Advanced Science and Technology Letters
Vol.120 (GST 2015), pp.175-180
http://dx.doi.org/10.14257/astl.2015.120.33

Implementation of a Compositional Microprogram Control Unit with e-DIA

Young-Jin Oh, Gi-Yong Song[1]

[1]School of Electronics Engineering, Chungbuk National University, Cheongju, Chungbuk, 362-763, Korea
{ goodmen913, gysong }@chungbuk.ac.kr

Abstract. The CMCU(compositional microprogram control unit) and CMCU with DIA(dedicated input area) are reviewed, and then the concept of TMI field of a DIA is extended to e-DIA. The occurrence of idle cycles is removed and microoperation missing in some cases is protected by applying the e-DIA to CMCU. The improvements are validated through simulation.

Keywords: Compositional microprogram control unit, Control memory, Dedicated input area, e-DIA.

1 Introduction

Interpretation and implementation of control algorithms for digital systems are based on finite-state machines or microprogram control units. The finite-state machine implementing control algorithms for digital system is represented by sequential circuit consisting of combinational circuit CC and register RG, with microinstruction sequence appearing on the output of RG. Implementation of a FSM(finite-state machine) logic circuit includes several steps such as derivation of state graph, state assignment, construction of structure table, determination of next state and output function and configuration of logic circuit using some logical elements[1]. In microprogram control units, each instruction of a high-level programming language is interpreted into a special microprogram that is kept in a separate CM(control memory) as a sequence of microinstructions. Composition of the finite-state machine and microprogram unit leads to a CMCU(compositional microprogram control unit) that has several particularities[2]. Among them are microinstruction format with operational part only, minimum length microprogram and multidirectional transition in one cycle of operation. In CMCU, the input address is generated by the block CC, and some additional block for address generation needs to be adopted to reduce the number of outputs of block CC. The CMCU with DIA(dedicated input area) adopted additional control memory for that purpose. The introduction of DIA causes idle cycles in the datapath in some cases. In this paper, e-DIA is proposed that removes idle cycles as well as preventing some microoperations from missing.

[1] Corresponding Author : gysong@chungbuk.ac.kr

ISSN: 2287-1233 ASTL

2 Compositional microprogram control unit

The properties of control algorithm to be executed in control unit have great influence on the hardware amount of corresponding control unit[3-5]. One of such properties is operational linear chain included in control algorithm. A sequence of only operator vertices is called an OLC(operational linear chain)[1]. CMCU - one of the approaches implementing OLCs effectively - is a composition of a FSM(finite state machine) and microprogram control unit[2]. The structural diagram of CMCU is shown in Fig.1.

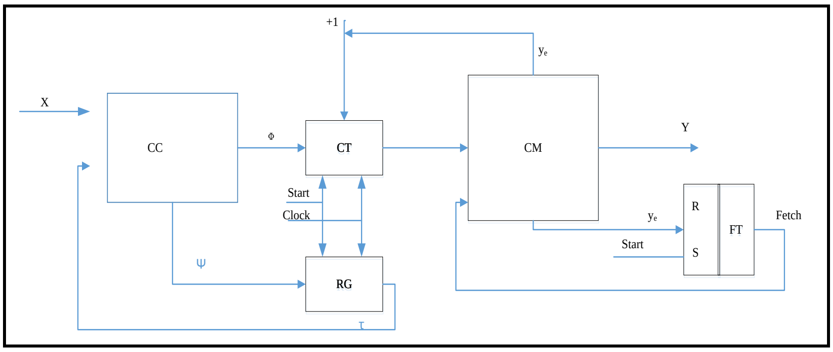

Fig. 1. The structural diagram of CMCU

Combinational circuit CC and register RG form an FSM for microinstruction addressing. Counter CT, control memory CM and start-stop flip-flop FT form microprogram control unit. Microinstructions for an OLC are placed in a CM area with consecutive address, and are fetched out sequentially for producing output signals. On reaching the output vertex of an OLC, y_0=0 is issued and CC generates an initial address of an OLC that is to be executed next, and that address is loaded into CT. At the same time CC determines the code of next state to be loaded into RG.

3 CMCU with e-DIA

In CMCU with DIA[2,6-7], DIA is a set of I_0 cells of CM having addresses from 0 to I_0-1, where I_0 is the number of OLC input in the FCA. In CMCU with DIA, field TMI from DIA area of CM is exclusively occupied with '00' indicating that next microinstruction address is always to be fetched from CM_0, and then loaded into CT without exception. This restriction on TMI code selection sometimes results in an idle cycle during execution of FCA(flow-chart of an algorithm). In e-DIA, field TMI is extended to codes other than '00' and the contents of corresponding location of AMP area is coordinated in accordance with the TMI code in e-DIA. Let us demonstrate the operation of a CMCU with e-DIA with a simple FCA shown in Fig. 2.[2] to examine the role of e-DIA.

The e-DIA consists of OLC input vertices marked with a circle in Fig.1.

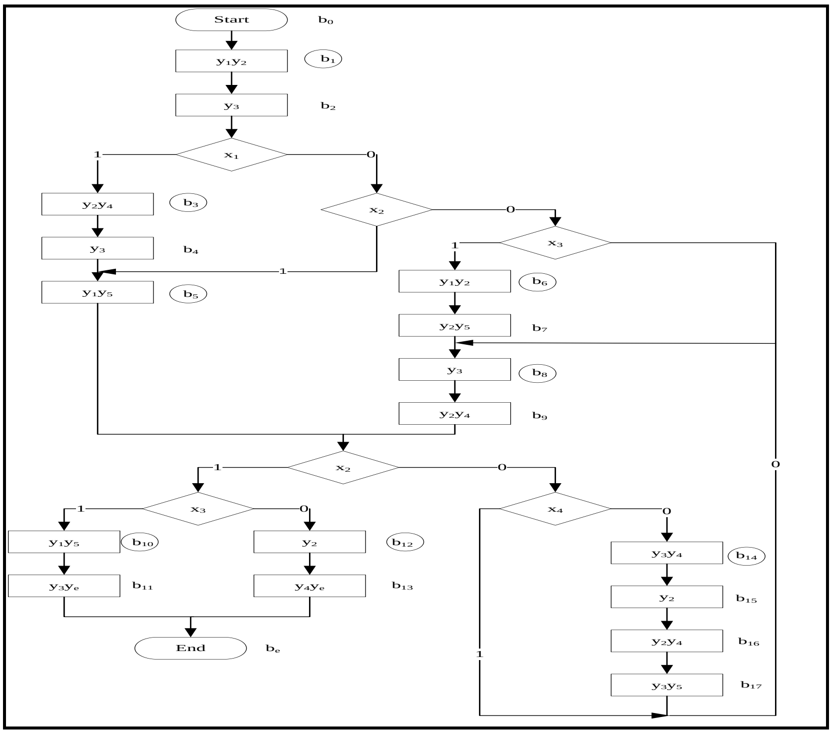

Fig. 2. Flow-chart of an algorithm

These operator vertices are assigned address in such a trivial way that first vertex is assigned address 0, second vertex address 1, and so on. After OLC input vertices are assigned addresses, the other operator vertices are assigned addresses in the same manner starting from next of the end of e-DIA address range, building up an AMP area. Contents of CM includes both e-DIA and AMP, and is fixed as shown in Table 1 in this case.

Table 1. Contents of CM

Address $T_1T_2T_3T_4T_5$	*Content* $m_1m_2m_3m_4m_5m_6m_7$
00000	0011000
00001	0001010
00010	1010001
00011	0011000
00100	0000100
00101	0010001
00110	0001000
00111	0000110
01000	1000100
01001	0100100
01010	1010001
01011	0101001
01100	0100100
01101	1001010
01110	1100100
01111	1100010
10000	0101000
10001	0101010
10010	1000101

First eight locations are occupied by e-DIA and the others by AMP. Each location of CM has TMI code and microoperation. Contents of CM_0 is fixed as shown in Table 2.

Table 2. Contents of CM_0

Address $T_3T_4T_5$	*Content* $a_1a_2a_3a_4a_5$
000	01000
001	01001
010	01010
011	01011
100	01101
101	01110
110	01111
111	10000

The column address represents the addresses of OLC inputs with minimal bits, and column contents keeps transition addresses from corresponding OLC input vertices to next operator vertices in CM.

In e-DIA, field TMI is not exclusively occupied by code '00', and at the same time some microinstruction appears at location of e-DIA area as well as at location of corresponding AMP area to coordinate the generation of microoperation when execution of FCA pass through operator vertex that is an OLC input as well as an OLC output. A CMCU with e-DIA having CM and CM_0 corresponding to FCA was implemented using VHDL[8], and then simulation[9] for some cases is performed to check for normal operation. Simulation wave views are shown in Fig. 3 and Fig. 4. The source code can be referred to at [10].

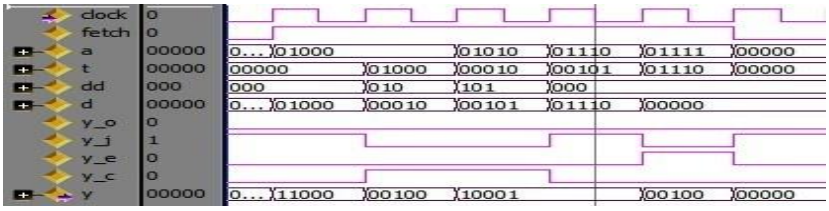

Fig. 3. Without idle cycle

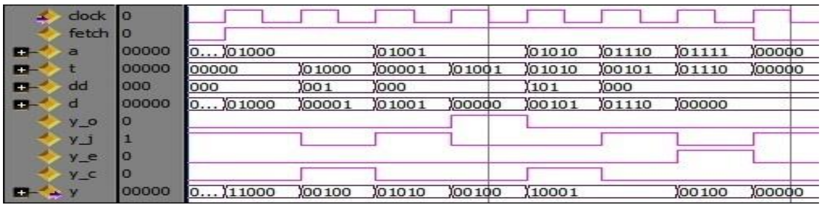

Fig. 4. Without microoperation missing

In Fig. 2, proceeding of b1 → b2 → b5 → b10 → b11 without an idle cycle is validated, and in Fig. 3, proceeding of b1 → b2 → b3 → b4 → b5 → b10 → b11 without microoperation missing is validated.

4 Conclusions

The CMCU and CMCU with DIA were reviewed, and then the concept of TMI field of DIA was extended to e-DIA by extending TMI code assignment and coordinating the contents of corresponding AMP location. By applying e-DIA to CMCU, the occurrence of idle cycles was removed and micrpoperation missing in some cases was protected. The improvement on idle cycle and microoperation missing were validated through simulation of VHDL implementation of CMCU with e-DIA.

Acknowledgments. The authors would like to express thanks to Prof. A. Barkalov, Prof. L. Titarenko and Prof. J. Bieganowski for their work on which this paper is based.

References

1. Barkalov, A., Titarenko, L.: Logic Synthesis for FSM-Based Control Units. Springer, Berlin (2009)
2. Barkalov, A., Titarenko, L.: Logic Synthesis for Compositional Microprogram Control Units. Springer, Berlin (2008)

3. Ki-Hwa Kim, A study on FPGA utilization For PC-Based Full-HD DVR System Impelmentation. Journal of the korea Academia-Industrial cooperation Society, Vol.15, No4, pp.2363—2369(2014)
4. Oh-Hyeong Kwon, Logic Optimization Using Boolean Resubstitution, Journal of the korea Academia-Industrial cooperation Society, Vol.10, No11, pp.3227—3233(2009)
5. Jongchul Beak, Hyungshin Kim, Soft error correction controller for FPGA configuration memory, Academia-Industrial cooperation Society, Vol.13, No11, pp.5465—5470(2012)
6. A.A. Barkalov, L.A. Titarenk, and J. Bieganowski. Synthesis of control unit with modified microinstructions. In Proc. of the Inter. Conf. Mixed Design of Integrated Circuits and Systems – MIXDES, pp.157--160. (2007)
7. A.A. Barkalov, L.A. Titarenk, and J. Bieganowski. Synthesis of control unit with modified system of microinstructions. In Proceedings of IEEE East-West Design & Test Symposium – EWDTS '07, pp. 545--549(2007)
8. Pedroni, V. A. Circuit Design and Simulation with VHDL, 2nd edition, Cambridge, MA: MIT Press(2010)
9. Altera Corporation, http://www.altera.com/
10. http://bandi.chungbuk.ac.kr/~gysong/

