

Media Processor Architecture for Video and Imaging on Camera Phones

Joseph Meehan, Youngjun Francis Yoo, Ching-Yu Hung, Mike Polley

Texas Instruments, Incorporated

ABSTRACT

A dedicated media processor is used in many camera phones to accelerate video and image processing. Increased demand for higher pixel resolution and higher quality image and video processing necessitates dramatically increased signal processing capability. To provide the increased performance at acceptable cost and power requires redesign of the traditional architecture. By wisely partitioning algorithms across programmable and fixed-function blocks, the performance goals can be met while still maintaining flexibility for new feature requirements and new standards. In this paper we provide an overview of media processor architectures for camera phones and describe the system architecture, power, and performance. We also address the challenges in supporting new imaging trends and high resolution video at low power and cost.

Index Terms— camera phone, media processor, architecture, imaging, video

1. INTRODUCTION

Explosive growth of multimedia-enabled cell phones and increased consumer expectation has fueled demand for higher performance video and image signal processing in these devices. On the other hand, slowly increasing battery capacity in portable electronics has tightly constrained power consumption, in spite of much faster increase in resolution and performance requirements with each generation. Cost, another primary concern, is strongly influenced by the economies of scale and massive unit volume and revenue at stake in this market. The intense competition has created cost pressure for semiconductor manufacturers to provide innovative solutions with superb performance, but at low enabling prices.

In this paper, we present coprocessor chips designed specifically for attachment to a host communications chip or other general purpose processor in a cell phone. We describe the video and imaging subsystem architecture and how to achieve the best balance between price, performance, power, and flexibility.

2. ARCHITECTURE PARTITION FOR MULTIMEDIA-ENABLED CELL PHONES

In some applications, integration of multiple chips into a single chip gives size, power, and cost reductions in new generations of products. The “single-chip cell phone” idea takes this integration to the extreme. However, for multimedia enabled cell phones, a two-chip solution enables a wider range of products to be more cost-effectively created. The baseband communications chip handles the wireless modem functions while a companion multimedia coprocessor chip implements the latest generation of

imaging, video, and other multimedia technology. This partition enables new phones to be quickly created that feature the newest camera/video features, performance, and standards compatibility, all without modifying the communications implementation.

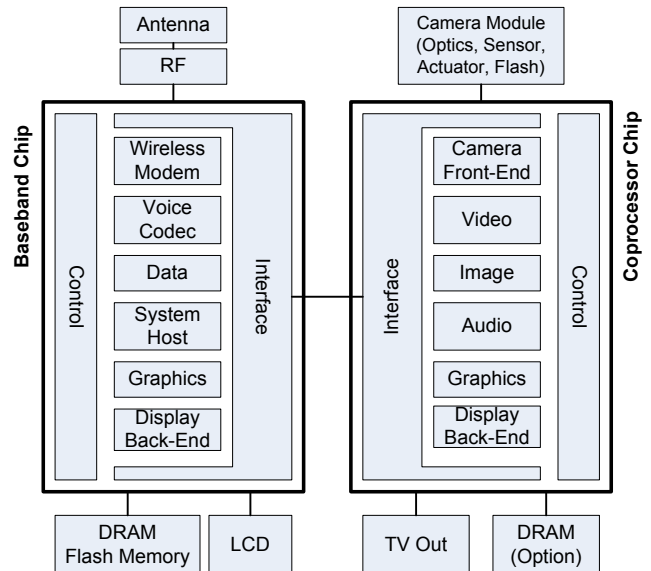


Figure 1 Functional partition between a baseband host chip and a camera/video coprocessor chip. LCD/TV connection is one example of many possible display connection options.

Figure 1 illustrates the major elements in a camera phone and the partition of functionality between a baseband chip and a camera/video chip. The baseband chip interfaces with the antenna and RF electronics and performs all of the communications processing, voice and data processing, system control, graphics, and display of system control on the LCD. The camera/video coprocessor chip interfaces with the optics and sensor and performs all of the image/video and audio processing. Additionally, it can process graphics, and optionally provide display back-end capability. Depending on the feature sets desired, the display backend electronics can be configured in a variety of ways to enable a range of product capability spanning LCD-only, all the way up to LCD plus concurrent HDTV output.

3. IMAGE SIGNAL PROCESSOR

An Image Signal Processor (ISP) refers to an embedded system that is responsible for generation of pictures by applying various

image processing steps and algorithms to raw image data from the image sensor. Under this broad definition, a typical ISP system consists of

- Hardware signal processor: hardwired processing unit, imaging accelerator, and/or general purpose processor
- 3A (automatic exposure, automatic white balance, and automatic focus) algorithms and application software
- Tuning of image processing parameters: sensor/optics calibration and imaging system characterization, and image processing parameter configuration

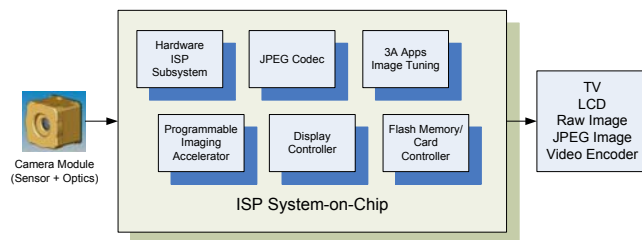


Figure 2 ISP as an SOC (system-on-chip)

A narrower definition of ISP could mean specifically the image processing accelerators in hardware implementation. Figures 2 and 3 illustrate examples of the ISP system of broad and narrow definitions respectively.

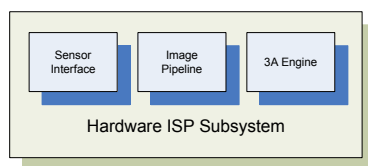


Figure 3 ISP as hardwired implementation of image processing algorithms

The last decade has seen rapid evolution of ISP in terms of both quality and feature capabilities as digital still camera (DSC) has become a mainstream consumer product. DSC and its ISP technology were already mature when camera phones were first introduced. However, due to many differences between two systems, camera phones were not able to take the advantage of the same ISP technologies. In the following are examples of differences that have been driving different requirements for camera phone ISP.

- Image sensors: While DSC's have been predominantly using CCD sensors, practically all camera phones use CMOS sensors. Although CMOS sensors have advantages in cost, power, and system integration, low signal quality requires the ISP to use more complex processing algorithms.
- Camera lens modules: Miniature camera phone lens modules use low quality optics and typically leave out the mechanical shutter and optical low-pass filter. Related ISP requirements are electronic rolling shutter support, compensation of aliasing and lens distortion (e.g., vignetting, colored shading, geometric distortion), and sharpness enhancement.
- Cost: Camera phones tend to have more strict system cost requirements. While the additional processing requirements can justify higher cost, the ISP cost increase should be limited such that the overall system cost can still benefit from the

low-cost sensor and optics components selected for the camera system.

- System constraints: Most camera phones either have no flash or have an ineffective flash and thus require an advanced ISP to avoid noisy low-light images. Large frame memory is often precluded in the camera subsystem of low-cost phones and the ISP should be able to process the sensor output in real-time. For example, a high pixel throughput of 70 megapixels/sec is needed for a 3-megapixel sensor.
- User scenarios and photo space: With camera phone, due to its easy access for the user, more pictures are taken indoors and often under extreme low-light situations. High ISO noise reduction and image stabilization are corresponding ISP requirements.

3.1. ISP Architecture

In this section we will discuss a camera phone ISP architecture and its major blocks using an example of Texas Instruments DM29x camera/video co-processor. The high-level chip architecture of TMS320DM29x is illustrated in Figure 4.

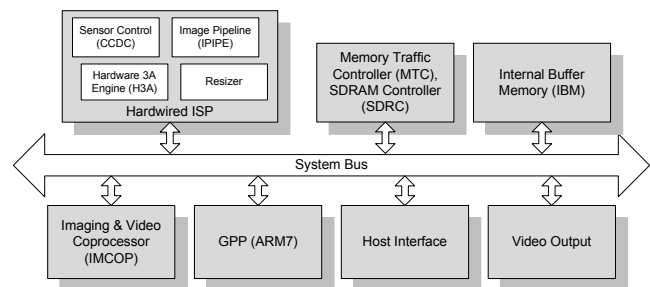


Figure 4 TMS320DM29x ISP chip architecture

Hardwired ISP: Camera interface and high-throughput hardwired image pipeline [1] are in this block. Also there are hardware 3A engine and image resizer functions. Although this block is implemented as hardwired logic gates, with carefully chosen algorithms and design techniques, a great degree of control flexibility and tuning options are provided.

General Purpose Processor (GPP): An ARM7-based GPP is available for system control and 3A applications [2][3]. It programs all other subsystem modules. It also takes the camera/video task commands from the system host (see Figure 1), controls the camera module, and sets up API's for camera/video processing tasks.

Imaging and Video Coprocessor (IMCOP): This programmable coprocessor includes the Imaging Extension (iMX) engine, a 4-MAC SIMD accelerator designed specifically for efficient image processing tasks. The IMCOP also includes accelerators for JPEG compression and MPEG-1, MPEG-4, and H.263 video compression.

Video Output: For the preview or video mode, DM29x has a high-speed serial port as well as a video parallel interface to support transmission of ISP-generated VGA frames to the system host at 30 fps.

SDRAM Controller (SDRC): Through the SDRC, DM29x interfaces to an SDRAM which is used to buffer the sensor raw image before ISP processing and the YUV output image from the ISP. iMX-based custom processing can be applied to buffered images. SDRAM image buffers, together with iMX

programmability, opens up various possibilities for camera feature/quality differentiation.

Internal Buffer Memory (IBM): To support camera systems without an SDRAM, and to reduce SDRAM traffic and thus power when an SDRAM is used, the DM29x features a relatively small but fast on-chip memory (IBM).

Note that, in addition to the major building blocks in the figure, DM29x has common peripheral options such as host interface, general input/output (GIO) ports, pulse width modulation (PWM) control, and NAND Flash interface in order to provide a complete ISP SOC.

The architecture of the TMS320DM29x can be compared with an early ISP architecture where a single TMS320C549 DSP and a camera interface companion chip were used to provide camera functionalities for DSC systems of much lower requirements on the image sensor resolution, the speed performance, and the feature set [4]. DM29x ISP is architected to achieve a balance among performance, power, area, and flexibility by combining hardwired core imaging functions, a programmable coprocessor, and a GPP. In a rough estimation, for typical image pipeline algorithms, DM29x iMX offers 4x more performance of TMS320C549, while the DM29x Hardware ISP is more than 200x efficient than TMS320C54x for the chosen image pipeline algorithms.

3.2. ISP Challenges and Trends

Today the image quality continues to be the main challenge of camera phones. The trends of increasing sensor resolution and shrinking camera module require each new ISP generation to have more advanced image processing at a higher speed. Additional challenges come from new camera feature requirements such as image/video stabilization, red eye reduction, lens distortion correction, advanced noise filter, adaptive light compensation, etc. Unless innovative algorithms and architectures are introduced in a timely manner, these challenges will adversely affect the performance, power, and cost of new ISP's beyond the point that silicon technology advancement can offset.

Following the trend of single-chip integration, the ISP has also been integrated into the phone system host chip recently. A major drawback in this case is that the system host chip may integrate outdated ISP technologies since the development time and the product life cycle of the host chip are usually longer than those of ISP and other camera system components. A different integration option is the CMOS sensor. This approach makes sense for low-end camera phones where the cost is much more important than the image quality or camera features. Thus a sensor-specific hardwired ISP is used for integration with CMOS sensor, limiting camera differentiation possibilities. Also it is difficult to achieve quality/feature consistency for a camera phone design that uses more than one CMOS sensor design in a single product line, which is not uncommon for high-volume camera phones.

Thus a dedicated ISP chip remains a competitive and attractive option for many camera phone system designs. Also it is very important to have programmability in the ISP as new sensor and other imaging system components can present quality challenges that are unexpected during initial design. An ISP with programmable flexibility can significantly reduce the risk associated with camera phone development and help the overall

system cost as well by providing flexibility in, for instance, sensor and optics selection.

4. VIDEO SYSTEM ARCHITECTURE

4.1. Mobile Video Trends and Requirements

There have been four main mobile video trends over the last few years: increasing complexity of video codecs, increasing number of video codecs to be supported, increasing camera pixel resolution, and increasing screen resolution.

At the start of the Texas Instruments OMAP1, there was one main video codec to be supported and that was H.263. Since then, the mobile domain has seen the arrival of MPEG2, MPEG4, H.264, VC-1, RV, DivX, VP6/7, Sorenson Spark and AVS 1.0. With the increasing number of codecs to be supported has come an increase in MHz complexity. For example, H.264 Baseline Profile (BP) decode is approximately 2.5x more complex than H.263 Profile 0 decode for the same resolution and frame rate. There has also been an increase in video quality requirements and expectations on the encoder side and with this has come an increase in encoder complexity. For example, the motion estimation algorithms have increased in complexity.

In the short history of mobile video, one point that has been seen on all OMAP generations is the request to support video codecs that were not available when the OMAP chip was designed. OMAP1 has seen many requests to support H.264 which was not standardized when OMAP1 was developed. The same goes for OMAP2, which has seen many requests to support ON2 VP6/7. The same will be expected for OMAP3 and DM29x.

At the start of the mobile video revolution, QCIF was seen as a good starting point for video content on the cellular networks. Newer phones are now promoting DVD quality video. Over this time, the mobile phone has gone from H.263 Profile 0 QCIF 15fps to H.264 BP D1 30fps, which is an increase in complexity of approximately 65x. This is a large increase in complexity and with this has come major changes in video system architecture.

4.2. Video System Trade-offs

The biggest trade-off in video hardware systems are flexibility/programmability versus hardwired. Adding flexibility allows new features (profiles/codecs) to be added after hardware is developed, allows third party developers to differentiate their codecs (in performance and/or quality) and also allows software codec performance to improve over time as the codec matures. The negative side is that this flexibility comes at a price of increased power and area and in some cases performance.

4.3. Video System Options

At one end, the programmable Digital Signal Processor (DSP) provides full flexibility and programmability and the hardware only solution has no flexibility or programmability. There is a possibility to trade-off between these two extremes where there can be a mix of DSP and hardware accelerators. This has certain system issues as the partitioning between hardware accelerators (HWA's) and DSP has to be chosen carefully. Otherwise, the overhead of transferring data between HWA's and DSP can mitigate the performance gain of the HWA's.

Based on this consideration, a good partitioning is to choose hardware accelerators that do large chunks of video processing. By taking this functional approach to video hardware accelerators, frequency of hardware/ software interactions is reduced and good level of efficiency and concurrency is sustained. Managing data traffic in larger blocks also optimizes the system/SDRAM bandwidth consumed by video.

An example is the Motion Estimation HWA. The HWA is provided the pointers to the current macroblock (MB) and the reference frame. The HWA outputs the minimum sum-of-absolute-difference (SAD), the MB residual and the motion vector (MV). It would be similar for the Loop Filter HWA; input would be the current MB and output would be the filtered MB. Entropy decoder would be the same; input would be the bit stream and output would be the MB and the MB information. Please see, for example, [5] for more in-depth discussions of various algorithm stages in video coding and challenges in DSP implementation.

This often leaves the problem that the only tasks left for video processing on the DSP are control code, which the DSP was not designed for and such workload is not optimal on a DSP. In fact, a general-purpose processor seems just as well-suited for such workload.

The following table outlines the Power/Performance/Area (PPA) of different video system options.

	DSP only	DSP+HWA	HWA only
Area	1.0x	1.2x	0.5x
Codec Performance	1.0x	2.0x	6.0x
Power	1.0x	0.8x	0.5x
Flexibility	1.0x	0.5x	0.0x

Table 1: Video System Power/Performance/Area Ratio Summary Based on Observation of Implementations

The DSP only option is the point of reference to which power, performance and area of the other options are compared. As can be seen from the table, the HWA only solution is smaller than a DSP only solution, and the DSP+HWA solution is between the two. The codec performance improvement from a DSP to a HWA is normally in the range of 2x to 6x. Normally for the same resolution and frame rate, the power of a HWA only solution would be approximately half that of a DSP only solution. A HWA only solution would have zero flexibility compared to a DSP only solution. No new codecs could be added and only one codec implementation would be possible and is fixed for the lifetime of the product.

Roughly, a high performance 300MHz DSP is achieving H.264 BP D1 30fps decode performance and H.264 BP HVGA 30fps encode performance. A similar sized 300MHz high performance set of hardware accelerators should be achieving H.264 High Profile 1080p 30fps encode and decode performance.

By having a DSP, the lifetime of the product can be extended as compared to a full hardware solution. New codecs can be added when required. As software is optimized, codec performance improves and hence increases the lifetime of the product. Since HWAs are not likely to handle new codecs, DSP eventually gets a more DSP-friendly workload, and proves to be the right platform.

In the wireless domain, error resilience, error concealment and error detection are important features that have to be supported. For H.264 BP in particular, these can be difficult to support in hardware as normally they imply a different control

flow for when arbitrary slice ordering (ASO)/flexible macroblock ordering (FMO) are used or not used. This can lead to extra difficulties if the video architecture is not programmable.

Post processing (de-blocking or error concealment) is also a very important factor of a video decoder and is very difficult to implement in hardware, as algorithms for these tasks are very customer or application specific and thus requires programmability.

While in this paper, the DSP is shown as being the flexible solution, there are other flexible solutions such as RISC processors, vector processors, etc.

4.4. Video System Conclusions

Combining DSP and hardware accelerated video functions in a video system gives the best trade-off between flexibility for future codec standards, programmability for third party differentiation, power, and performance. By providing hardware optimized and flexible processing in one package, the product life can be increased.

Future video hardware systems should include DSP and HWA's or programmable HWA's for optimum performance and flexibility.

5. SUMMARY

Rapidly increasing demand for performance, expanding functionality, evolving algorithms, and stringent area and power budgets are common trends in image and video processing for camera phone applications. To meet these challenges, we architected a combination of hardwired engines and programmable DSP's in Texas Instruments' camera phone coprocessors. Such an architecture strikes a good balance of area and power efficiency for known algorithms, flexibility in high-level control algorithms, and programmable performance to address algorithm needs throughout the product cycle.

6. REFERENCES

- [1] Ramanath, R., W.E. Snyder, Y. Yoo, and M.S. Drew, "Color Image Processing Pipeline in Digital Color Cameras," *IEEE Signal Process. Magazine; Special Issue on Color Image Process*, pp.34-43, Jan. 2005
- [2] Oh, H.-J., N. Kehtarnavaz, Y.F. Yoo, S. Reis, and R. Talluri, "Real-Time Implementation of Auto White Balancing and Auto Exposure on The TI DSC Platform," in *Proc. of SPIE Real-Time Imaging VI, Vol.4666*, pp.52-26, San Jose, CA, 2002
- [3] Gamadia, M., V. Peddigari, N. Kehtarnavaz, S.-Y. Lee, and G. Cook, "Real-Time Implementation of Autofocus on The TI DSC Processor," in *Proc. of SPIE Real-Time Imaging VIII, Vol.5297*, pp. 10-18, San Jose, CA, 2004
- [4] Illgner, K., H.-G. Gruber, P. Gelabert, J. Liang, Y. Yoo, W.Rabadi, and R. Talluri, "Programmable DSP Platform for Digital Still Cameras," in *Proc. of ICASSP 99*, Phoenix, AZ, 1999
- [5] Budagavi, M., W. R. Heinzelman, J. Webb., and R. Talluri, "Wireless MPEG-4 Video Communication on DSP Chips," *IEEE Signal Processing Magazine, Vol.17*, pp.36-53, Jan. 2000