

# Università degli Studi di Firenze, Facoltà di Ingegneria

## Esame di Microelettronica

### Domanda 1 : Gli ASIC

Fornire una descrizione generale dell'architettura delle CPLD ALTERA della serie MAX7000AE. Approfondire la descrizione degli elementi delle macrocelle riferendosi allo schema di Figura 1.

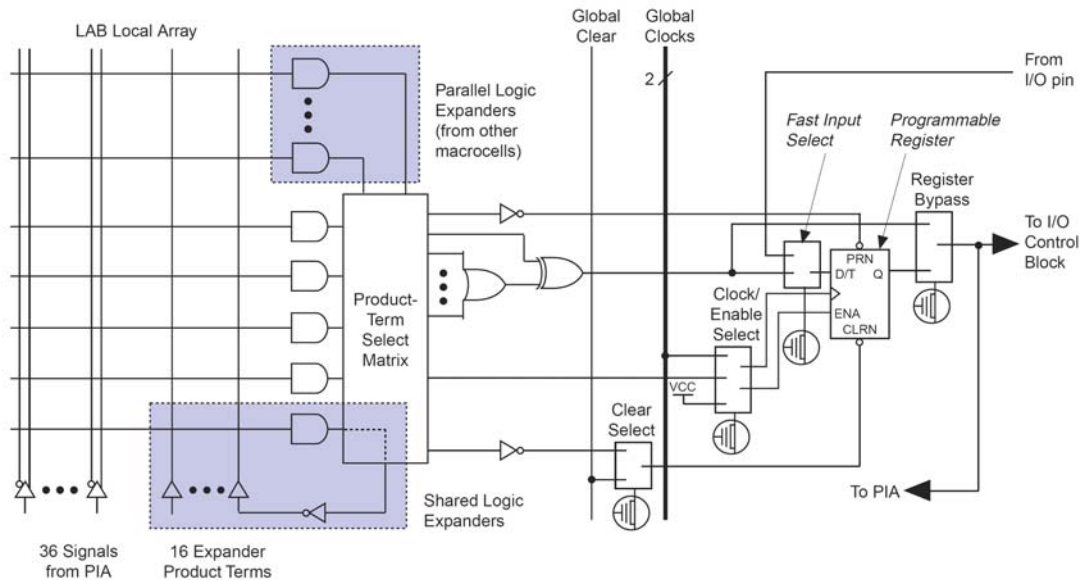


Figura 1 - Schema di riferimento di una macrocella delle CPLD della serie MAX7000AE

### Domanda 2 : Convertitore A/D a rampa

Descrivere il principio di funzionamento del convertitore A/D a rampa realizzato a componenti discreti e riportato nel seguente schema circuitale di Figura 2.

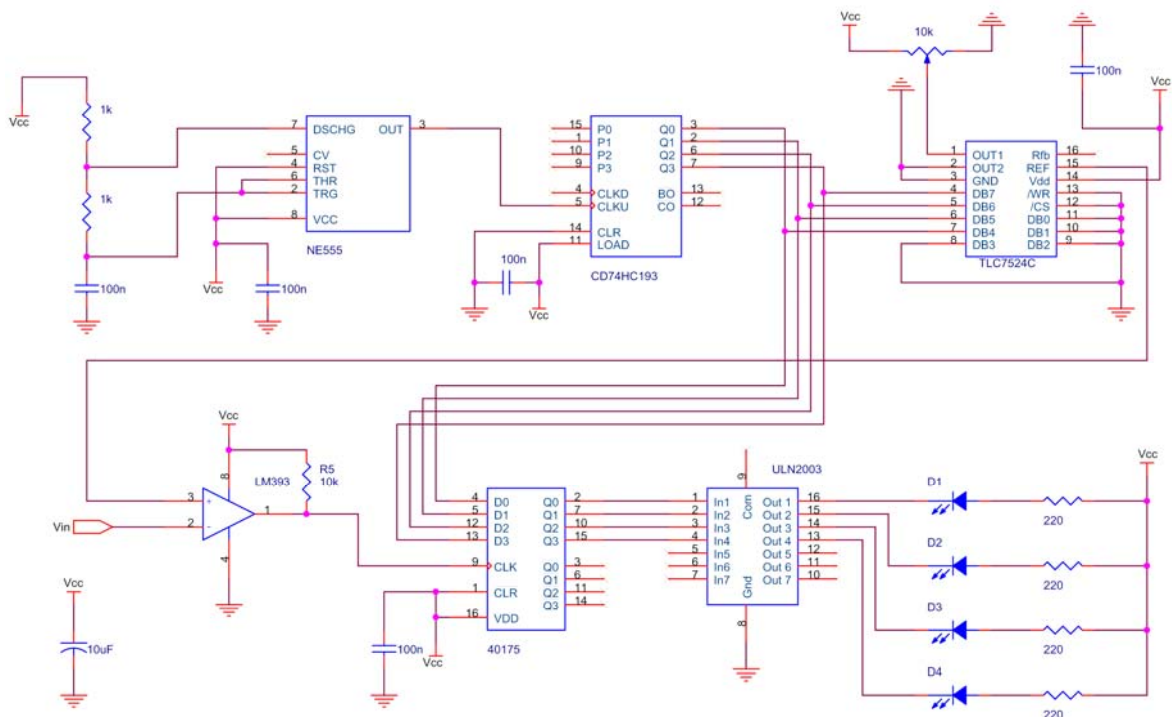


Figura 2 – Schema circuitale del convertitore A/D a rampa

### Domanda 3 : Timing Model di CPLD ALTERA della famiglia MAX7000

Supponendo di voler implementare il seguente schema circuitale in una CPLD EPM7032AE-10, il cui timing model e tabella dei timing sono riportati a pagina seguente, si calcolino :

- Tempi di setup ed hold per gli ingressi INPUT e FAST\_INPUT
- Ritardo fra fronte di clock valido ed uscita OUTPUT valida corrispondente
- Massima frequenza di clock utilizzabile
- Ritardo fra ingresso INPUT ed uscita OUTPUT\_ASYNC

Condizioni :

- Tensione di alimentazione 3.3V
- Slow Slew Rate disattivato
- Ingresso CLOCK dedicato su linea dedicata (il flip-flop inst1 riceve però un **gated clock**)
- Ingresso INPUT generico
- Ingresso FAST\_INPUT generico e configurato come input fast per il flip-flop inst1
- Ingresso ENABLE generico da non considerare nell'esercizio
- La porta NOT è in realtà un buffer che forza il fitter ad utilizzare uno SHAREABLE EXPANDER

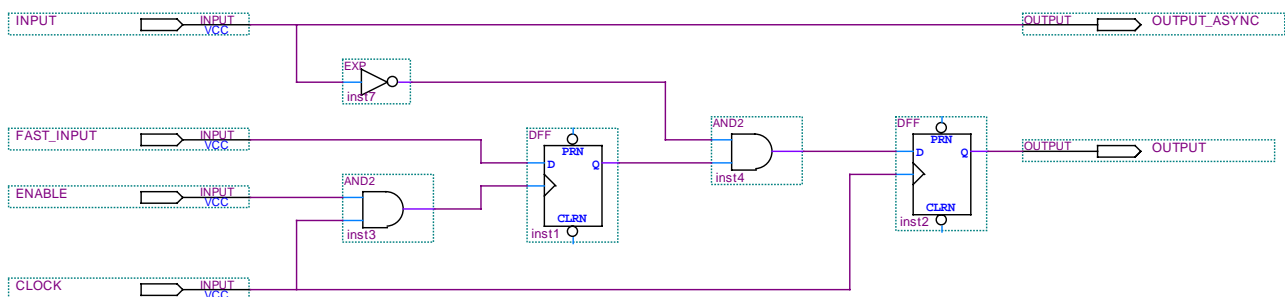


Figura 3 - Schema circuitale da analizzare

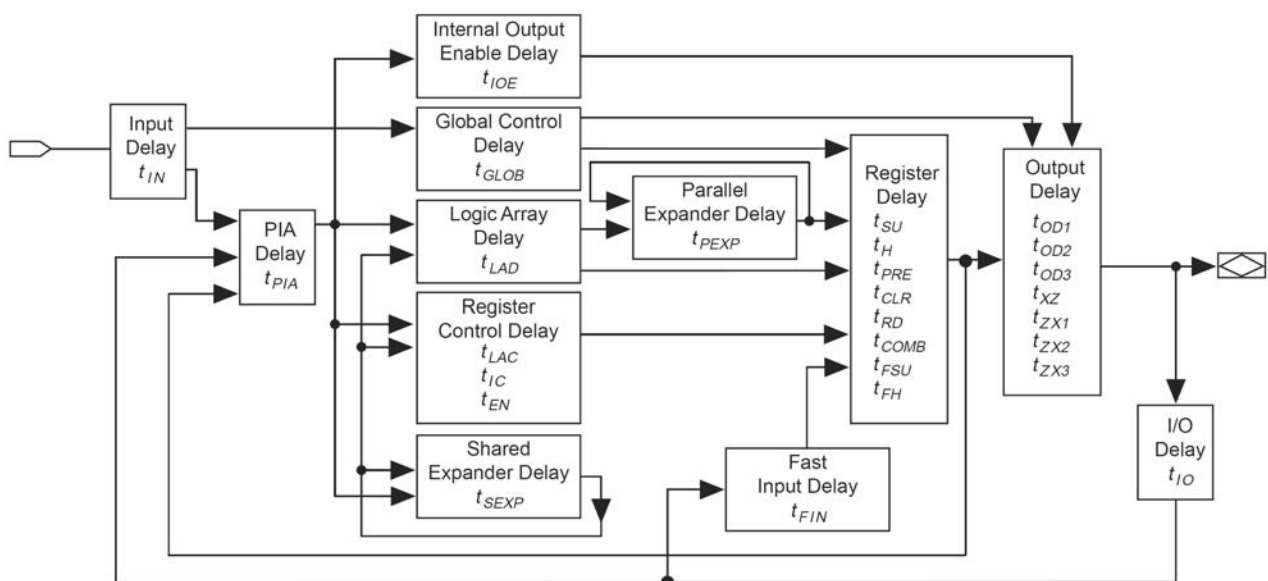


Figura 4 - Timing Model della CPLD serie MAX7000AE

<b>Symbol</b>	<b>Parameter</b>	<b>Min</b>	<b>Max</b>	<b>Unit</b>
<i>tIN</i>	Input pad and buffer delay		1.5	ns
<i>tIO</i>	I/O input pad and buffer delay		1.5	ns
<i>tFIN</i>	Fast input delay		3.4	ns
<i>tSEXP</i>	Shared expander delay		4.0	ns
<i>tPEXP</i>	Parallel expander delay		1.0	ns
<i>tLAD</i>	Logic array delay		3.3	ns
<i>tLAC</i>	Logic control array delay		1.2	ns
<i>tIOE</i>	Internal output enable delay		0.0	ns
<i>tOD1</i>	Output buffer and pad delay, slow slew rate = off VCCIO = 3.3V		1.8	ns
<i>tOD2</i>	Output buffer and pad delay, slow slew rate = off VCCIO = 2.5V		2.3	ns
<i>tOD3</i>	Output buffer and pad delay, slow slew rate = on VCCIO = 2.5V or 3.3 V		6.8	ns
<i>tZX1</i>	Output buffer enable delay, slow slew rate = off		5.0	ns
	VCCIO = 3.3 V			
<i>tZX2</i>	Output buffer enable delay, slow slew rate = off VCCIO = 2.5V		5.5	ns
<i>tZX3</i>	Output buffer enable delay, slow slew rate = on		10.0	ns
	VCCIO = 3.3 V			
<i>tXZ</i>	Output buffer disable delay		5.0	ns
<i>tSU</i>	Register setup time	2.8		ns
<i>tH</i>	Register hold time	1.3		ns
<i>tFSU</i>	Register setup time of fast input	1.5		ns
<i>tFH</i>	Register hold time of fast input	1.5		ns
<i>tRD</i>	Register delay		1.5	ns
<i>tCOMB</i>	Combinatorial delay		1.3	ns
<i>tIC</i>	Array clock delay		2.5	ns
<i>tEN</i>	Register enable time		1.2	ns
<i>tGLOB</i>	Global control delay		1.9	ns
<i>tPRE</i>	Register preset time		2.6	ns
<i>tCLR</i>	Register clear time		2.6	ns
<i>tPIA</i>	PIA delay		2.1	ns
<i>tLPA</i>	Low-power adder		5.0	ns

Tabella 1 - Tabella dei timing EPM7032AE-10

#### **Domanda 4 : Realizzazione di un progetto su PC tramite ambiente di sviluppo QUARTUS II**

Implementare, in una CPLD, la macchina a stati che gestisce un sistema di fornitura bevande in lattina. Realizzare il progetto in un linguaggio a scelta fra **AHDL** e **VHDL** senza utilizzare un **TOP Level Design** grafico.

Il sistema è composto da due contenitori (due gusti) in cui sono impilate le lattine, un lettore di monete (solo 50cent), due pulsanti (per selezionare il gusto desiderato) e due saracinesche (una per ogni contenitore) che lasciano passare una lattina per volta. Il sistema permette di prelevare una sola lattina dopo che è stata inserita una moneta e non tiene memoria del numero di monete inserite.

In condizioni di riposo il sistema attende l'inserimento di una moneta. Dopo l'inserimento il sistema attende che venga premuto uno dei due pulsanti. Premuto un pulsante, il sistema apre la saracinesca associata. Il passaggio della lattina dalla saracinesca viene rilevato attraverso una fotocellula. Il segnale della fotocellula pone il sistema in attesa di una nuova moneta.

Il lettore di monete fornisce un segnale al livello logico 1 per tutto il tempo in cui la moneta passa all'interno del lettore stesso, altrimenti livello logico 0. Il segnale è chiamato **MONETA**.

Le due saracinesche possono essere aperte portando al livello logico 1 il loro segnale di comando. Essendo le saracinesche in numero pari a due, i segnali saranno **SARACINESCA\_1** e **SARACINESCA\_2**. Quando i due segnali sono al livello logico 0 le saracinesche sono chiuse.

La fotocellula che legge il passaggio della lattina verso la bocca, attraverso la quale l'utente può prelevare, fornisce un segnale a livello logico 1 per tutto il tempo di passaggio della lattina stessa, altrimenti 0. Tale segnale è chiamato **FOTOCELLULA**.

I due pulsanti, chiamati **GUSTO\_1** e **GUSTO\_2**, indicano la scelta del gusto da parte dell'utente. Quando un pulsante è premuto il segnale è a livello logico 1 altrimenti 0.

La macchina a stati dovrà avere quindi un numero di ingressi pari a quattro ed un numero di uscite pari a due.

La macchina a stati dovrà avere un totale di quattro stati. Il primo stato è **ATTESA\_MONETA** nel quale il sistema attende l'arrivo di una moneta. In questo stato le uscite saranno tutte a livello logico 0. Arrivata la moneta la macchina a stati deve transitare in un secondo stato (**ATTESA\_SCELTA**) nel quale attende la scelta da parte dell'utente. Anche in questo stato le uscite dovranno essere tutte a livello logico 0. Alla pressione di un pulsante **GUSTO\_x** la macchina deve transitare nello stato **APERTURA\_x** relativo ed aprire la saracinesca **x** opportuna. Nello stato **APERTURA\_x** l'uscita **SARACINESCA\_x** relativa dovrà essere portata al livello logico 1.

La macchina dovrà permanere nello stato **APERTURA\_x** fino a quando la fotocellula non fornirà un livello logico 1 sul segnale **FOTOCELLULA** indicando il transito di una lattina verso la bocca di presa. In quel caso la macchina dovrà transitare dallo stato **APERTURA\_x** nello stato **ATTESA\_MONETA**.

Il diagramma degli stati è riportato nello schema presente nella seguente pagina insieme a due tabelle riassuntive delle condizioni di transizione fra stati e delle uscite relative.

L'azzeramento iniziale della macchina deve essere effettuato attraverso una normale linea **RESET**.

Il segnale di temporizzazione deve essere fornito attraverso una normale linea **CLOCK**.

Stato Corrente	Stato Futuro	Condizione di transizione
ATTESA_MONETA	ATTESA_SCELTA	MONETA = 1
ATTESA_SCELTA	APERTURA_1	GUSTO_1 = 1
	APERTURA_2	GUSTO_2 = 1
APERTURA_1	ATTESA_MONETA	FOTOCCELLULA = 1
APERTURA_2	ATTESA_MONETA	FOTOCCELLULA = 1

**Tabella 2 - Condizioni di transizione**

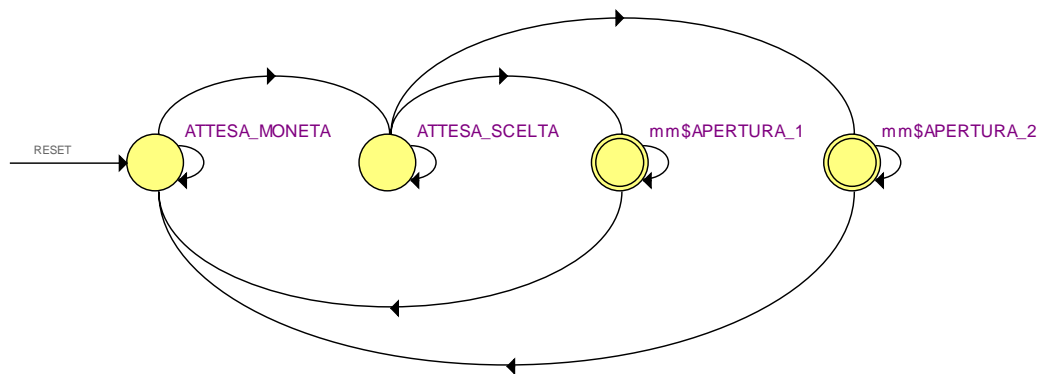
In tutte le combinazioni degli ingressi non esplicitamente dichiarate lo stato futuro dovrà essere uguale allo stato corrente.

Dipendentemente dal metodo di implementazione della macchina a stati che si intende utilizzare, le seguenti regole possono necessitare o non necessitare di integrazioni atte a garantire il corretto funzionamento della macchina stessa.

Stato Corrente	Uscite
ATTESA_MONETA	SARACINESCA_1 = 0, SARACINESCA_2 = 0
ATTESA_SCELTA	SARACINESCA_1 = 0, SARACINESCA_2 = 0
APERTURA_1	SARACINESCA_1 = 1, SARACINESCA_2 = 0
APERTURA_2	SARACINESCA_1 = 0, SARACINESCA_2 = 1

**Tabella 3 – Definizione delle uscite**

**La macchina descritta prevede che le uscite NON dipendano dagli ingressi ma dal solo stato corrente !**



**Figura 5 - Diagramma degli stati**

Lasciar scegliere automaticamente il dispositivo al FITTER di QUARTUS II fra i dispositivi della famiglia MAX7000AE.

Effettuare una simulazione che mostri il comportamento della macchina a stati effettuando il posizionamento di una sola scatola.

**Realizzare il progetto su una cartella del desktop inserendo un commento nel file sorgente contenente Nome, Cognome, Numero di matricola ed eventuale e-mail dello studente.**