



ZSP[®] Architecture for Wireless Applications White Paper



Trademark Acknowledgments

© VeriSilicon Holdings Co., Ltd. All rights reserved worldwide. VeriSilicon, the VeriSilicon logo, ZSP and the ZSP logo are the trademarks of VeriSilicon Holdings Co., Ltd. in the United States and/or other jurisdictions. All other trademarks are the property of their respective holders.

Printed in P.R.China.

VeriSilicon Holdings Co., Ltd. reserves all its copy rights and other intellectual property rights, ownership, powers, benefits and rights arising or to arise from this manual. All or part of the contents of this manual may be changed by VeriSilicon Microelectronics (Shanghai) Co., Ltd. without notice at any time for any reason, including but not limited to improvement of the product relating hereto.

VeriSilicon Holdings Co., Ltd. shall not undertake or assume any obligation, responsibility or liability arising out of or in respect of the application or use of the product described herein, except for reasonable, careful and normal uses.

Nothing, whether in whole or in part, within this manual can be reproduced, duplicated, copied, changed or disposed of in any form or by any means without prior written consent by VeriSilicon Holdings Co., Ltd.



Contents

1. Designed for Wireless Applications	4
2. DSP Needs for Wireless	4
3. Throughput	4
4. Power Dissipation	5
5. System Cost	6
6. Time to Market	7
7. Size	8
8. Choice of Supply	8
9. Summary	9

1. Designed for Wireless Applications

Selecting a DSP architecture for your wireless application is an important decision, one with many considerations. To stay competitive you need a DSP that can not only handle your current wireless needs, but also help make tomorrow's third generation applications possible. ZSP, with its high processing performance, efficient compiler, high code density, and multi-channel support with a single processor, is the ideal DSP to give you the edge you need to stay in front of the competition.

Wireless applications include:

- Basestations
- Multimode handsets
- Wireless local loop systems
- High-speed wireless LANs
- Wideband CDMA

2. DSP Needs for Wireless

There are two classes of wireless applications, portable and non-portable. Portables are consumer hand-held or pocket-size devices like cellular phones and pagers. DSPs for these applications should have a small die size, high integration, and minimal memory use to hold costs down, along with low power needs. Non-portable encompasses all other applications, including mobile and non-mobile devices like Wireless Local Loop (WLL) products, car phones and infrastructure equipment. In these applications processing power is a high priority along with C-compiler performance.

The DSP requirements of wireless applications can be generally categorized as raw throughput, low power dissipation, low systems cost, fast time to market, and size. Each one has a differing priority depending on the particular application.

3. Throughput

There is a clear demand in all wireless applications for more DSP horsepower, usually rated in MIPS (millions of instructions per second). In the case of portable applications, higher throughput can provide increased performance, multi-mode capability, and the inclusion of additional features. For infrastructure equipment, higher throughput can mean cost reduction through multi-channel processing, flexibility to accommodate future standards or features, and multi-mode capability.

DSP throughput is often a misused term. At face value, throughput is equated with operating frequency. While speed is a major part of the equation, there are other significant factors that contribute to the ability of a

device to perform a function. The architectural efficiency of the device to perform that function determines how well the device handles that specific function, irrespective of operating frequency. Hence, it is actually the efficiency and the operating frequency together that determine the throughput of a given DSP for a specific task.

Architectural efficiency is an objective measure of the “work done” by each instruction of the processor. Therefore, it’s calculated as the ratio between MIPS to MOPS, which equals the number of operations per instruction that the processor performs when executing a specific function.

$$\text{Architectural Efficiency} = \frac{(\# \text{ of instructions needed for a task})}{(\# \text{ of RISC Operations for that task})}$$

This ratio is less than one for traditional fixed-point DSP architectures, and is greater than one for new DSP architectures, like ZSP Architecture, that employ multiple execution units.

It is to be noted that the efficiency of the DSP is affected by the entire application. Therefore, both control and signal processing functions influence the DSP quality — and both are found in wireless applications. A DSP architecture optimized for wireless applications will include hardware to support both traditional signal processing and control type functions. For example, signal processing hardware might include multiple MAC units and Viterbi add/compare/select (ACS) support. Control code might support bit manipulation capability, branch prediction, and multiple ALUs.

The ZSP architecture answers these concerns, with its superior signal processing and control processing capability. Current ZSP processors are 200 MHz four-scalar (superscalar) devices with four execution units: two MACs and two ALUs. The four units can work in parallel for an effective throughput of 800 MIPS. The MAC units provide a single instruction Viterbi ACS for efficient channel decoding and equalization. The architectural efficiency of this device for various wireless applications is typically 1.1 to 1.5 operations per instruction.

4. Power Dissipation

Today's wireless applications need increasingly lower power dissipation. For portable applications, lower DSP power dissipation can mean longer talk and standby-times, as well as smaller and lighter terminals made possible by reducing battery cells. Equally significant is the demand for smaller, more densely populated, and less costly infrastructure equipment. For these, smaller cabinets and higher channel capacity raise the importance of dissipating heat. Costs can be reduced significantly by the elimination of

active cooling and the use of smaller power supplies.

When comparing power consumption of DSPs, it's important to consider how many channels of data can be processed at a given clock frequency. A 200 MHz four-way superscalar ZSP processor such as the VSI402ZX will be capable of processing over 90 channels of GSM full rate speech coding at less than 6mW per channel.

5. System Cost

Being consumer products, portable wireless devices are naturally very cost sensitive. Users of non-portable equipment, which aren't typically consumer devices, are also showing signs of becoming increasingly cost conscious. For example, cellular service providers are moving towards using mini, micro, and pico-cells for greater coverage and spatial reuse, which means more base station hardware.

Meanwhile, service providers are trying to protect their investment in infrastructure equipment by using base stations that can be reconfigured through software. This provides the ability to accommodate changing standards, add new features, and upgrade software without replacing hardware. This trend towards reducing system cost is affecting DSP devices in the following ways:

1. *Multi-channel capability* — This infrastructure issue relates directly to the cost per channel of the wireless system implementation. The ZSP's large internal RAM, multi-tasking support, and high throughput all address these needs.
2. *Multi-mode capability* — This concerns both portable and non-portable applications, though it actually has two meanings. The first can be interpreted as "multi-standard" capability, in which the same terminal or infrastructure device can operate in one of several independent modes depending on the service available. This is evident in cellular phones that are both GSM and PCS1900 compatible. In another sense, multi-mode may imply "multi-feature," such as a wireless device that offers several speech coders with the optimum one being selected based on the service available or the user's preference. This requires large internal memory and excess throughput, such as that found in the ZSP architecture.
3. *External hardware interfaces* — Ideally, no external logic should be required to interface the DSP to the surrounding components. For DSP vendors, this is a difficult demand to meet without an excess of hardware. Generally, DSPs targeted at portable applications will have very well defined and specific I/O interfaces, whereas non-portable devices will employ more flexible, generic interfaces. In most wireless systems, the DSP device must communicate with a host processor, which provides the upper layer protocol

processing, and one or more data conversion devices.

4. *External memory* — DSPs are high-speed processors, and fast external memory is expensive. In addition, the use of external memory typically reduces the throughput of the DSP device. For portable applications, there should be no external memory. For infrastructure applications, the trend is to eliminate external memory to reduce cost. However, some manufacturers prefer the option of adding memory for later feature expansion. That's why ZSP has a large internal program memory, an external memory port, and non-cycle stealing DMA to external memory space.

Note that memory requirements are greatly influenced by code density. Code density, which equates to the number of bits required to perform a specific task, is a function of instruction word width and instruction set architecture. Code density is inherently low for VLIW (Very Long Instruction Word) machines, where instruction word width is traded for parallel execution and raw throughput.

5. *Power* — As previously stated, power dissipation affects non-portable system cost due to the need for active cooling (fans) and a larger power supply. For portable devices, decreasing battery size and drain means using a DSP with the lowest possible power dissipation. Thus, the trend in both types of wireless applications is to reduce the operating voltage. A reduction in operating voltage yields a square-gain in power dissipation, pushing today's portable systems to 2.5V or lower nominal operating voltage while heading towards 1.0V in the future. Non-portable applications are following a similar trend.

There are typically tradeoffs to be made between system cost and performance. For wireless applications, sacrificing system cost for greater performance is economically unacceptable.

Current ZSP Processors such as the VSI401Z and VSI402ZX from VeriSilicon are designed for communications infrastructure applications. This is evident by the internal memory and hardware interfaces, 48K words on-board RAM, external memory port with DMA, 800 MIPS throughput, two serial ports, and a host port. Furthermore, this device has a fixed 16-bit instruction word and RISC-like instruction set to achieve a high code density. VeriSilicon's next product in this family, the VSI402ZX, will offer larger memory, multi-channel DMA support, and enhanced serial port functionality and will be binary compatible with its predecessor.

6. Time to Market

The product design cycle for consumer devices is typically six to twelve months. The design cycle for infrastructure equipment is typically one to three years. In both cases, the trend is toward shorter design cycles and faster response

time to market demands. For the wireless DSP device, the development tools, including the availability of a practical HLL (high-level language) compiler influence time to market. Keep in mind that writing DSP code is not particularly difficult, but optimizing and debugging it is.

Development tools should be easy to use, stable, and fast. The user takes on a large enough burden in writing signal processing software without taking time to learn or debug tools. In addition, the availability of an HLL compiler that generates working, optimized code initially saves the programmer the effort of learning a new DSP assembly language and allows for easy code maintenance.

The ZSP Software Development Toolkit provides a set of tools to assemble, link, and simulate programs on PC and UNIX platforms. The tools are based on industry-standard GNU tools, and provide a command line or GUI interface. The simulator offers two modes of operation for debugging. For functional behavior, the simulator can be invoked in a debugging mode, allowing the user full visibility and control of the machine's state. For optimization of a program, the simulator can be invoked in a profile mode, which executes at a higher speed, providing cycle count, pipeline efficiency, and statistical information.

An optimizing ANSI C-compiler is also available for the ZSP architecture. The compiler offers the advantages of high-level language development and code maintenance while maintaining efficient assembly code generation. By enhancing the compiler with features like fixed-point data types and defining C intrinsic functions, the compiler generates assembly code that's less than 1.5 times the cycle count of hand-coded assembly.

7. Size

While size has always been a priority in wireless applications, there's little pressure to reduce the package size of DSPs beyond those currently available. State-of-the-art DSPs in portable devices are generally packaged in 100 pin TQFPs, while DSPs in non-portable devices will generally have packaging with >132 pins due to I/O and power dissipation requirements.

8. Choice of Supply

Excessive supplier power is a major concern of developers in the highly competitive wireless marketplace. Proprietary architectures can force system designers into corners as their valuable software IP becomes closely tied to a single source range of processors. The higher the commitment is to these proprietary architectures, the lower the freedom for price negotiation. The ZSP architecture is open to multiple sources and has a strong roadmap, thus providing wireless product developers the best way of ensuring supply

while remaining competitive.

9. Summary

There are a number of major factors affecting portable and non-portable wireless equipment design, and each has an impact on the DSP requirements. To review, the established trend is for more processing horsepower, lower power dissipation, reduced system cost, and faster time to market. Thus, the next generation of DSP for wireless must include:

- High operating frequency ($\gg 100$ MHz)
- Multiple execution units
- Efficient signal processing and control code processing (architectural efficiency > 1)
- Large internal memory for program and data with good code density
- I/O interfaces targeted for the specific application
- Low power dissipation without sacrificing throughput
- Open architecture to ensure supply and competitiveness

The ZSP Architecture is the only DSP that has these features today and is available now in silicon, providing a flexible, scalable architecture for the future of wireless.