TI Developer Conference
March 7-9, 2007 • Dallas, TX

Minds in Motion

Scalable Multi-DM642-based MPEG-2 to H.264 Transcoder

Arvind Raman, Sriram Sethuraman
Ittiam Systems (Pvt.) Ltd.
Bangalore, India

Ittiam
i think therefore i am

Outline of Presentation

- MPEG-2 to H.264 Transcoding
- Need for a multiprocessor implementation
- TI C64x architecture
- Multiprocessor board details
- Design options and metrics
  - Spatial split
  - Functional split
  - Load balancing
  - Ability to scale from CIF to HD
- Demonstration
- Conclusions

Transcoding Approaches

- Transform domain
  - With Drift
  - Minimal Drift
  - Drift free
- Complete decode and re-encode

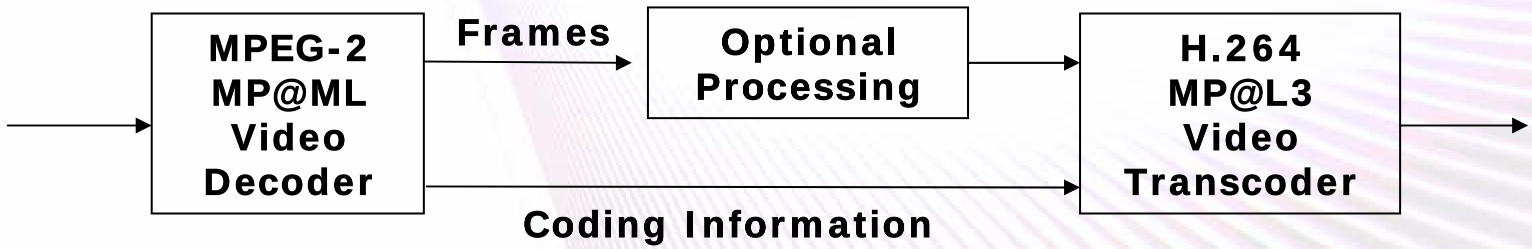
- Transcode
  - Convert from one standard to another
- Trans-rate
  - Convert the bit-rate
- Trans-scale
  - Downsample the input stream's resolution and then transcode

MPEG-2 to H.264 Transcoding

- MPEG-2 dominates the digital video broadcast space
- H.264 offers nearly 50% more compression over MPEG-2
- Transcoding from MPEG-2 to H.264 will help
  - Network edge servers to stream over lower bandwidth last mile links at good quality for VoD or multicasting applications
  - Set Top Boxes (STBs) with Personal Video Recorder (PVR) functionality to record more hours of content
- Complexity of transcoding is higher as
  - Compressed domain transcoding is not possible
    - Different block sizes are employed in MPEG-2 and H.264
    - Tools such as intra prediction, deblocking and CABAC are of higher complexity
  - Re-use of information can be used to reduce complexity

Lower Complexity Transcoder

- Re-uses information obtained from the MPEG-2 stream
  - Motion, modes, residual energy, bit allocation, …
- Drift-free as re-encoding is not done in the decoding loop



Compression Efficiency Gain

- Transcoded quality
  - can never be the same as the MPEG-2 stream quality
    - unless losslessly coded
  - Details in video cannot be distinguished from artifacts in decoded video
  - Optional processing can reduce artifacts or change resolution
- Similar visual quality is possible with coding tools such as
  - Better Intra prediction
    - In broadcast, I-frames occur every half a second
      - consume ~35% of the total bits
  - In-loop filter
  - CABAC
    - Better entropy coding
  - Quarter sample and segmented MC
    - Reduces residuals to code
  - 25-30% reduction is possible at similar quality with above tools
  - Further reduction with multiple reference frames, weighted pred, etc.

Need for Multi-processor Transcoding

- Brute force transcoding
  - MPEG-2 decoder cascaded with a H.264 encoder
  - Only CIF MP@ML transcoding is possible on a single DM642 600MHz
- Low complexity transcoding
  - Several approaches - transform domain, re-use of information
  - 3/8ths-D1 MP@ML transcoding is possible on a single DM642 600MHz
- Single processor solutions
  - Cater to the low end market (sub-SDTV resolutions).
  - Do not scale or adapt to market / application needs

Multi-Processing (MP)

- Several Processing Units (PUs) work together towards a common objective
- Many approaches to design such a system
  - Functional split
  - Spatial split
- Common objectives for MP design
  - Load balancing
  - Flexible to changes in the complexity of the algorithm / input
  - Adapts to the number of PUs in the cluster
  - Increase in workload is handled by increasing the number of PUs
  - Efficient in using the resources (memory, inter-PU bandwidth)

TI C64x DSP Core

- High speed VLIW architecture (VelociTI)
  - Orthogonal functional units
  - 2 Datapaths with 4 functional units each
  - Packed SIMD operations on 8 or 16-bit data
- 2-level cache (16 kB I and D – L1; 256 kB L2)
- 64-channel enhanced DMA (TCInt, Chaining, linking)
- Ideally suited for high performance 8-bit video processing

Multiprocessor DM642 Boards

- Boards from vendors such as Vitec Multimedia, Mango DSP
- Typical designs are based on
  - Multiple DM642s with their dedicated SDRAMs
  - An FPGA-based DMA capability from SDRAM of one DM642 to the SDRAM of any other DM642
    - Ability to concurrently handle independent transfers
  - PCI connectivity with host
  - Carrier or PMC form factors
- Examples: Vitec VP3-PMC board

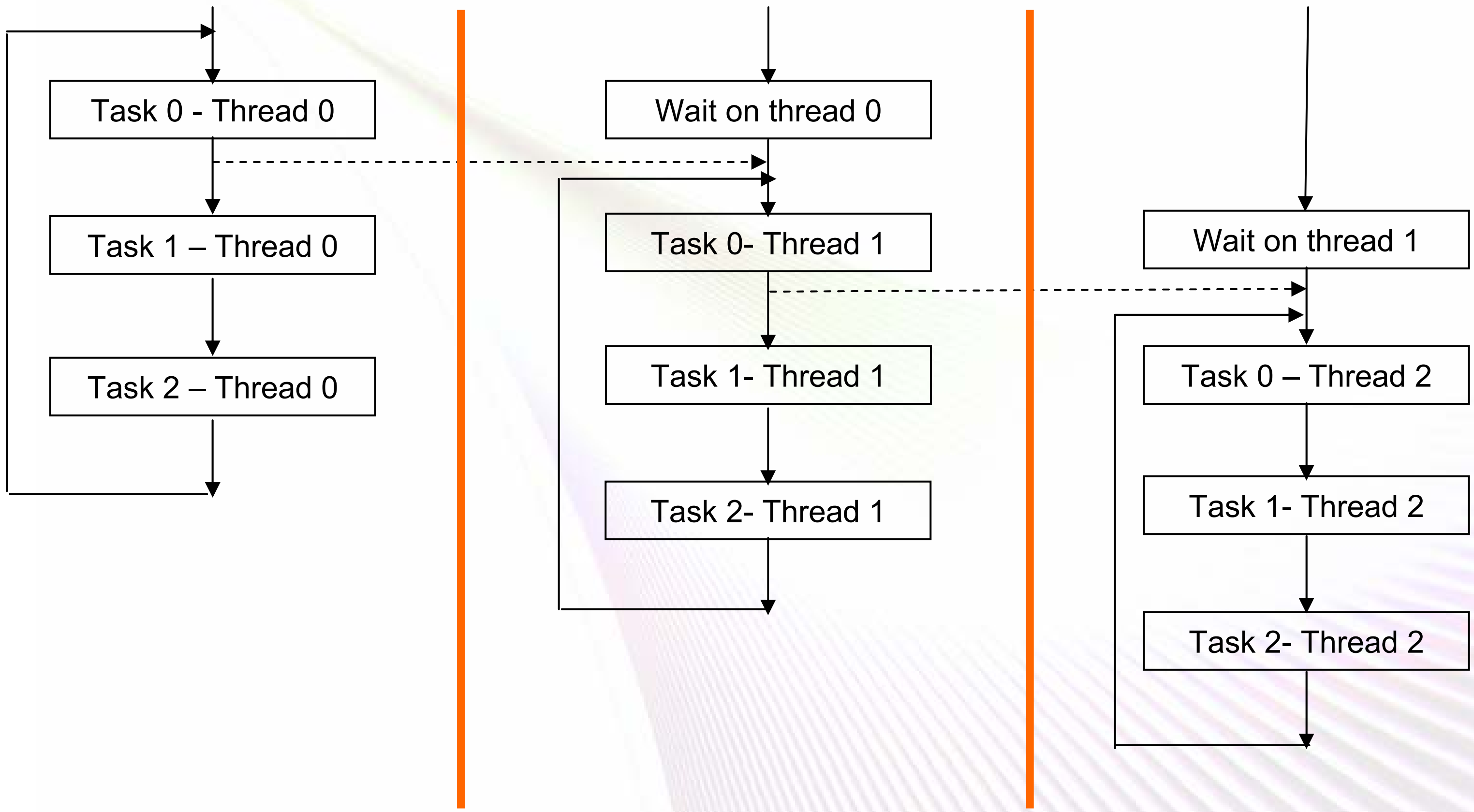
Video Codec – Multiprocessing Aspects

- Spatial prediction/availability constraints
  - Cannot proceed without spatial neighbor information availability
- Temporal prediction constraints
  - Motion vector can point anywhere in the reference frame
- Bandwidth of inter-processor communications
  - Intermediate precisions in video will vary
- SDRAM access
  - All code may not fit in internal memory
  - All scratch or persistent data may not fit in internal memory
  - Cache misses need to be balanced with inter-SDRAM transfers
- Scalability to resolutions and features

Ittiam

Functional split – Part I

- Output of one PU is the input to the other

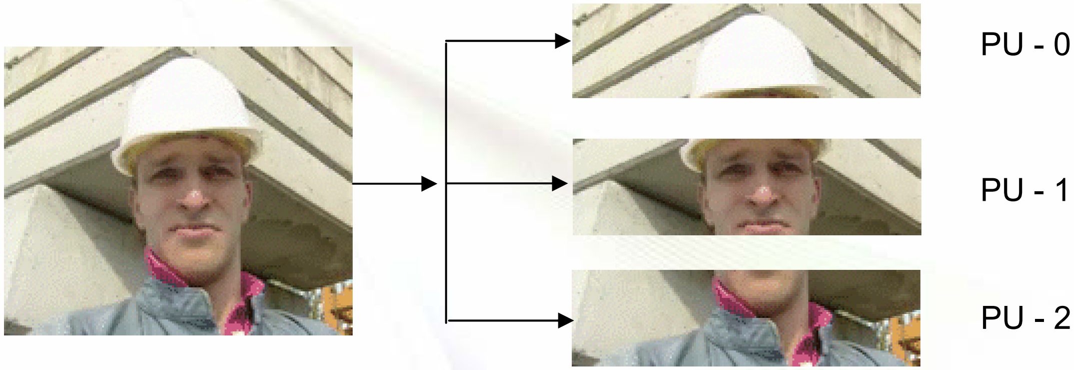


Functional split – Part II

- Issues with functional-split
  - Careful partitioning to balance loads across PUs
  - Does not scale well with the complexity of the algorithm
  - Does not adapt to the number of PUs in cluster
  - Adding extra PUs may not help process extra work load
  - Inter processor data exchange tends to be higher
- Not everything about it is bad
  - Code size on each PU is lower

Spatial split – Part I

- All PUs are identical
- Divide the input instead of tasks



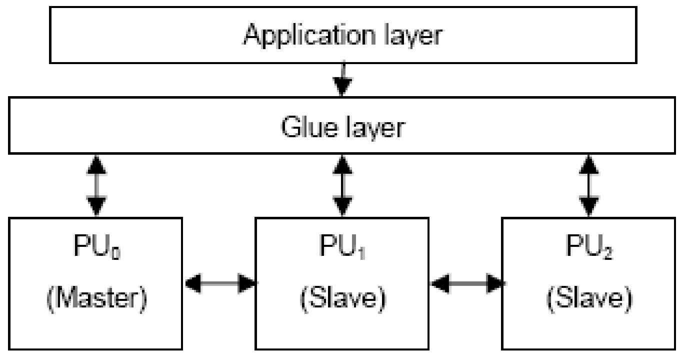
Spatial split – Part II

- Advantages
  - Load balancing is inherent since PUs are alike
  - Scales well with the complexity of the algorithm
  - Adapts well to the number of PUs in the cluster
  - Extra work load can be handled by increasing the number of PUs
  - Are efficient in handling resources
  - Interprocessor data exchange can be kept low
- Disadvantages
  - Code size hit as each CPU runs the same code
  - Large internal memory, cache and intelligent code sectioning mitigate the impact of the increased code size.

DM642 Code Management

- L1P is 16kB
- L2 can be configured as 224kB ISRAM and 32kB cache
  - as application code for a transcoder is minimal
- Load time vs. run time address specification allows dynamic downloading of code sections
  - Facilitates overlay of multiple code sections in ISRAM
- Structure code so as to perform the same processing operation on multiple data to reduce code thrash in L1P

Spatial split – Architecture



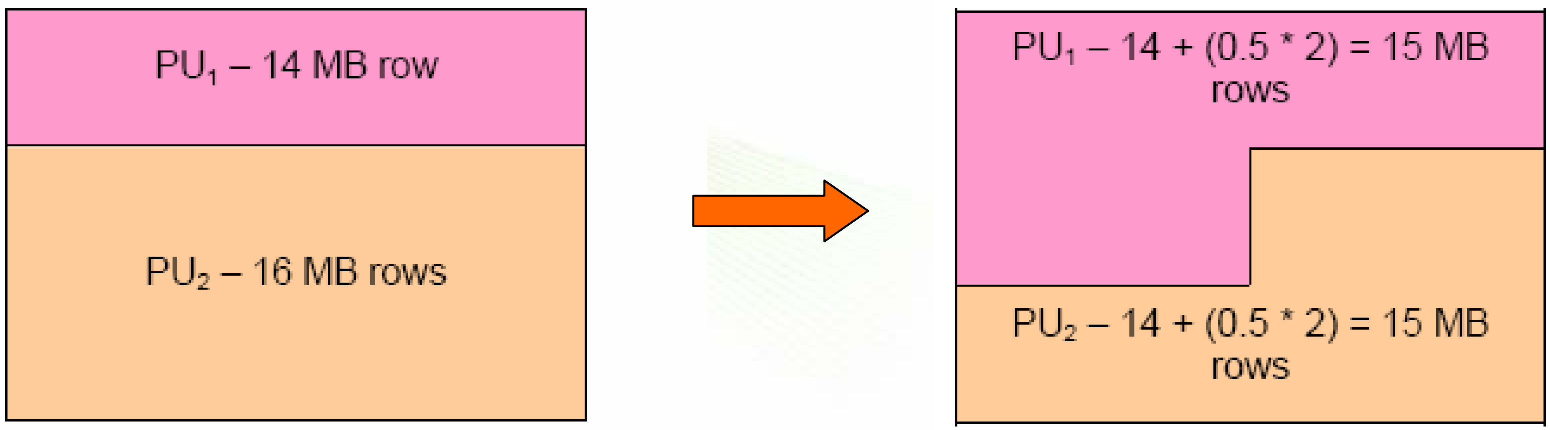
- PU-0 is the master PU ; PU-1, PU-2 are slave PUs
- Each PU process encodes equal number of MB rows as slice
  - Easily done since every MPEG-2 row starts on a new slice.
- PUs may have only Right or Left neighbor or both
- Application sends control / config parameters to master PU
- PUs exchange portions of their reconstructed picture buffers – MPEG2 and H.264 with their neighbors
- PUs synchronize with their neighbors and master PU.

Spatial split – Concurrency

- Little dependency across PUs
- PUs are equally loaded => synchronization time is minimal
- Some bottlenecks exist however
  - H.264 MP does not allow Arbitrary Slice Ordering (ASO)
    - Bitstream needs to be concatenated before sending
    - Similar constraints do not exist in H.264 BP
  - De-blocking should be done in a strict raster-scan order
    - Disabling de-blocking across slices overcomes the constraint
  - Rate control modifications
    - Slice level bit allocation model
    - Possibility of quality deterioration

Spatial split – Load sharing

- Simple rectangular partitions may not always suffice e.g partitioning a NTSC sequence across 2 PUs



- Requires the MPEG-2 decoder to decode two extra MB rows

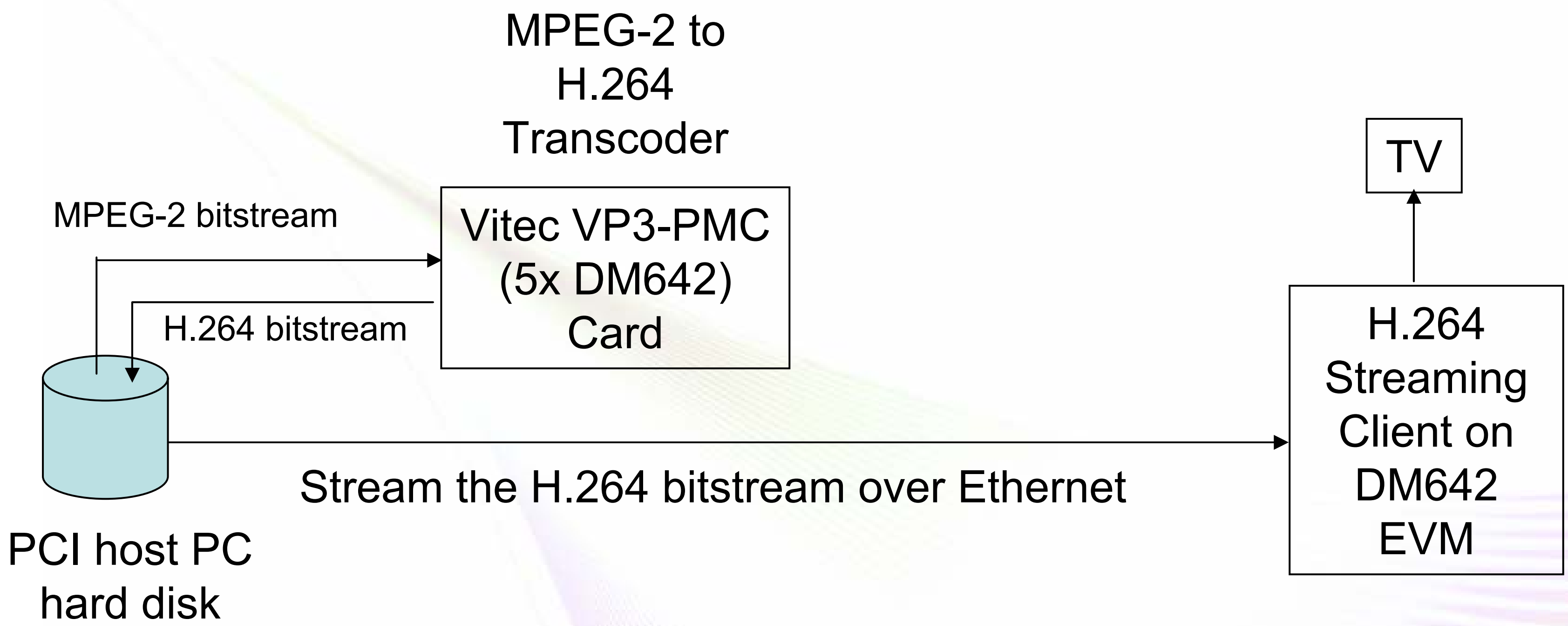
Resource requirements

- Spatial split configuration
  - 2 PUs
  - Inter PU data bandwidth ~ 134Mbps
- Functional split configuration
  - 1st PU – MPEG-2 decode + H.264 motion estimation
  - 2nd PU – Does the remaining of the encoding operation
  - Inter PU data bandwidth approx 330Mbps
- Resource requirements in functional split will depend on split but may disturb the load balance
- In general resource (spatial) <= resource (functional)

Functional vs. Spatial split

Comparison	Functional split	Spatial split
Load sharing	X	√
Flexibility	X	√
Inter PU data bandwidth	X	√

Demonstration Setup



- Only 2 DM642 processors are used for 3/4th D-1 transcoding
- Bit-rate reduction of 33% compared to the input

Conclusions

- Spatial split is superior to Functional split in terms of
  - Scaling with resolution
  - Memory bandwidth
  - Ease of design and maintenance
- Multi-DM642 transcoder
  - Scalable from sub-D1 to HD resolutions without effort
  - Algorithms can be improved without affecting the design

TI Developer Conference
March 7-9, 2007 • Dallas, TX

Scalable Multi-DM642-based MPEG-2 to H.264 Transcoder

Arvind Raman, Sriram Sethuraman
Ittiam Systems (Pvt.) Ltd.
Bangalore, India
sriram.sethuraman@ittiam.com
arvind.raman@ittiam.com

Minds in Motion

Application Scenario 1