MEMORY HIERARCHY EXPLORATION FOR LOW POWER ARCHITECTURES IN EMBEDDED MULTIMEDIA APPLICATIONS

N. Kavvadias, A. Chatzigeorgiou, N. Zervas, S. Nikolaidis

Section of Electronics and Computers, Department of Physics
Aristotle Univ. of Thessaloniki, 54006 Thessaloniki, Greece

This work was supported by the ED 501 PENED'99 project funded by G.S.R.T. of the Greek Ministry of Development an the European Union.

ABSTRACT (1)

- In data-dominated applications e.g. multimedia processing, the major part of the power consumption is due to memory accesses
  - i. Data memory accesses
  - ii. Instruction memory accesses

- Significant power savings can be achieved by applying data-reuse transformations aiming at a custom memory hierarchy

ABSTRACT (2)

- The effect of data-reuse transformations on power, area and performance for a single processor environment is efficiently explored

- Data-reuse transformation methodology is applied and application specific data-reuse trees are introduced

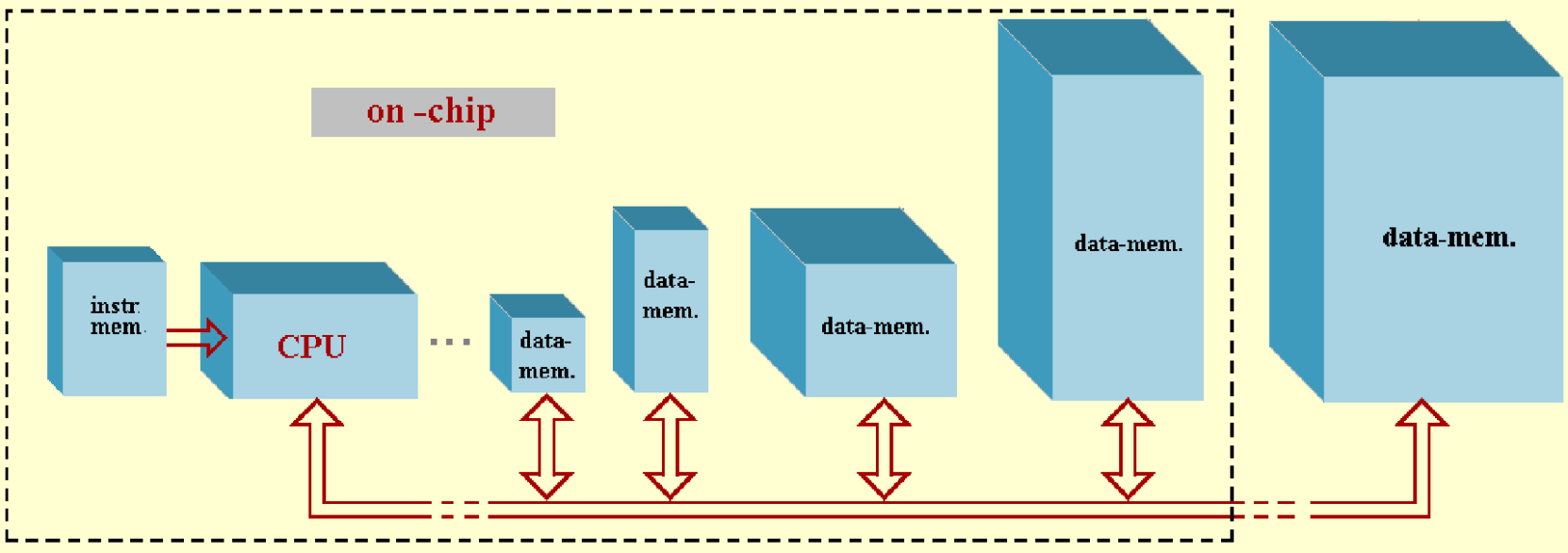
- As demonstration applications, two variations of the DCT algorithm are presented, namely the *2D row-column decomposition* (typical) and its fast version used in *MPEG-2*

TARGET ARCHITECTURE

Single-processor target architecture

- One embedded processing unit communicating with several data memory layers, through a global bus and one instruction memory

- Additional memory layers are considered to reside on-chip



ROW-COLUMN DECOMPOSITION DCT ALGORITHM

```
/* Transposition of the input data array */
for(i=0; i<N/Bi++)
    for(j=0; j<M/B; j++)
{
     /* Transposition of the block being processed */
}

 /* First 1-D DCT for the rows of the initial data array */
  for(i=0; i<N/B; i++)
    for(j=0; j<M/B; j++)
{
   for(k=0; k<B; k++){
     for(m=0; m<B; m++){
     temp=0;
     for(l=0; l<B; l++)
      temp+=coeff[m][l]*image[B*i+l][B*j+k];

    output1[B*i+m][B*j+k]=temp;
    }
   }
}
/*Same steps follow for transposition of the 1st output data array
  and the appliance of 1-D DCTs for the columns
  of the 1st output data array data array */
```

DATA-REUSE TRANSFORMATIONS

- They are used for exploiting the temporal locality of data memory accesses

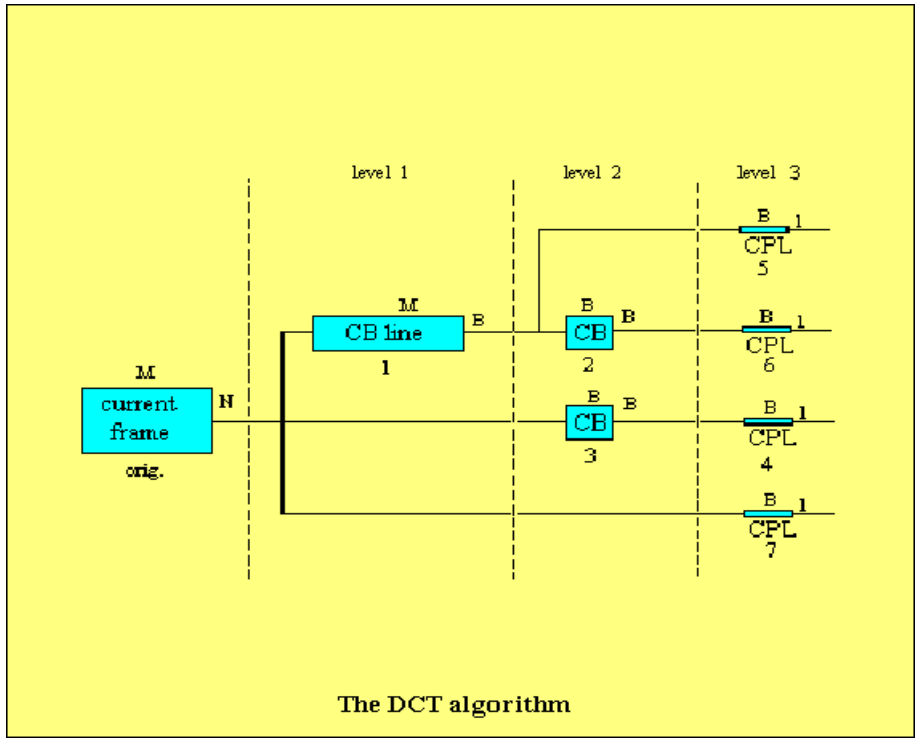
- Data-reuse leads to a custom memory organization
  - Data sets often being accessed in short periods of time, are identified and placed into smaller blocks of memory hierarchy (closer to the CPU)

- The total number of accesses increases while the average power cost per access decreases. An exploration procedure is needed

- Data reuse exploration is performed by applying a number of code transformations to the original code

COPY TREES FOR DATA-REUSE DECISION

- Data-reuse transformations and corresponding memory hierarchy levels for the DCT algorithms (processing is done on the current frame basis)



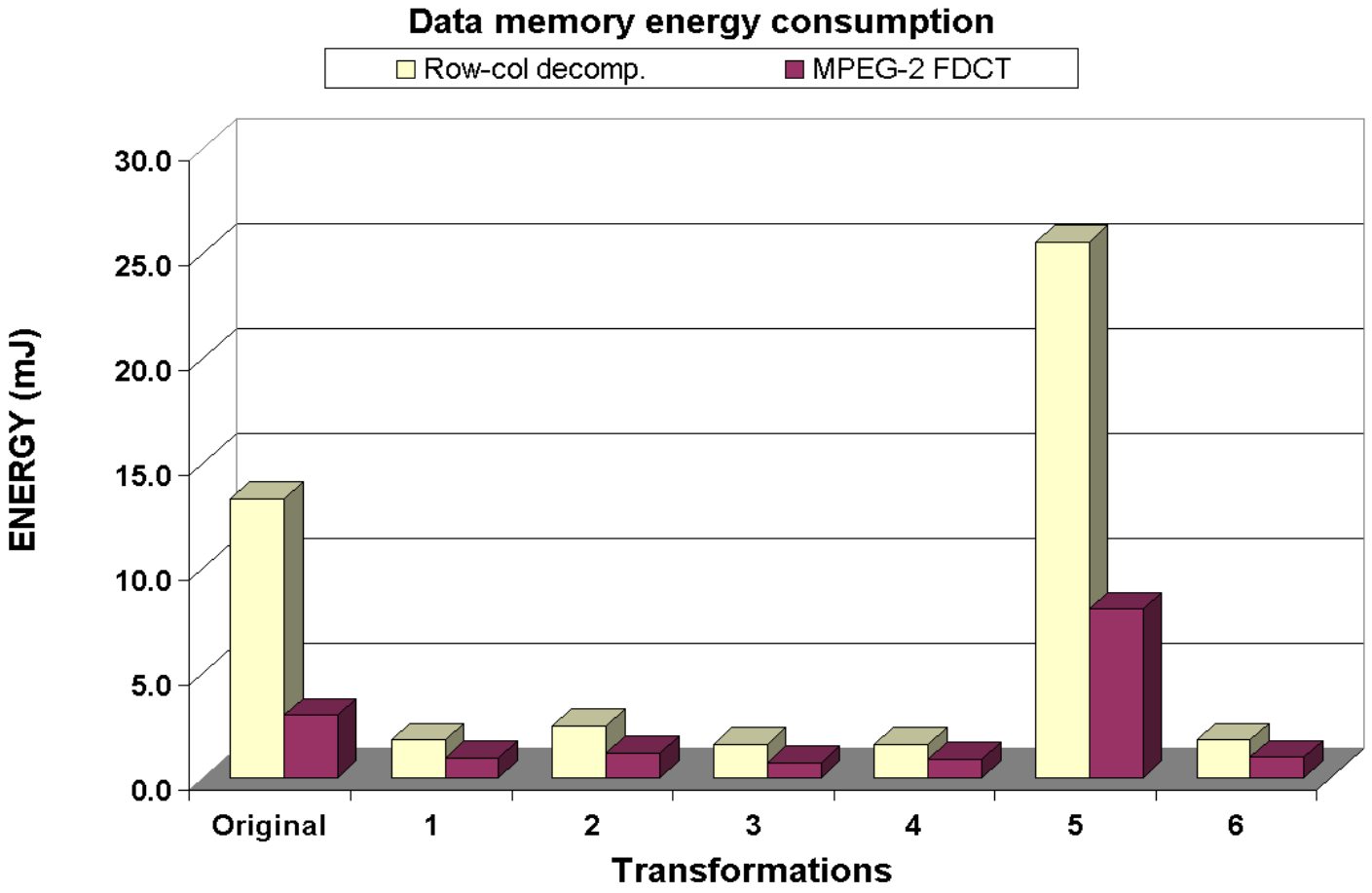
The DCT algorithm

ENERGY CONSUMPTION MODEL OF MEMORY

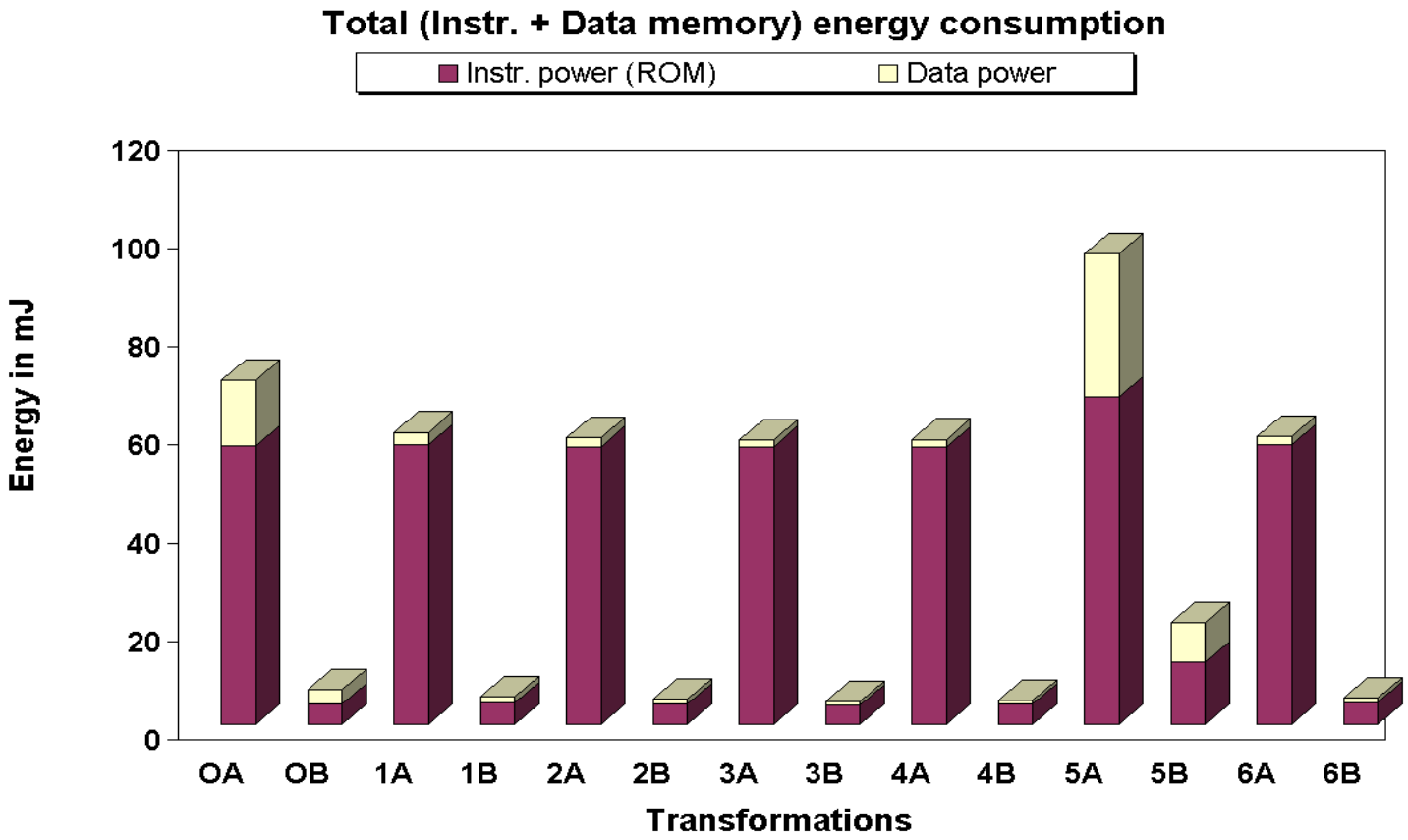
- Most of the energy consumption is due to memory accesses
- Energy consumption on data and instruction memories depends on:
  - ✓ Memory size (S)
  - ✓ Access frequency (f)
  - ✓ Technology related factors
  - ✓ Power supply (V_{DD})
  - ✓ Number of ports
- For a given technology and V_{DD} the consumed energy is:

$$E_{d\_total} = \sum_i f_i F\left(S_i, Nr\_ports_i\right)$$

DATA MEMORY ENERGY CONSUMPTION (Results)

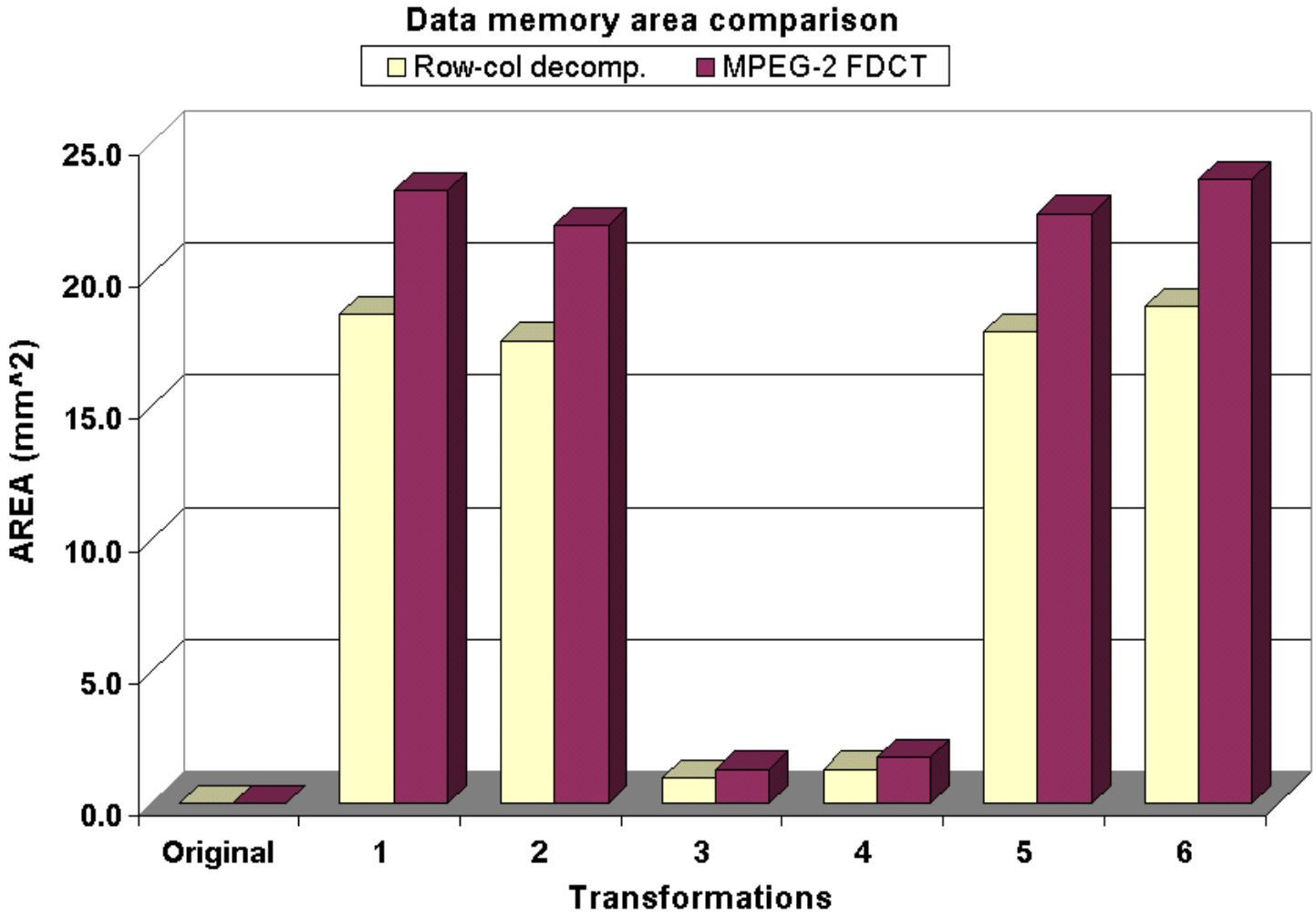


INSTR. MEMORY VS TOTAL ENERGY CONSUMPTION

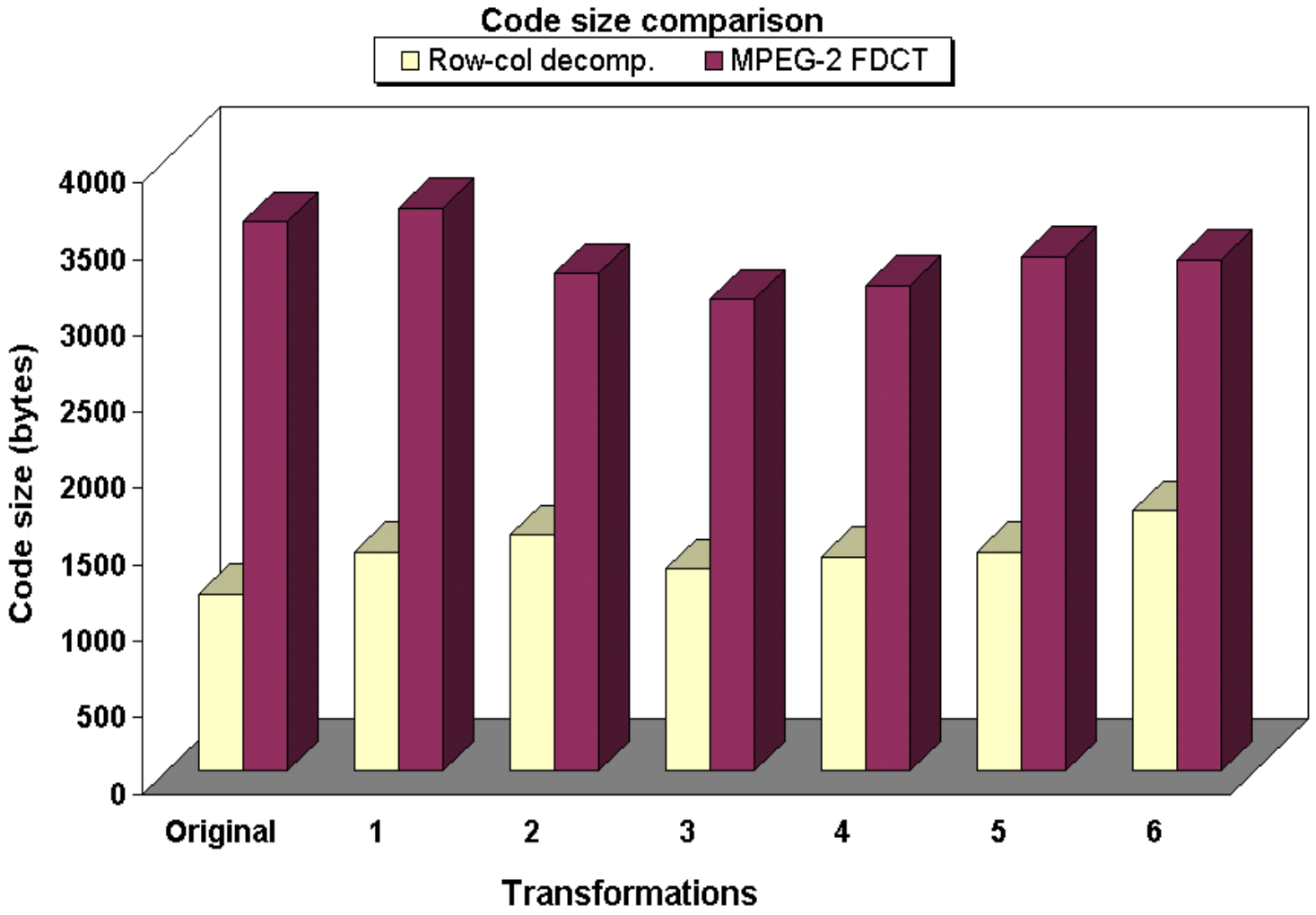


The DCT algorithm: (a) typical , (b) fast

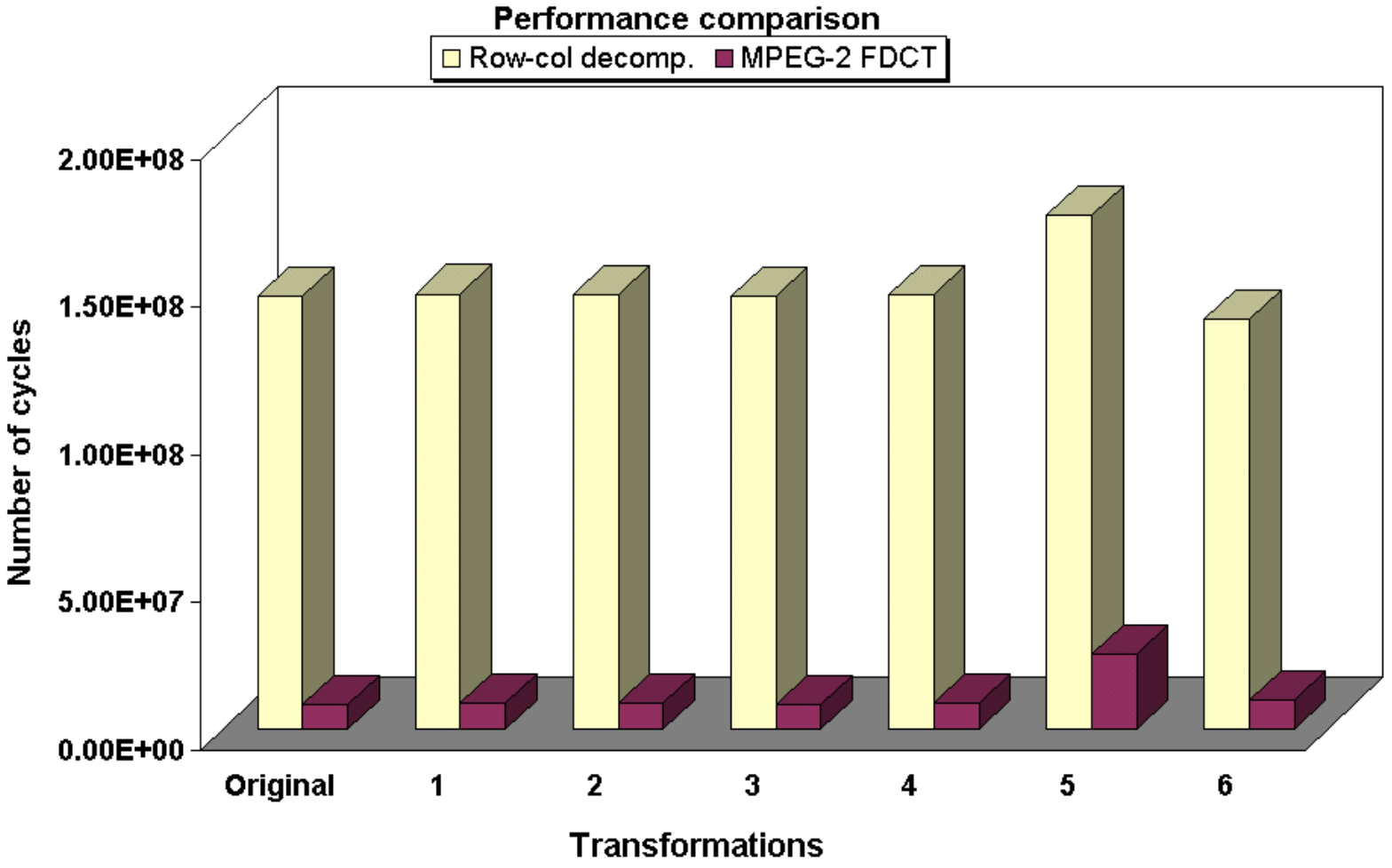
DATA MEMORY AREA OCCUPATION (Results)



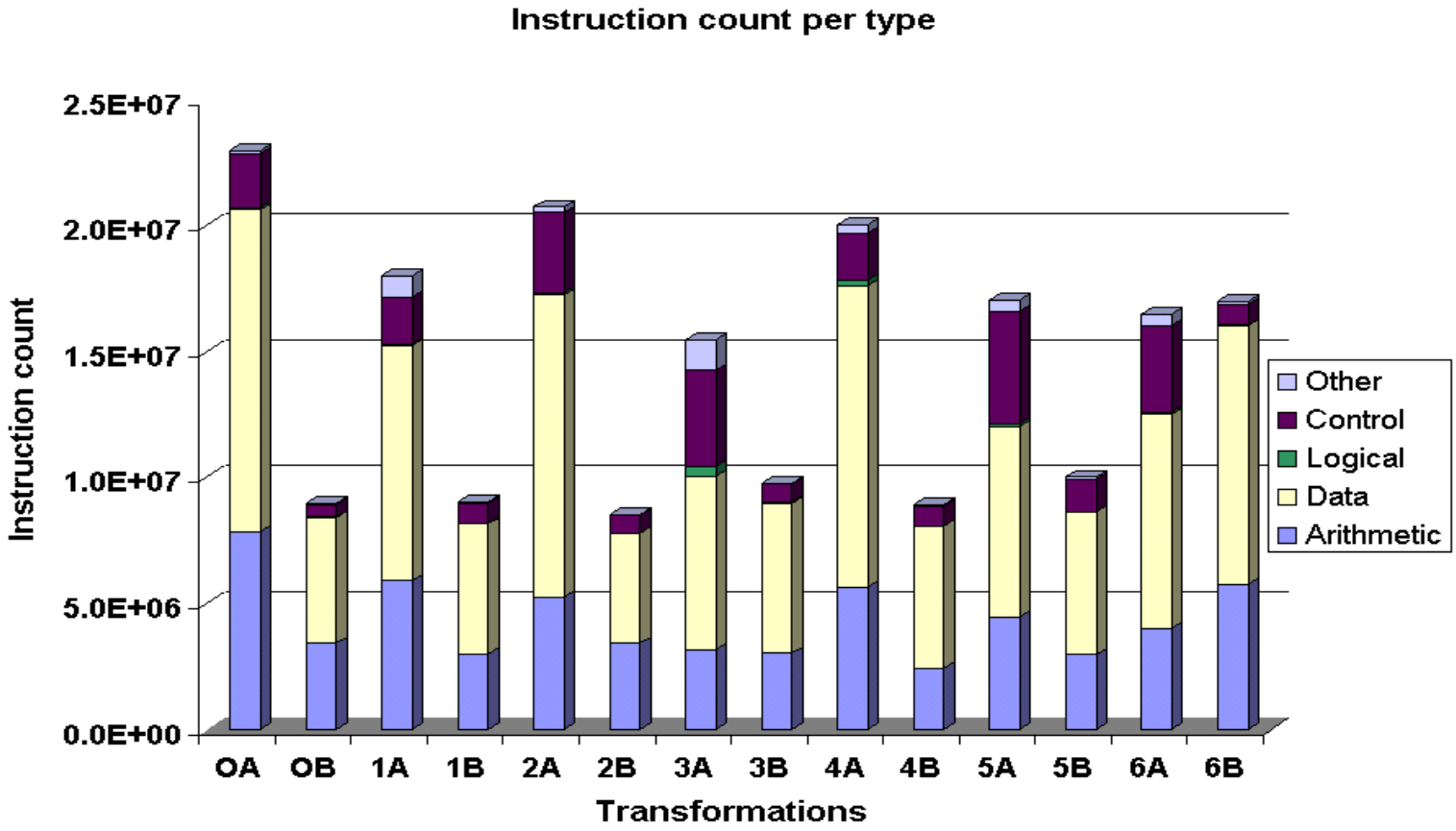
CODE SIZE COMPARISON (Results)



PERFORMANCE COMPARISON (Results)



INSTRUCTION COUNT (Results)



The DCT algorithm: (a) typical , (b) fast

DISCUSSION/CONCLUSIONS (1)

- Instruction-memory related energy component is significantly greater than the data-related energy, when implementing embedded multimedia applications on a general-purpose processor
- Considering data memory energy consumption, transformation **#3** provides the optimal solution for the ME algorithm.
- #3 indicates a memory hierarchy structure that incorporates a single additional layer of block (***BxB***) size
- For the most efficient data-reuse transformation, a power reduction factor of **10** and **3** is achieved for the typical and fast DCT, respectively

DISCUSSION/CONCLUSIONS (2)

- The introduction of additional data memory area comes with an acceptable penalty (transformations #3 and #4)
- Almost no performance reduction is introduced by the application of the data-reuse transformations
  - highly regular algorithms with small number of control operations
  - reduction of data and arithmetic operations as a result of *simplified addressing*
- The selection of memory hierarchy should be based mainly on power criteria and only in extreme cases, area occupation could be considered