Low Power Design and Synthesis using Multiple Supply Voltage, Variable Frequency and Multicycling

Saraju P. Mohanty
Dept. of CSE, University of South Florida
smohanty@csee.usf.edu

For more details visit research page at:
http://www.csee.usf.edu/~smohanty/

Outline of the Talk

- Introduction
- Related Work
- Power fluctuation minimization through datapath scheduling
- Design of a low power a chip
- Conclusions

Why Low-Power ?

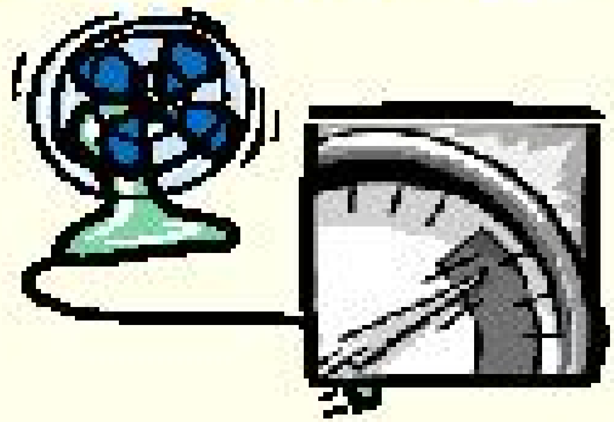
Major Motivation

Extending battery life for portable applications

Why Low-Power ? ....

Battery lifetime

Cooling and energy costs



Environmental concerns

System reliability

Dynamic Power Consumption

Let, C_L = load capacitor, V_{dd} = supply voltage, N = average number of transitions/clock cycle = E(sw) = α and f = clock frequency. The dynamic power consumption for CMOS:

$$P_{dynamic} = \frac{1}{2} C_L V^2_{dd} N f$$

- **Veendrick Observation**: In a well designed circuit, short-circuit power dissipation is less than 20% of the dynamic power dissipation.
- **Sylvester and Kaul**: At larger switching activity the static power is negligible compared to the dynamic power.

We focus on dynamic power reduction !!

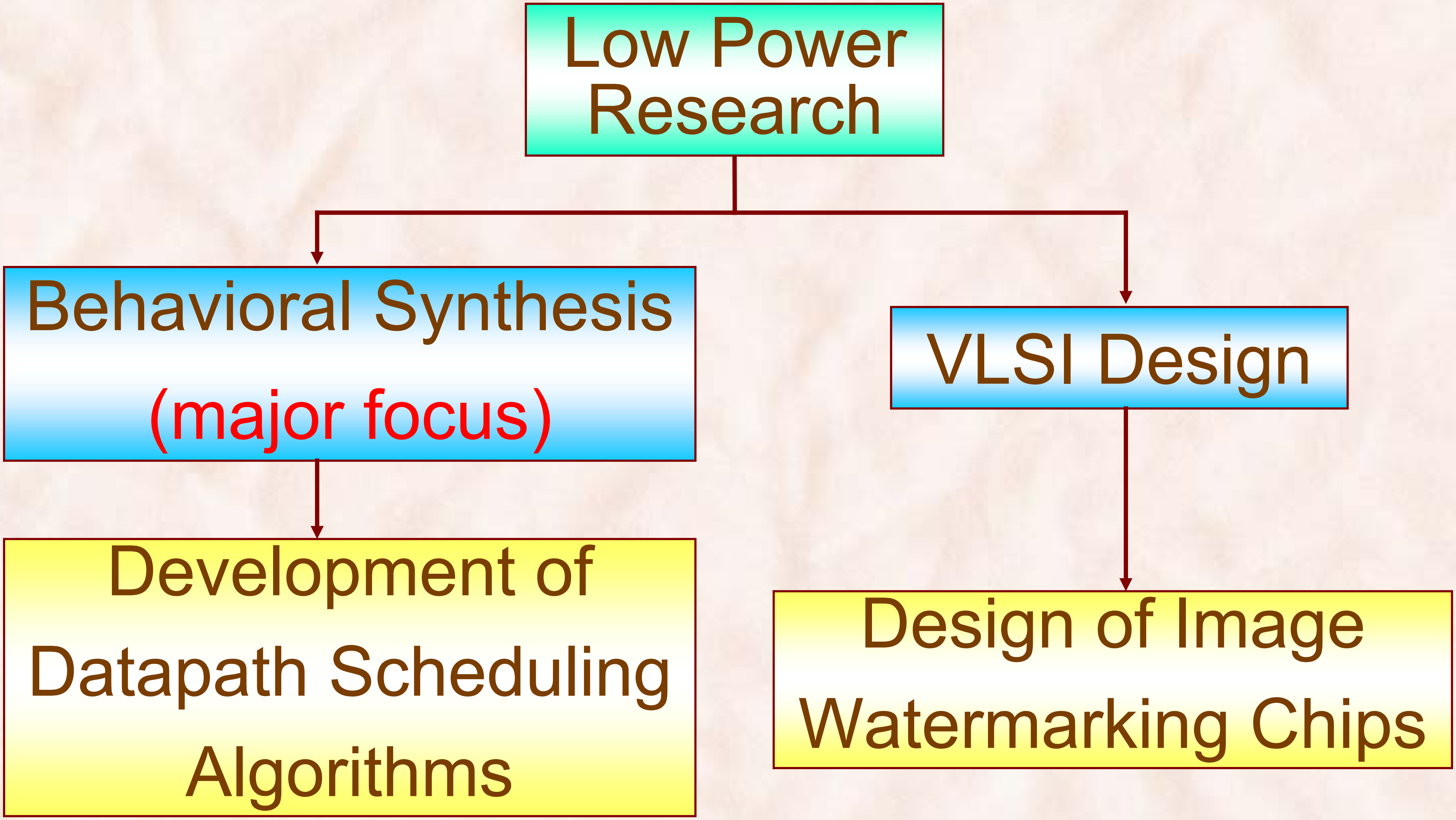
Dynamic Power Reduction ?

- Reduce Supply Voltage (V_{dd}): delay increases; performance degradation
- Reduce Clock Frequency (f): only power saving no energy savings; results in performance degradation
- Reduce Switching Activity (N or E(sw)): no switching no power loss !!! Not fully under designers control. Switching activity depends on the logic function and correlations are difficult to handle.
- Reduce Physical Capacitance: done by reducing device size reduces the current drive of the transistor making the circuit slow

Our approach ?

Adjust the frequency and supply voltage in a co-coordinated manner to reduce various forms of dynamic power while maintaining performance.

Research Overview



Power Fluctuation Minimization during Behavioral Synthesis using ILP-Based Datapath Scheduling

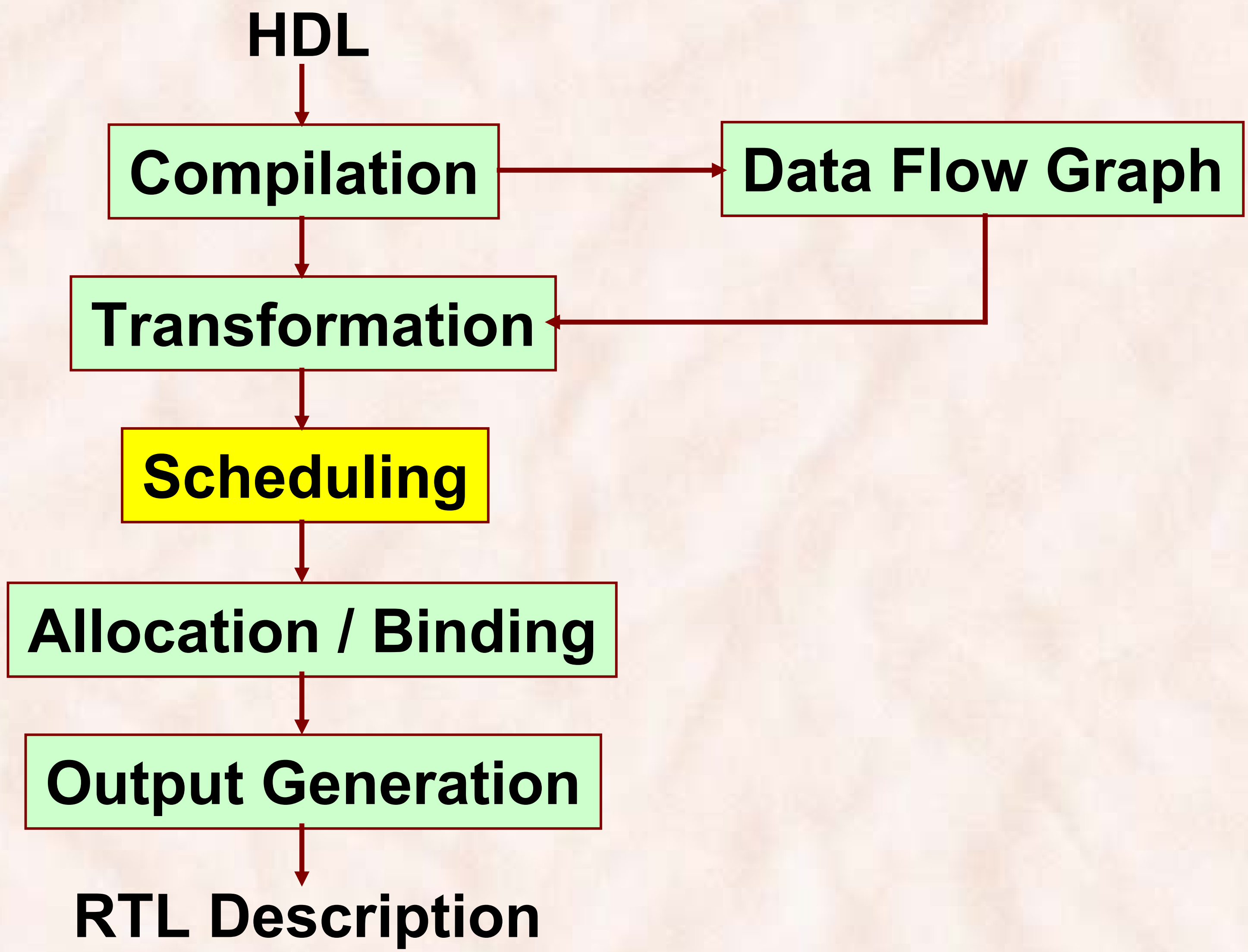
High-Level Synthesis ??

McFarland (1990)

"HLS is conversion from an algorithmic level specification of the behavior of a digital system to a RT level structure that implements that behavior."

NOTE: also known as Behavioral Synthesis.

Phases of Behavioral Synthesis



Datapath Scheduling ?

- Assumption: A datapath is represented as a data flow graph (DFG).
- Scheduling partitions the operations in a DFG into groups so that the operations in a same group can run concurrently.
- Considers the possible trade-offs between the total execution cost and hardware cost.
- Scheduling output:
  - total number of control steps needed to execute all operations
  - minimum number of FUs of each type to be used in the design
  - the lifetimes of the variables generated during the computation

Fluctuation Minimization ??

- Aim ? to minimize the fluctuation in the power consumption profile of the DFG over all the control steps during its execution.
- Two different design options :
  - Multiple voltage with dynamic frequency clocking (MVDFC)
  - multiple supply voltage with multicycling (MVMC)
- Why power fluctuation minimization ?
  - to reduce power supply noise (L di/dt)
  - to reduce cross talk (M di/dt)
  - to increase battery efficiency (electrochemical efficiency)
  - to increase reliability (high current peak during short time)

Related work

Energy efficient scheduling using voltage reduction

- Chang and Pedram 1997 – Dynamic programming
- Johnson and Roy 1997 – ILP based MOVER algorithm using multiple supply voltages
- Lin, Hwang and Wu 1997 – ILP and heuristic for variable voltages (VV) and multicycling (MC)

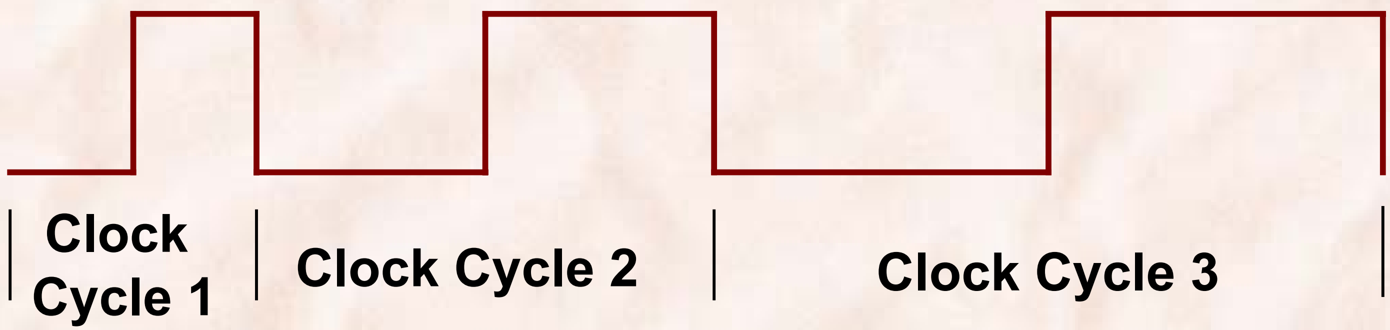
Peak and transient power minimization

Raghunathan, Ravi and Raghunathan 2001 – data monitor operations in VHDL

Dynamic / Variable Frequency?

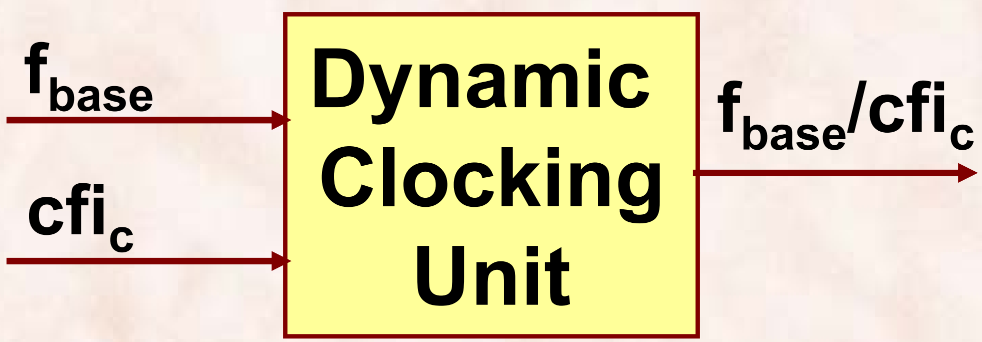
CC1 = CC2 = CC3

Single Frequency



CC1 ≠ CC2 ≠ CC3

Dynamic Frequency

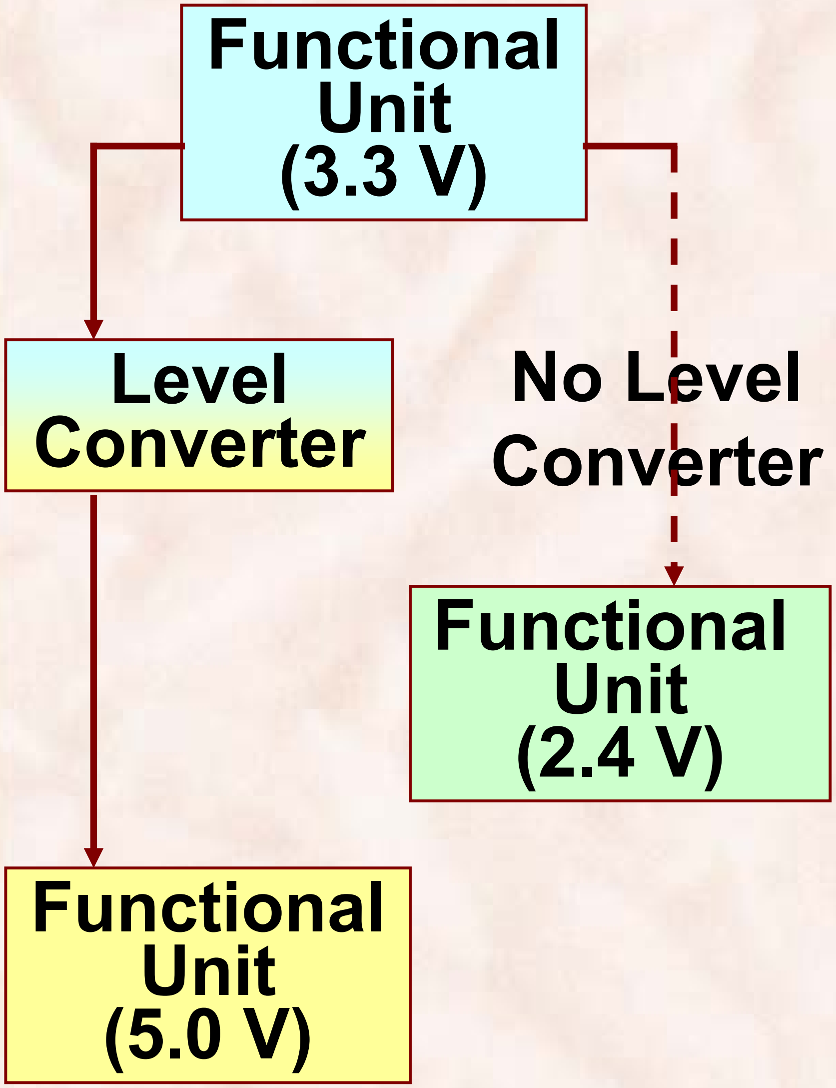


DCU uses clock divider strategy

Design details:

- Ranganathan, et.al.
- Byrnjolfson and Zilic

Target Architecture



(Multiple Supply Voltage)

- Each FU has one register and one MUX and operate at same voltage level as that of FU.
- Operational delay:

  $(d_{FU} + d_{Mux} + d_{Reg} + d_{Conv})$.
- Operating frequencies are calculated from the delays.
- Time for voltage conversion equals to time for frequency change.
- Controller has a storage unit to store cycle frequency indices.

MPG Minimization

- Approach: Use ILP-based datapath scheduling to minimize power fluctuation.
- Overall power fluctuation of the DFG is captured as mean power gradient (MPG).
- MPG is a *non-linear* function due to presence of absolute function, but we use integer linear programming (ILP) for its minimization.

MPG Minimization: Modeling

Background Material

- For a set of n observations, x_1, x_2, x_3, …..x_n, from a given distribution, the sample mean is $m = 1/n\ \Sigma_i\ x_i$.
- The observation-to-observation gradient can be defined as $\Delta x_i = |x_i - x_{i-1}|$.
- The mean gradient of the observations is given by $MG = 1/n\ \Sigma_i\ |x_i - x_{i-1}|$.

MPG Minimization: Modeling …

- Power gradient for a cycle c, PG_c : defined as the absolute difference of a cycle power from previous cycle power. $PG_c = |P_c - P_{c-1}|_{(\text{for any } c = 2 \text{ to } N)}$
- Peak of the power gradients PG_p: Maximum of power gradients of all control steps.

$$PG_p = \max(PG_c) = \max(|P_c - P_{c-1}|)_{(\forall\ c=2 \rightarrow N)}$$

- Mean power gradient MPG: Mean of the power gradients of all control steps.

$$MPG = 1/N-1\ \Sigma_{(\forall\ c=2 \rightarrow N)} PG_c = 1/N-1\ \Sigma_{(\forall\ c=2 \rightarrow N)}\ |P_c - P_{c-1}|$$

NOTE: The complete description is obtained after inserting the parameters, such as, capacitance, switching, voltage, frequency etc.

MPG Minimization: Modeling …

Linear Modeling of Nonlinearity

- General form involving absolute nonlinearity:

  Minimize: $\Sigma_i |y_i|$ (1)

  Subject to: $y_i + \Sigma_j a_{ij} x_j \leq b_i, \forall i$ and $x_j \geq 0 \ \forall j$

- Let y_i be expressed as, $y_i = y^1_i - y^2_i$, difference of two non-negative variables.
- After algebraic manipulations,

  Minimize: $\Sigma_i \ y^1_i + y^2_i$ (2)

  Subject to: $y^1_i - y^2_i + \Sigma_j a_{ij} x_j \leq b_i, \forall i$

  $x_j \geq 0 \ \forall j$ and $y^1_i, y^2_i \geq 0 \ \forall i$

- **Summary**: change difference in objective function to sum and introduce the difference as constraints.

MPG Minimization: ILP Notations

- $M_{k,v}$: max number of functional units of type $F_{k,v}$
- S_i : ASAP time stamp for the operation o_i
- E_i : ALAP time stamp for the operation o_i
- $P(C_{swi},v,f)$: power consumption of $F_{k,v}$ used by o_i
- $x_{i,c,v,f}$: decision variable, which takes the value of 1 if o_i is scheduled in control step c using $F_{k,v}$ and c has frequency f
- $y_{i,v,l,m}$: decision variable which takes the value of 1 if oi is using $F_{k,v}$ and scheduled in control steps $l \rightarrow m$
- $L_{i,v}$: latency in terms of number of clock cycles for operation o_i using $F_{k,v}$

NOTE: C_{swi} is a measure of effective switching capacitance of FU_i.

MPG Minimization: ILP (MVDFC)

(1) Objective Function: Minimize the MPG for the whole DFG over all the control steps.

Minimize: $1/N-1 \; \Sigma_{(\forall\ c=2 \rightarrow N)} |P_c - P_{c-1}|$ (1)

The absolute is replaced with sum and the appropriate constraints.

Minimize: $1/N-1 \; \Sigma_{(\forall\ c=2 \rightarrow N)} P_c + P_{c-1}$ (2)

Subject to: Power gradient constraints

After simplification,

Minimize: $2/N-1 \; \Sigma_{(\forall\ c=2 \rightarrow N-1)} P_c + P_1 + P_N$ (3)

Subject to: Power gradient constraints

Using decision variables,

Minimize: $2/N-1 \; \Sigma_c \Sigma_i \Sigma_v \Sigma_f \; x_{i,c,v,f} \; P(C_{swi}, v, f) + \Sigma_i \Sigma_v \Sigma_f \; x_{i,1,v,f} \; P(C_{swi}, v, f) + \Sigma_i \Sigma_v \Sigma_f \; x_{i,N,v,f} \; P(C_{swi}, v, f)$

Subject to: Power gradient constraints

MPG Minimization: ILP (MVDFC)...

(2) Uniqueness Constraints: ensure that every operation o_i is scheduled to one unique control step and represented as, $\forall i, 1 \le i \le O$, $\Sigma_c \Sigma_v \Sigma_f \, x_{i,c,v,f} = 1$

(3) Precedence Constraints: guarantee that for an operation o_i , all its predecessors are scheduled in an earlier control step and successors are scheduled in an later control step; $\forall i,j$, any $o_i \varepsilon$ $Pred(o_j)$, $\Sigma_v \Sigma_f \Sigma_{\{d=S_i \rightarrow E_i\}} \, d \;\; x_{i,c,v,f} - \Sigma_v \Sigma_f \Sigma_{\{d=S_j \rightarrow E_j\}} \, e \;\; x_{j,c,v,f} \le -1$

(4) Resource Constraints: make sure that no control step contains more than $F_{k,v}$ operations of type k operating at voltage v and are enforced as, $\forall c, 1 \le c \le N$ and $\forall v$, $\Sigma_{\{i \varepsilon F_{k,v}\}} \Sigma_f \, x_{i,c,v,f} \le M_{k,v}$

MPG Minimization: ILP (MVDFC)...

(5) Frequency Constraints: lower operating voltage functional unit can not be scheduled in a higher frequency control step; these constraints are expressed as, $\forall i$, $1 \le i \le O$, $\forall c$, $1 \le c \le N$, if $f < v$, then $x_{i,c,v,f} = 0$.

(6) Power Gradient Constraints : to eliminate the non-linearity introduced due to the absolute function introduced as, $\forall$ c, $2 \le c \le N$,

$$\Sigma_i \Sigma_v \Sigma_f \, x_{i,c,v,f} \, P(C_{swi}, v, f) - \Sigma_i \Sigma_v \Sigma_f \, x_{i,c-1,v,f} P(C_{swi}, v, f) \le PG_p$$

NOTE: The unknown PG_p is added to the objective function and minimized alongwith it.

MPG Minimization: ILP (MVMC)

- We followed similar steps as in the MVDFC case using the new decision variable $y_{i,v,l,m}$.
- No frequency constraints involved in MVMC.
- The following items are formulated:
  - (1) Objective Function
  - (2) Uniqueness Constraints
  - (3) Precedence Constraints
  - (4) Resource Constraints
  - (5) Power Gradient Constraints
- Calculations of subscripts for decision variables and limits of summations are more involved compared to MVDFC case due to the additional parameter $L_{i,v}$.

MPG Minimization: Scheduling

Step 1: Construct a look up table for (effective switching capacitance, average switching activity) pairs.

Step 2: Find ASAP and ALAP schedule for UDFG.

Step 3: Get the mobility graph.

Step 4: Use AMPL for ILP formulations of DFG.

Step 5: Solve the ILP formulations using LP-Solve.

Step 6: Find the scheduled DFG.

Step 7: Determine the cycle frequency indices and base frequency for MVDFC scheme.

Step 8: Estimate power consumptions of the scheduled DFG.

MPG Minimization: Results

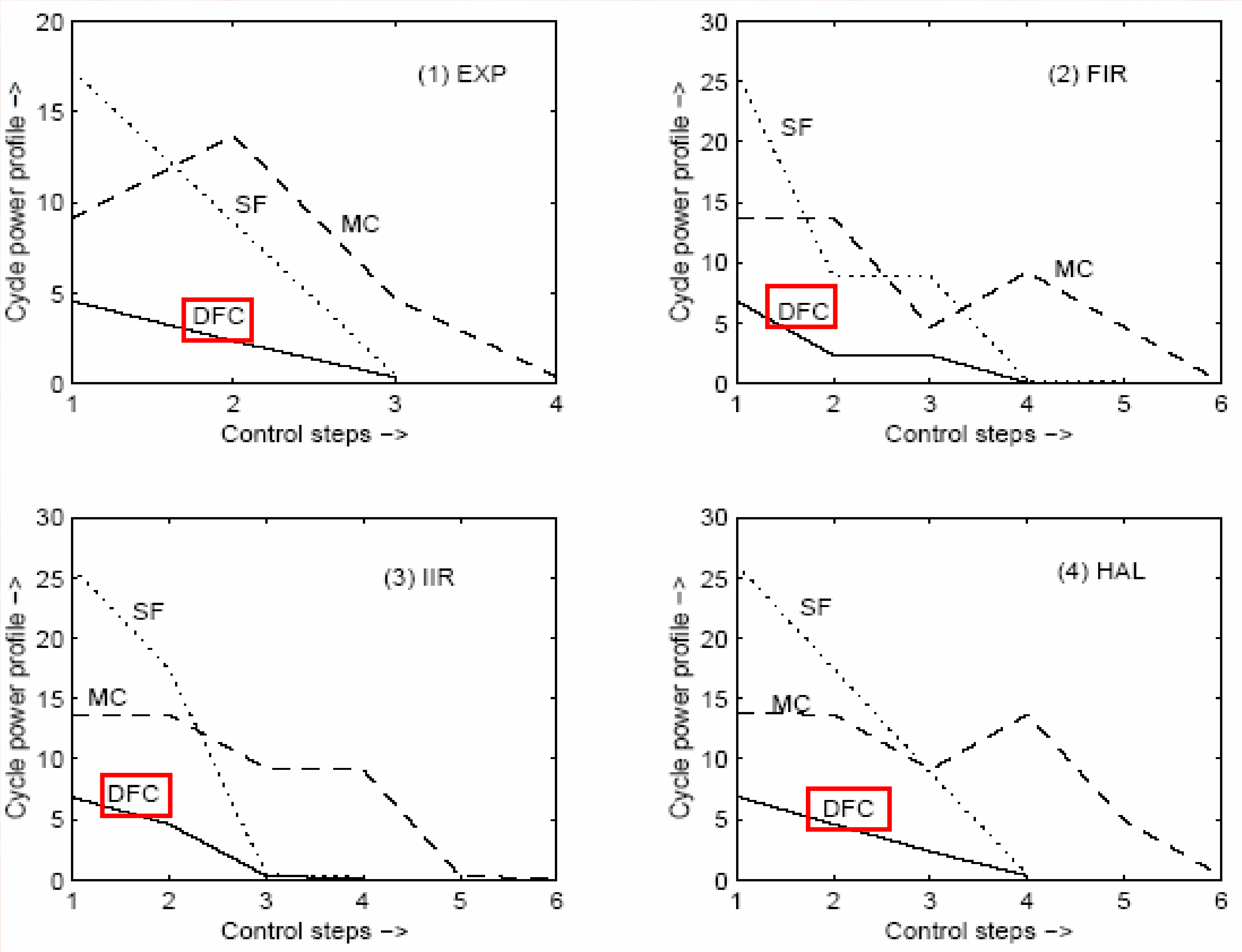
1. Example circuit (EXP)
2. FIR filter
3. IIR filter
4. HAL differential eqn. solver
5. Auto-Regressive filter (ARF)

Multipliers		ALUs		Serial
2.4V	3.3V	2.4V	3.3V	
2	1	1	1	RC1
3	0	1	1	RC2
2	0	0	2	RC3
1	1	0	1	RC4

MPG Minimization: Results …

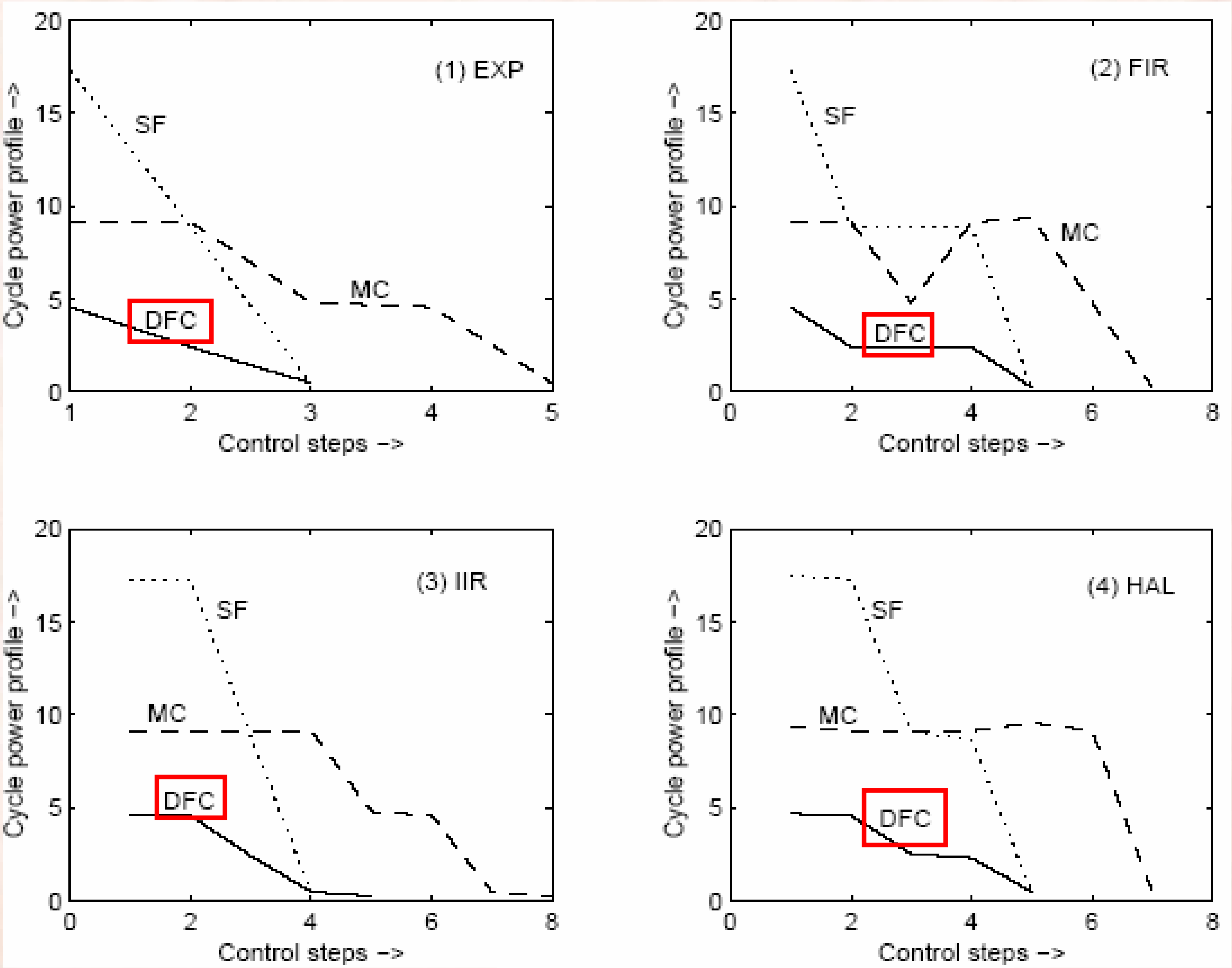
Percentage reduction using MVDFC or MVMC compared to SVSF		
Power	**MVDFC**	**MVMC**
Peak Power	72.50	27.41
MPG	73.42	47.10
Average Power	72.38	24.41
Energy (PDP)	54.13	0.0

MPG Minimization: Results …



Power Profile for RC1

MPG Minimization: Results …

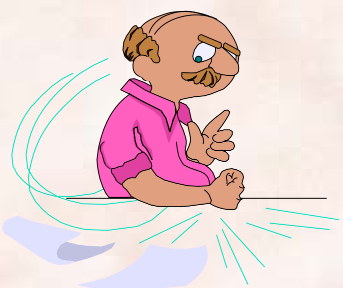


Power Profile for RC2

Dual Voltage Dual Frequency Low Power VLSI Implementation of Image Watermarking Scheme

Digital Watermarking ?

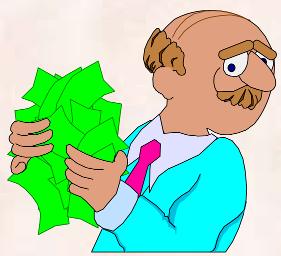
It is mine



Hacker

Multimedia Object

No, It is mine



Owner

Researcher

Whose is it this ? How to know ? What's the solution of this ownership problem?

Solution “ WATERMARKING”

Digital watermarking is a process for embedding data (watermark) into a multimedia object for its copyright protection and authentication.

Types

- Visible and Invisible
- Spatial/DCT/ Wavelet
- Robust and Fragile

An Watermarked Image (from IBM)

Watermarking: General Framework

- **Encoder:** Inserts the watermark into the host image
- **Decoder:** Decodes or extracts the watermark from image
- **Comparator:** Verifies if extracted watermark matches with the inserted one

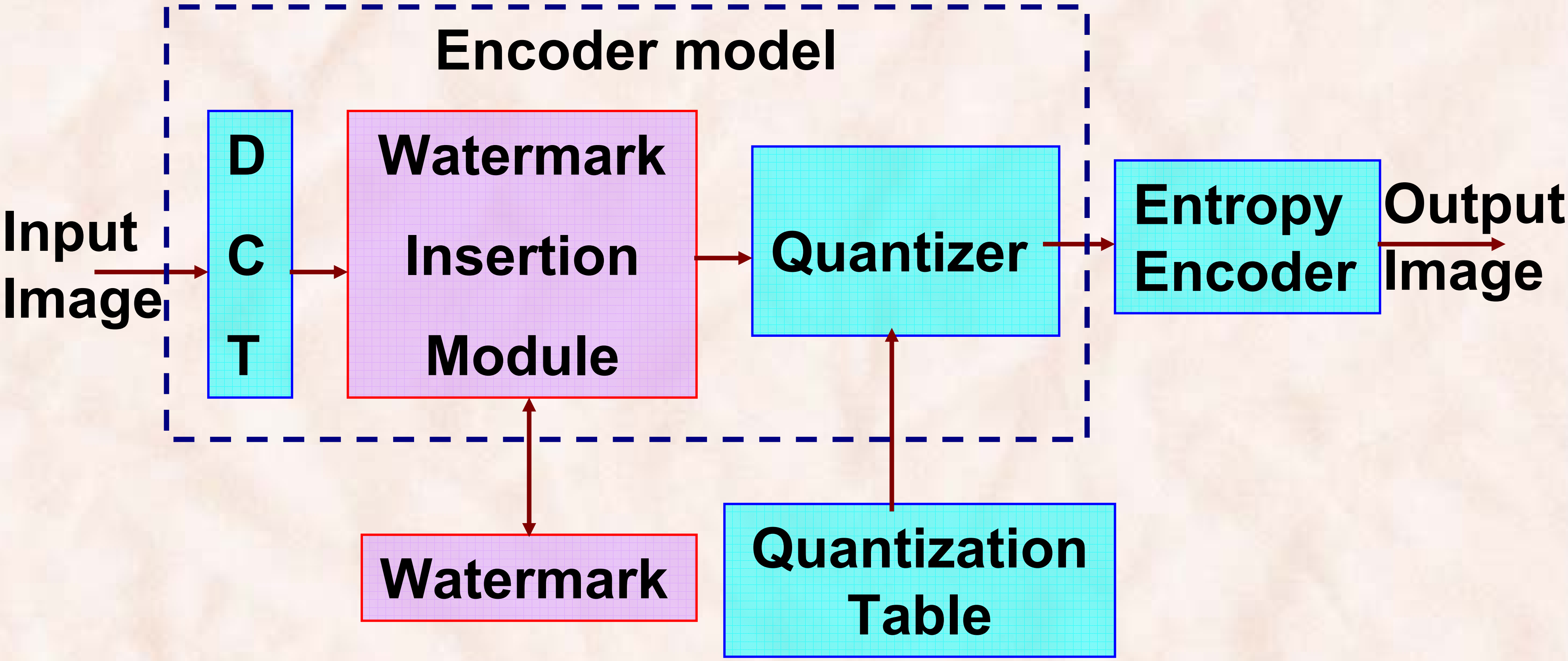
Previous Work (Hardware based Watermarking)

Work	Type	Target Object	Domain	Technology	Chip Power
Strycker, 2000	Invisible Robust	Video	Spatial	NA	NA
Tsai and Lu 2001	Invisible Robust	Video	DCT	0.35µ	62.8 mW
Mathai, 2003	Invisible Robust	Image	Wavelet	0.18µ	NA
Garimella, 2003	Invisible Fragile	Image	Spatial	0.13µ	37.6 µW

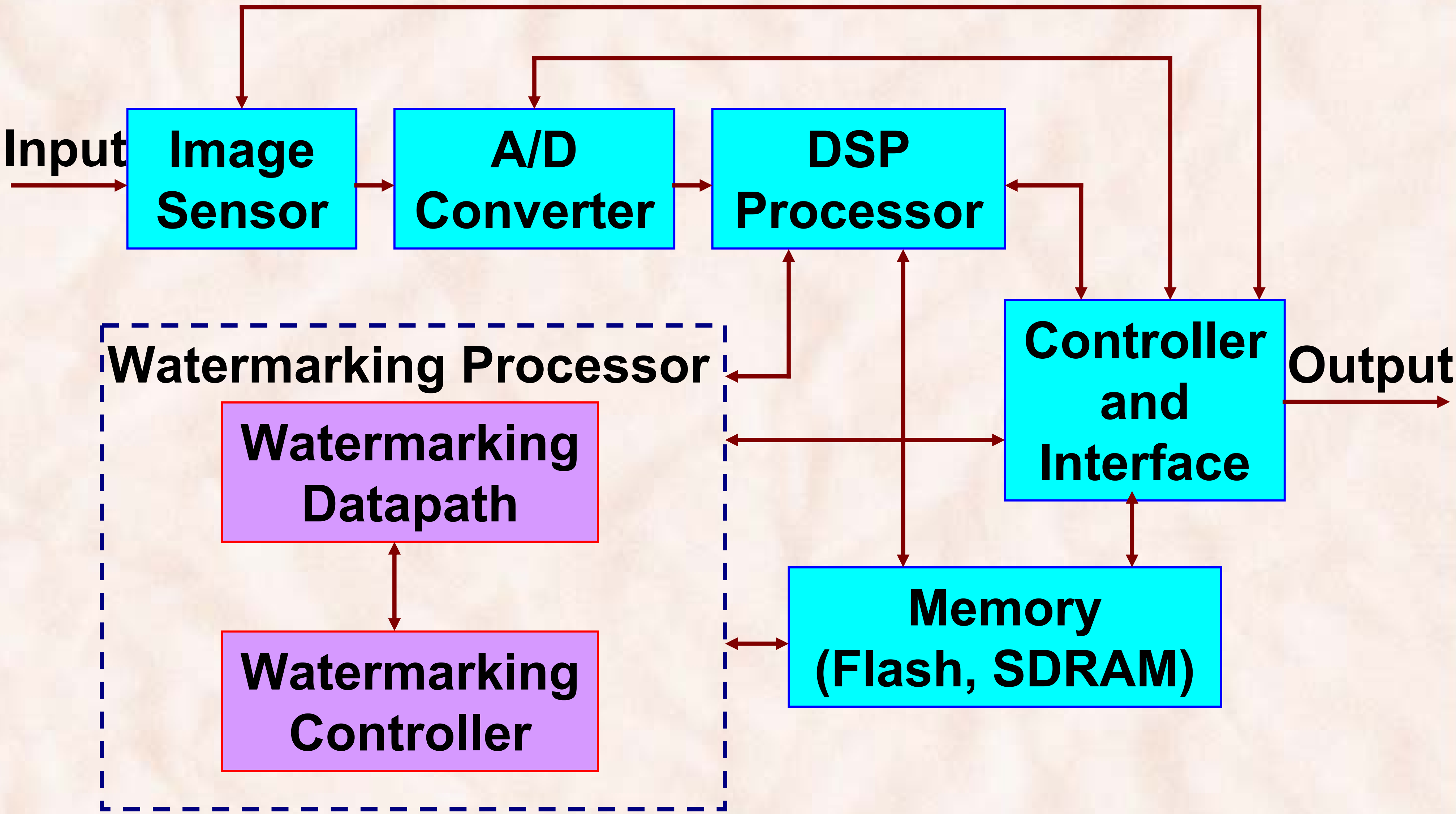
Highlights of our Designed Chip

- DCT domain Implementation
- First to insert both visible and / or invisible watermark
- First Low Power Design for watermarking using dual voltage and dual frequency
- Uses Pipelined / Parallelization for better performance

Watermarking: JPEG Encoder (DCT Domain)



Watermarking: Digital Still Camera



Invisible Algorithm Implemented

1. Divide the original image into blocks.
2. Calculate the DCT coefficients of all the image blocks.
3. Generate random numbers to use as watermark.
4. Consider the three largest AC-DCT coefficients of an image block for watermark insertion.

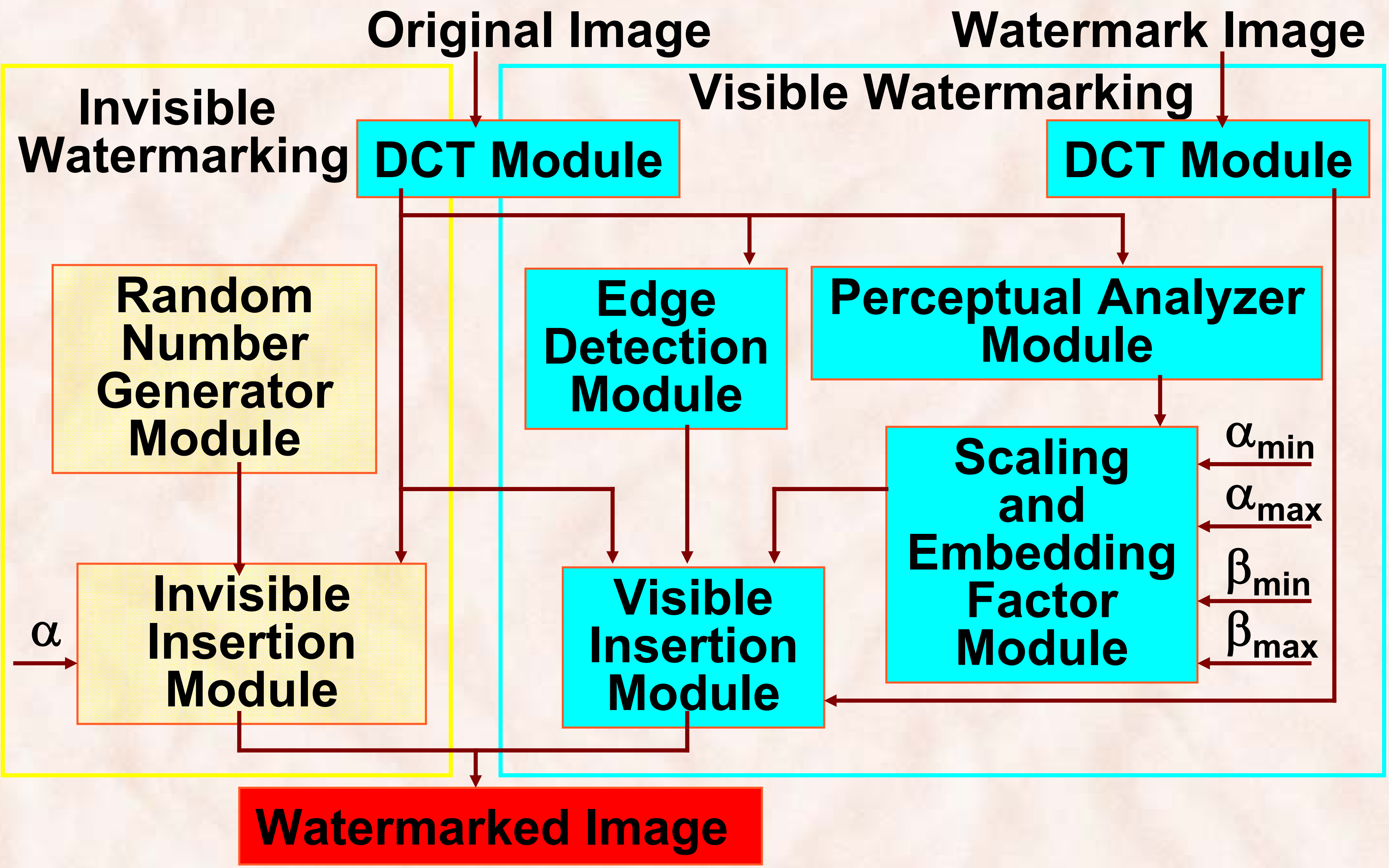
Reference: I.J. Cox, et. al., "Secure Spread Spectrum Watermarking for Multimedia", IEEE transactions on Image Processing, 1997.

Visible Algorithm Implemented

1. Divide Original and watermark image into blocks.
2. Calculate DCT coefficients of all the blocks.
3. Find the edge blocks in the original image.
4. Find the local and global statistics of original image using DC-DCT and AC-DCT coefficients.
5. The mean of DC-DCT coefficients and mean and the variance of AC-DCT coefficients are useful.
6. Calculate the Scaling and embedding factors.
7. Add the original image DCT coefficients and the watermark DCT coefficients block by block.

Reference: S. P. Mohanty, and et. al., "A DCT Domain Visible Watermarking Technique for Images", *Proc. of the IEEE ICME* 2000.

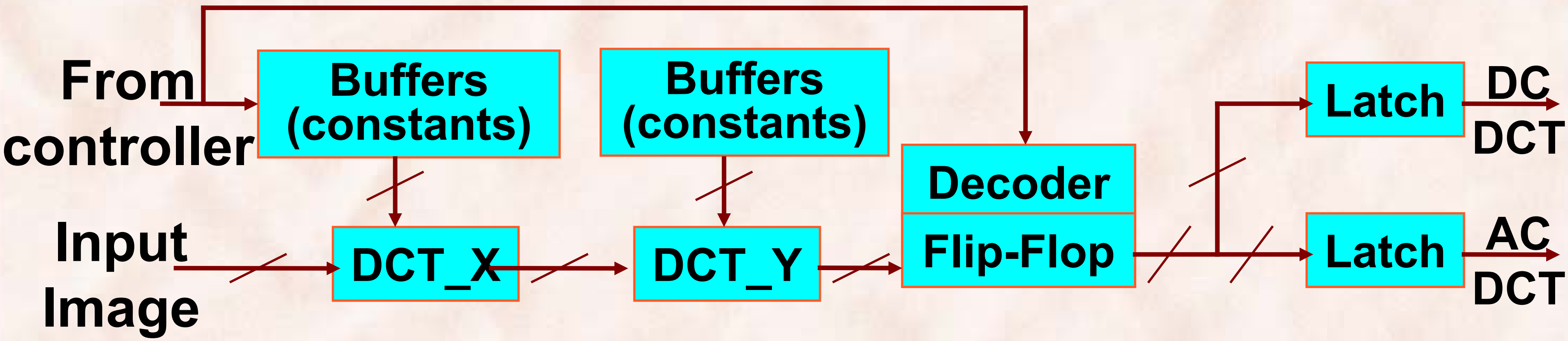
The Proposed Architecture



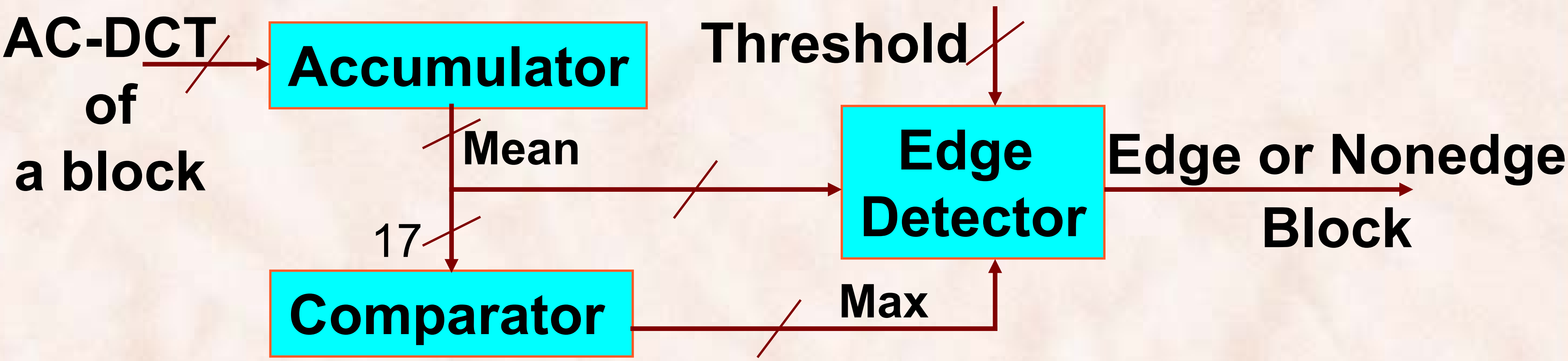
Modules in Proposed Architecture

- DCT Module: Calculates the DCT coefficients.
- Edge Detection Module: Determines edge blocks.
- Perceptual Analyzer Module: Determines perceptually significant regions using original image statistics.
- Scaling and Embedding Factor Module: Determines the scaling and embedding factors for visible watermark insertion.
- Watermark Insertion Module: Inserts the watermark
- Random Number Generator Module: Generates random numbers.

Modules in Proposed Architecture

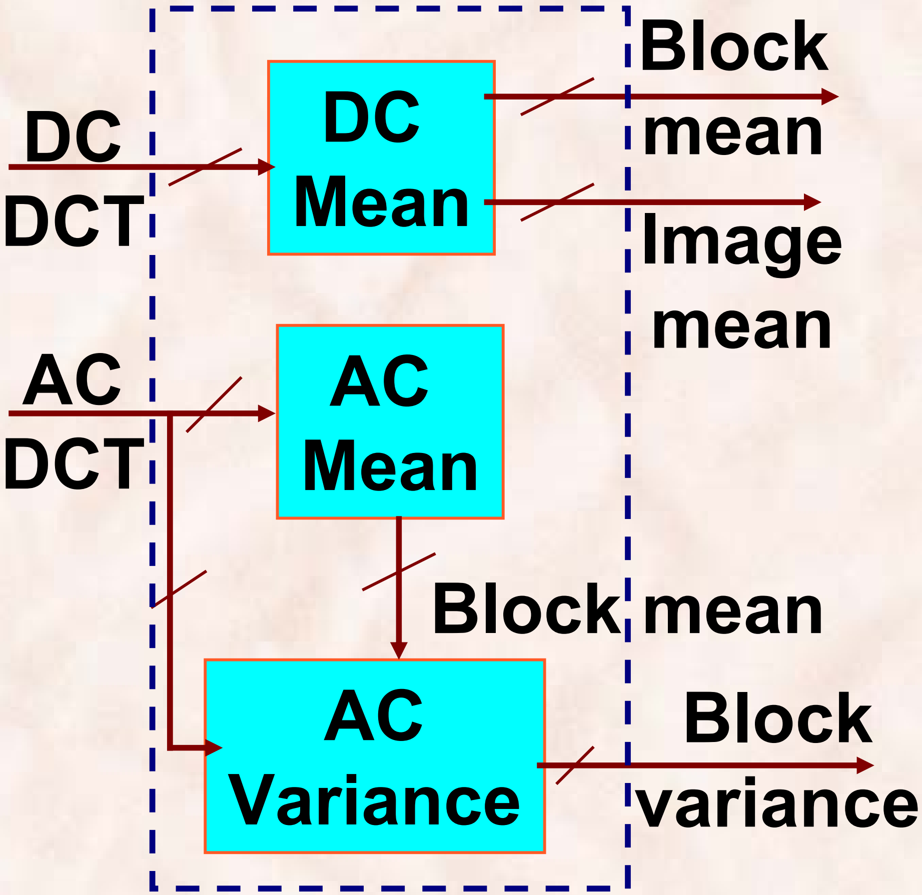


DCT Module

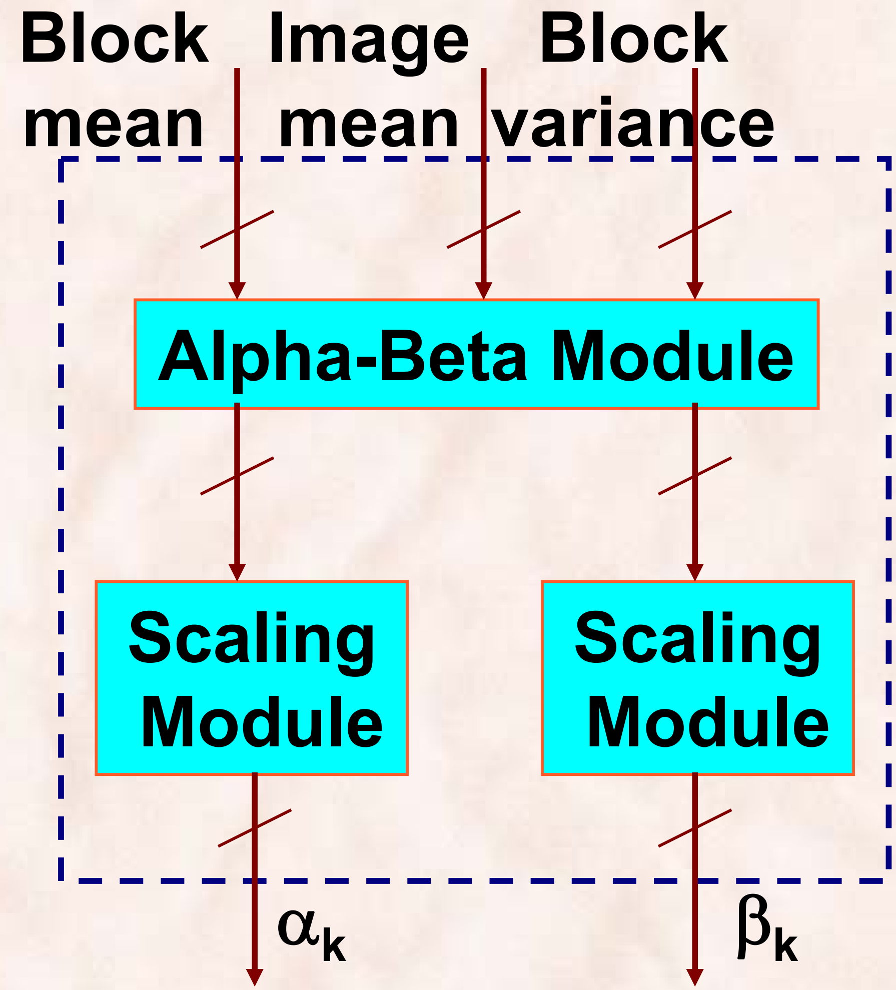


Edge Detection Module

Modules in Proposed Architecture

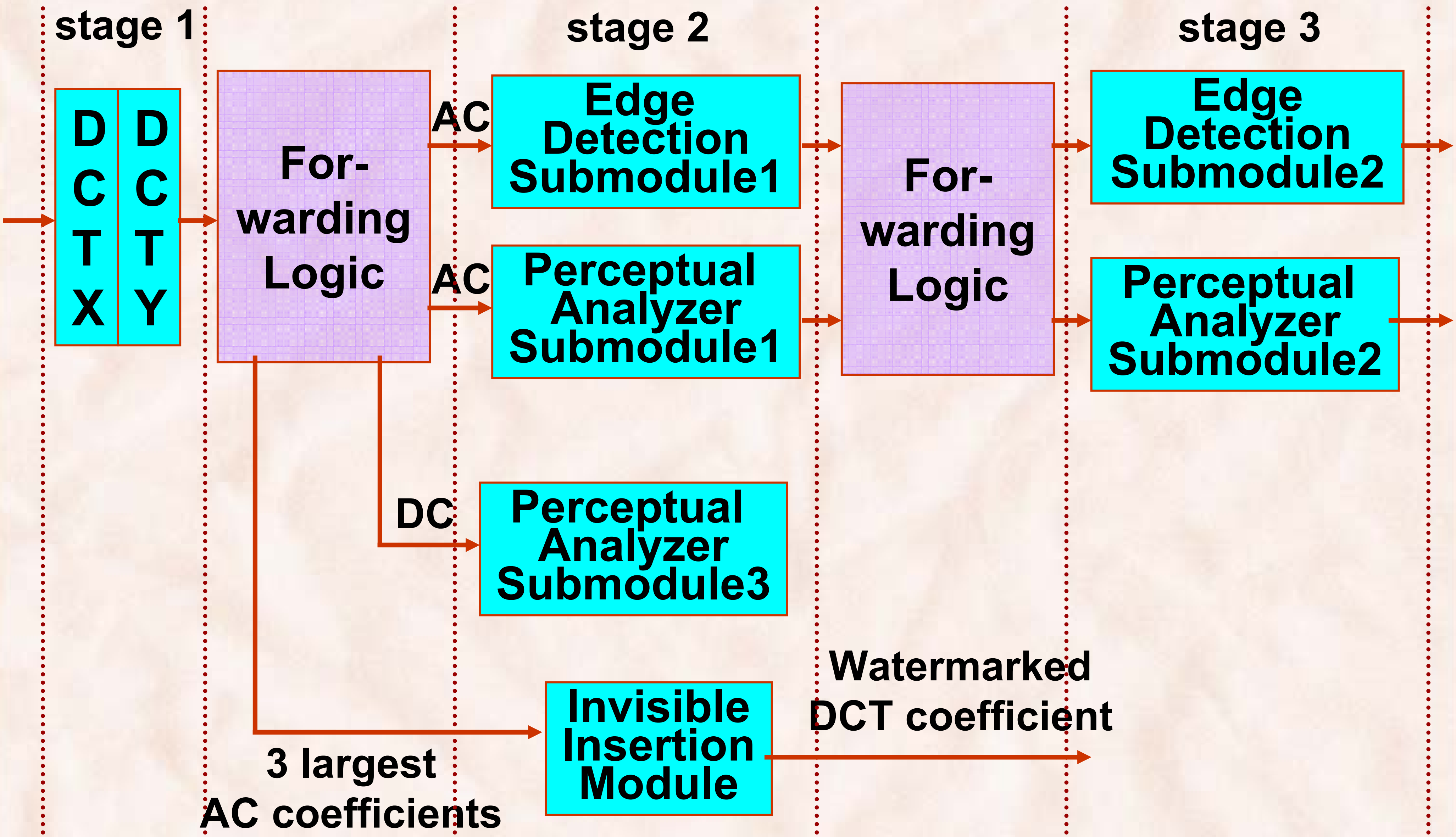


Perceptual Analyzer Module

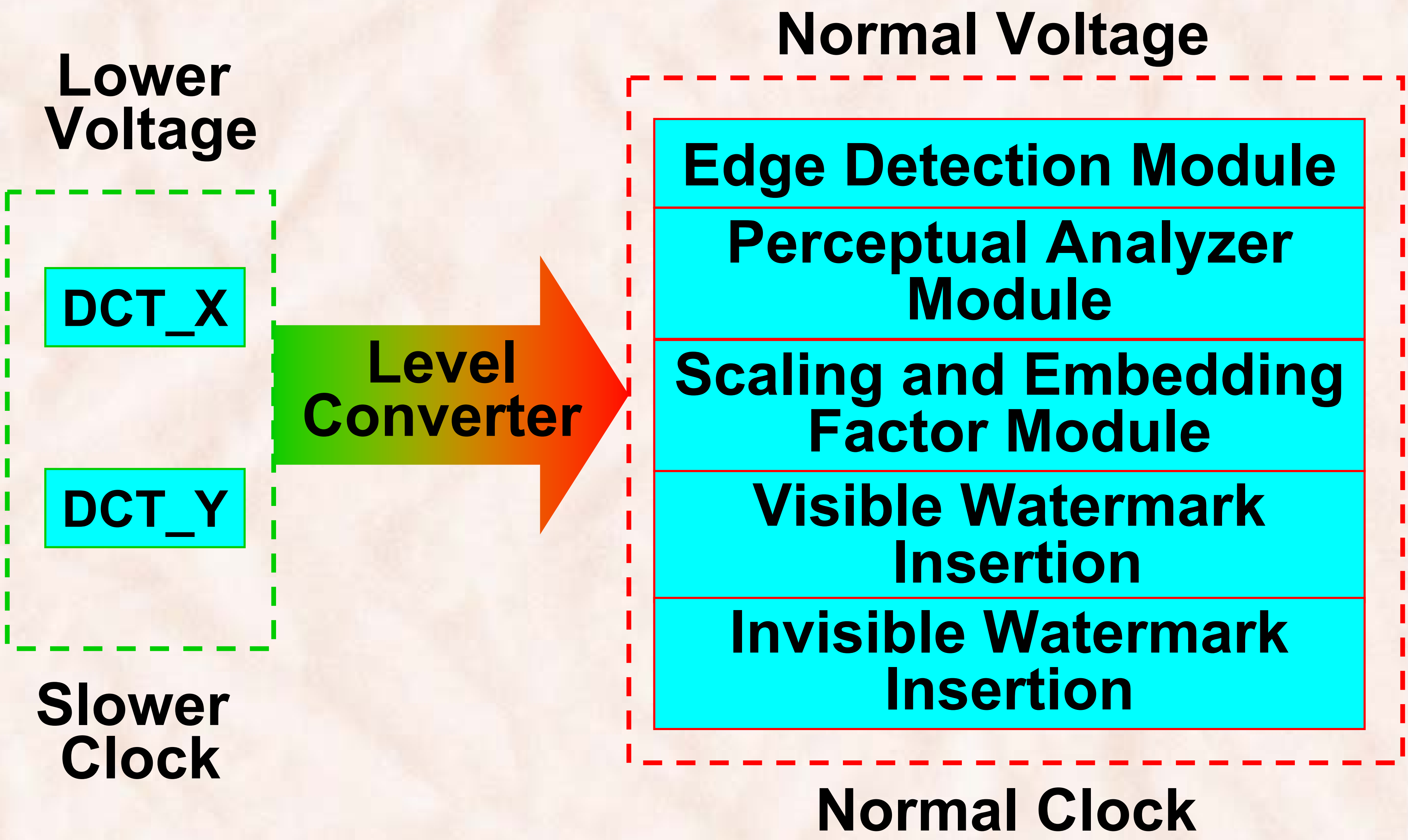


Scaling and Embedding Factor Module

Pipeline and Parallelism



Dual Voltage and Dual Frequency



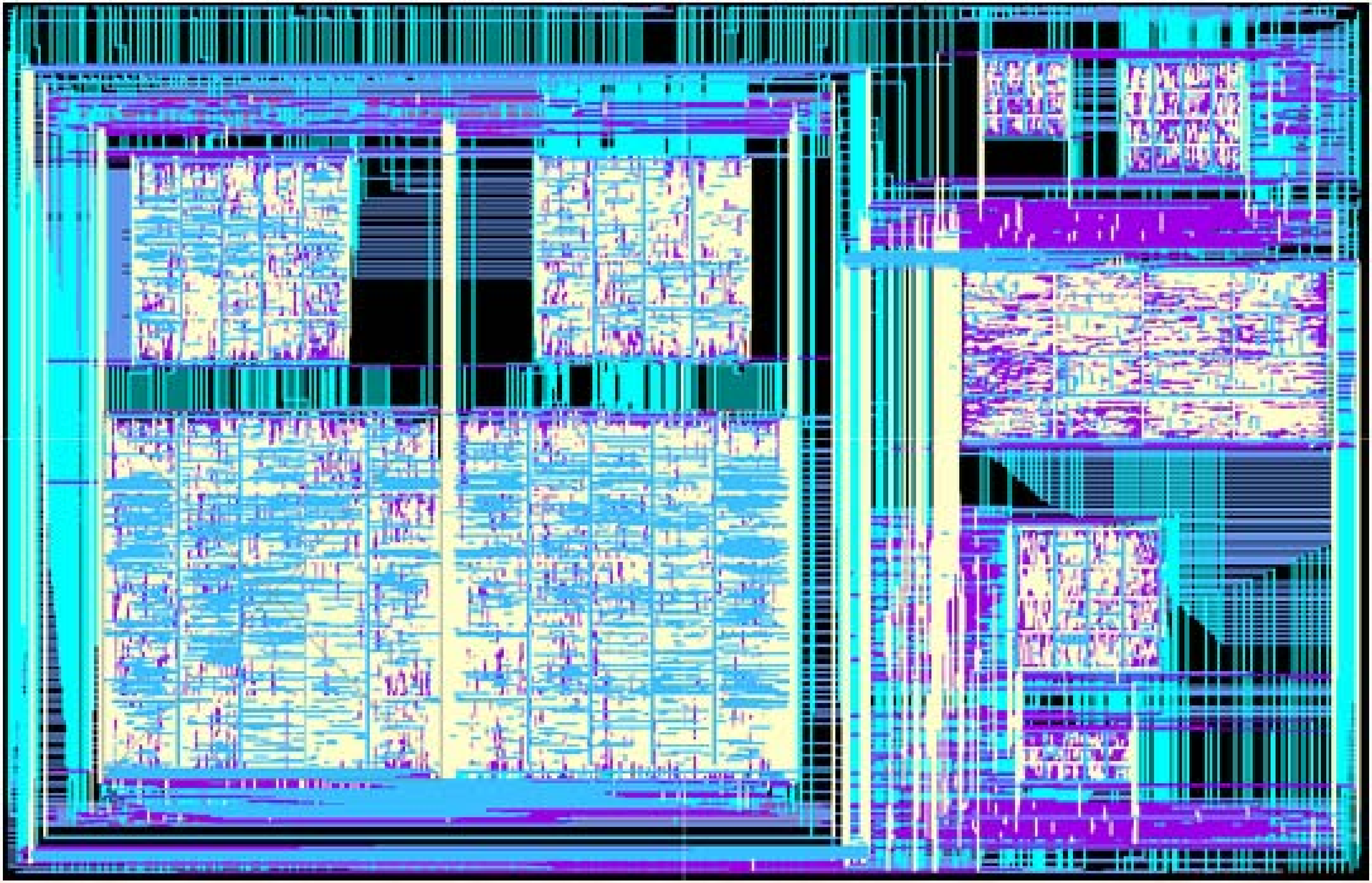
Prototype Chip Implementation

Tools used for the design

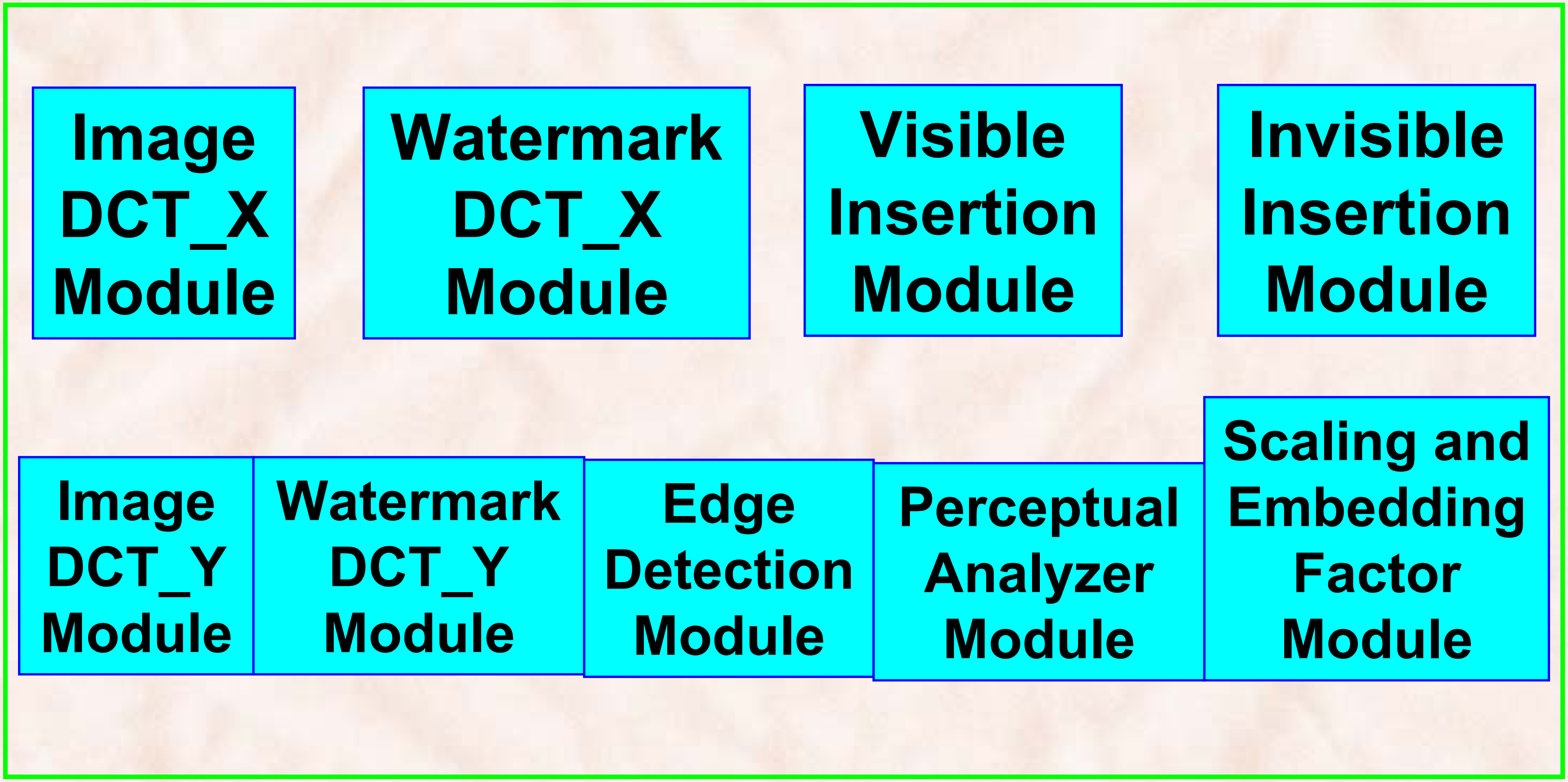
Tools used	Purpose
NC launch	VHDL simulator
Silicon Ensemble	Placement and routing
Abstract Generator	Abstract generation from layout
NCSU-Design Kit	Layout Editor
Design Analyzer	Verilog netlist generation
Nanosim	Power and delay calculations

Standard Cell Design Style adopted. Standard Cells obtained from Virginia Tech. Technology: TSMC 0.25 μm

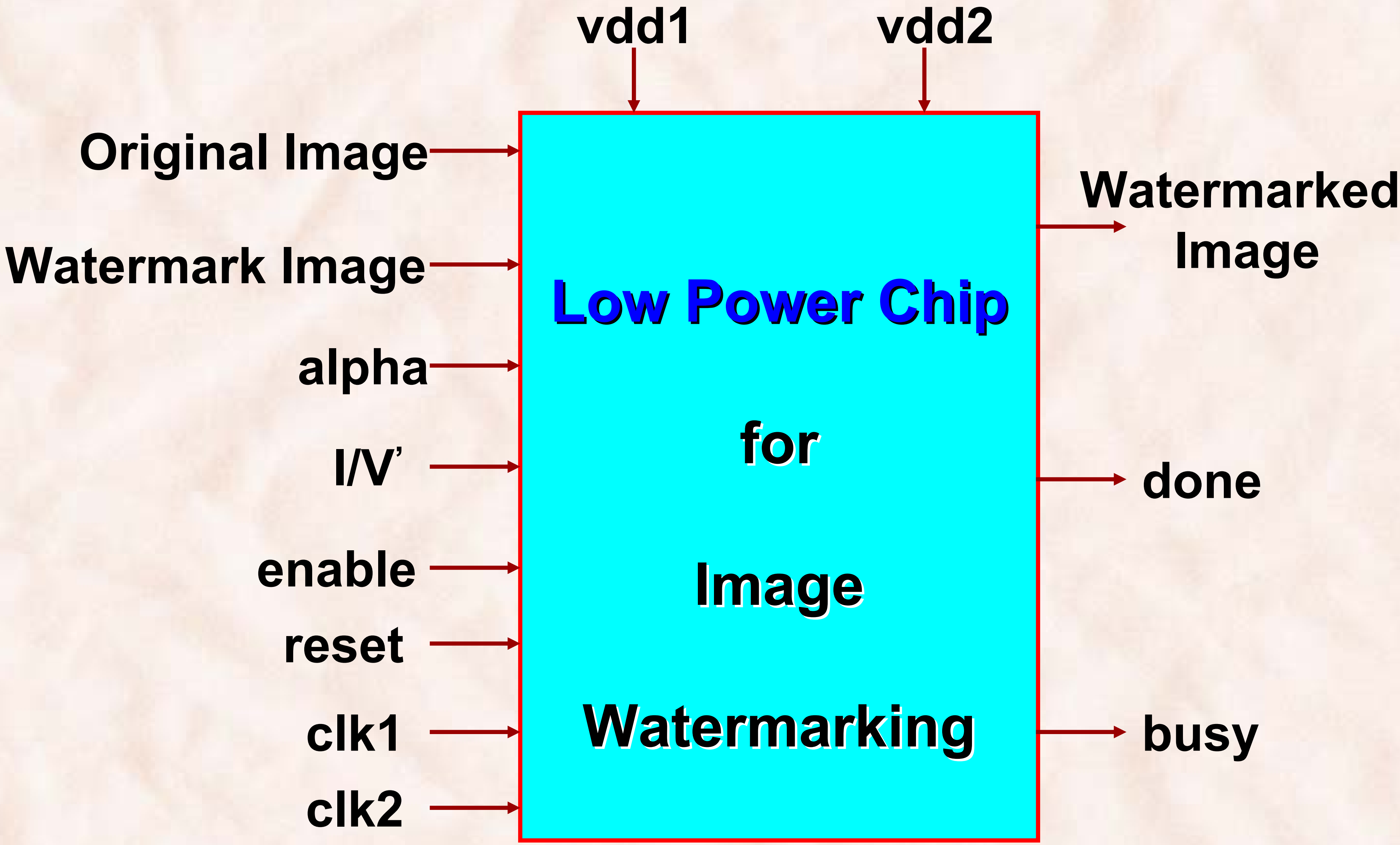
Overall Prototype Chip: Layout



Prototype Chip: Floor plan



Prototype Chip: Pin diagram



Prototype Chip: Statistics

Technology: TSMC 0.25 μ

Total Area : 16.2 sq mm

Dual Clocks: 284 MHz and 71 MHz

Dual Voltages: 2.5V and 1.5V

No. of Transistors: 1.4 million

Power (dual voltage and frequency): 0.364 mW

Chip (single voltage and frequency): 1.950 mW

Conclusions

- We capture power fluctuation in MVDFC and MVMC design scenario using the function MPG and minimize it using ILP.
- The MVDFC approach is better alternative. It is observed that for the circuits with equal number of addition and multiplication operations in the critical path the savings are maximum with no time penalty.
- Polynomial time complexity heuristic algorithms can be developed to obtain suboptimal, but faster solutions
- The scheduling schemes are useful for data intensive applications.
- It is observed that the designed chip consumes only one fifth of the power compared conventional design.

Thank you