

Design for Testability (DFT): *Full-Scan*

- **Definizioni**
- **Metodi ad-hoc**
- ***Scan design***
 - **Regole di progetto**
 - **Registri**
 - **Flip-flops**
 - **Scan test sequences**
 - **Overhead**
 - **Sistemi di progetto basati sulla scansione**
 - **Sommario**

1

Definizioni

- ***Design for testability (DFT)* si riferisce a quelle tecniche di progetto che rendono più efficaci la test generation e l'applicazione dei test.**
- **Metodi di DFT per circuiti digitali:**
 - **Metodi ad-hoc**
 - **Metodi strutturati:**
 - ***Scan***
 - ***Partial Scan***
 - ***Built-in self-test (BIST)***
 - ***Boundary scan***

2

Metodi di DFT Ad-Hoc

- **Buone pratiche di progetto imparate attraverso l'esperienza:**
 - Evitare retroazioni asincrone (unclocked).
 - Rendere i flip-flops inicializzabili.
 - Evitare gate ridondanti o con fan-in grande.
 - Aggiungere segnali di controllo di test per segnali difficili da controllare.
 - Evitare gated clocks.
 - Considerare le richieste degli ATE.
- **Revisioni di progetto condotte da esperti.**
- **Svantaggi:**
 - Esperti non sempre disponibili.
 - La test generation è spesso fatta a mano => bassa copertura.
 - Iterazioni sul progetto.

3

Scan Design

- Il circuito è progettato utilizzando regole predefinite.
- Si aggiungono strutture di test (hardware) al progetto:
 - Aggiunge un segnale di *test control* (TC) come primary input.
 - Sostituisce i flip-flop con *scan flip-flops* (SFF) e li connette per formare uno o più shift registers in modo test.
 - Rende l'input/output di ciascuno scan shift register controllabile/osservabile dai PI/PO.
- Usa l'ATPG combinatorio per ottenere tests per tutti i guasti collaudabili nella logica combinatoria.
- Aggiunge test per gli shift register tests e converte i test prodotti dall'ATPG in sequenze di scan da utilizzare nei test di produzione.

4

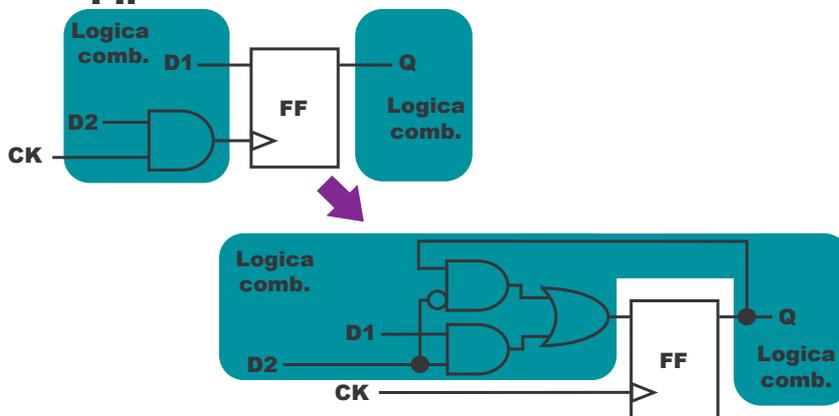
Regole di Scan Design

- Utilizzare solo flip-flop clocked di tipo D per tutte le variabili di stato.
- Almeno un pin di PI pin deve essere disponibile per il test; eventualmente se ne può usare più di uno.
- Tutti i clock devono essere controllati dai PIs.
- I clock non devono entrare negli ingressi dati dei flip-flop.

5

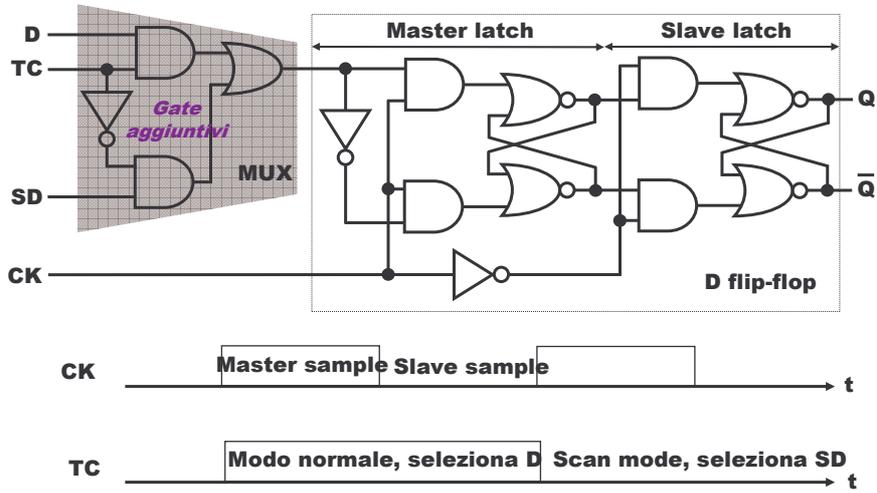
Correzioni

- Tutti i clock devono essere controllati dai PI.



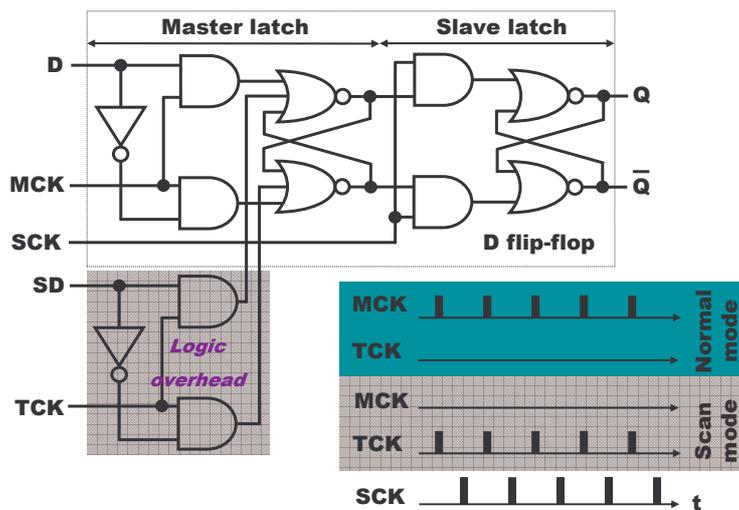
6

Scan Flip-Flop (SFF)



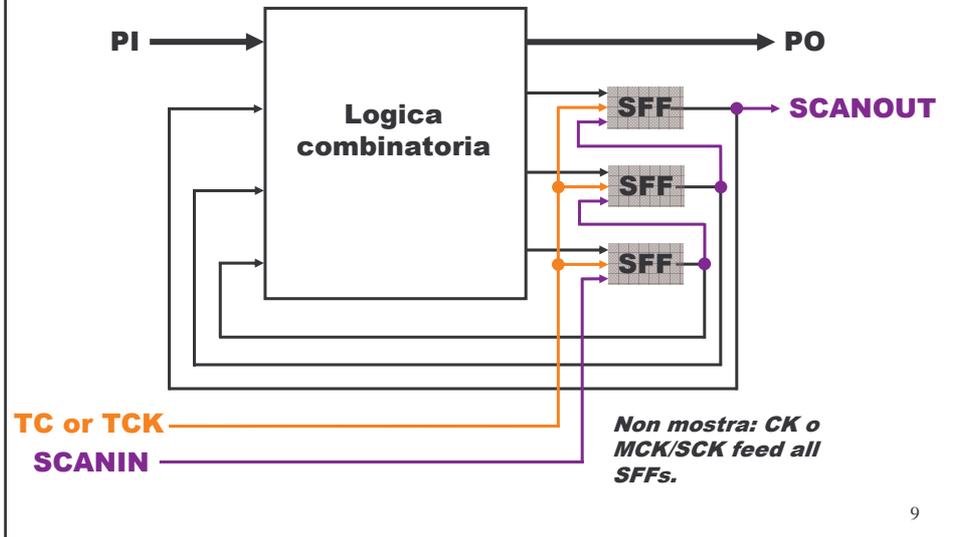
7

Level-Sensitive Scan-Design Flip-Flop (LSSD-SFF)

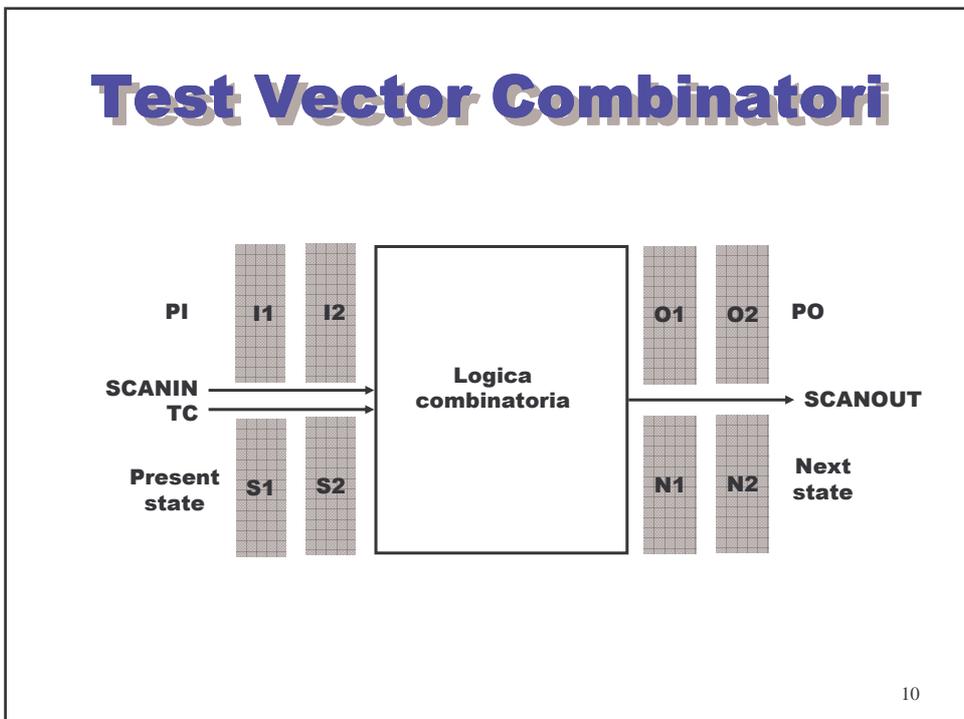


8

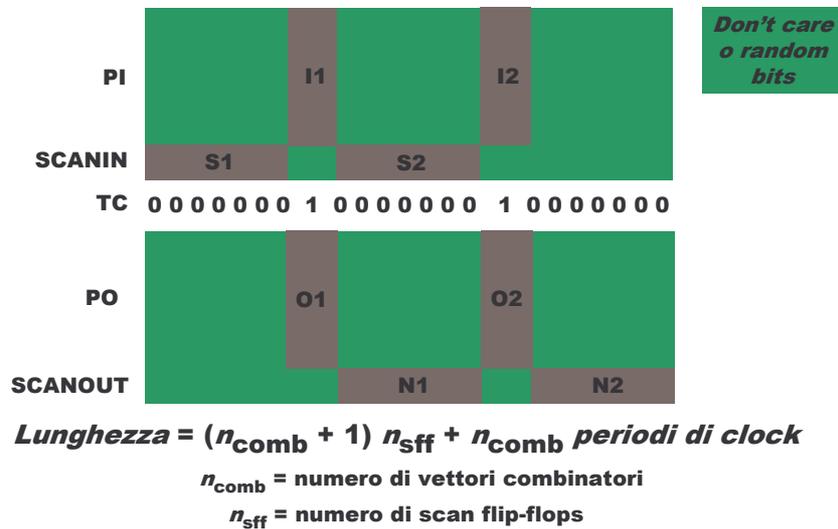
Strutture di Scan



Test Vector Combinatori



Comb. Test Vectors



11

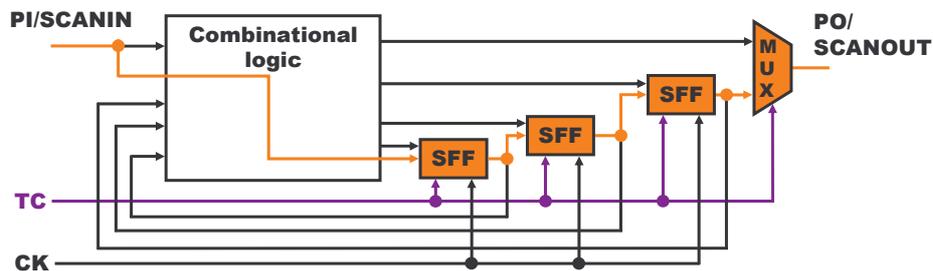
Testing Scan Register

- Gli scan register devono essere collaudati prima di applicare scan test sequences.
- Shiftare una sequenza 00110011 . . . di lunghezza $n_{\text{sff}}+4$ in modo scan (TC=0) che produce tutte le transizioni 00, 01, 11 e 10 in tutti i flip-flops e osserva i risultati all'uscita di SCANOUT.
- Lunghezza totale della sequenza di test: $(n_{\text{comb}} + 2) n_{\text{sff}} + n_{\text{comb}} + 4$ periodi di clock.
- Esempio: 2,000 scan flip-flops, 500 vettori comb., lunghezza totale di test $\sim 10^6$ periodi di clock.
- Scan register multipli riducono la lunghezza di test.

12

Scan Register multipli

- I flip-flop di scan possono essere distribuiti fra ciascun numero of shift registers, ciascuno con *scanin* e *scanout* separati.
- La lunghezza della sequenza di test è determinata dal registro più lungo.
- Basta solo un pin di *test control* (TC).



13

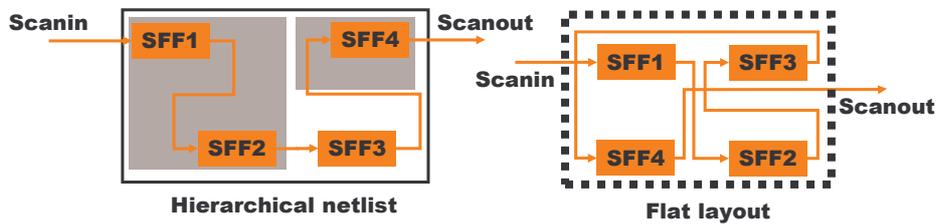
Scan Overheads

- I/O pin: un pin addizionale.
- Area overhead:
 - $Gate\ overhead = [4 n_{sff} / (n_g + 10 n_{ff})] \times 100\%$, $n_g = gate\ comb.$; $n_{ff} = flip-flops$;
 - Esempio – $n_g = 100k\ gates$, $n_{ff} = 2k\ flip-flops$, overhead = 6.7%.
 - Stime più accurate tengono in conto dell'area per le interconnessioni.
- Prestazioni:
 - il ritardo dei multiplexer delay viene aggiunto al critical path; circa due ritardi di gate.
 - Aumento del ritardo dei flip-flop dovuto all'aumento di capacità di carico dato dal fanout addizionale; circa 5-6%.

14

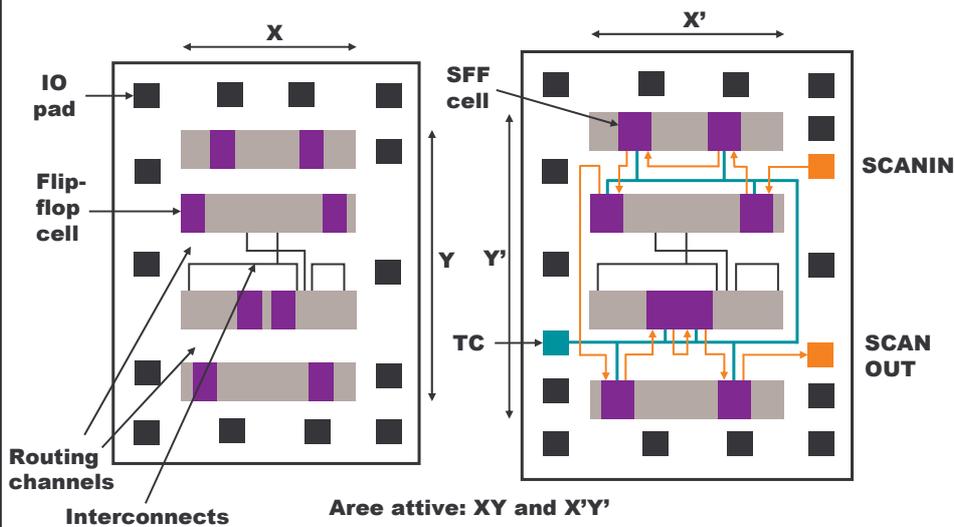
Scan gerarchico

- I flip-flop di scan sono concatenati con sottoreti prima di concatenare le sottoreti stesse chaining subnetworks.
- Vantaggi:
 - Inserzione automatica
 - Mantiene la gerarchia inalterata
- Svantaggi: layout non-ottimo.



15

Scan layout ottimo



16

Esempio: Scan Layout

- 2,000-gate CMOS chip
- Frazione di area delle celle dei flip-flop, $s = 0.478$
- Incremento di width dei scan flip-flop (SFF), $\alpha = 0.25$
- Frazione di area di routing, $\beta = 0.471$
- Altezza delle celle, $T = 10$
- Overhead calcolato = 17.24%
- Dati reali misurati:

Implementazione dello scan	Area overhead	Clock rate normalizzato
Nessuno	0.0	1.00
Gerarchico	16.93%	0.87
Layout ottimo	11.90%	0.91

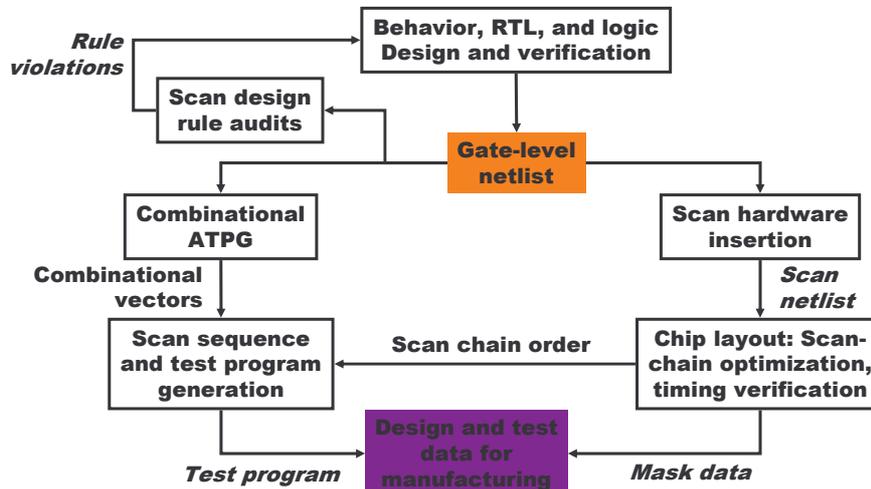
17

Esempio di ATPG: S5378

	Originale	Full-scan
Numero di gate combinatori	2,781	2,781
Numero di flip-flops non-scan (10 gates)	179	0
Numero di scan flip-flops (14 gates)	0	179
Gate overhead	0.0%	15.66%
Numero di guasti	4,603	4,603
PI/PO per ATPG	35/49	214/228
Fault coverage	70.0%	99.1%
Fault efficiency	70.9%	100.0%
CPU time su SUN Ultra II, processore a 200MHz	5,533 s	5 s
Numero di vettori generati	414	585
Lunghezza della sequenza di test	414	105,662

18

Scan Design automatico



19

Timing e consumo di potenza

- Piccoli ritardi nello scan path e clock skews possono dare luogo a corse.
- Grandi ritardi nello scan path richiedono un clock più lento.
- Multiplexers dinamici: skew fra diversi segnali di TC possono causare selezioni contemporanee di D e SD.
- L'attività casuale nel circuito durante la fase di scan può causare eccessivo consumo di potenza.

20

Sommario

- **Scan path è la tecnica più popolare di DFT:**
 - Progetto basato sulle regole
 - Inserimento automatico del DFT
 - Combinational ATPG
- **Vantaggi:**
 - Design automation
 - Copertura di guasto e diagnosi
 - Possibilità di sfruttare la gerarchia
 - Overhead moderati (area (~10%) e speed (~5%))
- **Svantaggi:**
 - Quantità di dati e tempi di test
 - Impossibilità di applicare test alla velocità operativa del circuito