Power and power distribution

Power may be important for two reasons (at least)

✦ Energy demands on batteries for mobile equipment
✦ Cooling requirements

[Also general ‘Green’ efficiency issues.]

The first issue is to get adequate power in to where it’s needed

✦ The current requirement means there will be a supply voltage drop
✦ Modelled/mapped by CAD tools
✦ May want extra power pins

The second issue is to get the waste (heat) out

✦ Potential ‘hot spots’ may be identified by CAD tools
✦ Better to have high-power demands near to supplies
✦ Overall power dissipation is a concern e.g. for packaging

Power, Energy and Moore’s law.

As devices continue to shrink their energy demand per operation reduces.
Roughly speaking:

- transistor gate length decreases ⇒ capacitance reduced ⇒ energy reduced
- transistor gate width decreases ⇒ capacitance reduced ⇒ energy reduced
- transistor gate spacing decreases ⇒ capacitance increased ⇒ energy increased
- local wiring length decreases ⇒ capacitance reduced ⇒ energy reduced
- more units integrated on chip ⇒ energy increased on one chip ⇒ power density increased
- clock speed(s) increase ⇒ power increased
- operating voltage decreases ⇒ energy decreased

Voltage drop

All active circuits have some demand for electric current: this depends on their capacitive load and their switching rate. Load can be extracted from the layout; the activity factor (α) for the logic may be simply estimated or derived from activity in a ‘typical’ set of functional simulations.

The power distribution network is made of metal wires which have some inherent resistance. This means that there is a voltage drop from the supply at the pad ring to the circuit itself.

[The figure assumes the power is supplied from the chip’s periphery, which is typical.]

Thus the supply voltage at the circuit will be less that the original specification. This means the circuit is operating under (slightly) different conditions from those intended.

These differences must be kept within controlled bounds for the system to work correctly. CAD tools allow the estimation of the voltage drop, taking into account the logic circuits, their activity and the power distribution wiring. This can translate into a map showing where problems may occur. The response is usually to supplement wiring to those areas although, more drastically, a different floor-plan may be required.

Decoupling

CMOS’ power demand is not constant but varies with the clock cycle; following a clock edge there lots of switching activity, which tends to diminish until the next clock edge. This means the current in the power rails is not constant.

Self-inductance of power supply wires resists change in current flow, thus a local demand for current will cause a drop in supply voltage (with a complementary spike as it switches ‘off’ again). To help suppress this noise ‘decoupling’ (or ‘bypass’) capacitors are provided across the power rails – as near to the demand as possible – to act as charge ‘tanks’. It is difficult to build sizeable capacitors on chip but, typically, many capacitors are scattered close to the device on the circuit board.

Some definitions

It is normal to use subscripts to locate particular parameters: thus ‘Vd’ is used for the voltage of the drain of a transistor.

The convention is to ‘double’ the subscript to designate the power supply to such terminals, thus ‘Vdd’ is the drain supply voltage (or ‘power’ in the vernacular). Similarly ‘Vcc’ (collector supply voltage) is sometimes used for the ‘low’ supply rail, although ‘ground’/GND’ is often interchanged.

‘Vcc’ (collector supply voltage) is still seen occasionally (and incorrectly): this is analogous to Vdd in bipolar transistor technology.

1. If this is jargon, don’t worry – just accept the description.
Where the power goes in CMOS logic

Energy = work done. Power = work done per unit time

The dominant dissipation in active (switching) logic is the dynamic dissipation due to charging/discharging load capacitance. This depends on the network load, the square of the supply voltage and the amount of activity.

The other dynamic component depends on the supply voltage, the transistor thresholds and the input edge speed.

Static (leakage) currents depend on the supply voltage and (for sub-threshold leakage) the transistor thresholds.

['α' is the 'activity factor', representing the probability of a signal switching on a given clock edge.]

### Dynamic power dissipation

Dynamic dissipation is that which is caused by switching gates. It is therefore proportional to the clock frequency; at double the frequency gates switch twice as often so power is doubled. It is also related to the activity in the circuit: not every gate will switch on every cycle (except for the clock drivers) and, sometimes, logic blocks can be idle for long periods.

Dynamic dissipation in CMOS circuits is made up from two components: the first component is related to the load capacitance of a network. To switch a node to 'high' it is connected – via a pull-up tree – to the power supply and charge flows from there into the net (wires and transistors) until the voltage equalises. This charge is defined by: \[ Q = C.V \]

When the node discharges via the pull-down tree the charge flows to ground. The effect of the cycle is to move charge \( Q \) through the potential between the rails \( V \), so the energy required/dissipated per cycle is: \[ E = Q \cdot V = C.V^2 \]

This term does not depend on edge speeds, etc. Note, particularly, that this is proportional to the square of the voltage: halving the supply voltage will reduce the energy dissipation by a factor of four.

In more detail, this assumes a rail-to-rail swing of the signal.

In some cases this change can be reduced, saving energy, but other problems may ensue.

The other dynamic dissipation comes from 'short circuit' or 'crowbar' current which results from both the pull-up and pull-down trees conducting (to some extent) as one turns off and the other turns on. This is difficult to quantify in general as it depends on a number of factors, including transistor widths and which transistors are switching. However there are two 'simple' ways to reduce this:

- keep input edges fast – reduce the time both trees are on simultaneously.
- Use high threshold transistors – these will turn ‘on’ later and ‘off’ sooner, as appropriate, minimising time spent when charge is wasted. However this slows down switching and increases the propagation delay of the gate. This is a strategy better employed on gates off the critical path.

### Static power dissipation

Static power dissipation happens continuously, regardless of switching activity. Historically it has been close to zero and therefore negligible except in very low power systems which are intended to do very little activity over very long periods using tiny batteries. However this source of inefficiency is increasing with technology advances.

Static dissipation is basically charge leakage through paths where charge is not supposed to go.

Subthreshold leakage is current through a transistor which is ‘turned off’. Transistors are not switches and will conduct to some extent at all times. A small current therefore flows through a gate even when it is not switching. Subthreshold leakage can be reduced by having a high transistor threshold.

Gate leakage is the leakage of charge from the transistor’s gate to the silicon channel. Electrons go through the thin insulating (‘oxide’) layer by quantum tunnelling; this effect increases exponentially with distance and so is an increasing problem as geometries shrink: the insulator thickness diminishes as well as the ‘horizontal’ dimensions.

Static power dissipation is still relatively small for most logic circuits but can account for a significant proportion of the power in memories, which have many transistors in a small space, most of which switch very infrequently.

Static dissipation in logic can be reduced by power gating: placing a high threshold transistor in series with the power supply which is switched ‘off’ when a set of gates is not in use. See later notes.

---

1. Not usually SiO₂ any more.
Transistor thresholds

A CMOS transistor may be regarded as a switch. (For digital purposes.)

The threshold of a transistor is the gate voltage where it switches from ‘off’ to ‘on’

- This voltage is measured from the gate to the channel (source or drain)

- The threshold can be determined when the transistor is manufactured.

- Applying different dopant concentrations allows the construction of different threshold devices – now have selections on a single chip.

Note: the transistor threshold is not the same as the gate threshold!
(i.e. where a logic element differentiates a logic ‘0’ and ‘1’.)

Transistor thresholds

Different thresholds can be used to control energy flow.

Inverter response to a (slow – for clarity) input edge:

With low threshold transistors the NMOS pull-down switches ‘on’ earlier so the output edge starts sooner. The PMOS pull-up switches ‘off’ later so there is a significant time when they’re both ‘on’ allowing ‘short circuit’ current to flow between the power rails. This wastes energy. Fast but power-hungry.

With high threshold transistors the NMOS pull-down switches ‘on’ later so the output edge is delayed. The PMOS pull-up switches ‘off’ earlier so there is little (or possibly no) time when they’re both ‘on’, minimising ‘short circuit’ current. Low power but slow.

‘Floating’ inputs

CMOS inputs should always be held at a logic level. If an input is not connected (“floating”) it has a (very) high impedance and may adopt just about any voltage. If it floats to somewhere between the logic levels then both the transistor stacks may turn ‘on’ (to some degree) and current ‘crowbars’ between the rails. This can be a significant power waste.

As long as the inputs are defined logic levels this is avoided and only the much smaller subthreshold leakage flows.

The threshold of a FET depends on the dopant concentrations in the channel and ‘diffusion’ implant regions. These are set by how many impurity atoms are embedded in the silicon substrate. This is a statistical process as atoms impact the surface.

F o r a g i v e n m e a n d o p i n g c o n c e n t r a t i o n , a s t r a n s i s t o r a r e a s d e c r e a s e t h e n u m b e r o f a t o m s d e c r e a s e s . T h i s m e a n s t h e d i f f e r e n c e o f a f e w a t o m s m o r e s i g n i f i c a n t o v e r a l l .

The result is that the manufactured threshold voltages vary over an increasingly wide range as Moore’s Law goes forwards.

Gate threshold

Do not confuse the transistor threshold with a logic gate’s threshold. The gate threshold is the input voltage where the whole gate will switch its output’s logic state. This is typically around half the supply voltage. It will vary according to the gate’s internal structure and may be (slightly) different on different gate inputs due to the position of connected transistors in the ‘stacks’.

Variation in gate threshold is a good reason to keep signal edges fast – to minimise differences in when they are perceived to change.
Transistors with different thresholds

A silicon foundry may offer a range of transistor thresholds, some subset of which may be available on a single SoC. E.g.

- **High-speed**: low threshold, fast switching, leaky transistors
- **‘Normal’**: compromise
- **Low-power**: high threshold, low leakage, slow switching

These may be selected as appropriate in a design.

Example: use high-speed gates on the critical path and low-power gates away from there.

Low-leakage devices can also be used to **power-gate** high-speed logic.

- **Power ‘off’** to curtail leakage when not in use
- **Power ‘on’** when required
  - switch on necessarily slow
    (slow transistor plus time to repower gated supply)

Power gating involves the creation of a separate, local supply rail for a gate of, more probably, a block of gates. This is supplied from the ‘real’ supply via a (large) **pass transistor**.

When active the pass transistor conducts and the gate will switch with (close to) the properties of the logic transistors. (There will be a slight penalty due to the additional channel impedance.)

When the logic is inactive the power can be cut off (the pass transistor is high-threshold so its leakage is small) and the remaining charge will leak to ground, after which leakage is minimised.

The slide shows a PMOS power gate: in practice, if a single transistor is used, it is more likely to be an NMOS device because it can be smaller for the same transconductance. It is even possible to gate both supply rails to choke leakage even further.

The threshold voltage is affected by the bulk silicon or ‘body’ voltage. It is possible to **bias** the body to affect the switching characteristics of the transistor. This is known as the **‘body effect’**.

Explicit **back gates** have also been proposed: these could be switched together with the ‘front’ gate or, perhaps more easily, used to bias the channel and alter the transistor’s threshold.

Rather than have a planar transistor it is possible to ‘surround’ the channel with the gate. This is considerably more complex to manufacture but gives better control of the electric field in the channel.

The results are higher switching speed and lower subthreshold leakage.

Although there was a specific original meaning, such devices—which come in a variety of configurations—tend to all be known as **FinFETs**.

These, and similar, structures are of great interest both for fast switching and for leakage reduction.

Ultra-low power applications

Very low power applications may have to work for years – possibly the lifetime of the device – of a small button cell. Alternatively they may need to run be energy harvesting from the environment (for example picking up energy from vibrations).

These applications are often (very) low-performance so the dominant energy dissipation mode may be static leakage.

A long-established low-power application is the wrist watch. A contemporary example is the new (remotely read) gas meters where a single cell has to sustain 10+ years of operation.

The design philosophy of this type of device is somewhat different from the typical, drive-for-more-performance-computing devices, but equally valid.
Power domains

Extending the idea of power gating, different parts of a SoC may be powered from different voltages.

Slower regions can be run off reduced voltage rails.

Problems

- More supply distribution networks
- Different signal levels need to cross borders between domains.

Domains

On a SoC it may be appropriate to run different parts of the device under different conditions, such as different supply voltages.

Lower supply voltages typically mean slower switching speeds but that may be acceptable for some units on a SoC. Lower voltages will also reduce power consumption.

A problem encountered with different supply voltages is the interpretation of signals at domain boundaries. (This is similar to the problem of crossing boundaries between differently clocked domains, but can be resolved reliably.)

Clock Gating

A clock distribution network is a high-load net (it fans out to many places) which switches frequently (twice per clock cycle). It therefore can be a major contributor to power dissipation.

It is now quite common to employ clock-gating to blocks which are idle, even if the rest of the logic is not switching. CAD tools will support this. Note – DIY gating is dangerous: clock gating must not introduce glitches when turned on and off, nor impose arbitrary clock skew.

The power can remain on so register/memory state is retained.

Example

It is not uncommon for an embedded processor core to go into a gated ‘sleep’ mode whilst waiting for an interrupt; the active interrupt signal restarts the clock so processing resumes.

“Dark Silicon”

The ultimate power-saving option is to switch off. “Dark Silicon” is a term coined to refer to parts of a system which are present but may be unpowered.

By switching off domains of an SoC which are not in use at a particular time all the dissipation can be avoided, including leakage currents.

This may be necessary – rather than just desirable – in some devices to stop the power dissipation exceeding that which keeps the temperature within operating limits.

There are some disadvantages to this strategy: from the logic point of view it must be ensured that signals which emerge from a powered-down domain are clamped to harmless values. More importantly, when switched off all the register and memory (RAM) state is lost so there may be considerable work in restoring context when switched on again.

(Future, non-volatile storage may address this, one day.)

This technology may be expected proliferate as many-core processors become common and the need to maximise processor utilisation is no longer a major consideration.

Example

ARM have a mechanism called ‘big.LITTLE™’ which employs two code-compatible ARM cores: a high-performance (high-power) one and a low-performance (low-power) device. The high-performance core is turned on only when processing demand merits it and execution is then migrated to it.

Note that, to achieve high performance, various mechanisms such as speculation may be employed which use energy for acceleration rather than evaluating the actual problem. For example, deeper pipelines will process more items which are, sometimes flushed and discarded.
Level shifting

Running from different supplies can cause problems:

Static power is dissipated … output ‘level’ may not fully switch next gate …

Crossing voltage domains

Normally ‘ground’ is a common reference so, when crossing between domains with different supply rails it is the definition of ‘high’ which will vary. The primary problem occurs when an input ‘high’ (say 0.8 V) may be below the threshold of a PMOS transistor in a higher voltage domain. In this case the PMOS transistor will not turn ‘off’. Even if the gate reads this signal as a ‘high’ there will be a continuous current drawn from the supply, dissipating power in the interface transistors.

In the case on the slide the output (‘low’) might be high enough to partially turn on the next NMOS transistor(s) too, propagating the problem further.

What is needed is a reliable way of achieving the output levels in the new supply domain.
Level shifter

Need to transform a ‘high’ when moving to a different supply domain.

Here’s one way:

![Diagram of level shifter](https://example.com/diagram)

In this design a ‘low’ is always 0.0 V so that needs no translation.

---

Level shifter designs are varied. The slide shows a typical structure. In operation the input pulls one side of the differential amplifier low whilst not affecting the other. This causes one of the PMOS transistors to turn ‘on’ whereas the other is not sure. The ‘turned on’ PFET has nothing to oppose it and will pull up the gate of the other, turning it ‘off’. This prevents further current flow through the NFETs whilst the output (or its inverse) is clamped to the higher voltage supply.

To allow you to become convinced, here are some copies to doodle with.

![Alternative level shifters](https://example.com/alternatives)

This is not the fastest level shifter but it illustrates how the job can be done.

---

Alternatives

There are many other level shifter designs. You might like to discover and sketch some here:
Dynamic Voltage/Frequency Scaling

- Power dissipation can be reduced (or increased) by reducing (increasing) clock frequency.
- Energy dissipation can be reduced by reducing supply voltage
- Reducing supply voltage makes logic evaluate more slowly

If there is not much work to do:
- Reduce clock frequency – lower power but same energy/computation
- Reduce supply voltage – lower energy/computation

When there is a high work demand:
- Increase supply voltage
- Increase clock frequency

A similar approach may be employed if the power dissipation (i.e. temperature) gets too high.

---

Dynamic Voltage and Frequency Scaling

DVFS (Dynamic Voltage and Frequency Scaling) is, perhaps, currently falling out of favour because supply voltages are reducing such that there is not much 'headroom' for scaling. However the principle works like this:

- Power consumption in logic circuits is chiefly dynamic.
- Dynamic power consumption is due to energy expenditure during switching.
  - Thus the energy per operation is constant. Halving the clock frequency will halve the power but the job takes twice as long; the energy (demand on a battery) is the same.
- Dynamic energy consumption is dominated by the $CV^2$ term.
- Halving the supply voltage divides this energy demand by four.
  - … as long as the circuit still switches fast enough that this term dominates: go too low and this assumption is dubious.
- Reducing the supply voltage slows the circuit. Supply voltage and speed are (very) roughly proportional.

So, if it is allowable to run more slowly:

- Reduce the clock frequency by a factor $\alpha$ …
- Reduce the power supply by a factor $\alpha$ …
- … reduces energy by $\alpha^2$.
- … increases time taken by $\frac{1}{\alpha}$

This can be applied dynamically if the workload can be predicted, slowing down when there's no rush and speeding up when the demand is high.

Note: reduce frequency before reducing supply voltage. Increase supply voltage before increasing frequency. (If you want it to work correctly?)