

Professor Dr.-Ing. Stefan Kowalewski
Dipl.-Inform. Dominik Franke
Dipl.-Inform. Sebastian Biallas

Aachen, 23. Dezember 2010
SWS: V4/Ü2, ECTS: 7

Einführung in die Technische Informatik

WS 2010/2011

Blatt 10: Musterlösung

ACHTUNG: Die Musterlösung ist ein zusätzliches Serviceangebot. Sie erhebt weder Anspruch auf Vollständigkeit noch auf Korrektheit.

Aufgabe 1: (★) Realisierung Boolescher Funktionen mittels komplexer Bauteile

Aufgabe 2: Kostenabschätzung

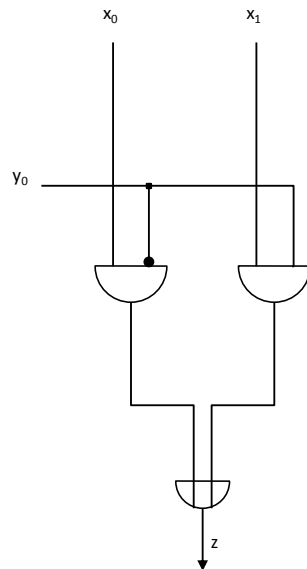
- a) Zeigen Sie durch Induktion, dass sich jeder $N \times M$ -MUX durch $3(2^d - 1)$ logische Gatter (zählen Sie nur UND- und ODER-Gatter, keine Negation) rekursiv konstruieren lässt. Alle Gatter haben einen Fan-In von 2. Dabei bezeichnet d die Anzahl der Steuersignale¹.
- b) Vergleichen Sie die rekursive Konstruktion aus a) mit der einfachen Konstruktion, wie sie in der Vorlesung in Kapitel 11 beschrieben wird, anhand der Anzahl verwendeter Gatter exemplarisch für den 4×16 -MUX. Leiten Sie dazu eine Formel zur Berechnung der Anzahl Gatter bzgl. der einfachen Konstruktion her, indem Sie die Realisierung eines 2-MUX (Kap. 11.1 der Vorlesung - Folie *Realisierung eines 2-MUX*) so anpassen, dass alle Gatter nur noch einen Fan-In von 2 besitzen. Es ist hilfreich die Anpassung als Zwischenschritt auch für den 3-MUX vorzunehmen.

Lösungsvorschlag

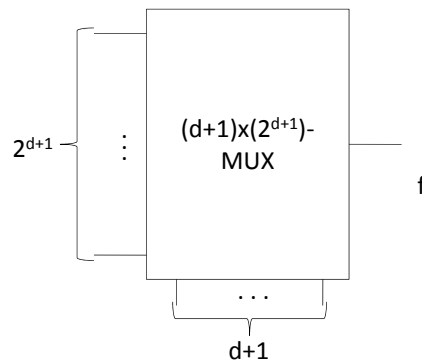
- a) Beweis durch vollständige Induktion über d :

IA $d = 1$. Ein 1×2 -MUX lässt sich durch 3 Gatter realisieren:

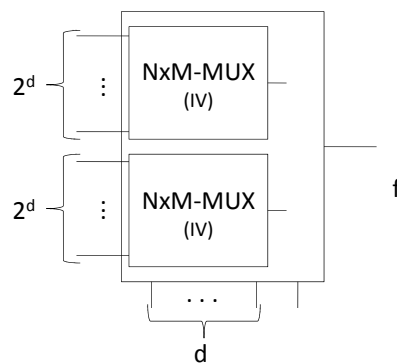
¹Die Bezeichnung d -MUX ist eine Alternative zu $N \times M$ -MUX. Mit dieser Bezeichnung entspricht ein z. B. ein 1-MUX einem 1×2 -MUX, da die Anzahl der Steuersignale (d) 1 ist.



- IV Sei die Behauptung für ein $d \in \mathbb{N}$ wahr. D. h. ein $N \times M$ -MUX sei durch $3 \cdot (2^d - 1)$ Gatter realisierbar.
- IS $d \mapsto d + 1$. Darzustellen ist ein MUX mit $d + 1$ Steuer- und 2^{d+1} Dateneingängen



Dann gibt es nach IV einen $N \times M$ -MUX, der 2^d Eingänge mit d Steuerleitungen verschaltet.



Es verbleiben 1 Steuereingang und die beiden Ausgänge der $N \times M$ -MUXe. Diese werden mittels eines 1×2 -MUXes verschaltet.
Für einen $(d + 1) \times (2^{d+1})$ -MUX gilt also:

$$\begin{aligned}
\underbrace{2}_{2\text{NxM-MUXE}} \cdot \underbrace{(3(2^d - 1))}_{\# \text{ Gatter nach IV}} + \underbrace{3}_{3 \text{ Gatter für 1x2-MUX (vgl. IA)}} &= 2 \cdot (3 \cdot 2^d - 3) + 3 \\
&= 3 \cdot 2^{d+1} - 6 + 3 \\
&= 3 \cdot 2^{d+1} - 3 \\
&= 3 \cdot (2^{d+1} - 1)
\end{aligned}$$

\Rightarrow die Behauptung nach dem Prinzip der vollständigen Induktion für alle $d \in \mathbb{N}$.

- b) Für die rekursive Konstruktion benötigt man $3(2^4 - 1)$ Gatter, das sind 45. Eine einfache Konstruktion benötigt hier $2^d \cdot (d + 1) - 1$ Gatter, also 79. Für $d = 8$ liegen die Gatterzahlen bereits bei 765 im Vergleich zu 2303. Die einfache Konstruktion benötigt also mehr Gatter als eine rekursive Lösung.

Begründung: Ein MUX nach einfacher Konstruktion gliedert sich in UND- und ODER-Ebene. Um ein Gatter mit einem Fan-In von n ($n > 2$) durch Gatter mit einem Fan-In von genau 2 darzustellen, benötigt man $n - 1$ Gatter. Für einen MUX mit d Steuer- und 2^d Dateneingängen gilt dann: Jedes UND-Gatter hat 1 Daten- und d Steuereingänge, insgesamt also einen Fan-In von $d + 1$. Für die Darstellung jedes UND-Gatters werden d Gatter mit Fan-In 2 benötigt. In der UND-Ebene gibt es 2^d Gatter, die auf diese Art dargestellt werden müssen, insgesamt also $d \cdot 2^d$.

Für die ODER-Ebene hat man ein Gatter, dessen Fan-In gleich der Anzahl an UND-Gattern ist. Stellt man das ODER-Gatter nur durch Gatter mit Fan-In 2 dar, erhält man $2^d - 1$.

Insgesamt erhält man, dass für die einfache Konstruktion $d \cdot 2^d + 2^d - 1$ Gatter benötigt werden.

Aufgabe 3: (★) Darstellung Boolescher Funktionen mittels komplexer Bausteine

Aufgabe 4: Realisierung Boolescher Funktionen mittels komplexer Bauteile

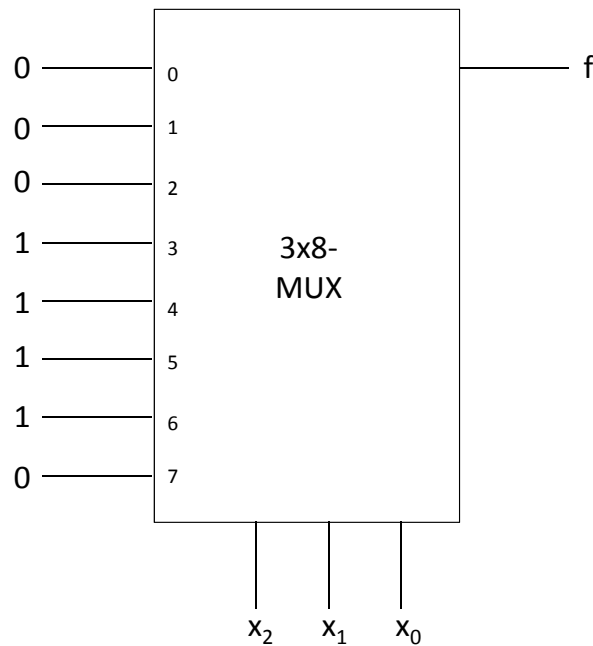
- a) Sei die Funktion $f : \mathbb{B}^3 \rightarrow \mathbb{B}$ durch folgende Wertetabelle definiert. Entwerfen Sie einen Multiplexer mit der in der Vorlesung vorgestellten Hardware-Lookup Methode. Dieser Multiplexer soll die Funktion f realisieren.

x_2	x_1	x_0	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

- b) Welche Vor- und Nachteile hat diese Methode?

Lösungsvorschlag

a) Eine mögliche Realisierung ist folgende:



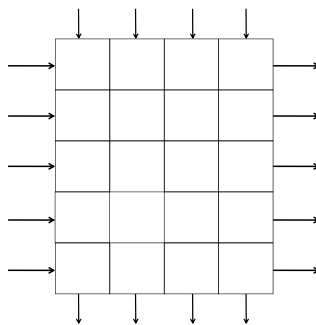
b) Hardware Lookup ist einfacher zu realisieren, benötigt aber i.d.R. mehr Hardware-Komponenten. Denn ein MUX ist ein komplexer Baustein und benötigt, wie in Aufgabe 2 gezeigt, mehr Gatter.

Aufgabe 5: Realisierung Boolescher Funktionen mittels PLAs

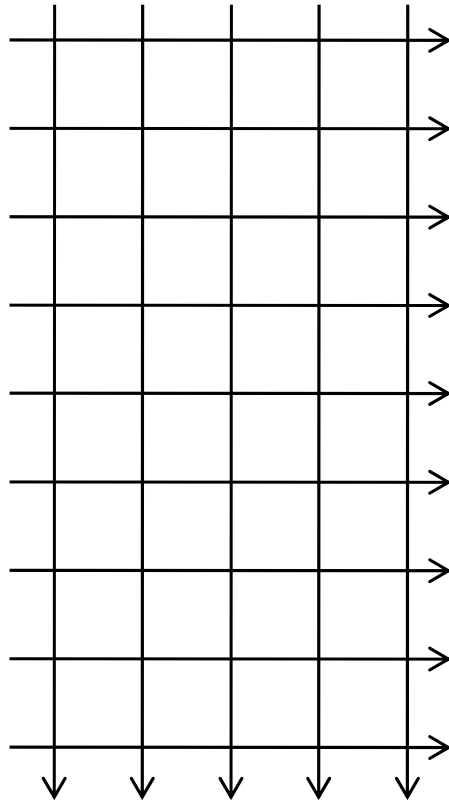
Gegeben sei die Funktion $f : \mathbb{B}^3 \rightarrow \mathbb{B}^2$ mit

$$f(x, y, z) = (\bar{x}y + x\bar{y}\bar{z} + x\bar{z}, yz + x\bar{z})$$

a) Programmieren Sie das gegebene PLA, sodass es die Funktion f realisiert. Verwenden Sie alle in der Vorlesung vorgestellten Bausteine (Identifier, Addierer, ...) und beschriften Sie **alle** Ein- und Ausgänge vollständig.

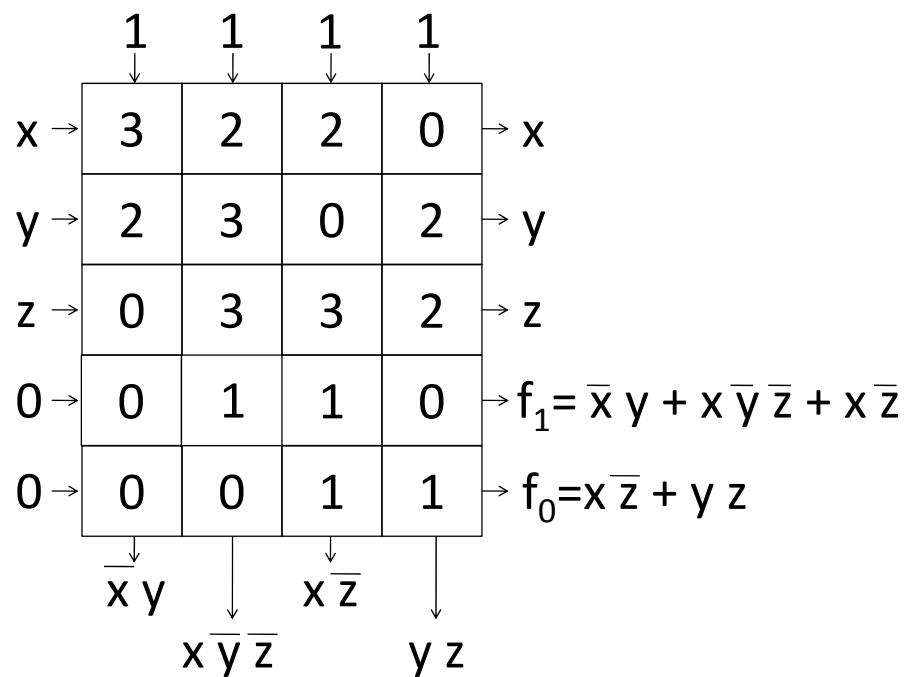


- b) Realisieren Sie die Funktion jetzt in punktorientierter Schreibweise. Beschriften Sie zudem **alle** Ein- und Ausgänge.
Hinweis: Markieren Sie die Knoten erst, nachdem Sie alle Ein- und Ausgänge beschriftet haben.

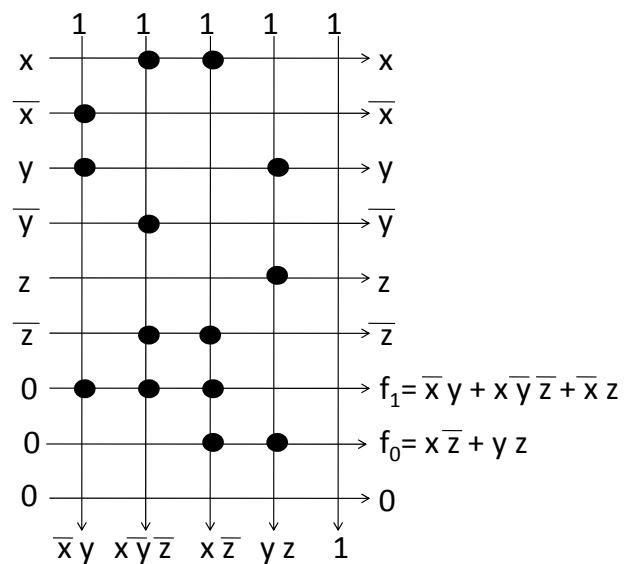


Lösungsvorschlag

- a) Ein mögliches resultierendes PLA:



b) Punktorientiert:



Aufgabe 6: (★) Programmable Logic Arrays

Gegeben sei ein Schaltnetz mit vier Eingängen x_3, x_2, x_1, x_0 und den Ausgängen z_3, z_2, z_1, z_0 ,

welches eine Schaltfunktion $f : \mathbb{B}^4 \rightarrow \mathbb{B}^4$ realisiert. Dabei gilt

$$f(x_3, x_2, x_1, x_0) = (z_3, z_2, z_1, z_0) \iff (z_3 z_2 z_1 z_0)_2 = (x_3 x_2)_2 \cdot (x_1 x_0)_2$$

(Multiplikation zweier zweistelliger Binärzahlen). Im Folgenden sei $f_i(x_3, x_2, x_1, x_0) = z_i$ für $i \in \{0, 1, 2, 3\}$.

- Realisieren Sie diese Schaltfunktion mittels eines PLAs, indem Sie die Minterme für jede Funktion f_i bestimmen und das PLA entsprechend programmieren.
- Bestimmen Sie je ein Minimalpolynom für jede Funktion f_i . Hätte Ihnen diese Optimierung beim Aufbau des PLAs zu einer Einsparung von Spalten verhelfen können? Begründen Sie Ihre Antwort.
- Gibt es ein PLA mit 7 Spalten, das die obige Schaltfunktion realisiert? Begründen Sie Ihre Antwort, eventuell mit Lösungsmatrix.

Lösungsvorschlag

- Tabelle und PLA:

Minterm-Nr.	x_3	x_2	x_1	x_0	z_3	z_2	z_1	z_0	$(x_3 x_2)_2 \cdot (x_1 x_0)_2$ dezimal	Spalte im PLA
0	0	0	0	0	0	0	0	0	$0 \cdot 0 = 0$	
1	0	0	0	1	0	0	0	0	$0 \cdot 1 = 0$	
2	0	0	1	0	0	0	0	0	$0 \cdot 2 = 0$	
3	0	0	1	1	0	0	0	0	$0 \cdot 3 = 0$	
4	0	1	0	0	0	0	0	0	$1 \cdot 0 = 0$	
5	0	1	0	1	0	0	0	1	$1 \cdot 1 = 1$	1
6	0	1	1	0	0	0	1	0	$1 \cdot 2 = 2$	2
7	0	1	1	1	0	0	1	1	$1 \cdot 3 = 3$	3
8	1	0	0	0	0	0	0	0	$2 \cdot 0 = 0$	
9	1	0	0	1	0	0	1	0	$2 \cdot 1 = 2$	4
10	1	0	1	0	0	1	0	0	$2 \cdot 2 = 4$	5
11	1	0	1	1	0	1	1	0	$2 \cdot 3 = 6$	6
12	1	1	0	0	0	0	0	0	$3 \cdot 0 = 0$	
13	1	1	0	1	0	0	1	1	$3 \cdot 1 = 3$	7
14	1	1	1	0	0	1	1	0	$3 \cdot 2 = 6$	8
15	1	1	1	1	1	0	0	1	$3 \cdot 3 = 9$	9

$x_3 \rightarrow$	3	3	3	2	2	2	2	2	
$x_2 \rightarrow$	2	2	2	3	3	3	2	2	
$x_1 \rightarrow$	3	2	2	3	2	2	3	2	
$x_0 \rightarrow$	2	3	2	2	3	2	2	3	
	0	0	0	0	0	0	0	0	$\rightarrow f_3$
	0	0	0	0	1	1	0	1	$\rightarrow f_2$
	0	1	1	1	0	1	1	1	$\rightarrow f_1$
	1	0	1	0	0	0	1	0	$\rightarrow f_0$

b) Minimierung mithilfe von Karnaugh-Diagrammen:

$x_3x_2 \backslash x_1x_0$		f_3			
		00	01	11	10
00					
01					
11				1	
10					

$x_3x_2 \backslash x_1x_0$		f_2			
		00	01	11	10
00					
01					
11					1
10				1	1

$x_3x_2 \backslash x_1x_0$		f_1			
		00	01	11	10
00					
01				1	1
11			1		1
10			1	1	

$x_3x_2 \backslash x_1x_0$		f_0			
		00	01	11	10
00					
01			1	1	
11			1	1	
10					

Die resultierenden Minimalpolynome sind:

$$f_3 = x_3 x_2 x_1 x_0$$

$$f_2 = x_3 x_1 \bar{x}_0 + x_3 \bar{x}_2 x_1$$

$$f_1 = x_3 \bar{x}_2 x_0 + x_3 \bar{x}_1 x_0 + \bar{x}_3 x_2 x_1 + x_2 x_1 \bar{x}_0$$

$$f_0 = x_2 x_0$$

Die Minimalpolynome besitzen insgesamt 8 unterschiedliche Terme. Darauf folgt direkt, dass man das PLA auch mit höchstens 8 Spalten realisieren kann.

$x_3 \rightarrow$	2	2	2	2	2	3	0	0	
$x_2 \rightarrow$	2	3	0	3	0	2	2	2	
$x_1 \rightarrow$	2	2	2	0	3	2	2	0	
$x_0 \rightarrow$	2	0	3	2	2	0	3	2	
	1	0	0	0	0	0	0	0	$\rightarrow f_3$
	0	1	1	0	0	0	0	0	$\rightarrow f_2$
	0	0	0	1	1	1	1	0	$\rightarrow f_1$
	0	0	0	0	0	0	0	1	$\rightarrow f_0$

- c) Die Funktionen f_3 und f_0 bleiben unverändert. Für ihre Darstellung werden im PLA zwei Spalten benötigt. Um insgesamt weniger als 8 Spalten zu benötigen, werden f_1 und f_2 derart vereinfacht, dass nur 5 Spalten für ihre Realisierung notwendig sind.

An der Wertetabelle aus a) kann man ablesen, dass f_2 und f_1 beide die Minterme Nummer 11 und 14 haben. Betrachte daher nur noch solche Vereinfachungsmöglichkeiten für die Funktionen, die die Minterme 11 und 14 nicht einbeziehen.

Damit verbleibt für f_2 nur noch ein Minterm, der abgedeckt werden muss, nämlich Nr. 10. Dieser Minterm wird durch eine eigene Spalte dargestellt.

Für f_1 verbleiben die Minterme 6, 7, 9, 13. Vereinfacht man die Funktion

$$f'_1(x_3, x_2, x_1, x_0) = x_3 \overline{x_2} \overline{x_1} x_0 + x_3 x_2 \overline{x_1} x_0 + \overline{x_3} x_2 x_1 \overline{x_0} + \overline{x_3} x_2 x_1 x_0,$$

die genau die noch nicht abgedeckten Minterme von f_1 umfasst, erhält man:

		$x_1 x_0$			
		00	01	11	10
$x_3 x_2$	00				
	01			1	1
	11		1		
	10		1		

$$f'_1 = x_3 \overline{x_1} x_0 + \overline{x_3} x_2 x_1$$

Jeder der vereinfachten Terme von f'_1 wird durch eine Spalte im PLA dargestellt. Insgesamt ergibt sich:

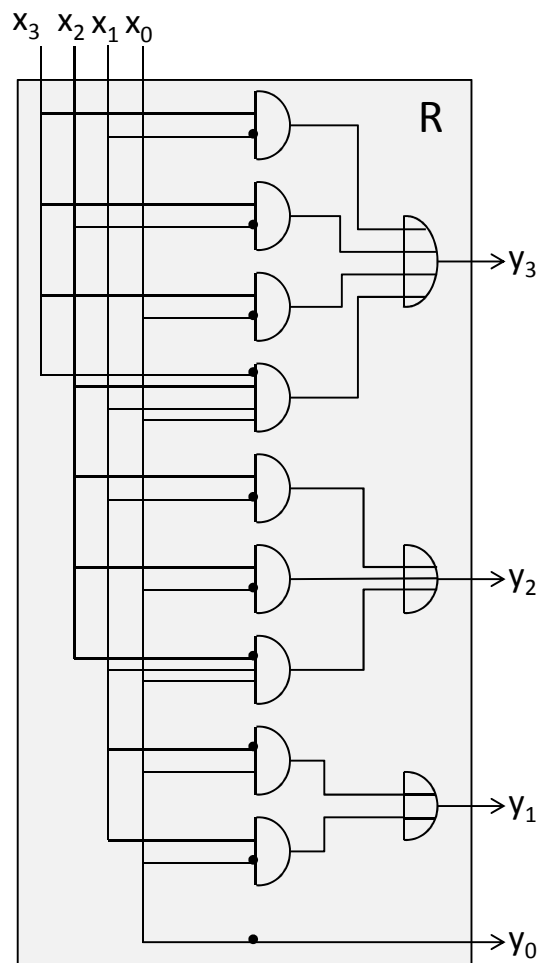
$x_3 \rightarrow$	2	2	2	2	3	2	0	
$x_2 \rightarrow$	2	3	3	0	2	2	2	
$x_1 \rightarrow$	2	2	2	3	2	2	0	
$x_0 \rightarrow$	3	3	2	2	0	2	2	
	0	0	0	0	0	1	0	$\rightarrow f_3$
	1	1	1	0	0	0	0	$\rightarrow f_2$
	1	0	1	1	1	0	0	$\rightarrow f_1$
	0	0	0	0	0	0	1	$\rightarrow f_0$

Aufgabe 7: Ringzähler

Sei die Funktion $R : \mathbb{B}^4 \rightarrow \mathbb{B}^4$ definiert durch

$$R(d(i)) = d((i + 1) \bmod 16)$$

$d(i)$ ist vierstellige Binärzahl von $i \in \{0, 1, \dots, 15\}$. R wird wie folgt realisiert:



Realisieren Sie die Schaltung als PLA.

Lösungsvorschlag

Zur Realisierung des PLA einfach die vier Funktionen ableiten

$$y_3 = x_3\overline{x_1} + x_3\overline{x_2} + x_3\overline{x_0} + \overline{x_3}x_2x_1x_0$$

$$y_2 = x_2\overline{x_1} + x_2\overline{x_0} + \overline{x_2}x_1x_0$$

$$y_1 = x_1\overline{x_0} + \overline{x_1}x_0$$

$$y_0 = \overline{x_0}$$

Daraus ergibt sich:

$x_3 \rightarrow$	0	0	0	0	0	0	2	2	2	3	
$x_2 \rightarrow$	0	0	0	2	2	3	0	3	0	2	
$x_1 \rightarrow$	0	2	3	3	0	2	3	0	0	2	
$x_0 \rightarrow$	3	3	2	0	3	2	0	0	3	2	
	0	0	0	0	0	0	1	1	1	1	$\rightarrow y_3$
	0	0	0	1	1	1	0	0	0	0	$\rightarrow y_2$
	0	1	1	0	0	0	0	0	0	0	$\rightarrow y_1$
	1	0	0	0	0	0	0	0	0	0	$\rightarrow y_0$