Technical Note on "Composite Theoretical Performance" (Con't):

Details for these steps are given in the following sections.

Note 1:
For aggregations of multiple "CEs" which have both shared and unshared memory subsystems, the calculation of "CTP" is completed hierarchically, in two steps: first, aggregate the groups of "CEs" sharing memory; second, calculate the "CTP" of the groups using the calculation method for multiple "CEs" not sharing memory.

Note 2:
"CEs" that are limited to input/output and peripheral functions (e.g., disk drive, communication and video display controllers) are not aggregated into the "CTP" calculation.

Note W:
For a pipelined "CE" capable of executing up to one arithmetic or logic operation every clock cycle after the pipeline is full, a pipelined rate can be established. The effective calculating rate (R) for such a "CE" is the faster of the pipelined rate or non-pipelined execution rate.

Note X:
For a "CE" which performs multiple operations of a specific type in a single cycle (e.g., two additions per cycle or two identical logic operations per cycle), the execution time t is given by:

$$t = \frac{\text{cycle time}}{\text{the \# of identical operations per machine cycle}}$$

"CEs" which perform different types of arithmetic or logic operations in a single machine cycle are to be treated as multiple separate "CEs" performing simultaneously (e.g., a "CE" performing an addition and a multiplication in one cycle is to be treated as two "CEs", the first performing an addition in one cycle and the second performing a multiplication in one cycle).
If a single "CE" has both scalar function and vector function, use the shorter execution time value.

Note Y:
For the "CE" that does not implement FP add or FP multiply, but that performs FP divide:

$$R_{fp} = \frac{1}{t_{fp\ divide}}$$

If the "CE" implements FP reciprocal but not FP add, FP multiply or FP divide, then:

$$R_{fp} = \frac{1}{t_{fp\ reciprocal}}$$

If none of the specified instructions is implemented, the effective FP rate is 0.

The following table shows the method of calculating the Effective Calculating Rate R for each "CE": ***Step 1: The Effective Calculating Rate R:***	
For "CEs" Implementing: Note: Every "CE" must be evaluated independently	Effective calculating Rate, R

XP only
 	
(R_{xp})
	$\frac{1}{3 * (t_{xp\ add})}$
	if no add is implemented use:
	$\frac{1}{(t_{xp\ mult})}$
	if neither add nor multiply is implemented use the fastest available arithmetic operation as follows:
	$\frac{1}{3 * t_{xp}}$
	See Notes X and Z.

FP Only (R_{fp})	**max** $\frac{1}{t_{xp\ add}}$, $\frac{1}{t_{fp\ mult}}$ See Notes X and Y.
Both FP and XP (R)	Calculate both R xp, R fp
For simple logic processors not implementing any of the specified arithmetic operations.	$\frac{1}{3 * t_{log}}$ Where t log is the execute time of the XOR, or for logic hardware not implementing the XOR, the fastest simple logic operation. See Notes X and Z.
For special logic processors not using any of the specified arithmetic or logic operations.	R = R' * WL / 64 Where R' is the number of results per second, WL is the number of bits upon which the logic operation occurs, and 64 is a factor to normalize to a 64 bit operation.