Microelectronics System Design Research Group
School of Electrical and Electronic Engineering
Merz Court
Newcastle University
Newcastle upon Tyne, NE1 7RU, UK

http://async.org.uk/



# Voltage, throughput, power, reliability and multi-core scaling

Fei Xia

fei.xia@ncl.ac.uk

NCL-EECE-MSD-MEMO-2013-008

October 2013

**Abstract**

This memo tries to relate the crucial parameters voltage, throughput, power and reliability in typical CMOS systems and studies them in the environment of multi-core scaling. Example CMOS systems used in this study include a low-power microprocessor and an asynchronous SRAM. The region of reliable operation is defined and its shape analyzed under different power, voltage, throughput and scaling factor conditions to create a general picture of how these issues interplay.

It has been general knowledge that in digital CMOS systems, a higher supply voltage (Vdd) usually allows a higher operating (clock) frequency and hence a higher throughput. The scheme of DVFS scales Vdd and clock together in order to obtain the best throughput possible under a given quantity of supply power or to save power for a given throughput requirement.

It has also been generally known that when power is limited, it is possible to obtain an increase in system throughput by scaling to multiple computation units – cores when we talk about processors, if the computation can be reasonably parallelized and spread out to these cores. Parallelization in this fashion could also be used to tackle the related problem of reducing power consumption while faced with a certain throughput requirement.

This type of core scaling has been known to produce the best advantage when Vdd is scaled down to just above the threshold voltage of the CMOS node concerned. This is known as near threshold computing (NTC), which maximally takes advantage of the core scaling without entering the sub-threshold region, where the pictures change and the advantages of core scaling become progressively outweighed by disadvantages brought by variability.

These general knowledge points were obtained usually by considering reliability as a separate issue, i.e. people have studied these problems whilst assuming reliability is 100%. This document sets out to study the inter-relationship of all these issues together, namely voltage, throughput, power, reliability and multi-core scaling.

## The region of reliable operation

In order to retain reliability, a system must operate within certain constraints. For instance, a particular hardware implementation may not behave correctly if the supply voltage Vdd goes below or above certain values. Different hardware components in a system may have different minimum and maximum Vdd values and this means that software components, depending on how they are mapped onto hardware, could also be constrained by different Vdd limits, even within the same system.

Another type of constraint is the performance/throughput requirement specification. A system or a part of a system may be required to execute at least a certain level of throughput for the application to be meaningful.

A third type of constraint is the power supply limitation. With mobile and embedded systems whose energy and power are drawn from limited or uncertain sources, the amount of available power limits the behaviour of the system.

The minimum latency, which is related to computation throughput, of any specific hardware logic is related to the Vdd supplied to it. This means that if this logic is run on a clock too fast for a certain Vdd the computation would not complete in time before the next clock pulse arrives, which usually leads to unreliable or unusable results. A common technique for determining the appropriate clock frequency for computation logic is to first determine the critical path delay of the logic under the given Vdd condition, add enough delay margins to account for potential effects of process, voltage and temperature variations, and then invert the result to obtain the frequency.

Because of this traditionally well-understood interplay relation between frequency/throughput and voltage, we will explore the concept of the region of reliable operation in the context of a Vdd – throughput space.

From the intuitive discussions above, the region of operation for a system within the Vdd – throughput space is bounded by constraints on power, timing reliability, minimum throughput requirements and low and high Vdd boundaries.

Within the Vdd – throughput space, boundaries related to constant values of Vdd and throughput are straightforward.

The minimum and maximum Vdd boundaries can be described by

$$\text{minVdd} \leq \text{Vdd} \leq \text{maxVdd} \quad (1)$$

The minimum throughput requirement can be described by

$$\text{Throughput} \geq \text{minThroughput} \quad (2)$$

The power limit and timing reliability boundaries, however, are not as easy to obtain. On the other hand, their general shapes can be derived from theories on semiconductor characteristics. For instance, the timing reliability boundary within a Vdd – throughput/frequency space usually starts off at very low Vdd with the frequency climbing from almost zero upwards exponentially until Vdd increases to around the threshold voltage, from there upwards the increase in frequency is more or less linear, until usually when Vdd is around the nominal Vdd of the technology in question where the frequency increase starts to saturate.

From these qualitative arguments, the region of reliable operation can be illustrated by

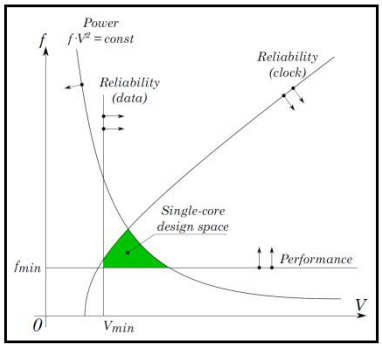


Figure 1 the region of reliable operation.

As can be seen, for reliable operation, the system's throughput must be restricted below the timing (clock) reliability and power limit lines and must be above the throughput requirement line. The system's Vdd must also be restricted between the high and low Vdd limits.

## Exploring the issue of system hardware scaling for super-threshold operations

*Super-threshold operations*

In the super-threshold region, we can hypothesize that dynamic power dominates power consumption and leakage power and its influence can be ignored.

Dynamic power is known to be related to switching activity (and through which to system frequency), switching swing voltage (and through which to system Vdd) and switching element capacitance (and through which to system size/area – which is a constant before hardware scaling). Lumping all the constants together we can say that power is related to frequency and Vdd in the following manner:

$$P = A \times F \times Vdd^2 \quad (3)$$

where $A$ is a constant, $P$ the power and $F$ the frequency.

Experimental data with power, Vdd, and frequency can be found for a variety of systems and here we use data collected from a low-power SRAM implementation, which contains a 1kb SRAM bank and its asynchronous control logic to investigate the region of reliable operation. This implementation is in UMC 90nm technology with a nominal Vdd of 1.2V.

In order to find the Vdd range where the assumption that dynamic power dominates applies, we put the total power consumption, frequency and Vdd data into equation (2) to solve for A. Figure 2 gives the results.

As can be seen, $A$ remains relatively constant in the range of $0.6V \leq Vdd \leq 1.2V$, with the difference between the highest and lowest $A$ values being less than 8%. This means that dynamic power dominates in this range and the power limit boundary can be estimated from equation (3) directly.

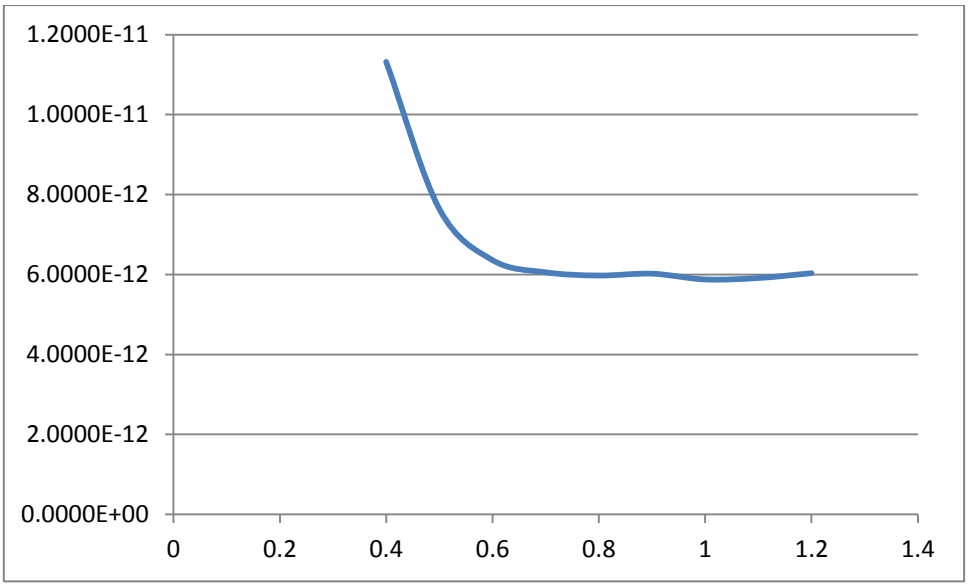


Figure 2 Values of A across Vdd.

This is resulted from trying to estimate $A$ from the total power, whereas it is a dynamic power parameter only. When the total power is approximately the same as the dynamic power, i.e. leakage power being negligible, this estimation is reasonably accurate. However, when leakage power cannot be ignored, the $A$ value obtained this way is meaningless. Under this light, we can say that the $A$ values plotted in Figure 2 are meaningful when Vdd is 0.6V and above. It is perhaps OK when Vdd is 0.5V but certainly cannot be used below that.

Using the $A$ value obtained this way ($A$=6.0335E-12), we can plot constant power curves across the timing condition curve, as shown in Figure 3.

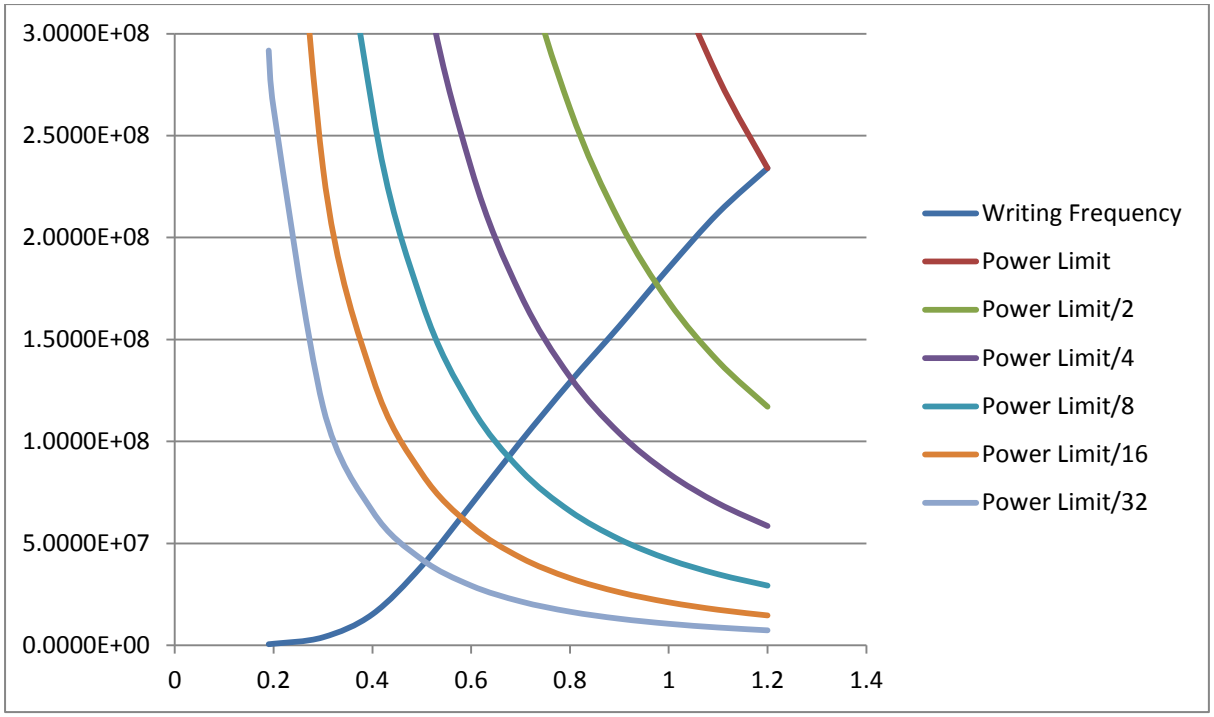


Figure 3 Timing closure with power limits.

Figure 3 shows a sequence of power limit curves, starting from maximum usable power, which allows the system to operate at nominal Vdd and at its highest safe frequency at this Vdd, and half, a quarter, 1/8, etc. down to 1/32 of this amount of power. There is no point to continue these plots with scaling factors greater than 32 because as discussed above, this will extend the border of the reliable operating region contributed by the power line to below 0.5V, where the constant power lines derived from the *A* values given in Figure 2 become too inaccurate.

Also included in Figure 3 is the maximum frequency vs. Vdd relationship data obtained from the SRAM. The shape of this curve conforms with the qualitative expectations described in the previous section. This means that although this is memory and not a processor, we can explore processor scaling with this data without losing generality. As can be seen the maximum frequency at which the system can be operated reduces when the power limit is lowered. The reliable operation region (here without considering Vdd and throughput limits) is also reduced when the power limit is lowered.

The most throughput-efficient operating point under each constant power setting is the intersection between the timing limit curve and power limit curve. Because of the roughly linear nature of the timing limit curve and the quadratic shapes of the power limit curves, reducing power by half does not reduce frequency (or computational throughput) also by half. The reduction in throughput is smaller. As a rule of thumb, in order to reduce throughput by half, power needs to be reduced by ¾. Note that Figure 3 has linear-scale axes on both directions.

How does scaling to larger hardware sizes (more cores) affect this relationship between system power limit and highest computation throughput?

In this work, we explore the issue of scaling with an assumption of perfect scaling. In other words, the throughput and power overheads associated with hardware scaling are both assumed to be zero. Multiplied hardware operating at the same frequency will provide multiplied throughput and require a

multiplied amount of power based on the same constant multiplier. In a later section we will discuss non-zero scaling overheads.

First of all, in perfect scaling with a scaling factor of *N*, the constant *A* is scaled in the same way, i.e.

$$A = N \times A_1 \quad (4)$$

where $A_1$ is the *A* for the hardware before scaling (e.g. a single core). In general, scaling with a factor of *N* will give a new *A* which is a factor of *N* of the unscaled *A*.

For each core in a new scaled set-up, the available power is also changed by a factor of 1/*N*. Considering these factors, for each core, equation (3) now becomes

$$P = A \times F \times Vdd^2 / N \quad (5)$$

and the overall system power equation with *N* cores stays the same as (3).

The cumulative effect of scaling on the maximum system throughput, given the same power limit and a scaling factor of *N*, is as follows:

Each core will have its power reduced by a factor of *N*, which reduces its maximal throughput by a factor of less than *N* (recall the earlier observation that scaling power down by a factor of 4 reduces maximal throughput by a factor of 2). Then this performance multiplied across *N* cores provides a net increase of usable throughput. An example of this is shown in Figure 4.

From a starting point of a fixed power budget, which is enough to operate a single core at the nominal Vdd and maximum frequency without losing timing correctness, the system is scaled up to 4 and 16 cores. With 4 cores, each core would operate under ¼ of the total power budget, this roughly corresponds with 0.8V and 129MHz. All 4 cores together gives 0.8V and a throughput of 516MHz, a vast improvement over the single-core's capability of 234MHz at 1.2V. Scaling up to 16 cores will see the system working at around 0.6V and over 1GHz.

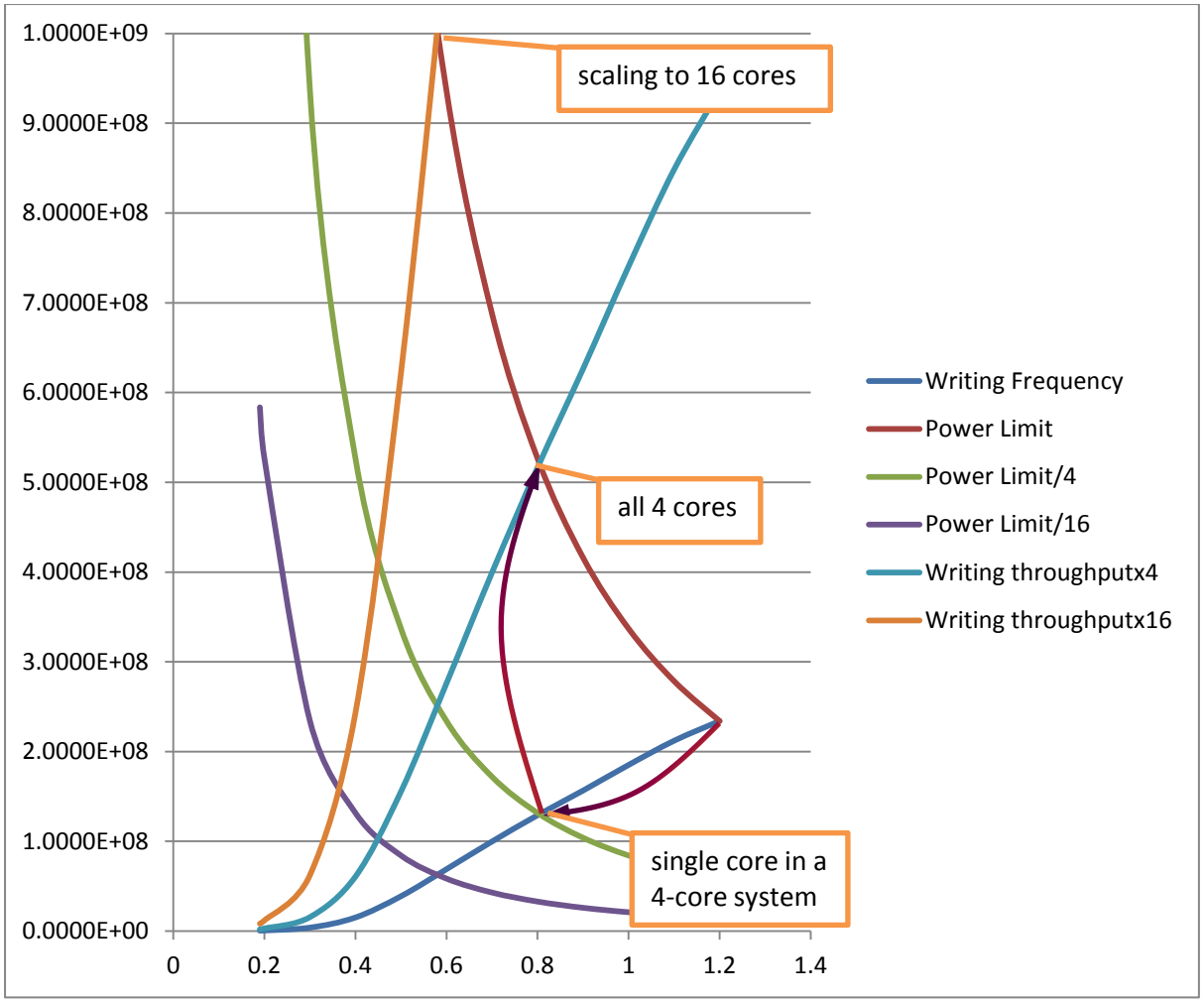


Figure 4 Scaling to 4 and 16 cores.

*In the trans- and sub-threshold regions*

With a Vdd from close to threshold to below threshold (in the above case below about 0.6V) the super-threshold power relation Equation (3) no longer describes the total power as leakage power starts to become an equally important or more important factor compared with dynamic switching power. It does not mean that the method of analysing how systems would scale under the same amount of power could no longer be used.

Instead of drawing constant power curves from equations (3) and (5) or dealing with the much more complex power equations taking leakage power into account, observed power from experiments can be used to estimate the points along the frequency curves pertaining to any known power budget. Here is another study based on experimental data on an ARM M0 core implementation (from Jatin Mistry, currently of ARM and formerly of Southampton University).

To determine the shape of power limit boundary curves, we start by setting a power limit amount, $P_{lim}$, usually the power consumption when operating an unscaled system at nominal Vdd and the appropriate reliable frequency for this Vdd ($P_{max}$). Then for each point *i* where there is experimental power data, we calculate the maximum possible scaling factor $N_i$ for that Vdd based on

$$N_i = P_{lim}/P_i \qquad (6)$$

where $P_i$ is the experimental power observed at data point i. This, as (4) and (5), is based on the perfect scaling assumption.

Plotting

Throughput = $N_i$ x Throughput $_i$

gives the power limit curve for the particular $P_{lim}$.

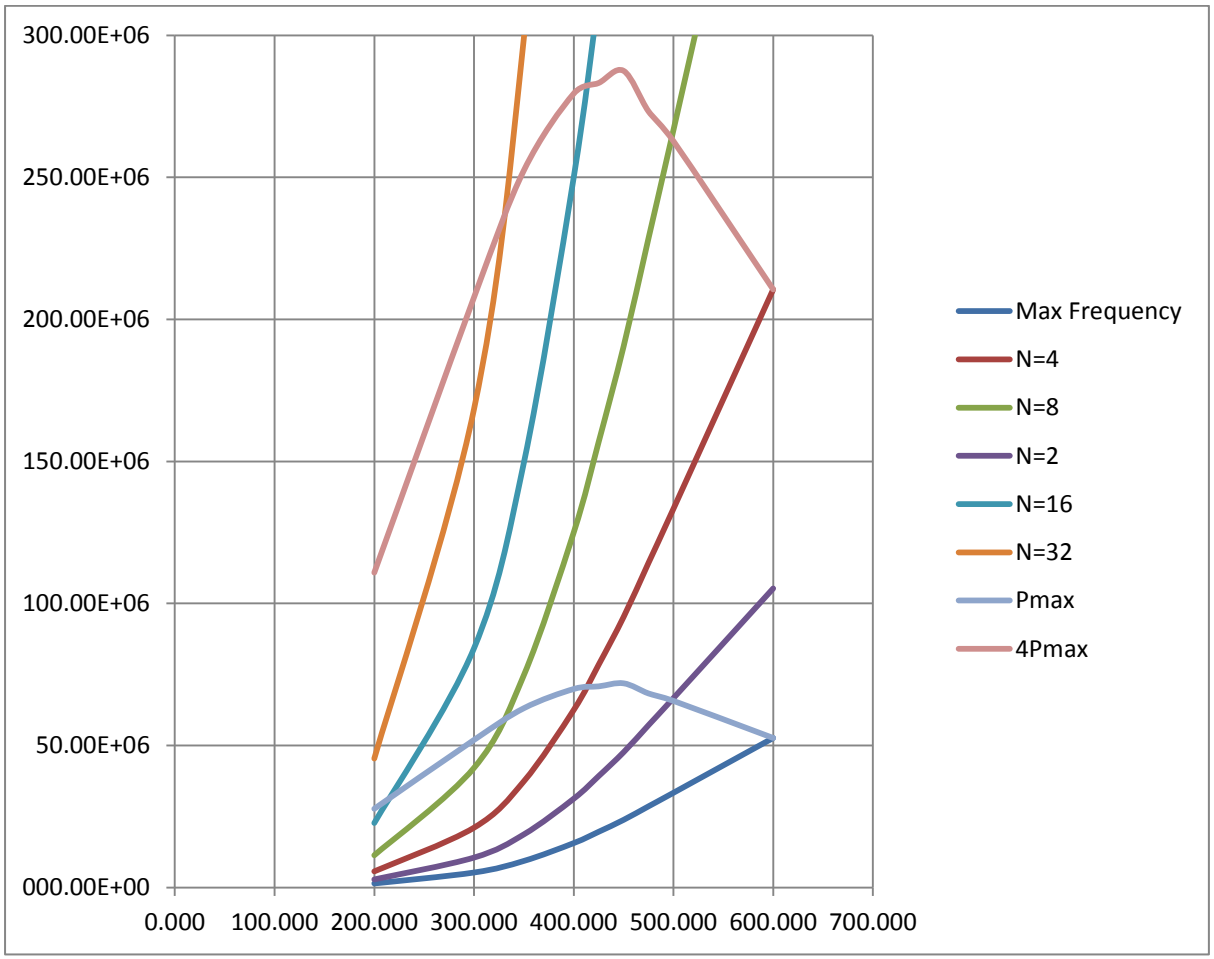


Figure 5 Scaling close to and in the subthreshold region.

As can be seen in Figure 5, the benefit of scaling is reduced in the subthreshold region. Scaling with a factor of 4 from 0.6V makes the system work at just above 0.4V with an overall computation throughput increase of roughly 1/3 only. Scaling any further actually reduces the max throughput.

When the available amount of power is increased (1 x $P_{max}$ and 4 x $P_{max}$ power boundaries included in Figure 5 for comparison), scaling further will provide more throughput benefits, even in the trans- and sub-threshold regions. This is because of the intersections between the power boundary and timing reliability curves happen at different parts of each curve. For instance the N=16 timing reliability curve intersects the 1 x power boundary deep in its exponential region but intersects the 4 x power boundary in its linear region. The linear region is where perfect scaling brings a lot of throughput improvement.

In terms of reliability and operable areas, ideal scaling will always improve the situation, assuming no low Vdd limit.

Figure 6 shows the scaling effects based on the SRAM data, with consideration for the subthreshold region, now with the total power from experimental data (and not calculated from (3) and (5)) including both dynamic and leakage power. As can be seen after about $N$ = 256, further scaling would not improve the system throughput of the system given the constant power limit reached at the max writing frequency and voltage of 1.2V without scaling, but the operable region continues to be expanded to the left, although the increase becomes smaller even without considering minVdd.

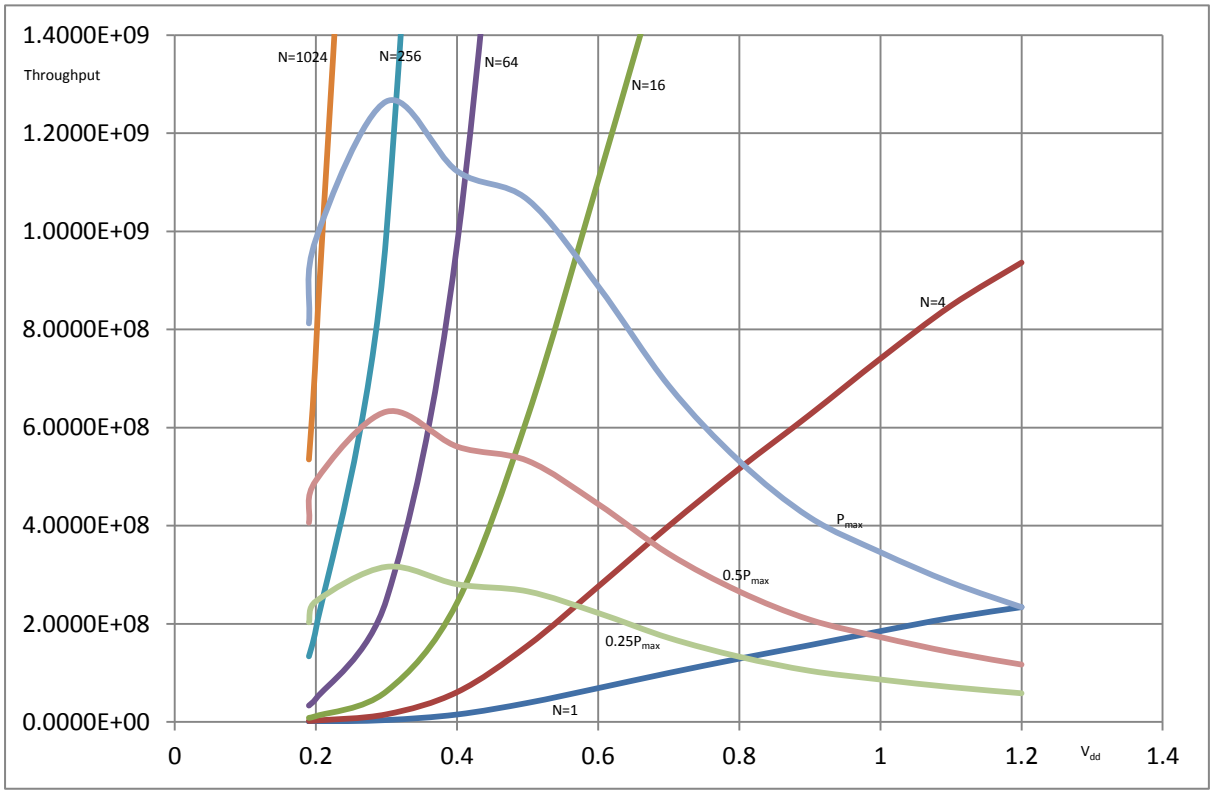


Figure 6 SRAM constant max power curve and scaling lines.

In Figure 6, power limit boundaries with smaller amounts of power than required for operating the system at nominal Vdd and maximum frequency are included. As can be seen higher power limits help provide better throughput benefits for scaling, mainly because they allow systems with higher scaling factors to operate in the linear range.

Scaling to more cores will open up more of the areas below the constant power curve towards the left, hence allow the system to work at lower Vdd at a higher computation throughput than before scaling at the same Vdd. This higher throughput may not be higher than before scaling at the same power. Hence it may seem power inefficient. However, the power supply may be more voltage-bound than power-bound (i.e. have a low maxVdd), in which case scaling could be advantageous even into the region where throughput no longer improves.

## Future work

Gross simplifications exist in this work so far, and the influence of PVT variability is not investigated. These are obvious future work issues. More experimental data should also be studied to make this modeling method more complete. Eventually it is envisaged that such performance-energy-reliability properties could be incorporated into general design- and run-time models.