

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

Appendix 316-02

The following list illustrates certain grounds for invalidity:

- U.S. Patent Application Publication No. 2010/0115304 to Finkelstein et al. (“*Finkelstein*”), which was filed on October 31, 2008, and published on May 6, 2010, qualifies as prior art to the '316 patent under at least pre-AIA §§ 102(a), (b), and (e).
- U.S. Patent Application Publication No. 2006/0288241 to Felter et al. (“*Felter*”), which was filed on June 16, 2005, and published on December 21, 2006, qualifies as prior art to the '316 patent under at least pre-AIA §§ 102(a), (b), and (e).
- U.S. Patent Application Publication No. 2012/0054511 to Brinks et al. (“*Brinks*”), which was filed on August 31, 2010, and published on March 1, 2012, qualifies as prior art to the '316 patent under at least pre-AIA § 102(e).
- U.S. Patent Application Publication No. 2009/0089602 to Bose et al. (“*Bose-602*”), which was filed on September 27, 2007, and published on April 2, 2009, qualifies as prior art to the '316 patent under at least pre-AIA §§ 102(a), (b), and (e).

As illustrated below, to the extent that *Finkelstein* is found not to anticipate, expressly or inherently, one or more of the asserted claims of the '316 Patent, each of claims 8-12 is invalid as obvious in view of *Finkelstein* alone or in combination with other prior art references, including, but not limited to, the prior art discussed below, the prior art identified in Defendant's Invalidity Contentions, and the prior art described in additional claim charts attached as Appendices 316-01a through 316-01b and 316-03 through 316-06. Exemplary invalidity grounds based on the *Finkelstein* reference include the following, although these examples are by no means limiting:

- Anticipation and obviousness based on *Finkelstein* (claims 8, 9, and 11);
- Obviousness based on *Finkelstein* in combination with *Felter* (claims 8-12);
- Obviousness based on *Finkelstein* in combination with *Felter* and *Brinks* (claim 10);
- Obviousness based on *Finkelstein* in combination with *Bose-602* (Claim 12); and
- Obviousness based on *Finkelstein* in combination with *Felter* and *Bose-602* (claim 12).

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

The citations included in the table below are exemplary only and are not intended to be limiting or exhaustive. Defendant reserves the right to cite additional disclosures and evidence in support of their positions as this case develops further. Additionally, citation to a particular figure in a reference should be understood to encompass the caption and description of the figure and any text relating to or discussing the figure. Similarly, citation to any text referring to a figure should be understood to include the figure as well. Additionally, Defendant may be basing the arguments below on one or more claim construction positions that Daedalus has not expressly disclosed but seems to apply in its Patent Local Rule 3-1 disclosures or Defendant's present understanding of Daedalus's view and application of claim scope. By applying the prior art in this way, Defendant does not concede its agreement with Daedalus's positions, nor do these Invalidity Contentions constitute any admission regarding claim scope. Nor is Defendant asserting any claim construction positions through these charts. Moreover, claim preambles generally are not limiting, and nothing in these contentions should be deemed an admission that the preamble in the asserted claims is limiting. As this case further develops, Defendant may amend or supplement these exemplary disclosures and any related contentions as appropriate, and in a manner consistent with all applicable rules governing this case. In particular, Defendant reserves the right to rely on (1) uncited portions of the identified prior art; (2) other prior art not identified herein; (3) references that show the state of the art (irrespective of whether such references themselves qualify as prior art to the asserted patents); (4) factual testimony from the inventors or authors of the prior art references; and/or (5) expert testimony, to provide context to or aid in understanding the prior art and the state of the art at the time of the alleged inventions. Defendant also reserves the right to amend or supplement these exemplary disclosures based on changes in Daedalus's positions as the case progresses, including changes to Daedalus's infringement theories, claim interpretations, or apparent claim constructions.

Descriptions of Exemplary Obviousness Combinations Based on *Finkelstein*

Finkelstein and each of the additional references cited below are directed to device architectures with similar structures and functions. They are analogous art because they come from the same field of endeavor as the '316 patent and are reasonably pertinent to the problems the '316 patent addresses. Because they are all directed to the same types of device and have similar structures and functions, a person of ordinary skill in the art ("POSITA") would have appreciated that *Finkelstein* and the additional references cited below provide disclosures that relate to and complement one another. Moreover, a POSITA would have found it obvious to combine their disclosures in the ways discussed below.

Exemplary Reasons for Modifying *Finkelstein*'s Power Allocation Equation to Account for a Minimum Power Allocation

Finkelstein teaches allocating power among power planes according to the equation: $E^{k,i} = W_{ki}E^k$, where E^k is the power budget, $E^{k,i}$ is the amount of power allocated to power plane i , and W_{ki} is a vector of length m "such that entry i corresponds to the portion of the budget

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

that goes to power plane i .” *Finkelstein* ¶ [0034]. A POSITA would have found it obvious to modify *Finkelstein*’s power allocation equation to account for a minimum power allocation, as taught, for example, by *Felter*.

Felter teaches a method for allocating portions of a power budget to components in a system by “allocating each component its corresponding standby power and a share of the system power available for dynamic powering.” *Felter* at Abstract. According to *Felter*, the “disclosed power management method is . . . applicable to single-chip systems”—like *Finkelstein*’s processor 102-1—“in which multiple types of components (e.g., processor cores, cache memories, and the like) are integrated within a single piece of silicon.” *Felter* ¶ [0028].

Felter teaches a method of power management by which a control logic determines a “standby” (i.e., minimum) power for each component. See *Felter* ¶ [0058] (“The set of values determined in block 504 includes the standby values (minimum power values) of each system component. . . . [E]ach component always draws its standby or minimum power.”), Fig. 5.

After determining these minimum powers, *Felter* teaches that the control logic allocates power to each power domain N (P_{ALLOC_N}) as the sum of its minimum power P_{MIN_N} and a percentage of the available power headroom (P_{AVAIL}). See *Felter* ¶ [0061] (teaching that “ $P_{ALLOC_N} = P_{MIN_N} + P_{AVAIL} * (P_{ACT_N} / P_{ACT_{TOT}})$ ”).

By ensuring each active component is allocated a standby power, *Felter*’s allocation guarantees that “each component receives a minimum level of power,” *id.* ¶ [0061], which can be set such that “there is always enough power for each component to obtain some minimal work from it,” *id.* ¶ [0065]. By first ensuring each active component receives a minimum level of power and then allocating the rest of the power to the component that can best use it, *Felter* ensures “stable operation of the system within significantly lower power budgets and consequently smaller cooling resources while reducing the occasional negative impact of the reduced power budgets on system performance.” *Id.* ¶ [0009]. Thus, a POSITA would have been motivated to guarantee each active domain receives its minimum level of power to obtain the benefits *Felter* teaches.

Finkelstein teaches that an energy range [E_{low}^i , E_{high}^i] may be mapped onto a discrete power range [P_n^i , P_o^i]. See *Finkelstein* ¶ [0032]. Although *Finkelstein* suggests that budget values below E_{low}^i may be allocated, a POSITA would have applied the teachings of *Felter* to allocate a minimum power value instead. See *Felter* ¶ [0058], ¶ [0065]. According to *Finkelstein*, at least E_{low}^i — the power allocation corresponding to the maximal efficiency state (i.e., minimum power state) P_n^i —would be allocated. See *Finkelstein* ¶ [0032] (“Budget values below E_{low}^i are mapped into P_n^i ”); Advanced Configuration and Power Interface (ACPI) Specification, Rev. 4.0, at 41 (June 16, 2009) (explaining P_n represents the state in which “the performance capability of a device or processor is at its minimum level and consumes minimal power while remaining in an active state”).

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

A POSITA would have found it obvious to apply *Felter*'s teachings to modify the power allocation calculation *Finkelstein* describes (i.e., $E^{k,i} = W_{ki}E^k$) to account for a minimum power allocation (E_{low}^i) to the i th power domain. See *Felter* ¶¶ [0058]-[0061]. More specifically, a POSITA would have found it obvious to design power manager 140 to allocate power $E^{k,i}$ to the i th power domain using the following, slightly modified expression taught by *Felter*: $E^{k,i} = W_{k,i}(E^k - \sum E_{low}^j) + E_{low}^i$, where $W_{k,i}$ represents the user's or user application's preferences for how to allocate the dynamic power headroom—i.e., the power that exceeds the minimum requirements of the various power domains ($E^k - \sum E_{low}^j$). See *Finkelstein* ¶ [0034] (“For constraint k, user (or application defined) preferences that describe how budget E^k is distributed among power planes may be determined. For example, such information may be provided as an input in the form of vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power plane i.”).

This is the same allocation *Felter* teaches using an equivalent expression: “ $PALLOC_N = P_{MIN_N} + P_{AVAIL} * (PACT_N / PACT_{TOT})$.” *Felter* ¶ [0061]. A POSITA would have understood $P_{AVAIL} = (P_{BUDGET} - \sum P_{MIN_j})$. See *id.* ¶ [0058] (“Using the standby values [i.e., minimum power values], an available power figure (P_{AVAIL}) is determined by subtracting from the maximum power deliverable by the system [P_{BUDGET}] the sum of all the standby values.”). Since the variable syntax is arbitrary, the index N may be changed to i; the power allocation ($PALLOC_N$) may be rewritten as $E^{k,i}$; the power budget P_{BUDGET} may be rewritten as E^k ; the portion $PACT_N / PACT_{TOT}$ of P_{AVAIL} allocated to the domain may be rewritten as $W_{k,i}$; and the minimum/standby powers P_{MIN_j} may be rewritten as E_{low}^j .

Putting this together, a POSITA would have understood the expression “ $PALLOC_N = P_{MIN_N} + P_{AVAIL} * (PACT_N / PACT_{TOT})$,” *id.* ¶ [0061], may be rewritten as $PALLOC_i = (PACT_i / PACT_{TOT}) * (P_{BUDGET} - \sum P_{MIN_j}) + P_{MIN_i}$, which is mathematically equivalent to the expression above: $E^{k,i} = W_{k,i}(E^k - \sum E_{low}^j) + E_{low}^i$.

A person of ordinary skill in the art would have considered modifying *Finkelstein*'s equation to account for a minimum power allocation to be an example of (1) combining prior art elements (e.g., combining *Felter*'s technique for ensuring each power domain always receives a minimum power allocation with a processor with *Finkelstein*'s architecture that allocates power based on vectors using user or application defined preferences as inputs) according to known methods (e.g., simply accounting for such a value in *Finkelstein*'s equation (2)) to yield predictable results (e.g., a processor like *Finkelstein*'s but that ensures each power domain receives a minimum amount of power, thereby ensuring stable system operation and that each power domain functions at least at a minimal level); (2) use of a known technique (e.g., *Felter*'s technique for ensuring each power domain always receives a minimum power allocation) to improve similar devices (e.g., a processor like *Finkelstein*'s that allocates power based on user- or application-defined vectors) in the same way (e.g., ensuring each power domain always receives a minimum power allocation); (3) applying a known technique (e.g., *Felter*'s technique for ensuring each power domain always receives a minimum power allocation) to a known device (e.g., a processor like *Finkelstein*'s that allocates power based on user- or application-defined vectors) ready for improvement to yield predictable results (e.g., a processor like *Finkelstein*'s but that ensures each power domain receives a minimum amount of power, thereby ensuring stable system operation

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

and that each power domain functions at least at a minimal level); and (4) a teaching, suggestion, or motivation in the prior art (e.g., the exemplary teachings in *Felter* that allocated power may include a minimum power allocation) that would have led a person of ordinary skill in the art to modify *Finkelstein* according to, or to combine *Finkelstein* with, the teachings of *Felter*, for example, to arrive at the claimed invention. *See, e.g., KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 416-21 (2007).

Exemplary Reasons for Obtaining a Minimum Reservation Value from a Configuration Register Written by User-Level Software in *Finkelstein*'s Power Allocation Method

A POSITA would have found it obvious to control the minimum reservation values (i.e., E_{low}^{CPU} and E_{low}^{GPU}) with user-level software. *Finkelstein* teaches, for example, that the inputs to the sharing policy vectors $W_{k,i}$ are “user (or application defined) preferences.” *Finkelstein* ¶ [0034]. From this, a POSITA would have found it obvious to control the minimum reservation values (i.e., E_{low}^{CPU} and E_{low}^{GPU}) with user-level software as well, simply using another user-defined vector that contains the E_{low}^i values. *See id.* ¶ [0032]; *infra* [2]. A POSITA would have found it beneficial to do so to provide the user with additional control over the user experience by balancing the tradeoff between performance and power consumption.¹

A POSITA would have known a common way to control power management features is to obtain them from a configuration register.² Because *Finkelstein* teaches that the policy sharing values are controllable by user-level software, a POSITA would have found it obvious to obtain the minimum reservation values from a configuration register written by user-level software, as taught, for example, by *Brinks*.

¹ *See, e.g., Brinks* ¶ [0051] (“The policy defines the tradeoffs between user experience and power consumption. Accordingly, the . . . user may determine a set of policies that appropriately balance these tradeoffs.”); *de Cesare* ¶ [0009] (“[T]he configureability [sic] of the performance states may permit more fine-grained control of the performance level in the system”); *Gwennap* at 6 (“If the GPU is not in use or is operating at a low speed, the power manager can assign some of its thermal budget to the CPUs, allowing them to run at a higher clock speed. Conversely, on a graphics-intensive application, the power manager can overclock the GPU while slowing down the CPUs.”).

² *See, e.g., Brinks* ¶ [0049] (“[P]ower management policies may be stored in a table, hierarchical table, register, matrix or some combination of the three”); *de Cesare* ¶ [0046] (“Whether or not performance domains transition may be hardware controlled by the PMU 28, or may be software controlled using the valid indications in the performance configuration registers as described

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

A person of ordinary skill in the art would have considered obtaining the minimum reservation value for the second domain from a configuration register written by user-level software to be an example of (1) combining prior art elements (e.g., the power allocation method of *Finkelstein*, and storing power control variables in a configuration register written by user-level software, as taught by, for example, *Brinks*) according to known methods (e.g., using software to write power allocation variables to corresponding configuration registers for use in power allocation determinations, as taught by, for example, *Brinks*) to yield predictable results (e.g., more fine-grained control of power allocation in a multi-domain SoC, allowing for a more satisfying user experience based on user preferences); (2) use of known technique (e.g., *Brinks*'s technique of using user-level software and pre-defined models to determine power allocation variables in a multi-domain SoC) to improve similar devices (e.g., a device like *Finkelstein*'s, which allocates power to components based on user preferences) in the same way (e.g., by allowing user-level software to help determine power control variables in a multi-domain SoC); (3) applying a known technique (e.g., *Brinks*'s technique of using user-level software and pre-defined models to determine power allocation variables in a multi-domain SoC) to a known device ready for improvement (e.g., a multi-domain SoC like *Finkelstein*'s, which allocates power to components based on user preferences) to yield predictable results (e.g., more fine-grained control of power allocation in a multi-domain SoC, allowing for a more satisfying user experience based on user preferences); and (4) a teaching, suggestion, or motivation in the prior art (e.g., *Brinks*'s teaching that providing both user-level software control and pre-defined hardware control over power policy variables accounts for actual user feedback, improves user experience, and accounts for tradeoffs between performance and power conservation) that would have led a person of ordinary skill in the art to modify *Finkelstein* according to, or to combine *Finkelstein* with, the teachings of *Brinks*, for example, to arrive at the claimed invention. *See, e.g., KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 416-21 (2007).

previously.”). A POSITA would have also understood that such configuration registers may be written by user-level software. *See, e.g., Brinks* ¶ [0050] (“[T]he table, register, or matrix may be programmable, so that its behavior can be changed at run-time to support different chip operating modes.”); *de Cesare* ¶ [0041] (“The selection may be made by writing the pointer in each register corresponding to the performance domain in the registers 30B.”).

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

Exemplary Reasons for Determining Sharing Policy Values Based on Component Active Power Requirements

Under *Finkelstein*, the sharing policy value of the i th domain is $W_{k,i}$. See *Finkelstein* ¶ [0034]. A POSITA would have found it obvious to determine the sharing policy value based on expected activity as taught, for example, by *Felter*.

Under the *Finkelstein-Felter* combination power is allocated according to *Finkelstein*'s modified power allocation equation: $E^{k,i} = W_{ki}(E^k - \sum E_{low}^j) + E_{low}^i$. See *supra* [8c]. A POSITA would have recognized that *Finkelstein*'s modified power allocation parallels *Felter*'s power allocation equation: $PALLOC_N = P_{MIN_N} + P_{AVAIL}(PACT_N/PACT_{TOT})$. See *Felter* ¶ [0061], Fig. 5. For example, the minimum reservation value of the *Finkelstein-Felter* combination (e.g., E_{low}^i) is comparable to *Felter*'s P_{MIN_N} (e.g., the minimum standby power). See *Felter* ¶ [0058]. *Felter* also explains that P_{AVAIL} represents the amount of power that is available for dynamic allocation and is calculated as: $P_{AVAIL} = P_{BUDGET} - \text{SUM}(P_{MIN_N})$. *Id.* This is the same calculation as is performed in *Finkelstein*'s modified power allocation equation (e.g., $E^k - \sum E_{low}^j$). Accordingly, a POSITA would have understood that *Felter*'s $(PACT_N/PACT_{TOT})$ term would be comparable to *Finkelstein*'s W_{ki} term, as both would represent the portion of the available power allocated to a particular domain. See *Finkelstein* ¶ [0034] (“[S]uch information may be provided as an input in the form of vectors W_k of length m , such that entry i corresponds to the portion of the budget that goes to power plane i .”); *Felter* ¶ [0061] (“The portion of P_{AVAIL} allocated to a system is determined by the ratio of the components active power requirements to the total active power requirements of the system.”).

Felter explains that $PACT_N$ represents the amount of active power expected to be consumed by domain N in the upcoming interval. See *Felter* ¶ [0059]. $PACT_N$ may be calculated “as the product of the activity (UTIL) predicted for component N ($UTIL_N$) and a constant of proportionality for component N (C_N).” *Id.* By allocating available power according to the active power requirements, *Felter* explains that scarce system power may be allocated “equitably so that the system performance of each component is impacted proportionately.” *Felter* ¶ [0061]. Thus, if the power budget is not sufficient to allocate enough power to each component for it to operate in an unconstrained state, allocating power according to the active power requirements of each component “has the effect of distributing performance throttling among multiple components of the system so that the overall performance penalty is reduced.” *Felter* ¶ [0063]. Accordingly, a POSITA would have found it obvious to calculate *Finkelstein*'s W_k vectors according to the component active power requirements to obtain the benefits taught by *Felter* (e.g., reducing an overall performance penalty when there is insufficient power for each component to operate in an unconstrained manner). *Id.*

A person of ordinary skill in the art would have considered calculating *Finkelstein*'s W_k vectors according to the component active power requirements as taught by *Felter* to be an example of (1) combining prior art elements (e.g., combining *Finkelstein*'s method of allocating power according to W_k vectors with *Felter*'s method for allocating power to components according to their active power

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

requirements) according to known methods (e.g., calculating *Finkelstein*'s W_k vectors according to the component active power requirements, as taught by, for example, *Felter*) to yield predictable results (e.g., a reduced overall performance penalty when there is insufficient power for each component to operate in an unconstrained manner); (2) a simple substitution of one known element (e.g., *Finkelstein*'s W_k vectors, determined based on user preference information) for another (e.g., *Felter*'s method for allocating power to components according to their active power requirements) to obtain predictable results (e.g., a reduced overall performance penalty when there is insufficient power for each component to operate in an unconstrained manner); (3) use of a known technique (e.g., *Felter*'s method for allocating power to components according to their active power requirements) to improve similar devices (e.g., *Finkelstein*'s SoC, which allocates power based on $W_{k,i}$) in the same way (e.g., by reducing the overall performance penalty when there is insufficient power for each component to operate in an unconstrained manner); (4) applying a known technique (e.g., *Felter*'s method for allocating power to components according to their active power requirements) to a known device ready for improvement (e.g., *Finkelstein*'s SoC, which allocates power based on $W_{k,i}$) to yield predictable results (e.g., a reduced overall performance penalty when there is insufficient power for each component to operate in an unconstrained manner); and (5) a teaching, suggestion, or motivation in the prior art (e.g., *Felter*'s teaching that allocating power according to active power requirements may reduce the overall performance penalty when there is insufficient power for each component to operate in an unconstrained manner) that would have led a person of ordinary skill in the art to modify *Finkelstein* according to, or to combine *Finkelstein* with, the teachings of *Felter*, for example, to arrive at the claimed invention. See, e.g., *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 416-21 (2007).

Exemplary Reasons for Incrementing the Sharing Policy Value when a Request for a Higher Frequency is not Granted

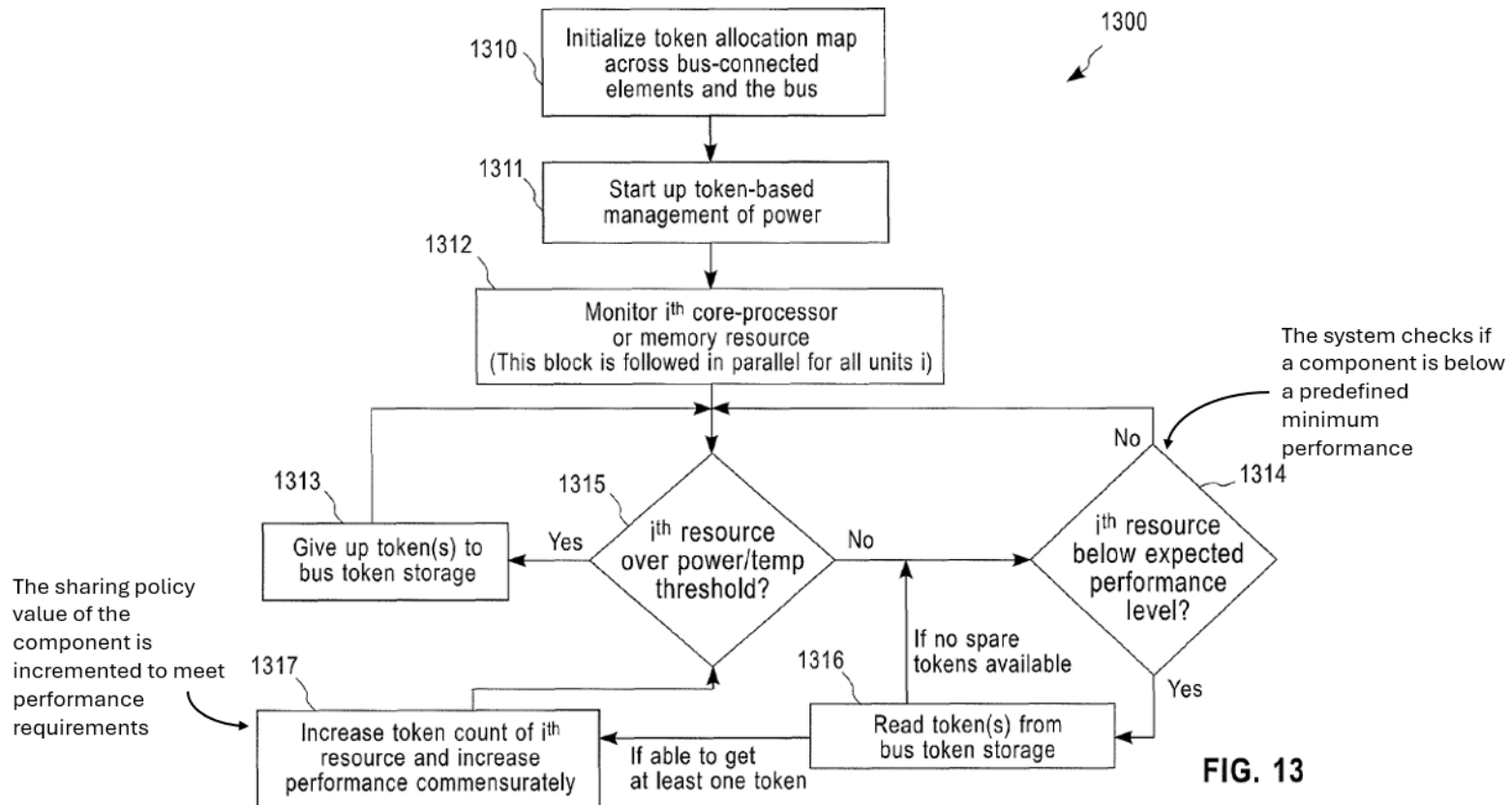
Under *Finkelstein*, the sharing policy value of the i th domain is $W_{k,i}$. See *Finkelstein* ¶ [0034]. A POSITA would have found it obvious to increment the sharing policy value when a request for a higher frequency is not granted as taught, for example, by *Bose-602*.

Bose-602, for example, teaches allocating a power budget to components by “initializing a token allocation map across the connected components, wherein each component is assigned a power budget as determined by a number of allocated tokens in the token allocation map.” *Bose-602* ¶ [0013]. *Bose-602*'s method monitors processor cores, assigns more tokens to the cores that can make use of more power, and then increases performance based on the number of tokens. See *id.* at Fig. 13.

According to *Bose-602*'s technique, the system checks whether a component is operating “below a predefined minimum performance.” *Id.* ¶ [0086]; see also *id.* at Fig. 13 (block 1314). If that component is operating below the predefined performance, “extra tokens are used to boost up performance in the targeted unit . . . by increasing its voltage and frequency.” *Id.* ¶ [0086]; see also *id.* at Fig. 13 (block 1317). Once extra tokens have been assigned and performance has been increased, *Bose-602*'s method again checks whether the component is operating within the power and temperature limits and, if so, also whether it is operating below a predefined performance.

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

See id. ¶ [0086], Fig. 13 (blocks 1314 and 1315). If the component is still operating below the predefined performance, more tokens are allocated to it. Accordingly, *Bose-602* teaches incrementing the number of tokens allocated to the component to meet a performance requirement (which would include a request for a higher power and, thus, a higher frequency).



Like *Finkelstein*'s sharing policy values ($W_{k,i}$), the number of *Bose-602*'s tokens determines the portion of the power budget allocated to a component. A POSITA would have found it obvious to borrow the way in which *Bose-602* uses tokens and apply it to *Finkelstein*'s $W_{k,i}$ values (e.g., by defining the vector components $W_{k,i}$ based on the number of tokens).

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

A POSITA would have been motivated to do so to ensure the domains necessary to process a given workload during a runtime interval can meet the relevant performance requirements. A POSITA would have understood that if a request for a higher frequency is not met it would mean that the domain is operating below a desired performance level. *See, e.g., Nussbaum* ¶ [0037] (“[T]he power of a particular computational unit (voltage×current) is based on the frequency of the clock signal, the supply voltage, and the amount of activity in the computational unit.”), *Bose-602* ¶ [0086] (“If it is assessed to be below a predefined performance, block 1316 is used to try and read additional tokens from the centralized bus token storage; such extra tokens are used to boost up performance in the targeted unit in block 1317 (e.g., by increasing its voltage and frequency, if allowed by the system architecture).”). A POSITA would have been motivated to increase the sharing policy value of a domain to allocate additional power to it if the domain is not meeting a frequency request. A POSITA would have applied the teachings of *Bose-602* to increment the sharing policy value of the domain so that its frequency may be increased.

A person of ordinary skill in the art would have considered incrementing *Finkelstein*'s sharing policy values ($W_{k,i}$) when a request for a higher frequency was not granted to be an example of (1) combining prior art elements (e.g., *Finkelstein*'s multi-core processor, which allocates power based on $W_{k,i}$, with *Bose-602*'s teaching to increment the allocated power allocated when performance requirements are not met) according to known methods (e.g., checking whether each component meets performance requirements and, if not, incrementing the power allocated to the associated domain(s) to yield predictable results (e.g., multi-core processor that satisfies the given workload demands over a given processing interval); (2) use of known technique (e.g., *Bose-602*'s teaching to increment the power allocated to a domain if a performance requirement is not met) to improve similar devices (e.g., *Finkelstein*'s multi-core processor, which allocates power based on $W_{k,i}$) in the same way (e.g., incrementing $W_{k,i}$ when higher performance is required to meet a frequency request by one of the components, such that the request may be satisfied); (3) applying a known technique (e.g., *Bose-602*'s teaching to increment the power allocated to a domain if a performance requirement is not met) to a known device (e.g., *Finkelstein*'s multi-core processor, which allocates power based on $W_{k,i}$) ready for improvement to yield predictable results (e.g., a multi-core processor that satisfies the given workload demands over a given processing interval); and (4) a teaching, suggestion, or motivation in the prior art (e.g., *Bose-602*'s teaching to increment the power allocated to a domain if a performance requirement is not met) that would have led a person of ordinary skill in the art to modify *Finkelstein* according to, or to combine *Finkelstein* with, the teachings of *Bose-602*, for example, to arrive at the claimed invention. *See, e.g., KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 416-21 (2007).

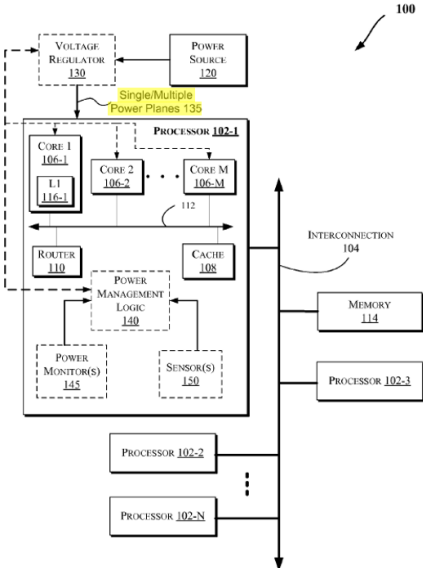
Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
Claim 8	
<p>[8pre] A method comprising:</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests a “method.”</p> <p>For example, <i>Finkelstein</i> teaches “[m]ethods and apparatus relating to power management for multiple processor cores.” <i>Finkelstein</i> at Abstract.</p> <p><i>Felter</i> describes a complementary method for managing power in a system with multiple components. See <i>Felter</i> at Abstract, ¶ [0010].</p> <p>Exemplary teachings from the prior art include the following:</p> <p><i>Finkelstein</i> discloses the following at Abstract:</p> <p style="padding-left: 40px;">Methods and apparatus relating to power management for multiple processor cores are described. In one embodiment, one or more techniques may be utilized locally (e.g., on a per core basis) to manage power consumption in a processor. In another embodiment, power may be distributed among different power planes of a processor based on energy-based considerations. Other embodiments are also disclosed and claimed.</p> <p>See also <i>Finkelstein</i> at ¶¶ [0022], [0025]-[0038], [0040]-[0041], Figs. 3, 5.</p> <p><i>Felter</i> discloses the following at Abstract:</p> <p style="padding-left: 40px;">A method for managing power in a data processing system having multiple components includes determining a power budget for the system. Activity levels during a forthcoming time interval are then predicted for each of the components. Using the predicted activity levels, the power budget is allocated among the system components. An activity limit is then established for each component based on its corresponding portion of the power budget. The activity of a component is then monitored and, if the</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>component's activity exceeds the component's corresponding activity limit, constrained. Determining the predicted level of activity may include determining a predicted number of instructions dispatched by a processor component or a predicted number of memory requests serviced for a system memory component. Allocating the power budget includes allocating each component its corresponding standby power and a share of the system power available for dynamic powering based on the expected levels of activity.</p> <p><i>Felter</i> discloses the following at ¶ [0010]:</p> <p>The objective identified above is achieved with a method for managing power in a data processing system having multiple components as disclosed herein. The method includes determining a power budget for the system. Activity levels during a forthcoming time interval are then predicted for each of the components. Using the predicted activity levels, the power budget is allocated among the system components. An activity limit is then established for each component based on its corresponding portion of the power budget. The activity of a component is then monitored and, if the component's activity attempts to exceed the component's corresponding activity limit, it is constrained from doing so. Determining the predicted level of activity may include determining a predicted number of instructions dispatched by a processor component or a predicted number of memory requests serviced for a system memory component. Allocating the power budget includes allocating each component its corresponding standby power and a share of the system power available for dynamic powering based on the expected levels of activity. Monitoring an activity limit may include monitoring a processor performance monitor configured to count the number of instructions dispatched during a timing interval. In this embodiment, constraining the activity of the processor may include pipeline throttling in which the processor is prevented from dispatching additional instructions until the current timing interval expires.</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations
	<p>Alternatively, constraining processor activity may be achieved by reducing the voltage and frequency applied to the processor.</p> <p><i>See also Felter</i> at Title, ¶¶ [0014], [0016], [0019], [0028], [0033], [0057]-[0061], Figs. 3-5.</p>
<p>[8a] determining, in a power controller of a multi-domain processor, a power budget for the multi-domain processor for a current time interval, the multi-domain processor including at least a first domain and a second domain;</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “determining, in a power controller of a multi-domain processor, a power budget for the multi-domain processor for a current time interval, the multi-domain processor including at least a first domain and a second domain.”</p> <p><i>Finkelstein</i> teaches a processor with multiple power planes 135:</p>  <p align="center"><i>FIG. 1</i></p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations
	<p><i>Finkelstein</i> teaches that “voltage regulator(s) 130 may be coupled to a single power plane 135 (e.g., supplying power to all the cores 106) or to multiple power planes 135 (e.g., where each power plane may supply power to a different core or group of cores).” <i>Finkelstein</i> ¶ [0014]. A POSITA would have understood that a power plane is synonymous with a power domain. <i>See, e.g.</i>, '316 pat. at 4:3-4 (explaining domains may “also be referred to as ‘planes’”).</p> <p><i>Finkelstein</i> also teaches that each of the power planes operates at its own voltage and frequency. For example, <i>Finkelstein</i> teaches that logic 140 may “throttle or modify an operational characteristic of one or more of the cores 106 (such as an operating voltage and/or an operating frequency of a processor core 106).” <i>Finkelstein</i> ¶ [0024]. Because the voltage and frequency of one or more processor cores 106 may be controlled independently of others, a POSITA would have understood those processor cores 106 operate at a voltage and frequency controlled independently of other power planes. <i>See also, e.g., Nussbaum</i> ¶ [0033] (“A performance state is characterized by a unique pair of core voltage and frequency values.”), Tbl. 3. Therefore, core 106-1 would operate on a first domain and core 106-2 would operate on a second domain.</p> <p>Further, <i>Finkelstein</i> describes a power management logic 140 that may “control supply of power to components of the processor 102 (e.g., cores 106).” <i>Finkelstein</i> ¶ [0016]; <i>see also id.</i> at Fig. 1. According to <i>Finkelstein</i>, “the operations discussed herein, e.g., with reference to FIGS. 1-7, may be implemented as hardware (e.g., logic circuitry),” thereby indicating power management logic 140 may be implemented as a “power controller,” as claimed. <i>Finkelstein</i> ¶ [0053].</p>

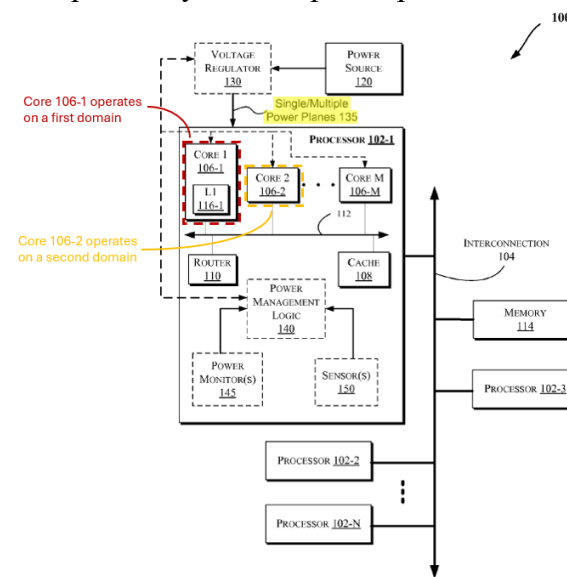


FIG. 1

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations
	<div data-bbox="945 276 1543 885" data-label="Diagram"> </div> <p data-bbox="1081 889 1171 922">FIG. 1</p> <p data-bbox="598 966 1900 1323"><i>Finkelstein</i> further teaches determining a power budget for a current time interval. According to the '316 patent, “E_n is the power budget for the current time interval,” which can be calculated as “$E_n = E_{n-1} * \alpha + (1 - \alpha) * (\text{Power_Limit} * \Delta T - \text{Energy})$.” ’316 pat. at claim 7; <i>see also id</i> at 3:16-62, 4:38-51. Although this is technically an “energy budget” rather than a power budget and “can be measured in Joules,” <i>id.</i> at 3:28-29, the ’316 patent nevertheless also refers to it as a “power budget,” <i>id.</i> at claim 7. <i>Finkelstein</i> teaches calculating a similar “power” budget, which <i>Finkelstein</i> refers to as an “energy budget,” using a nearly identical equation: “$E_{n+1} = \alpha E_n + (\text{TDP}_n - P_n) \Delta t_n$.” <i>Finkelstein</i> ¶¶ [0028]-[0029]. A POSITA would have understood the difference between energy and power in this context is simply the runtime Δt_n. <i>See, e.g.,</i> S. Kaxiras & M. Martonosi, <i>Computer Architecture for Power Efficiency</i> at 10 (2008) (“Energy = Watts * runtime”).</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p><i>Finkelstein</i> explains that certain “constraints (e.g., we may not desire to decrease the frequency of a hot core too much to maintain smooth operation of critical applications)” may be used to help determine power allocation among various power planes. <i>Id.</i> ¶ [0024]; <i>see also id.</i> ¶ [0026], ¶ [0031], ¶ [0034], ¶ [0039]. According to <i>Finkelstein</i>, there may be one or more constraints on specific power planes, and/or there may be one or more global constraints on all power planes. <i>Id.</i> ¶ [0026], ¶ [0031]. For instance, the power planes may be subject to one or more global thermal budgets. <i>See id.</i> ¶ [0031]. “In an embodiment, a set of energy budgets, E_n^k, may be maintained, e.g., where k corresponds to a specific constraint, and n to the time step.” <i>Id.</i></p> <p>According to <i>Finkelstein</i>, an SoC may be subject to a global constraint k—e.g., a global thermal budget. <i>See Finkelstein</i> ¶ [0031] (“[T]he system (e.g., logic 140) may set TDP constraints on each of the power planes separately, or/and set TDP constraints on the whole IC package.”); <i>accord Nussbaum</i> ¶ [0021] (“A TDP (Thermal Design Point) represents the power that can be consumed by the entire SOC [A]ssume that maximum power of a 4-Core SOC is limited by a 40 w TDP envelope.”). A POSITA thus would have understood the global thermal budget E^k of <i>Finkelstein</i> is shared among all SoC power domains, including the first domain (comprising processor core 106-1) and the second domain (comprising processor core 106-2). <i>See Finkelstein</i> ¶¶ [0026]-[0041], Fig. 5.</p> <p>Exemplary teachings from the prior art include the following:</p> <p><i>Finkelstein</i> discloses the following at ¶ [0014]:</p> <p style="padding-left: 40px;">Also, the voltage regulator(s) 130 may be coupled to a single power plane 135 (e.g., supplying power to all the cores 106) or to multiple power planes 135 (e.g., where each power plane may supply power to a different core or group of cores).</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p><i>Finkelstein</i> discloses the following at ¶ [0016]:</p> <p>As shown in FIG. 1, the processor 102 may further include a power management logic 140 to control supply of power to components of the processor 102 (e.g., cores 106). Logic 140 may have access to one or more storage devices discussed herein to store information relating to operations of logic 140 such as information communicated with various components of system 100 as discussed here. As shown, the logic 140 may be coupled to the VR 130 and/or other components of system 100 (such as the cores 106). For example, the logic 140 may be coupled to receive information (e.g., in the form of one or more bits or signals) to indicate status of one or more sensors 150 (where the sensor(s) 150 may be provided proximate to components of system 100 (or other computing systems discussed herein such as those discussed with reference to other figures including 4 and 5, for example), such as the cores 106, interconnections 104 or 112, etc., to sense variations in temperature, operating frequency, operating voltage, power consumption, inter-core communication activity, etc.) and/or information from one or more power monitoring logics 145 (e.g., which may indicate the operational status of various components of system 100 such as operating temperature, operating frequency, operating voltage, operating status (e.g., active or inactive), power consumption (instantly or over a period of time), etc.). The logic 140 may instruct the VR 130, power source 120, and/or individual components of system 100 (such as the cores 106) to modify their operations.</p> <p><i>Finkelstein</i> discloses the following at ¶¶ [0028]-[0029]:</p> <p>In some embodiments, energy budget may be managed and/or power setting(s) (e.g., voltage and/or frequency changes) may be made accordingly to a current budget. Energy-based power management may be performed by controlling the energy budget defined iteratively as follows:</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p align="center">$E_{n+1} = \alpha E_n + (TDP_n - P_n) \Delta t_n$</p> <p>where TDP_n is the Thermal Design Power (TDP) power limit on step n; P_n is the power spent on step n over time Δt_n, and α is the decay component.</p> <p><i>Finkelstein</i> discloses the following at ¶ [0031]:</p> <p>Accordingly, the system (e.g., logic 140) may set TDP constraints on each of the power planes separately, or/and set TDP constraints on the whole IC package. According to the imposed constraints, multiple budgets may be maintained in some embodiments, for example per power plane for individual constraints and/ or per package for shared constraints.</p> <p><i>Finkelstein</i> discloses the following at ¶ [0053]:</p> <p>In various embodiments of the invention, the operations discussed herein, e.g., with reference to FIGS. 1-7, may be implemented as hardware (e.g., logic circuitry), software, firmware, or combinations thereof, which may be provided as a computer program product, e.g., including a machine-readable or computer-readable medium having stored thereon instructions (or software procedures) used to program a computer to perform a process discussed herein.</p> <p><i>See also Finkelstein</i> at ¶¶ [0026]-[0027], [0030]. [0032]-[0041].</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
<p>[8b] determining, in the power controller, a portion of the power budget to be allocated to the first and second domains,</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “determining, in the power controller, a portion of the power budget to be allocated to the first and second domains.”</p> <p>According to <i>Finkelstein</i>, “power distribution may be managed among different power planes and under energy-based definition(s).” <i>Finkelstein</i> ¶ [0027]. For example, “energy budget may be managed and/or power setting(s) (e.g., voltage and/or frequency changes) may be made accordingly to a current budget. Energy-based power management may be performed by controlling the energy budget defined iteratively as follows:</p> $E_{n+1} = \alpha E_n + (TDP_n - P_n)\Delta t_n \tag{1}$ <p>where TDP_n is the Thermal Design Power (TDP) power limit on step n; P_n is the power spent on step n over time Δt_n and α is the decay component.” <i>Id.</i> ¶ [0028]-[0029].</p> <p>To allocate a power/energy budget for a constraint k among m associated power planes, <i>Finkelstein</i> teaches using “vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power plane i.” <i>Id.</i> ¶ [0034]. For example, $E^{k,core1}$ would correspond to the portion of the energy budget E^k that goes to the first domain (e.g., processor core 106-1), and $E^{k,core2}$ would correspond to the portion of the energy budget that goes to the second domain (e.g., processor core 106-2). In general, <i>Finkelstein</i> allocates a portion $E^{k,i}$ of the energy budget E^k to the <i>i</i>th power plane based on the following equation: $E^{k,i} = W_{ki}E^k$, such that $E^{k,core1} = W_{k,core1}E^k$, and $E^{k,core2} = W_{k,core2}E^k$. <i>See id.</i> Accordingly, the portion $E^{k,i}$ of the power budget E^k allocated to the <i>i</i>th domain is determined according to the sharing policy value W_{ki} for the <i>i</i>th domain. The portion allocated to the first domain would be $E^{k,core1}$ and the portion allocated to the second domain would be $E^{k,core2}$.</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>Further, <i>Finkelstein</i> describes a power management logic 140 that may “control supply of power to components of the processor 102 (e.g., cores 106).” <i>Finkelstein</i> ¶ [0016]; <i>see also id.</i> at Fig. 1. According to <i>Finkelstein</i>, “the operations discussed herein, e.g., with reference to FIGS. 1-7, may be implemented as hardware (e.g., logic circuitry),” thereby indicating power management logic 140 may be implemented as a “power controller,” as claimed. <i>Finkelstein</i> ¶ [0053].</p> <p><u>Finkelstein-Felter Combination:</u></p> <p>Additionally, as explained above, <i>Finkelstein</i>’s power allocation equation ($E^{k,i} = W_{ki}E^k$) may be rewritten to account for a minimum power allocation as: $E^{k,i} = W_{ki}(E^k - \sum E_{low}^j) + E_{low}^i$. <i>See supra Exemplary Reasons for Modifying Finkelstein’s Power Allocation Equation to Account for a Minimum Power Allocation.</i> Thus, for energy budget E^k, the portion of power allocated to the first domain would be $E^{k,core1} = W_{k,core1}(E^k - \sum E_{low}^j) + E_{low}^{core1}$, and the portion allocated to the second domain would be calculated as $E^{k,core2} = W_{k,core2}(E^k - \sum E_{low}^j) + E_{low}^{core2}$.</p> <p>Exemplary teachings from the prior art include the following:</p> <p><i>Finkelstein</i> discloses the following at ¶¶ [0028]-[0029]:</p> <p style="padding-left: 40px;">In some embodiments, energy budget may be managed and/or power setting(s) (e.g., voltage and/or frequency changes) may be made accordingly to a current budget. Energy-based power management may be performed by controlling the energy budget defined iteratively as follows:</p> $E_{n+1} = \alpha E_n + (TDP_n - P_n) \Delta t_n$ <p style="padding-left: 40px;">where TDP_n is the Thermal Design Power (TDP) power limit on step n; P_n is the power spent on step n over time Δt_n, and α is the decay component.</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations
	<p><i>Finkelstein</i> discloses the following at ¶ [0034]:</p> <p>Regarding the constraints, assume existence of m controllers, $f(E)$ corresponding to different power planes. For constraint k, user (or application defined) preferences that describe how budget E_k is distributed among power planes may be determined. For example, such information may be provided as an input in the form of vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power plane i. For individual constraints, a single power plane may obtain the entire budget, so the corresponding weight vector is a unit vector. In the general case, power plane i may obtain the portion of the budget:</p> $E^{k,i} = W_{ki}E^k$ <p>See also <i>Finkelstein</i> at ¶¶ [0016], [0027], [0032], [0053].</p> <p><i>Felter</i> discloses the following at ¶¶ [0057]-[0061]:</p> <p>FIG. 5 depicts additional details of a method 500 for allocating power according to one embodiment. In the depicted embodiment, method 500 includes determining (block 502) the total power budget (P_{BUDGET}) of the system. This operation may require nothing more than retrieving a static value indicating the maximum rating of the system power supply. In other cases, determining the maximum power may include determining the power that the power supply is currently capable of delivering and may take into consideration external factors such as the ambient temperature and the like, possibly in conjunction with a cooperating service processor, management module, or both.</p> <p>In block 502, the system determines a set of values required to properly allocate the total power budget. The set of values determined in block 504 includes the standby values (minimum power values) of each system component. Using the standby values,</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>an available power figure (P_{AVAIL}) is determined by subtracting from the maximum power deliverable by the system the sum of all the standby values. Under the assumption that each component always draws its standby or minimum power, P_{AVAIL} represents the amount of power that is available for dynamic allocation.</p> <p>As depicted in FIG. 5, allocating the system power includes using the values of predicted activity generated in block 304 (FIG. 3) to compute the amount of active power per component expected during the upcoming interval. These computations require knowledge of the relationship between the monitored activity and the component's power consumption. The implementation illustrated in FIG. 5 uses a linear model for the processor and memory components. Specifically, the active power predicted for a component N is indicated as the product of the activity (UTIL) predicted for component N ($UTIL_N$) and a constant of proportionality for component N (C_N).</p> <p>Method 500 thus includes determining the expected level of active power consumption required by each component N. From this set of calculations, the total amount active power expected to be required for an upcoming interval ($PACT_{TOT}$) is computed as the sum of the individual component's expected active power. Having thus determined the total expected active power and the total amount of standby power required, method 500 includes determining the total required power (P_{TOT}) for all components as the sum of the total active power $PACT_{TOT}$ and the total standby power required $SUM(P_{MIN_N})$.</p> <p>Having determined the power budget and the active and standby demands of each of the components, the depicted embodiment of method 500 includes allocating (block 508) power to the system components. In the embodiment depicted in FIG. 5, for example, power is allocated to the various components used a modified pro rata technique. More specifically, the amount of power budgeted for each component is equal to the component's standby power requirements (which must always be met) and a portion of the available power P_{AVAIL}. The portion of P_{AVAIL} allocated to a system is determined by the ratio of the components active power requirements to the total active</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>power requirements of the system. This formula is represented in block 508 as $P_{ALLOC_N} = P_{MIN_N} + P_{AVAIL} * (P_{ACT_N} / P_{ACT_{TOT}})$. This formula effectively distributes the available system power to the system components in a manner that (1) ensures that each component receives a minimum level of power and (2) allocates the available active power in proportion to the requirement for the components by the workload(s) executing on the system. The formula expressed in block 508 represents a technique for distributing scarce system power equitably so that the system performance of each component is impacted proportionately.</p>
<p>[8c] including allocating a minimum reservation value to the first domain and a minimum reservation value to the second domain,</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “including allocating a minimum reservation value to the first domain and a minimum reservation value to the second domain.”</p> <p>As explained above, <i>Finkelstein</i> teaches a method to dynamically allocate the power budget (e.g., energy budget E^k for a global constraint k) between the first domain (i.e., one or more processor cores 106) and the second domain (i.e., the GPU). <i>See supra</i> [8b].</p> <p><i>Finkelstein</i> teaches that power management logic 140 may dynamically allocate a power budget (e.g., energy budget E^k for a global constraint k) according to a minimum reservation value (e.g., the power supplied by the P_n^{core1} state). <i>See Finkelstein</i> ¶ [0032]. <i>Finkelstein</i> explains that, for each power plane i, power management logic 140 implements “a controller function, denoted by $f^i(E)$,” that “maps [each] energy budget onto the discrete set of power states.” <i>Id.</i> ¶ [0032]. More specifically, $f^i(E)$ is “a nondecreasing function” that “maps a range $[E_{low}^i, E_{high}^i]$ onto the discrete range of $[P_n^i, P_o^i]$, where P_o^i is the maximal turbo state for this power plane, and P_n^i is the maximal efficiency state.” <i>Id.</i> A POSITA would have understood that P_n^i represents a state in which “the performance capability of a device or processor is at its minimum level and consumes minimal power while remaining in an active state.” Advanced Configuration and Power Interface (ACPI) Specification, Rev. 4.0, at 41 (June 16, 2009).</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>“Budget values below E_{low}^i are mapped into P_n^i.” <i>Finkelstein</i> ¶ [0032]. Accordingly, the ith power domain is guaranteed to be further allocated at least the “minimal power” associated with the P_n^i state. Advanced Configuration and Power Interface (ACPI) Specification, Rev. 4.0, at 41 (June 16, 2009).</p> <p>Mathematically, <i>Finkelstein</i> teaches that $f(E^{k,i}) = P_n^i$ when $E^{k,i} \leq E_{low}^i$. <i>Finkelstein</i> ¶ [0032]. In other words, the controller function guarantees at least the power associated with the P_n^i state. In the case of processor core 106-1 and a global thermal budget, the core 1 power allocation $E^{k,core1}$ is thus further allocated by applying a core 1 controller function, such that $f^{core1}(E^{k,core1}) = P_n^{core1}$ when $E^{k,core1} \leq E_{low}^{core1}$. <i>See id.</i> Similarly, the core 2 power allocation $E^{k,core2}$ is further allocated by applying a core 2 controller function: $f^{core2}(E^{k,core2}) = P_n^{core2}$ when $E^{k,core2} \leq E_{low}^{core2}$. <i>See id.</i></p> <p><i>Finkelstein</i> thus teaches dynamically allocating the power budget (e.g., E^k) according to a first minimum reservation value (i.e., the power supplied by the P_n^{core1} state) for the first domain (i.e., processor core 106-1) and a second minimum reservation value (i.e., the power supplied by the P_n^{core2} state) for the second domain (i.e., processor core 106-2). <i>See id.</i></p> <p><u><i>Finkelstein-Felter Combination:</i></u></p> <p>Additionally, as explained above, <i>Finkelstein</i>’s power allocation equation ($E^{k,i} = W_{ki}E^k$) may be rewritten to account for a minimum power allocation as: $E^{k,i} = W_{ki}(E^k - \sum E_{low}^j) + E_{low}^i$. <i>See supra Exemplary Reasons for Modifying <i>Finkelstein</i>’s Power Allocation Equation to Account for a Minimum Power Allocation; Felter</i> ¶¶ [0058]-[0061]; <i>Finkelstein</i> ¶ [0034]. The minimum reservation value of the ith domain is E_{low}^i, the energy budget value corresponding to maximal efficiency state P_n^i. Thus, the minimum reservation value of the first domain (i.e., processor core 106-1) would be E_{low}^{core1} and the minimum reservation value of the second domain (i.e., processor core 106-2) would be E_{low}^{core2}.</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>Exemplary teachings from the prior art include the following:</p> <p><i>Finkelstein</i> discloses the following at ¶ [0032]:</p> <p>In an embodiment, a controller budget may be defined (e.g., implemented by the logic 140). Per power plane i, we define a controller function, denoted by $f^i(E)$, which maps energy budget onto the discrete set of power states. In one embodiment, a controller, which is required to be a nondecreasing function, maps a range $[E_{low}^i, E_{high}^i]$ onto the discrete range of $[P_n^i, P_0^i]$, where P_0^i is the maximal turbo state for this power plane, and P_n^i is the maximal efficiency state. Budget values below E_{low}^i are mapped into P_n^i, while budget values above E_{high}^i, are mapped onto P_0^i.</p> <p><i>Finkelstein</i> discloses the following at ¶ [0034]:</p> <p>Regarding the constraints, assume existence of m controllers, $f^i(E)$ corresponding to different power planes. For constraint k, user (or application defined) preferences that describe how budget E_k is distributed among power planes may be determined. For example, such information may be provided as an input in the form of vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power plane i. For individual constraints, a single power plane may obtain the entire budget, so the corresponding weight vector is a unit vector. In the general case, power plane i may obtain the portion of the budget:</p> $E^{k,i} = W_{ki}E^k$ <p><i>Felter</i> discloses the following at ¶¶ [0057]-[0061]:</p> <p>FIG. 5 depicts additional details of a method 500 for allocating power according to one embodiment. In the depicted embodiment, method 500 includes determining (block 502) the total power budget (P_{BUDGET}) of the system. This operation may require</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>nothing more than retrieving a static value indicating the maximum rating of the system power supply. In other cases, determining the maximum power may include determining the power that the power supply is currently capable of delivering and may take into consideration external factors such as the ambient temperature and the like, possibly in conjunction with a cooperating service processor, management module, or both.</p> <p>In block 502, the system determines a set of values required to properly allocate the total power budget. The set of values determined in block 504 includes the standby values (minimum power values) of each system component. Using the standby values, an available power figure (P_{AVAIL}) is determined by subtracting from the maximum power deliverable by the system the sum of all the standby values. Under the assumption that each component always draws its standby or minimum power, P_{AVAIL} represents the amount of power that is available for dynamic allocation.</p> <p>As depicted in FIG. 5, allocating the system power includes using the values of predicted activity generated in block 304 (FIG. 3) to compute the amount of active power per component expected during the upcoming interval. These computations require knowledge of the relationship between the monitored activity and the component's power consumption. The implementation illustrated in FIG. 5 uses a linear model for the processor and memory components. Specifically, the active power predicted for a component N is indicated as the product of the activity (UTIL) predicted for component N ($UTIL_N$) and a constant of proportionality for component N (C_N).</p> <p>Method 500 thus includes determining the expected level of active power consumption required by each component N. From this set of calculations, the total amount active power expected to be required for an upcoming interval ($PACT_{TOT}$) is computed as the sum of the individual component's expected active power. Having thus determined the total expected active power and the total amount of standby power required, method</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>500 includes determining the total required power (P_{TOT}) for all components as the sum of the total active power $PACT_{TOT}$ and the total standby power required $SUM(PMIN_N)$.</p> <p>Having determined the power budget and the active and standby demands of each of the components, the depicted embodiment of method 500 includes allocating (block 508) power to the system components. In the embodiment depicted in FIG. 5, for example, power is allocated to the various components used a modified pro rata technique. More specifically, the amount of power budgeted for each component is equal to the component's standby power requirements (which must always be met) and a portion of the available power P_{AVAIL}. The portion of P_{AVAIL} allocated to a system is determined by the ratio of the components active power requirements to the total active power requirements of the system. This formula is represented in block 508 as $PALLOC_N = PMIN_N + P_{AVAIL} * (PACT_N / PACT_{TOT})$. This formula effectively distributes the available system power to the system components in a manner that (1) ensures that each component receives a minimum level of power and (2) allocates the available active power in proportion to the requirement for the components by the workload(s) executing on the system. The formula expressed in block 508 represents a technique for distributing scarce system power equitably so that the system performance of each component is impacted proportionately.</p> <p><i>See also Felter</i> at ¶¶ [0010], [0030]-[0031], [0037]-[0049], [0054], [0065].</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
<p>[8d] and sharing a remaining portion of the power budget according to a first sharing policy value for the first domain and a second sharing policy value for the second domain; and</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “sharing a remaining portion of the power budget according to a first sharing policy value for the first domain and a second sharing policy value for the second domain.”</p> <p>As explained above, <i>Finkelstein</i> teaches a method to dynamically allocate the power budget (e.g., energy budget E^k for a global constraint k) between the first domain (i.e., a processor core 106) and the second domain (i.e. the GPU). <i>See supra</i> [8b]. As further explained above, <i>Finkelstein</i> teaches allocating at least a first minimum reservation value (i.e., the power supplied by the P_n^{core1} state) to the first domain (i.e., processor core 106-1) and a second minimum reservation value (i.e., the power supplied by the P_n^{core2} state) to the second domain (i.e., processor core 106-2). <i>See supra</i> [8c].</p> <p><i>Finkelstein</i> also teaches that a remaining portion (i.e., any portion exceeding the minimum reservation values of the power budget) would be allocated according to sharing policy values.</p> <p>To allocate a power budget for a constraint k among m associated power planes, <i>Finkelstein</i> teaches using “vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power plane i.” <i>Finkelstein</i> ¶ [0034]. For example, $E^{k,core1}$ would correspond to the portion of the energy budget E^k that goes to the first domain (e.g., processor core 106-1), and $E^{k,core2}$ would correspond to the portion of the energy budget that goes to the second domain (e.g., processor core 106-2).</p> <p>In general, <i>Finkelstein</i> allocates a portion $E^{k,i}$ of the energy budget E^k to the ith power plane based on the equation: $E^{k,i} = W_{ki}E^k$, such that $E^{k,core1} = W_{k,core1}E^k$, and $E^{k,core2} = W_{k,core2}E^k$. <i>See Finkelstein</i> ¶ [0034]. When $W_{k,i}E^k \geq E_{low}^i$, power management logic 140 assigns the ith domain to a performance state P_{n+q}^i with a power exceeding the ith minimum reservation value (i.e., the power supplied by the P_n^i performance state) according to the ith sharing policy value (i.e., $W_{k,i}$). <i>See id.</i> ¶ [0032], ¶¶ [0034]-[0036].</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations
	<p>More specifically, <i>Finkelstein</i> teaches the recommended power state (i.e., the recommended P-state P^i) for the ith power plane is the minimum $f^i(E)$ considering all constraints k that apply to power plane i:</p> $P^i = \min_k f^i(E^{k,i}) = \min_k f^i(W_{k,i}E^k)$ <p><i>See id.</i> ¶ [0036]. For a single, global constraint k, $P^i = f^i(W_{k,i}E^k)$, such that the following holds: $P^{core1} = f^{core1}(W_{k,core1}E^k)$, and $P^{core2} = f^{core2}(W_{k,core2}E^k)$. <i>See id.</i> ¶ [0032], ¶¶ [0034]-[0036]. Writing the power of the P^i performance state as $Pow(P^i)$, a POSITA would have understood the power of the ith performance state for a single, global constraint k is: $Pow(f^i(W_{k,i}E^k))$. <i>See id.</i> Accordingly, the remaining portion of the power budget for the ith domain is: $Pow(f^i(W_{k,i}E^k)) - Pow(P_n^i)$, which depends on the sharing policy value $W_{k,i}$ for the ith domain. <i>See id.</i> ¶ [0034].</p> <p>Thus, for a single, global constraint k, a POSITA would have understood power management logic 140 shares a remaining portion (i.e., $Pow(f^i(W_{k,core1}E^k)) - Pow(P_n^{core1})$) of the power budget (i.e., E^k) according to a first sharing policy value (i.e., $W_{k,core1}$) for the first domain (i.e., processor core 106-1). And a POSITA would have understood power management logic 140 shares the remaining portion (i.e., $Pow(f^i(W_{k,core2}E^k)) - Pow(P_n^{core2})$) of the power budget (i.e., E^k) according to a second sharing policy value (i.e., $W_{k,core2}$) for the second domain (i.e., processor core 106-2).</p> <p><u>Finkelstein-Felter Combination:</u></p> <p>Additionally, as explained above, the power budget (e.g., energy budget E^k for a global constraint k) for the combination of <i>Finkelstein</i> and <i>Felter</i> is allocated to the ith domain as $E^{k,i} = W_{k,i}(E^k - \sum E_{low}^j) + E_{low}^i$. <i>See supra</i> [8b]. Since E_{low}^i represents the minimum reservation value of the ith domain, the difference between the power budget E^k and the sum of all minimum reservation values (i.e., $E^k - \sum E_{low}^j$) is the remaining portion of the power budget. Since $E^{k,i} = W_{k,i}(E^k - \sum E_{low}^j) + E_{low}^i$, the</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>remaining portion of the power budget (i.e., $E^k - \sum E_{low}^j$) is allocated according to the ith domain according to its policy sharing value $W_{k,i}$. See <i>supra</i> [8b].</p> <p>Thus, power management logic 140 distributes the remaining portion (i.e., $E^k - \sum E_{low}^j$) of the power budget (e.g., E^k) according to a first sharing policy value (i.e., $W_{k,core1}$) for the first domain (processor core 106-1), such that the first domain is allocated $W_{k,core1}(E^k - \sum E_{low}^j)$ of the remaining portion. And power management logic 140 distributes the remaining portion (i.e., $E^k - \sum E_{low}^j$) of the power budget (e.g., E^k) according to a second sharing policy value (i.e., $W_{k,core2}$) for the second domain (i.e., processor core 106-2), such that the second domain is allocated $W_{k,core2}(E^k - \sum E_{low}^j)$ of the remaining portion.</p> <p>Exemplary teachings from the prior art include the following:</p> <p><i>Finkelstein</i> discloses the following at ¶ [0032]:</p> <p style="padding-left: 40px;">In an embodiment, a controller budget may be defined (e.g., implemented by the logic 140). Per power plane i, we define a controller function, denoted by $f^i(E)$, which maps energy budget onto the discrete set of power states. In one embodiment, a controller, which is required to be a nondecreasing function, maps a range $[E_{low}^i, E_{high}^i]$ onto the discrete range of $[P_n^i, P_0^i]$, where P_0^i is the maximal turbo state for this power plane, and P_n^i is the maximal efficiency state. Budget values below E_{low}^i are mapped into P_n^i, while budget values above E_{high}^i, are mapped onto P_0^i.</p> <p><i>Finkelstein</i> discloses the following at ¶¶ [0034]-[0036]:</p> <p style="padding-left: 40px;">Regarding the constraints, assume existence of m controllers, $f^i(E)$ corresponding to different power planes. For constraint k, user (or application defined) preferences that describe how budget E_k is distributed among power planes may be determined. For example, such information may be provided as an input in the form of vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>plane <i>i</i>. For individual constraints, a single power plane may obtain the entire budget, so the corresponding weight vector is a unit vector. In the general case, power plane <i>i</i> may obtain the portion of the budget:</p> $E^{k,i} = W_{ki}E^k$ <p>In some cases, we may have that for a power plane <i>i</i>, its portion $W_{ki}E_k$ is high enough to provide maximal turbo (namely, $W_{ki}E^k \geq E_{high}^i$). In this case, the “unused” budget for this power plane, $W_{ki}E^k - E_{high}^i$, may be distributed among the rest of the power planes proportionally to their weights.</p> <p>Let us denote by $E_n^{k,i}$ the portion of the budget E^k that receives power plane <i>i</i> at step t_n, then the resulting P-state recommendation on this step for power plane <i>i</i> may be written as:</p> $u_n^i = \min_k f^i(E_n^{k,i})$ <p><i>See also Felter</i> at ¶¶ [0057]-[0061], [0065].</p>
<p>[8e] controlling a frequency of the first domain and a frequency of the second domain based on the allocated portions.</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “controlling a frequency of the first domain and a frequency of the second domain based on the allocated portions.”</p> <p><i>Finkelstein</i> teaches that the allocated portions of the energy budget may be mapped onto discrete power states. <i>See Finkelstein</i> ¶ [0032] (“Per power plane <i>i</i>, we define a controller function, denoted by $f^i(E)$, which maps energy budget onto the discrete set of power states.”). For example, a non-decreasing function may map “a range $[E_{low}^i, E_{high}^i]$ onto the discrete range of $[P_n^i, P_0^i]$, where P_0^i is the maximal turbo state for this power plane, and P_n^i is the maximal efficiency state.” <i>Id.</i> Thus, based on the</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>allocated portion of the energy budget, the power plane will be assigned to a performance state. <i>See id.</i> ¶ [0032], ¶¶ [0034]-[0036] (describing the P-state recommendation process).</p> <p>A POSITA would have understood that a performance state is defined by a frequency value and a voltage value. <i>See, e.g., Nussbaum</i> ¶ [0033] (“A performance state is characterized by a unique pair of core voltage and frequency values.”); <i>Brinks</i> ¶ [0055] (“[T]he CPU domain state will be expressed by the (F, V) doublet.”). A POSITA thus would have understood power management logic 140 controls the frequency (F) of each domain (i.e., the one or more processor cores 106 and the GPU) by selecting their performance states based on the allocated portions of the power budget. <i>See Finkelstein</i> ¶ [0032], ¶¶ [0034]-[0036]; <i>see supra</i> [8b]-[8d].</p> <p>Exemplary teachings from the prior art include the following:</p> <p><i>Finkelstein</i> discloses the following at ¶ [0032]:</p> <p style="padding-left: 40px;">In an embodiment, a controller budget may be defined (e.g., implemented by the logic 140). Per power plane i, we define a controller function, denoted by $f^i(E)$, which maps energy budget onto the discrete set of power states. In one embodiment, a controller, which is required to be a nondecreasing function, maps a range $[E_{low}^i, E_{high}^i]$ onto the discrete range of $[P_n^i, P_0^i]$, where P_0^i is the maximal turbo state for this power plane, and P_n^i is the maximal efficiency state. Budget values below E_{low}^i are mapped into P_n^i, while budget values above E_{high}^i, are mapped onto P_0^i.</p> <p><i>Finkelstein</i> discloses the following at ¶¶ [0034]-[0036]:</p> <p style="padding-left: 40px;">Regarding the constraints, assume existence of m controllers, $f^i(E)$ corresponding to different power planes. For constraint k, user (or application defined) preferences that describe how budget E_k is distributed among power planes may be determined. For example, such information may be provided as an input in the form of vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>plane i. For individual constraints, a single power plane may obtain the entire budget, so the corresponding weight vector is a unit vector. In the general case, power plane i may obtain the portion of the budget:</p> $E^{k,i} = W_{ki}E^k$ <p>In some cases, we may have that for a power plane i, its portion $W_{ki}E_k$ is high enough to provide maximal turbo (namely, $W_{ki}E^k \geq E_{high}^i$). In this case, the “unused” budget for this power plane, $W_{ki}E^k - E_{high}^i$, may be distributed among the rest of the power planes proportionally to their weights.</p> <p>Let us denote by $E_n^{k,i}$ the portion of the budget E^k that receives power plane i at step t_n, then the resulting P-state recommendation on this step for power plane i may be written as:</p> $u_n^i = \min_k f^i(E_n^{k,i})$
Claim 9	
9. The method of claim 8, wherein	<i>Finkelstein</i> , alone or in combination with one or more references, discloses or suggests the method of claim 8. <i>See supra</i> claim 8.

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
<p>determining the portion of the power budget to be allocated to the first and second domains includes allocating the power budget to the second domain and not to the first domain if the power budget is less than the minimum reservation value for the second domain.</p>	<p>Claims 9 through 11 are invalid under 35 U.S.C. § 112, first paragraph, because they are incompatible with claim 8, from which they depend. Additionally, under Daedalus’s apparent view of the claims, they are invalid as anticipated and/or obvious in view of one or more prior art references.</p> <p>Under Daedalus’s apparent view of the claims, <i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “determining the portion of the power budget to be allocated to the first and second domains includes allocating the power budget to the second domain and not to the first domain if the power budget is less than the minimum reservation value for the second domain.”</p> <p>If the power budget is less than the minimum reservation value of the second domain, then it would be impossible for the system to allocate enough power to each domain to satisfy both minimum reservation values. A POSITA would have understood the system must determine how to allocate the available power budget under such circumstances.</p> <p>In this scenario, there would be only two possibilities: (1) allocate all of the power budget to one domain or (2) split the power budget between the domains.</p> <p>In <i>KSR Int’l Co. v. Teleflex Inc.</i>, the Supreme Court held that when “there is a design need . . . to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp.” 550 U.S. 398, 421 (2007). Here, a POSITA would have understood there is a design need to allocate a power budget when there is not enough power to meet the minimum reservation requirements, and there are only two “identified, predictable solutions.” <i>Id.</i></p> <p>Two solutions is a “finite number.” For example, in <i>Uber Techs., Inc. v. XOne, Inc.</i>, the Federal Circuit found there were only two possible methods for displaying a map with plotted locations: “server-side plotting and terminal-side plotting.” 957 F.3d 1334, 1339 (Fed. Cir. 2020). Because of this, the Federal Circuit determined there were only a “finite number of identified, predictable solutions,” and determined the claims at issue were obvious. <i>Id.</i>; see also <i>Google LLC v. Koninklijke Philips N.V.</i>, 795</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>F. App'x 840, 841 (Fed. Cir. 2020) (non-precedential); <i>Cardiovalve Ltd. v. Edwards Lifesciences Corp.</i>, No. 2022-2230, 2024 WL 1208638, at *2 (Fed. Cir. Mar. 21, 2024) (non-precedential). Here, as in <i>Uber</i>, there are only two possible solutions: assign all the power to one domain or split it between the domains.</p> <p>Likewise, in <i>Perfect Web Techs., Inc. v. InfoUSA, Inc.</i>, Perfect Web asserted a patent for “managing bulk e-mail distribution to groups of targeted consumers.” 587 F.3d 1324, 1326 (Fed. Cir. 2009). Step (D) of representative claim 1 required repeating steps (A)-(C) if the calculated quantity of emails does not exceed a prescribed minimum quantity of successfully received e-mails. <i>Id.</i> The Federal Circuit only identified three possible solutions for the problem step (D) attempts to solve: “(1) oversending, or e-mailing an excess of addresses to ensure the quota is met; (2) if some addresses failed or “bounced” back messages, re-sending to those same addresses in the hope that a second transmission somehow succeeds; and (3) identifying a new group of addresses and sending messages to them, which is step (D).” <i>Id.</i> at 1331. The Federal Circuit concluded there were only a “finite number of identified, predictable solutions,” i.e., three solutions. <i>Id.</i> Accordingly, the method proposed by Perfect Web in step (D) “would have been obvious.” <i>Id.</i> because two solutions is fewer than three solutions, it follows from <i>Perfect Web</i> that two possible solutions would also represent a “finite number of identified, predictable solutions.” <i>Id.</i></p> <p>A POSITA would have needed to account for all possible power supply conditions, including the claimed conditions, when designing a system. Although a system may lack sufficient power to obtain a minimum amount of work from either domain under the claimed conditions, a POSITA would have understood a domain may draw power for other purposes over the next interval, not just performing work. This includes, for instance, maintaining domain cache memories and volatile device contexts. <i>See, e.g.</i>, Advanced Configuration and Power Interface (ACPI) Specification, Rev. 4.0, at 32, 35, 38-41, 46, 51, 298-311, 333, 483-89 (June 16, 2009). Otherwise, the domain may need to be put to sleep. <i>See id.</i></p> <p>Here, when considering the claimed circumstances, a POSITA would have understood one of the domains (e.g., the processor core 106-2) should receive a <i>de minimis</i> amount of power to avoid putting</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>it to sleep, thereby avoiding the attendant delays caused by moving volatile cache memory states and device contexts into non-volatile memory (and back again once the domain wakes up). <i>See id.</i> Because putting one domain to sleep rather than two improves system responsiveness, a POSITA would have found it obvious under the claimed conditions to allocate all the power budget to one of the domains (e.g., the processor core 106-2) to avoid putting it to sleep.</p>
Claim 10	
<p>10. The method of claim 9, further comprising</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests the method of claim 9. <i>See supra</i> claim 9.</p>
<p>obtaining the minimum reservation value for the second domain from a configuration register written by user-level software.</p>	<p>Claims 9 through 11 are invalid under 35 U.S.C. § 112, first paragraph, because they are incompatible with claim 8, from which they depend. Additionally, under Daedalus’s apparent view of the claims, they are invalid as anticipated and/or obvious in view of one or more prior art references.</p> <p>Under Daedalus’s apparent view of the claims, <i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “obtaining the minimum reservation value for the second domain from a configuration register written by user-level software.”</p> <p><u><i>Finkelstein-Felter</i> Combination:</u></p> <p>As explained above, the combination of <i>Finkelstein</i> and <i>Felter</i> describes a minimum reservation value for the second domain (E_{low}^{core2}). <i>See supra</i> [8c]. A POSITA would have found it obvious to obtain the minimum reservation value for the second domain from a configuration register written by user-level software.</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>A POSITA would have found it obvious to control the minimum reservation values (i.e., E_{low}^{core1} and E_{low}^{core2}) with user-level software. <i>Finkelstein</i> teaches, for example, that the inputs to the sharing policy vectors $W_{k,i}$ are “user (or application defined) preferences.” <i>Finkelstein</i> ¶ [0034]. From this, a POSITA would have found it obvious to control the minimum reservation values (i.e., E_{low}^{core1} and E_{low}^{core2}) with user-level software as well, simply using another user-defined vector that contains the E_{low}^i values. <i>See id.</i> ¶ [0032]. A POSITA would have found it beneficial to do so to provide the user with additional control over the user experience by balancing the tradeoff between performance and power consumption. <i>See, e.g., Brinks</i> ¶ [0051] (“The policy defines the tradeoffs between user experience and power consumption. Accordingly, the . . . user may determine a set of policies that appropriately balance these tradeoffs.”); <i>de Cesare</i> ¶ [0009] (“[T]he configureability [sic] of the performance states may permit more fine-grained control of the performance level in the system”); <i>Gwennap</i> at 6 (“If the GPU is not in use or is operating at a low speed, the power manager can assign some of its thermal budget to the CPUs, allowing them to run at a higher clock speed. Conversely, on a graphics-intensive application, the power manager can overclock the GPU while slowing down the CPUs.”).</p> <p>A POSITA would have known a common way to control power management features is to obtain them from a configuration register. <i>See, e.g., Brinks</i> ¶ [0049] (“[P]ower management policies may be stored in a table, hierarchical table, register, matrix or some combination of the three”); <i>de Cesare</i> ¶ [0046] (“Whether or not performance domains transition may be hardware controlled by the PMU 28, or may be software controlled using the valid indications in the performance configuration registers as described previously.”). A POSITA would have also understood that such configuration registers may be written by user-level software. <i>See, e.g., Brinks</i> ¶ [0050] (“[T]he table, register, or matrix may be programmable, so that its behavior can be changed at run-time to support different chip operating modes.”); <i>de Cesare</i> ¶ [0041] (“The selection may be made by writing the pointer in each register corresponding to the performance domain in the registers 30B.”). Because <i>Finkelstein</i> teaches that the policy sharing values are controllable by user-level software, a POSITA would have found it obvious to obtain the minimum reservation values from a configuration register written by user-level software.</p> <p>Exemplary teachings from the prior art include the following:</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p><i>Finkelstein</i> discloses the following at ¶ [0032]:</p> <p>In an embodiment, a controller budget may be defined (e.g., implemented by the logic 140). Per power plane i, we define a controller function, denoted by $f^i(E)$, which maps energy budget onto the discrete set of power states. In one embodiment, a controller, which is required to be a nondecreasing function, maps a range $[E_{low}^i, E_{high}^i]$ onto the discrete range of $[P_n^i, P_0^i]$, where P_0^i is the maximal turbo state for this power plane, and P_n^i is the maximal efficiency state. Budget values below E_{low}^i are mapped into P_n^i, while budget values above E_{high}^i, are mapped onto P_0^i.</p> <p><i>Finkelstein</i> discloses the following at ¶ [0034]:</p> <p>Regarding the constraints, assume existence of m controllers, $f^i(E)$ corresponding to different power planes. For constraint k, user (or application defined) preferences that describe how budget E_k is distributed among power planes may be determined. For example, such information may be provided as an input in the form of vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power plane i. For individual constraints, a single power plane may obtain the entire budget, so the corresponding weight vector is a unit vector. In the general case, power plane i may obtain the portion of the budget:</p> $E^{k,i} = W_{ki}E^k$ <p><i>Brinks</i> discloses the following at ¶ [0049]:</p> <p>The power management policies may be stored in a table, hierarchical table, register, matrix or some combination of the three that map different system operating states to different operating points 406 to improve and/or maximize energy savings for a given policy. The policies may be configured and/or programmed into the table, register, or matrix by a chip designer. The chip designer may run multiple simulations to perfect</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>knowing which components affect other components when the chip runs and how much power is saved. The chip designer may make a matrix of power saving to be referenced and which components can be powered off for small granular time periods depending on which application is running.</p> <p><i>Brinks</i> discloses the following at ¶ [0051]:</p> <p>In an exemplary embodiment, the chip designer or user will be responsible for providing the power management policies. The policy defines the tradeoffs between user experience and power consumption. Accordingly, the chip designer or user may determine a set of policies that appropriately balance these tradeoffs. An example power management policy 404 may favor faster performance over battery life, or vice-versa.</p> <p><i>See also Brinks</i> at ¶¶ [0022], [0046]-[0047], [0052]-[0061], [0062]-[0066], [0071].</p>
Claim 11	
<p>11. The method of claim 9, further comprising</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests the method of claim 9. <i>See supra</i> claim 9.</p>
<p>allocating the minimum reservation value for the second domain to the second domain, and allocating a remaining</p>	<p>Claims 9 through 11 are invalid under 35 U.S.C. § 112, first paragraph, because they are incompatible with claim 8, from which they depend. Additionally, under Daedalus's apparent view of the claims, they are invalid as anticipated and/or obvious in view of one or more prior art references.</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
<p>portion of the power budget to the first domain when the power budget is greater than the minimum reservation value for the second domain but less than a sum of the minimum reservation value for the second domain and the minimum reservation value for the first domain.</p>	<p>Under Daedalus’s apparent view of the claims, <i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “allocating the minimum reservation value for the second domain to the second domain, and allocating a remaining portion of the power budget to the first domain when the power budget is greater than the minimum reservation value for the second domain but less than a sum of the minimum reservation value for the second domain and the minimum reservation value for the first domain.”</p> <p>If the power budget is less than the sum of the minimum reservation value of the second domain and the minimum reservation value of the first domain, then it would be impossible for the system to allocate enough power to each domain to meet both minimum reservation values. A POSITA would have understood the system must determine how to allocate the available power budget under such circumstances.</p> <p>In this scenario, there would be three ways to allocate the power budget: (1) all of the power budget could be allocated to one domain, (2) the minimum reservation value of one domain may be allocated to that domain and the remaining available power budget may be allocated to the other domain, or (3) the power budget may be split between the two domains such that neither domain is allocated enough power to meet its minimum reservation value.</p> <p>In <i>KSR Int’l Co. v. Teleflex Inc.</i>, the Supreme Court held that when “there is a design need . . . to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp.” 550 U.S. 398, 421 (2007). Here, there are only three possible solutions, which is a finite number. For example, in <i>Perfect Web Techs., Inc. v. InfoUSA, Inc.</i>, Perfect Web asserted a patent for “managing bulk e-mail distribution to groups of targeted consumers.” 587 F.3d 1324, 1326 (Fed. Cir. 2009). Step (D) of representative claim 1 required repeating steps (A)-(C) if the calculated quantity of emails does not exceed a prescribed minimum quantity of successfully received e-mails. <i>Id.</i> The Federal Circuit only identified three possible solutions for the problem step (D) attempts to solve: “(1) oversending, or e-mailing an excess of addresses to ensure the quota is met; (2) if some addresses failed or ‘bounced’ back messages, re-sending to those same addresses in the hope that a second transmission somehow succeeds; and (3)</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>identifying a new group of addresses and sending messages to them, which is step (D).” <i>Id.</i> at 1331. The Federal Circuit concluded there were only a “finite number of identified, predictable solutions,” i.e., three solutions. <i>Id.</i> Accordingly, the method proposed by Perfect Web in step (D) “would have been obvious.” <i>Id.</i></p> <p>A POSITA would have needed to account for all possible power supply conditions when designing the system, including the claimed circumstances. Although the system may lack sufficient power to obtain the minimum amount of work from the first domain under the claimed circumstances, a POSITA would have understood that the first domain may draw power for other purposes over the next interval, too, not just performing work. This includes, for instance, maintaining domain cache memories and volatile device contexts. <i>See, e.g.,</i> Advanced Configuration and Power Interface (ACPI) Specification, Rev. 4.0, at 32, 35, 38-41, 46, 51, 298-311, 333, 483-89 (June 16, 2009). Otherwise, the first domain may need to be put into a sleep state. <i>See id.</i></p> <p>A POSITA thus would have found it obvious under the claimed conditions (i.e., when power budget E^k is such that the power supplied by the P_n^{core2} state $< E^k <$ the power supplied by the P_n^{core2} state + the power supplied by the P_n^{core1} state) to allocate the minimum reservation value for the second domain to the second domain, thereby allowing the processor core 106-2 to process a workload at the desired performance level, and to allocate the remaining portion of the power budget (i.e., $E^k -$ the power supplied by the P_n^{core2} state) to the first domain (e.g., processor core 106-1) to either process a workload with worse performance, or, failing that, avoid putting the processor core 106-1 into a sleep state, which would slow the system. <i>See supra</i> [9].</p> <p><u>Finkelstein-Felter Combination:</u></p> <p>Additionally, as explained above, <i>Finkelstein</i>’s power allocation equation ($E^{k,i} = W_{ki}E^k$) may be rewritten to account for a minimum power allocation as: $E^{k,i} = W_{ki}(E^k - \sum E_{low}^j) + E_{low}^i$. <i>See supra Exemplary Reasons for Modifying Finkelstein’s Power Allocation Equation to Account for a Minimum Power Allocation; Felter ¶¶ [0058]-[0061]; Finkelstein ¶ [0034]</i>. In this case, a POSITA would have found it obvious under the claimed conditions (i.e., when power budget E^k is such that</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>$E_{low}^{core2} < E^k < E_{low}^{core2} + E_{low}^{core1}$) to allocate the minimum reservation value for the second domain to the second domain, thereby allowing the processor core 106-2 to process a workload at the desired performance level, and to allocate the remaining portion of the power budget (i.e., $E^k - E_{low}^{core2}$) to the first domain (e.g., processor core 106-1) to either process a workload with worse performance, or, failing that, avoid putting the processor core 106-1 into a sleep state, which would slow the system. <i>See supra</i> [9].</p>
Claim 12	
<p>12. The method of claim 8, wherein</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests the method of claim 8. <i>See supra</i> claim 8.</p>
<p>the first sharing policy value is to be incremented when a request for a higher frequency for the first domain is not granted, and the second sharing policy value is to be incremented when a request for a higher frequency for the second domain is not granted.</p>	<p><i>Finkelstein</i>, alone or in combination with one or more references, discloses or suggests “the first sharing policy value is to be incremented when a request for a higher frequency for the first domain is not granted, and the second sharing policy value is to be incremented when a request for a higher frequency for the second domain is not granted.”</p> <p><u><i>Finkelstein-Felter</i> Combination:</u></p> <p>As explained above, a POSITA would have found it obvious to calculate <i>Finkelstein</i>'s $W_{k,i}$ values according to component active power requirements. <i>See supra Exemplary Reasons for Determining Sharing Policy Values Based on Component Active Power Requirements</i>. For example, the entry for a domain N (e.g., $W_{k,N}$) would be equal to $(PACT_N/PACT_{TOT})$. <i>See Felter</i> ¶¶ [0058]-[0061], Fig. 5. $PACT_N$ represents the “amount of active power . . . expected during the upcoming interval” for</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations
	<p>domain N. <i>Felter</i> ¶ [0059]. $PACT_N$ is calculated “as the product of the activity (UTIL) predicted for component N ($UTIL_N$) and a constant of proportionality for component N (C_N).” <i>Id.</i></p> <p>A POSITA would have understood that if a request for a higher frequency was not granted, the amount of expected power, based on the predicted activity, was too low. <i>See, e.g., Nussbaum</i> ¶ [0037] (“[T]he power of a particular computational unit (voltage × current) is based on the frequency of the clock signal, the supply voltage, and the amount of activity in the computational unit.”). The expected activity in subsequent intervals, and thus the corresponding $PACT_N$ value, would increase. For example, <i>Felter</i> explains that the system may predict activity using a behavior model 400. <i>See Felter</i> ¶ [0051], Fig. 4.</p> <div data-bbox="835 662 1654 1128" data-label="Diagram"> <pre> graph TD A[CURRENT ACTIVITY OF SUBJECT SYSTEM 402] --> C[ACTIVITY MODEL 401] B[CURRENT ACTIVITY OF OTHER SYSTEMS 404] --> C D[EXTERNAL FACTORS (TIME, DAY, 406)] --> C E[HISTORICAL ACTIVITY DATA 408] --> C C --> F((ACTIVITY PREDICTION 410)) </pre> </div> <p>The behavior model 400 receives inputs from current activity input 402, representing “the current activity levels of the subject component.” <i>Felter</i> ¶ [0052]. The behavior model 400 may use the current activity input 402 to “generate an activity prediction 410 that is used to develop a power allocation for the corresponding component.” <i>Felter</i> ¶ [0053]. Accordingly, a POSITA would have understood that if the current activity was higher than had been predicted for an interval, and a request for a higher</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>frequency was not granted, the behavior model 400 would compute a higher activity prediction, and thus a higher PACT_N value for the next interval. <i>See Felter</i> ¶¶ [0052]-[0053], ¶ [0059].</p> <p>Exemplary teachings from the prior art include the following:</p> <p><i>Finkelstein</i> discloses the following at ¶ [0034]:</p> <p>Regarding the constraints, assume existence of m controllers, f(E) corresponding to different power planes. For constraint k, user (or application defined) preferences that describe how budget E_k is distributed among power planes may be determined. For example, such information may be provided as an input in the form of vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power plane i. For individual constraints, a single power plane may obtain the entire budget, so the corresponding weight vector is a unit vector. In the general case, power plane i may obtain the portion of the budget:</p> $E^{k,i} = W_{ki}E^k$ <p><i>Felter</i> discloses the following at ¶¶ [0052]-[0053]:</p> <p>Referring to FIG. 4 momentarily, a block diagram of selected elements of a behavior model 400 includes an activity modeling module 401 that receives input in the form of current activity inputs 402 and 404, external factors 406, and historical activity data 408. Current activity inputs 402 indicates the current activity levels of the subject component (the component for which activity is being predicted) whereas activity inputs 404 represents the activity levels of other components in the dynamic power allocation domain. External factors 406 may include factors such as the time of day, day of week, and the like.</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>Historical activity data 408, as its name suggests, includes archived activity data for one or more of the system components. Activity model 401 processes all or some of these inputs to generate an activity prediction 410 that is used to develop a power allocation for the corresponding component. Activity model may incorporate substantially any prediction technique or prediction algorithm that is compatible with the timing requirements of power administrator 200 (i.e., the prediction algorithm should not create a bottle neck for power administrator 200).</p> <p><i>Felter</i> discloses the following at ¶¶ [0059]-[0060]:</p> <p>As depicted in FIG. 5, allocating the system power includes using the values of predicted activity generated in block 304 (FIG. 3) to compute the amount of active power per component expected during the upcoming interval. These computations require knowledge of the relationship between the monitored activity and the component's power consumption. The implementation illustrated in FIG. 5 uses a linear model for the processor and memory components. Specifically, the active power predicted for a component N is indicated as the product of the activity (UTIL) predicted for component N ($UTIL_N$) and a constant of proportionality for component N (C_N).</p> <p>Method 500 thus includes determining the expected level of active power consumption required by each component N. From this set of calculations, the total amount active power expected to be required for an upcoming interval ($PACT_{TOT}$) is computed as the sum of the individual component's expected active power. Having thus determined the total expected active power and the total amount of standby power required, method</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

U.S. Patent No. 8,769,316	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>500 includes determining the total required power (P_{TOT}) for all components as the sum of the total active power $PACT_{TOT}$ and the total standby power required $SUM(PMIN_N)$.</p> <p><i>See also Felter</i> at ¶¶ [0050]-[0051], [0057]-[0058], [0061]-[0066], Figs. 3-5.</p> <p><u>Additional Combination with <i>Bose-602</i>:</u></p> <p>As explained above, a POSITA would have found it obvious that the $W_{k,i}$ values would be incremented when a request for a higher frequency is not granted. <i>See supra Exemplary Reasons for Incrementing the Sharing Policy Value when a Request for a Higher Frequency is not Granted.</i></p> <p>Exemplary teachings from the prior art include the following:</p> <p><i>Finkelstein</i> discloses the following at ¶ [0034]:</p> <p>Regarding the constraints, assume existence of m controllers, $f^i(E)$ corresponding to different power planes. For constraint k, user (or application defined) preferences that describe how budget E_k is distributed among power planes may be determined. For example, such information may be provided as an input in the form of vectors W_k of length m, such that entry i corresponds to the portion of the budget that goes to power plane i. For individual constraints, a single power plane may obtain the entire budget, so the corresponding weight vector is a unit vector. In the general case, power plane i may obtain the portion of the budget:</p> $E^{k,i} = W_{ki}E^k$ <p><i>Bose-602</i> discloses the following at ¶ [0013]:</p>

Appendix 316-02: Invalidity of U.S. Patent No. 8,769,316 (the '316 Patent) Based on *Finkelstein*

<u>U.S. Patent No. 8,769,316</u>	<u>Exemplary Disclosures Relevant to <i>Finkelstein</i> and Related Combinations</u>
	<p>According to an exemplary embodiment of the present invention, a method of power management of a system of connected components includes initializing a token allocation map across the connected components, wherein each component is assigned a power budget as determined by a number of allocated tokens in the token allocation map, monitoring utilization sensor inputs and command state vector inputs, determining, at first periodic time intervals, a current performance level, a current power consumption level and an assigned power budget for the system based on the utilization sensor inputs and the command state vector inputs, and determining, at second periodic time intervals, a token re-allocation map based on the current performance level, the current power consumption level and the assigned power budget for the system, according to a re-assigned power budget of at least one of the connected components, while enforcing a power consumption limit based on a total number of allocated tokens in the system.</p> <p><i>Bose-602</i> discloses the following at ¶ [0086]:</p> <p>Alternatively, if the resource is below power and temperature limits, its performance level is gauged in block 1314. If it is assessed to be below a predefined minimum performance, block 1316 is used to try and read additional tokens from the centralized bus token storage; such extra tokens are used to boost up performance in the targeted unit in block 1317 (e.g., by increasing its voltage and frequency, if allowed by the system architecture) and then block 1315 is again entered to check if the resource is within the power and temperature limits.</p>