

Power Consumption Application

The goal of this project is to estimate the power consumption of running processes on a computer. This document provides a comprehensive overview of the code base, its components, and the mathematical calculations used for power consumption estimation. The code base uses the standard power formula for the estimation:

$$P = \frac{1}{2}CV^2\alpha + P_{static}$$

where C is the capacitance, V is the voltage, α is the activity factor for both the GPU and CPU, and P_{static} is the static power consumption.

Code Base Overview

The code base consists of three main components: the PowerModel, PowerView, and PowerController. It is based on the Model-View-Controller (MVC) architecture.

PowerModel

The PowerModel is responsible for estimating the power consumption of a running process. It utilizes the `psutil` and `GPUutil` libraries to collect data on the system and running processes and uses this data to estimate power consumption. The main steps involved in estimating the power consumption are as follows:

1. Obtain the clock frequency, number of cores, supply voltage, and temperature of the computer's CPU and GPU.
2. Calculate the capacitor energy using the following formula:

$$E = \frac{1}{2}CV^2$$

3. Calculate the activity factor for the CPU and GPU by monitoring their usage over a certain period of time.
4. Estimate the dynamic power consumption using the capacitor energy and the activity factor for both the CPU and GPU.
5. Estimate the static power consumption using a constant value (k), given a processor and temperature assumptions.

6. Calculate the total power consumption by summing the dynamic and static power consumption.

The calculations are performed differently for Windows and Linux, as Linux has more available options in the `psutil` library to determine constants like voltage and temperature. Ideally, we would want to replace those constants and assumptions with actual values; however, there are no libraries available to get the hardware information. We would require users to input them accordingly.

PowerView

The PowerView is responsible for displaying the running processes and allowing the user to select a process to estimate power consumption. It utilizes the `tkinter` library to create a graphical user interface. The main components of the PowerView are:

1. A search entry to filter the displayed processes.
2. A listbox displaying the running processes with their names and process IDs (PIDs).
3. A scrollbar for easy navigation through the list of processes.
4. A "Calculate" button for initiating the power consumption estimation of the selected process.
5. Labels to display the progress and results of the estimation.

PowerController

The PowerController is responsible for connecting the PowerModel and PowerView. It collects the data on running processes from the PowerModel and passes it to the PowerView for display. It also handles the user interaction with the PowerView, such as selecting a process and initiating the power consumption estimation.

Installation

1. Clone the project repository.
2. Install the required dependencies by running the following command:

```
pip install -r requirements.txt
```

It is recommended to install the dependencies in a virtual environment using `venv` or `conda`.