

SoCs for Portable Video Applications: Architecture level Considerations

Mahesh Mehendale
Texas Instruments (India) Ltd.
Bagmane Tech Park, Bangalore,
India 560093

E-mail: m-mehendale@ti.com

ABSTRACT

In this paper, we first review the digital video processing requirements of portable multimedia applications, and highlight the need for a multi-format, multi-standard digital video (DV) engine. We also highlight the variability in compute requirements across the digital video processing spectrum. Given the consumer portable nature of these applications, cost and energy are the key factors driving the architecture of the DV engine. We present various power management strategies for both dynamic and leakage power reduction, and show how these can be applied in the context of DV engine design by leveraging the “variability”. We discuss SoC architecture level implications of extending these power management strategies and finally present EDA challenges

Categories and Subject Descriptors

B.7.1 [Integrated Circuits]: Types and Design Styles - *Algorithms implemented in hardware*, C.3 [Computer Systems Organization] Special-purpose and Application-based Systems - *Real-time and embedded systems, Signal processing systems*, C.5.4 [Computer Systems Organization] Computer System Implementation, *VLSI Systems*

General Terms

Algorithms, Management, Performance, Design .

Keywords

Digital video processing, Low power design, Power management, Hardware Accelerators,

1. INTRODUCTION

With the advances in technology, there is increasing availability of compute, communication and storage capacity per unit cost

and also per unit energy. This is fueling the growth of digital video functions in portable personal entertainment systems. The spectrum of these applications includes:

- Portable Video Recorder
 - Portable TV (DVB-T, DVB-H)
 - Portable Media Player
 - Digital Camcorder
 - Portable Navigation
 - Video phone
 - Web Terminal
- and more.

While some of these digital video capabilities are being integrated into mobile phones (for example, camera phone), there are stand alone devices which integrate many of these functions.

The digital video processing chain starts with capture of the video, followed by digital encoding to enable efficient storage as well as efficient delivery and finally decoding and display of the video content. There are multiple mechanisms available for each step in the video processing chain, and they aim at achieving high quality under the constraints of compute, storage and communication bandwidth. The digital video processing requirements continue to go up at a rapid pace. There are multiple coding standards (MPEG2, MPEG4, H.264, VC1, ...) and for each standard various levels/profiles are specified providing tradeoff between computational complexity and compression efficiency. In an application such as a portable media player, the coding standard of the video data being decoded varies depending on the source, and hence needs a Digital Video (DV) engine which supports these multiple standards and profiles. In terms of format/resolution digital video is transitioning from standard definition (SD) to High Definition (HD) and this is beginning to happen in portable space as well. Going from D1 to 720P format, the number of pixels per frame goes up by a factor of 2.66, and consequently to maintain the same real-time performance of 30 frames per second, the compute need goes up by a factor of 2.66. One way to support this is to run the hardware 2.66 times faster, but in many cases scaling the MHz many not be feasible and in most cases it is not likely to be power efficient. The rate of increase in the video processing performance requirement, is significantly faster than the rate at which the silicon (transistor + interconnect) performance is going up with each technology node. There is thus clearly a need to look at architecture level solutions which

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

EDP'07, April 12–13, 2007, Monterey, California, U.S.A.
Copyright 2007 ACM 1-58113-000-0/00/0007...\$5.00.

can provide the necessary performance while keeping the power and area optimality in mind.

While the high end of the digital video applications spectrum will continue to push HD resolutions (next is 1080P HD format – which is 2.25 times more pixels per frame over 720P), the digital video content will co-exist in all formats including smaller resolutions such as QVGA/QCIF. This provides a unique challenge (as well as the opportunity) in designing the digital video engine.

A System-on-a-Chip (SoC) for a portable video application thus needs to support the multi-format, multi-standard video encode/decode function in a cost effective way and at low power. Low power design has been a much researched area over the recent years, and multiple techniques for both dynamic as well as leakage power reduction have been published. These techniques span the entire spectrum of silicon process, circuit design to architecture, software and system level strategies. The variability of compute requirements across different standards and resolutions provides a unique opportunity to apply many of these techniques to design an energy efficient DV engine. The video processing is not only compute intensive but data intensive as well. The data bandwidth requirements also vary across standards and formats. The SoC architecture thus has to not only ensure the data availability but also needs to leverage the variability for reducing power. From a customer point of view, the cost and power goals apply to the entire system and not just the SoC. These drive system level considerations (power management complexity, SDRAM size and performance etc.) while architecting the solution. In this paper, we present some of these opportunities and challenges.

The rest of the paper is organized as follows. In section 2, we characterize the variability in the context of multi-format, multi-standard digital video processing. In section 3, we present low power techniques and show how the variability can be exploited to reduce both dynamic and leakage power. In section 4, we discuss a few SoC/System level considerations/tradeoffs, and finally conclude in section 5 by summarizing key challenges and opportunities.

2. Multi-standard, Multi-format Digital Video Engine

Digital video is a representation of a real-world visual scene, sampled temporally and spatially. A higher temporal sampling rate (frame rate) gives apparently smoother motion in the video scene but requires more samples to be captured. Sampling at 25 to 30 frames per second is standard for television pictures; 50 to 60 frames per second produces smooth apparent motion. In terms of spatial sampling, the visual quality of an image is influenced by the number of sampling points. Higher resolution thus implies better visual quality but translates to higher data per frame. The advances in technology (capture, process and display) now enable higher resolution, and consumer entertainment systems are rapidly moving from SD (Standard Definition) to HD (High Definition) resolution. Table 1, lists the resolution for various formats. As can be seen from the table, 720P HD format has more than 9 times the pixels per frame over CIF (Common

Intermediate Format). Since video content is available across this entire range of formats, the digital video engine needs to support them. The multi-format requirement thus results in significant variability in both compute and data bandwidth requirements in video processing.

Table 1: Video Formats from CIF to HD

Format	Resolution (rows x column)	# pixels
CIF	288 x 355	1.00 times CIF
VGA	480 x 640	3.03 times CIF
D1	480 x 720	3.41 times CIF
720P	720 x 1280	9.09 times CIF
1080i	1080 x 1920	20.45 times CIF

‘Raw’ or uncompressed digital video typically requires a large bit-rate (~216Mbits for 1 second of uncompressed SD TV-quality video) and compression is necessary for practical storage and transmission of digital video. Since lossless compression gives only a moderate amount of compression, lossy compression is necessary. Lossy video compression systems are based on the principle of removing subjective redundancy in both spatial as well as temporal domain, without significantly impacting the viewer’s perception of visual quality. While a higher compression can be achieved at the cost of quality, newer video compression standards aim to achieve higher compression for the same visual quality. This however is achieved at the expense of significantly higher computational complexity. There are industry standards for video coding such as MPEG-2, MPEG-4, H.264, VC1, AVS and also proprietary algorithms such as On2 and Real Video. In the context of a digital media player, the video content can be received in various formats depending on the source (DVD, Digital video broadcast DVB-H/DVB-T, streaming over internet etc.). The digital video engine hence needs to support these multiple standards. Also, for a standard such as MPEG4 or H.264 various levels/profiles are defined, each enabling tradeoffs between the compression ratio, computational complexity and visual quality. The digital video engine needs to support these variants as well.

While each standard and each level/profile for a given standard requires unique processing, most of the standards are based on a CODEC (enCOder/DECOder) model which uses block-based motion compensation, transform, quantization and entropy coding. Figures 1 and 2 below, show block diagrams of H.264 encoder and the decoder.

Architecture of the CODEC engine and the digital video subsystem has the biggest impact on the die size and the power of a portable video system. Given the requirement to support multiple standards, one approach is to use a general purpose embedded RISC processor. While this approach gives maximum flexibility, such a solution does not scale to meet the performance requirements of HD video processing (decoding requirement of H.264 - 720P decode at 30 frames per second is 6 to 9 GOPS) and is also energy inefficient. A VLIW multimedia processor with video specific instruction set and co-processor support (e.g.

Sum of Absolute Difference computation (SAD)) can meet the performance requirements but is not as energy efficient as a dedicated custom hardware accelerator. The most energy efficient solution thus is to build dedicated custom hardware accelerators for each standard – but such an approach is inflexible and not likely to be area efficient, as the number of standards to be supported goes up. There is thus a need to build a configurable, scalable digital video encode/decode engine which is area efficient while achieving the energy efficiency of a custom hardware accelerator.

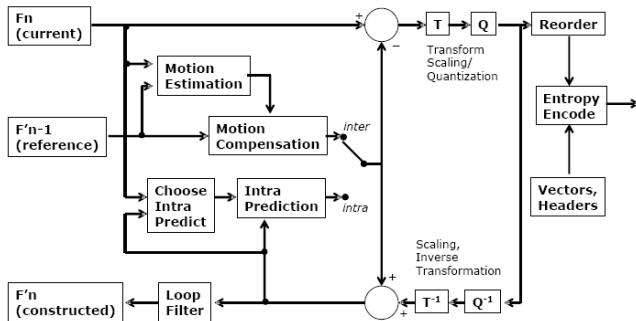


Figure 1: H.264 Encoder

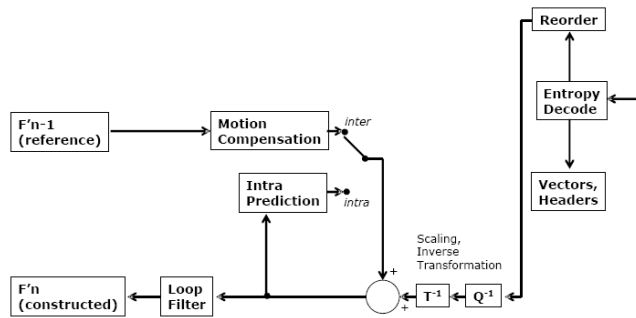


Figure 2: H.264 Decoder

As mentioned earlier most video standards do block based processing. Each frame is partitioning into 16x16 macro-blocks and these are processed thro’ the encode/decode engines similar to those shown in figures 1 and 2. The specifics of the functionality in each of the blocks however vary. The DV engine thus can be built around hardware accelerators which perform each of these functions (i.e. motion estimation, motion compensation, scaling and transformation, entropy coding and loop filtering). The individual hardware accelerators can be made programmable to cover the variations across various standards, and also across different profiles within a given standard. Table 2 lists some of these differences in MPEG-4 Visual and H.264 standards.

A detailed study across the standards and profiles to be supported can help outline the programmability requirements of each of the HWAs. In cases where the functionality differs significantly (e.g. CAVLC based entropy coding in H.264 baseline profile vs CABAC based entropy coding in H.264 main

profile), it may be more power efficient to have multiple implementations as against one programmable engine.

Table 2: MPEG-4 Visual vs H.264

	MPEG-4 Visual	H.264
MC min block size	8x8	4x4
Motion vector accuracy	Half or quarter pixel	Quarter pixel
Transform	8x8 DCT	4x4 DCT approximation
Built-in deblocking filter	No	Yes
Intra prediction for I frame coding	No	Yes

While multiple standards and multiple formats introduce variability in compute and data bandwidth requirements, the characteristics of input data add to the variability in both encoding as well as the decoding process. Figure 3 shows some these factors leading to the variability in case of H.264 decoder. A similar set of factors can be identified for each of the standards/profiles.

Fetch input bit stream	Entropy decoder and reorder	Inverse quantization, IDCT Fetch reference frame data for MC	Motion compensation and add residues	Deblocking Filtering
Data rate, System traffic	Data per frame	#MVs, I/B/P frames, System traffic	#MV, MV resolution, I/B/P frames	Boundary Strength

Figure 3: Data driven variability in Decoding

3. Low Power Design

In a portable application, battery life is one of the key customer care-about, and hence the low power/energy focus needs to be an integral part of the development covering the entire spectrum of process technology, circuit design, logic design, micro-architecture, software and system. While using a low leakage CMOS process helps, leakage power can still be a significant portion of the total power, and hence the power optimization strategies need to comprehend both dynamic as well as leakage power.

Dynamic power is a function of switched capacitance, the supply voltage and the frequency. The leakage power is a function of the supply voltage, V_T of the transistors, and varies significantly with process and temperature variation.

The variability in the compute requirements can be used to “configure” the DV engine so as to reduce the power. The configuration is dynamic and applied at the application level, frame level and also at a macro-block level.

Here are the knobs which can be applied at an application/video stream level:

1. If decoding switch off encoding specific modules (e.g. motion estimation).
2. For a given standard and the profile, switch off modules catering to other standards and profiles. For example, if H.264 baseline profile, switch off CABAC functionality of the Entropy Decoder, if doing MPEG-4 encode, switch off intra-prediction module
3. Depending on the format being supported set the frequency and scale the voltage accordingly. Since D1 decode requires ~2.66 times lower compute than 720P decode, the engine can be operated at lower frequency while still meeting the 30 frames per second rate. DVFS (Dynamic Voltage and Frequency Scaling) can be used to scale down the voltage accordingly, and thus resulting in significant power reduction.

Here are some of the knobs which can be applied at a frame level:

1. Switch off unused logic depending on I vs P vs B frames
2. Switch off unused logic depending on interlaced vs progressive content

Here are some of the knobs which can be applied at a macro-block level:

1. Switch off individual hardware accelerators as soon as the computation for the current Macro-block is done (due to variability, the pipeline cannot be fully balanced)
2. During motion-compensation the compute requirements vary depending on 1 motion vector vs 4 motion vectors per macro block, they also vary depending on motion vector resolution in terms of pixel vs half pixel vs quarter pixel.
3. Switch off de-blocking filter, if boundary strength is 0 or there is significant change (gradient) across block boundary in the original image.

The process of “switching off” a module can be accomplished using clock and/or data gating, and also powering down the entire module. Dynamic power switching however has latencies associated with powering the hardware up and down, and also there is area overhead of voltage domain, power islands, power switches, isolation cells, retention flops and more. While powering down hardware helps reduce both dynamic as well as the leakage power, given the overhead, it is most practical when applied at an application level.

It can also be noted that since the clock and data gating mentioned above leverages the variability, the support for the gating needs to be designed in at an architecture level and comprehended while programming the DV engine.

As shown in figure 4, the leakage power varies significantly with process variation. While the leakage is significantly higher at the strong process corner, so is the performance of the transistor. Since the design is implemented to meet the desired performance at weak process corner, this additional performance at strong process corner can be traded-off against power. This is accomplished by lowering the supply voltage while maintaining the desired performance. Since the process corner varies with each device, the device needs to include process sensors and at a system level needs an intelligent power supply which can adapt the voltage as desired. It can be noted that AVS (adaptive voltage scaling) needs to be applied in conjunction with DVFS (Dynamic Voltage and Frequency Scaling) to get the most benefit. This however requires for a sophisticated power management unit on the application board.

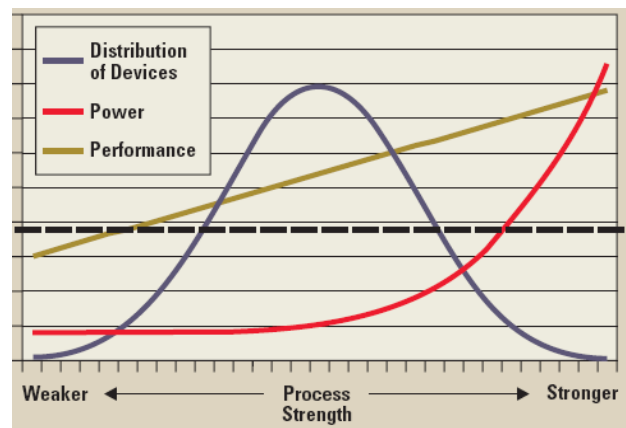


Figure 4: Device Variation over Process at Constant Voltage

Each process technology supports a range of operation for the core supply voltage – V_{min} below which the devices are not functional to V_{max} which cannot be exceeded to avoid reliability issues. Given that CIF resolution requires ~9 times lesser computation than 720P, the power reduction for CIF operation is in most cases limited by V_{min} . In other words, at V_{min} the DV engine can run at a frequency significantly higher than required to process CIF resolution. While dynamic power reduces by more than 9 times going from 720P to CIF, the leakage power reduction is marginal, and as a result at CIF resolution leakage power can potentially exceed the dynamic power of a DV engine. One approach to handle this is to dynamically switch off the DV engine. Given the overhead of switching, the power reduction can be maximized by decoding in a burst multiple frames, thus enabling the engine to be power downed over a longer stretch at a time. The second approach is to architect the hardware accelerators in a bit slice fashion, such that at CIF resolution, half the functional units can be switched off. This however can have software implications which need to be managed.

Various power management strategies such as adaptive voltage scaling, dynamic voltage and frequency scaling, dynamic power switching, multiple voltage/power domains etc. require Silicon IP and design flow support. These enabling technologies are captured in Table 3.

Table 3: Silicon IP support for Power Management

Technology	Description
<i>Retention SRAM and logic</i>	SRAM and logic retention cells support dynamic power switching without state loss, lowering voltage and reducing leakage.
<i>Dual-threshold voltages</i>	Higher threshold for lower leakage and lower threshold for higher performance.
<i>Power management cell library</i>	Switching, isolation and level shifters support multiple domains in SoC implementations.
<i>Process and temperature sensor</i>	Adapts voltage dynamically in response to silicon processes and temperature variations.
<i>Design flow support</i>	Complete, nonintrusive support for easily integrating SmartReflex technologies.

4. System Level Considerations

As mentioned earlier the customer requirements of lower cost and lower power/energy apply to the entire system as against the digital video sub-system.

4.1 DVFS at SoC Level

As discussed earlier, DVFS can be applied to leverage the compute variability to lower power of the DV engine. At an SoC level however there may be components which cannot be scaled in conjunction with the DV engine. For example, the processing of audio which accompanies the video, requires the same compute bandwidth independent of the video resolution. The audio DVFS for the audio processor thus needs to be done independent of the video engine.

While the supply voltage for the SRAM periphery logic can be scaled, the supply to the memory array cannot be scaled down in newer technology nodes. The memories thus have dual supply rails, and memory array power is supplied by an on-chip LDO which holds the array voltage constant independent of the voltage scaling for the logic.

The digital video system also involves processing the decoded video stream for display on an LCD panel or a TV. One of the steps in this post processing is resizing the video to match the resolution of the display. Any other computations done post resizing (e.g. rotation) are independent of the resolution of the video being decoded. Thus while DV engine voltage can be scaled with video resolution, the components of the display sub-system post resizing function cannot be scaled.

These considerations make the overall power management implementation at an SoC level complex. Multiple voltage and power domains also increase the complexity of the power management unit, and the cost increase due to additional regulator needs to be weighed against the relative power reduction.

4.2 Managing the data bandwidth

As a DV engine is scaled to handle higher resolution, the data bandwidth requirements go up as well. Thus if an SoC handling D1 resolution requires DDR running at 90MHz, at 720P, this will translate to DDR performance requirement of 240MHz. Such a scaling is not efficient in terms of both cost and power. One option is to have a wider interface (64 bit instead of 32bit) run at 120MHz, but this can increase the cost of the SoC and also

the power. Thus while scaling the DV engine, it's equally important to enhance the SoC architecture to handle the higher resolution without having to increase DDR performance proportionally. This can be done in multiple ways including adding on chip buffer memory, customized external memory interface which can efficiently handle 2D data transfer, adding hardware accelerators to enable on-the-fly computation etc.

In addition to SDRAM bandwidth consideration, SDRAM also contributes significantly to the system cost and power. The digital video sub-system architecture thus should aim at not only reducing SDRAM data rate but also the size. The SDRAM size requirements vary depending on the resolution and the standard. The system software can leverage this to switch off banks of the SDRAM and reduce system level power.

5. EDA Challenges

While design of a multi-standard, multi-format digital video engine can leverage the entire gamut of power management strategies, given the complexity and other system overheads, these need to be judiciously deployed. System level power modeling is an important enabler to drive these tradeoffs.

Comprehensive power management architecture of an SoC needs to provide for adaptive voltage scaling, dynamic voltage and frequency scaling, dynamic power switching, and partitioning the SoC into multiple voltage/power domains. This requires incorporating appropriate power management IPs including power switches, isolation cells, level shifters, retention flops, on-chip LDOs, process sensors etc. A design flow which supports synthesis and verification of a power management architecture is critical to ensure quality and also reduce the cycle time.

The power optimization needs to be supported at the physical design level as well. Some of the challenges include multi Vt optimization, timing closure across multiple voltage levels, physically aware low power synthesis and automated clock gating.

Finally, there is a role for ESL capability which supports designing of a modular and scalable DV engine generator which can be configured for a given functionality (standard/resolution) with the desired area-power tradeoff in a given technology node.

6. Summary

Portable video applications need a multi-format, multi-standard digital video encoding/decoding capability at low cost and low power. Given the computational complexity of HD video processing, a scalable, configurable digital video accelerator is required to meet the desired performance at low power. The variability in compute and data bandwidth requirements across standards and formats, can be leveraged to reduce power by employing multiple power management techniques including adaptive voltage scaling, dynamic voltage and frequency scaling and dynamic power switching. Since the cost and power goals apply to the entire solution, system level tradeoffs need to be considered while architecting the system-on-a-chip. EDA tools and flows have an important role to play in ensuring quality, reducing effort/cycle time and achieving desired power efficiency of the portable video processors.

7. REFERENCES

- [1] Iain E. G. Richardson, H.264 and MPEG-4 Video Compression: Video Coding for Next Generation Multimedia, Wiley, 2003
- [2] Jeremiah Golston, Ajir Rao, Video codecs tutorial: Trade-offs with H.264, VC-1 and other advanced codecs, White Paper, Texas Instruments, September 2006
- [3] Olli Silven, Tero Rintaluoma, Kari Jyrkkä, Implementing energy efficient embedded multimedia, Proceedings of SPIE - Volume 6074, Feb 10, 2006
- [4] Krisztián Flautner, David Flynn, Mark Rives, A Combined Hardware-Software Approach for Low-Power SoCs: Applying Adaptive Voltage Scaling and Intelligent Energy Management Software, DesignCon 2003
- [5] Gene Frantz, The Future of Digital Video, White Paper, Texas Instruments, September 2005
- [6] Gene Frantz, Leon Adams, DaVinci™ Technology for Digital Video, White Paper, Texas Instruments, September 2005
- [7] Brian Carlson, “SmartReflex™ power and performance management technologies,” White Paper, Texas Instruments, 2006
- [8] DSP Silicon Enhancements Use SmartReflex™ Technologies, Heinrich Hillmayr, White Paper, Texas Instruments, Jan 2007