

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

Calcolatori — a.a. 2012–2013

Prova in itinere del 19 novembre 2012

Cognome e Nome dello studente: _____

L'algoritmo di Euclide per il MCD

Due interi positivi a e b si dicono primi tra loro se non hanno divisori in comune, ovvero se $\text{MCD}(a, b) = 1$. Un algoritmo basato su sottrazioni ripetute per il calcolo del Massimo Comun Divisore, riportato da Euclide nei suoi “Elementi” (libro 7, proposizioni 1 e 2, circa 300 a.C.), è il seguente:

0. $\alpha \leftarrow a, \beta \leftarrow b$
1. $\alpha = \beta \Rightarrow \text{MCD}(a, b) = \alpha$; stop.
2. $\alpha < \beta \Rightarrow \alpha \leftrightarrow \beta$
3. $\gamma \leftarrow (\alpha - \beta), \alpha \leftarrow \beta, \beta \leftarrow \gamma$; vai a 1

Progettare una macchina sequenziale che calcoli $\text{MCD}(a, b)$ con a e b interi positivi minori di 512. La macchina deve segnalare separatamente in uscita se i numeri in ingresso sono primi tra loro.

- ♥ Disegnare la parte operativa della macchina;
 - ◇ Disegnare il diagramma degli stati della parte di controllo;
 - ♠ Indicare l'andamento temporale di ingressi, stati e uscite del controllo nel caso $a = 153, b = 68$;
 - ♣ Discutere la realizzazione hardware della parte di controllo.
-

Algoritmo di Euclide per il MCD:

parte operativa & parte di controllo

(cfr. compito 19/11/2012.)

a e b interi positivi.

0. $\alpha \leftarrow a, \beta \leftarrow b$

1. se $\alpha = \beta$ allora $\text{MCD}(a, b) = \alpha$; stop.

2. se $\alpha < \beta$ allora scambia α con β

3. $\gamma \leftarrow \alpha - \beta, \alpha \leftarrow \beta, \beta \leftarrow \gamma$; vai a 1.

Esempio:

$a = 153,$
 $b = 68$

α	β	γ
153	68	85
85	68	17
17	68	-51
68	17	51
51	17	34
34	17	17
17	17	0

$\downarrow$ MCD(153, 68)

La parte operativa dovrà prevedere almeno un paio di registri dati per la memorizzazione degli ingressi esterni a e b e la successiva elaborazione.

Chiamiamo questi registri α e β . Il numero

$\gamma = \alpha - \beta$ sarà prodotto da una ALU in modalità sottrazione. La stessa ALU fornirà i bit di condizione

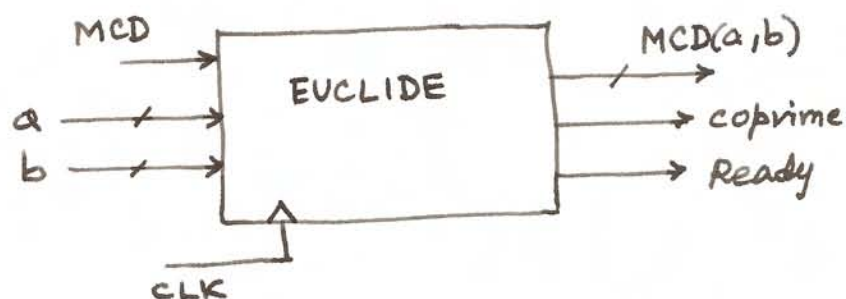
(normalmente chiamati "flags") che informeranno la parte di controllo sulla relazione tra α e β ad ogni passo dell'algoritmo. In particolare, calcolato $\gamma = \alpha - \beta$, si avrà:

1) $\alpha = \beta \iff \gamma = 0$, nel quale caso il bit ZF (= zero flag) sarà asserted

2) $\alpha < \beta \iff \gamma < 0$, nel quale caso il bit SF (= sign flag) sarà asserted

Notiamo come i passi 1 e 2 dell'algoritmo non debbano necessariamente corrispondere ad altrettanti stati della macchina di controllo. Infatti in tali passi si campioneranno gli ingressi di condizione (ZF e SF rispettivamente) provenienti dalla parte operativa, saltando all'opportuno stato futuro di controllo (abbiamo eliminato i salti pilotati da ingressi esterni "salti condizionati").

La macchina all'esterno appare così:

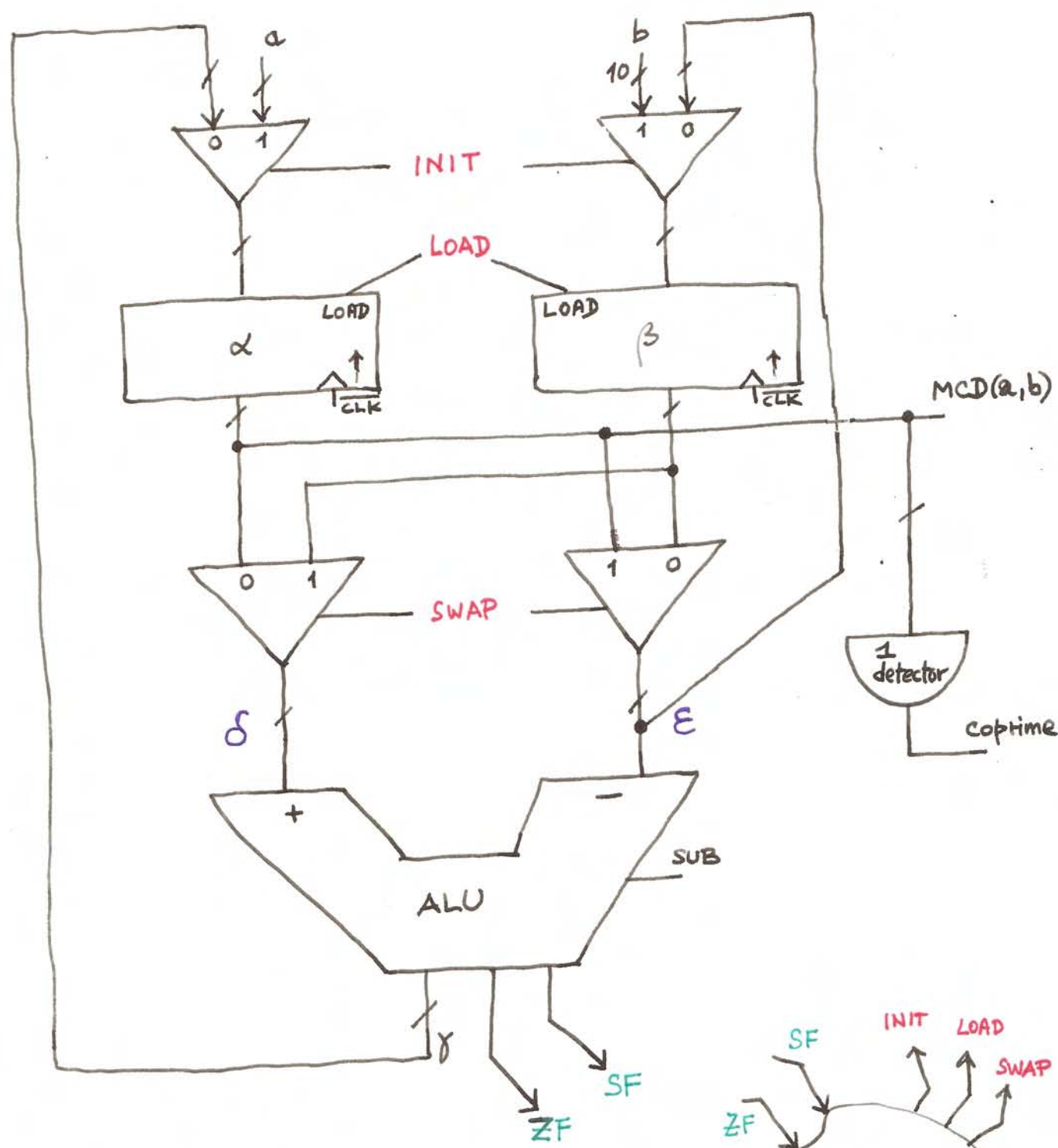


Visto che a e b sono < 512 , ipotizziamo una rappresentazione in complemento a 2 a 10 bit

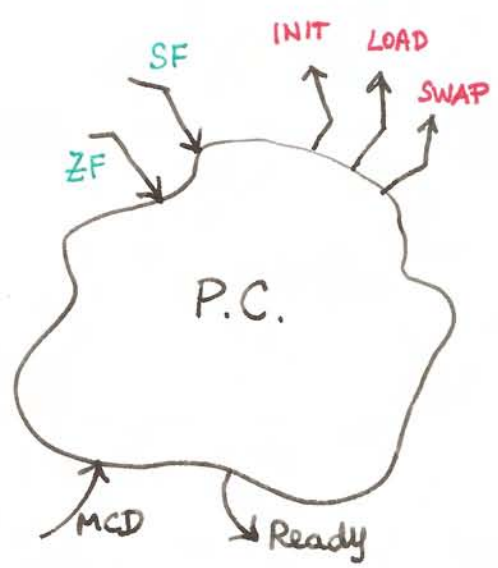
Prima di disegnare la parte operativa (cosa che ci permetterà di identificare i bit di condizione e di controllo necessari al colloquio tra P.O. e P.C.) osserviamo che al passo 3 dell'algoritmo il numero più grande viene eliminato, fu fare posto al numero più piccolo e alla differenza tra i due numeri. Ciò suggerisce una piccola modifica all'algoritmo originale: sostituire il numero più grande con la differenza ($x \leftarrow x$)

e lasciare inalterato il numero più piccolo ($\beta \leftarrow \beta$).

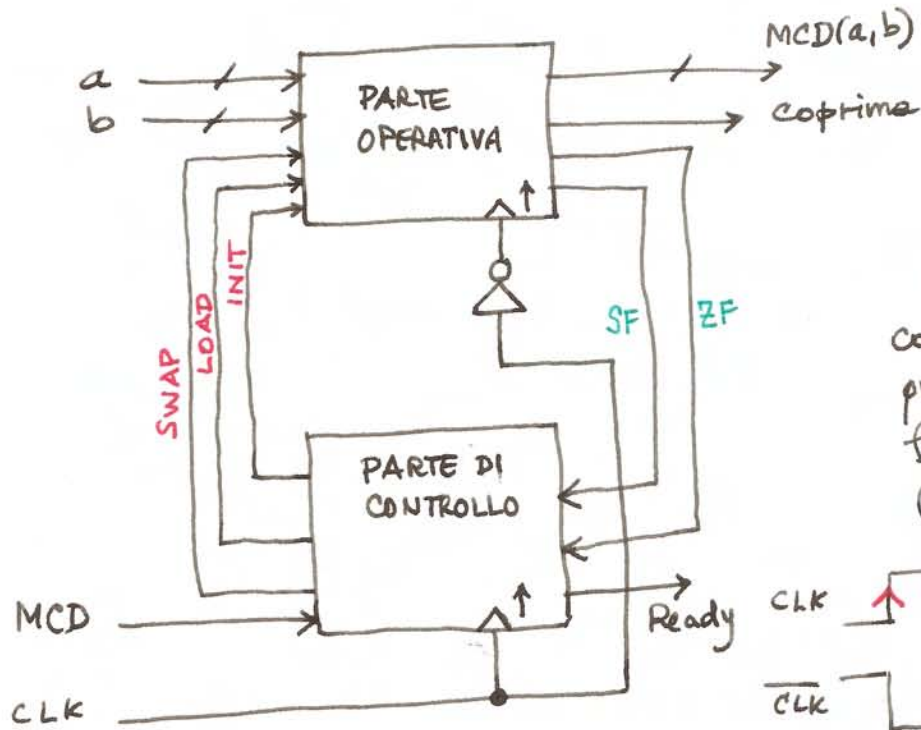
Parte Operativa



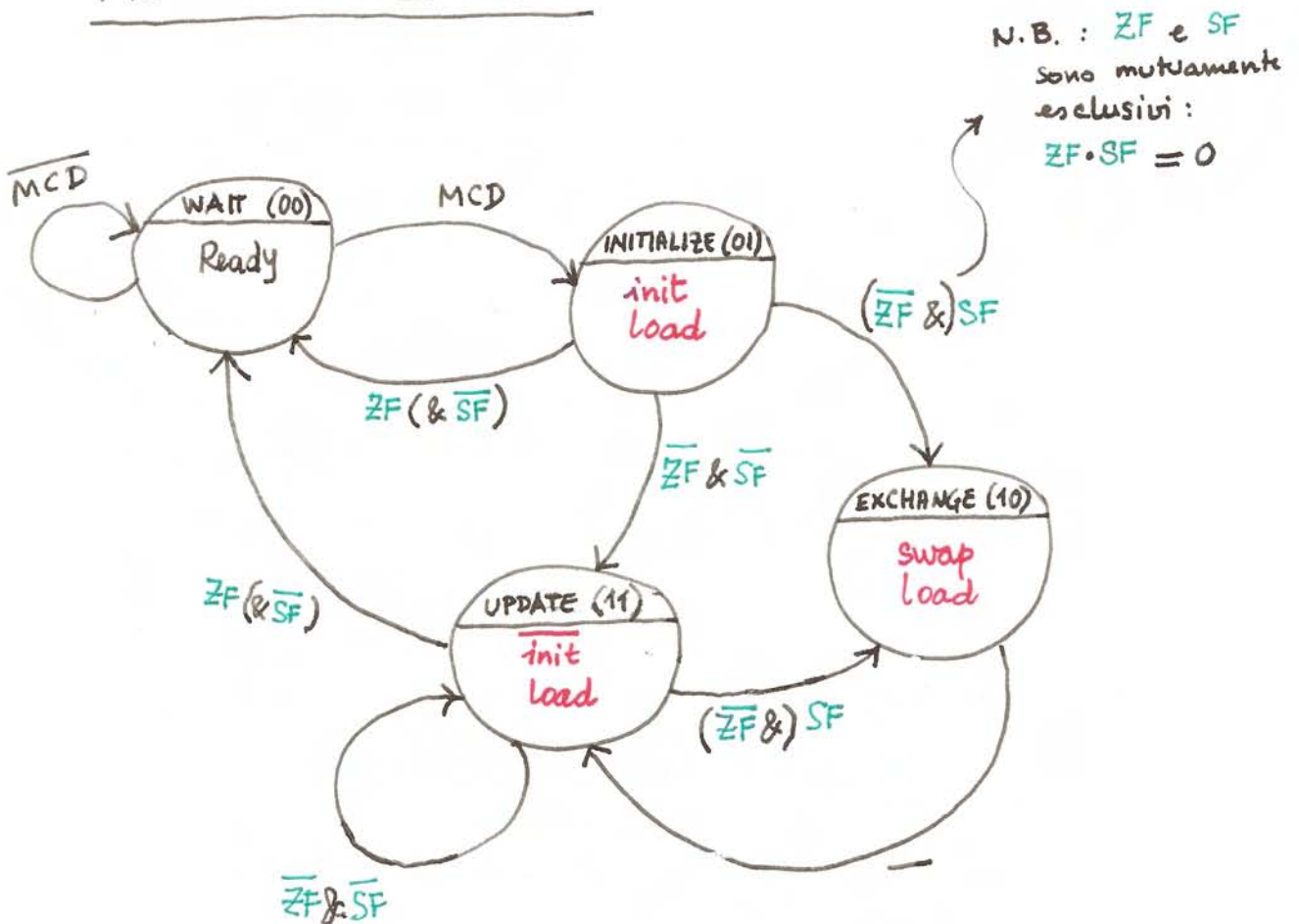
Notiamo come i numeri δ , ϵ e γ possano cambiare valore (a seguito di uno **SWAP**) indipendentemente dalla temporizzazione (fronti $\downarrow$) della P.O. Essi sono dunque asincroni rispetto al clk , mentre α e β sono sincroni.



Colloquio tra le due macchine

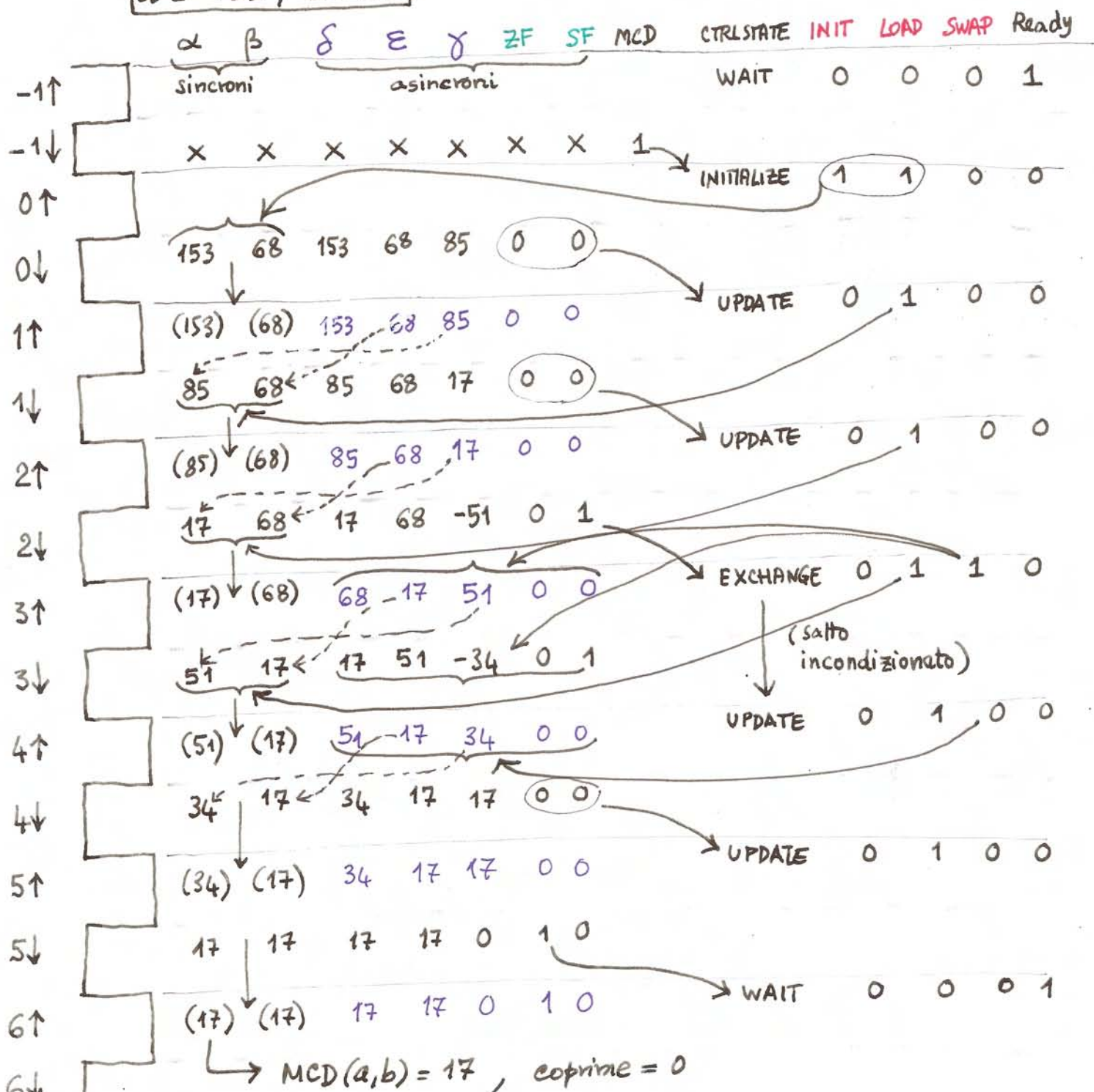


Automa di Controllo



Simulazione del funzionamento della macchina

$a = 153, b = 68$



Notare come i segnali asincroni possono cambiare valore durante i semiperiodi positivi del clock, mentre quelli sincroni restano in hold.

$$\alpha \leftarrow \gamma = \delta - \epsilon = \text{SWAP}(\alpha - \beta) + \text{SWAP}(\beta - \alpha)$$

$$\beta \leftarrow \epsilon = \text{SWAP} \beta + \text{SWAP} \alpha$$

Progetto della parte di controllo

Considerando che non ci sono lunghe catene di stati obbligati, sintetizziamo il controllo usando l'usuale registro di stato e non un contatore. Il registro sarà costituito da 2 ($= \log_2 \# \text{STATI CONTROLLO}$) flip-flop D. Usando la codifica degli stati scritta nell'automa, abbiamo:

STATO DI CONTROLLO	CODIFICA ($Q_1 Q_0$)	INGRESSI CAMPIONATI NELLO STATO	LEGGE DI TRANSIZIONE	USCITE = 1
WAIT	00	MCD	$\overline{MCD} \cdot \text{WAIT} + \overline{MCD} \cdot \text{INITIALIZE}$	Ready
INITIALIZE	01	ZF, SF	$\overline{ZF} \cdot \overline{SF} \cdot \text{UPDATE} + \overline{SF} \cdot \text{EXCHANGE} + \overline{ZF} \cdot \text{WAIT}$	INIT, LOAD
EXCHANGE	10	—	UPDATE	SWAP, LOAD
UPDATE	11	ZF, SF	(come INITIALIZE)	LOAD

Visto che non tutti gli ingressi sono campionati in ogni stato, facciamo la sintesi della funzione f con un multiplexer con i bit di stato $Q_1 Q_0$ come ingressi di selezione. Abbiamo:

STATO PRESENTE = WAIT
(00)

MCD	Q_1'	Q_0'
0	0	0
1	0	1
	↓	↓
	0	MCD

STATO PRESENTE = INITIALIZE
(01)

STATO PRESENTE = UPDATE
(11)

ZF	SF	Q_1'	Q_0'
0	0	1	1
0	1	1	0
1	0	0	0
1	1	X	X
		↓	↓
		$\overline{ZF}$	$\overline{ZF} \cdot \overline{SF}$

ZF e SF sono mutuamente esclusivi, quindi non possono essere entrambi 1 nello stesso tempo.

STATO PRESENTE = EXCHANGE
(10)

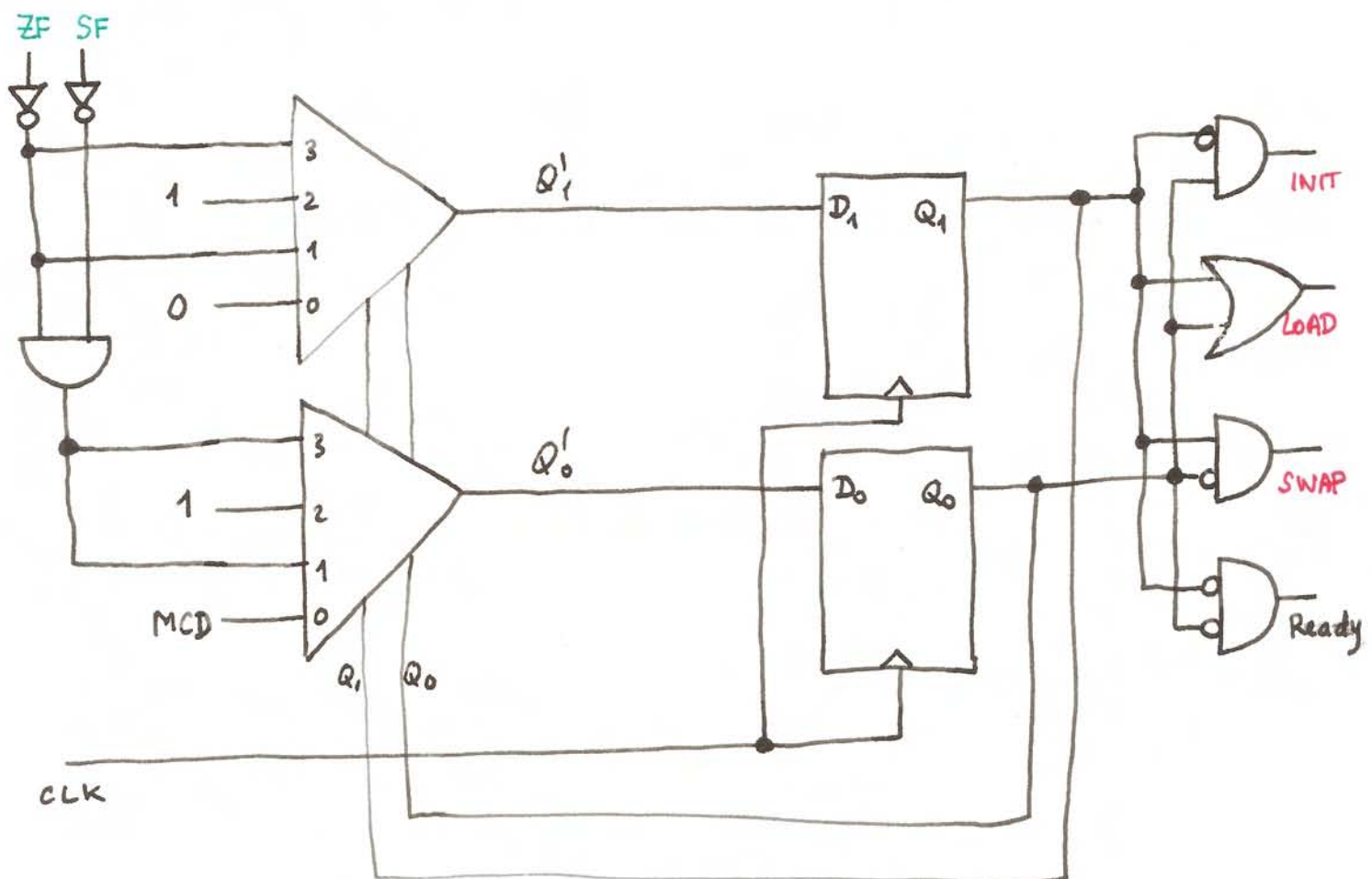
$Q_1' Q_0' = 11$

La funzione di uscita g vale invece

Q_1	Q_0	INIT	LOAD	SWAP	Ready
0	0	0	0	0	1
0	1	1	1	0	0
1	0	0	1	1	0
1	1	0	1	0	0

$\downarrow$ $\downarrow$ $\downarrow$ $\downarrow$
 $\bar{Q}_1 Q_0$ $Q_1 + Q_0$ $Q_1 \bar{Q}_0$ $\bar{Q}_1 \bar{Q}_0$

Disegniamo la parte di controllo progettata:



Questa soluzione richiede (trascurando le porte NOT):

$$\begin{aligned}
 & 2 \times (4 \text{ AND a } 3 \text{ ingressi} + 1 \text{ OR a } 4 \text{ ingressi}) + \\
 & 1 + 3 \text{ AND a } 2 \text{ ingressi} + 1 \text{ OR a } 2 \text{ ingressi} = \\
 & = \underline{8 \text{ AND a } 3 \text{ ingressi} +} \\
 & \quad \underline{4 \text{ AND a } 2 \text{ ingressi} + 2 \text{ OR a } 4 \text{ ingressi} + 1 \text{ OR a } 2 \text{ ingressi}}.
 \end{aligned}$$

Proviamo a explicitare la funzione dei multiplexer per vedere se riusciamo ad arrivare ad una soluzione più semplice, onta che richiede meno hardware :

$$\begin{cases} Q'_1 = \bar{Q}_1 \bar{Q}_0 \cdot 0 + \bar{Q}_1 Q_0 \cdot \bar{Z}F + Q_1 \bar{Q}_0 \cdot 1 + Q_1 Q_0 \cdot \bar{Z}F \\ Q'_0 = \bar{Q}_1 \bar{Q}_0 \cdot MCD + \bar{Q}_1 Q_0 \cdot \bar{Z}F \bar{S}F + Q_1 \bar{Q}_0 \cdot 1 + Q_1 Q_0 \cdot \bar{Z}F \bar{S}F \end{cases}$$

Usando le proprietà dell'algebra di Boole possiamo semplificare le espressioni in questo modo :

$$\begin{cases} Q'_1 = Q_0 \bar{Z}F (\bar{Q}_1 + Q_1) = \bar{Z}F \cdot Q_0 + Q_1 \bar{Q}_0 \\ Q'_0 = \bar{Q}_1 \bar{Q}_0 MCD + (\bar{Q}_1 + Q_1) Q_0 \bar{Z}F \bar{S}F + Q_1 \bar{Q}_0 = \\ = \bar{Q}_0 (\underbrace{\bar{Q}_1 MCD + Q_1}_{MCD + Q_1}) + Q_0 \bar{Z}F \bar{S}F = MCD \bar{Q}_0 + \bar{Z}F \bar{S}F Q_0 + Q_1 \bar{Q}_0 \end{cases}$$

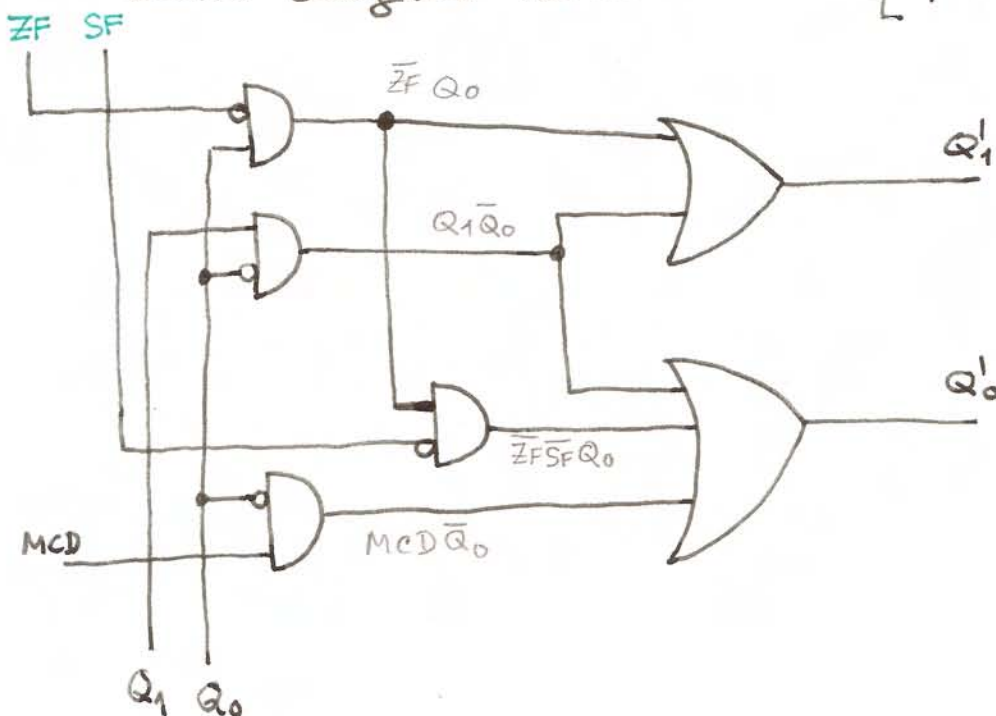
Questa soluzione richiede :

- 1 OR a 2 ingressi
- 1 OR a 3 ingressi
- 4 AND a 2 ingressi

Rispetto alla soluzione coi multiplexer, che richiedeva

- 8 AND a 3 ingressi
- 2 OR a 4 ingressi
- 1 AND a 2 ingressi

c'è un indubbio risparmio.



Vediamo per esercizio anche la sintesi di f con la tabella completa (dove sfruttiamo con don't care la condizione $ZF \cdot SF = 0$): (*)

(*) Vediamo che la soluzione trovata a pag. 8 corrisponde a quella trovata qui con Karnaugh a 5 variabili: è dunque la soluzione minima.

Nota: con Karnaugh a 5 variabili si cercano sottogruppi identici nelle due tabelle ($MCD=0, MCD=1$). Quelli rimanenti vengono presi con l'opportuna MCD .

MCD	ZF	SF	Q_1	Q_0	Q_1'	Q_0'
0	0	0	0	0	0	0
0	0	0	0	1	1	1
0	0	0	1	0	1	1
0	0	0	1	1	1	1
0	0	1	0	0	0	0
0	0	1	0	1	1	0
0	0	1	1	0	1	1
0	0	1	1	1	1	1
0	1	0	0	0	0	0
0	1	0	0	1	1	1
0	1	0	1	0	0	0
0	1	0	1	1	0	0
0	1	1	0	0	X	X
0	1	1	0	1	X	X
0	1	1	1	0	X	X
0	1	1	1	1	X	X
1	0	0	0	0	0	1
1	0	0	0	1	1	1
1	0	0	1	0	1	1
1	0	0	1	1	1	1
1	0	1	0	0	0	1
1	0	1	0	1	1	0
1	0	1	1	0	1	1
1	0	1	1	1	1	0
1	1	0	0	0	0	1
1	1	0	0	1	0	0
1	1	0	1	0	1	1
1	1	0	1	1	0	0
1	1	1	0	0	X	X
1	1	1	0	1	X	X
1	1	1	1	0	X	X
1	1	1	1	1	X	X

$Q_1 Q_0$

00	01	11	10
00	0	1	1
01	0	1	1
11	X	X	X
10	0	0	0

MCD=0

$$Q_1' = \overline{ZF} Q_0 + Q_1 \overline{Q_0}$$

identici: non c'è d.p. da MCD

$Q_1 Q_0$

00	01	11	10
00	0	1	1
01	0	1	1
11	X	X	X
10	0	0	0

MCD=1

$$Q_1' = \overline{ZF} Q_0 + Q_1 \overline{Q_0}$$

$Q_1 Q_0$

00	01	11	10
00	0	1	1
01	0	0	0
11	X	X	X
10	0	0	0

MCD=0

$$Q_0' = \overline{ZF} \overline{SF} Q_0 + Q_1 \overline{Q_0}$$

questi sono prodotti individuali: non portano sempre alla soluzione minima

$Q_1 Q_0$

00	01	11	10
00	1	1	1
01	1	0	0
11	X	X	X
10	1	0	0

MCD=1

$$Q_0' = \overline{Q_0} + \overline{ZF} \cdot \overline{SF}$$

$$Q_0' = Q_0 \overline{ZF} \overline{SF} + Q_1 \overline{Q_0} + MCD \overline{Q_0}$$

(sol. minima)