

**Note Z:**

In simple logic operations, a single instruction performs a single logic manipulation of no more than two operands of given lengths.

In complex logic operations, a single instruction performs multiple logic manipulations to produce one or more results from two or more operands.

Rates should be calculated for all supported operand lengths considering both pipelined operations (if supported), and non-pipelined operations using the fastest executing instruction for each operand length based on:

1. Pipelined or register-to-register operations. Exclude extraordinarily short execution times generated for operations on a predetermined operand or operands (for example, multiplication by 0 or 1.. If no register-to-register operations are implemented, continue with 2;
2. The faster of register-to-memory or memory-to-register operations; if these also do not exist, then continue with 3;
3. Memory-to-memory.

In each case above, use the shortest execution time certified by the manufacturer.

**Step 2: TP for each supported operand length WL:**

Adjust the effective rate R (or R') by the word length adjustment L as follows:

$$TP = R * L$$

where  $L = (1/3 + WL/96)$

**Note:**

The word length WL used in these calculations is the operand length in bits. (If an operation uses operands of different lengths, select the largest word length.)

The combination of a mantissa ALU and an exponent ALU of a floating point processor or unit is considered to be one "CE" with a Word Length (WL) equal to the number of bits in the data representation (typically 32 or 64) for purposes of the "CTP" calculation.

This adjustment is not applied to specialised logic processors which do not use XOR instructions. In this case  $TP = R$ .

Select the maximum resulting value of TP for:

- Each XP-only "CE" ( $R_{xp}$ );
- Each FP-only "CE" ( $R_{fp}$ );
- Each combined FP and XP "CE" (R);
- Each simple logic processor not implementing any of the specified arithmetic operations; **and**
- Each special logic processor not using any of the specified arithmetic or logic operations.

**Step 3: "CTP" for aggregations of "CEs", including CPUs.**

For a CPU with a single "CE",

$$"CTP" = TP$$

(for "CEs" performing both fixed and floating point operations  
 $TP = \max (TP_{fp}, TP_{xp})$ )

"CTP" for aggregations of multiple "CEs" operating simultaneously is calculated as follows:

**Notes:**

1. For aggregations that do not allow all of the "CEs" to run simultaneously, the possible combination of "CEs" that provides the largest "CTP" should be used. The TP of each contributing "CE" is to be calculated at its maximum value theoretically possible before the "CTP" of the combination is derived.

**N.B.:**

To determine the possible combinations of simultaneously operating "CEs", generate an instruction sequence that initiates operations in multiple "CEs", beginning with the slowest "CE" (the one needing the largest number of cycles to complete its operation) and ending with the fastest "CE". At each cycle of the sequence, the combination of "CEs" that are in operation during that cycle is a possible combination. The instruction sequence must take into account all hardware and/or architectural constraints on overlapping operations.

2. A single integrated circuit chip or board assembly may contain multiple "CEs".

3. Simultaneous operations are assumed to exist when the computer manufacturer claims concurrent, parallel or simultaneous operation or execution in a manual or brochure for the computer.
4. "CTP" values are not to be aggregated for "CE" combinations (inter)connected by "Local Area Networks", Wide Area Networks, I/O shared connections/devices, I/O controllers and any communication interconnection implemented by software.
5. "CTP" values must be aggregated for multiple "CEs" specially designed to enhance performance by aggregation, operating simultaneously and sharing memory,- or multiple memory/"CE"- combinations operating simultaneously utilizing specially designed hardware. This aggregation does not apply to "electronic assemblies" described by 1041.3.c.

$$"CTP" = TP_1 + C_2 * TP_2 + \dots + C_n * TP_n,$$

where the TPs are ordered by value, with TP 1 being the highest, TP 2 being the second highest, ..., and TP n being the lowest. C i is a coefficient determined by the strength of the interconnection between "CEs", as follows:

For multiple "CEs" operating simultaneously and sharing memory:

$$C_2 = C_3 = C_4 = \dots = C_n = 0.75$$

**Notes:**

1. When the "CTP" calculated by the above method does not exceed 194 Mtops, the following formula may be used to calculate C i:

$$C_i = \frac{0.75}{\sqrt{m}} \quad (i = 2, \dots, n)$$

where m= the number of "CEs" or groups of "CEs" sharing access provided:

- a. The TP i of each "CE" or group of "CEs" does not exceed 30 Mtops;
- b. The "CEs" or groups of "CEs" share access to main memory (excluding cache memory) over a single channel; **and**
- c. Only one "CE" or group of "CEs" can have use of the channel at any given time.

**N.B.:**

This does not apply to items controlled under Category 1030.

2. "CEs" share memory if they access a common segment of solid state memory. This memory may include cache memory, main memory or other internal memory. Peripheral memory devices such as disk drives, tape drives or RAM disks are not included.

For Multiple "CEs" or groups of "CEs" not sharing memory, interconnected by one or more data channels:

$$C_i = 0.75 * k_j \quad (i = 2, \dots, 32) \text{ (see Note below)}$$

$$= 0.60 * k_j \quad (i = 33, \dots, 64)$$

$$= 0.45 * k_j \quad (i = 65, \dots, 256)$$

$$= 0.30 * k_j \quad (i > 256)$$

The value of C i is based on the number of "Ces", not the number of nodes.

where  $k_j = \min (S_j / K_r, 1)$ , and

$K_r =$  normalizing factor of 20 MByte/s.

$S_j =$  sum of the maximum data rates (in units of MByte/s) for all data channels connected to the i th "CE" or group of "CEs" sharing memory.

When calculating a C i for a group of "CEs", the number of the first "CE" in a group determines the proper limit for C j. For example, in an aggregation of groups consisting of 3 "CEs" each, the 22nd group will contain "CE" 64, "CE" 65 and "CE" 66. The proper limit for C i for this group is 0.60.

Aggregation (of "CEs" or groups of "CEs") should be from the fastest-to-slowest; i.e.:

$$TP_1 \geq TP_2 \geq \dots \geq TP_n, \text{ and}$$

in the case of  $TP_j = TP_{j+1}$ , from the largest to smallest; i.e., :  $C_j \geq C_{j+1}$

**Note:**

The k i factor is not to be applied to "CEs" 2 to 12 if the TP i of the "CE" or group of "CEs" is more than 50 Mtops; i.e., C i for "CEs" 2 to 12 is 0.75.