



UNIVERSITÀ DI PERUGIA
Dipartimento di Matematica e Informatica



ESAME DI DIDATTICA DELL'INFORMATICA
PER ARCHITETTURA DEGLI ELABORATORI

Esercitazione su circuiti combinatori

Professore

Prof. Arturo Carpi

Studenti

Bianchi Samuele

Cerami Cristian

Ronti Nicola

Sabia Maria Vitina

Anno Accademico 2023-2024

1 Prerequisiti

Per poter svolgere adeguatamente questa esercitazione è necessario possedere una conoscenza di base dell'algebra Booleana, avere familiarità con la costruzione delle mappe di Karnaugh, capacità di formalizzare problemi relativi ai circuiti combinatori e saperli rappresentare mediante diagrammi.

2 Obiettivi per l'esercitazione

L'obiettivo di questa esercitazione è consentire agli studenti di interpretare problemi reali e di estrarre gli elementi fondamentali per la loro risoluzione, utilizzando gli strumenti acquisiti nel corso di Architettura degli Elaboratori. Gli studenti saranno in grado di progettare e minimizzare circuiti logici combinatori, definendone le specifiche, implementandoli e ottimizzandoli.

3 Esercizio

1. Presso il parcheggio del servizio di continuità assistenziale (guardia medica) di Perugia sono disponibili 4 postazioni auto:

- una riservata al centralino (C)
- una riservata all'infermiere (I) di turno
- una riservata alla guardia medica (G) in servizio
- una riservata all'ambulanza (A) disponibile per questa sede

Nello stesso parcheggio sono presenti due pompe di rifornimento carburante che possono essere utilizzate secondo il seguente ordine di priorità: $A > G > I > C$.

Si chiede di progettare un sistema automatico per gestire la priorità di utilizzo delle pompe di rifornimento considerando che tutte le utilitarie potrebbero richiedere di fare rifornimento in contemporanea.

Soluzione

Specifica del comportamento

Il sistema di rifornimento è gestito tramite 4 differenti variabili "L", "M", "N" e "O", secondo la codifica riportata nella seguente tabella:

Codifica utilizzo pompa di rifornimento				
L	M	N	O	Utilizzo
0	0	0	0	Nessun utilizzo
1	0	0	0	A
0	1	0	0	G
0	0	1	0	I
0	0	0	1	C
1	1	0	0	AG
1	0	1	0	AI
1	0	0	1	AC
0	1	1	0	GI
0	1	0	1	GC
0	0	1	1	IC

Formulazione

Creazione della tavola di verità per mostrare le relazioni tra input e output.

A	G	I	C	O	N	M	L	Utilizzo
0	0	0	0	0	0	0	0	Nessuno
0	0	0	1	1	0	0	0	C
0	0	1	0	0	1	0	0	I
0	0	1	1	1	1	0	0	IC
0	1	0	0	0	0	1	0	G
0	1	0	1	1	0	1	0	GC
0	1	1	0	0	1	1	0	GI
0	1	1	1	0	1	1	0	GI
1	0	0	0	0	0	0	1	A
1	0	0	1	1	0	0	1	AC
1	0	1	0	0	1	0	1	AI
1	0	1	1	0	1	0	1	AI
1	1	0	0	0	0	1	1	AG
1	1	0	1	0	0	1	1	AG
1	1	1	0	0	0	1	1	AG
1	1	1	1	0	0	1	1	AG

Ottimizzazione

Creazione delle Mappe di Karnaugh per la semplificazione delle espressioni booleane.

1. Mappa di Karnaugh per "O".

		<i>IC</i>			
		00	01	11	10
<i>AG</i>	00	0	1	1	0
	01	0	1	0	0
	11	0	0	0	0
	10	0	1	0	0

$$O = \overline{A}IC + \overline{G}IC + \overline{A}GC$$

2. Mappa di Karnaugh per "N".

		<i>IC</i>			
		00	01	11	10
<i>AG</i>	00	0	0	1	1
	01	0	0	1	1
	11	0	0	0	0
	10	0	0	1	1

$$N = \overline{A}I + \overline{G}I$$

3. Mappa di Karnaugh per “M”.

		<i>IC</i>			
		00	01	11	10
<i>AG</i>	00	0	0	0	0
	01	1	1	1	1
	11	1	1	1	1
	10	0	0	0	0

$$M = G$$

4. Mappa di Karnaugh per “L”.

		<i>IC</i>			
		00	01	11	10
<i>AG</i>	00	0	0	0	0
	01	0	0	0	0
	11	1	1	1	1
	10	1	1	1	1

$$L = A$$

Generazione e disegno del circuito

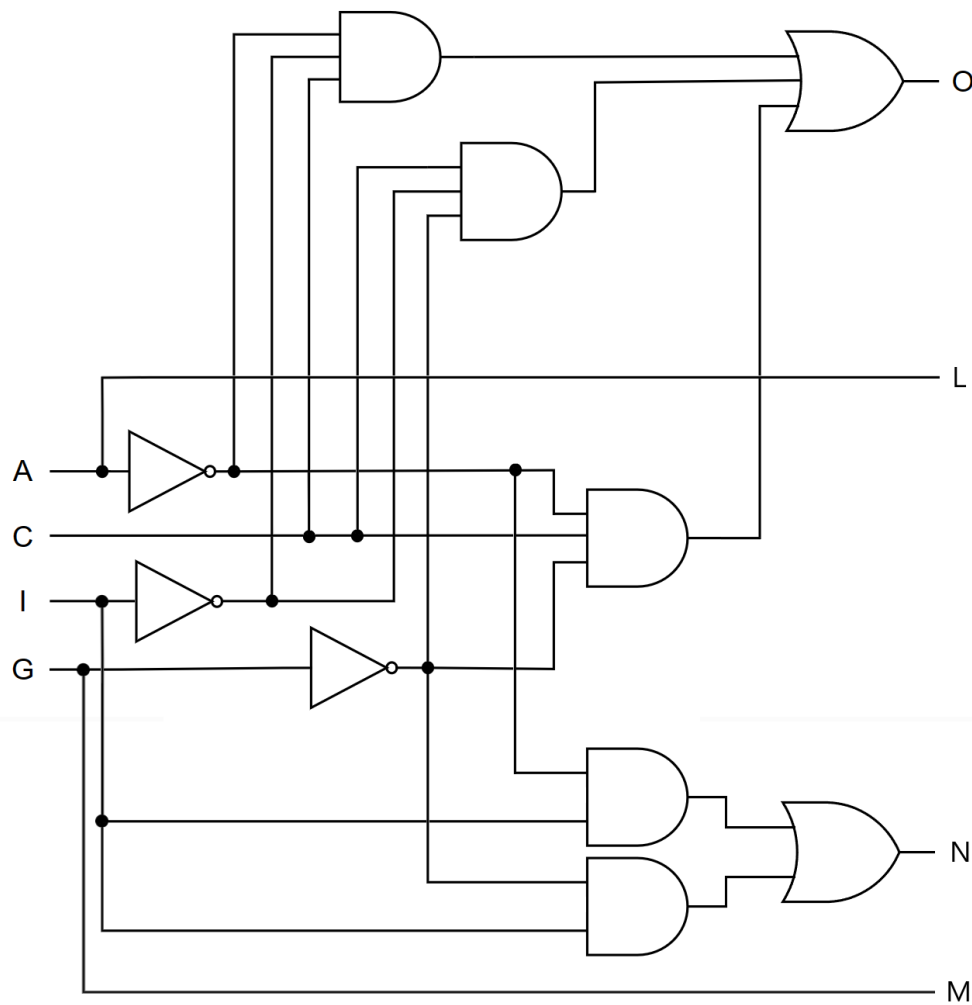
Disegno del circuito combinatorio ottenuto dalle seguenti formule:

$$O = \overline{A}IC + \overline{G}IC + \overline{A}GC$$

$$N = \overline{A}I + \overline{G}I$$

$$M = G$$

$$L = A$$



Ottimizzazione del circuito

Per ottimizzare il circuito dobbiamo andare a trovare delle equazioni booleane equivalenti che costino meno di quelle che abbiamo attualmente.

Per la variabile dipendente “O”:

$$O = \overline{A}IC + \overline{G}IC + \overline{A}GC \quad (1)$$

$$= C(\overline{A}I + \overline{G}I + \overline{A}G) \quad , \text{Raccolta la } C \quad (2)$$

$$= C((\overline{A} + \overline{I}) + (\overline{G} + \overline{I}) + (\overline{A} + \overline{G})) \quad , \text{Applicato Demorgan} \quad (3)$$

$$= C(\overline{A} + \overline{G} + \overline{I}) \quad , \text{Applicato teorema Idempotenza} \quad (4)$$

Possiamo notare come:

- Per la funzione iniziale ($O = \overline{A}IC + \overline{G}IC + \overline{A}GC$) si hanno i seguenti costi:
 - Costo letterale = 9
 - Costo ingressi = 12
- Per la funzione minimizzata ($O = C(\overline{A} + \overline{G} + \overline{I})$) si ha la riduzione dei costi ai seguenti valori:
 - Costo letterale = 4
 - Costo ingressi = 5

Per la variabile dipendente “N”:

$$N = \overline{A}I + \overline{G}I \quad (5)$$

$$= I(\overline{A} + \overline{G}) \quad , \text{Raccolta la } I \quad (6)$$

Possiamo notare come:

- Per la funzione iniziale ($N = \overline{A}I + \overline{G}I$) si hanno i seguenti costi:
 - Costo letterale = 4
 - Costo ingressi = 6
- Per la funzione minimizzata ($N = I(\overline{A} + \overline{G})$) si ha la riduzione dei costi ai seguenti valori:
 - Costo letterale = 3
 - Costo ingressi = 4

Generazione e disegno del circuito minimizzato

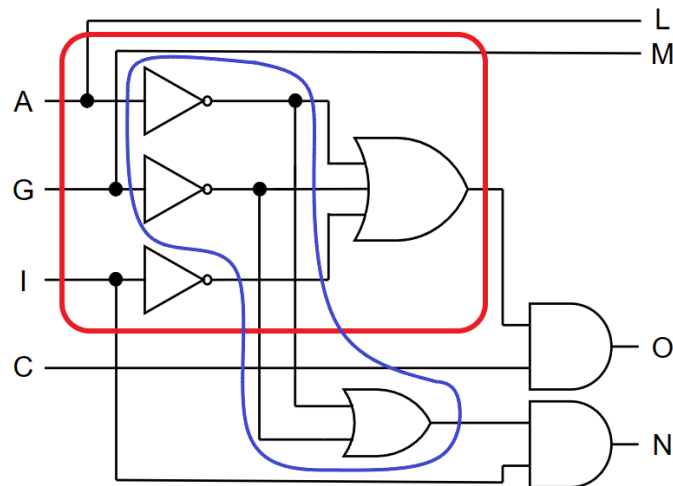
Disegniamo ora il circuito combinatorio minimizzato tramite le seguenti formule:

$$O = C(\bar{A} + \bar{G} + \bar{I})$$

$$N = I(\bar{A} + \bar{G})$$

$$M = G$$

$$L = A$$



Notiamo però come sia possibile sostituire l'insieme di porte cerchiato in rosso e blu con due porte NAND in quanto aventi il medesimo comportamento. Possiamo quindi minimizzare ulteriormente la seguente funzione ottenendo il sottostante circuito:

$$O = C(\bar{A} + \bar{G} + \bar{I}) = C(\overline{AGI}) \quad , \text{ Applicato Demorgan}$$

$$N = I(\bar{A} + \bar{G}) = I(\overline{AG}) \quad , \text{ Applicato Demorgan}$$

