

Accelerating Mobile Multimedia with the Intel® PXA27x Processor Family

N.C. Paver, M.H. Khan, B.C. Aldrich,
Intel Corporation, 1501 S. Mopac, Suite 400, Austin, Texas, USA, 78746

nigel.paver@intel.com

Abstract - Demand for mobile video applications is growing today in wireless handheld platforms. The Intel® PXA27x processor family has been designed to accelerate mobile multimedia and applications processing in a power efficient manner. The PXA27x processor is a highly integrated system on a chip including the Intel XScale® Microarchitecture with 64-bit Intel® Wireless MMX™ technology, 256KBytes of on-chip SRAM, multimedia interfaces and a host of serial ports. This paper provides an overview of the PXA27X processor architecture, machine organization and functional units. It also provides analysis of the key features of the architecture that specifically enhance the multi-media performance.

I. INTRODUCTION

Mobile multimedia is growing at a startling rate. This is fueling the trend toward rich multimedia and communications capabilities on mobile devices. End users in the handheld wireless market segment are demanding multimedia and communication experiences similar to those they enjoy on their desktop—but in a mobile setting. Video playback, multi-player gaming, and video conferencing are a few of the key applications driving the path to higher performance multimedia. The availability of more incoming multimedia data via wireless networks, camera sensors, and audio capture is feeding these ever hungry multimedia applications

One of the biggest challenges for multimedia on mobile devices is to provide high performance with low power consumption. Playing a richly detailed 3D game on a phone or Personal Digital Assistant (PDA) can be highly enjoyable until the phone or PDA runs out of power. This paper, presents an architectural overview of Intel PXA27x family of processors which are designed to address this need.

II. CONVERGENCE

As mobile users demand devices with more data functionality, such as access to services and applications on their wireless clients, the role of today's handheld clients

and services are under going profound change. Historically mobile phones were primarily a method of transferring voice traffic to the intended recipient. With the emergence of new cellular radio standards, the transmission of packetized data is also possible over the cellular network. Mobile phones are evolving from voice-centric devices to voice-plus-data devices. With the arrival of data, especially multimedia data, the application processing power of phones is increasing.

At the other end of the convergence spectrum are personal digital assistants (PDA). Here the focus is providing the end user with a rich set of application services such as video and gaming. As the growth of the Internet has spread onto handheld devices, these products are now integrating wireless radio communications. In this case the data-centric device is adding voice and radio technology.

III. TECHNOLOGY DRIVERS

The growing demand for multimedia processing on the converged platforms is driven by two primary factors. The first is the growing capability and resolution of the display devices. The second factor is the increasing supply of multimedia data arriving over the network and through onboard sensors such as cameras.

Cellular phone handsets in the past had very restricted display capabilities. This was limited to a few lines of monochrome text on a small LCD panel. The recent evolution in both display technology and available computing power is producing more advanced products with higher resolution displays. Figure 1 shows that the trend towards increasing resolution follows two tracks, depending on the physical size, or form factor of the product. The PDA form factor has historically been physically larger than a phone so has supported bigger display resolutions. Today quarter VGA (QVGA) displays (320×240 pixels) are common with VGA displays (640×480 pixels) emerging.

In the smaller physical form factor of the phone handset the common display resolution is around 176×144 size, with a trend towards QVGA (and ultimately VGA) as the data processing capabilities of the phone increase.

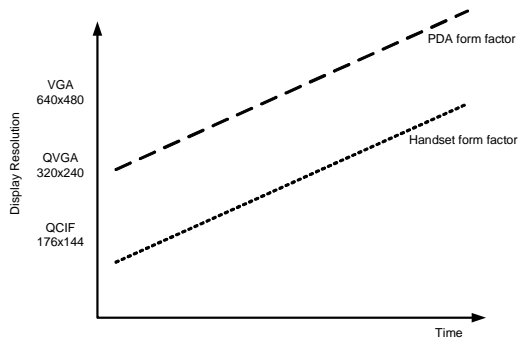


Fig. 1. Display resolution evolution

As the numbers of pixels in the display increases so does the processing power needed to calculate each pixel value. For example a VGA display typically takes four times the computation that a quarter VGA does to generate the content.

The early cellular standards were constrained to limited spectrum bandwidth and so could not provide substantial data transport capability. However, with the introduction of next generation (2.5G) cellular radio standards such as General Packet Radio service (GPRS) and the Enhanced Data rates for Global Evolution (EDGE), the amount of data available increases up to 144 kilobits per second. This is a substantial improvement over the original 9.6–14 kilobits per second of the 2G standards. This evolution is set to continue with the introduction of 3G standards such as WCDMA and UMTS where data rates of 384 kilobits per second are possible and eventually reaching 2.048 megabits per second. Figure 2 shows how the data transmission rates are increasing.

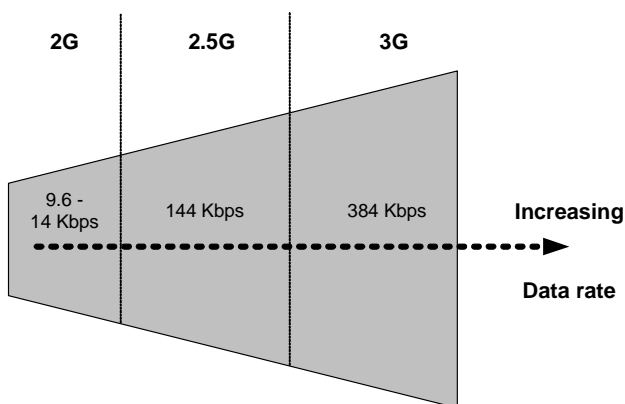


Fig. 2. Evolution of cellular data rates

The other primary source of multimedia content on handheld devices is from onboard interfaces such as camera interfaces, audio devices, and content stored on removable media such as flash memory cards. The resolution of camera sensors in handheld platforms is evolving rapidly from VGA still image capture to megapixel resolutions more typically seen in digital cameras. The resolution and frame rate for

video captured from the same sensors is following a similar growth path; today video can be encoded at QCIF (176x144) with QGVA and VGA resolutions on the roadmap for many product lines. The more pixels in an image or image sequence, the more computation power is required to process them.

The capacity of removable storage media is increasing today. 512-megabyte cards are available today and 1 gigabyte and beyond are being introduced. With this amount of storage available it becomes possible to store many hours of VGA resolution video content. This is driving the need for an increase in both the computing power to process the VGA content and the power efficiency so that many hours of video can be viewed from a single battery charge. The increased capacity of the storage cards is also being utilized for audio content where many hundred of MP3 tracks may be stored. In this case power efficiency becomes critical to ensure playback time meets user expectations

IV. PXA27x PROCESSOR FAMILY

In wireless platforms area, performance, power and cost and key metrics for product success. This is driving increasing levels of on-chip integration in state-of-the-art application processors. The Intel® PXA27x processor family is a highly integrated System-on-a-Chip (SoC) targeting wireless and handheld platforms. Figure 3 shows an overall block diagram of the Intel® PXA270 processor and shows the Intel® XScale® microarchitecture and Intel® Wireless MMX™ technology as key features. It also includes 256KB of SRAM memories which is useful in video and graphics applications as a frame buffer. The PXA27x processor also provides multimedia components such as an LCD controller, camera interface, and an extensive set of peripheral interfaces such as UART, USB, AC97, SSP and I2S. The memory subsystem of the XScale core contains 32 KB caches for both instruction and data. The PXA27x also supports a wide range of flash memory card interfaces for program and data storage.

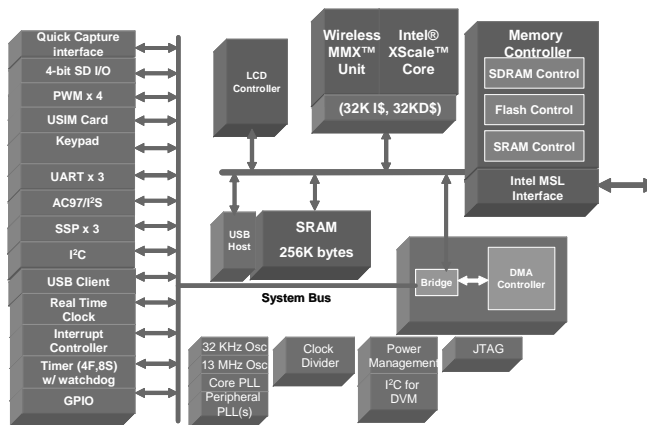


Fig. 3. Intel PXA270 Processor Block Diagram

V. INTEL XSCALE® MICROARCHITECTURE

The Intel XScale microarchitecture [11] is an implementation of the ARM® V5TE architecture [13][14]. The XScale core supports both dynamic frequency and voltage scaling with a maximum frequency today of 624MHz. The design is a scalar, in-order single issue architecture with concurrent execution in 3 pipes that support out-of-order return. To support the frequency targets a 7-stage integer pipeline is employed with dynamic branch prediction supplied to mitigate the cost of a deeper pipeline. The memory subsystem contains 32KByte instruction and data caches with corresponding 32 entry I translation look-aside buffer (TLB) and 32 entry D TLB. The memory subsystem also contains an eight entry write buffer that supports write coalescing and a four entry fill buffer to support multiple outstanding load operations. Figure 4 also shows the core also supports a debug interface and JTAG in addition to a high-speed interface to the Wireless MMX unit.

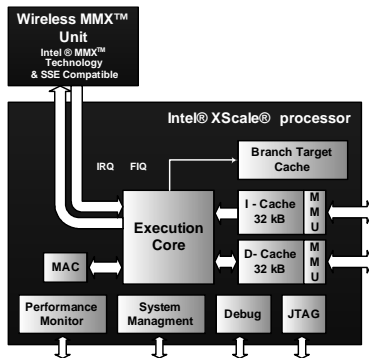


Fig. 4. Intel XScale Microarchitecture

VI. WIRELESS MMX ARCHITECTURE

Significant research effort and desktop processor development has been under-taken related to SIMD processing for media applications [1][2][3][4]. Wireless MMX technology [6][7][8][9][10] integrates equivalent functionality to all of Intel® MMX™ technology [5] and the integer functions from SSE [2] to the Intel® XScale® microarchitecture [11]. Like MMX technology and SSE, Wireless MMX technology utilizes 64-bit wide SIMD instructions, which allows it to concurrently process up to eight data elements in a single cycle. This style of programming is well known to existing software developers.

Wireless MMX technology defines three packed data types (8-bit byte, 16-bit half word and 32-bit word) and the 64-bit double word. The elements in these packed data types may be represented as signed or unsigned fixed point integers. Using special SIMD instructions it is possible to operate on

*. Other names and brands may be claimed as the property of others

data elements in the packed format, where each data element is treated as an independent item.

The Wireless MMX unit is a tightly coupled coprocessor of the XScale microarchitecture. The programmer's model is an extension of the XScale microarchitecture programming model. Thus, a multimedia application can maintain a single thread of control while taking advantage of the SIMD acceleration only on the critical section of the algorithm. As Wireless MMX is an extension of the XScale microarchitecture it takes advantage of the existing memory subsystem for the XScale microarchitecture without the need for extra dedicated memories and the power consumption associated with them. The power efficiency is further improved by using advanced power management, where the Wireless MMX unit is only activated, when required, on an instruction by instruction basis.

Wireless MMX technology comprises five key functional units to implement the programmers model. Figure 5 shows the organization of the functional units within the coprocessor.

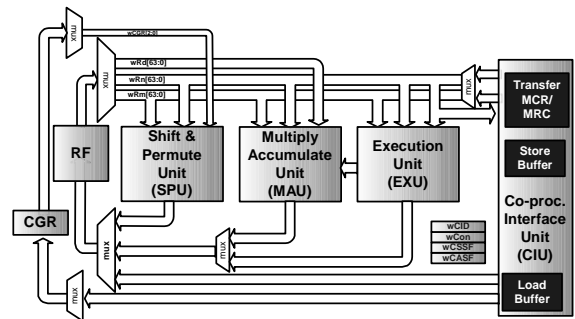


Fig. 5. Wireless MMX Micro-Architecture

The Shift and Permute Unit is responsible for performing shift and permute operations. These operations include the alignment, shift, rotation, packing and shuffling.

The Execute Unit is responsible for performing arithmetic and logic operations and it also provides a saturation capability. Operands may be received from the SPU when saturation is required and the CIU when the transfer instructions are issued.

The Multiply and Accumulator Unit is responsible for performing all multiply and accumulate operations. Operands are also received from the EXU when executing the sum of absolute difference instruction. The MAU unit is a three stage pipeline with internal accumulator forwarding.

The Coprocessor Interface Unit transfers data between the Wireless MMX™ unit registers and the Intel® XScale® microarchitecture. In addition to supporting coprocessor data transfer, it is also responsible for storing and loading data to and from the memory.

The main Register File (RF) is organized as sixteen 64-bit registers, located in the coprocessor 0 space (CP0). The

large register file allows the Wireless MMX instructions to support an increased number of intermediate values in complex calculations. For example, multiple output samples of a filter may be calculated in parallel or a single pass, half pixel motion search can be implemented (all eight intermediate block comparison results are calculated concurrently). The increased storage allows the programmer to take advantage of the spatial and temporal data locality as found in many multi-media applications. This reduces the required load store bandwidth and improves processing efficiency. Alignment support instructions also increase the effectiveness of this data re-use. These combined techniques are referred to as *Multi-Sample Technology*.

The control and status registers are mapped into coprocessor 1 space (CP1). There are four 32-bit general-purpose control registers used for alignment and shift control. As the shift and alignment offset is usually invariant across the inner loops the registers are designed to hold constant values. This saves the use of a packed data registers.

There are also a number of control and status registers also mapped onto coprocessor 1 space:

wCASFSIMD Arithmetic Status Flags- A set of four flags for each byte/half-word/word/double-word operation N when the result is negative C when there is a carry out Z if the result is zero V if the result over-flowed

wCSSFSIMD Saturation Flags which set the respective flag if an operation on a particular element saturates

wCON coprocessor control register—support for reducing memory traffic on a context switch.

A. Pipeline Structure

To achieve the same clock speed as the Intel XScale core, the Wireless MMX unit employs the same pipeline structure, as shown below in figure 6. The coprocessor is a five-stage pipeline with two extra stages of instruction fetch provided by the main core. Wireless MMX pipeline operates in lock step with the main core pipeline, providing a single thread of control and no complex synchronizations required between the two pipelines.

The XScale microarchitecture with Wireless MMX technology is a single issue machine. An instruction can be issued to the main core pipeline or coprocessor pipeline. The architecture allows instructions to be retired out of order. The pipeline organization also supports multiple outstanding loads, which improves memory throughput. This feature combined with the out-of-order completion, allows non-dependent instructions to execute, reducing the impact of memory latency in system on a chip applications.

B. Instruction set Overview

Wireless MMX technology provides a rich set of instructions that perform parallel operations on multiple data elements packed into 64-bit words. In addition, backwards

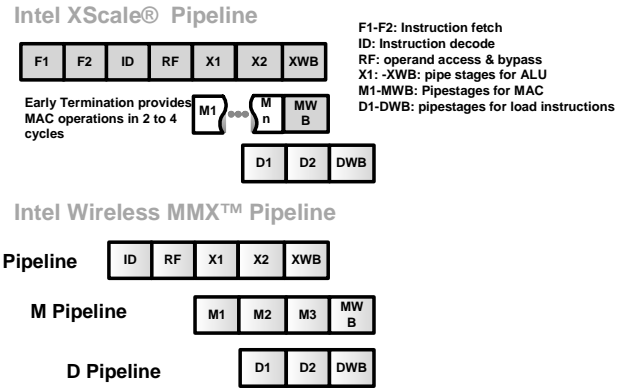


Fig. 6. Pipeline Structure

compatibility is provided with existing XScale microarchitecture coprocessor 0 instructions, which operate on XScale core registers (TMIA, TMIAxy, TMIAPH).

In total there are 43 new instructions in the architecture. Table 1 provides an overview of the instruction set.

Table 1: Key Instruction Overview

Instruction	Description
WACC	Addition of all 8xbytes, 4xhalf words, or 2xwords in 1 reg
WADD/WSUB	Add/Subtract 8xbytes, 4xhalf words, or 2xwords
WALIGN	Extracts 64-bit value on byte boundaries from 2x64 bit.
WAND/WOR/WXOR	64-bit logical operations
WAVG2	Unsigned average on vectors of 8- or 16-bit data
WCMPEQ/WCMPGT	Compare 8x bytes, 4xhalf words, or 2x word elements in parallel. Result is mask of 1's if true or 0's if false
WMAC	Multiply four signed or unsigned 16-bit half words in parallel and accumulate with a 64-bit register.
WMAX/WMIN	Vector maximum/minimum selection
WMADD	Multiply four 16-bit words in parallel and add
WMUL	Multiply four signed 16-bit words in parallel. Low- or high-order 16 bits of 32-bit result are produced
WROR/WSRA/WSLL/WSRL	Rotate right, shift arithmetic/logical right, left shift of 4 half words, 2 words, or 64-bit double word, in parallel
WPACK	Pack double word to words or words to bytes
WSAD	Sum of absolute differences on 8xByte or 4x16-bit data.
WSHUFH	Shuffles 16-bit data specified by an 8-bit immediate
WUNPCK	Unpack 8xbytes, 4x16-bit half words, or 2x32-bit words
WLDR/WSTR	Load/Store Byte, half word, word or double word
TANDC/TORC	Logical operations across the fields of the SIMD PSR (wCASFS) and sends the result to the XScale core CPSR
TBCST	Broadcasts a value from the XScale core source register to every element in the packed destination register
TMIA	32x32 signed multiply-accumulate using operands from the two source XScale core registers

Table 1: Key Instruction Overview

Instruction	Description
TMIAXy	16x16-bit multiply-accumulate selecting high/low 16-bits from two source XScale core registers
TMIAPH	Dual 16x16 multiply accumulate into 64-bit using signed 16-bit operands from the two source XScale core registers

VII. ARCHITECTURAL SUPPORT FOR VIDEO

To enable the efficient transmission of video data compression techniques are employed to reduce the amount of data that needs to be transferred. To ensure the inter-operability of video equipment, standards such as MPEG-4[16] and H.263[19] have been defined. While the detail of each compression algorithm changes from one standard to the next, the fundamental building blocks have many similarities. It is these inner loop building blocks that Wireless MMX technology is designed to accelerate. The programmable nature of Wireless MMX technology allows the inner loops to be adapted to the specific standard and track any future changes by modifying just the software.

A. Instruction support for Video

While generic SIMD operations such as addition (WADD) and subtraction (WSUB) provide the basic building blocks for algorithm development, some specific instructions have been added to accelerate various aspects of video encoding and decoding.

For motion compensation, when the motion vector indicates a non-integer displacement, the spatial interpolation of the pixels is done in three directions, horizontal, vertical, and diagonal, (1/2X, 1/2Y, and 1/2XY). The two way average (WAVG2) instruction is provided to accelerate this function. The instruction allows up to 8 pixels to be processed in parallel and is particularly useful in motion compensation routines. Figure 7 shows the half-word version of this instruction where 4 half-word pixels can be processed. The instruction also provides the capability for optional biased rounding which is useful for the different rounding specifications of

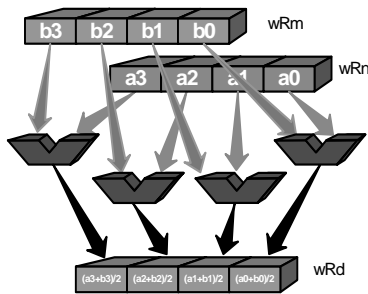


Fig. 7. Two Way Average Instruction

each standard.

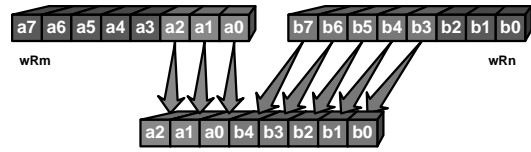


Fig. 8. WALIGN: Dealing with Unaligned data

For horizontal interpolation, WALIGN and WAVG2 can be used together effectively. WALIGN instruction extracts a 64-bit value from two source operands at any byte boundary, as shown below in Figure 8. The alignment offset can be specified as either an immediate value encoded in the instruction word or by using a dynamic value store in a register (usually a function of a data address pointer). In video, this is useful for dealing with unaligned macroblocks. It is particularly effective when multiple different alignments can be extracted from the same source data, for example in final half-pixel motion search

For video encoding process, motion estimation and motion search algorithms can consume as much as 40% of the encode operation [12]. The sum of absolute difference instruction (WSADB) instruction is designed to accelerate the motion estimation part of a video encoder. The instruction, shown below in Figure 9 can perform 8 absolute difference calculations in parallel, accumulate all of the results and keep a running total.

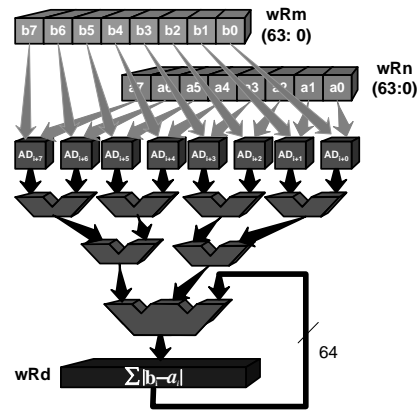


Fig. 9. Sum of Absolute Difference Instruction

This allows a whole line of an 8x8 macroblock to be compared to a corresponding candidate 8x8 macroblock line in a single instruction

B. Data organization for Video

Video data processing operations have a certain regularity in terms of the data accesses. For example, many operations are often limited to a block size. Larger register file of Wireless MMX can be used as a level-0 cache and data local algorithms can be further accelerated. With sixteen 64-bit

registers an entire 8x8 macroblock can be store in eight registers, as shown below in Figure 10, with eight registers still being available for intermediate data calculation and storage. This is particularly useful for the motion search used in video encode as it enables the candidate 8x8 macroblock to be store in the register file and reused for the comparisons at each search position. This dramatically reduces the required load store bandwidth of the search algorithm, with the corresponding reduction in power consumption.

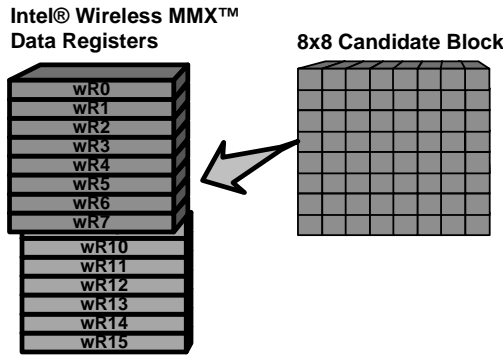


Fig. 10. Caching the Candidate block in the register file

Another method of utilizing the large register files is to concurrently compute a number of different macroblock comparisons in parallel and keep all the running totals in different registers. For example, with Wireless MMX it is possible to perform a single pass half-pixel motion search, by calculating the 8 different half-pixel macroblock SAD calculations and keeping all the intermediate results active. At the end of the single pass through the data, the lowest SAD value is selected as the best match. This technique makes maximum use of the loaded data and again contributes to reducing the power consumption by minimizing memory accesses. It is also possible to use a combination of macroblock storage and multiple output calculations to further improve video processing efficiency.

More detailed information about Wireless MMX technology and optimization techniques is described in detail in an Intel Press Book [9]

VIII. ARCHITECTURAL SUPPORT FOR GRAPHICS

The Intel® PXA27x processor has several architectural features which improve performance and power efficiency for graphics applications. These features include:

Instruction support: Specialized Wireless MMX instruction support such as WMADD – Multiply and add assists bilinear interpolation etc. Similarly, WALIGN (Align instructions) handle pixels if they are not aligned conveniently. Following the ARM architectural concept, all the Intel Wireless MMX Technology instructions are conditionally executed. Conditional execution assists efficient execution of tight loops.

Register cache: Wireless MMX Technology has large register file (16 64-bit registers). Register file can be effectively used as L0 memory and this feature reduces number of memory load and stores.

Data prefetch: The XScale core and Wireless MMX unit support data prefetching into the 32KByte data caches and allows multiple outstanding loads operations to mitigate the effects of memory latencies

At the PXA27x processor system level, architectural features such as the on-chip SRAM can be used for applications and also as LCD frame-buffer, enhances power and performance for gaming and graphics.

IX. MULTI-MEDIA INTERFACES

The ability to send, receive, and capture digital images and video has been one of the more important developments in the cell phone and PDA market segment in recent years. Management of data streams to image display and from capture resources becomes a necessary and critical aspect of the system design. Any inefficiencies when dealing with video data streams have a direct impact on the user experience. The effect is often manifested with reduction in battery life and decreases in video frame rates and resolution. In order to address these issues, key multimedia features have been introduced with the PXA27x processor family. The features have been integrated with the multimedia interfaces used for image display and image capture.

There are a number of critical flows which target the LCD panel in handheld devices. These include the display of video data following decode for playback mode, the simultaneous display of two streams of video data used in a video conferencing mode, and also the display of a digital viewfinder stream when performing camera or camcorder functions. Figure 11 illustrates some possible multimedia streams.

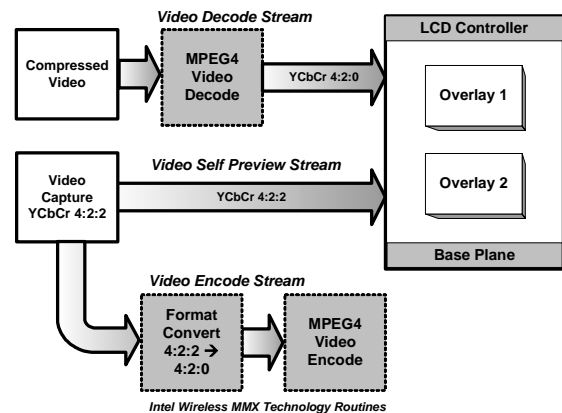


Fig. 11. Possible video imaging scenarios.

The PXA27x processor family introduces both flexibility and coherency in the image display and capture peripherals. The quick capture interface provides the flexibility to con-

nect to a wide variety of CMOS sensors using several possible interface options. In addition, image data formatting using Wireless MMX technology facilitates efficient processing and display using the LCD controller. The LCD controller also provides the ability to connect to a wide variety of LCD panels using a number of programmable options. The image data may be received in several different formats including various flavors of RGB and YCbCr.

A. LCD Controller

The LCD controller provides an interface between the PXA27x processor and a flat-panel display module. The configuration of the controller is established through programmable options for display type, resolution, external frame buffer, pixel depth, overlays, hardware cursor, and output data formatting. The block diagram for the LCD controller is shown in Figure 12.

The display type can be passive (DSTN), active (TFT), or an LCD panel with internal frame buffering. The display size can be up to 800 x 600 pixels and may be single or dual scan. Pixel depths of 2, 4, 8, 16, 18, 19, 24 and 25 bits per pixel (bpp) are used with RGB and RGBT formats. The RGBT format uses the most significant bit to indicate transparency for overlay support. For bit depths less than 8 bpp, there are three separate 256x256-bit palette RAMs that can be used to map the 2,4, or 8bpp values to 16- or 25-bit values.

A seven channel dedicated DMA engine supports memory accesses. One channel for the base plane, one channel for Overlay1, three channels for Overlay 2, one channel for the hardware cursor, and one channel for command data. The combination of the three image planes and cursor allows multiple images to be displayed simultaneously with software control of window size and position.

The data for the Base, Overlay 1, Overlay 2, and cursor are combined to go through either the dither engine, for passive displays, or directly to the output FIFO for active displays. The data output from the dither logic is grouped into the selected format and placed in the output FIFO.

The LCD controller is designed to work efficiently with both video playback and video preview from a CMOS or CCD image sensor by supporting the following three YCbCr video formats:

- YCbCr 4:4:4 sampling format
- YCbCr 4:2:2 sampling format
- YCbCr 4:2:0 sampling format.

The YCbCr 4:2:0 format is used with MPEG video compression and the YCbCr 4:2:2 format is the color space provided by a large number of modern CMOS image sensors with integrated pixel processing. The conversion from YCrCb to RGB is done based on the CCIR 601-2 standard. When the pixel data is in 4:2:0 or 4:2:2 YCbCr planar format,

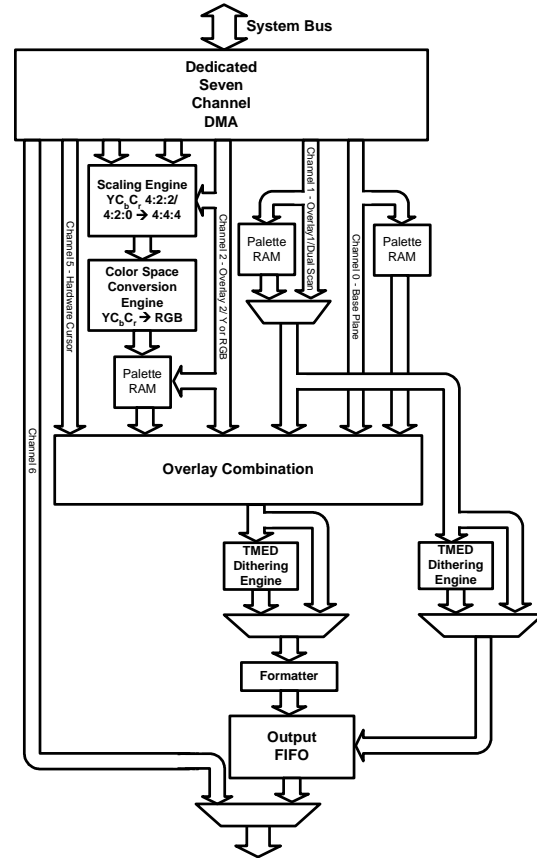


Fig. 12. PXA27x LCD Controller Block Diagram.

the data is stored in three different memory regions in the frame buffer. The chrominance channels are first scaled to the target resolution followed by the 3x3 matrix operation to convert the pixel data to the RGB color space.

B. Quick Capture Interface

The Intel quick capture interface is a highly configurable and allows the direct connection of the PXA27x processor to the majority of the CMOS image sensors available in the market today. The interface is responsible for acquiring both data and control signals from the CMOS sensor as well as providing the appropriate formatting of the data prior to being routed to memory through DMA. The block diagram for the capture interface is illustrated in Figure 15.

The CMOS sensor may provide either raw or preprocessed image data to the interface through a variety of programmable options. The interface receives the video/image data stream from the CMOS sensor and provides all control signaling for operation as either a Master or Slave device. In the Master mode, the line and frame synchronization signals are provided by the CMOS sensor and in Slave mode, the synchronization signals are provided by the interface. Several reduced pin-count alternatives are supported as subsets of the

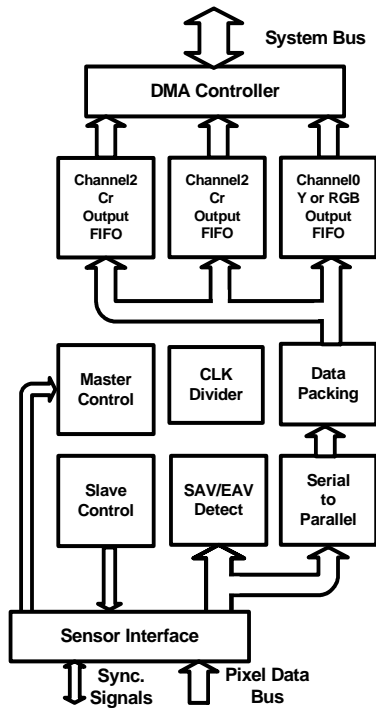


Fig. 13. Capture interface block diagram.

Master mode of operation. These include allowing the data to be serialized and the elimination of the separate synchronization signals for sensors that utilize the ITU-R BT.656 Start-of-Active-Video (SAV) and End-of-Active-Video (EAV) embedded in the data stream. The width of the data bus is also flexible and may be configured in an 8, 9, or 10-wire parallel mode or in a 4 or 5-wire serialized mode. Table 2 provides an overview of the interface modes.

Table 2: Summary of Intel Quick Capture Interface Modes.

Instruction	Description
Master-Parallel	The synchronization signals are internally generated by the CMOS sensor, the interface to the sensor is a parallel data bus either 8, 9, or 10-bits in width
Slave-Parallel	The synchronization signals are externally generated for the CMOS sensor, the interface to the sensor is a parallel data bus either 8, 9, or 10-bits in width.
Master-Serial	The synchronization signals are internally generated by the CMOS sensor, the interface to the sensor is a serial data bus either 4 or 5-bits in width.
Embedded-Parallel	The synchronization signals are internally generated by the CMOS sensor with the Start-of-Active-Video (SAV) and End-of-Active-Video (EAV) embedded in the data stream for a data bus which is 8-bits in width
Embedded-Serial	The synchronization signals are internally generated by the CMOS sensor with the Start-of-Active-Video (SAV) and End-of-Active-Video (EAV) embedded in the serialized data stream for a data bus 4-bits in width.

The pixel data received may be in several possible formats. These formats include RAW, YCbCr 4:2:2, RGB 8:8:8,

RGB 6:6:6, RGB 5:6:5, RGB 5:5:5, and RGB 4:4:4. When a RAW capture mode is enabled, the data may be in 8, 9, or 10-bit formats. The RAW data is de-serialized if necessary, and then packed as either 8-bit or 16-bit elements prior to transfer to memory. In a similar manner the pre-processed RGB and YCbCr image formats are packed into 16-bit or 32-bit elements.

The YCbCr format has the additional option of being planarized prior to being stored to memory. This planarization facilitates SIMD processing using Wireless MMX as well as immediate display using the color space conversion engine available with Overlay2 of the LCD controller. The RGB formats may also be formatted to enable immediate preview using the LCD controller. Programmable options are provided to reduce RGB component precisions and provide the transparency management for use with the LCD overlays. The formatting options offer value in reducing power consumption during the digital viewfinder preview sequence for image and video capture.

X. WIRELESS INTEL SPEEDSTEP® TECHNOLOGY

To support extended battery life the Intel® PXA27x processor family introduces Wireless Intel Speedstep® Power Manager Technology for advanced power management. At the hardware level, the technology provides several power domains and modes. A dedicated PMU (Performance Monitoring unit [11]) provides for interaction with the software control components of the system. The software components enable the hardware features to be integrated into embedded operating systems through performance and policy management strategies.

A. Power Modes

The PXA27x processor family supports six power modes. Three new power modes, Deep Idle, Standby, and Deep Sleep enhance the power management capabilities introduced with previous products.

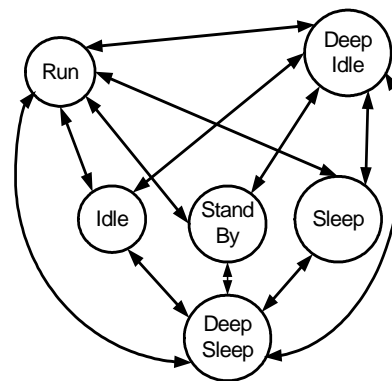


Fig. 14. State transition diagram between different power modes

The power modes are primarily differentiated by available functionality, total power consumption, and the amount of time to enter and exit a mode. As a function of workload and resource utilization, the device can transition between the power modes to minimize power consumption. The state transition diagram between the different modes is shown in Figure 14.

Each of the six modes provide different levels of power consumption and resource availability in addition to variation in the transition time to the fully-running mode. A description of the power modes is provided in Table 3.

Table 3: Description of Power Modes

Instruction	Description
Run	All internal power domains (chip areas powered by separately switched and regulated voltage sources). All clocks are enabled and running.
Idle	The clocks to the CPU core are gated off. Clock frequency (can be the highest available) will be delivered to the CPU upon recovery from Idle Mode. Recovery is triggered by interrupts.
Deep Idle	This mode is similar to Idle Mode except for it is at the lowest (13MHz) frequency.
Standby	All internal power domains are placed in their lowest power mode except the real-time clock and the oscillator. Both PLLs are disabled to save power. Recovery is via external means or internal events such as time delays.
Sleep	Similar to Standby Mode in preserving oscillator and real-time clock functionality, except PXA27x core power is turned off. Remaining domains, such as on-chip memory, are placed in a state-preserving low-power state. Recovery requires reboot..
Deep Sleep	Similar to Sleep Mode, except all clock sources are disabled. High-voltage supplies are disabled. All remaining power domains are powered directly from the backup battery so states are preserved. Recovery is via external means or internal events such as time delays. Recovery requires reboot.

To support these powers modes the PXA27X processor has been partitioned into a number of power domains. The main core domain allows the voltage and frequency of the Intel® XScale core to be dynamically adjusted to meet the processing requirements.

B. SW Power Management Component

Effective usage of the Wireless SpeedStep Technology can be achieved through a software level management solution. The PXA27x family offers a generic framework to implement the management policy. Information is extracted about the system level activity from the OS, application, user preference and various other sources to decide optimal voltage and frequency settings. The PMU is used to monitor different system level activities (i.e. CPI-cycles per instruction, cache-efficiency etc.), providing additional information for dynamic voltage and frequency management. The software power management framework is shown in Figure 15.

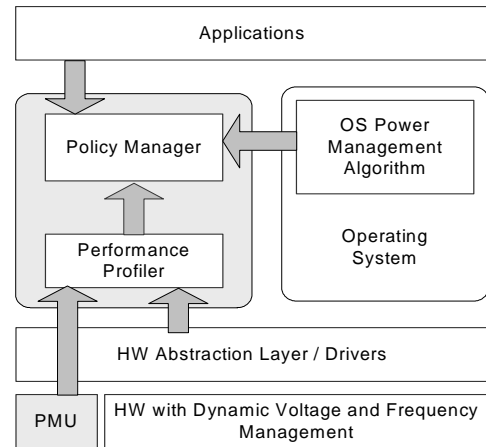


Fig. 15. High Level Structure of Power Management Framework

There are two key components to the solution, the policy manager and the performance profiler. The policy manager receives task-level activity information (how many threads are running, or how often the processor is idle) from the OS. In addition, it can receive information from OS's own power manager. The policy manager decides the power mode in the system, such as, when should the system be in sleep or deep-sleep. When the system is in the run mode the policy manager performs dynamic voltage and frequency management based on the inputs from performance profiler. The performance profiler periodically samples performance monitoring event counters and determines frequency for the different components in the system (i.e. core-frequency, memory frequency, system communication fabric frequency). Based on the periodic profile information, if the current mix of application is computation bound, the core-frequency is raised and similarly, if the application is memory bound, the memory controller frequency will be raised. On the other hand, as the thread-level activity reduces core and memory frequency is reduced. Based on the chosen operating frequency the voltage is adjusted as well. The performance profiler also communicates with the software-drivers for different peripherals so that peripherals can be turned off/on based on the usage. The policy manager and performance manager co-ordinate these transition activities. Depending on the target platform the policy can be optimized in terms of transition thresholds, frequency changing steps and method of profiling.

XI. SYSTEM IN A PACKAGE

As well as providing high levels of integration in the system on a chip the PXA27x processor also provides further integration with stacked multi-chip packaging (MCP). Table 4 shows the currently available MCP configurations. A key advantage of the stacked packages in phone and PDA

applications is the saving in PCB real estate required to implement the platform which allows smaller form factor products to be created.

Table 4. System in a Package Configurations

Part	Max MHz	Other Stack	Size	Type
PXA270	624		13x13x1	356 pin VFBGA
PXA271	416	256Mbits Intel® Flash in 16-bit memory bus & 256Mbits Low Power SDRAM in 16-bit memory bus ^a	14x14x1.45	336 pin Intel® Stacked Chip Scale Packaging
PXA272	520	512Mbits Intel® Flash in 32-bit memory bus ^a		
PXA273	416	256Mbits Intel® Flash in 32-bit memory bus ^a		

a. Actual configurations may vary please refer to Intel web site for latest configurations[20]

XII. SUMMARY

The Intel PXA27x processor family provides a highly-integrated low power system on a chip for wireless and PDA applications. Utilizing the Intel XScale Microarchitecture with Wireless MMX technology coupled with multi-media interfaces allows the device to support a rich, low-power, multimedia experience. The advanced power management of Wireless Intel Speedstep technology supports extended battery life in target systems.

XIII. ACKNOWLEDGEMENTS

We acknowledge the significant contribution of the entire Intel PXA27x development team in Austin, TX, Colorado Springs, CO, Chandler, AZ and the PCG systems engineering in Hudson, MA and Shanghai, China.

XIV. REFERENCES

- [1] Alex Peleg and Uri Weiser, "MMX Technology Extension to Intel Architecture", IEEE Micro, 16(4):42-50, Aug. 1996.
- [2] Keith Diendroff, "Pentium III = Pentium II+ SSE", Micro Processors Report, 13(3):6-11, March, 1999.
- [3] Lee, Ruby, "Subword Parallelism in MAX-2", IEEE Micro, 16(4), 51-59, Aug. 1996
- [4] Tremblay, Marc, et. al. "VIS Speeds Media Processing", IEEE Micro, 16(4): 10-20, Aug. 1996.
- [5] Weiser, Uri, et al, "The Complete Guide to MMX™ Technology", McGraw-Hill, 1997, ISBN 0-07-006192-0.
- [6] Khan, M.H, Paver N.C., Aldrich, B.A, Hux, A. "Optimization Techniques for Mobile Graphics and Gaming Applications using Intel® Wireless MMX™ Technology", GSP 2004
- [7] N. C. Paver et. al. "Accelerating Mobile Video with Intel® Wireless MMX™ Technology", IEEE Workshop on Signal Processing Systems (SIPS), Aug 27-29, 2003.
- [8] N. C. Paver et. al. "Intel® Wireless MMX(TM) Technology: A 64-Bit SIMD Architecture for Mobile Multimedia", International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2003.
- [9] Paver, Nigel C., Bradley C. Aldrich, Moinul H. Khan, Programming

- with Intel® Wireless MMX™ Technology: A Developer's Guide to Mobile Multimedia Applications, Hillsboro, OR: Intel Press, 2004.
- [10] Aldrich, B.A, Paver N.C, Khan, M.H, Emmons, C.D., "A Spatial Clustering Approach for Reduced Memory Traffic in Motion Estimation using Intel Wireless MMX™ Technology", 8th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2004), July 2004
- [11] Intel XScale(R) Core Developer's Manual, <http://www.intel.com/design/intelxscale/273473.htm>
- [12] Kuhn, Peter, "Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation", Kluwer Academic Press, ISBN 0-7923-8516-0
- [13] Seal, David "Advanced RISC Machines Architecture Reference Manual", Prentice Hall, 1996. ISBN 0-201-73719-1
- [14] Furber, S.B., "ARM System-on-Chip Architecture", Addison Wesley, 2000. ISBN 0-201-67519-6.
- [15] "Integrated Performance Primitive", <http://intel.com/software/products/ipp/>
- [16] "MPEG4 Overview (V.21)", Edited by Rob Konen, ISO/IEC JTC1/SC29/WG11 N4668.
- [17] International Organization for Standardization. "ISO/IEC JTC1/SC29/WG11N1902 14496-2 Committee Draft (MPEG-4)." November 1997.
- [18] [http://www.arm.com/aboutarm/55CE4Z/\\$File/ARM_Architecture.pdf](http://www.arm.com/aboutarm/55CE4Z/$File/ARM_Architecture.pdf)
- [19] IUT-T Recommendation H.263: "Video Coding for Low Bitrate Communication", Geneve, 1996
- [20] <http://developer.intel.com/design/pca/prodbref/253820.htm>