Microarquitectura - Estado del arte de la Tecnología

Alejandro Furfaro

7 de mayo de 2025

Temario

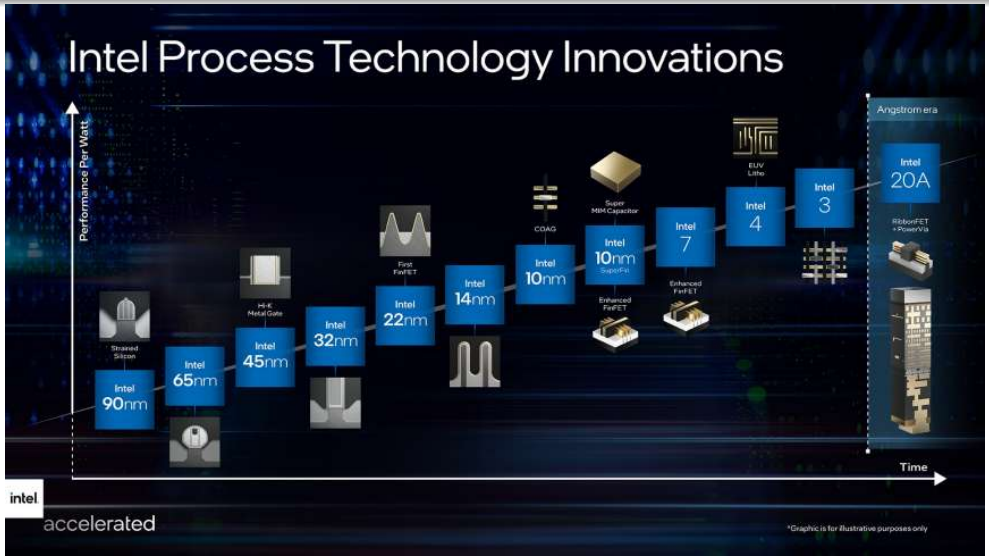
1. Tecnología de Integración
   - En solo poco más de 50 años
   - Métricas
   - Consumo: Una perspectiva “mas física”
2. Arquitectura de Computadores
   - Bases
   - Instruction Set Architecture (ISA)
   - Organización y Hardware
3. Objetivos de estudiar organización y hardware

Temario

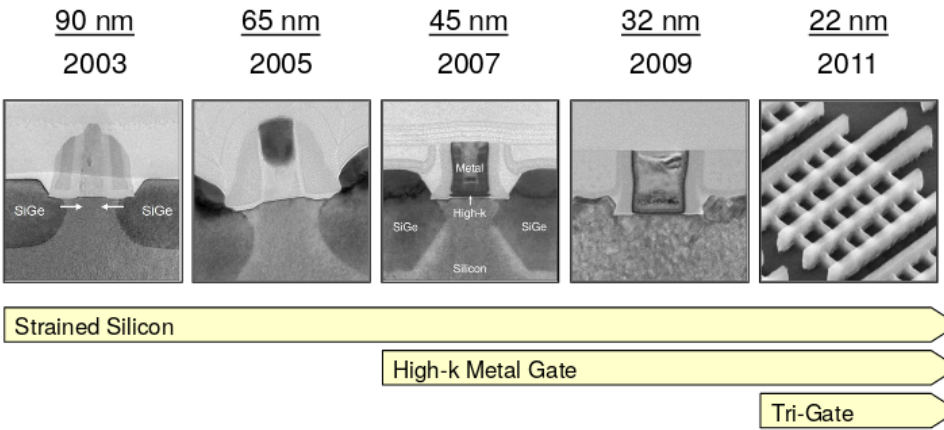
Ley de Moore

“The number of transistors incorporated in a chip will approximately double every 24 months.”
– Gordon Moore, Intel co-founder.
©Cortesía Intel

Evolución Tecnológica. Ej: Intel



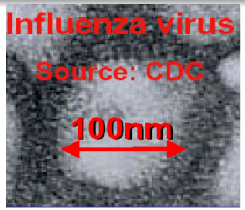
Evolución Tecnológica



Evolución de las tecnologías de integración.©Cortesía Intel.

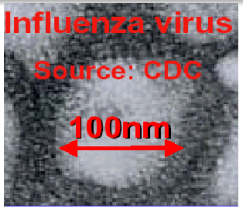
Los progresos en Scaling se logran en base a modificaciones permanentes en la estructura de los transistores y en los materiales que los componen.

¿De que estamos hablando?

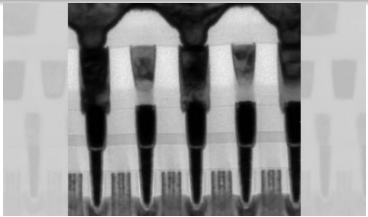


microscópica de una célula del virus de la gripe.

¿De que estamos hablando?

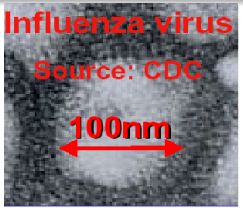


microscópica de una célula del virus de la gripe.

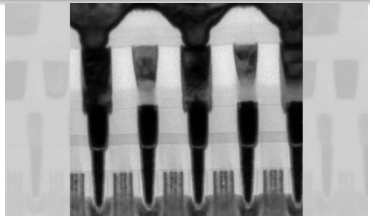


Dos FinFET de 5 nm (Vista en un Microscopio termoeléctrico).

¿De que estamos hablando?



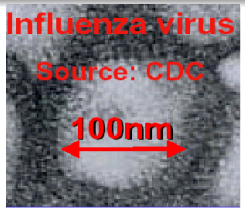
microscópica de una célula del virus de la gripe.



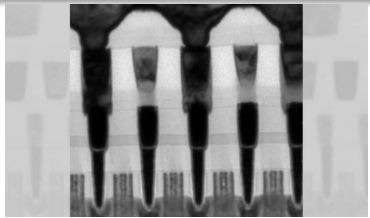
Dos FinFET de 5 nm (Vista en un Microscopio termoeléctrico).

- 2012: primer implementación con transistores MOS tri-gate en 22 nm.

¿De que estamos hablando?



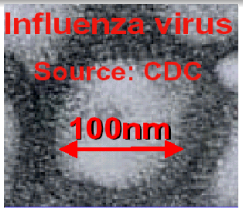
microscópica de una célula del virus de la gripe.



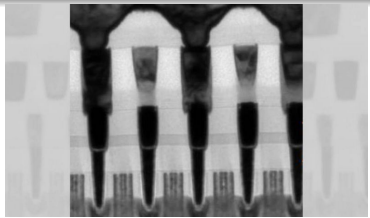
Dos FinFET de 5 nm (Vista en un Microscopio termoeléctrico).

- 2012: primer implementación con transistores MOS tri-gate en 22 nm.
- 2014: Implementación Nodo Tecnológico de 14 nm.

¿De que estamos hablando?



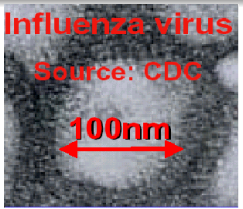
microscópica de una célula del virus de la gripe.



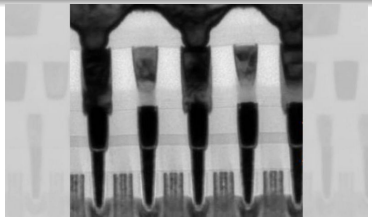
Dos FinFET de 5 nm (Vista en un Microscopio termoeléctrico).

- 2012: primer implementación con transistores MOS tri-gate en 22 nm.
- 2014: Implementación Nodo Tecnológico de 14 nm.
- 2017: Implementación Nodo Tecnológico de 10 nm.

¿De que estamos hablando?



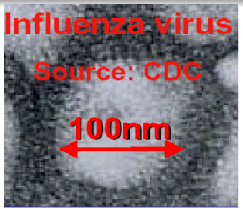
microscópica de una célula del virus de la gripe.



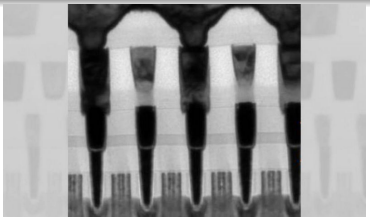
Dos FinFET de 5 nm (Vista en un Microscopio termoeléctrico).

- 2012: primer implementación con transistores MOS tri-gate en 22 nm.
- 2014: Implementación Nodo Tecnológico de 14 nm.
- 2017: Implementación Nodo Tecnológico de 10 nm.
- Actualmente se trabaja en 2.0 a 1,0 nm.

¿De que estamos hablando?



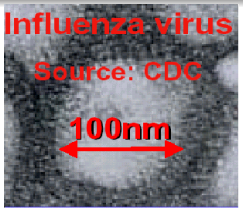
microscópica de una célula del virus de la gripe.



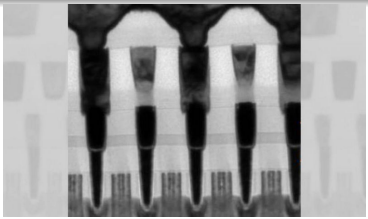
Dos FinFET de 5 nm (Vista en un Microscopio termoeléctrico).

- 2012: primer implementación con transistores MOS tri-gate en 22 nm.
- 2014: Implementación Nodo Tecnológico de 14 nm.
- 2017: Implementación Nodo Tecnológico de 10 nm.
- Actualmente se trabaja en 2.0 a 1,0 nm.
- El espesor total de un transistor de 10 nm equivale a 20 átomos de silicio. Un transistor se compone de diferentes capas de materiales, alguna de las cuales tienen 1 átomo de espesor...

¿De que estamos hablando?



microscópica de una célula del virus de la gripe.



Dos FinFET de 5 nm (Vista en un Microscopio termoeléctrico).

- 2012: primer implementación con transistores MOS tri-gate en 22 nm.
- 2014: Implementación Nodo Tecnológico de 14 nm.
- 2017: Implementación Nodo Tecnológico de 10 nm.
- Actualmente se trabaja en 2.0 a 1,0 nm.
- El espesor total de un transistor de 10 nm equivale a 20 átomos de silicio. Un transistor se compone de diferentes capas de materiales, alguna de las cuales tienen 1 átomo de espesor...
- ¿Cómo lo hacemos?

Evolución Tecnológica

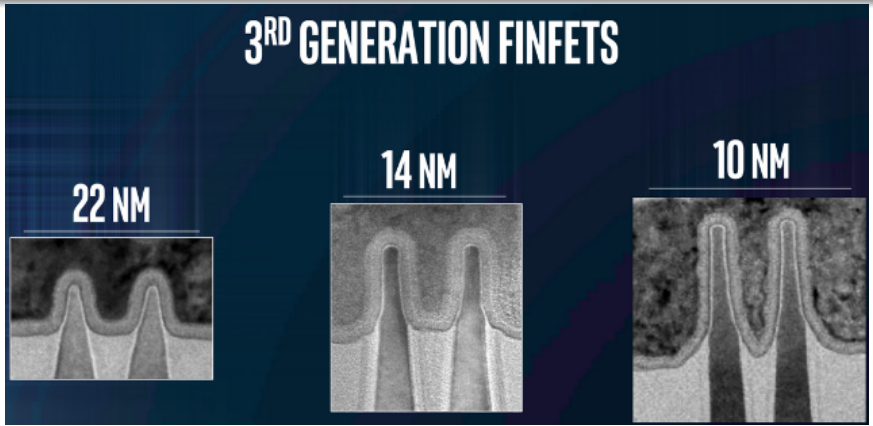


Figura: Evolución de las tecnologías de integración.©Cortesía AMD

un transistor FINFET de 10 nm es 25 % mas alto y 25 % mas angosto que un FINFET de 14 nm

Evolución Tecnológica

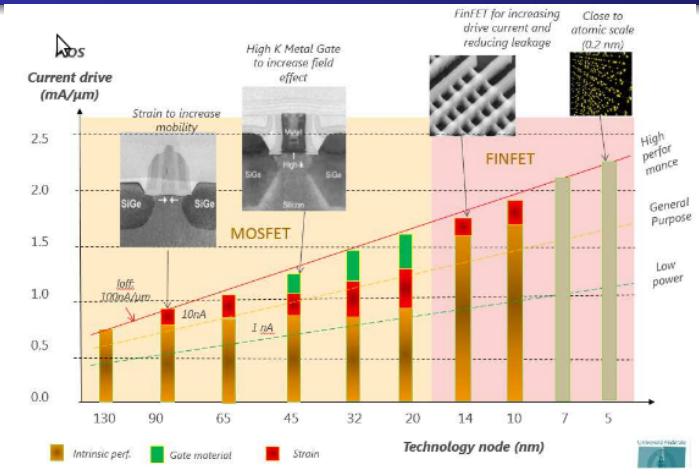


Figura: Velocidad de conmutación vs Scaling. Etienne Sicard. Introducing 10-nm FinFET technology in Microwind. 2017. hal-01551695

Tendencias tecnológicas

La tarea de un diseñador es permanente e inevitablemente moldeada por el rumbo de las tecnologías

Tendencias tecnológicas

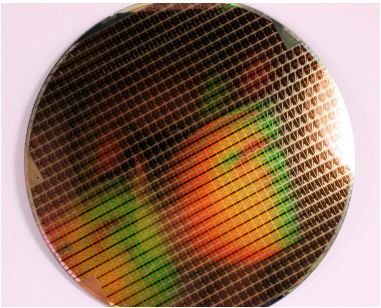
La tarea de un diseñador es permanente e inevitablemente moldeada por el rumbo de las tecnologías

1. La densidad de transistores por unidad de superficie aumenta 35 % por año en promedio. (Otra forma de la ley de Moore).

Tendencias tecnológicas

La tarea de un diseñador es permanente e inevitablemente moldeada por el rumbo de las tecnologías

1. La densidad de transistores por unidad de superficie aumenta 35 % por año en promedio. (Otra forma de la ley de Moore).
2. El tamaño del die aumenta 10 a 20 % por año. Esto deriva en un crecimiento en la cantidad de transistores de entre 40 % y 55 % de un año a otro.



Tendencias tecnológicas

3. La velocidad de clock ya no crece. Parecería haber alcanzado un techo en los 3 GHZ aproximadamente.

Tendencias tecnológicas

3. La velocidad de clock ya no crece. Parecería haber alcanzado un techo en los 3 GHZ aproximadamente.

4. La capacidad de almacenamiento de las memorias DRAM crece a razón de un 40 % por año.

Tendencias tecnológicas

3. La velocidad de clock ya no crece. Parecería haber alcanzado un techo en los 3 GHZ aproximadamente.

4. La capacidad de almacenamiento de las memorias DRAM crece a razón de un 40 % por año.

5. Los discos rígidos aumentan su capacidad 25 a 30 % por año. Su costo por bit de almacenamiento se mantiene entre 50 y 100 veces por debajo del costo de un bit de memoria DRAM

Scaling

Scaling

- El proceso de un circuito integrado está caracterizado por un solo parámetro: *tamaño*, que es el mínimo valor en la dimensión de un transistor en las dimensiones *x* o *y*.

Scaling

- El proceso de un circuito integrado está caracterizado por un solo parámetro: *tamaño*, que es el mínimo valor en la dimensión de un transistor en las dimensiones x o y.
- El tamaño de un transistor en 1971 era de 10 micrones.

Scaling

- El proceso de un circuito integrado está caracterizado por un solo parámetro: *tamaño*, que es el mínimo valor en la dimensión de un transistor en las dimensiones *x* o *y*.
- El tamaño de un transistor en 1971 era de 10 micrones.
- Actualmente es de 0,01 micrones: 1000 veces mas pequeño...

Scaling

- El proceso de un circuito integrado está caracterizado por un solo parámetro: *tamaño*, que es el mínimo valor en la dimensión de un transistor en las dimensiones x o y.
- El tamaño de un transistor en 1971 era de 10 micrones.
- Actualmente es de 0,01 micrones: 1000 veces mas pequeño...
- Los transistores se cuentan por mm^2 de silicio, de modo que podemos esperar una función de incremento del tipo cuadrática.

Scaling

- El proceso de un circuito integrado está caracterizado por un solo parámetro: *tamaño*, que es el mínimo valor en la dimensión de un transistor en las dimensiones x o y.
- El tamaño de un transistor en 1971 era de 10 micrones.
- Actualmente es de 0,01 micrones: 1000 veces mas pequeño...
- Los transistores se cuentan por mm^2 de silicio, de modo que podemos esperar una función de incremento del tipo cuadrática.
- Otro parámetro importante en un transistor es su rendimiento.

Scaling

- El proceso de un circuito integrado está caracterizado por un solo parámetro: *tamaño*, que es el mínimo valor en la dimensión de un transistor en las dimensiones x o y.
- El tamaño de un transistor en 1971 era de 10 micrones.
- Actualmente es de 0,01 micrones: 1000 veces mas pequeño...
- Los transistores se cuentan por mm^2 de silicio, de modo que podemos esperar una función de incremento del tipo cuadrática.
- Otro parámetro importante en un transistor es su rendimiento.
- Al disminuir el tamaño en sentido vertical un transistor requiere reducir su tensión de alimentación. De otro modo su rendimiento decae o puede dañarse.

Scaling

- El proceso de un circuito integrado está caracterizado por un solo parámetro: *tamaño*, que es el mínimo valor en la dimensión de un transistor en las dimensiones x o y.
- El tamaño de un transistor en 1971 era de 10 micrones.
- Actualmente es de 0,01 micrones: 1000 veces mas pequeño...
- Los transistores se cuentan por mm^2 de silicio, de modo que podemos esperar una función de incremento del tipo cuadrática.
- Otro parámetro importante en un transistor es su rendimiento.
- Al disminuir el tamaño en sentido vertical un transistor requiere reducir su tensión de alimentación. De otro modo su rendimiento decae o puede dañarse.
- Como no es posible cambiar la tensión de operación cada vez que se reduce la escala, la mejora en el rendimiento con cada avance en scaling no es cuadrática, sino que tiende a ser lineal.

Scaling

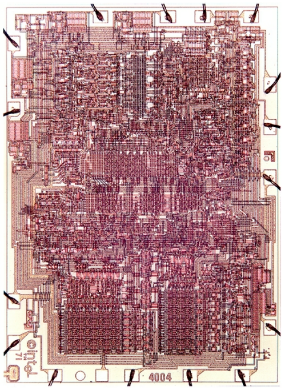


Figura: Procesador 4004, 2300 transistores 10 micrones. 1971 ©Cortesía Intel

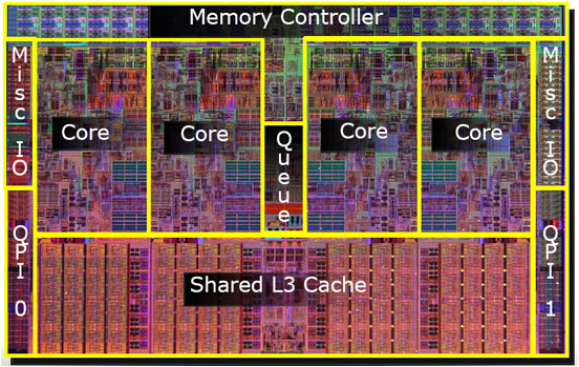
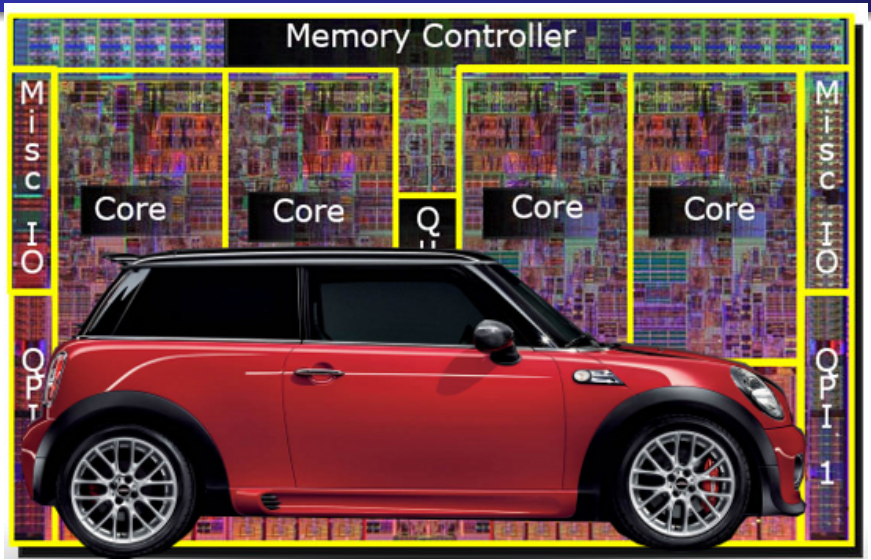


Figura: Procesador Core i7, 2.000.000.000 transistores 22 nm. 2012 ©Cortesía Intel

Scaling


Memory Controller
Misc IO
Core
Core
Core
Core
Misc IO
QPI
QPI

Scaling

Scaling

- A medida que disminuye el parámetro *tamaño*, los transistores ganan linealmente en rendimiento.

Scaling

- A medida que disminuye el parámetro *tamaño*, los transistores ganan linealmente en rendimiento.

- Los “alambres” son los caminos de señal que conectan los diferentes componentes.

Scaling

- A medida que disminuye el parámetro *tamaño*, los transistores ganan linealmente en rendimiento.
- Los “alambres” son los caminos de señal que conectan los diferentes componentes.
- Conforman los buses internos.

Scaling

- A medida que disminuye el parámetro *tamaño*, los transistores ganan linealmente en rendimiento.
- Los “alambres” son los caminos de señal que conectan los diferentes componentes.
- Conforman los buses internos.
- Estos caminos de señal, en las escalas actuales y a las frecuencias de trabajo actuales generan dos problemas:

Scaling

- A medida que disminuye el parámetro *tamaño*, los transistores ganan linealmente en rendimiento.
- Los “alambres” son los caminos de señal que conectan los diferentes componentes.
- Conforman los buses internos.
- Estos caminos de señal, en las escalas actuales y a las frecuencias de trabajo actuales generan dos problemas:
  1. Delays.

Scaling

- A medida que disminuye el parámetro *tamaño*, los transistores ganan linealmente en rendimiento.
- Los “alambres” son los caminos de señal que conectan los diferentes componentes.
- Conforman los buses internos.
- Estos caminos de señal, en las escalas actuales y a las frecuencias de trabajo actuales generan dos problemas:
  1. Delays.
  2. Consumo de Energía.

Delays

Delays

- A pesar de que los “alambres” también se acortan a medida que reducimos las dimensiones, sus efectos no se reducen en consecuencia.

Delays

- A pesar de que los “alambres” también se acortan a medida que reducimos las dimensiones, sus efectos no se reducen en consecuencia.
- Por donde pasar un bus, es un aspecto muy complejo en el diseño.

Delays

- A pesar de que los “alambres” también se acortan a medida que reducimos las dimensiones, sus efectos no se reducen en consecuencia.
- Por donde pasar un bus, es un aspecto muy complejo en el diseño.
- Estos ”nano alambres”finalmente son un medio de transmisión.

Delays

- A pesar de que los “alambres” también se acortan a medida que reducimos las dimensiones, sus efectos no se reducen en consecuencia.
- Por donde pasar un bus, es un aspecto muy complejo en el diseño.
- Estos ”nano alambres”finalmente son un medio de transmisión.
- Con la frecuencia de trabajo actual cobran relevancia las capacidades e inductancias mutuas entre cada ”nanoalambreçon su vecino.

Delays

- A pesar de que los “alambres” también se acortan a medida que reducimos las dimensiones, sus efectos no se reducen en consecuencia.
- Por donde pasar un bus, es un aspecto muy complejo en el diseño.
- Estos ”nano alambres”finalmente son un medio de transmisión.
- Con la frecuencia de trabajo actual cobran relevancia las capacidades e inductancias mutuas entre cada ”nanoalambreçon su vecino.
- Esto genera que la señal inyectada en un extremo se propague al otro con una determinada demora.

Delays

- A pesar de que los “alambres” también se acortan a medida que reducimos las dimensiones, sus efectos no se reducen en consecuencia.
- Por donde pasar un bus, es un aspecto muy complejo en el diseño.
- Estos ”nano alambres”finalmente son un medio de transmisión.
- Con la frecuencia de trabajo actual cobran relevancia las capacidades e inductancias mutuas entre cada ”nanoalambreçon su vecino.
- Esto genera que la señal inyectada en un extremo se propague al otro con una determinada demora.
- El problema es si esta demora no es la misma para cada ”nanoalambre”.

Delays

- A pesar de que los “alambres” también se acortan a medida que reducimos las dimensiones, sus efectos no se reducen en consecuencia.
- Por donde pasar un bus, es un aspecto muy complejo en el diseño.
- Estos ”nano alambres”finalmente son un medio de transmisión.
- Con la frecuencia de trabajo actual cobran relevancia las capacidades e inductancias mutuas entre cada ”nanoalambreçon su vecino.
- Esto genera que la señal inyectada en un extremo se propague al otro con una determinada demora.
- El problema es si esta demora no es la misma para cada ”nanoalambre”.
- A las frecuencias de trabajo actuales las señales pueden llegar a destiempo al extremo receptor.

Consumo / Energía

Consumo / Energía

- En cada ”nano alambre”se disipan solo algunos picoJoules de energía cada vez que se transmite una señal de un extremo a otro.

Consumo / Energía

- En cada ”nano alambre”se disipan solo algunos picoJoules de energía cada vez que se transmite una señal de un extremo a otro.
- La energía que disipa cada alambre es despreciable.

Consumo / Energía

- En cada ”nano alambre”se disipan solo algunos picoJoules de energía cada vez que se transmite una señal de un extremo a otro.
- La energía que disipa cada alambre es despreciable.
- Pero cientos de millones de transistores requieren cientos de millones de cables conectores. Hagan cuentas. . .

Temario

Performance

Performance

$$CPI = \frac{TEC}{TIC} \tag{1}$$

CPI = Cycles per Instruction; ***TEC*** = Total execution cycles; ***TIC*** = Total user-level Instructions Committed

Performance

$$CPI = \frac{TEC}{TIC} \tag{1}$$

CPI = Cycles per Instruction; ***TEC*** = Total execution cycles; ***TIC*** = Total user-level Instructions Committed

$$MemSys\_CPI\_Ovrh = RealCPI - TeoricCPI \tag{2}$$

MemSys_CPI_Ovrh = Memory System CPI Overhead; ***Real CPI*** = CPI Real (el que se mide); ***TeoricCPI*** = CPI considerando Memoria Ideal.

Performance

$$CPI = \frac{TEC}{TIC} \tag{1}$$

CPI = Cycles per Instruction; ***TEC*** = Total execution cycles; ***TIC*** = Total user-level Instructions Committed

$$MemSys\_CPI\_Ovrh = RealCPI - TeoricCPI \tag{2}$$

MemSys_CPI_Ovrh = Memory System CPI Overhead; ***Real CPI*** = CPI Real (el que se mide); ***TeoricCPI*** = CPI considerando Memoria Ideal.

$$MCPI = \frac{TMC}{TIC} \tag{3}$$

MCPI = Memory Cycles Per Instruction; ***RealTMC*** = Ciclos de clock totales insumidos por la memoria; ***TIC*** = Total user-level Instructions Committed.

Energía y Potencia

Energía y Potencia

- En Física, Energía es el trabajo efectuado para realizar una tarea.

Energía y Potencia

- En Física, Energía es el trabajo efectuado para realizar una tarea.
- Aplicado a un sistema de cómputo, puede pensarse como la cantidad de carga de una batería que demanda un algoritmo.

Energía y Potencia

- En Física, Energía es el trabajo efectuado para realizar una tarea.
- Aplicado a un sistema de cómputo, puede pensarse como la cantidad de carga de una batería que demanda un algoritmo.
- Sin embargo, en general, los fabricantes de Microprocesadores y memorias, suelen especificar Potencia instantánea. (Watts)

Energía y Potencia

- En Física, Energía es el trabajo efectuado para realizar una tarea.
- Aplicado a un sistema de cómputo, puede pensarse como la cantidad de carga de una batería que demanda un algoritmo.
- Sin embargo, en general, los fabricantes de Microprocesadores y memorias, suelen especificar Potencia instantánea. (Watts)
- La Potencia Instantánea promedio, se refiere al consumo en el período de conmutación de un transistor CMOS (corte a saturación o visceversa).

Energía y Potencia

- En Física, Energía es el trabajo efectuado para realizar una tarea.
- Aplicado a un sistema de cómputo, puede pensarse como la cantidad de carga de una batería que demanda un algoritmo.
- Sin embargo, en general, los fabricantes de Microprocesadores y memorias, suelen especificar Potencia instantánea. (Watts)
- La Potencia Instantánea promedio, se refiere al consumo en el período de conmutación de un transistor CMOS (corte a saturación o visceversa).

$$P_{avg} = (P_{dynamic} + P_{static}) \approxeq C_{tot} V_{dd}^2 f + I_{leak} V_{dd} \quad (4)$$

$\boldsymbol{P_{avg}}$ = Potencia promedio disipada;
$\boldsymbol{C_{tot}}$ = Capacidad total de carga del CMOS;
$\boldsymbol{V_{dd}}$ = Tensión de alimentación;
$\boldsymbol{f}$ = Frecuencia de conmutación (clock);
$\boldsymbol{I_{leak}}$ = Corriente de pérdida (Aún en corte hay una pequeña corriente de salida)

Energía y Potencia

- La potencia dinámica es la que se disipa en un transistor CMOS en conmutación.

Energía y Potencia

- La potencia dinámica es la que se disipa en un transistor CMOS en conmutación.
- Es proporcional a la Capacidad de carga del dispositivo, al cuadrado de la tensión de alimentación y a la frecuencia de conmutación.

Energía y Potencia

- La potencia dinámica es la que se disipa en un transistor CMOS en conmutación.
- Es proporcional a la Capacidad de carga del dispositivo, al cuadrado de la tensión de alimentación y a la frecuencia de conmutación.

$$P_d = \frac{1}{2} C_c V^2 f_c \tag{5}$$

Energía y Potencia

- La potencia dinámica es la que se disipa en un transistor CMOS en conmutación.
- Es proporcional a la Capacidad de carga del dispositivo, al cuadrado de la tensión de alimentación y a la frecuencia de conmutación.

$$P_d = \frac{1}{2} C_c V^2 f_c \tag{5}$$

- En procesadores destinados a dispositivos portátiles, para dimensionar la capacidad de una batería y su tiempo de duración, mas que la potencia, interesa la energía en Joules:

Energía y Potencia

- La potencia dinámica es la que se disipa en un transistor CMOS en conmutación.
- Es proporcional a la Capacidad de carga del dispositivo, al cuadrado de la tensión de alimentación y a la frecuencia de conmutación.

$$P_d = \frac{1}{2} C_c V^2 f_c \tag{5}$$

- En procesadores destinados a dispositivos portátiles, para dimensionar la capacidad de una batería y su tiempo de duración, mas que la potencia, interesa la energía en Joules:

$$E_d = C_c V^2 \tag{6}$$

Energía y Potencia

Conclusiones de (5) y (6)

1. La tensión de alimentación se redujo en los últimos 30 años de 5V a 0,85V. Esto por sí solo es una reducción drástica en el consumo de un transistor.

Energía y Potencia

Conclusiones de (5) y (6)

1. La tensión de alimentación se redujo en los últimos 30 años de 5V a 0,85V. Esto por sí solo es una reducción drástica en el consumo de un transistor.
2. La capacidad de carga depende de la cantidad de dispositivos que se conecten a la salida de un transistor y de la tecnología de integración empleada.

Energía y Potencia

Conclusiones de (5) y (6)

1. La tensión de alimentación se redujo en los últimos 30 años de 5V a 0,85V. Esto por sí solo es una reducción drástica en el consumo de un transistor.
2. La capacidad de carga depende de la cantidad de dispositivos que se conecten a la salida de un transistor y de la tecnología de integración empleada.
3. Para una tarea fija, reducir la frecuencia de clock disminuye la potencia disipada pero no tiene efecto con la energía consumida.

Energía y Potencia

- La corriente de pérdida I_{leak}, es muy baja pero a partir de las tecnologías de 45 a 30 nm dejó de disminuir conforme se miniaturiza un transistor. Otra mala noticia para el tramo final de la Ley de Moore: leakage pasó a ser un “big issue”.

Energía y Potencia

- La corriente de pérdida I_{leak}, es muy baja pero a partir de las tecnologías de 45 a 30 nm dejó de disminuir conforme se miniaturiza un transistor. Otra mala noticia para el tramo final de la Ley de Moore: leakage pasó a ser un “big issue”.
- La Energía está directamente relacionada con la potencia a través del tiempo. De hecho, la potencia media en un intervalo de tiempo es la relación entre la Energía y la duración del intervalo.

Energía y Potencia

- La corriente de pérdida I_{leak}, es muy baja pero a partir de las tecnologías de 45 a 30 nm dejó de disminuir conforme se miniaturiza un transistor. Otra mala noticia para el tramo final de la Ley de Moore: leakage pasó a ser un “big issue”.
- La Energía está directamente relacionada con la potencia a través del tiempo. De hecho, la potencia media en un intervalo de tiempo es la relación entre la Energía y la duración del intervalo.
- En un intervalo T, la cantidad de Joules consumidos es:

Energía y Potencia

- La corriente de pérdida $\boldsymbol{I_{leak}}$, es muy baja pero a partir de las tecnologías de 45 a 30 nm dejó de disminuir conforme se miniaturiza un transistor. Otra mala noticia para el tramo final de la Ley de Moore: leakage pasó a ser un “big issue”.
- La Energía está directamente relacionada con la potencia a través del tiempo. De hecho, la potencia media en un intervalo de tiempo es la relación entre la Energía y la duración del intervalo.
- En un intervalo T, la cantidad de Joules consumidos es:

$$E = P_{avg}.T \approxeq C_{tot} V_{dd}^2 N + I_{leak} V_{dd}.T \quad (7)$$

$\boldsymbol{N}$ = Cantidad de eventos de switching ocurridos en el intervalo T.

Energía y Potencia

- La corriente de pérdida I_{leak}, es muy baja pero a partir de las tecnologías de 45 a 30 nm dejó de disminuir conforme se miniaturiza un transistor. Otra mala noticia para el tramo final de la Ley de Moore: leakage pasó a ser un “big issue”.
- La Energía está directamente relacionada con la potencia a través del tiempo. De hecho, la potencia media en un intervalo de tiempo es la relación entre la Energía y la duración del intervalo.
- En un intervalo T, la cantidad de Joules consumidos es:

$$E = P_{avg}.T \approxeq C_{tot} V_{dd}^2 N + I_{leak} V_{dd}.T \quad (7)$$

  N = Cantidad de eventos de switching ocurridos en el intervalo T.
- Una primer conclusión es que la energía consumida por un algoritmo (la medida del trabajo que realiza el computador) *no depende de la frecuencia de clock*.

Energía y Potencia

- Cuando nos referimos al consumo, lo correcto es hablar de Joules, es decir, Energía. Esto es duración de batería, o KWh en la factura del servicio de electricidad. Sin embargo coloquialmente se suele hablar de energía cuando se miden los Watts instantáneos de un chip. Esto no es correcto.

Energía y Potencia

- Cuando nos referimos al consumo, lo correcto es hablar de Joules, es decir, Energía. Esto es duración de batería, o KWh en la factura del servicio de electricidad. Sin embargo coloquialmente se suele hablar de energía cuando se miden los Watts instantáneos de un chip. Esto no es correcto.
- Típico anuncio de marketing: “Nuestra nueva familia de procesadores “low power”mejora a la línea anterior en un factor de 2”.

Energía y Potencia

- Cuando nos referimos al consumo, lo correcto es hablar de Joules, es decir, Energía. Esto es duración de batería, o KWh en la factura del servicio de electricidad. Sin embargo coloquialmente se suele hablar de energía cuando se miden los Watts instantáneos de un chip. Esto no es correcto.
- Típico anuncio de marketing: “Nuestra nueva familia de procesadores “low power”mejora a la línea anterior en un factor de 2”.
- Bien. Esto puede lograrse simplemente subclockeando a la mitad de frecuencia, reduciendo así a la mitad la potencia disipada.

Energía y Potencia

- Cuando nos referimos al consumo, lo correcto es hablar de Joules, es decir, Energía. Esto es duración de batería, o KWh en la factura del servicio de electricidad. Sin embargo coloquialmente se suele hablar de energía cuando se miden los Watts instantáneos de un chip. Esto no es correcto.
- Típico anuncio de marketing: “Nuestra nueva familia de procesadores “low power”mejora a la línea anterior en un factor de 2”.
- Bien. Esto puede lograrse simplemente subclockeando a la mitad de frecuencia, reduciendo así a la mitad la potencia disipada.
- La potencia puede ser conceptualizada como la velocidad a la que se consume la energía.

Energía y Potencia

- Cuando nos referimos al consumo, lo correcto es hablar de Joules, es decir, Energía. Esto es duración de batería, o KWh en la factura del servicio de electricidad. Sin embargo coloquialmente se suele hablar de energía cuando se miden los Watts instantáneos de un chip. Esto no es correcto.
- Típico anuncio de marketing: “Nuestra nueva familia de procesadores “low power”mejora a la línea anterior en un factor de 2”.
- Bien. Esto puede lograrse simplemente subclockeando a la mitad de frecuencia, reduciendo así a la mitad la potencia disipada.
- La potencia puede ser conceptualizada como la velocidad a la que se consume la energía.
- Subclockear solo disminuye la velocidad con que se descarga la batería. El trabajo realizado en Joules para un dado algoritmo será el mismo. Solo que tardará el doble de tiempo en realizarlo.

Energía y Potencia: Métricas de Interés

- Considerando los slides anteriores, las métricas que valen la pena y suelen encontrarse son las siguientes:

Energía y Potencia: Métricas de Interés

- Considerando los slides anteriores, las métricas que valen la pena y suelen encontrarse son las siguientes:
- **Energy-Delay Product**

$$Energy - DelayProduct = EnergiaRequeridaPorLaTarea.TiempoRequeridoPorLaTarea \quad (8)$$

Energía y Potencia: Métricas de Interés

- Considerando los slides anteriores, las métricas que valen la pena y suelen encontrarse son las siguientes:
- **Energy-Delay Product**

$$Energy-DelayProduct = EnergiaRequeridaPorLaTarea.TiempoRequeridoPorLaTarea \tag{8}$$

- **Power-Delay Product**

$$Power-DelayProduct = PotenciaConsumidaPorLaTarea.TiempoRequeridoPorLaTarea \tag{9}$$

Energía y Potencia: Métricas de Interés

- Considerando los slides anteriores, las métricas que valen la pena y suelen encontrarse son las siguientes:
- **Energy-Delay Product**

$$Energy-DelayProduct = EnergiaRequeridaPorLaTarea.TiempoRequeridoPorLaTarea \tag{8}$$

- **Power-Delay Product**

$$Power-DelayProduct = PotenciaConsumidaPorLaTarea.TiempoRequeridoPorLaTarea \tag{9}$$

- **MIPS per watt**

$$MIPSperWatt = \frac{PerformanceBenchmarckEnMIPS}{PotenciaPromedioDisipadaPorLaTarea} \tag{10}$$

¿Por que razón es un problema el Consumo?

- El incremento en la cantidad de transistores CMOS por mm^2 de superficie tiene preeminencia por sobre los ahorros de energía individuales de cada transistor debidos al cambio de tecnología.

¿Por que razón es un problema el Consumo?

- El incremento en la cantidad de transistores CMOS por mm^2 de superficie tiene preeminencia por sobre los ahorros de energía individuales de cada transistor debidos al cambio de tecnología.
- Por lo tanto cada vez es mas crítico el manejo del consumo.

¿Por que razón es un problema el Consumo?

- El incremento en la cantidad de transistores CMOS por mm^2 de superficie tiene preeminencia por sobre los ahorros de energía individuales de cada transistor debidos al cambio de tecnología.
- Por lo tanto cada vez es mas crítico el manejo del consumo.
- El procesador 4004 de Intel en 1971 tenía poco mas de 2300 transistores y su consumo era de algunas décimas de Watts.Su clock era de 108 KHz (si. . . leyó bien. . . Kilo Hertz)

¿Por que razón es un problema el Consumo?

- El incremento en la cantidad de transistores CMOS por mm^2 de superficie tiene preeminencia por sobre los ahorros de energía individuales de cada transistor debidos al cambio de tecnología.
- Por lo tanto cada vez es mas crítico el manejo del consumo.
- El procesador 4004 de Intel en 1971 tenía poco mas de 2300 transistores y su consumo era de algunas décimas de Watts.Su clock era de 108 KHz (si. . . leyó bien. . . Kilo Hertz)
- El procesador Pentium IV Extreme Edition desarrollado en 2001 (30 años después), llegó a consumir 135 Watts. Tenía cerca de 40 millones de transistores y un clock de 3GHz

¿Por que razón es un problema el Consumo?

- El incremento en la cantidad de transistores CMOS por mm^2 de superficie tiene preeminencia por sobre los ahorros de energía individuales de cada transistor debidos al cambio de tecnología.
- Por lo tanto cada vez es mas crítico el manejo del consumo.
- El procesador 4004 de Intel en 1971 tenía poco mas de 2300 transistores y su consumo era de algunas décimas de Watts.Su clock era de 108 KHz (si...leyó bien...Kilo Hertz)
- El procesador Pentium IV Extreme Edition desarrollado en 2001 (30 años después), llegó a consumir 135 Watts. Tenía cerca de 40 millones de transistores y un clock de 3GHz
- Por lo tanto cada vez existen mas limitaciones tanto con la distribución de la alimentación como con el ahorro de potencia y energía.

Tendencias en reducción del Consumo

- La mayoría de los procesadores actuales contiene bloques de hardware para control de consumo, que se encargan de mantener alimentados solo los bloques funcionales que se necesitan en cada momento.

Tendencias en reducción del Consumo

- La mayoría de los procesadores actuales contiene bloques de hardware para control de consumo, que se encargan de mantener alimentados solo los bloques funcionales que se necesitan en cada momento.

- Si el procesador no está ejecutando operaciones de Punto Flotante, entonces la Unidad de Punto Flotante se mantiene apagada.

Tendencias en reducción del Consumo

- La mayoría de los procesadores actuales contiene bloques de hardware para control de consumo, que se encargan de mantener alimentados solo los bloques funcionales que se necesitan en cada momento.
- Si el procesador no está ejecutando operaciones de Punto Flotante, entonces la Unidad de Punto Flotante se mantiene apagada.
- Lo mismo con cada Unidad interna.

Corrientes de fuga (leakage)

- A pesar de que un transistor esté al corte, circula una muy pequeña de todos modos

Corrientes de fuga (leakage)

- A pesar de que un transistor esté al corte, circula una muy pequeña de todos modos
- Interesa determinar la Potencia estática, relacionada con esta corriente:

Corrientes de fuga (leakage)

- A pesar de que un transistor esté al corte, circula una muy pequeña de todos modos
- Interesa determinar la Potencia estática, relacionada con esta corriente:

$$E_e = I_e V^2 \tag{11}$$

Corrientes de fuga (leakage)

- A pesar de que un transistor esté al corte, circula una muy pequeña de todos modos
- Interesa determinar la Potencia estática, relacionada con esta corriente:

$$E_e = I_e V^2 \tag{11}$$

- V = Tensión de Alimentación, I_e = corriente de fuga (likage). Cada transistor tiene así una componente adicional de potencia cuando está en corte.

Corrientes de fuga (leakage)

- A pesar de que un transistor esté al corte, circula una muy pequeña de todos modos
- Interesa determinar la Potencia estática, relacionada con esta corriente:

$$E_e = I_e V^2 \tag{11}$$

- V = Tensión de Alimentación, I_e = corriente de fuga (likage). Cada transistor tiene así una componente adicional de potencia cuando está en corte.
- A medida que aumenta la cantidad de transistores esta corriente se hace mas significativa.

Corrientes de fuga (leakage)

- A pesar de que un transistor esté al corte, circula una muy pequeña de todos modos
- Interesa determinar la Potencia estática, relacionada con esta corriente:

$$E_e = I_e V^2 \tag{11}$$

- V = Tensión de Alimentación, I_e = corriente de fuga (likage). Cada transistor tiene así una componente adicional de potencia cuando está en corte.
- A medida que aumenta la cantidad de transistores esta corriente se hace mas significativa.
- En 2006 los principales diseñadores establecieron como meta que esta corriente represente solo el 25 % del consumo total del chip. Aún así los modelos de mas alto rendimiento no lograron esta marca.

Distribución de la tensión de alimentación

- La alimentación en un circuito integrado moderno es otro tema a considerar, por varios factores.
- Se debe distribuir la tensión de alimentación a todo el circuito integrado. Esto motiva
- Desde hace mas de una década que los circuitos integrados dediquen una buena cantidad de terminales de conexión a V_{DD} y Tierra.

Temario

Potencia Disipada en un sistema de cómputo

Potencia Disipada en un sistema de cómputo

- La potencia dispada en circuitos basados en transistores CMOS proviene de dos factores

Potencia Disipada en un sistema de cómputo

- La potencia dispada en circuitos basados en transistores CMOS proviene de dos factores

✓ ***Potencia estática (leakage power)***: Proviene del hecho que un transistor CMOS cuando está en estado de corte no está completamente “apagado”.

Potencia Disipada en un sistema de cómputo

- La potencia dispada en circuitos basados en transistores CMOS proviene de dos factores

✓ ***Potencia estática (leakage power)***: Proviene del hecho que un transistor CMOS cuando está en estado de corte no está completamente “apagado”.

✓ ***Potencia dinámica***: Resultado de conmutar a una carga capacitiva en la malla de salida entre dos estados de tensión.

Potencia Disipada en un sistema de cómputo

- La potencia dispada en circuitos basados en transistores CMOS proviene de dos factores

✓ ***Potencia estática (leakage power)***: Proviene del hecho que un transistor CMOS cuando está en estado de corte no está completamente “apagado”.

✓ ***Potencia dinámica***: Resultado de conmutar a una carga capacitiva en la malla de salida entre dos estados de tensión.

- La ***Potencia dinámica*** depende de la actividad de conmutación del circuito. Es decir de la frecuencia con que conmute. Si no cambia el valor de tensión en al salida del CMOS, no hay conmutación, y no se disipa potencia.

Potencia Disipada en un sistema de cómputo

- La potencia dispada en circuitos basados en transistores CMOS proviene de dos factores

✓ ***Potencia estática (leakage power)***: Proviene del hecho que un transistor CMOS cuando está en estado de corte no está completamente “apagado”.

✓ ***Potencia dinámica***: Resultado de conmutar a una carga capacitiva en la malla de salida entre dos estados de tensión.

- La ***Potencia dinámica*** depende de la actividad de conmutación del circuito. Es decir de la frecuencia con que conmute. Si no cambia el valor de tensión en al salida del CMOS, no hay conmutación, y no se disipa potencia.
- Por su parte la ***Potencia estática*** es independiente de la frecuencia, y existe simplemente porque el chip está alimentado.

leakage

leakage

- La tecnología CMOS se destacó por su potencia de leakage prácticamente despreciable. Esa fue la razón de su adopción para transistores de circuitos lógicos.

leakage

- La tecnología CMOS se destacó por su potencia de leakage prácticamente despreciable. Esa fue la razón de su adopción para transistores de circuitos lógicos.
- Conforme avanzó la tecnología de integración, el tamaño de los transistores se redujo notablemente (scaling)

leakage

- La tecnología CMOS se destacó por su potencia de leakage prácticamente despreciable. Esa fue la razón de su adopción para transistores de circuitos lógicos.
- Conforme avanzó la tecnología de integración, el tamaño de los transistores se redujo notablemente (scaling)
- La Potencia dinámica se reduce linealmente con el tamaño del gate del transistor CMOS.

leakage

- La tecnología CMOS se destacó por su potencia de leakage prácticamente despreciable. Esa fue la razón de su adopción para transistores de circuitos lógicos.
- Conforme avanzó la tecnología de integración, el tamaño de los transistores se redujo notablemente (scaling)
- La Potencia dinámica se reduce linealmente con el tamaño del gate del transistor CMOS.
- La potencia de leakage no.

leakage

- La tecnología CMOS se destacó por su potencia de leakage prácticamente despreciable. Esa fue la razón de su adopción para transistores de circuitos lógicos.
- Conforme avanzó la tecnología de integración, el tamaño de los transistores se redujo notablemente (scaling)
- La Potencia dinámica se reduce linealmente con el tamaño del gate del transistor CMOS.
- La potencia de leakage no.
- La consecuencia de esta situación es que con el tiempo pasó a ser mas significativa que la potencia dinámica.

leakage

- La tecnología CMOS se destacó por su potencia de leakage prácticamente despreciable. Esa fue la razón de su adopción para transistores de circuitos lógicos.
- Conforme avanzó la tecnología de integración, el tamaño de los transistores se redujo notablemente (scaling)
- La Potencia dinámica se reduce linealmente con el tamaño del gate del transistor CMOS.
- La potencia de leakage no.
- La consecuencia de esta situación es que con el tiempo pasó a ser mas significativa que la potencia dinámica.
- En 2005 aproximadamente los diseñadores tenían como objetivo de diseño mantener la potencia de leakage en no mas del 25 % de la potencia disipada total.

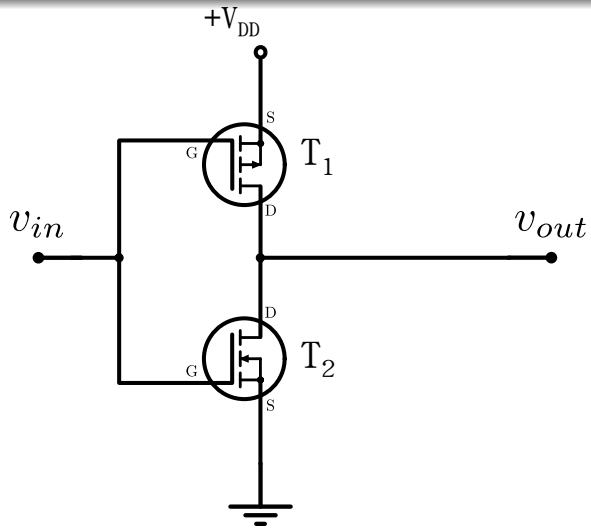
leakage

- La tecnología CMOS se destacó por su potencia de leakage prácticamente despreciable. Esa fue la razón de su adopción para transistores de circuitos lógicos.
- Conforme avanzó la tecnología de integración, el tamaño de los transistores se redujo notablemente (scaling)
- La Potencia dinámica se reduce linealmente con el tamaño del gate del transistor CMOS.
- La potencia de leakage no.
- La consecuencia de esta situación es que con el tiempo pasó a ser mas significativa que la potencia dinámica.
- En 2005 aproximadamente los diseñadores tenían como objetivo de diseño mantener la potencia de leakage en no mas del 25 % de la potencia disipada total.
- Luego de los 30nm a 40nm, la Potencia de leakage ya no disminuye con el scaling.

Conmutación con carga capacitiva

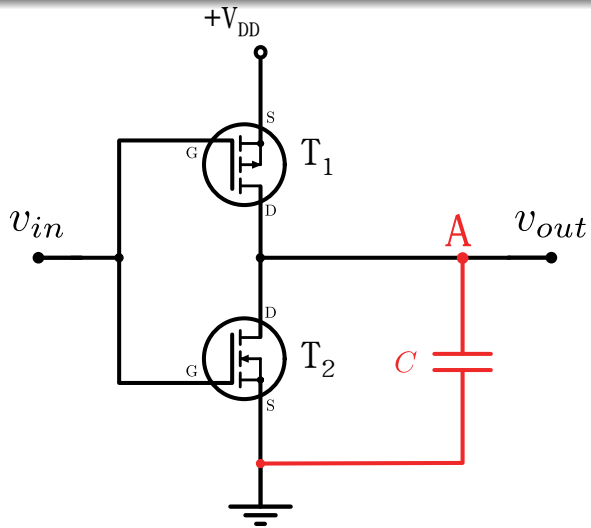
Conmutación con carga capacitiva

- A la derecha un circuito Inversor CMOS



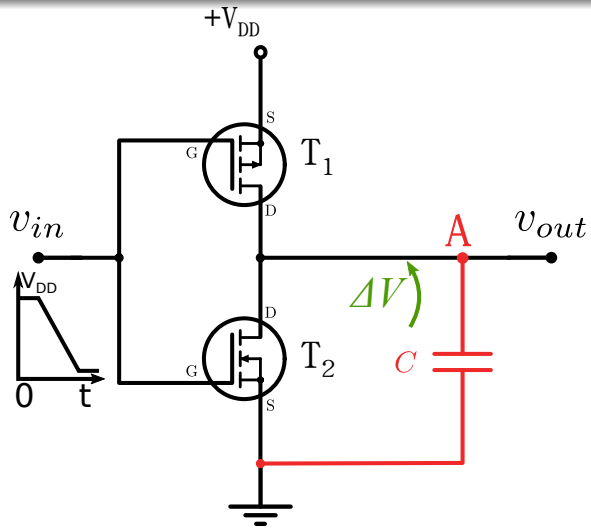
Conmutación con carga capacitiva

- A la derecha un circuito Inversor CMOS
- Si lo cargamos con otro CMOS la carga es capacitiva.



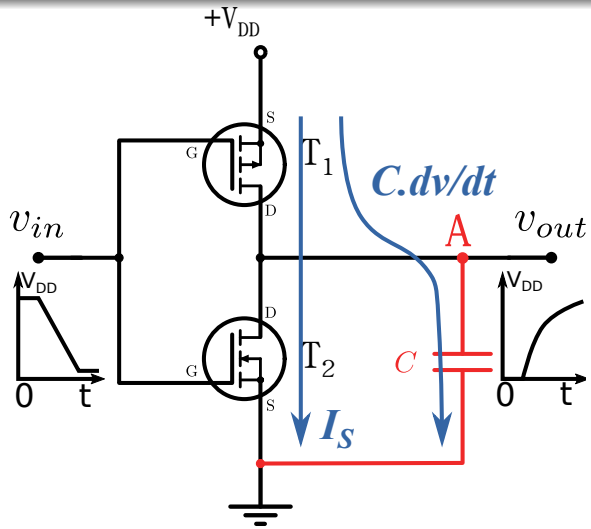
Conmutación con carga capacitiva

- A la derecha un circuito Inversor CMOS
- Si lo cargamos con otro CMOS la carga es capacitiva.
- Si en el nodo **A** se produce una variación de tensión ΔV, se genera una corriente para cargar el capacitor C ΔV Volts, y descargarlo a su valor de tensión original.



Conmutación con carga capacitiva

- A la derecha un circuito Inversor CMOS
- Si lo cargamos con otro CMOS la carga es capacitiva.
- Si en el nodo **A** se produce una variación de tensión ΔV, se genera una corriente para cargar el capacitor C ΔV Volts, y descargarlo a su valor de tensión original.
- Se genera un flujo de carga igual a $C.\Delta V$ desde el Nodo V_{DD},hasta el capacitor y desde este por la malla de descarga.



Conmutación con carga capacitiva

Finalizado el ciclo carga/descarga el CMOS y el capacitor movieron desde V_{DD} hasta tierra una cantidad de carga eléctrica igual a $C.\Delta V$. Esto significa que han utilizado una cantidad de Energía igual a $C.\Delta V.V_{DD}$
Y esa Energía es independiente del ciclo de tiempo en el que se realiza el movimiento de carga

Conmutación con carga capacitiva

La potencia dinámica promedio de este nodo es la velocidad a la que se consume esta energía, que viene dada por [1]:

$$P_{dyn} = \frac{C.\Delta V.V_{DD}.\alpha}{T}, \tag{12}$$

[1] Chapter 3. Low Power Digital CMOS Design. Chandrakasan & R. W. Brodersen. Publisher: Springer US, Year: 1995

Conmutación con carga capacitiva

La potencia dinámica promedio de este nodo es la velocidad a la que se consume esta energía, que viene dada por [1]:

$$P_{dyn} = \frac{C.\Delta V.V_{DD}.\alpha}{T}, \tag{12}$$

donde T es el período de carga / descarga, es decir, la inversa de la frecuencia de clock, y la *razón de actividad* $\alpha, /0 \leq \alpha \leq 1$, es la probabilidad que el nodo conmute, en cuyo caso consumirá energía (si el nodo no conmuta no se consume energía).

[1] Chapter 3. Low Power Digital CMOS Design. Chandrakasan & R. W. Brodersen. Publisher: Springer US, Year: 1995

Conmutación con carga capacitiva

La potencia dinámica promedio de este nodo es la velocidad a la que se consume esta energía, que viene dada por [1]:

$$P_{dyn} = \frac{C.\Delta V.V_{DD}.\alpha}{T}, \tag{12}$$

donde T es el período de carga / descarga, es decir, la inversa de la frecuencia de clock, y la *razón de actividad* $\alpha, /0 \leq \alpha \leq 1$, es la probabilidad que el nodo conmute, en cuyo caso consumirá energía (si el nodo no conmuta no se consume energía).
Incluir α permite estimar el consumo del nodo durante mucho mas que un período de la señal de clock, permitiendo calcular la potencia promedio durante horas enteras de computación, siempre que la *razón de actividad* se mantenga.

[1] Chapter 3. Low Power Digital CMOS Design. Chandrakasan & R. W. Brodersen. Publisher: Springer US, Year: 1995

Conmutación con carga capacitiva

La potencia dinámica promedio de este nodo es la velocidad a la que se consume esta energía, que viene dada por [1]:

$$P_{dyn} = \frac{C.\Delta V.V_{DD}.\alpha}{T}, \tag{12}$$

donde T es el período de carga / descarga, es decir, la inversa de la frecuencia de clock, y la *razón de actividad* $\alpha, /0 \leq \alpha \leq 1$, es la probabilidad que el nodo conmute, en cuyo caso consumirá energía (si el nodo no conmuta no se consume energía).
Incluir α permite estimar el consumo del nodo durante mucho mas que un período de la señal de clock, permitiendo calcular la potencia promedio durante horas enteras de computación, siempre que la *razón de actividad* se mantenga.
La suma de la ecuación (12) a lo largo de todos los nodos del chip dá como resultado la ***Potencia Dinámica*** total.

[1] Chapter 3. Low Power Digital CMOS Design. Chandrakasan & R. W. Brodersen. Publisher: Springer US, Year: 1995

Conmutación con carga capacitiva

De la ecuación (12) surge que si disminuyen la Capacidad de carga, o V_{DD}, disminuirá de manera directa la ***Potencia Dinámica***.

Conmutación con carga capacitiva

De la ecuación (12) surge que si disminuyen la Capacidad de carga, o V_{DD}, disminuirá de manera directa la ***Potencia Dinámica***.

En un circuito lógico la excursión de Tensión $C.\Delta V$ normalmente es desde un valor muy cercano a 0 Volts hasta V_{DD}, de modo que la ecuación (12) se transforma en:

$$P_{dyn} = C.V_{DD}^2.\alpha.f, \tag{13}$$

Conmutación con carga capacitiva

De la ecuación (12) surge que si disminuyen la Capacidad de carga, o V_{DD}, disminuirá de manera directa la ***Potencia Dinámica***.

En un circuito lógico la excursión de Tensión $C.\Delta V$ normalmente es desde un valor muy cercano a 0 Volts hasta V_{DD}, de modo que la ecuación (12) se transforma en:

$$P_{dyn} = C.V_{DD}^2.\alpha.f, \tag{13}$$

Además se verifica empíricamente que la *razón de actividad* en circuitos lógicos es ½, de modo que la ecuación (16) queda:

$$P_{dyn} = \frac{1}{2}.C.V_{DD}^2.f, \tag{14}$$

Conmutación con carga capacitiva

De la ecuación (12) surge que si disminuyen la Capacidad de carga, o V_{DD}, disminuirá de manera directa la ***Potencia Dinámica***.
En un circuito lógico la excursión de Tensión $C.\Delta V$ normalmente es desde un valor muy cercano a 0 Volts hasta V_{DD}, de modo que la ecuación (12) se transforma en:

$$P_{dyn} = C.V_{DD}^2.\alpha.f, \tag{13}$$

Además se verifica empíricamente que la *razón de actividad* en circuitos lógicos es ½, de modo que la ecuación (16) queda:

$$P_{dyn} = \frac{1}{2}.C.V_{DD}^2.f, \tag{14}$$

La ecuación (14) es la que mejor representa la ***Potencia Dinámica*** en un chip, considerando que es la suma de todos los nodos del chip.

leakage

leakage

- Como ya se dijo, es la imposibilidad de apagar por completo al transistor CMOS cuando éste está en el estado de corte.

leakage

- Como ya se dijo, es la imposibilidad de apagar por completo al transistor CMOS cuando éste está en el estado de corte.
- Así que conduce corriente por debajo del umbral de conducción.

leakage

- Como ya se dijo, es la imposibilidad de apagar por completo al transistor CMOS cuando éste está en el estado de corte.
- Así que conduce corriente por debajo del umbral de conducción.
- La compuerta se acopla al canal activo principalmente a través de la capacitancia de óxido de la compuerta.

leakage

- Como ya se dijo, es la imposibilidad de apagar por completo al transistor CMOS cuando éste está en el estado de corte.
- Así que conduce corriente por debajo del umbral de conducción.
- La compuerta se acopla al canal activo principalmente a través de la capacitancia de óxido de la compuerta.
- Hay otras capacitancias en un transistor que acoplan la compuerta del CMOS a una suerte de “carga fija”(una carga que no puede moverse) presente en el bloque y no asociada con el flujo de corriente.

leakage

- Como ya se dijo, es la imposibilidad de apagar por completo al transistor CMOS cuando éste está en el estado de corte.
- Así que conduce corriente por debajo del umbral de conducción.
- La compuerta se acopla al canal activo principalmente a través de la capacitancia de óxido de la compuerta.
- Hay otras capacitancias en un transistor que acoplan la compuerta del CMOS a una suerte de “carga fija”(una carga que no puede moverse) presente en el bloque y no asociada con el flujo de corriente.
- Si estas Capacidades son grandes pueden alterar la polarización de la compuerta cambiando la densidad de carga fija acumulada en el canal impidiendo que este corte.

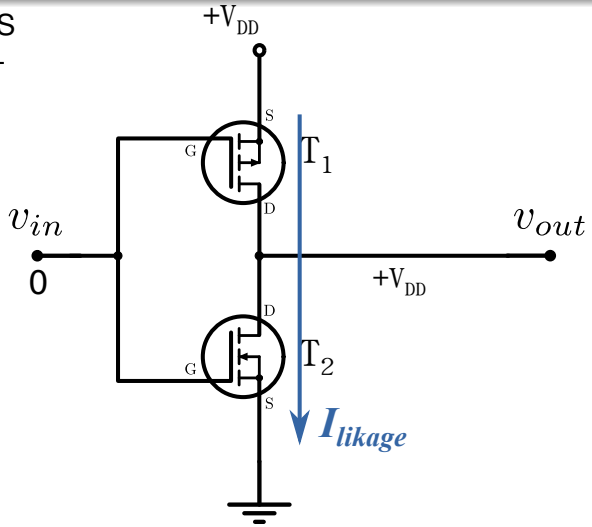
leakage

- Como ya se dijo, es la imposibilidad de apagar por completo al transistor CMOS cuando éste está en el estado de corte.
- Así que conduce corriente por debajo del umbral de conducción.
- La compuerta se acopla al canal activo principalmente a través de la capacitancia de óxido de la compuerta.
- Hay otras capacitancias en un transistor que acoplan la compuerta del CMOS a una suerte de “carga fija”(una carga que no puede moverse) presente en el bloque y no asociada con el flujo de corriente.
- Si estas Capacidades son grandes pueden alterar la polarización de la compuerta cambiando la densidad de carga fija acumulada en el canal impidiendo que este corte.
- Estas capacidades no se reducen con el scaling ya que no dependen de las dimensiones físicas del canal.

leakage

leakage

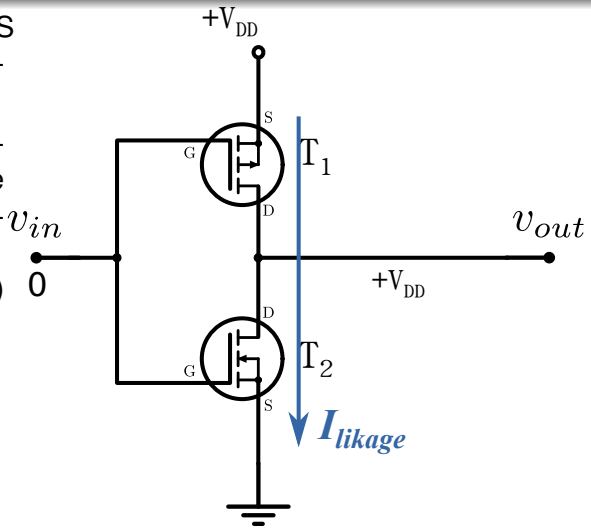
- Una vez terminada la conmutación el CMOS queda en un estado estable como el mostrado en la figura de la derecha.



leakage

- Una vez terminada la conmutación el CMOS queda en un estado estable como el mostrado en la figura de la derecha.
- La Potencia de leakage es linealmente dependiente de la tensión de alimentación, de acuerdo con una expresión bastante simple:

$$P_{stat} = I_{leakage}.V_{DD}, \qquad (15)$$

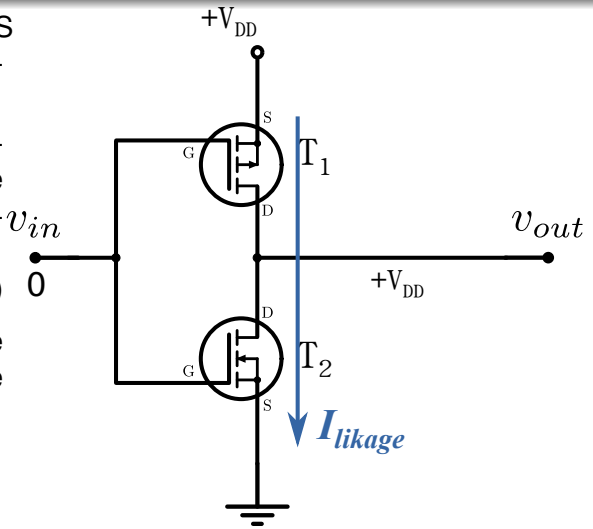


leakage

- Una vez terminada la conmutación el CMOS queda en un estado estable como el mostrado en la figura de la derecha.
- La Potencia de leakage es linealmente dependiente de la tensión de alimentación, de acuerdo con una expresión bastante simple:

$$P_{stat} = I_{leakage}.V_{DD}, \qquad (15)$$

- Y la Energía de leakage es el producto de la Potencia de leakage por el período de operación.

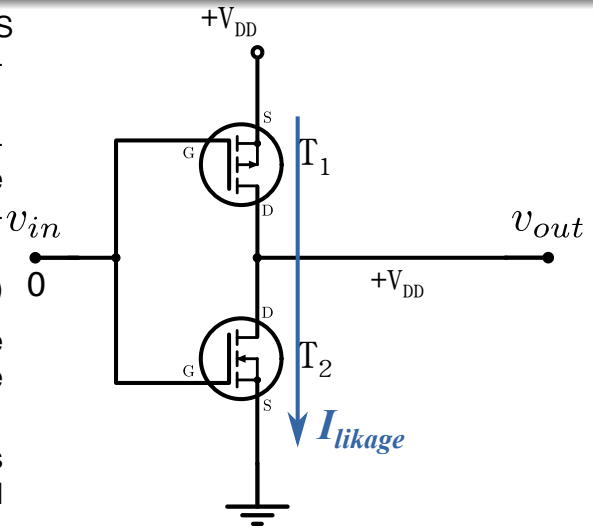


leakage

- Una vez terminada la conmutación el CMOS queda en un estado estable como el mostrado en la figura de la derecha.
- La Potencia de leakage es linealmente dependiente de la tensión de alimentación, de acuerdo con una expresión bastante simple:

$$P_{stat} = I_{leakage}.V_{DD}, \qquad (15)$$

- Y la Energía de leakage es el producto de la Potencia de leakage por el período de operación.
- Para calcularlas a nivel del chip todas las fórmulas se multiplican por la cantidad N de nodos.



leakage

leakage

- Las dimensiones del ancho del gate y del espesor la capa de óxido del transistor CMOS disminuyen linealmente con el avance en el scaling.

leakage

- Las dimensiones del ancho del gate y del espesor la capa de óxido del transistor CMOS disminuyen linealmente con el avance en el scaling.
- No así la tensión de alimentación.

leakage

- Las dimensiones del ancho del gate y del espesor la capa de óxido del transistor CMOS disminuyen linealmente con el avance en el scaling.
- No así la tensión de alimentación.
- Por ello la potencia disipada no disminuye con el scaling, ya que la cantidad de Nodo por área aumenta mas de lo que disminuyen algunos de los drivers de la Potencia Disipada.

leakage

- Las dimensiones del ancho del gate y del espesor la capa de óxido del transistor CMOS disminuyen linealmente con el avance en el scaling.
- No así la tensión de alimentación.
- Por ello la potencia disipada no disminuye con el scaling, ya que la cantidad de Nodo por área aumenta mas de lo que disminuyen algunos de los drivers de la Potencia Disipada.
- Si bien estos problemas son mas graves en los microprocesadores, han afectado también a las memorias.

Estrategias para reducir Potencia y Energía

Estrategias para reducir Potencia y Energía

- En resumen, la energía Total consumida por un chip está dada por:

$$E_{Tot} = [\frac{1}{2}.C_{tot}.V_{DD}^2.f + N_{tot}.I_{leakage}.V_{DD}].T \quad (16)$$

Estrategias para reducir Potencia y Energía

- En resumen, la energía Total consumida por un chip está dada por:

$$E_{Tot} = [\frac{1}{2}.C_{tot}.V_{DD}^2.f + N_{tot}.I_{leakage}.V_{DD}].T \quad (16)$$

- N_{tot} es la cantidad de nodos del chip, C_{tot} es la suma de todas las cargas capacitivas en los N_{tot} nodos, y T el período de operación, durante el cual, la corriente de leakage no cesa de drenar.

Estrategias para reducir Potencia y Energía

- En resumen, la energía Total consumida por un chip está dada por:

$$E_{Tot} = [\frac{1}{2}.C_{tot}.V_{DD}^2.f + N_{tot}.I_{leakage}.V_{DD}].T \quad (16)$$

- N_{tot} es la cantidad de nodos del chip, C_{tot} es la suma de todas las cargas capacitivas en los N_{tot} nodos, y T el período de operación, durante el cual, la corriente de leakage no cesa de drenar.
- Cuando la carga de trabajo es baja los circuitos integrados siguen conmutando aunque no cambien el estado de sus salidas, consumiendo energía para nada.

Estrategias para reducir Potencia y Energía

- En resumen, la energía Total consumida por un chip está dada por:

$$E_{Tot} = [\frac{1}{2}.C_{tot}.V_{DD}^2.f + N_{tot}.I_{leakage}.V_{DD}].T \quad (16)$$

- N_{tot} es la cantidad de nodos del chip, C_{tot} es la suma de todas las cargas capacitivas en los N_{tot} nodos, y T el período de operación, durante el cual, la corriente de leakage no cesa de drenar.
- Cuando la carga de trabajo es baja los circuitos integrados siguen conmutando aunque no cambien el estado de sus salidas, consumiendo energía para nada.
- Una solución es inhibir la frecuencia de clock (clock gating) en los momentos de baja carga para disminuir la pérdida de energía cuando el chip está inactivo.

Estrategias para reducir Potencia y Energía

- En resumen, la energía Total consumida por un chip está dada por:

$$E_{Tot} = [\frac{1}{2}.C_{tot}.V_{DD}^2.f + N_{tot}.I_{leakage}.V_{DD}].T \quad (16)$$

- N_{tot} es la cantidad de nodos del chip, C_{tot} es la suma de todas las cargas capacitivas en los N_{tot} nodos, y T el período de operación, durante el cual, la corriente de leakage no cesa de drenar.
- Cuando la carga de trabajo es baja los circuitos integrados siguen conmutando aunque no cambien el estado de sus salidas, consumiendo energía para nada.
- Una solución es inhibir la frecuencia de clock (clock gating) en los momentos de baja carga para disminuir la pérdida de energía cuando el chip está inactivo.
- Para bajar la componente estática solo queda disminuir V_{DD}.

Inversor en detalle

Símbolo

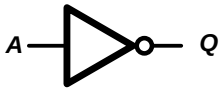
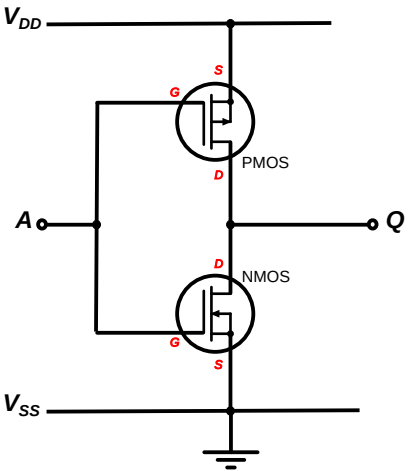


Tabla de Verdad

A	Q
0	1
1	0

Esquemático



Inversor en detalle

Símbolo

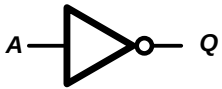
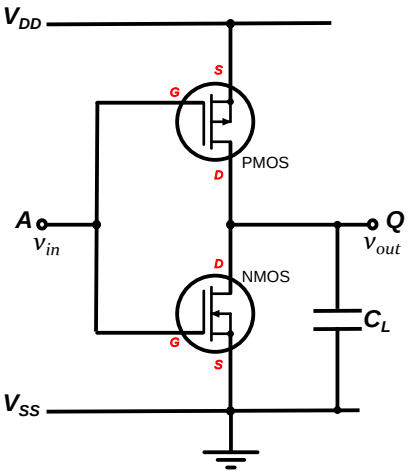


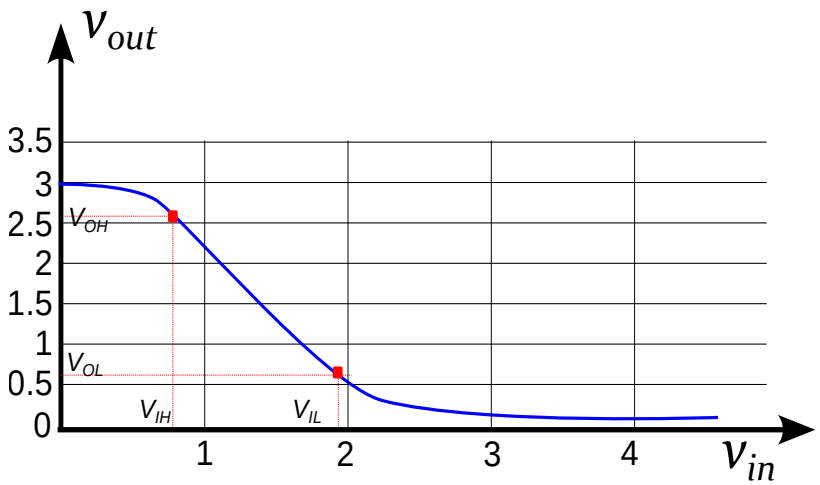
Tabla de Verdad

A	Q
0	1
1	0

Esquemático

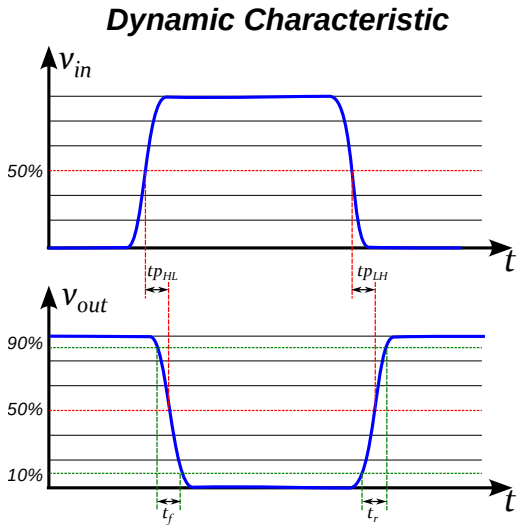


Inversor en detalle



Voltage Transfer Curve CTV

Inversor en detalle



tp_{HL}: Propagation Time High-Low

tp_{LH}: Propagation Time Low-High

Propagation Delay

$tp = (tp_{HL} + tp_{LH}:) / 2$

t_r: Rise Time (Excursión de10% a 90%)

t_f: Fall Time (Excursión de 90% a 10%)

Edge Rate

$t_{rf} = (t_f + t_r:) / 2$

Inversor three state

Símbolo

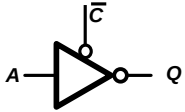
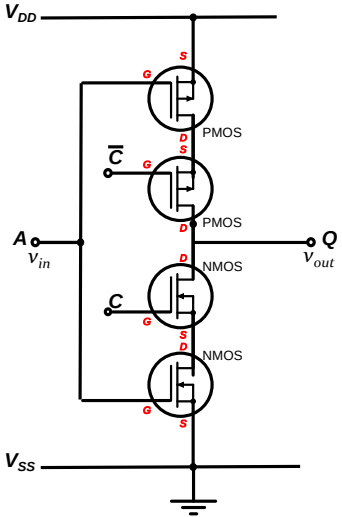


Tabla de Verdad

$\overline{C}$	A	Q
0	0	1
0	1	0
1	X	Hi-Z

Esquemático



Temario

Arquitectura vs Microarquitectura

Arquitectura

Es el conjunto de recursos accesibles para el programador, que por lo general se mantienen a lo largo de los diferentes modelos de procesadores de esa arquitectura (puede evolucionar pero la tendencia es mantener compatibilidad hacia los modelos anteriores).

Arquitectura vs Microarquitectura

Arquitectura

Es el conjunto de recursos accesibles para el programador, que por lo general se mantienen a lo largo de los diferentes modelos de procesadores de esa arquitectura (puede evolucionar pero la tendencia es mantener compatibilidad hacia los modelos anteriores).

- Registros

Arquitectura vs Microarquitectura

Arquitectura

Es el conjunto de recursos accesibles para el programador, que por lo general se mantienen a lo largo de los diferentes modelos de procesadores de esa arquitectura (puede evolucionar pero la tendencia es mantener compatibilidad hacia los modelos anteriores).

- Registros
- Set de instrucciones

Arquitectura vs Microarquitectura

Arquitectura

Es el conjunto de recursos accesibles para el programador, que por lo general se mantienen a lo largo de los diferentes modelos de procesadores de esa arquitectura (puede evolucionar pero la tendencia es mantener compatibilidad hacia los modelos anteriores).

- Registros
- Set de instrucciones
- Estructuras de memoria (descriptores de segmento y de página p. ej.)

Arquitectura vs Microarquitectura

Arquitectura

Es el conjunto de recursos accesibles para el programador, que por lo general se mantienen a lo largo de los diferentes modelos de procesadores de esa arquitectura (puede evolucionar pero la tendencia es mantener compatibilidad hacia los modelos anteriores).

- Registros
- Set de instrucciones
- Estructuras de memoria (descriptores de segmento y de página p. ej.)

Micro Arquitectura

Es la implementación en el silicio de la arquitectura. Lo que está detrás del set de registros y del modelo de programación. Puede ser muy simple o sumamente robusta y poderosa. Cambia de un modelo a otro dentro de una misma familia.

Arquitectura vs Microarquitectura

Ejemplo

La arquitectura IA-32 se inicia con el procesador 80386 en 1985, y llega hasta los procesadores Intel Core i7, i5, i3, ATOM y Xeon actuales.
En el camino han pasado diferentes generaciones de Micro-Arquitectura para mas de 25 modelos de procesadores.

Definición de la Arquitectura de un Computador

- Es sin dudas la tarea mas ardua de un diseñador.

Definición de la Arquitectura de un Computador

- Es sin dudas la tarea mas ardua de un diseñador.
- Se trata de definir los aspectos mas relevantes en la arquitectura de un computador que maximicen su rendimiento, sin dejar de satisfacer otras limitaciones impuestas por los usuarios, como un costo económico que lo haga accesible, o un consumo de energía moderado.

Definición de la Arquitectura de un Computador

- Es sin dudas la tarea mas ardua de un diseñador.
- Se trata de definir los aspectos mas relevantes en la arquitectura de un computador que maximicen su rendimiento, sin dejar de satisfacer otras limitaciones impuestas por los usuarios, como un costo económico que lo haga accesible, o un consumo de energía moderado.
- Comprende el diseño del set de instrucciones, el manejo de la memoria y sus modos de direccionamiento, los restantes bloques funcionales que componen el CPU, el diseño lógico, y el proceso de implementación

Definición de la Arquitectura de un Computador

- Es sin dudas la tarea mas ardua de un diseñador.
- Se trata de definir los aspectos mas relevantes en la arquitectura de un computador que maximicen su rendimiento, sin dejar de satisfacer otras limitaciones impuestas por los usuarios, como un costo económico que lo haga accesible, o un consumo de energía moderado.
- Comprende el diseño del set de instrucciones, el manejo de la memoria y sus modos de direccionamiento, los restantes bloques funcionales que componen el CPU, el diseño lógico, y el proceso de implementación
- La implementación comprende el diseño del circuito integrado, su encapsulado, montaje, alimentación y refrigeración.

Definición de la Arquitectura de un Computador

skills necesarios

No es posible realizar esta tarea con éxito sin tener manejar de manera solvente varias tecnologías diferentes:

Definición de la Arquitectura de un Computador

skills necesarios

No es posible realizar esta tarea con éxito sin tener manejar de manera solvente varias tecnologías diferentes:

- Diseño lógico.

Definición de la Arquitectura de un Computador

skills necesarios

No es posible realizar esta tarea con éxito sin tener manejar de manera solvente varias tecnologías diferentes:

- Diseño lógico.
- Tecnología de encapsulado

Definición de la Arquitectura de un Computador

skills necesarios

No es posible realizar esta tarea con éxito sin tener manejar de manera solvente varias tecnologías diferentes:

- Diseño lógico.
- Tecnología de encapsulado
- Funcionamiento y diseño de compiladores y de Sistemas Operativos.

Temario

Definiendo un ISA

Nos referimos a *Instruction Set Architecture*, como el set de instrucciones visibles por el programador. Es además el límite entre el software y el hardware.

Definiendo un ISA

Nos referimos a *Instruction Set Architecture*, como el set de instrucciones visibles por el programador. Es además el límite entre el software y el hardware.

Clases de ISA. ISAs con Registros de Propósito general vs. Registros dedicados. ISAs registro-memoria vs. ISAs Load Store.

Definiendo un ISA

Nos referimos a *Instruction Set Architecture*, como el set de instrucciones visibles por el programador. Es además el límite entre el software y el hardware.

Clases de ISA. ISAs con Registros de Propósito general vs. Registros dedicados. ISAs registro-memoria vs. ISAs Load Store.

Direccionamiento de Memoria. Alineación obligatoria de datos vs. administración de a bytes.

Definiendo un ISA

Nos referimos a *Instruction Set Architecture*, como el set de instrucciones visibles por el programador. Es además el límite entre el software y el hardware.

Clases de ISA. ISAs con Registros de Propósito general vs. Registros dedicados. ISAs registro-memoria vs. ISAs Load Store.

Direccionamiento de Memoria. Alineación obligatoria de datos vs. administración de a bytes.

Modos de Direccionamiento. Como se especifican los operandos.

Definiendo un ISA

Nos referimos a *Instruction Set Architecture*, como el set de instrucciones visibles por el programador. Es además el límite entre el software y el hardware.

Clases de ISA. ISAs con Registros de Propósito general vs. Registros dedicados. ISAs registro-memoria vs. ISAs Load Store.

Direccionamiento de Memoria. Alineación obligatoria de datos vs. administración de a bytes.

Modos de Direccionamiento. Como se especifican los operandos.

Tipos y tamaños de operandos. Enteros, Punto Flotante, Punto Fijo.Diferentes tamaños y precisiones.

Definiendo un ISA

Nos referimos a *Instruction Set Architecture*, como el set de instrucciones visibles por el programador. Es además el límite entre el software y el hardware.

Clases de ISA. ISAs con Registros de Propósito general vs. Registros dedicados. ISAs registro-memoria vs. ISAs Load Store.

Direccionamiento de Memoria. Alineación obligatoria de datos vs. administración de a bytes.

Modos de Direccionamiento. Como se especifican los operandos.

Tipos y tamaños de operandos. Enteros, Punto Flotante, Punto Fijo.Diferentes tamaños y precisiones.

Operaciones. Pocas Simples (RISC) o Muchas Complejas (CISC).

Definiendo un ISA

Nos referimos a *Instruction Set Architecture*, como el set de instrucciones visibles por el programador. Es además el límite entre el software y el hardware.

Clases de ISA. ISAs con Registros de Propósito general vs. Registros dedicados. ISAs registro-memoria vs. ISAs Load Store.

Direccionamiento de Memoria. Alineación obligatoria de datos vs. administración de a bytes.

Modos de Direccionamiento. Como se especifican los operandos.

Tipos y tamaños de operandos. Enteros, Punto Flotante, Punto Fijo.Diferentes tamaños y precisiones.

Operaciones. Pocas Simples (RISC) o Muchas Complejas (CISC).

Instrucciones de control de flujo Saltos condicionales, calls.

Definiendo un ISA

Nos referimos a *Instruction Set Architecture*, como el set de instrucciones visibles por el programador. Es además el límite entre el software y el hardware.

Clases de ISA. ISAs con Registros de Propósito general vs. Registros dedicados. ISAs registro-memoria vs. ISAs Load Store.

Direccionamiento de Memoria. Alineación obligatoria de datos vs. administración de a bytes.

Modos de Direccionamiento. Como se especifican los operandos.

Tipos y tamaños de operandos. Enteros, Punto Flotante, Punto Fijo.Diferentes tamaños y precisiones.

Operaciones. Pocas Simples (RISC) o Muchas Complejas (CISC).

Instrucciones de control de flujo Saltos condicionales, calls.

Longitud del código Instrucciones de tamaño fijo vs. variable.

Temario

Microarquitectura = Organización + Hardware

Organización

Se refiere a los detalles de implementación de la ISA. Es decir

- Organización e interconexión de la memoria.

Microarquitectura = Organización + Hardware

Organización

Se refiere a los detalles de implementación de la ISA. Es decir

- Organización e interconexión de la memoria.
- Diseño de los diferentes bloques de la CPU y para implementar el set de instrucciones.

Microarquitectura = Organización + Hardware

Organización

Se refiere a los detalles de implementación de la ISA. Es decir

- Organización e interconexión de la memoria.
- Diseño de los diferentes bloques de la CPU y para implementar el set de instrucciones.
- Implementación de paralelismo a nivel de Instrucciones, y/o de datos.

Microarquitectura = Organización + Hardware

Organización

Se refiere a los detalles de implementación de la ISA. Es decir

- Organización e interconexión de la memoria.
- Diseño de los diferentes bloques de la CPU y para implementar el set de instrucciones.
- Implementación de paralelismo a nivel de Instrucciones, y/o de datos.
- Podemos así encontrar procesadores que poseen la misma ISA pero una organización muy diferente.

Microarquitectura = Organización + Hardware

Organización

Se refiere a los detalles de implementación de la ISA. Es decir

- Organización e interconexión de la memoria.
- Diseño de los diferentes bloques de la CPU y para implementar el set de instrucciones.
- Implementación de paralelismo a nivel de Instrucciones, y/o de datos.
- Podemos así encontrar procesadores que poseen la misma ISA pero una organización muy diferente.

Ejemplo:

Los procesadores AMD FX y los Intel Core i7, tienen la misma ISA, sin embargo organizan su caché y su motor de ejecución de maneras diferentes.

Microarquitectura = Organización + Hardware

Hardware

Microarquitectura = Organización + Hardware

Hardware

- Se refiere a los detalles de diseño lógico y tecnología de fabricación.

Microarquitectura = Organización + Hardware

Hardware

- Se refiere a los detalles de diseño lógico y tecnología de fabricación.
- Existirán por lo tanto, procesadores con la misma ISA y la misma organización, pero absolutamente distintos a nivel de hardware y diseño lógico detallado.

Microarquitectura = Organización + Hardware

Hardware

- Se refiere a los detalles de diseño lógico y tecnología de fabricación.
- Existirán por lo tanto, procesadores con la misma ISA y la misma organización, pero absolutamente distintos a nivel de hardware y diseño lógico detallado.

Ejemplo:

El Pentium 4, diseñado para desktop, y el Pentium 4 M para notebooks. Este últiimo tiene una cantidad de lógica para control del consumo de energía, sindo su hardware muy diferente del del Pentium 4 desktop.

¿Porque necesitamos saber todo esto?

¿Porque necesitamos saber todo esto?

- Para comprender lo que ocurre debajo del software.

¿Porque necesitamos saber todo esto?

- Para comprender lo que ocurre debajo del software.
- Comprender como los diseños de hardware impactan en el software y en el programador

¿Porque necesitamos saber todo esto?

- Para comprender lo que ocurre debajo del software.
- Comprender como los diseños de hardware impactan en el software y en el programador
- Estar en condiciones de cruzar verticalmente las capas que separan el silicio de la aplicación.

¿Porque necesitamos saber todo esto?

- Para comprender lo que ocurre debajo del software.
- Comprender como los diseños de hardware impactan en el software y en el programador
- Estar en condiciones de cruzar verticalmente las capas que separan el silicio de la aplicación.
- Compender las nociones básicas que rigen el diseño de un sistema de cómputo moderno

¿Preguntas?