



다중화기 기반 논리 설계를 위한 무정의 조건의 고찰 (아두이노 설계 교육에의 활용을 위한)

이재민

가톨릭관동대학교 전자공학과

Consideration of Don't-care Condition for Multiplexer-based Logic Design (For Application to Arduino-based Design Education)

Jae Min Lee

Department of Electronic Engineering, Catholic Kwandong University

[요 약]

다중화기를 이용한 논리설계는 구조적 디지털 시스템 설계에서 설계의 편리성과 유연성을 위한 유용한 방법으로 사용되어 왔다. 본 논문에서는 다중화기(multiplexer : MUX) 기반 논리설계에서 종래의 연구들에서 세밀히 다루지 않았던 무정의 조건(don't care condition)이 논리최적화에 미치는 영향을 분석해보고 단일 다중화기 기반의 설계와 복수 다중화기 기반의 설계를 위한 무정의 조건의 활용방법을 제시한다. 특히 데이터 입력의 개수가 2^m 개보다 적은 경우(선택선의 개수는 m 개일 때)의 설계 방법을 고찰한다. 제시하는 기법을 디지털논리설계 교육과 관련하여 최근 창의적 공학교육에서 크게 활용되고 있는 아두이노(Arduino)를 이용한 마이크로프로세서 설계와 연계하여 활용하는 방법에 대해서도 기술한다.

[Abstract]

Logic design using multiplexer has been used as a useful method for design convenience and flexibility in structural digital system design. In this paper, we analyze the effect of don't care conditions on logic optimization in a multiplexer-based logic design, which was not discussed enough in the previous studies in multiplexer based logic design, and describe the use of don't care conditions for designing of a single multiplexer and multiple multiplexer-based logic design. Especially, the design method when the number of data input is not 2^m (as the number of selection lines is m) is considered. We also describe how to apply the proposed technique to the digital logic design education in conjunction with microprocessor design using Arduino which is widely used in creative engineering education recently.

색인어 : 다중화기 기반 논리설계, 무정의 조건, 아두이노 기반 설계, 논리최적화, 창의적 공학교육

Key word : Multiplexer-based Logic design, Don't Care Conditions, Arduino-based Design, Logic Optimization, Creative Engineering Education

<http://dx.doi.org/10.9728/dcs.2017.18.5.881>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 21 August 2017; Revised 29 August 2017

Accepted 31 August 2017

*Corresponding Author; Jae Min Lee

Tel: +82-33-649-7552

E-mail: leejm@cku.ac.kr

I. 서론

디지털설계와 설계교육에서 설계문제에 대한 해법을 위한 다양한 접근은 최적화문제와 시스템설계의 능력을 배양하는데 효과적이며 이런 측면에서 다양한 디지털시스템 설계에 활용되고 있는[1-9] 다중화기를 이용한 설계문제는 매우 유용한 주제가 될 수 있다[9-11].

다중화기 기반 설계에 관한 기존 연구는 크게 두 가지로 나눌 수 있다. 즉 (1)입력변수가 m 개 일때 선택선의 개수가 $m-1$ 인 다중화기를 이용한 설계 방식과 (2)선택선의 개수가 $m-1$ 보다 작은 다중화기를 이용한 설계 방식 등이다. 즉 한 개의 다중화기를 이용한 설계의 경우에 비해 크기가 작은 다중화기를 여러 개 이용하여 설계함으로써 전체적으로는 회로의 크기를 줄이는 방법이다. 그런데 이 경우에 기존 연구는 무정의 조건을 고려하지 않고 있으며 또한 선택선의 개수가 m 개 일 때 다중화기의 데이터 입력의 개수가 2^m 인 경우만을 고려하고 있다 [9-11].

많은 기존의 설계방식들은 여전히 논리에 중복성(redundancy)을 포함하고 있어 최적화의 가능성을 내포하고 있다. 다중화기를 이용한 논리설계는 일반적으로 구현할 논리의 입력 변수가 m 개 일 때 다중화기의 선택선의 개수를 $m-1$ 개로 하는 크기의 다중화기를 사용하여 설계한다. 그런데 설계할 논리의 내용에 따라 또는 무정의 항의 존재여부에 따라 주변회로와의 연결 관계를 고려한 설계가 필요하다. 무정의항을 고려할 경우 데이터입력에 필요한 논리의 종류를 최소화할 수 있으며 선택선의 개수가 m 개 일 때 다중화기의 데이터 입력의 개수가 2^m 보다 적은 데이터 입력을 갖는 다중화기로 설계할 수도 있다.

본 논문에서는 무정의 조건과 다중화기 기반의 최적 설계 방법을 제시하고 최적화를 위한 다양한 설계방법을 창의적 공학교육에 매우 유용한 아두이노를 이용한 설계 교육에 활용하는 방안에 대해 기술한다.

II. 다중화기 기반 설계에서의 무정의 조건

그림 1은 무정의 조건을 고려하지 않고(무정의항을 갖고 있지 않는 논리) 다중화기를 이용한 일반 논리 설계방식을 나타낸 것이다.

표준 집적회로인 다중화기는 구현 대상 논리함수의 입력 개수가 정해지면 그 크기가 일정하게 정해진다. 설계 대상 논리는 종종 무정의항을 포함하게 된다. 무정의 항은 회로에 고장(physical failure)이 발생하지 않을 경우 유용한 논리가 아니다. 하지만 유효한 논리항과 결합하여 논리의 크기를 줄이거나 회로 입력의 구성조건을 유리하게 만들어 줄 수 있다. 표준화 집적회로로 제조되는 다중화기는 크기에 따라 내부 논리가 결정되어 있어 설계 대상 논리의 최적화과정이 무의미하게 인식된다. 하지만 설계 대상 논리의 적절한 형태로 구성하고 무정의항을 활용하면 다중화기 외부의 연결회로의 설계 및 포트구

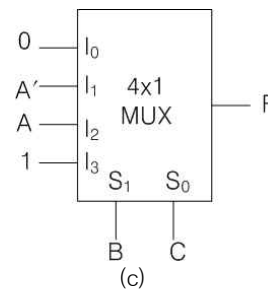
성을 효과적으로 할 수도 있고 크기가 작은 다중화기를 조합하여 전체 회로 크기 면에서 보다 효과적인 설계를 할 수도 있다.

Inputs			Output
A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

(a)

BC→	00	01	10	11
A'(0)	0	①	2	③
A(1)	4	5	⑥	⑦
MUX data inputs	0	A'	A	1

(b)



(c)

그림 1. 무정의 조건을 고려하지 않은 기존의 다중화기 기반의 논리 설계 (a)진리표 (b)구현표 (c) Mux로 설계한 회로

Fig. 1. Conventional Mux-based design of logic without don't care condition (a)Truth table (b)Implementation table (c)Mux-based design

그림 2는 무정의항을 갖는 논리를 다중화기로 설계한 것이다. 회로의 논리함수가 3개의 입력을 가지므로 필요로 하는 다중화기의 크기는 4×1 이다. 이 회로를 4×1 다중화기로 설계하면 (c)와 같은 회로가 된다.

구현표 (b)에서 알 수 있듯이 무정의항 $AB'C'$ 와 ABC' 를 고려하지 않을 경우 다중화기의 데이터 입력 I_0, I_1, I_2, I_3 에는 0, A', A, 1이 인가되고 무정의항을 고려하면 데이터 입력 I_0, I_1, I_2, I_3 에 0, 1, 1, 1이 인가될 수 있어 시스템 구성 시 주변 회로 모듈과의 연결을 고려할 때 최적의 선택이 가능하게 된다.

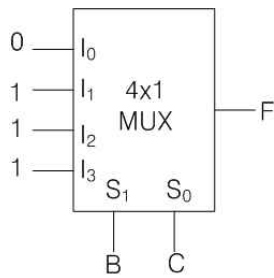
Inputs			Output
A	B	C	F
0	0	0	0
0	0	1	1
0	1	1	1
1	0	0	0
1	1	0	1
1	1	1	1

(a)

		BC→			
		00	01	10	11
A'(0)		0	①	②d	③
A(1)		0	⑤d	⑥	⑦
MUX data inputs	Not using don't care terms	0	A'	A	1
	Using don't care terms	0	1	1	1

(②d, ⑤d : don't care terms)

(b)



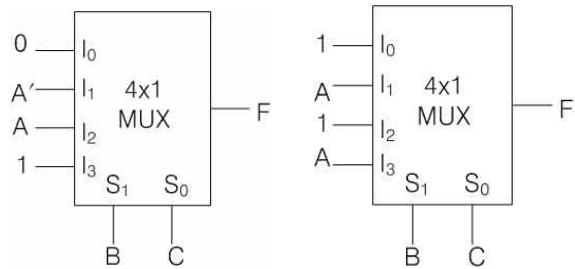
(c)

그림 2. 무정의 조건이 있는 논리의 다중화기 기반 설계 (a)진리표 (b) 구현표 (c)무정의 조건을 고려한 설계 **Fig. 2.** Mux-based design of logic with don't care conditions (a)Truth table (b)Implementation table (c)Mux-based design without don't care condition (d)Mux-based design with don't care condition

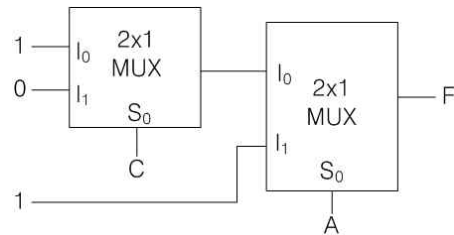
다중화기를 사용하는 설계에서 선택선의 개수를 m개일 때 일반적으로 $2^m \times 1$ 크기의 다중화기를 사용한다. 반면에 설계 대상논리를 다중화기로 구현할 때 입력의 개수가 2^m 이 아닌 경우는 일반적인 구현표(implementation table)로는 설계가 어렵다[12]. 그림 3은 무정의항을 갖는 논리함수 $F(a,b,c)=\Sigma(0,2,5,7)+\Sigma d(4,6)$ 을 4x1 MUX 및 2x1 MUX 등으로 설계한 것이다.

		BC→			
		00	01	10	11
A'(0)		①	1	②	3
A(1)		④d	⑤	⑥d	⑦
MUX data inputs	Not using don't care terms	A'	A	A'	A
	Using don't care terms	1	A	1	A

(a)



(b)



(c)

그림 3. 무정의 항을 갖는 논리의 다중화기로의 설계 (a)일반 4x1 Mux 기반 설계 (b)무정의항을 고려한 4x1 Mux 기반 설계 (c)무정의항을 고려한 2x1 Mux 기반 설계 **Fig. 3.** A Mux-based logic design with don't care terms (a)General 4x1 Mux-based design (b)4x1 Mux-based design using don't care terms (c)2x1 Mux-based design using don't care terms

무정의항을 이용하면, 다중화기의 데이터 입력으로 무정의항을 고려하지 않고 구한 데이터 입력값 A', A, A', A 대신에 1, A, 1, A를 인가할 수 있어 논리 설계 시 일반 입력 변수의 수를 줄일 수 있으며 주변회로와의 연결관계에서 보다 다양한 선택이 가능하다.

III. 무정의 조건을 고려한 새로운 다중화기 기반 논리 설계

다중화기 기반의 논리설계에서는 구현 대상 논리함수의 입력 개수가 정해지면 그 크기가 일정한 다중화기를 사용하여 설계하는 것이 일반적이다. 즉 n개의 입력변수를 갖는 논리회

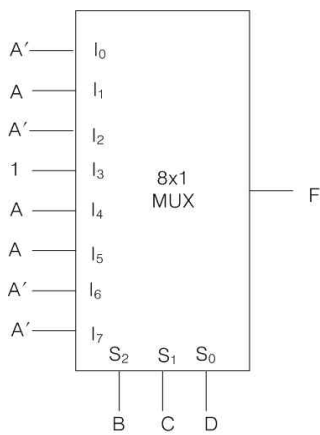
로를 다중화기로 설계하기 위해서 선택선의 개수가 n-1개이고 데이터 입력의 개수가 2ⁿ⁻¹개인 다중화기를 선택한다. 예를 들어 다중화기 기반의 논리설계 시 무정의항의 사용은 몇 가지 경우에 논리간소화의 효과를 가져 온다. 예를 들어 식 (1)의 논리함수를 다중화기를 사용하여 일반적인 방법으로 설계하려면 입력변수의 개수가 4개이므로 선택선의 개수는 3개 이어야 하고 따라서 데이터입력의 개수는 8(=2³)개이므로 결과적으로 8x1 다중화기를 필요로 한다.

$$F(A, B, C, D) = \sum(0, 2, 3, 6, 7, 9, 11, 13, 14) + \sum d(1, 4, 5) \quad (1)$$

그림 4는 식 (1)로 나타낸 논리함수를 8x1다중화기로 설계한 것이다.

BCD→	000	001	010	011	100	101	110	111
A'(0)	⓪	1	②	③	4	5	⑥	⑦
A(1)	8	⑨	10	⑪	⑫	⑬	14	15
	A'	A	A'	1	A	A	A'	A'

(a)



(b)

그림 4. $F(A, B, C, D) = \sum(0, 2, 3, 6, 7, 9, 11, 13, 14) + \sum d(1, 4, 5)$ 의 다중화기 설계 (a)구현표 b)8x1다중화기로 설계한 회로

Fig. 4. Mux-based design of $F(A, B, C, D) = \sum(0, 2, 3, 6, 7, 9, 11, 13, 14) + \sum d(1, 4, 5)$ (a)Implementation table (b)8x1 Mux-based logic circuit

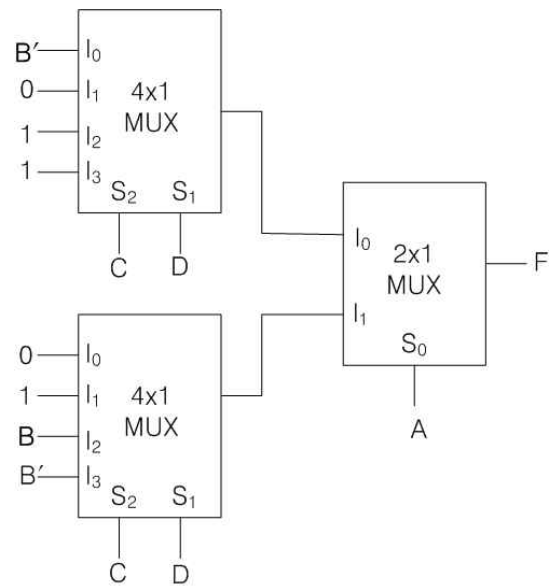
그런데 이 논리를 8x1 다중화기 대신 4x1 다중화기와 2x1 다중화기로 설계할 경우 회로의 크기를 줄일 수 있다. 그

림 5는 식 (1)의 논리함수를 8x1 다중화기 대신 그 크기가 4x1과 2x1인 다중화기로 설계한 것이다.

데이터 입력의 논리값을 구할 때 무정의항을 이용하면 A의 입력값이 0일 때 모든 출력이 1이 되므로 8x1 다중화기를 사용하는 대신 4x1 다중화기와 2x1 다중화기로 설계가 가능하게 된다.

BCD→	000	001	010	011	100	101	110	111
A'(0)	⓪	①d	②	③	④d	⑤d	⑥	⑦
A(1)	8	⑨	10	⑪	⑫	⑬	14	15
	A'	A(1)	A'	1	A(1)	A(1)	A'	A'

(a)



(b)

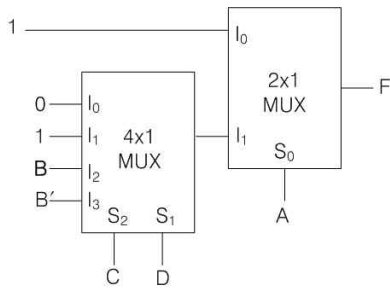
그림 5. $F(A, B, C, D) = \sum(0, 2, 3, 6, 7, 9, 11, 13, 14) + \sum d(1, 4, 5)$ 의 다중화기 설계 (a)구현표 (b)4x1 다중화기 및 2x1다중화기로 설계한 회로

Fig. 5. Mux-based Design of $F(A, B, C, D) = \sum(0, 2, 3, 6, 7, 9, 11, 13, 14) + \sum d(1, 4, 5)$ (a)Implementation table (b)Logic circuit by 4x1 and 2x1 Muxs

그림 6은 식 (1)의 논리함수를 4x1 다중화기와 2x1인 다중화기로 설계한 또 다른 방식이다. 2x1 선택선의 입력 A에 0이 인가될 때 BCD입력의 모든 논리값에 대해 데이터 입력 I0에 1이 입력된다.

BCD→	000	001	010	011	100	101	110	111
A'(0)	①	①d	②	③	④d	⑤d	⑥	⑦
A(1)	8	⑨	10	⑪	⑫	⑬	14	15
	A'	A(1)	A'	1	A(1)	A(1)	A'	A'

(a)

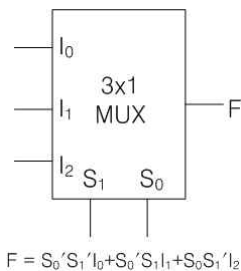


(b)

그림 6. $F(A, B, C, D) = \sum(0, 2, 3, 6, 7, 9, 11, 13, 14) + \sum d(1, 4, 5)$ 의 무정의항을 이용한 다중화기 설계
(a) 구현표 (b) 4x1 다중화기 및 2x1 다중화기로 설계한 회로

Fig. 6. Mux-based Design of $F(A, B, C, D) = \sum(0, 2, 3, 6, 7, 9, 11, 13, 14) + \sum d(1, 4, 5)$ with don't care terms
(a) Implementation table (b) Logic circuit by 4x1 and 2x1 Muxs with don't care terms

다중화기를 이용한 논리설계 시 3x1 다중화기가 사용될 수 있다. 3x1 다중화기는 그림 7과 같이 설계하여 표준화(IC) 하여 사용하는 것도 가능할 것이다.



$$F = S_0' S_1' I_0 + S_0' S_1 I_1 + S_0 S_1 I_2$$

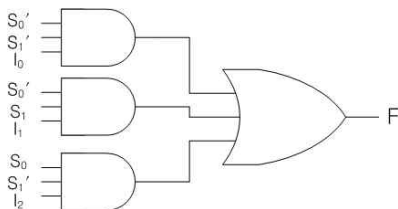


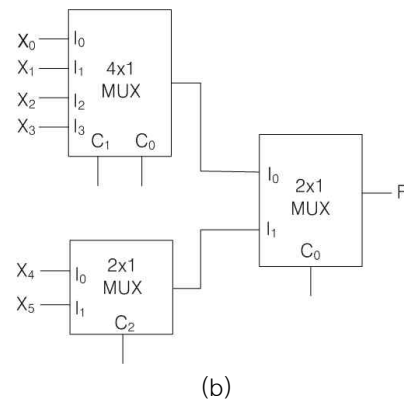
그림 7. 3x1 다중화기의 설계
Fig. 7. Design of 3x1 multiplexer

또한 무정의항이 있는 논리설계를 위해 다음 예와 같은 논리의 경우 1개의 4x1 다중화기와 2개의 2x1 다중화기를 사용하여 6x1 다중화기의 기능으로 그림 8과 같이 설계하여 사용할 수 있다.

Selection Inputs			Output
C2	C1	C0	Y
0	0	0	X0
0	0	1	X1
0	1	0	X2
0	1	1	X3
1	0	0	X4
1	0	1	X5
1	1	0	D
1	1	1	D

(D: Don't care term)

(a)



(b)

그림 8. 6 데이터 입력 회로의 다중화기 기반 설계
(a) 진리표 (b) 논리도

Fig. 8. Mux-based design of six data inputs logic circuit
(a) Truth table (b) Logic circuit

IV. 아두이노를 이용한 설계 검증 및 실험

다중화기를 이용한 논리설계에서 회로 최적화를 고려하는 설계는 설계교육에서도 핵심주제가 된다. 설계한 회로를 검증하기 위해서는 PLD 등으로 회로를 구현하고 여기에 적절한 입력을 인가하여 출력을 관찰하여 회로 동작을 검증하는 과정이 필요하다. 그림 9는 최근에 개발되어 주목받고 있는, 마이크로프로세서의 구조와 기능을 쉽게 이해하고 활용하도록 개발된 아두이노[13]를 사용하여 다중화기 설계회로의 동작을 검증하는 시스템 즉 다중화기 기반 피테스트 논리회로(Circuit under Test : CUT), 논리검증용 패턴생성기 및 출력 비교기 회로(Verification Circuits)를 나타낸 것이다.

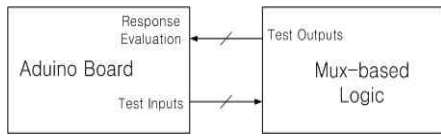


그림 9. 아두이노를 사용한 회로 검증 시스템
Fig. 9. Verification system for Mux-based Logic using Arduino

아두이노는 주변장치와의 연결을 위한 인터페이스를 갖는 소형 마이크로프로세서로서 캡스턴설계 등에서는 이를 중앙 제어장치로 하여 센서, 모터, 통신모듈 등 주변장치들을 제어하는 목적으로 주로 사용한다. 여기서는 설계교육에서 아두이노의 활용능력을 향상시키기 위한 목적으로 아두이노를 논리검증을 위한 시스템 구성에 활용하는 방법을 제시하고자 한다.

다중화기 기반으로 설계된 논리회로를 검증하기 위한 알고리즘은 그림 10과 같다.

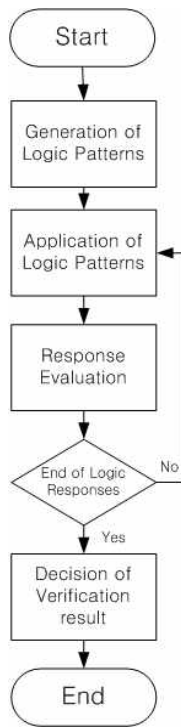


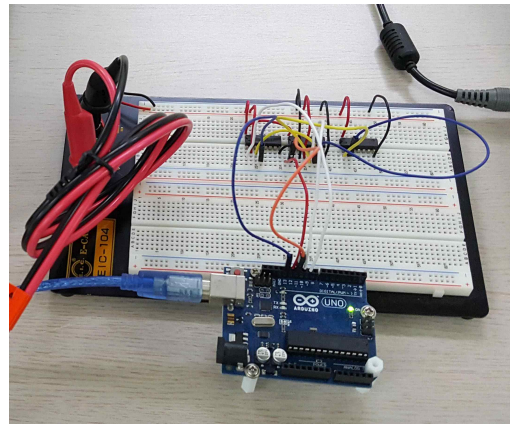
그림 10. 아두이노를 사용한 회로 검증 시스템의 알고리즘
Fig. 10. An algorithm of logic verification system using Arduino

다중화기를 사용하여 설계한 회로의 논리 검증을 위해서는 먼저 입력값(입력 패턴)을 생성한 다음 생성된 모든 패턴을 회로의 주 입력에 인가하여 주 출력으로 출력된 값을 진리표의 논리값과 비교하여 정상적인 응답인지의 여부를 판단한

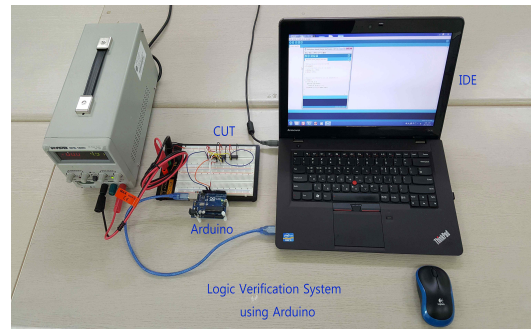
다.

그림 11은 아두이노 통합개발환경 (IDE, Integrated Development Environment)에서 프로그래밍 하여 다중화기 기반의 논리회로를 실험하기 위한 검증시스템을 구성한 것이다. 여러 가지 예제를 실험한 결과 각각의 경우 진리표와 동일한 실험(시뮬레이션) 결과를 얻게 됨을 확인할 수 있었다.

학습자들은 이러한 주제를 통해서 다중화기 기반의 설계 기법과 논리최적화 및 아두이노 응용 등 다양한 학습효과를 얻을 수 있게 된다.



(a)



(b)

그림 11. 아두이노를 사용한 다중화기 기반 설계 회로 검증 실험
 (a)다중화기 기반 실험 회로(4x1 MUX과 2x1 MUX를 사용한 회로) (b)아두이노를 이용한 논리 검증 시스템

Fig. 11. Experimental results using Arduino for logic verification of MUX-based logic circuits. (a)Mux-based experimental circuit(Using 4x1 MUX and 2x1 MUX) (b)Logic verification system using Arduino

V. 결 론

본 논문에서는 무정의 조건과 다중화기 기반의 설계 최적화 기법에 대해 기술하고 최적화를 위한 다양한 설계 방법

과 이를 아두이노와 관련하여 설계 교육에 활용하는 방안을 제시하였다. 다중화기 기반의 논리설계 시 무정의항을 고려하여 주변회로와의 연결부에서 회로최적화가 가능함과 선택선의 개수 m 개 일 때 무정의항을 고려하여 $2^m \times 1$ 보다 작은 크기의 다중화기로 설계 하는 방법과 선택선의 개수가 m 개일 때 $2^m \times 1$ 보다 작은 크기의 데이터 입력을 갖는 다중화기로 설계하는 방법 등을 제시하였다. 표 1은 제안하는 방식과 전통적인 논리설계 방식을 비교하여 나타낸 것이다.

표 1. 제안하는 방식과 기존 방식의 비교

Table 1. Comparison of proposed design with conventional design

Logic circuits to be compared	Number of gates for conventional design without don't care terms	Number of gates for proposed design with don't care terms
3 x 1 Mux-based Design	3-inputs AND : 4EA, 4-input OR: 1EA	3-inputs AND : 3EA, 3-input OR: 1EA
4 x 1 Mux-based Design	3-inputs AND : 4EA, 4-input OR: 1EA	2-inputs AND : 4EA, 2-input OR: 2EA
5 x 1 Mux-based Design	4-inputs AND : 5EA, 5-input OR: 1EA	3-inputs AND : 4EA, 4-input OR: 1EA, 2-inputs AND : 2EA, 2-input OR: 1EA
8 x 1 Mux-based Design	4-inputs AND : 8EA, 8-input OR: 1EA	3-inputs AND : 4EA, 4-input OR: 1EA, 2-inputs AND : 4EA, 2-input OR: 2EA

표 1에서 알 수 있듯이 설계 대상회로의 논리가 무정의항을 가지고 있고 일정한 논리 조건을 만족할 때 전통적인 다중화기 기반 논리설계 방식에 비하여 다중화기의 크기를 줄일 수 있다. 특히 선택선의 개수 m 개 일 때 무정의항을 고려하면 $2^m \times 1$ 보다 작은 크기의 다중화기로 설계 하는 것 또한 가능하다.

또 다른 내용으로서 최근 공학교육에서 강조되고 있는 창의적 설계 즉 캡스톤 설계와 관련하여 아두이노 마이크로프로세서 보드의 활용을 위한 주제로서 제시하는 다중화기를 이용한 논리설계와 접목하여 논리 검증을 위한 검증데이터(벡터) 생성과 응답 비교를 위한 검증시스템 구현하는 방안도 제시하였다. 이와 같은 제안 내용을 통하여 (1)다중화기 기반의 무정의 조건을 고려한 설계능력 배양, (2)아두이노 프로그래밍 기법과 활용설계능력 배양 및 (3)종합설계 주제로서 디지털 설계 다양성과 창의성 배양의 효과를 얻을 수 있을 것이다.

참고문헌

- [1] Y. Jiang, A. et-al, "A Novel Multiplexer-Based Low-Power Full Adder," *IEEE Trans on Circuits and Systems*, Vol. 51, No. 7, pp. 345-348, July 2004.
- [2] Hyun-Sang Park, "Approximate Computing Based 24×8 Multiplier Architecture for Image Processing Algorithms," *Journal of Korean Institute of Information Technology* Vol. 15, No. 1, pp. 123-129, Jan. 2017.
- [3] Seo, Soo-Duk, Cho, Hak-Rae, Yang, Doo-Yeong, "Triple-band Multiplexer for a Low Power Portable Base Station," *Korea Academy Industrial Cooperation Society*, Vol. 51, No. 12, pp. 7309-7316. 2014.
- [4] Su-Nam Jung, Jeong-Beom Kim, "Design of a Low-Power 4-2 Compressor for Low-Power Multiplication," *Journal of Korean Institute of Information Technology*, Vol. 15, No 4, April 2017.
- [5] Haksun Kim, Byungha Choi, Hyungjae Lee, "Design of the High-Speed 4:1 Multiplexer," in *Proceedings of IEEK*, pp. 3-7, Jan. 1989.
- [6] Kyosun Kim, "FPGA Mapping Incorporated with Multiplexer Tree Synthesis," *Journal of the Institute of Electronics and Information Engineers*, Vol. 53, No. 4, pp. 37-47, April 2016.
- [7] Jong-hak Hwang, Seung-Young Park, Boo-Sik Shin, Heung-Soo Kim, "Multiplexer-Based Array Multipliers over GF (2^m)," *The Institute of Electronics Engineers of Korea - System and Control*, Vol. 37, No. 4, pp. 35-41, July 2000.
- [8] Jae-hyung Park, Dong-young Kim, Dong-soo Lee, Kang-Yoon Lee, "Design of Low Power Touch Sensor Analog Front-End using Multiplexer," in *Proceedings of IEEK*, pp. 661-662, Nov. 2013.
- [9] Jae-Min Lee, "An Approach to Implementation of Creative Design-Oriented Curriculum and Engineering Design Complex," *Society of Digital Contents*, Vol. 16, No. 3, pp. 53-61, Feb. 2015.
- [10] Jae-Min Lee, *Creative Digital System Design*, 1st ed. Cheongsong Publishing Co, 2016.
- [11] M. Mano and M. Ciletti, *Digital Design*, prentice Hall, 2006.
- [12] Charles C. Lin :Case Study: Implementing a 5-1 MUX, *Class Notes*, Univ. of Maryland, 2003.
- [13] <https://www.arduino.cc/>



이재민(Jae-Min Lee)

1979년 : 한양대학교(공학사)

1981년 : 한양대학교 대학원(공학석사)

1987년 : 한양대학교 대학원(공학박사)

1990년~1991년 : 일리노이대학(Urbana-Champaign) Post-Doc. (한국연구재단)

1992년~1994년 : 관동대학교 전자계산소 소장

1994년~1996년 : 대한전자공학회 강원지부장

2001년~2002년 : 뉴욕주립대(Buffalo) 연구교수

2011년~2013년 : 가톨릭관동대학교 공과대학 학장, 방재대학원장

2010년~현재 : 한국정보전자통신기술학회 부회장

관심분야 : SoC 및 컴퓨터회로 설계, 태양광발전시스템, LED 감성조명, 공학교육콘텐츠 설계