Procurability Aware Design
Explanation of Model

Independent Variables and Objectives

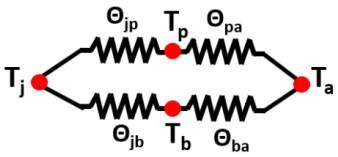
The independent variables of the model are the number of cores N_{core}, size of the L3 cache (in terms of the number of x MB slices N_{L3}), number of memory controllers N_{MC}, and number of IO pads N_{IO}. The core includes L1 and L2 caches as part of a monolithic structure. The term ***processor*** is used to refer to the entire system including cores, L3 cache, memory controllers, and IO. The term ***memory subsystem*** is used to refer to the memory hierarchy outside of the core, consisting of L3 and main memory.

The objectives of the model include power, area, performance, or any combination of the above.

The area of the processor A_{die} is computed as the dot product of the component area vector and component count vector. The power consumption of the processor P_{die} is computed as the dot product of the component power vector and component count vector.

Thermal Limitation

There are two paths where the thermal energy generated by the processor can be dissipated (i) through the package casing (ii) through the back of the PCB. The four parameters θ_{jc}, θ_{ca}, θ_{jb}, and θ_{ba} characterize the thermal resistance values from junction to casing, casing to air, junction to PCB, and PCB to air, respectively. The resistances are arranged in the following configuration (package means casing). The total effective thermal resistance from junction to air, θ_{ja}, is given by $\theta_{ja} = \frac{(\theta_{jc}+\theta_{ca})(\theta_{jb}+\theta_{ba})}{\theta_{jc}+\theta_{ca}+\theta_{jb}+\theta_{ba}}$. Given the maximum junction temperature, $T_{j,max}$, and the ambient temperature, $T_{ambient}$, which are parameters of the model, the maximum rate of the thermal dissipation of the processor, P_{max}, is given by $P_{max} = (T_{j,max} - T_{ambient})/\theta_{ja}$. The power consumption of the processor, P_{die}, must not exceed P_{max}.



Voltage and Power of the Core

All cores are assumed to operate at the same frequency with the same IPC (same microarchitecture). The voltage of the processor, V_{die}, depends linearly on the frequency of the cores as follows $V_{die} = \frac{f_{core}}{f_{core,nominal}} \cdot V_{die,nominal}$ where the nominal core frequency and nominal die voltage are parameters.

Given V_{die}, the power of each core, P_{core}, is computed by $P_{core} = C_{core} \cdot V_{die}^2 \cdot f_{core}$ where C_{core} is the capacitance of each core. For the same number of cores, a lower-powered design can be achieved by running the cores at a lower frequency. A reduction in f_{core} gives a cubic reduction in P_{core}.

Bumps and Wire

There is a lower bound on the area of the processor, A_{die}, since there must be enough space on the back of the die to fit bumps for power delivery, memory channels, and IO. The number of bumps for memory and IO are parameters. The number of bumps for power delivery, $N_{b,power}$, is computed by $N_{b,power} = \frac{P_{die}}{V_{die} \cdot I_{bump}} \cdot 2$ where I_{bump} is the maximum current rating of each bump. Each pair of bumps consist of VDD and GND pins, hence the factor of 2. Given bump pitch, W, as a parameter, the area taken up by all the bumps is $W^2 \cdot (N_{b,power} + N_{b,mc} + N_{b,io})$, which must be less than A_{die}.

Assuming a 3:2 aspect ratio for the die, the area of the processor, A_{die}, must also be large enough so that there is enough length on the perimeter of the die for wires. The model assumes wires are only connected to the two longer sides of the die. Given A_{die}, the combined lengths of the two longer sides, L, is given by $L = 6\sqrt{\frac{A_{die}}{6}}$. The maximum number of wires that can connect to the perimeter of the die is given by $N_{wire,max} = \frac{L \cdot N_{PCB}}{D}$ where N_{PCB} is the number of PCB layers and D is the link pitch. The model requires $N_{MC} \cdot N_{wire/MC} \leq N_{wire,max}$ (only wires for the memory controller are modeled).

Core Frequency and Area Scaling

The operating frequency of the core f_{core} is bounded by $f_{core,min}$ and $f_{core,max}$. While $f_{core,min}$ is a fixed parameter, $f_{core,max}$ is a variable in the model. The operating frequency of the core can be increased by increasing $f_{core,max}$. However, beyond a cutoff frequency f_c, the area of the core starts to increase with $f_{core,max}$. For $f_{core,max} > f_c$, an $x\%$ increase in $f_{core,max}$ results in $2x\%$ increase in area of each core, up to some absolute maximum frequency. This accounts for the microarchitecture changes required to achieve high clock speed.

Memory Model

The model makes the following assumptions about the memory subsystem (i) all accesses from the core go to L3 first, and only L3 misses are handled by the main memory (ii) the bandwidth between the core and L3 depends entirely on the size of L3 (iii) the bandwidth between L3 and main memory depends entirely on the number of memory controllers.

The hit rate of L3 depends on the relative sizes of the working set and L3. If the entire working set can fit within L3, the hit rate is equal to some nominal hit rate $r_{nominal}$ which is a parameter. If the working set is larger than L3, the hit rate is discounted as the following $\frac{N_{L3} \cdot Capacity_{L3}}{Capacity_{workset}} \cdot r_{nominal}$.

The effective bandwidth of the memory subsystem B_{sys} depends on whether accesses are bounded by L3 bandwidth or main memory bandwidth. The L3 and main memory bandwidths are computed as $N_{L3} * B_{L3}$ and $\frac{N_{MC} * B_{MC}}{L3\ miss\ rate}$, respectively, where B_{L3} and B_{MC} are parameters specifying the bandwidth of each slice of L3 and each memory controller. The denominator in the main memory bandwidth formula exists because only L3 misses are handled by the main memory. Then, B_{sys} is the minimum of the L3 bandwidth and main memory bandwidth.

Performance Model

Performance is described by the roofline model, which states that attainable performance is either compute-bound or memory-bound. The compute throughput π is computed as $\pi = CPI \cdot f_{core} \cdot N_{core}$.

The FLOPs rate that can be supported by a given B_{sys} is the product of B_{sys} and arithmetic intensity I_{sys}. Since B_{sys} is the bandwidth of the memory subsystem consisting of L3 and main memory, I_{sys} must be the arithmetic intensity seen by L3 and main memory. I_{sys} is computed as $I_{sys} = \frac{Capacity_{L1}+Capacity_{L2}}{Capacity_{workset}} \cdot I_{app}$ where I_{app} is the arithmetic intensity inherent to a given application. If the size of the working set is small compared to the size of L1 and L2, which are assumed to be exclusive, then requests from the core will rarely need to go outside of L2. Hence, the effective arithmetic intensity as seen by the memory subsystem will be high.

Based on the roofline model, the performance is given by $min(\pi,\ I_{sys} \cdot B_{sys})$.