through experimental result as providing another STG example on optimization and implementation of asynchronous control circuits based on SDI model as illustrate in Fig 7 (a).

As illustrate in Fig. 7 (a) the providing STG has a multiple-cycle input signal (*Ai*/t*), non-primary input signal (*Co**, *Xo**) and multiple-cycle output signal (*Bo*/t*).There is a set of distributed-lock relation <*Ai*/t*, *Co**, *Xo**>, then it form on its simple cycle signal as illustrate in Fig.7 (b). As

ISBN: 978-988-14047-7-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

the distributed-lock relation is determine the volunteer signal as Co* as illustrated in Fig.7 (c).

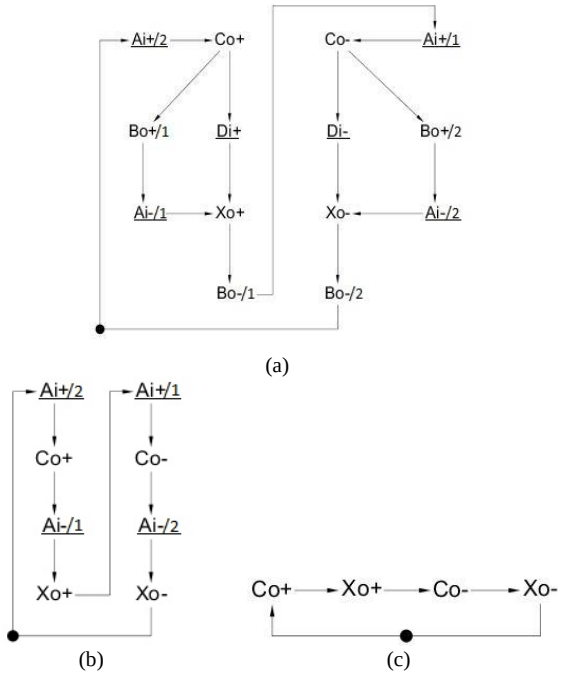

Fig.7 As Example (a) the providing STG (b) distributed-lock relation of providing STG (c) to determine the volunteer signal on simple cycle

For SDI implementation, the partial transition graph was established for *Xo** and *Co** circuit implementation as illustrated in Fig.8 (a) and (b), respectively. The partial transition graph for Xo* which signal *CO** was eliminated for the non-primary input signal of signal *Xo**.

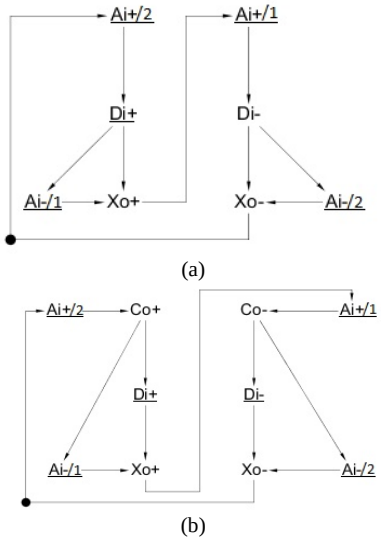

Fig. 8 The partial transition graph from STG fig.7 (a) are (a) for Xo circuit implementation and (b) for Co circuit implementation.

As the result of the providing STG, the circuit implementation based on QDI model and SDI model, which are based on S.Park's synthesized method as illustrated on Fig9(a), and Fig9(b), respectively. This is indicated that SDI implementation circuit is smaller than QDI implementation circuit.

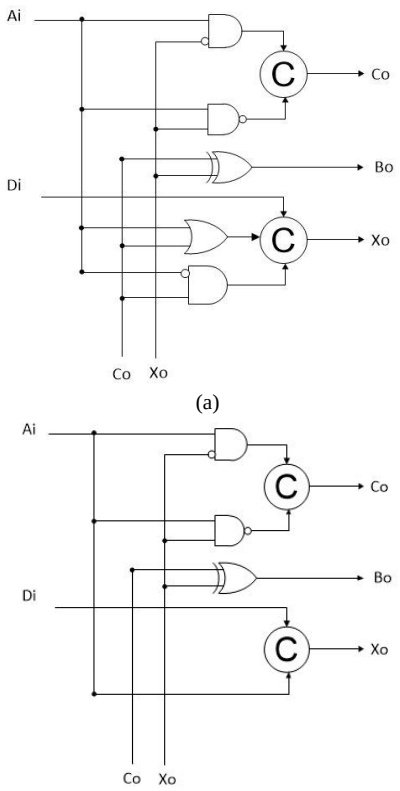

Fig.9 As result of wholly signal of circuit implementation (a) is based on QDI and (b) SDI

Since, the effectiveness of the proposed methodology is evaluated by comparing as result cost area of circuit implementation, which is literal counts of non-primary input, number of conventional logic gate and number of c-element on which obtain 4 NAND gate. The table below illustrate the circuit implementation from the other STG benchmarks which is comparing with QDI model and the present method as illustrated on Table II.

TABLE II
COMPARING WITH QDI IMPLEMENTATION

Example	The present method		QDI Implementation	
	#literal	#gate/ #c-element	#literal	#gate/ #c-element
ebergen (multiple-cycle)	4	3/2	6	5/2
Fig.9 converta (multiple-cycle)	10	6/3	12	7/3
wrdata (multiple-cycle)	10	1/3	10	1/3
wrdatab (multiple-cycle)	27	10/5	27	10/5

As the table III illustrate, there is a significant interesting result of circuit implementation between the other method and the present method. There is indicated that the present method is perform for handled STG with multiple-cycle signal.

This can summarize that the present method can be improved the cost area of implementation circuit, if the STG have a distributed-lock relation.

ISBN: 978-988-14047-7-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE III
COMPARING WITH THE OTHER METHOD

Example	The present method		The other method[5]	
	#literal	#gate/ #c-element	#literal	#gate/ #c-element
ebergen (multiple-cycle) Fig.9	4	3/2	N/A	N/A
converta (multiple-cycle)	10	6/3	N/A	N/A
wrdata (multiple-cycle)	10	1/3	8	2/3
wrdatab (multiple-cycle)	27	10/5	27	10/5

VI. CONCLUSION

In this paper, we have presented the novel design of SDI optimization and implementation for STGs with multiple-cycle signals. We also introduce distributed-lock relation to simplify the multiple-cycle on the STG. Our method determines the guaranteed path which is undertaken K value using distributed-lock relation as corresponding SDI model. In order to determine volunteer signal, which is a volunteer successor acknowledge for all completion signals on guarantee path. Then, this method is established to the partial transition graph for other non-primary input signal on distributed-lock relation, on which eliminated volunteer signal. As the result of this study shows that the improving cost area of implemented circuit is depended on distributed-lock relation of whose STG. In the future, we plan to establish to the element circuit for synthesized SDI circuit with single-cycle and multiple-cycle signal.

REFERENCES

[1] K. van Berkel, A.Peeters and F.Beest, "Stretching Quasi Delay Insensitivity by Means of Extended Isochronic Forks," Proceeding of 2nd Working Conference Asynchronous Design Methodologies, 1995, pp. 99-106.

[2] K van Berkel "Beware the isochronic fork" *Tech. Rep. UR 003/91, Philips Research Laboratorise, Netherland,* 1991.

[3] T. Nanya, A. Takamura, M. Kuwako, M. Imai, M. Ozawa, M. Ozcan,r, R. Morizawa and H. Nakamura, "Scalable-Delay-Insensitive Design: A High-performance Approach to Dependable AsynchronousSystems" (Invited paper). *Proceedings of International Symposium on Future of Intellectual Integrated Electronics*, (March 1999) .pp. 531- 540.

[4] P.Srikram and A.Thongtak, "A design of Asynchronous Control Circuits Based on SDI Model, " *the International MutiConference of Engineering and Computer Scientists (IMECS 2014)*, Hong Kong,pp.728-732,March 2014

[5] P.Srikram and A.Thongtak, "Scalable-Dalay-Insensitive Optimization based on Lock Relation," Internation Technical Conference on Circuits Systems, Computer and Communication (ITC-CSCC 2015), Korea,pp.786-789,June-July 2015

[6] S.B. Park, "Synthesis of Asynchronous VLSI circuits from Signal Transition Graph Specifications", Phd. Thesis, Tokyo Institute of Technology, Japan (1996)

[7] J.Sparso and Steve Furber, "Principles of Asynchronous Circuit Design,a system perspective" Springer Publishing Company,2010.

[8] W.Lawsunnee, A.Thongtak and W. Vatanawood "Signal Persistence Checking of Asynchronous System Implementation using SPIN," *the International MutiConference of Engineering and Computer Scientists (IMECS 2015),*Hong Kong,pp.604-609, March 2015

[9] K.J.Lin, J.W.Kuo, and C.S.Lin "Direct synthesis of hazard-free asynchronous circuits from STGs based on lock relation and MG-decomposition approach,"*EDAC,* pp.178-183, 1994.

[10] M. Imai, M. Ozcan, T. Nanya. "Evaluation of Delay Variation in Asynchronous Circuits based on the Scalable-Delay-Insensitive Model" the 10th International Symposium on Asynchronous Circuits and Systems (ASYNC'04), pp.62-71,April 2014

ISBN: 978-988-14047-7-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)