



UNIVERSITÀ DELLA CALABRIA

DIPARTIMENTO DI
INGEGNERIA INFORMATICA,
MODELLISTICA, ELETTRONICA
E SISTEMISTICA

DIMES

Report Tecnico n.2

MARGINE DI RUMORE STATICO IN UNA CELLA DI MEMORIA SRAM 6-T

Corso di Progettazione Low Power

Corso di Laurea Magistrale in Ingegneria Elettronica

Unical, aa 2020/2021

Andrea Alecce Matricola 214611

Prof. F. Frustaci

Sommario

- 1. Intro 3
- 2. Margine di rumore in fase di Hold 3
 - 2.1. Metodo grafico..... 3
 - 2.2. Metodo Seevinck 5
 - 2.3. Corrente di Leakage 11
- 3. Margine di rumore in fase di Read 15

1. Intro

Si effettuano delle simulazioni per misurare il margine di rumore statico SNM in fase di Hold e in fase di Read.

Tali misure verranno effettuate sia in modo *manuale* che in modo *automatico*:

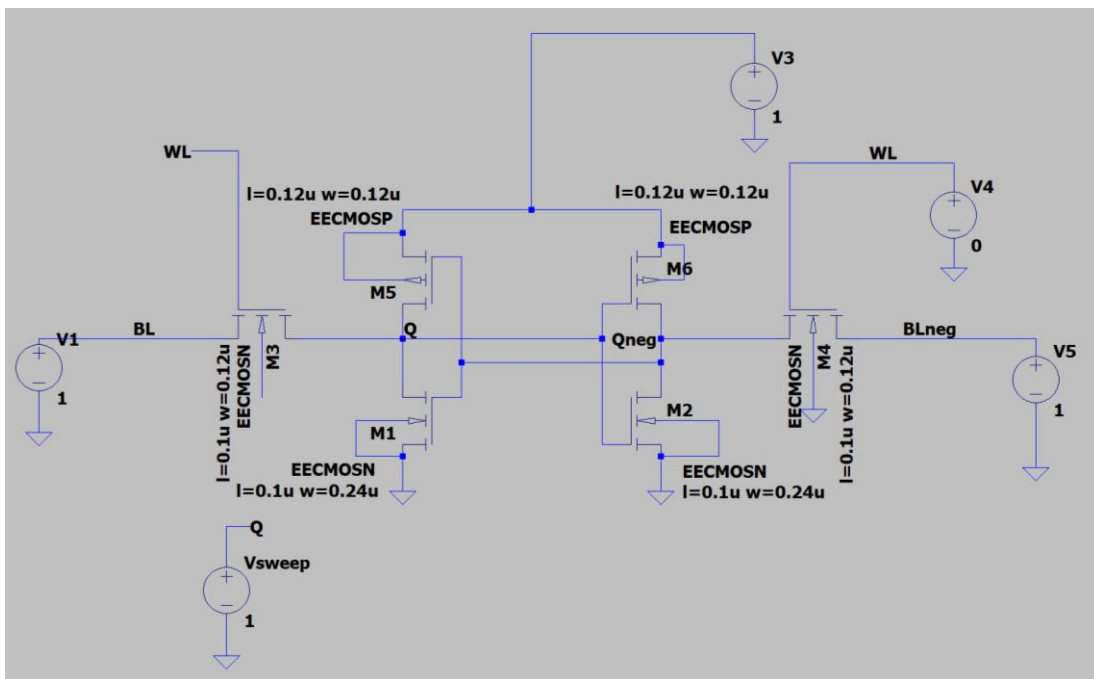
- In manuale, tramite simulazione DC variando la tensione su VQ e valutando quella su VQ_negato ottenendo il famoso grafico a farfalla, valutando la dimensione del lato del quadrato inscritto nel lobo più piccolo.
- In automatico, implementando quanto descritto nell'articolo di Seevinck & co "*Static-noise margin analysis of MOS SRAM cells*", metodo che consiste nel simulare il circuito non come se fosse solo una cella ma inserendo nel circuito altri elementi quali generatori di tensione controllati in tensione.

Si procede con la prima misura tramite metodo grafico.

2. Margine di rumore in fase di Hold

2.1. Metodo grafico

Si realizza il circuito in LTSpice.



Siccome si analizza la fase di Hold, si pone la Word Line a zero, in quanto non si accede alla cella. I due inverter vengono alimentati con una alimentazione di valore nominale pari a 1V. Per le Bit Line, la condizione di Hold presume che le Bit Line siano precaricate ad un valore prestabilito, generalmente VDD. Alcune tecniche low power consistono nel precaricare le Bit Line a valori più bassi di VDD. Nel caso in esame, supponiamo siano precaricate a VDD=1V.

Al fine di rendere simmetrico il grafico a farfalla, è necessario che le dimensioni dei transistor siano in scala 1:2 (PMOS più grande), in modo da garantire la stessa conducibilità. In realtà, i PMOS di pull-up sono dimensionati sempre al minimo perché il loro unico scopo è quello di mantenere il nodo intermedio a VDD, non intervenendo nelle performance di timing, generalmente riferite alla lettura (operazione che offre il

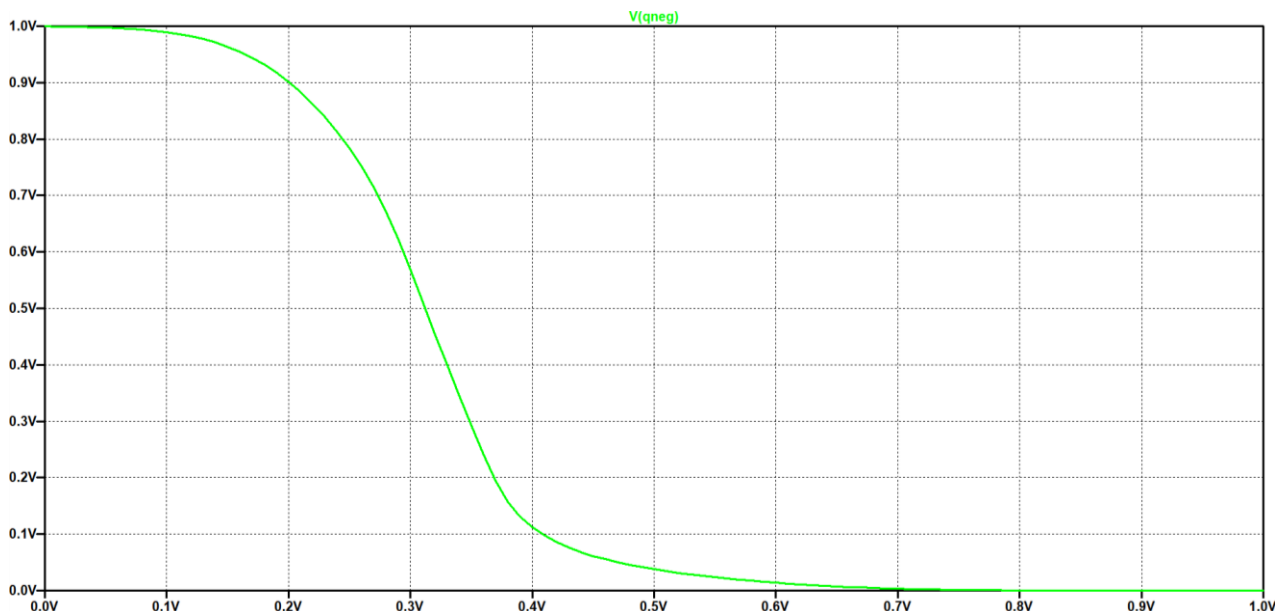
ritardo più critico). Il ritardo in lettura è infatti funzione del path di scarica della Bit Line, ovvero tramite il transistor di accesso e l'NMOS di pull-down. Per tale motivo nel circuito i PMOS sono dimensionati a 0.12u.

Come detto, la simulazione è di tipo DC, variando dunque il valore di un generatore di tensione costante. Si suppone che nei risultati ottenuti da questa simulazione i transistori siano tutti esauriti. Il generatore costante usato nel circuito è nominato **Vsweep**. Questo generatore è collegato direttamente al nodo **Q**, tramite *net label*. Ciò che voglio ottenere è la caratteristica del potenziale su **Qneg** al variare proprio di **Q**. La direttiva Spice relativa a tale simulazione sarà:

.dc Vsweep 0 1 0.01

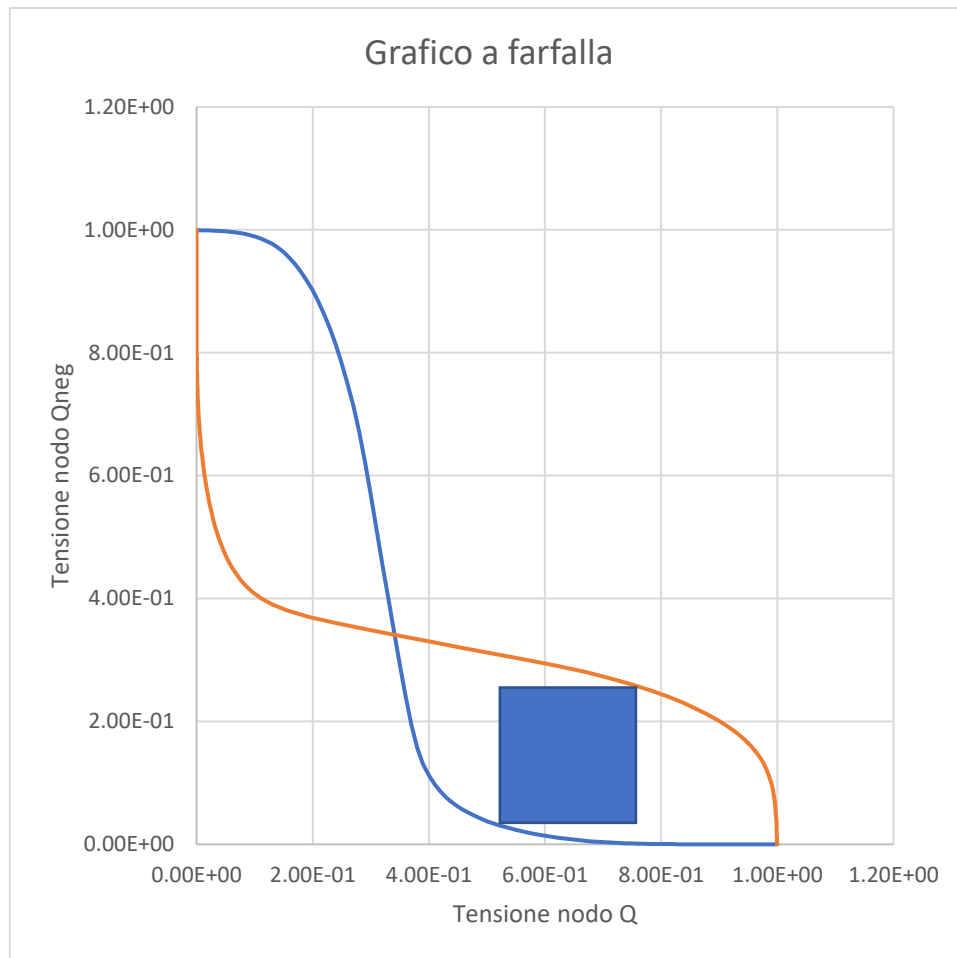
Dove viene indicato il nome del generatore da far variare, il valore di partenza, il valore finale e il passo. Per come definita, verranno effettuate 100 simulazioni. Si include anche il file di modello **RIT_Models_For_LTSPICE.txt**, come per la precedente relazione.

Avviata e conclusa la simulazione, si ricavano i grafici della tensione su **Qneg** rispetto alla variazione di **Q**.



Si nota il fatto che il grafico sia fortemente non simmetrico: se così fosse, dovrebbe passare esattamente per 0.5V. Questo perché l'NMOS è il doppio del PMOS in termini di dimensionamento, contrariamente a quanto detto prima.

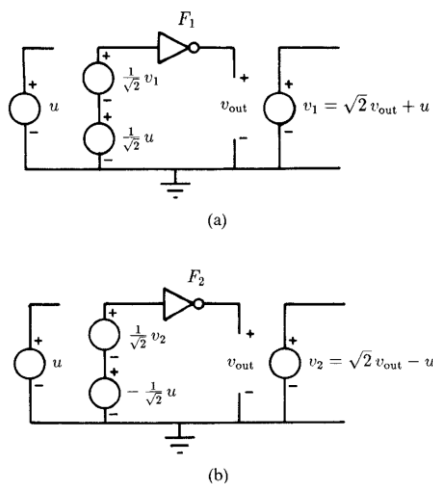
Prodotto il grafico su LTSpice, è necessario esportare i valori che caratterizzano la curva per importarli su Excel (o simili). Per esportare la forma d'onda in formato numerico, si procede tramite l'opzione di LTSpice *File/Export data as text*, selezionando quando richiesto il segnale di interesse. Il grafico che si ottiene su Excell è riportato in figura.



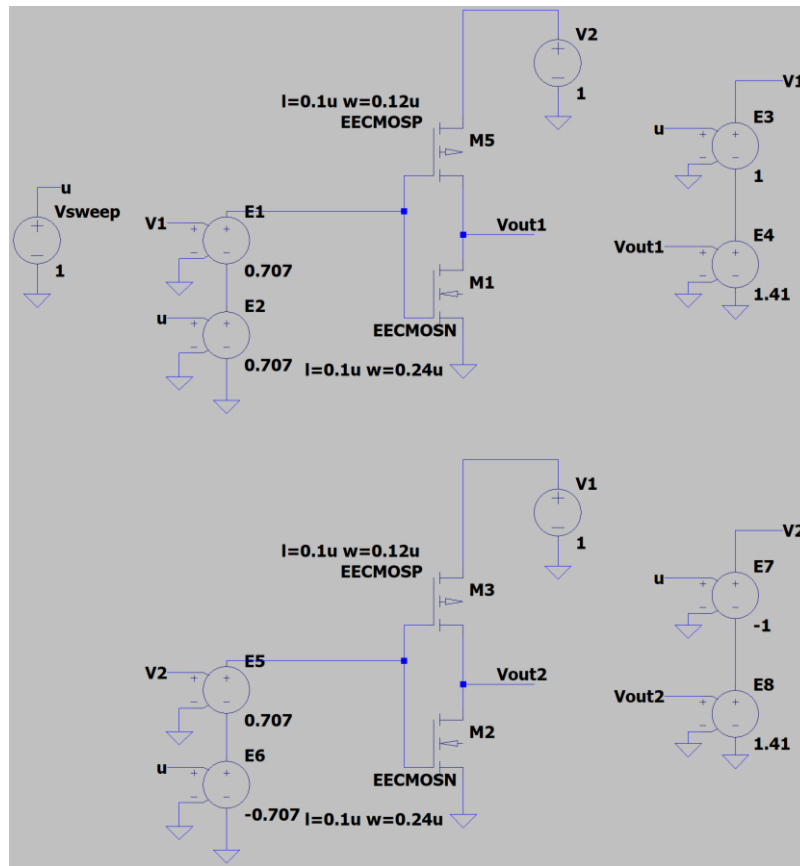
L'ulteriore curva si ottiene invertendo l'asse dei valori tra ascissa e ordinata. Si evidenzia la necessità di ottenere un grafico con valori degli assi equamente distribuiti, a fronte della misura grafica che verrà effettuata. Infine, si iscrive un quadrato nel lobo più piccolo. Nel caso in esame, il lato di questo quadrato è pari a 220 mV, valore del margine di rumore in fase di hold.

2.2. Metodo Seevinck

Si vuole utilizzare il metodo descritto nell'articolo di Seevinck & co. Senza scendere troppo nei dettagli dell'articolo, esso illustra uno schema circuitale (riportato in figura) utile a ricavare in maniera automatica il margine di rumore.



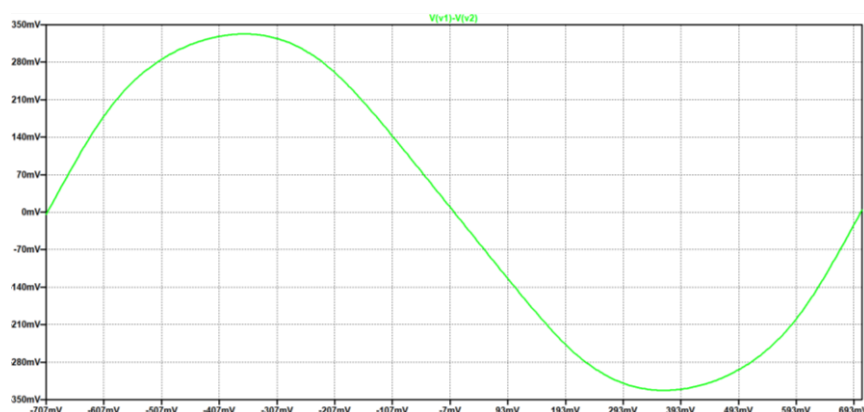
I circuiti sono due, formati da due inverter della cella di memoria F1 e F2, con le stesse dimensioni usate precedentemente. Inoltre, si nota la presenza di alcuni generatori di tensione controllati in tensione (elementi ideali). Si richiede di riprodurre un circuito di questo tipo su LTSpice come riportato in figura, per poi simularlo.



Si assume il generatore “u” in figura come generatore V_{sweep} nella simulazione, grazie al quale si otterrà il valore in tensione $V1$ e $V2$. Come definito nell’articolo, considerate le differenze tra $V1$ e $V2$ simulati, prendendone il valore assoluto del massimo e minimo e moltiplicandoli per $1/\sqrt{2}$ si ottiene proprio il margine di rumore SNM.

Ciò che il circuito fa è lo shifting della curva che è stato precedentemente fatto su Excel per la precedente misura. Come definito nell’articolo, è necessario effettuare una simulazione DC variando V_{sweep} in un range ben specifico, ovvero $\left[-\frac{\sqrt{2}}{2}V_{dd}, \frac{\sqrt{2}}{2}V_{dd}\right]$.

Si avvia la simulazione e si grafica esattamente la differenza tra $V1$ e $V2$, ottenendo la seguente figura:



Dal quale bisogna estrapolare i valori di massimo e minimo, pari a circa 300mV e poi implementare le operazioni descritte nell'articolo. Per agevolare questi passaggi, si utilizza un semplice script *measSNM1V.txt* richiamabile da LTSpice tramite *File/Execute .MEAS script*. Il contenuto del file è il seguente:

```
.measure dc SNMMAX MAX(0.707*V(v1)-0.707*V(v2))  
.measure dc SNMMIN MIN(0.707*V(v1)-0.707*V(v2))
```

I risultati ottenuti sono:

```
.Measure Output File: D:\UNI\Low_Power\Materiale_Corso\measSNM1V.mout  
* D:\UNI\Low_Power\Materiale_Corso\measSNM1V.mout  
SNMMAX: MAX(0.707*V(v1)-0.707*V(v2))=0.235205 FROM -0.707 TO 0.707  
SNMMIN: MIN(0.707*V(v1)-0.707*V(v2))=-0.235222 FROM -0.707 TO 0.707
```

Ovvero 0.235222 e -0.235205. Considerando il valore assoluto di entrambi e prendendo il minimo ottengo il margine di rumore pari a circa 235mV, valore paragonabile a quello ottenuto tramite metodo grafico.

Questo metodo risulta utile in quanto i margini di rumore sono suscettibili alle variazioni di processo. Occorre quindi fare diverse simulazioni (migliaia o milioni) per caratterizzare statisticamente il margine di rumore. A tal fine, si effettuano delle simulazioni MonteCarlo andando a prevedere delle variazioni random su alcuni parametri di processo (ad esempio, la V_{th}), valutando la distribuzione statistica dei valori del margine di rumore. Un modello che tiene conto delle variazioni di processo (generalmente fornito dal produttore), può essere usato per le simulazioni. Nel caso in esame, si usa il file *RIT_Models_For_LTSPICE_MonteCarlo.txt*, in cui sono stati cambiati arbitrariamente i valori di V_{th} per NMOS e PMOS. In particolare:

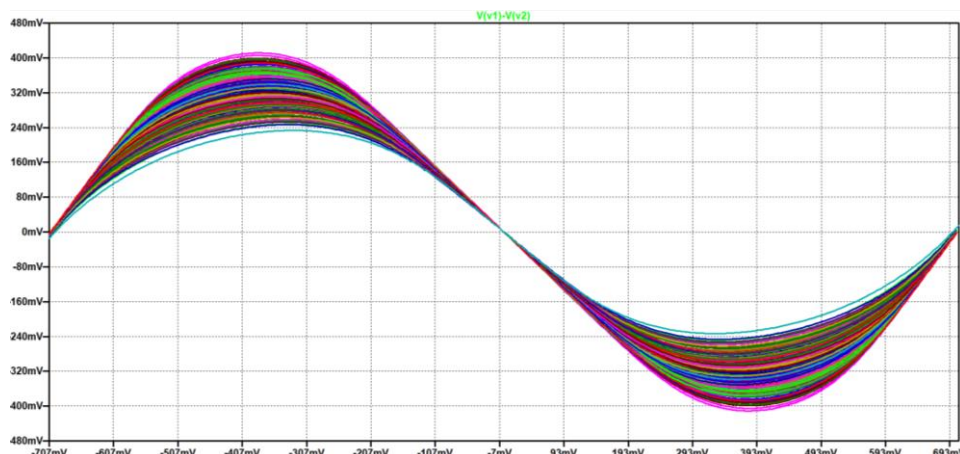
- $V_{TH0}=\{0.4+\text{gauss}(0.04)\}$ per NMOS
- $V_{TH0}=\{-1*(0.4+\text{gauss}(0.04))\}$ per PMOS

Dove il valore effettivo sarà random, ottenuto da una distribuzione di probabilità di tipo gaussiano, con valor medio pari a 0.4 e deviazione standard pari al 10% (0.04). Stessa cosa per il PMOS.

Si procede con la simulazione, utilizzando dunque questo modello per i Mosfet. Si inserisce inoltre la direttiva per indicare di effettuare diverse simulazioni parametriche, tramite:

```
.step param run 1 400 1
```

Dove si effettuano 400 simulazioni (valore basso, ma utile ai fini didattici), con il parametro preso direttamente da modello. Si effettua nuovamente la simulazione, attendendo fino alla numero 400. Si ottiene il seguente grafico, dove sono visibili le 400 curve ottenute dai diversi valori di V_{th} generati casualmente.



Applicando nuovamente lo script di prima, questa volta si ottengono 400 valori per il margine massimo e altrettanti per il minimo, come riportato di seguito in parte:

Measure Output File: D:\UNI\Low_Power\Materiale_Corso\measSNM1V.mout

* D:\UNI\Low_Power\Materiale_Corso\measSNM1V.mout

Measurement: SNMMAX

step	MAX (0.707*V (v1) -0.707*V (v2))	FROM
1	0.245128	-0.707
2	0.250799	-0.707
3	0.240288	-0.707
4	0.228048	-0.707
5	0.235143	-0.707
6	0.274294	-0.707
7	0.18793	-0.707
8	0.250595	-0.707
9	0.244912	-0.707
10	0.199505	-0.707
11	0.248169	-0.707
12	0.212914	-0.707
13	0.226738	-0.707
14	0.273874	-0.707
15	0.202331	-0.707
16	0.248908	-0.707
17	0.198328	-0.707
18	0.192078	-0.707
19	0.262429	-0.707
20	0.219771	-0.707
21	0.21873	-0.707
22	0.211507	-0.707
23	0.244895	-0.707
24	0.23531	-0.707
25	0.223176	-0.707
26	0.261537	-0.707
27	0.226117	-0.707
28	0.214368	-0.707
29	0.182881	-0.707
30	0.165067	-0.707

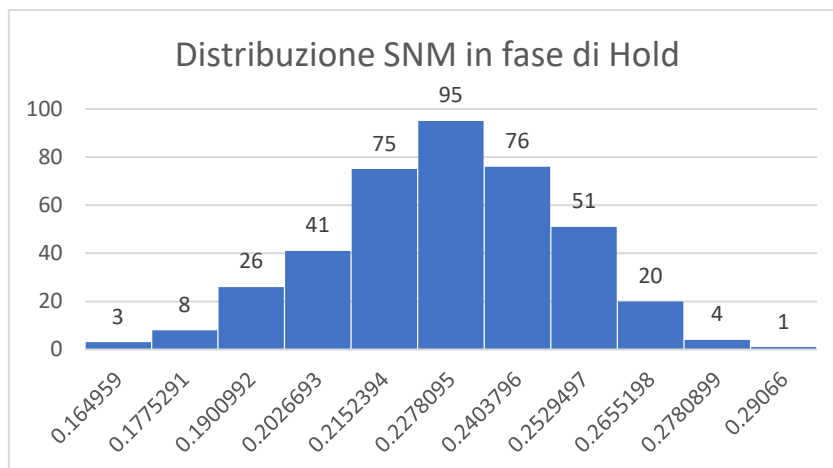
Si vuole determinare che tipo di distribuzione ha assunto il margine di rumore. Anche in questo caso, si utilizza Excell. Si ricava il valore assoluto e si seleziona il minimo, per ogni simulazione. Il risultato è un vettore di valori di margini di rumore, associati ad ogni singola simulazione.

Partendo dall'ipotesi che la distribuzione associata alla V_{th} è di tipo gaussiano, si potrebbe supporre che anche la distribuzione del margine di rumore sia di tipo gaussiano. Si dimostra successivamente che ciò non è sempre valido, soprattutto al diminuire della V_{dd} . In particolare, si dimostrerà che per valori di V_{DD} vicini a quelli di V_{th} , la distribuzione assumerà una forma log-normale.

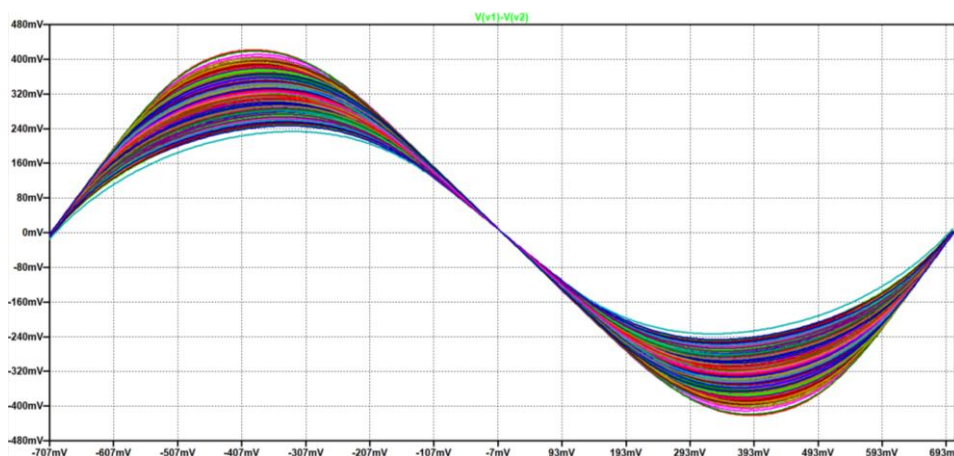
Tornando alla simulazione attuale, si richiede di graficare l'andamento di questo vettore, per valutare che forma assume la distribuzione. Si considerano dunque come estremi dell'intervallo il valore massimo e minimo, che sono rispettivamente 0.29066 e 0.164959. Si divide l'intervallo in un numero arbitrario intero di sottointervalli, ad esempio 10, di valore Δx . Il grafico si ottiene associando ad ogni intervallo il numero di simulazioni il cui valore ricade in quell'intervallo: si riporta una tabella con tali valori.

intervalli	occorrenze
0.164959	3
0.1775291	8
0.1900992	26
0.2026693	41
0.2152394	75
0.2278095	95
0.2403796	76
0.2529497	51
0.2655198	20
0.2780899	4
0.29066	1

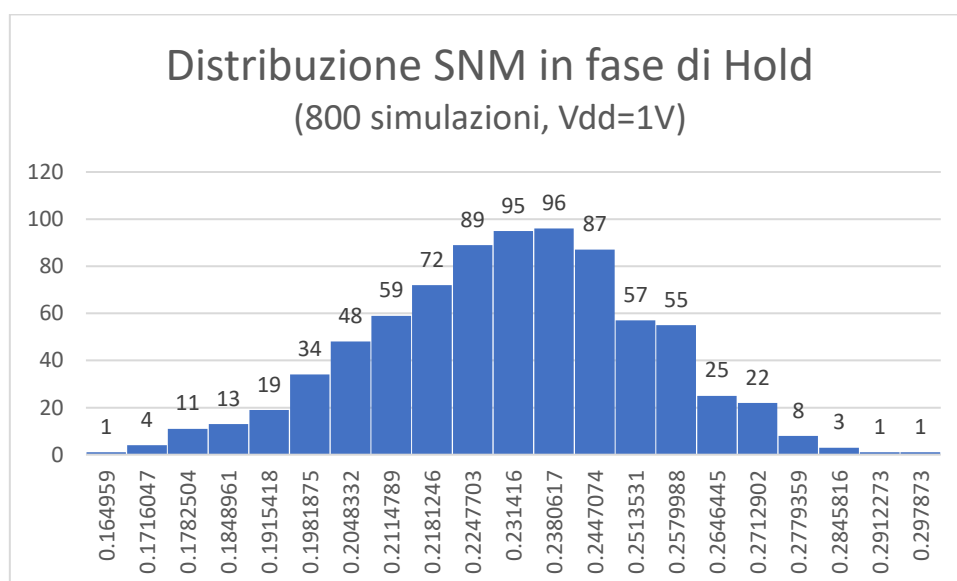
Graficando questa tabella in formato istogramma si ottiene il grafico della distribuzione del margine di rumore in fase di Hold.



Si vuole effettuare la stessa misura, con stesso circuito e parametri, iterando però 800 simulazioni. Si ottiene la seguente figura, nettamente più fitta di curve.



Riportando i dati su Excel si ottiene il seguente grafico:

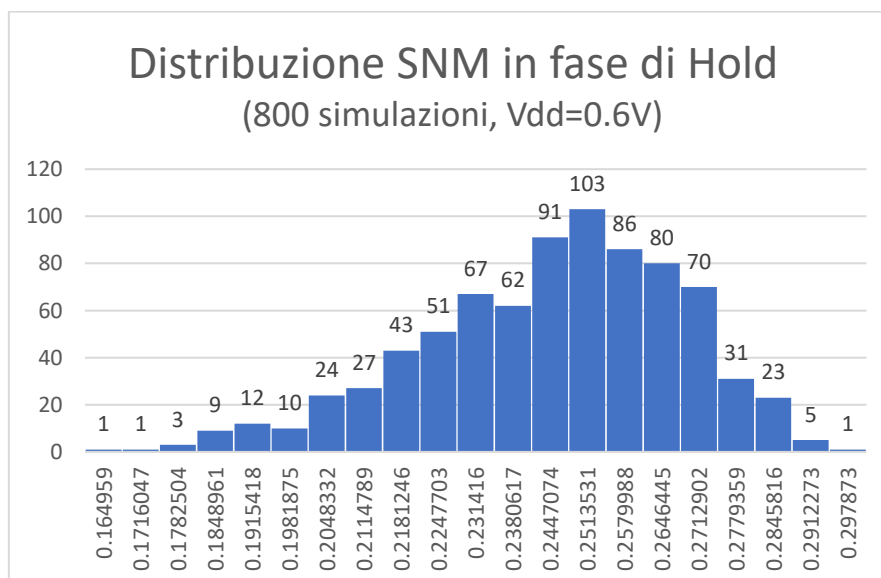
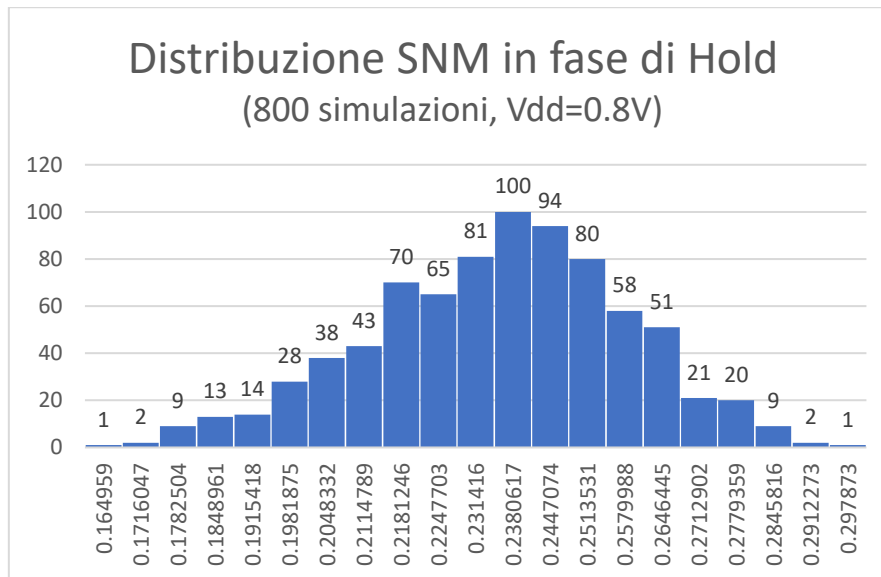


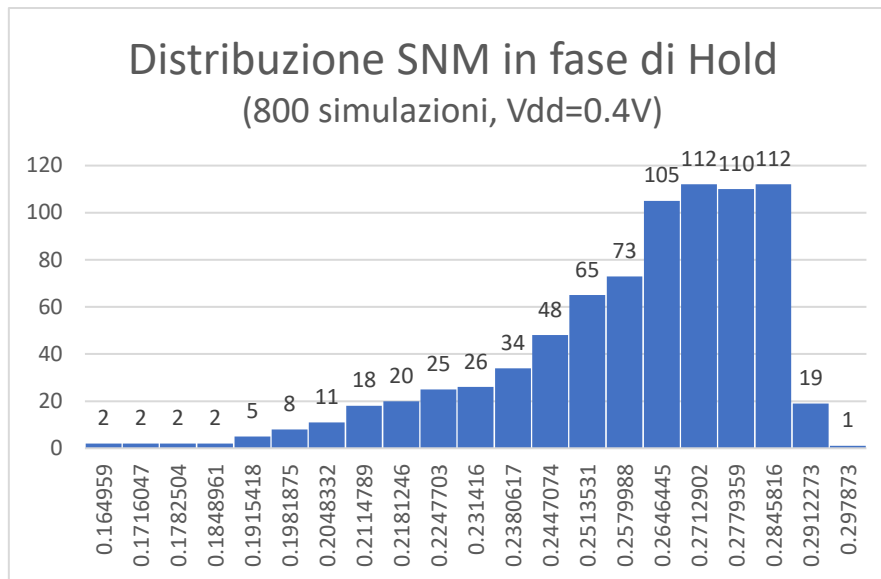
Come anticipato e confermato dai grafici, aver introdotto una distribuzione di tipo gaussiano per la V_{th} ha comportato una forma della distribuzione del SNM anch'esso di tipo gaussiano. Ciò è spiegabile in quanto il margine di rumore dipende dalle correnti presenti nella cella. Se $V_{dd} \gg V_{th}$, siamo in regime di *sopra soglia*,

quindi la corrente sarà proporzionale a $(V_{dd} - V_{th})^\alpha$, da cui il legame lineare tra V_{th} e la corrente, e corrente con il margine di rumore.

Se $V_{dd} \cong V_{th}$, siamo in regime di *near threshold*, quindi la corrente sarà proporzionale a $e^{-V_{th}}$, per cui si perde il legame lineare e quindi la forma gaussiana assunta dalla V_{th} . In particolare, l'esponentiale di una funzione gaussiana si definisce *Log-normale*: ciò significa che facendone il logaritmo, si otterrà una distribuzione gaussiana. La forma grafica di una distribuzione log-normale è sempre a campana, ma sbilanciata verso uno lobo dei lobi.

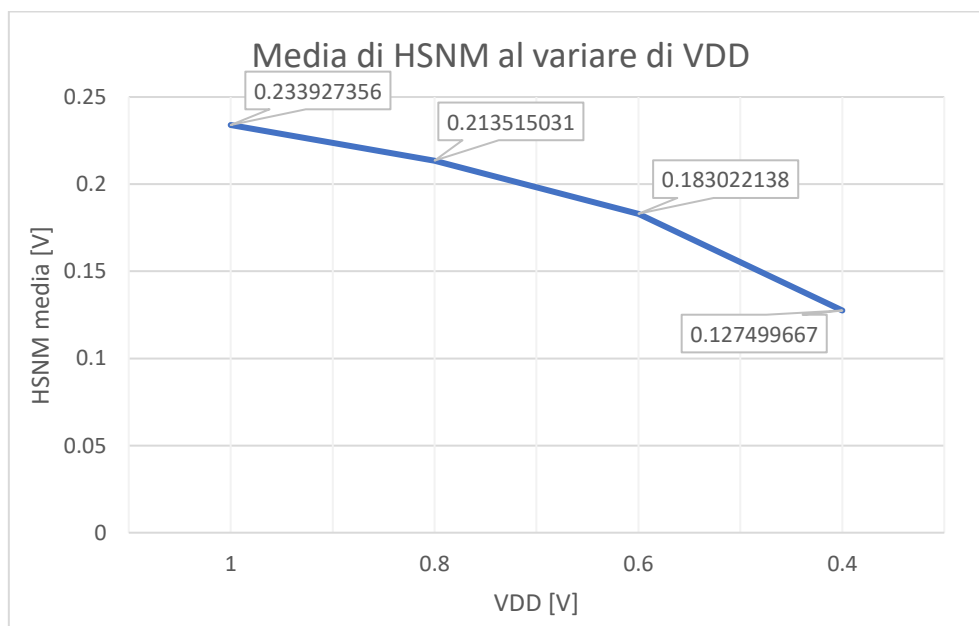
Al fine di verificare questo comportamento, si simulano e graficano gli andamenti del margine di rumore statico in fase di Hold al diminuire della V_{dd} e modificando opportunamente lo swing della tensione u.





Di riportano i risultati ottenuti nella seguente tabella riepilogativa:

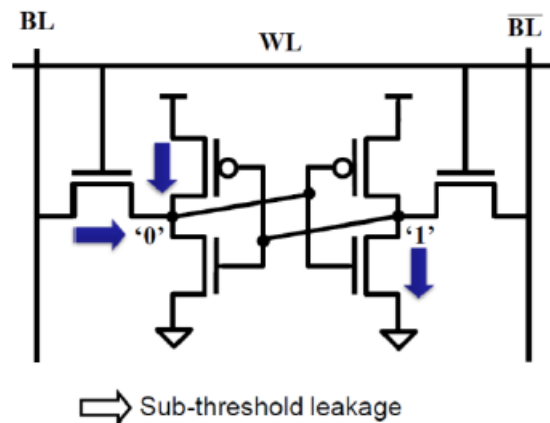
VDD	media	max	min
1	0.233927	0.297873	0.164959
0.8	0.213515	0.267353	0.146101
0.6	0.183022	0.220332	0.120502
0.4	0.1275	0.147353	0.072273



Come previsto, al diminuire di VDD fino a raggiungere valori prossimi a V_{th} , la curva di distribuzione del HSNM tende ad assumere una distribuzione log-normale. Inoltre, il valor medio del HSNM decresce al diminuire della Vdd.

2.3. Corrente di Leakage

Il leakage complessivo è costituito sia dal leakage della cella che da quello dei circuiti periferici (decode, sense amplifier etc.). Si analizza la potenza statica dissipata per leakage della cella in figura.

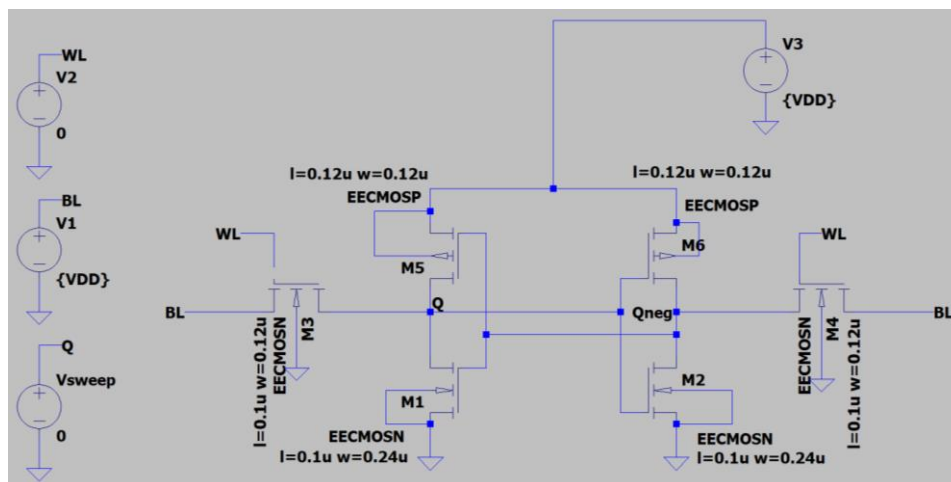


Si considera la cella con $Q=0$ e $Q_{neg}=1$. In queste condizioni, nello stato di Hold con $BL=VDD$ e $WL=0$, le principali correnti di leakage sono le seguenti (evidenziate anche in figura):

- Corrente di sottosoglia che dalla BL precaricata a 1 scarica verso il nodo Q a potenziale 0 (transistor di accesso);
- Leakage del NMOS con VDS non nulla;
- Leakage del PMOS con VDS non nulla.

Ci sarebbero anche le correnti di gate, ma di valore molto inferiore a quelle elencate e per cui trascurabili.

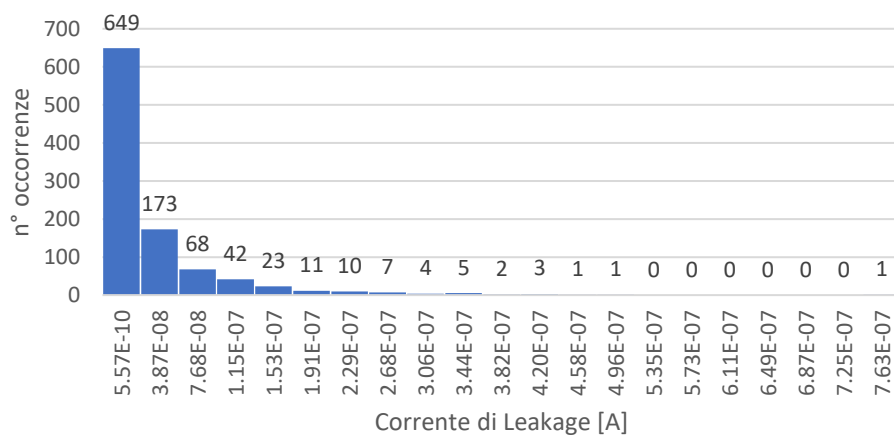
Si realizza dunque il circuito su LTSpice:



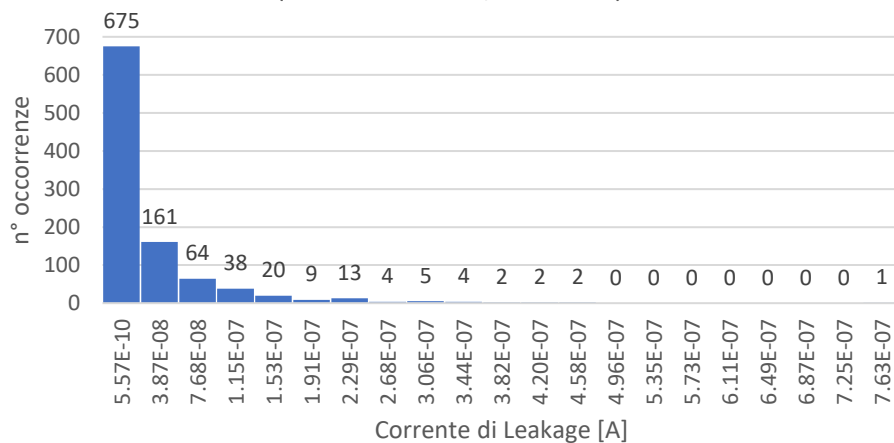
Da notare la presenza dei transistor di accesso, connessi alle BL precaricate a VDD, mentre le WL sono a 0. Vengono utilizzati gli stessi modelli e parametri del transistor usati in precedenza e viene avviata la sequenza di 1000 simulazioni. Dalle simulazioni, si estrapolano le correnti di leakage provenienti dalla BL e dall'alimentazione degli inverter, rispettivamente $I(V1)$ e $I(V3)$.

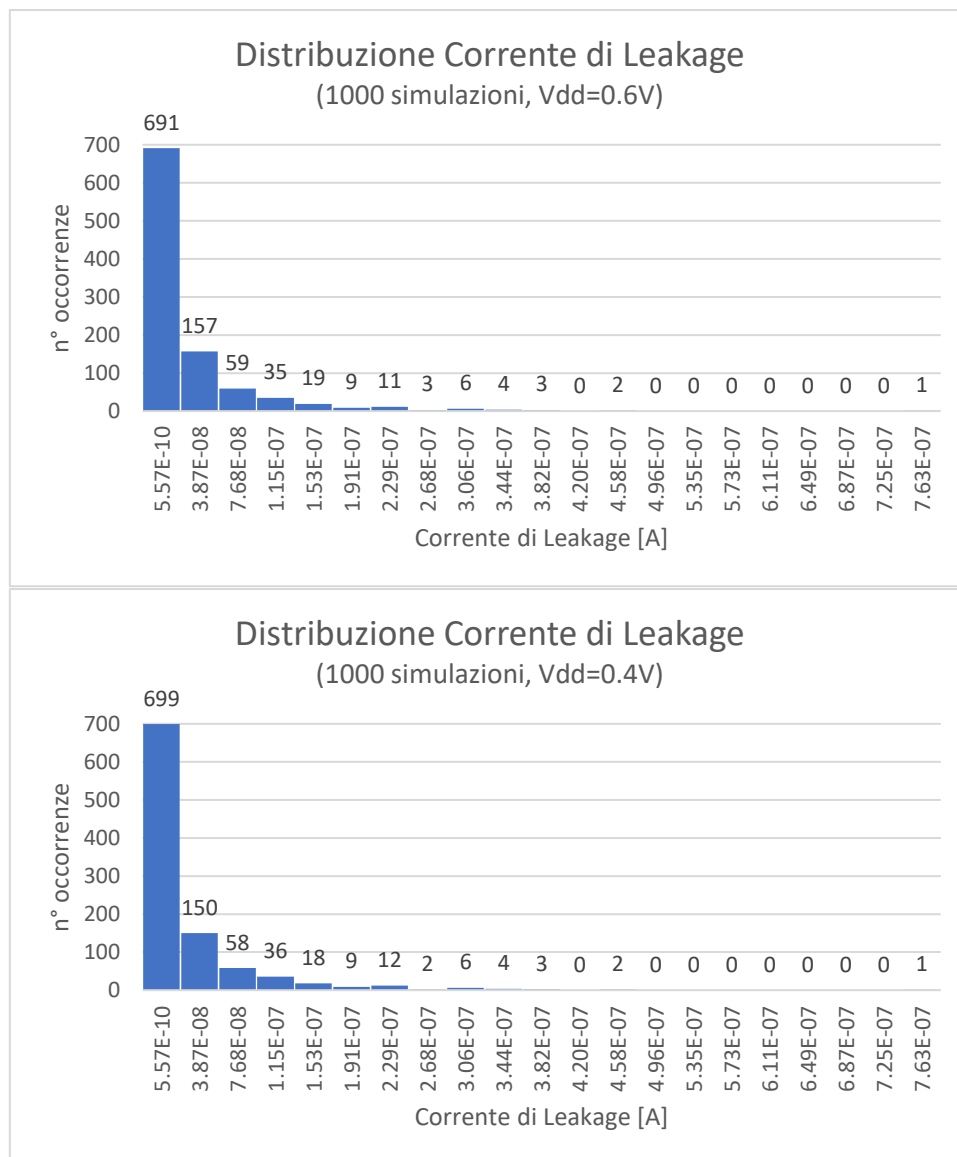
Si ripetono le simulazioni diminuendo il valore della VDD e acquisendo i dati, generando i seguenti grafici:

Distribuzione Corrente di Leakage
(1000 simulazioni, Vdd=1V)



Distribuzione Corrente di Leakage
(1000 simulazioni, Vdd=0.8V)

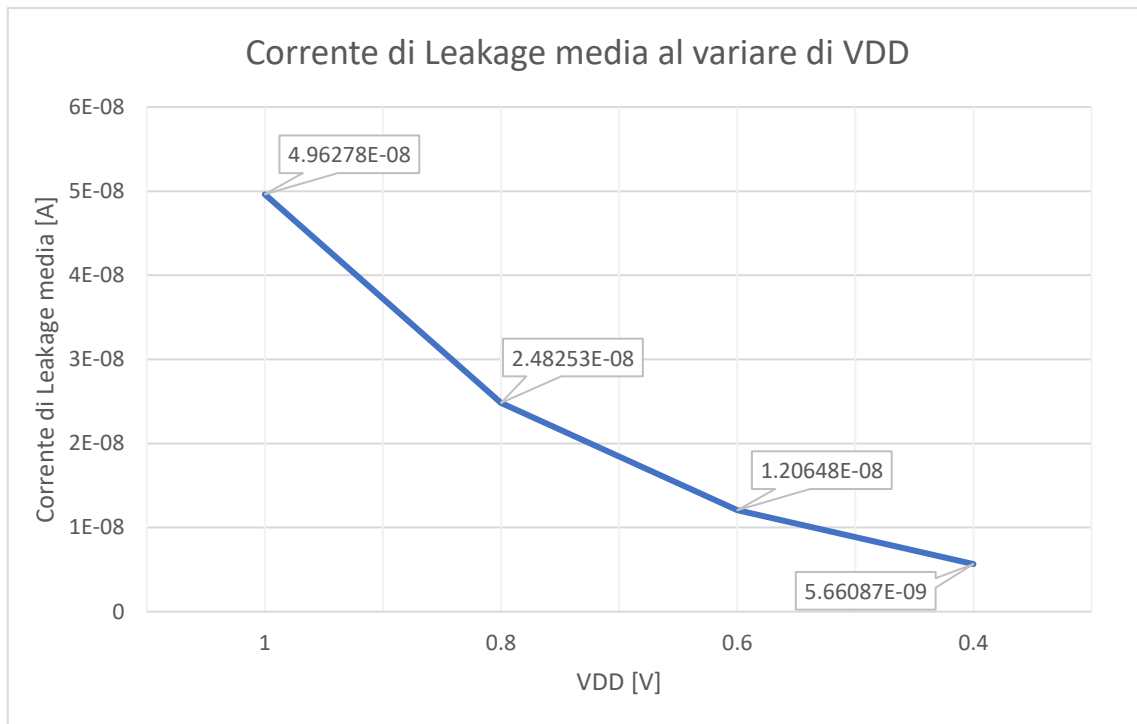




Si riporta la seguente tabella riepilogativa dei dati elaborati nelle diverse simulazioni

VDD	media	max	min
1	4.96E-08	7.63E-07	5.57E-10
0.8	2.48E-08	4.1E-07	2.8E-10
0.6	1.21E-08	2.1E-07	1.41E-10
0.4	5.66E-09	1.01E-07	7.04E-11

Dalla quale si ricava il grafico in figura:



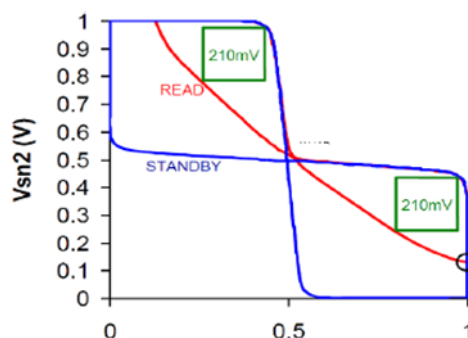
Dai dati ricavati si evince la dipendenza della corrente di leakage dalla VDD: in particolare, al diminuire della tensione di alimentazione si avrà una corrispettiva diminuzione del leakage, come visibile in figura. Inoltre, il legame tra corrente di leakage e tensione è di tipo esponenziale, motivo per il quale i grafici risultano di tipo log-normale.

Dai risultati delle due misure si evidenzia che la tensione di alimentazione impatta positivamente sulla corrente di leakage, diminuendola, ma d'altra parte degrada il margine di rumore della cella, risultando meno robusta.

3. Margine di rumore in fase di Read

Si è analizzata la cella in fase di Hold, dove viene "isolata" dall'esterno spegnendo i transistor di accesso, per mantenere il dato memorizzato. In fase di Read non è così, in quanto si pone $WL=VDD$ ed entrambe le BLs vengono precaricate a Vdd e successivamente lasciate floating.

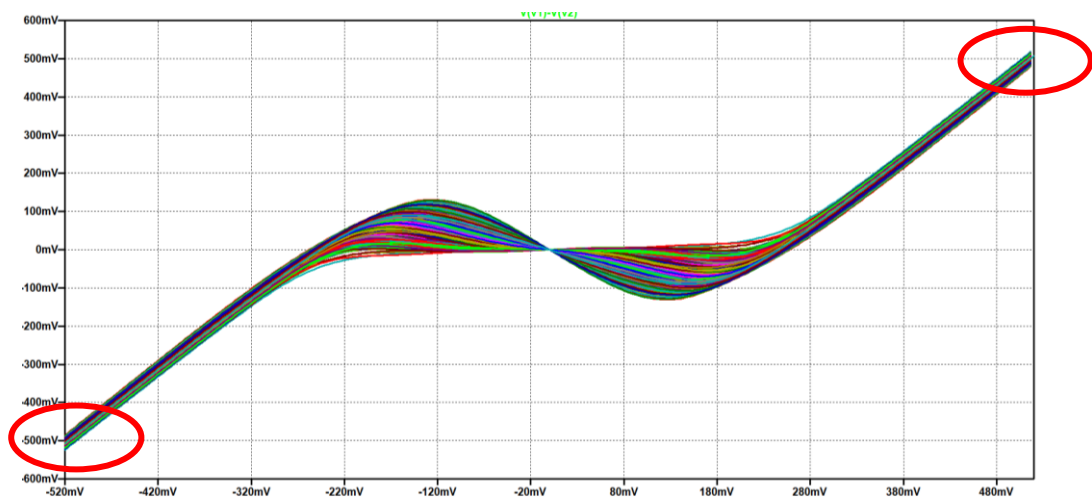
La curva a farfalla tipica della fase di Read è riportata in figura:



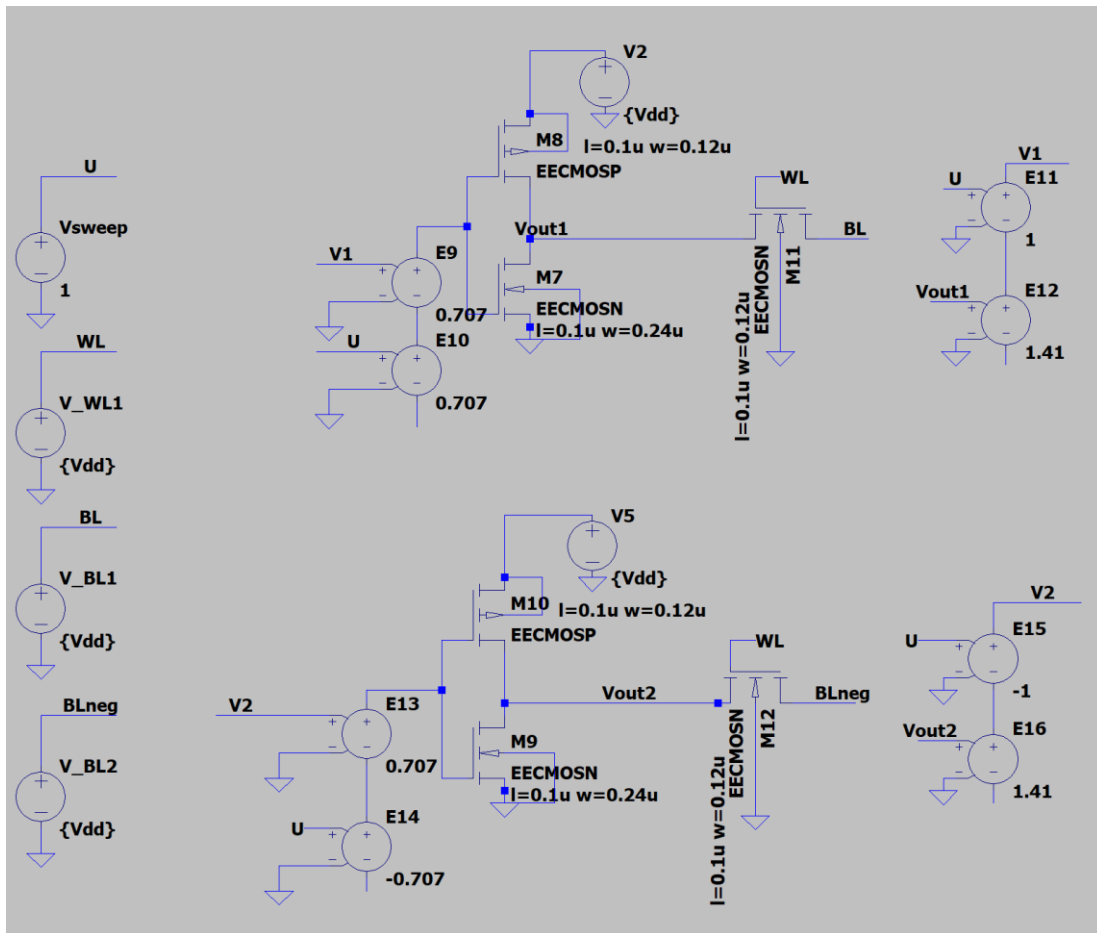
Si nota che la curva ha i lobi intrinsecamente più schiacciati, a causa del partitore resistivo che si viene a formare tra NMOS di PDN ed il transistor di accesso. Di conseguenza, il lato del quadrato inscrivibile nel lobo più piccolo risulta ridotto.

In particolare, se $Q=0$ ($Q_{neg}=1$), la capacità su Q_{neg} rimane floating e non si scarica. Quella su Q si scarica attraverso il transistor di accesso tramite la rete di pull-down della cella. La tensione su BL scende (in quanto dovuta alla precarica della capacità, non si hanno tensioni costanti) creando una differenza di potenziale tra BL e BL_{neg} , che sarà amplificata e catturata dal sense amplifier. In base a quale BL si scarica, si capisce se il dato memorizzato è 0 o 1. Nella realtà, la presenza del partitore influirà su questo processo, formando un piccolo glitch. Questo glitch, se di ampiezza e durata relativamente elevate, può spegnere il PMOS a destra e accendere l'NMOS. Ciò porta $Q_{neg}=0$, accendendo il PMOS a sinistra e spegnendo l'NMOS. Si ha una retroazione positiva con conseguente flip del dato.

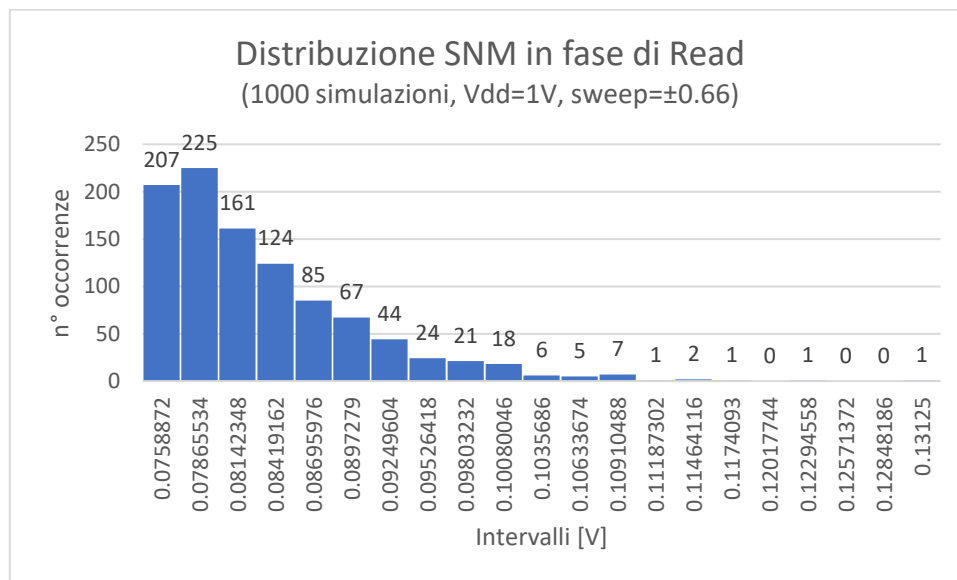
Si procede con la realizzazione del circuito da simulare in LTSpice relativo alla stessa cella di memoria, ma in fase di lettura. Per la simulazione, è necessario introdurre il transistor di accesso alla cella, mentre il resto del circuito rimane uguale a prima. Cambieranno inoltre i potenziali sulle varie linee, come richiesto per la fase di lettura. Si rende necessario adattare gli estremi di sweep del generatore u al variare della tensione di alimentazione, in caso contrario l'algoritmo di calcolo dei minimi/massimi non funzionerebbe. Il motivo è deducibile analizzando la seguente figura, ricavata non adattando gli estremi. Il valore di massimo e minimo risulterebbe sempre uguale agli estremi.

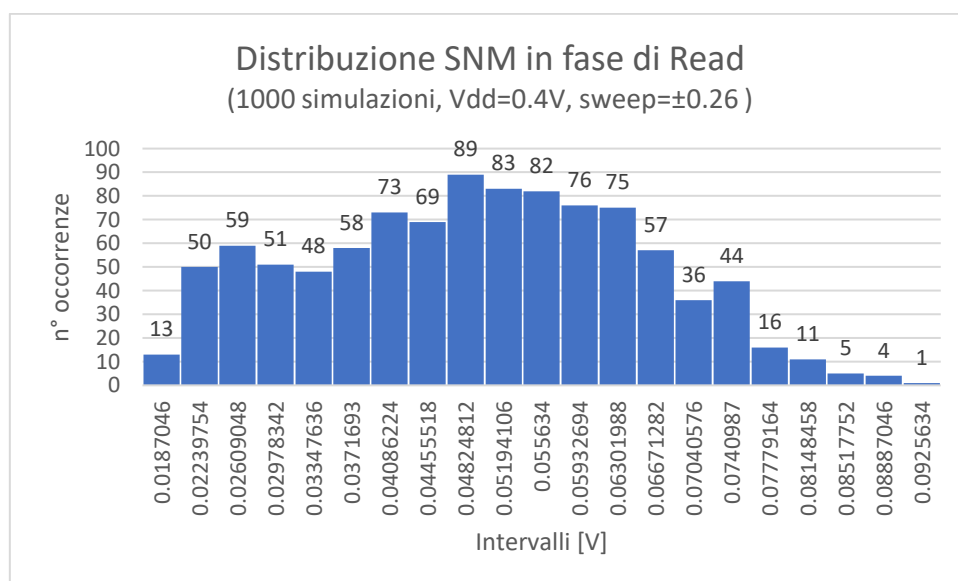
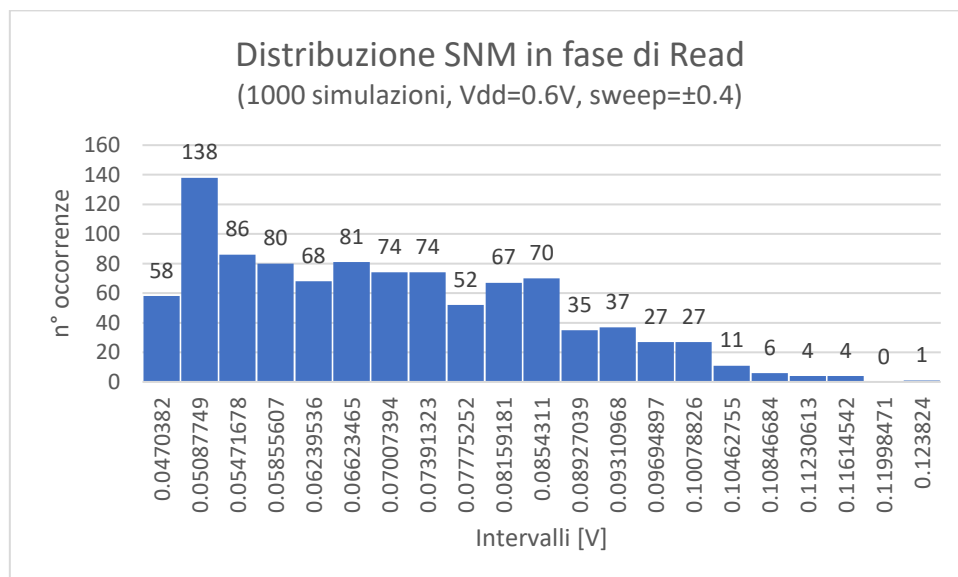
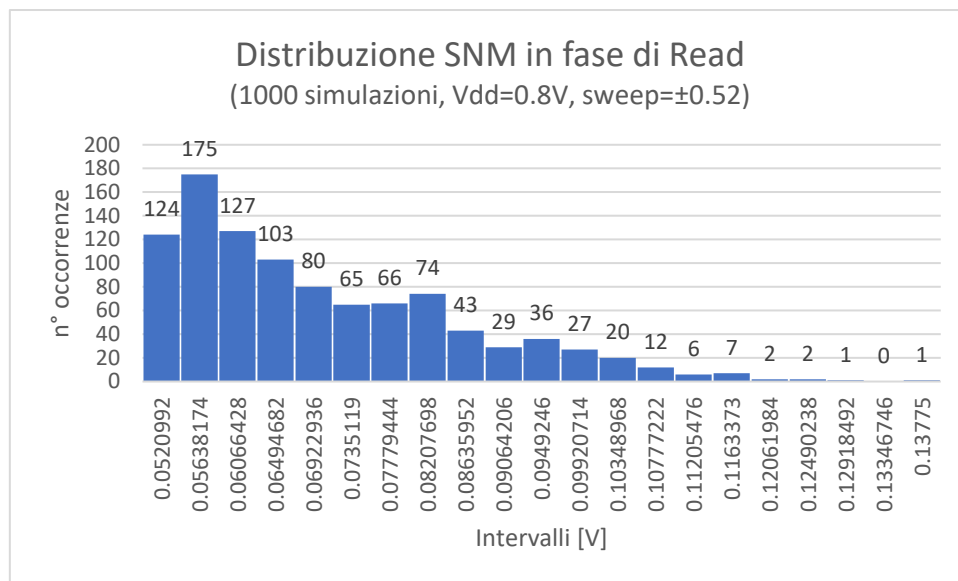


Si riporta il circuito aggiornato.



Si riportano i grafici ottenuti:

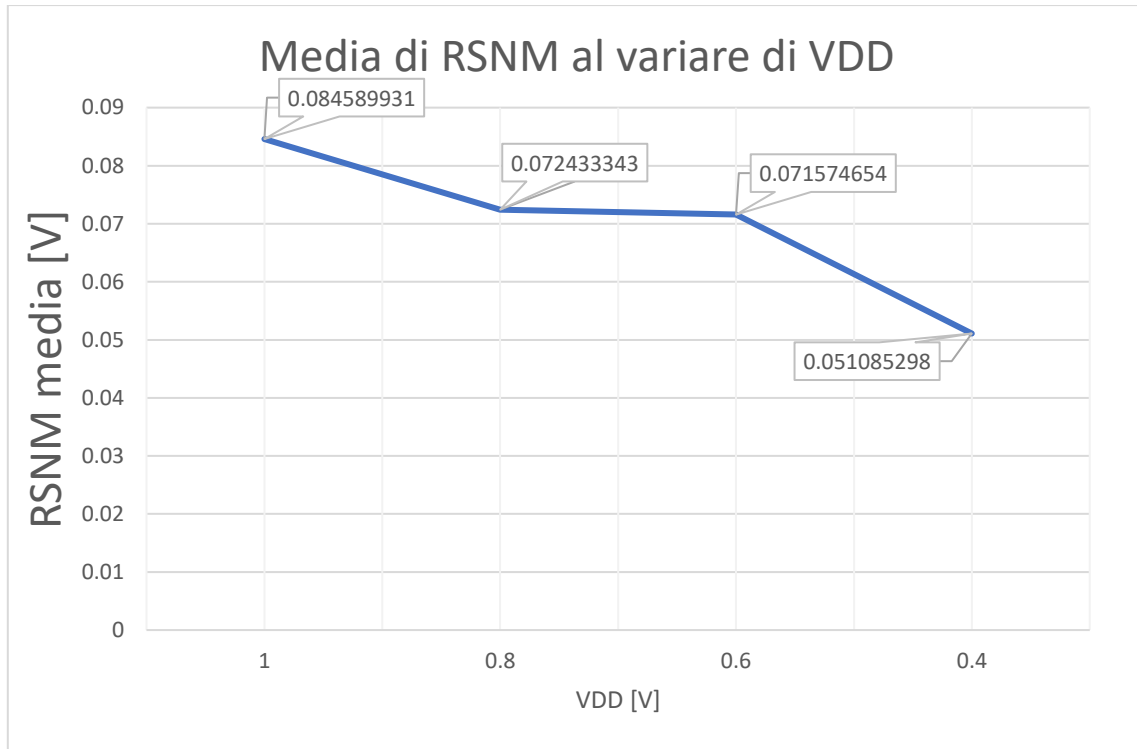




Si riporta la seguente tabella riepilogativa dei dati elaborati nelle diverse simulazioni.

VDD	media	max	min
1	0.08459	0.13125	0.075887
0.8	0.072433	0.13775	0.052099
0.6	0.071575	0.123824	0.047038
0.4	0.051085	0.092563	0.018705

Dalla quale si ricava il grafico in figura:



Dai grafici risulta che anche in fase di Read abbassare la tensione di alimentazione comporta una conseguente riduzione del RSNM per cui la memoria diventa meno leggibile.