RotateAI Simulator Derivations

Maxence Morel Dierckx

February 26, 2026

1 Disclaimer

The simulator runs x86 binaries and measures x86 instruction counts. All derived metrics use these counts as proxies for ARM Cortex-M33 execution. Important disclaimers:

1. x86 instructions are not the same as ARM instructions. Different Instruction Set Architectures (ISAs) produce different instruction counts for identical computation.
2. x86 vectorises float32s to perform 4-8 floating point operations in a single instruction. The M33 architecture does not implement this "Single Instruction Multiple Data" (SIMD) technology, so simulator results will heavily undercount the FLOPS of an actual tag.
3. All hardware-specific constants (voltage, current draw, DMIPS, maximum frequency) are configurable. Actual values vary with peripheral activity, temperature, and process variation.

The derived values should be interpreted as **fermi estimates** of eventual characteristics. They have utility as estimates of feasibility and for relative comparison of different model pipelines. They should not be used as engineering specifications.

2 Variables

Symbol	Description	Source
N	Instructions per inference	`perf stat`
f_s	Sample rate (Hz)	Config
$f_{\max}$	Maximum operating frequency (MHz)	Config
D	DMIPS per MHz	Config
μ	Current per MHz (µA/MHz)	Config
V	Supply voltage (V)	Config

3 Minimum Operating Frequency

The Minimum Operating Frequency is the minimum CPU frequency for which the required instructions per inference can be completed in the given sample period (e.g., 200 ms at 5 Hz).

- The sample period is the reciprocal of the sample rate f_s.
- The Dhrystone Million Instructions Per Second (DMIPS) per MHz ($s^{-1}MHz^{-1}$) determines the number of instructions our processor can execute per unit time. We multiply by 10^6 to convert to $s^{-1}Hz^{-1}$.

$$f_{\text{min}} = \frac{N \cdot f_s}{D \cdot 10^6} \tag{1}$$

Note that DMIPS is a benchmark-specific metric. Dhrystone is an integer performance benchmark, which means it differs significantly from floating-point heavy ML inference. Using D as a proxy for the general throughput of the processor is erroneous, but it is the best option STM publishes.

4 Energy Per Inference

Energy is the product of voltage, current, and time.

- At a given frequency f, the MCU draws $\mu \cdot f$ microamps of current.
- One inference takes $N/(f \cdot D \cdot 10^6)$ seconds to complete.

$$E = V \cdot (\mu \cdot f) \cdot \frac{N}{f \cdot D \cdot 10^6}$$

The frequency f cancels. Running faster draws more current for less time, and the product is constant within a voltage range. This simplifies to:

$$E = \frac{V \cdot \mu \cdot N}{D \cdot 10^{12}} \tag{2}$$

5 Duty Cycle

The duty cycle is the fraction of time the processor spends running inference versus sleeping. This is not measured from the x86 process—it is estimated from $f_{\min}$ and the target's maximum operating frequency $f_{\max}$. Since $f_{\min}$ is the frequency at which inference exactly fills the sample period ($\delta = 1$), running at any higher frequency proportionally reduces the duty cycle:

$$\delta = \frac{f_{\min}}{f_{\max}} \tag{3}$$

A duty cycle above 1 indicates the inference cannot complete within the sample period at $f_{\max}$.

6 Power Consumption

Average power is the energy consumed per unit time. With f_s inferences per second:

$$P = E \cdot f_s \tag{4}$$