# A System-level Power-estimation Methodology based on IP-level Modeling, Power-level Adjustment, and Power Accumulation

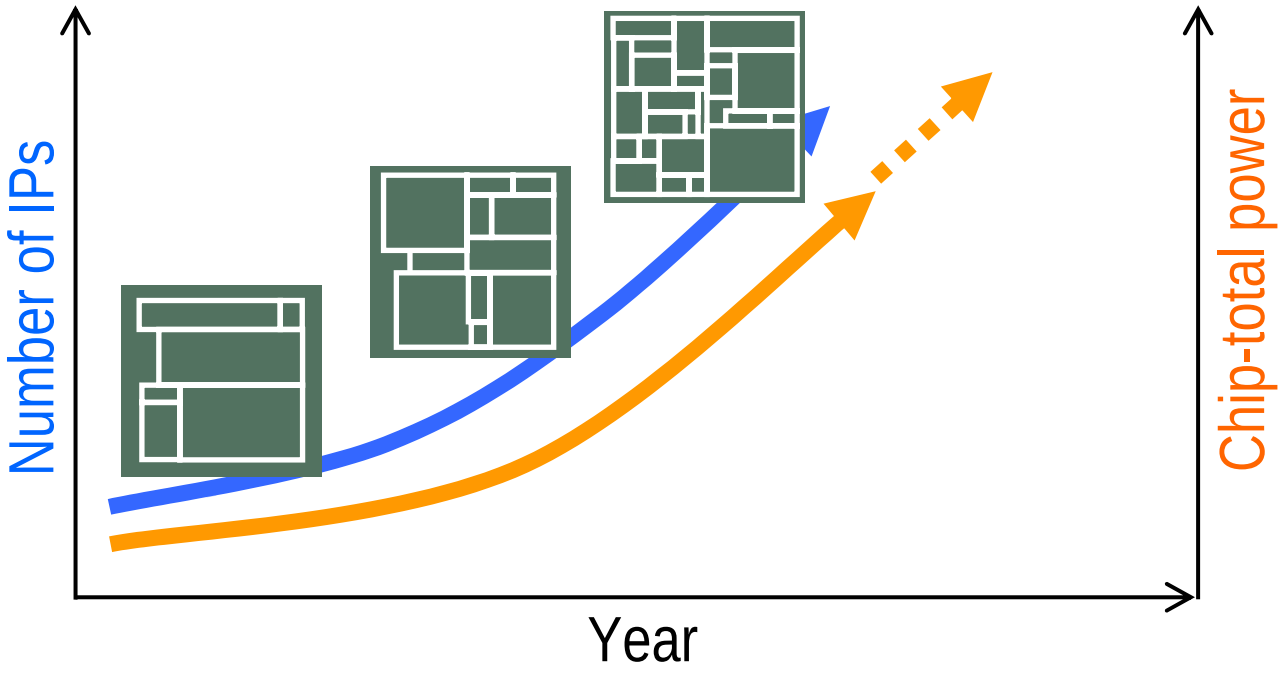
M. Onouchi, T. Yamada, K. Morikawa,[1]
I. Mochizuki,[1] and H. Sekine[1]
Hitachi, Ltd., Tokyo, Japan
[1]Renesas Technology Corp., Tokyo, Japan

# Outline

- Introduction
- System-level power-estimation methodology
  - Effective modeling
  - Precise database
  - Power correlation
- Experimental results
- Summary

# Introduction

- Design and power crises
  - Too many IPs on a single SoC
  - Increasing of chip-total power consumption




SoC: System-on-a-chip, IP: Intellectual properties

# SoC design requirements

- To reduce system-level power consumption, designing low-power SoCs is very important.
  - Selection of suitable IP
  - Revision of IP
  - Judgment of effect of new low-power technology
- Highly accurate system-level power estimation is most necessary in the early design stage.

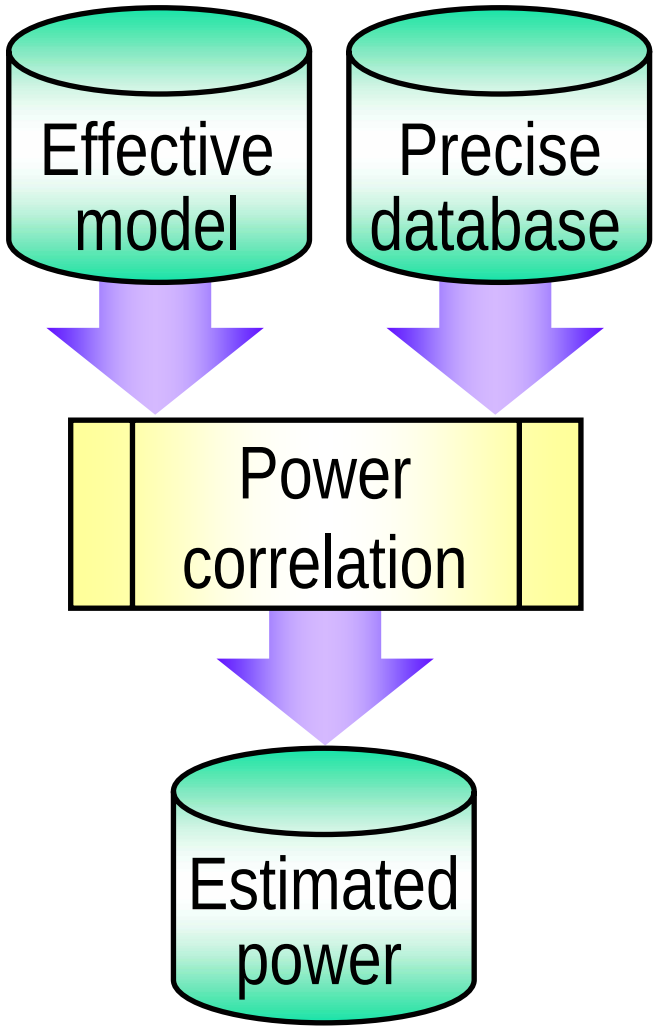
# Issues

- Precise estimation of power consumption in early design stage is difficult.
  - Issue 1:
    - No actual applications in early design stage.
    - Simulating the whole application is impractical.
  - Issue 2:
    - Simple IP-based power estimation leads to overestimation.

# Overview of our solution

- Effective modeling (for issue 1)
  - Abstracting a principle estimation unit from the application
  - Decomposing each IP process
- Precise database (for issue 1)
  - Benchmark selection and simplifying relations between IPs
- Power correlation (for issue 2)
  - Correcting database powers according to the model

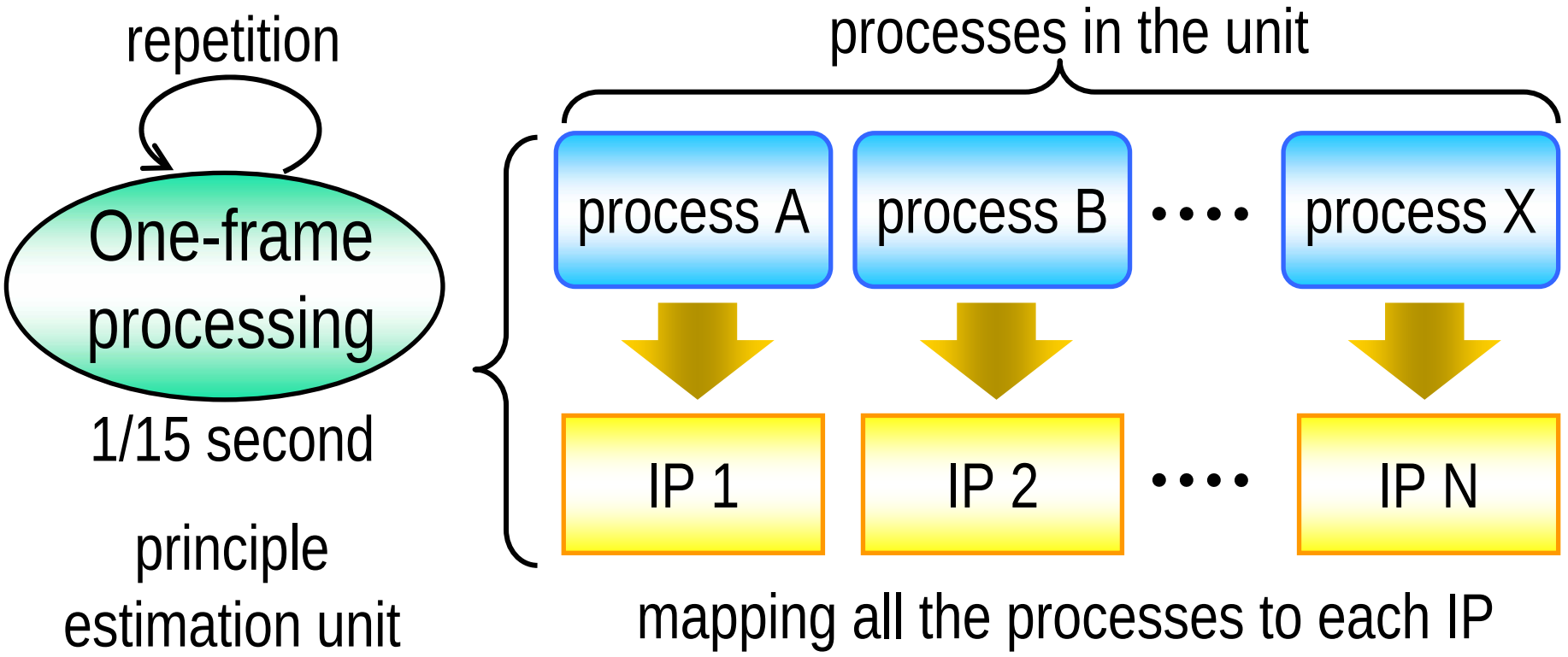
# Step 1: Effective modeling



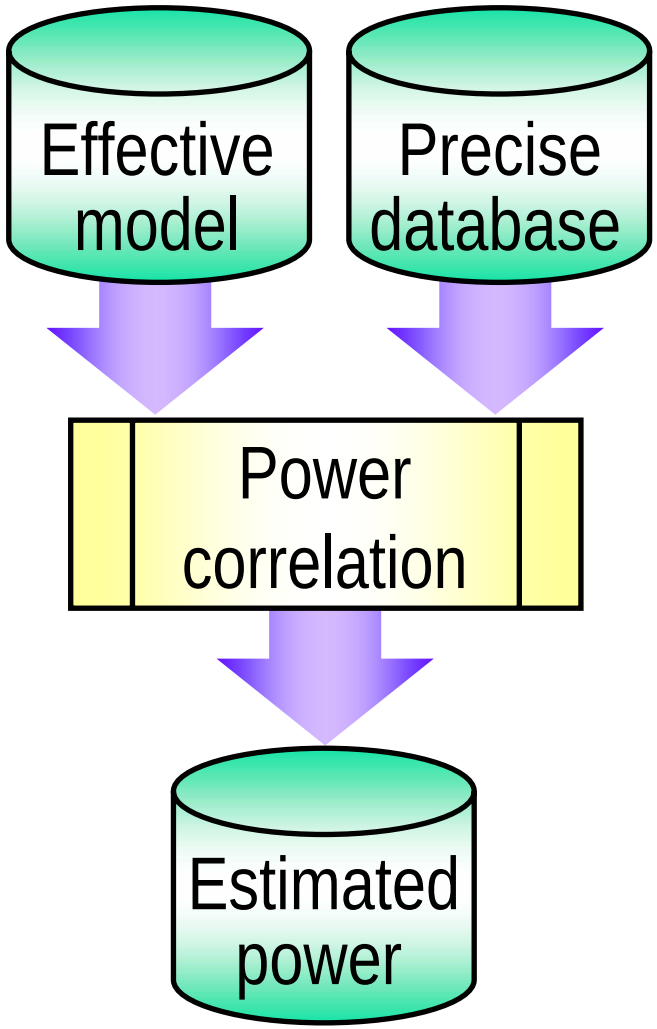

- Find a repetition of the principle estimation unit in application sequences.
- Map the processes in the unit into each IP process.

# Process mapping on each IP

- Find a repetition of the principle estimation unit and its duration time in the blueprint for the application.
- Map all processes in the unit to each IP.



# Step 2: Precise database




- Prepare the precise database with benchmarks; the same processes that are in the effective modeling.
  - Dynamic power
  - Operating cycle
  - ……

# Power calculation methodology

$$P_{Total} = P_{Leakage} + P_{Dynamic}$$

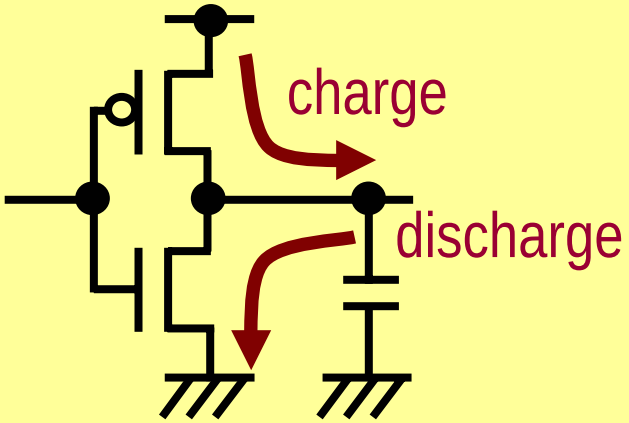
easy to estimate

$$P_{Leakage} \propto \sum_{all} W_i$$

difficult to estimate

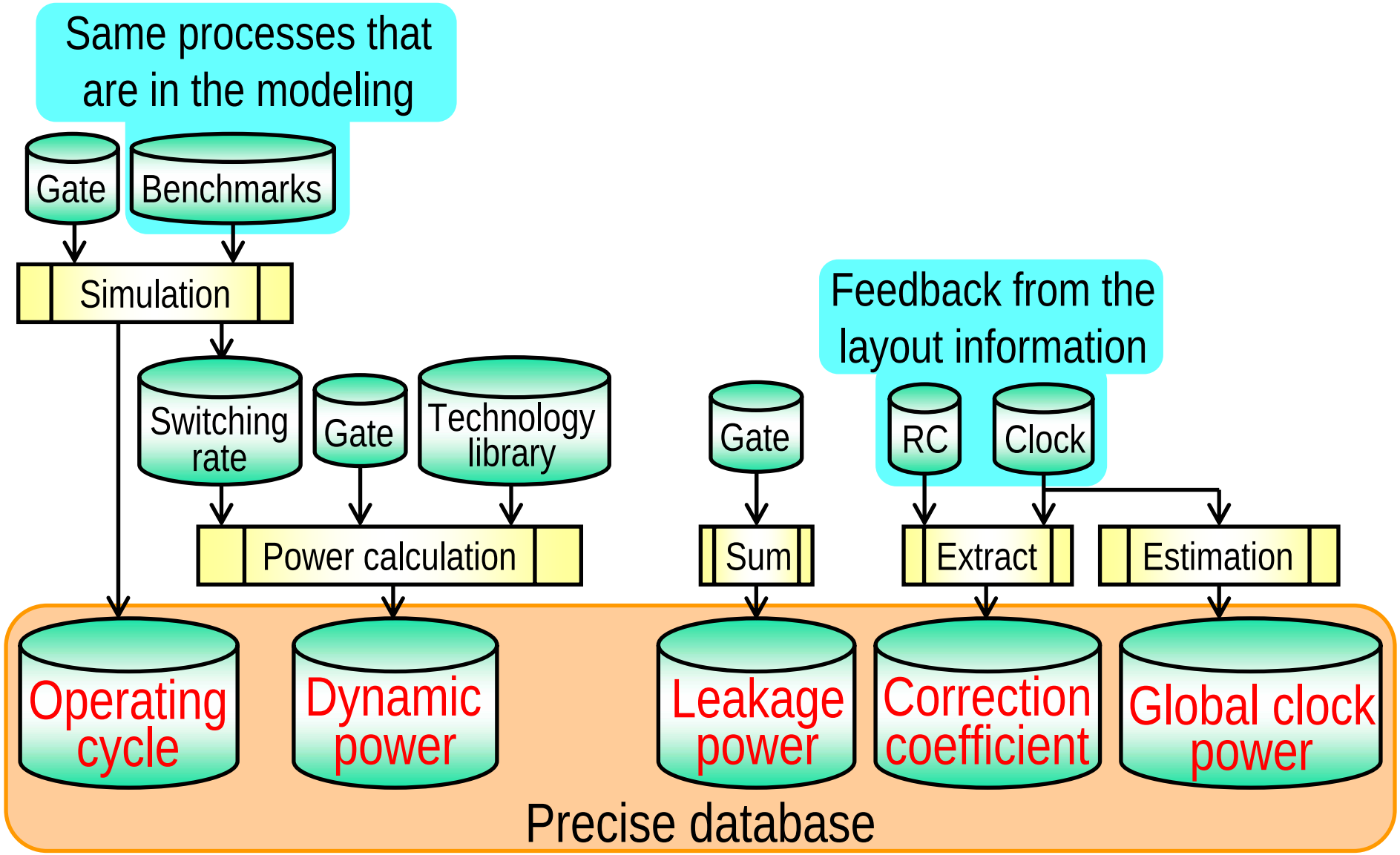
$$P_{Dynamic} = \frac{1}{2} \sum_{all} \alpha_i \cdot C_i \cdot F_i \cdot V_i^2$$

$\alpha$ changes largely due to application



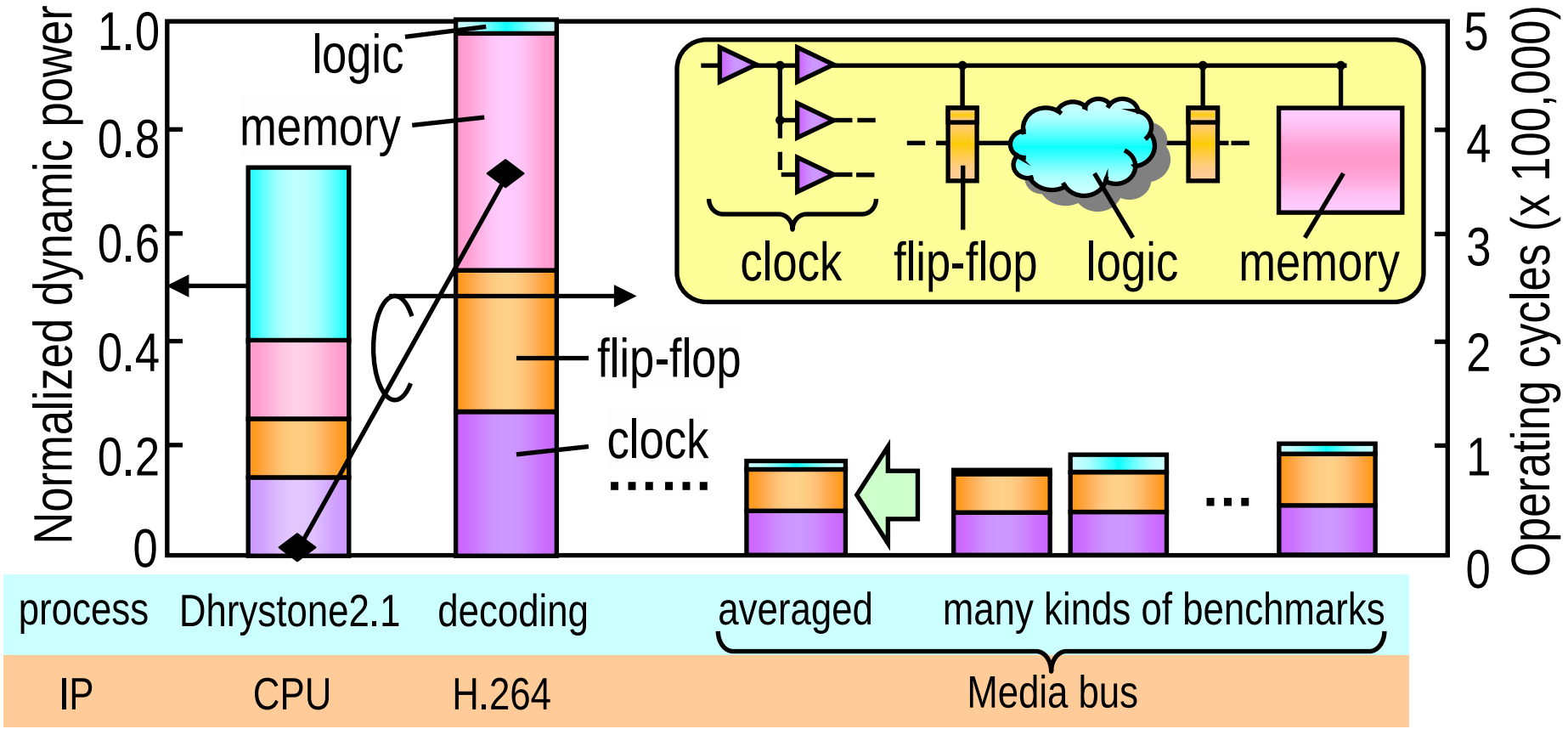

$\alpha$: switching rates  C: capacity  F: frequency  V: voltage

# Simulation environment

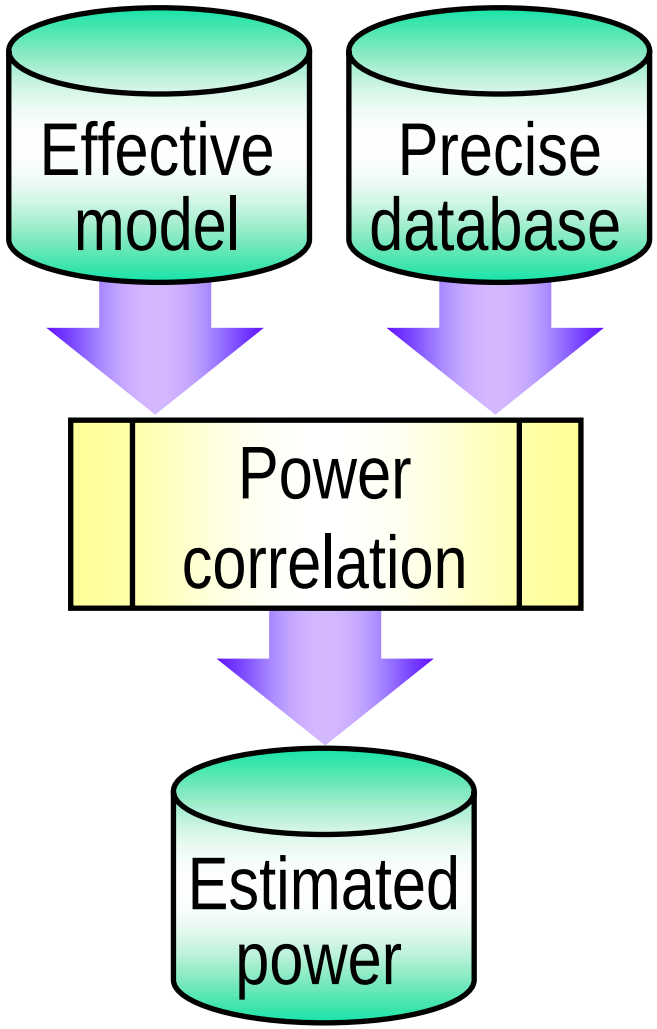


# Database example

- Precise IP operating cycle to avoid power overestimation.
- Media bus IP power rarely depends on benchmarks, so averaged power is applied to bus IP power.



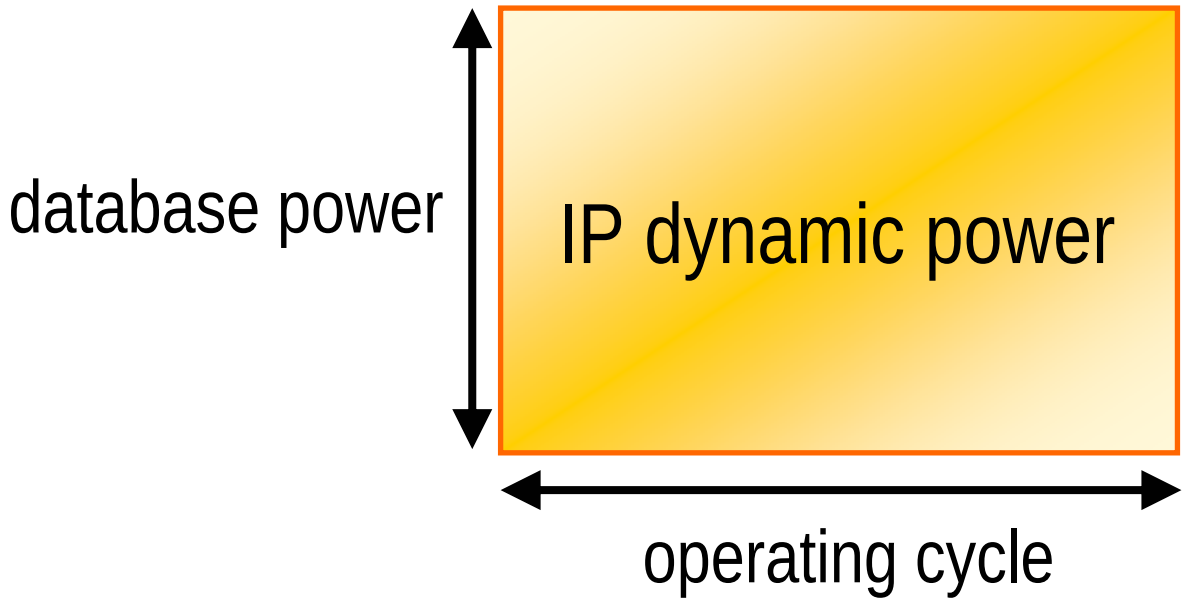
# Step 3: Power correlation




To connect the effective modeling and the database, power correlation methodology is developed.

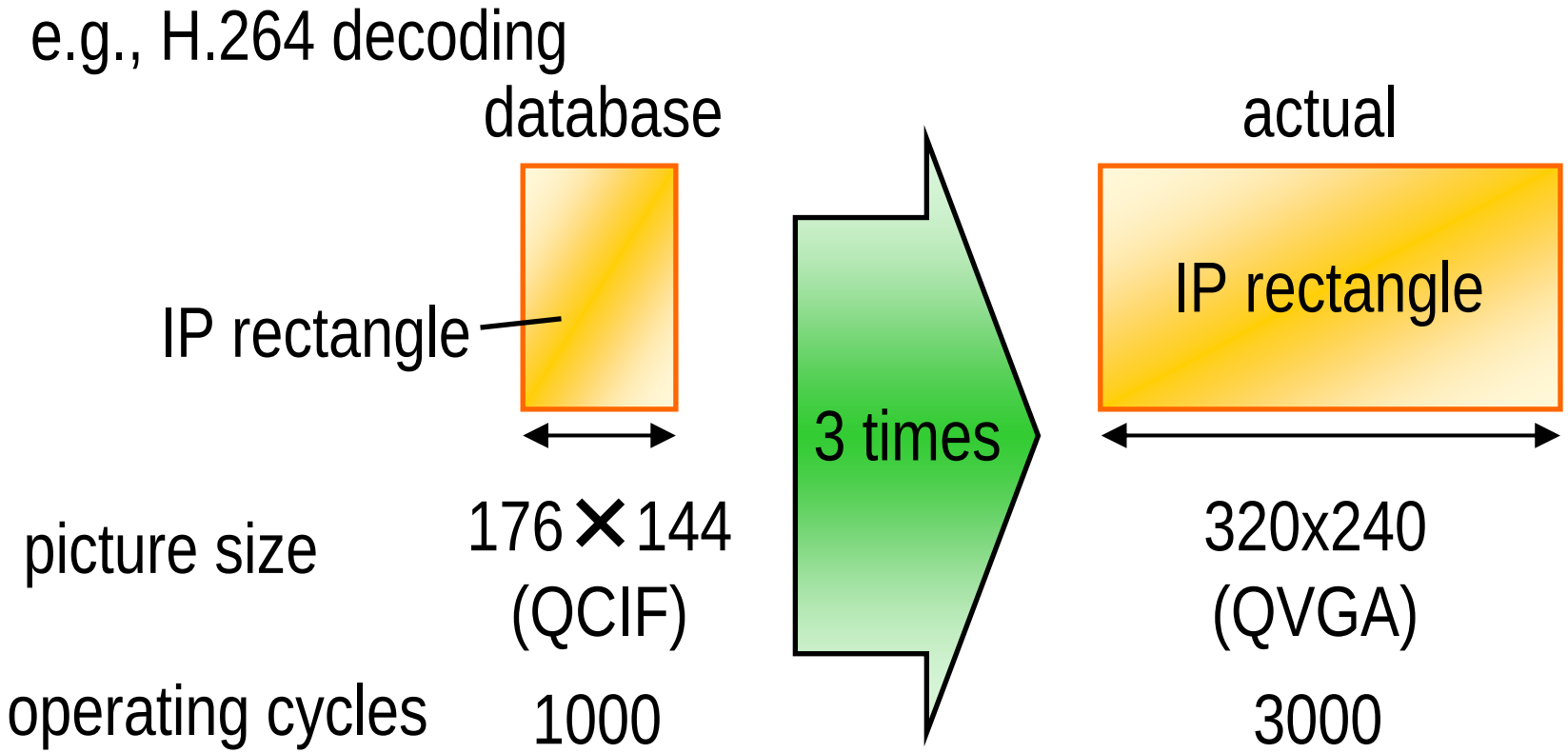
# IP rectangle

- IP rectangle area equals its dynamic power consumption.



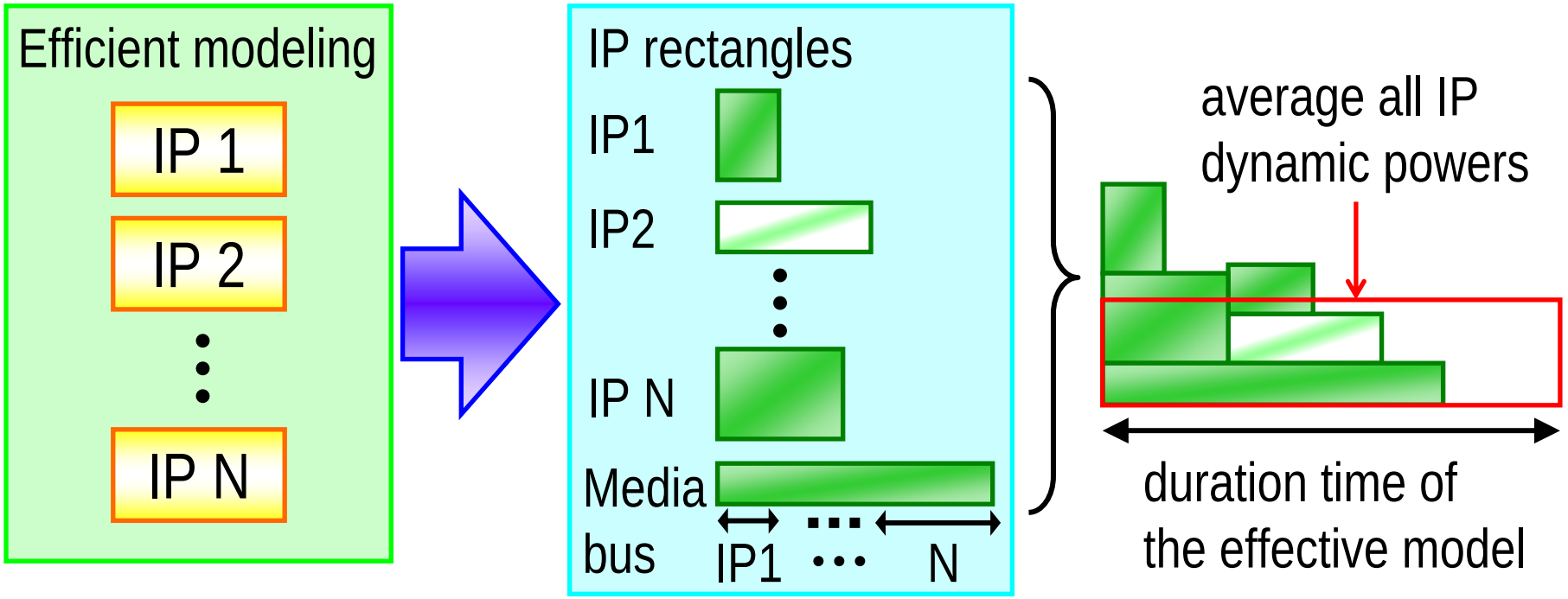
# Power correlation methodology

- To adjust IP's operating cycles, we assume that the operating cycle is linearly scaled according to its load.



# Power accumulation methodology (1)

- Draw all rectangles assuming as follows:
  - IP clock signal ideally stops when not executing processes.
  - Media bus IP cycle = total cycles of all IPs using the media bus.
- Average all IP dynamic powers in duration time.



# Power accumulation methodology (2)

- To obtain system-level power, three powers are added.

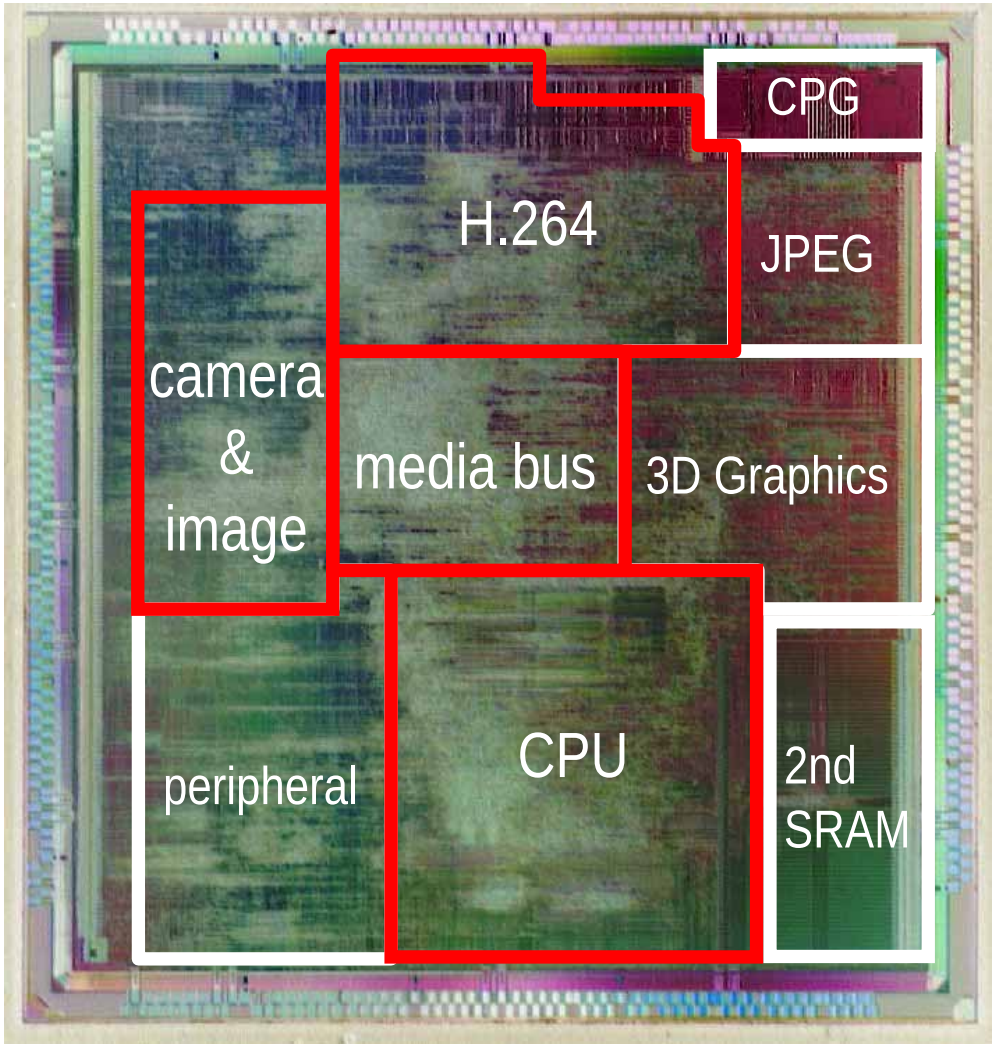
$$P_{System\text{-}level} = P_{Leakage} + P_{Dynamic}$$

$$= \underbrace{P_{Leakage} + P_{Global\_clock}}_{\text{calculated by the database}} + P_{Rectangle}$$

the total of the IP rectangle area

# Target chip and its application

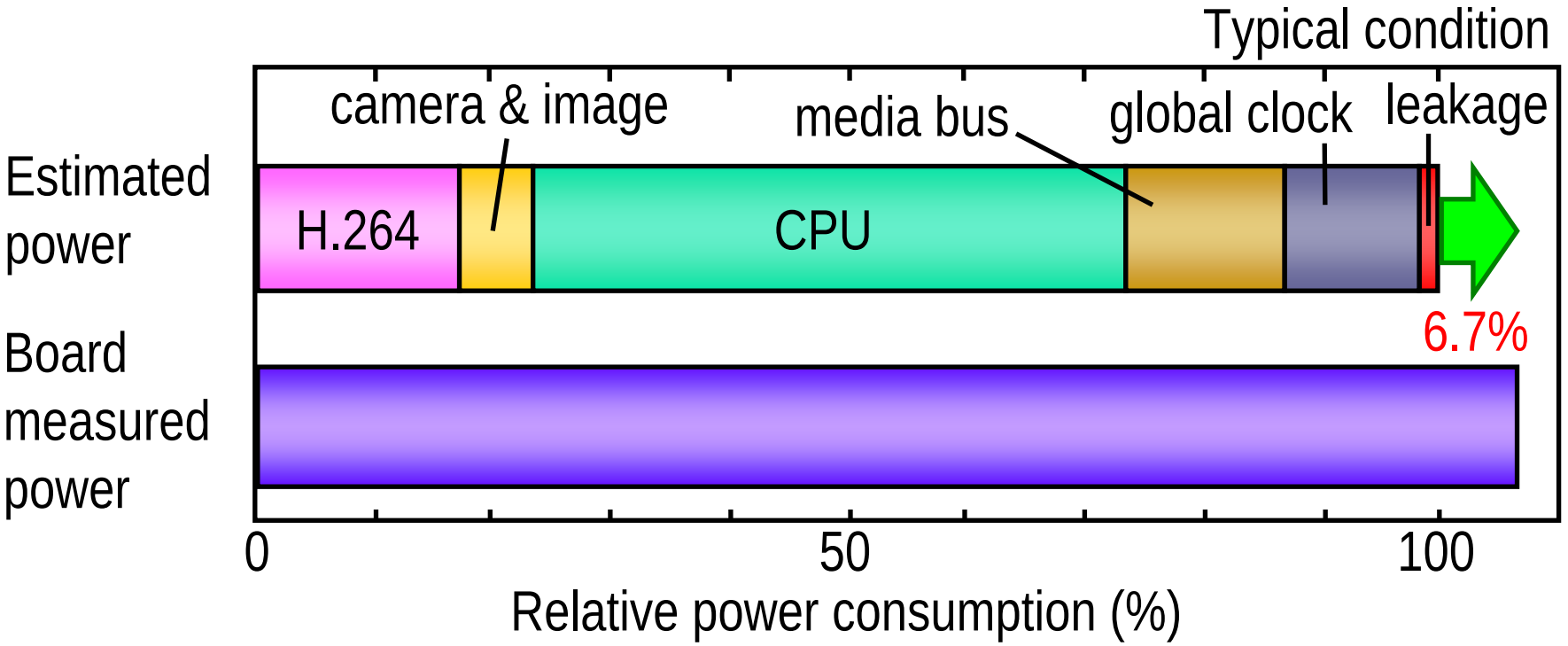
- Application processor designed using 90-nm low-power process.
- Multimedia applications
  - Digital broadcast TV
  - TV phone
- Process-mapped IPs: CPU, H.264, camera & image, media bus.




CPG : Clock Pulse Generator

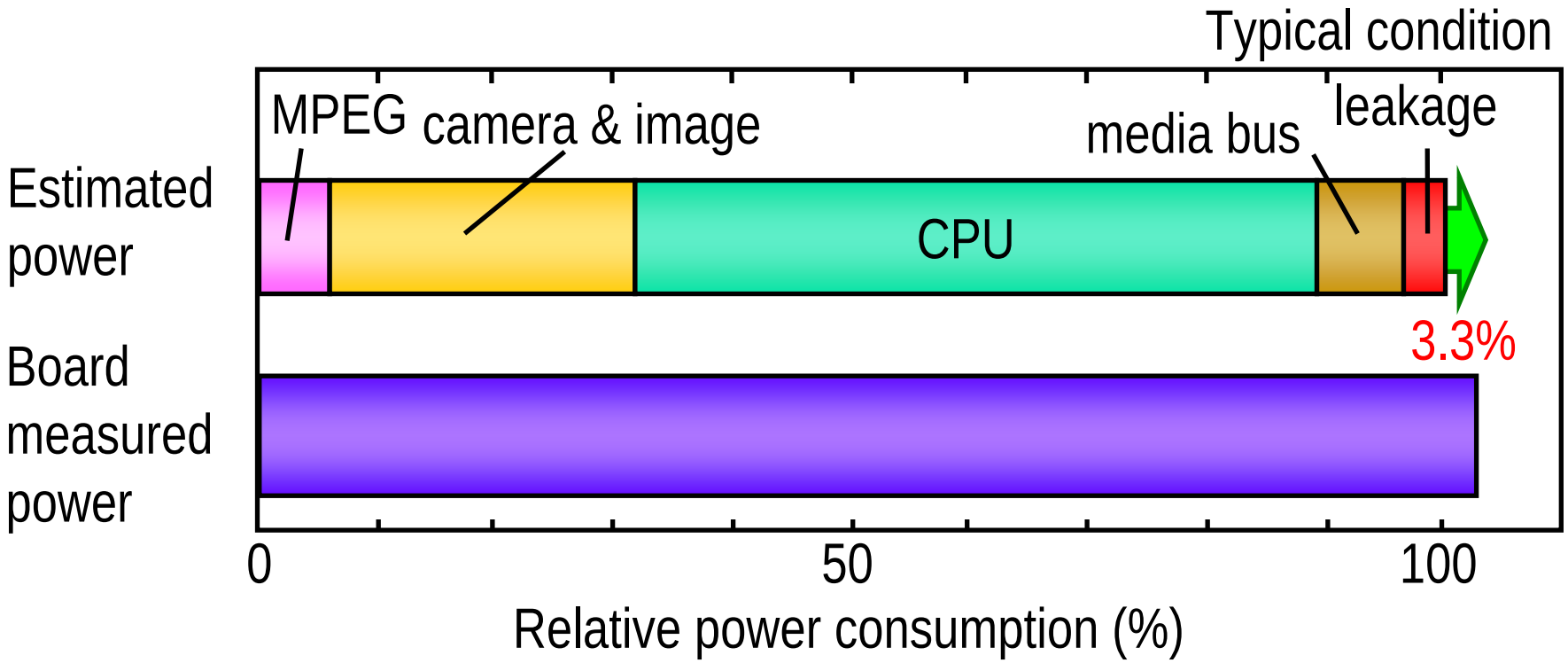
# Experimental results: digital broadcast TV

- Two reasons for errors between two powers:
  - Lack of accuracy in correcting each IP's power
  - Unoptimized software (IP clock signal does not stop even when not executing processes)



# Experimental results: TV phone

- Error was 3.3%; software is optimized in this board measurement.



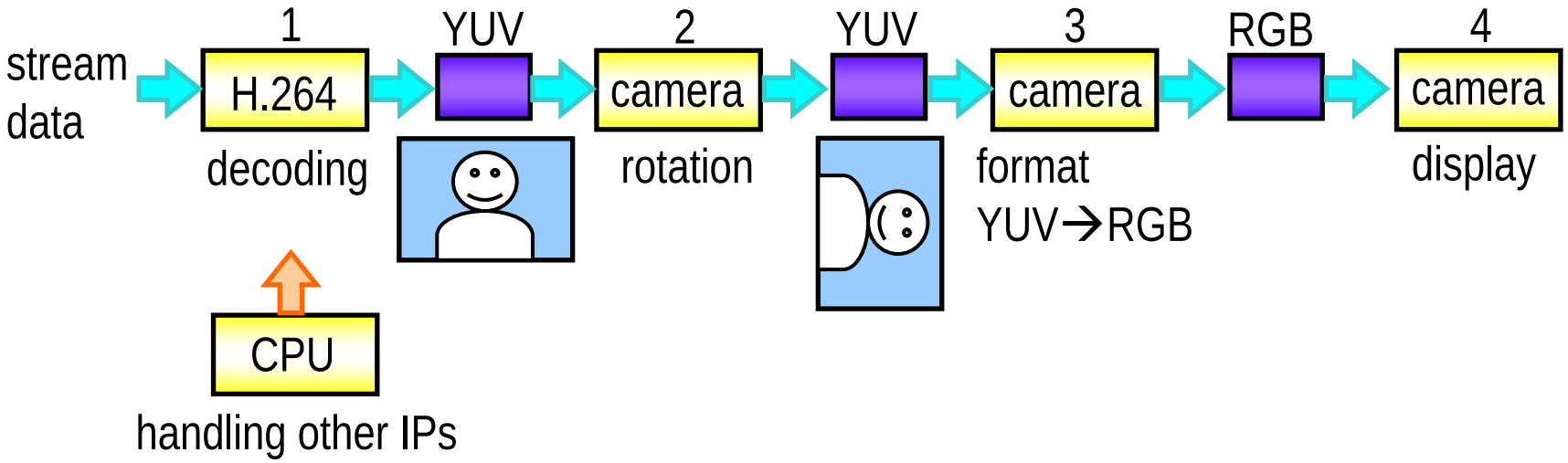
# Summary

- System-level power-estimation methodology
  - Three steps : Effective modeling
    Precise database
    Power correlation
- Errors between estimated power and board-measured power:
  - Digital broadcast TV: 6.7%
  - TV phone: 3.3%

# IP-level model: digital broadcast TV

- Sequential processing of one-frame picture

Image processing




Sound processing