

# Collaudo basato sui difetti

M. Favalli

Engineering Department in Ferrara

## Resa

### Difetti nei sistemi digitali VLSI

- Difetti di produzione: area finita di un chip contenente circuiti malfunzionanti a causa di errori o anomalie nel processo di produzione

- Extra e missing material



extra material

missing material

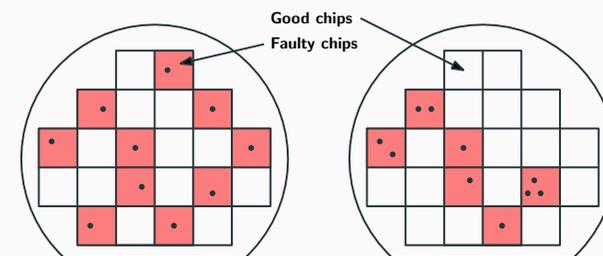
- Costo del collaudo
- Un chip privo di difetti viene considerato come "buono"

### Resa (yield)

- La frazione (o percentuale) di chip buoni prodotti di definisce resa ( $Y$ )
- La resa contribuisce a determinare il costo di un chip:

$$\frac{\text{Costo di fabbricazione e collaudo di un wafer}}{Y \times \text{Numero di chip per wafer}}$$

- La resa dipende dalla distribuzione dei difetti in un wafer



Difetti unclustered  
 $Y = 12/22 = 0.55$

Difetti clustered  
 $Y = 16/22 = 0.72$

- Modello spot semplificato dei difetti (piccola area quadrata di dimensione fissa)
  - la dimensione di solito non é fissa e la forma é rappresentabile meglio come una circonferenza
- Area del chip  $A$  vista come il numero di piccole aree
- Ogni difetto risulta in un guasto
  - in realtà esiste un area critica per ogni tipo di difetto
- Difetti spazialmente non correlati
  - vale solo per alcune classi di difetti

- Densità media dei difetti  $d$  stimabile contando il numero di difetti per chip su un campione di  $n$  chip

$$d = \frac{\sum_{i=1..n} \sum_{j=1..A} e_{ij}}{nA}$$

- ove  $e_{ij} = 1$  se il quadratino ha un difetto e 0 altrimenti
- Se i difetti sono non correlati  $d$  é una stima della probabilità che un quadratino abbia un difetto
- La probabilità di avere  $x$  difetti in un chip é data dalla distribuzione binomiale

$$Prob\{x \text{ difetti}\} = \binom{A}{x} d^x (1 - d)^{A-x}$$

- La resa  $Y$  si ha per  $x = 0$

$$Y = Prob\{\text{chip corretto}\} = \binom{A}{0} d^0 (1 - d)^A \simeq 1 - Ad$$

- Se la densità dei difetti é sufficientemente bassa, la binomiale é approssimata da una distribuzione di Poisson con v.m.  $\lambda = Ad$

$$Prob\{x \text{ difetti}\} = \frac{\lambda^x e^{-\lambda}}{x!}$$

- Per  $x = 0$  si ha  $Y = e^{-\lambda} = e^{-Ad} \simeq 1 - Ad$
- Questo modello é stato esteso per tenere conto della correlazione fra i difetti

- Parametri
  - Densità dei difetti ( $d$ ): numero medio di difetti per unità di area
  - Area del chip ( $A$ )
  - Parametro di clustering ( $\alpha$ )
- Probabilità di avere  $x$  difetti su un chip:

$$p(x) = \frac{\Gamma(\alpha + x)}{x! \Gamma(\alpha)} \frac{\left(\frac{Ad}{\alpha}\right)^x}{\left(1 + \frac{Ad}{\alpha}\right)^{\alpha+x}}$$

- Dove,  $\Gamma$  é la funzione gamma
- Se  $\alpha = 0$ ,  $p(x)$  é la funzione delta di Dirac (max clustering)
- Se  $\alpha = \infty$ ,  $p(x)$  é la distribuzione di Poisson (min clustering)

- Resa:  $Y = \text{Prob}\{\text{zero difetti in un chip}\}$

$$Y = p(0) = \left(1 + \frac{Ad}{\alpha}\right)^{-\alpha}$$

- esempio:  $Ad = 1.0$ ,  $\alpha = 0.5$ ,  $Y = 0.58$
- Difetti unclustered:  $\alpha = \infty$ ,  $Y = e^{-Ad}$ 
  - esempio  $Ad = 1.0$ ,  $\alpha = \infty$ ,  $Y = 0.37$  (stima pessimistica)

- Defect level ( $DL$ ) é il rapporto fra il numero di chip guasti che passano il test e quello di chip che passano il test
- Misura l'efficacia del processo di test, di solito si misura in parti per milione (ppm)
- Rappresenta contemporaneamente anche una misura della qualità del prodotto
- Per una tecnologia matura  $DL > 500\text{ppm}$  non é un livello accettabile da chi acquista i chip

## Calcolo del defect level

- Dati di ritorno dal campo:** statistica sui chip guasti che vengono restituiti al produttore
  - problemi dati dal fatto che alcuni tipi di difetto si manifestano durante il normale funzionamento del chip senza essere presenti alla fine del processo di produzione
- Dati di collaudo:** il defect level viene calcolato a partire dalla copertura di guasto delle sequenze di test applicate e dal rateo di eliminazione dei chip. Un modello di resa modificato viene fittato su tali dati

## Equazione di resa modificata

- Il collaudo si basa su modelli di guasto e non sui difetti reali, per cui bisogna riformulare l'equazione di resa
- Parametri
  - Densità di guasti ( $f$ ): numero medio di guasti per unità di area del chip
  - Parametro di clustering ( $\beta$ )
  - Copertura di guasti ( $C$ )
- Equazione di resa modificata sostituendo i guasti ai difetti

$$Y(C) = \left(1 + \frac{CAf}{\beta}\right)^{-\beta} \quad (1)$$

- Se il collaudo rimuove tutti i chip guasti  $C = 1$  (approssimazione ideale, guasti sono diversi dai difetti)

$$Y = Y(1) = \left(1 + \frac{Af}{\beta}\right)^{-\beta}$$

- Per una sequenza con copertura  $C$ ,  $Y(C)$  chip passano il collaudo

$$DL(C) = \frac{Y(C) - Y(1)}{Y(C)} = 1 - \frac{(\beta + CAf)^\beta}{(\beta + Af)^\beta} \quad (2)$$

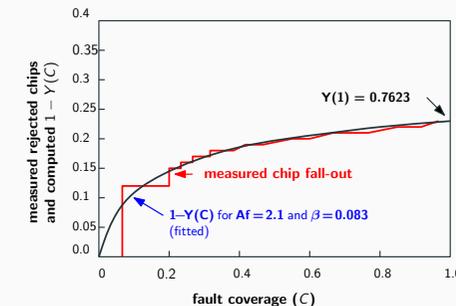
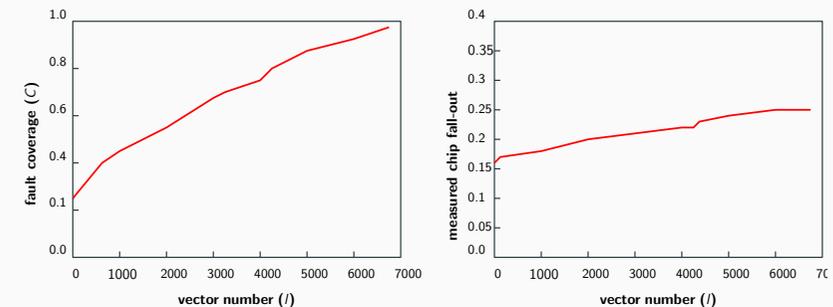
- Dove  $Af$  é il numero medio di guasti per chip e  $\beta$  é il fattore di clustering
- Entrambi i dati sono stimati a partire dai dati di collaudo

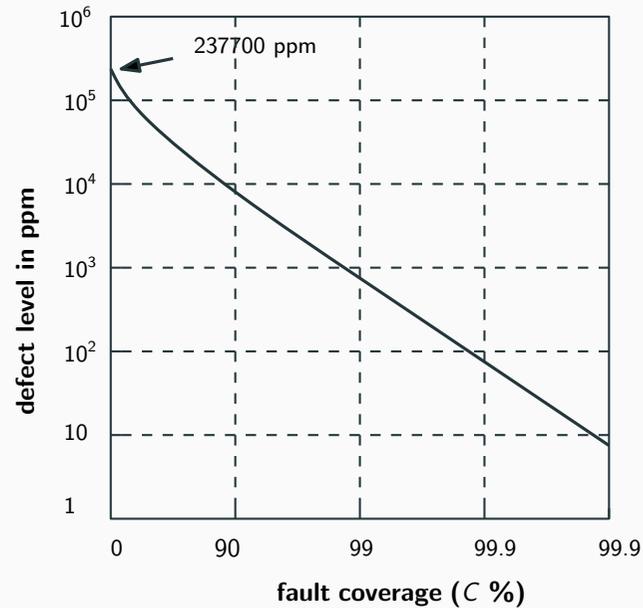
- Tramite un simulatore di guasto di esprime  $C = C(L)$  ove  $L$  é la lunghezza delle sequenza di collaudo
- Dall'osservazione del processo di collaudo, si determina la frazione di rejected chips ( $1 - Y(C)$ ) in funzione di  $L$
- Eliminando  $L$  ottengo la frazione di rejected chips in funzione di  $C$
- Utilizzando valori stimati di  $Af$  e  $\beta$  interpolo  $1 - Y(C)$  utilizzando l'Eq. 1 da cui calcolo  $Y(1)$  che utilizzata in Eq. 2 mi fornisce il  $DL$  in funzione di  $C$

## Chip Sematech

- Bus interface controller ASIC fabbricato e collaudato presso l'IBM
- 116,000 2-input NAND gate equivalenti
- 304-pin package, 249 I/O
- Clock: 40MHz, alcune parti a 50MHz
- 0.45 $\mu$ m CMOS, 3.3V, 9.4mm x 8.8mm area
- Full scan, 99.79% fault coverage
- Advantest 3381 ATE, 18,466 chips tested con un test clock a 2.5MHz

## Curve (approssimate per il chip Sematech)





- A parità di  $d$  e  $\alpha$  la resa cala con l'aumentare dell'area del chip
- La copertura di guasto misura in parte la qualità del collaudo che dipende fortemente anche da quanto i guasti sono rappresentativi dei difetti di produzione
- Il defect level misura la qualità dei chip e viene determinato a partire da analisi sui dati di collaudo
- Per avere  $DL < 500 \text{ ppm}$  servono coperture di guasto  $> 99\%$

## Modelli di guasto

## Modelli di guasto

- Motivazioni
- Difetti reali nei circuiti digitali integrati
- Modelli di guasto
- Il modello di guasto di tipo stuck-at
  - molteplicità
  - equivalenza e dominanza
  - pratiche di tipo industriale
- Guasti a livello dei transistori

- Il collaudo esaustivo delle funzionalità di I/O é troppo costoso
- I difetti reali sono troppo numerosi, difficili da analizzare e da simulare
- I modelli di guasto identificano gli obiettivi del collaudo
- I modelli di guasto rendono l'analisi e la sintesi di sequenze di test fattibili
- La loro efficacia può essere misurata mediante esperimenti

- Difetti di processo
  - contatti mancanti (missing contact)
  - fluttuazioni nei processi fotolitografici
  - transistori parassiti
  - breakdown negli ossidi (oxide breakdown) . . .
- Difetti nei materiali
  - difetti del bulk (rotture, imperfezioni nei cristalli)
  - impurità superficiali (ion migration) . . .
- Guasti dipendenti dal tempo
  - breakdown nei dielettrici
  - elettromigrazione . . .
- Guasti dovuti al packaging
  - degrado dei contatti
  - correnti di perdita (seal leaks)

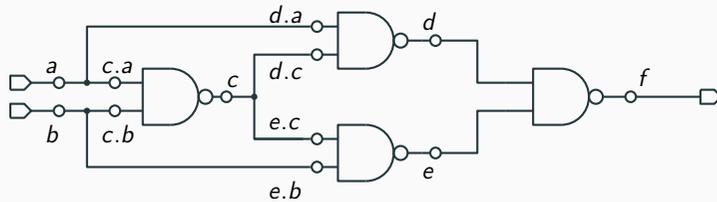
## Caratteristiche dei modelli di guasto

- Classe di difetti descritti
- Livello di descrizione del modello (elettrico, switch, gate, RTL, behavioral)
- Comportamento modificato dal guasto (funzione, timing, guasti parametrici)
- Durata del guasto (permanente, transitorio)
- Molteplicità

## Modelli di guasto

- Stuck-at singoli
- Transistor stuck-open/on
- Bridging
- Open/break
- Guasti per componenti specifici
  - guasti nelle memorie
  - guasti nelle PLA e nano-fabbriche (cross-point)
  - guasti nelle CPU
- Guasti funzionali al livello behavioral
- Guasti di tipo ritardo (transition, path, crosstalk)
- Guasti transitori

- Modello di guasto maggiormente utilizzato nell'industria
- Una linea di ingresso o uscita di un gate é bloccata a 0 o a 1
  - molteplicitá: **guasti singoli** (in seguito si vedrá il caso di guasti multipli)
  - occorrenza: **guasti permanenti**
  - livello di descrizione: **gate**
- Esempio: un gate XOR realizzato a NAND ha 12 siti di guasto e 24 possibili guasti singoli



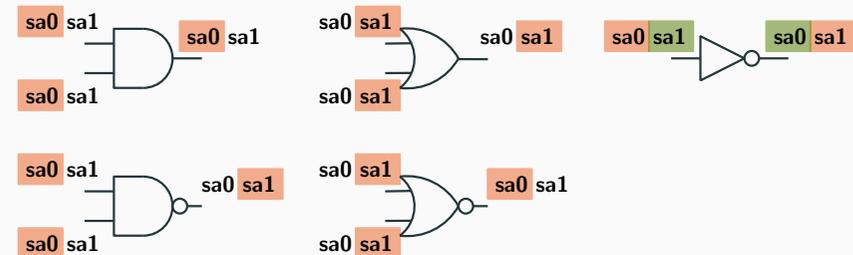
- Il numero di guasti da considerare condiziona i tempi di generazione e applicazione delle sequenze di collaudo
- Nel caso di stuck-at singoli, abbiamo:
 
$$\#guasti = 2(\#PI + \#gates + \#fan - out\ branches)$$
- L'equivalenza di guasto consente di ridurre il numero dei guasti
- **Due guasti  $f_1$  e  $f_2$  sono equivalenti se i test che rivelano  $f_1$  rivelano anche  $f_2$**
- In pratica, le funzioni della rete in presenza dei due guasti sono identiche
- **Fault collapsing**: tutti i guasti singoli di tipo stuck-at di una rete logica possono essere divisi in classi di equivalenza disgiunte, dove i guasti di una classe sono fra loro equivalenti

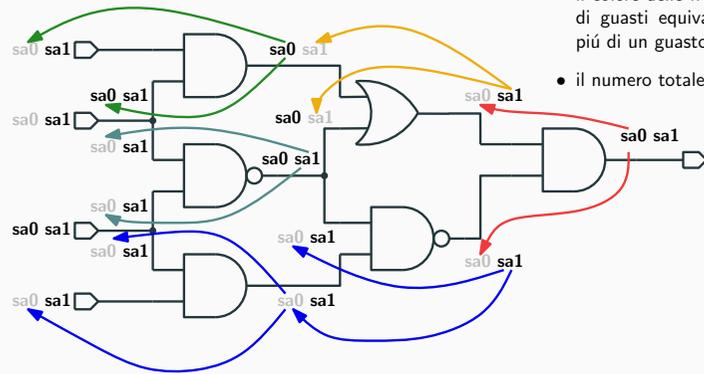
## Equivalenza di guasto: note

- L'equivalenza di guasto e il fault collapsing devono essere tenute in conto nella valutazione della copertura di guasto
- Se la classe  $A$  contiene molti piú guasti di  $B$ , allora  $A$  rappresenterá molti piú difetti di  $B$  e la mancata rivelazione dei guasti in  $A$  avrá un impatto maggiore sul defect level rispetto alla mancata rivelazione dei guasti in  $B$
- Per motivi computazionali le classi di guasti equivalenti vengono calcolate in maniera locale

## Regole di equivalenza fra i guasti

Per ciascun tipo di gate, i guasti equivalenti sono inclusi nelle aree colorate

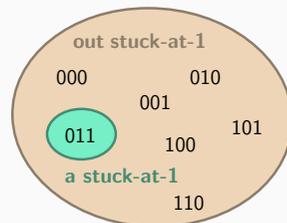
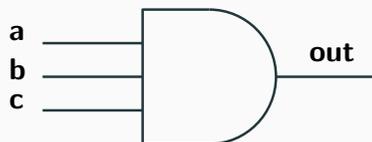




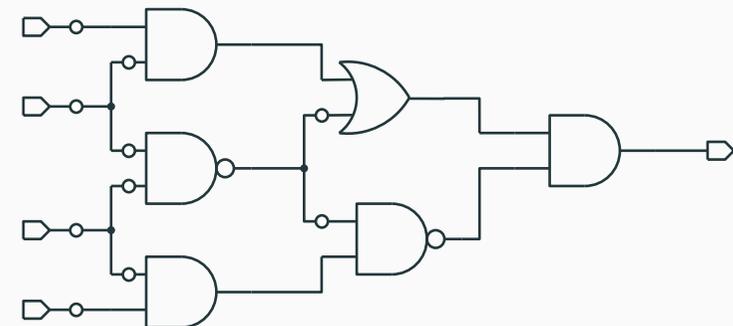
- i guasti in grigio vengono eliminati
- le frecce danno un'idea del modo in cui procede il fault collapsing
- il colore delle frecce individua le classi di guasti equivalenti che contengono più di un guasto
- il numero totale di classi è 19

- Se tutti i test che rivelano  $f_1$  rivelano anche  $f_2$ , si dice che  $f_2$  domina  $f_1$
- L'insieme di test che rivela  $f_2$  contiene l'insieme di test che rivela  $f_1$
- Se  $f_2$  domina  $f_1$ , allora può essere rimosso dalla lista perché la sua rivelazione è implicata da quella di  $f_1$
- In una rete ad albero, i guasti sugli ingressi rappresentano un insieme di guasti collassato sulla base della relazione di dominanza
- Mentre la relazione di equivalenza si utilizza durante la simulazione di guasto per determinare la copertura, la dominanza si utilizza durante la test generation

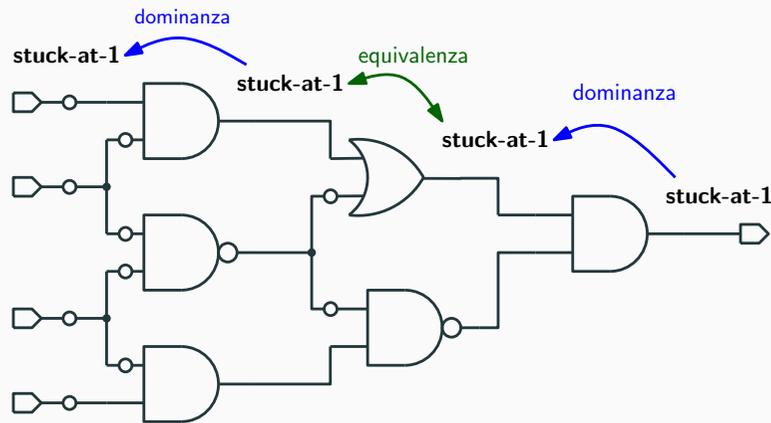
- *out* stuck-at-1 domina *a*-stuck-at-1 e quindi può essere eliminato dalla lista dei guasti
- per il gate AND a 3 ingressi si ha:
  - classi colassate con la relazione di equivalenza: {*out* stuck-at-0, *a* stuck-at-0, *b*-stuck-at-0, *c* stuck-at-0}, {*out* stuck-at-1}, {*a* stuck-at-1}, {*b* stuck-at-1}, {*c*-stuck-at-1}
  - classi colassate con la relazione di dominanza: {*out* stuck-at-0, *a* stuck-at-0, *b*-stuck-at-0, *c* stuck-at-0}, {*a* stuck-at-1}, {*b* stuck-at-1}, {*c*-stuck-at-1}



- Gli ingressi di una rete combinatoria e le fan-out branches si chiamano **checkpoint**
- **Teorema sui checkpoint:** un insieme di test che rivela tutti i guasti di tipo stuck-at singoli o multipli sui checkpoint di un circuito combinatorio, rivela anche tutti i guasti stuck-at singoli o multipli in quel circuito



## Esempio di check point



- Alcuni test possono rivelare piú guasti senza che questi siano legati da alcuna relazione di equivalenza
- $T(f_1) \cap T(f_2) \neq \emptyset \wedge T(f_1) \cap T(f_2) \neq T(f_1) \neq T(f_2)$
- Il numero finale di test in una rete combinatoria é tipicamente inferiore rispetto al numero di classi di guasto

## Copertura e classi di dominanza

- La relazione di dominanza puó essere utilizzata per ridurre la lista di guasti obiettivo per la test generation
- La copertura su questi guasti non é rappresentativa della copertura attuale sui possibili difetti
- Un guasto dominato puó non essere rivelato da una sequenza di test anche se i guasti dominanti lo sono

## Classi di stuck-at

- Classi identificate dal processo di test generation e fault simulation
  - Guasti potenzialmente rivelabili: la sequenza di collaudo produce un valore  $X$  ai PO
  - Guasti di inizializzazione: prevengono l'inizializzazione del circuito
  - Guasti iperattivi: producono molta attivitá interna senza raggiungere i PO
  - Guasti ridondanti: non esiste alcun test per tale guasto (richiede una prova formale)
  - Guasti non collaudati: non si riesce a generare un test per tali guasti (per motivi di efficienza computazionale)

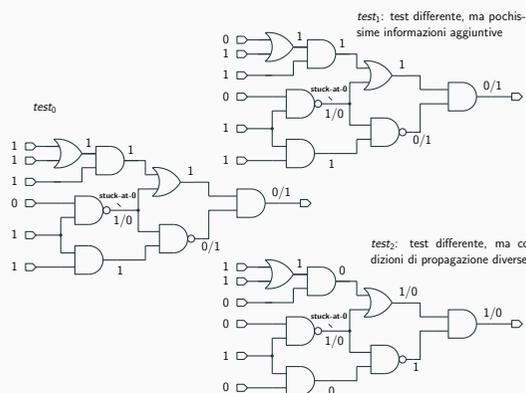
- All'aumentare della densità di integrazione dei circuiti integrati, si è scoperto che diversi difetti non sono correttamente descritti dal modello di guasto di tipo stuck-at
- Per tenere conto di tali difetti, conviene utilizzare nuovi modelli di guasto
- In alternativa si può utilizzare il modello di guasto di tipo stuck-at richiedendo che l'insieme di vettori di test contenga riveli *n* vettori che lo rivelino
- L'obiettivo è quello di esplorare le condizioni locali che consentono la rivelazione del guasto
- Sono necessarie opportune metriche che valutino quanto sono diversi due vettori di test che lo rivelino

- Combinazioni di linee che si trovano contemporaneamente bloccate a 1 o 0
- Se si hanno *k* siti di guasti singoli, il numero di guasti multipli è  $3^k - 1$  (un sito può essere s-at-0, s-at-1 o fault-free)
- Anche se in alcuni casi gli effetti dei guasti interagiscono in modo tale da cancellarsi, un test che ha una buona copertura sui guasti singoli avrà una buona copertura sui guasti multipli
- Recentemente è stato provato formalmente che un piccolo (0-15) numero di test aggiuntivi consente di coprire tutti i guasti multipli

***n*-detect stuck-at fault**

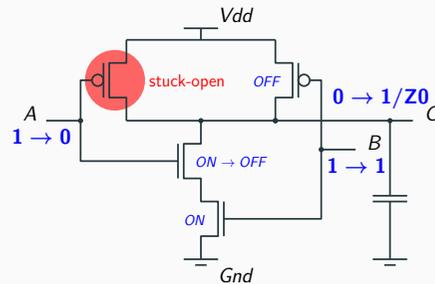
**Transistor faults**

Un insieme di 3 test diversi che rivelano lo stesso guasto, in cui due sono però equivalenti dal punto di vista delle condizioni locali



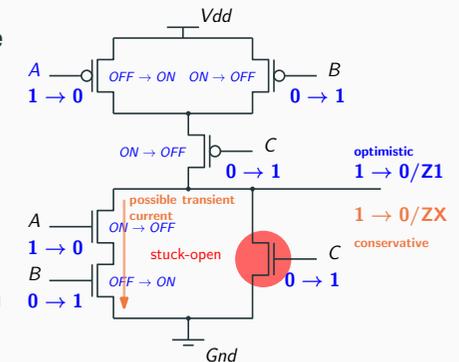
- Guasti che corrispondono al livello switch dei transistori MOS
  - **transistor stuck-on:** il transistor conduce indipendentemente dal valore di tensione di gate applicata
    - difetti: bridging fra source e drain
  - **transistor stuck-open:** il transistor non conduce indipendentemente dalla tensione di gate applicata
    - difetti: contatto di source o drain mancante
- Questi modelli di guasto possono essere gestiti sia a livello switch che a livello gate

- Attivando il guasto con un singolo vettore di test si produce  $C = Z$  invece di  $C = 1$
- Il valore di tensione dell'uscita dipende dalla carica immagazinata dalla capacità di uscita (il guasto introduce memoria)
- Per produrre un errore, é necessario applicare un vettore di inizializzazione che porti l'uscita a 0
- La sequenza di due vettori di test produce l'uscita in uno stato Z, ma con tensione a 0 ( $Z_0$ )



- Quando viene applicato un test con due pattern, le commutazioni degli ingressi di un gate non avvengono istantaneamente
- É possibile che temporaneamente si aprano cammini conduttivi che rimuovo la carica di inizializzazione mascherando cosí gli effetti del guasto

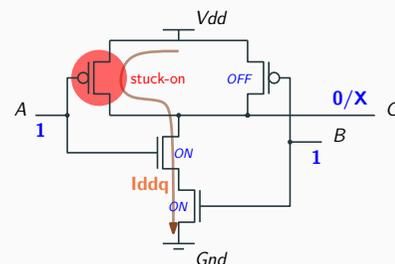
- Un test che non può avere tale problema si dice "robusto" e "non-robusto" nel caso in cui tale problema sia potenzialmente presente
- In un test "non-robusto" il dispositivo può passare il collaudo anche se contiene un guasto



## Transistor stuck-on

## Collaudo basato sull'assorbimento di corrente statica

- Attivando il guasto si produce un uscita connessa sia a  $Vdd$  che a  $Gnd$
- Il valore di tensione dell'uscita dipende dal conflitto fra la conduttanza della rete di pull-down correttamente ON e quella di pull-up ON a causa del guasto
- Per produrre un errore, é necessario che tale tensione sia interpretata come alta dai gate nel fan-out ( $X1$ )
- Il guasto produce anche assorbimento di corrente statica



- Le logiche complementari non assorbono corrente statica in condizioni ideali
- Alcuni guasti quali i transistor stuck-on e i bridging provocano assorbimento di corrente statica
- La cosa viene sfruttata dal collaudo basato sulla misura di corrente statica che può anche sfruttare sensori integrati nel circuito
- Le tecnologie attuali hanno però un forte assorbimento di corrente statica che può rendere problematica questa tipologia di collaudo

- I modelli di guasto rappresentano l'astrazione del comportamento indotto da diverse tipologie di difetti
- Devono soddisfare obiettivi di accuratezza e di efficienza computazionale
- Tipicamente operano al livello logico
- Rimangono da vedere due classi molto rilevanti di modelli di guasto: i bridging e i delay fault