An Independent Analysis of the:

MIPS Technologies MIPS32® 74K™ Licensable Processor Core

By the staff of

Berkeley Design Technology, Inc.

Overview

MIPS Technologies, Inc. is a provider of semiconductor IP, including processor architectures and IP cores based on these architecture. The company offers a range of licensable 32-bit processor cores for use by SoC designers. These processor cores target applications ranging from deeply embedded, real-time control applications to high-performance embedded systems with demanding digital signal processing requirements. The 74K core is a MIPS high-performance processor core, and includes features intended to address applications with significant signal processing requirements.

In this white paper, BDTI, an independent analysis company focused on digital signal processing (DSP) technologies, assesses the signal processing features and capabilities of the 74K core.

Contents

Introduction

MIPS Technologies, Inc. is a provider of licensable, synthesizable 32-bit processor cores and other semiconductor design IP. MIPS processors are commonly found in set-top boxes, VoIP SoCs, digital TVs, DVD recorders/players, and networking equipment, applications in which signal processing is becoming increasingly important. To address applications with combined RISC/DSP demands, MIPS offers 32-bit cores with DSP-oriented hardware and instructions.

The first 32-bit MIPS core with significant DSP capabilities was the 24KE core, which was introduced in 2005. This core was based on the earlier 24K core but added a set of DSP instruction set extensions, called DSP ASE (Application Specific Extension) Rev 1, that provided a significant step up in signal processing capabilities. MIPS then introduced the 34K core, which also supports the DSP ASE instructions and adds multithreading capabilities.

In 2007 MIPS announced the high-performance 74K core. The 74K core is a dual-issue superscalar core that supports MIPS' next generation of DSP-oriented instruction set extensions, called DSP ASE Rev 2. The 74K core targets demanding multimedia and networking applications, such as WiMaX, DVD players, and VoIP. According to MIPS, the 74K core is fully synthesizable and operates at up to 1.11 GHz in a 65 nm process; core details are shown in Table 1. The 74K core can include an optional floating-point unit; if this unit is included the core is called the 74Kf. According to MIPS, the 74K core has been delivered to initial licensees.

Core Implementation	Max. Clock Speed	Die Area
High-Performance	1.11 GHz	2.5mm^2
Area-Efficient	830 MHz	2.1mm^2

TABLE 1. Worst-case core details for the MIPS 74K core fabricated in TSMC 65 nm process using standard cells (no structured or hard IP). Die area is for the core plus 32K instruction cache and 32K data cache. (Data is courtesy of MIPS, and has not been verified by BDTI.)

BDTI recently assessed the 74K core's suitability for digital signal processing (DSP) tasks. This white paper presents the results of that assessment.

About BDTI

Berkeley Design Technology, Inc. (BDTI) is widely recognized as a trusted source of independent analysis of processing engines and tools targeting embedded applications. BDTI uses its signal processing benchmark suites and in-house expertise to evaluate the signal processing capabilities of various processing engines, including DSP processors, general-purpose processors, FPGAs, and multi-core devices. BDTI also provides contract software development services, and has completed numerous embedded software projects targeting general-purpose processors, For further information see www.BDTI.com.

Architecture

OVERVIEW

The 74K core is a 32-bit RISC CPU that implements the MIPS32 Release 2 instruction set architecture and supports the DSP ASE Revision 2 instruction-set extensions. It can include an optional floating-point unit for support of floating-point computations.

The 74K core can issue and execute up to two instructions in parallel (or up to four instructions—two integer and two floating-point—if the floating-point unit is included). The base core contains a load/store unit and separate data path; one instruction can be executed in each of these two units in parallel. Figure 1 provides a block diagram of the 74K core.

The 74K core's data path contains a 32-bit ALU (which is part of the "integer execution unit," or "IEU"), a multiply/divide unit (MDU), and an optional "CorExtend" interface (described later in this paper). The load/store unit (or AGEN pipe) and data path share thirty-two 32-bit general-purpose registers and four 64-bit accumulators.

Like other MIPS cores, the core uses a load/store architecture, where all ALU, shifter, and MDU operations operate on data from (and store results to) core registers. Load/store operations support 8-, 16-, and 32-bit data transfers to and from memory. If the core includes the floating-point unit, it can also perform 64-bit load/store operations. The core includes a level-1 instruction cache and optional level-1 data cache.

DATA PATH EXECUTION UNITS

The 74K core data path includes a multiply-divide unit (MDU) and an arithmetic logic unit (ALU).

Multiply and Divide Unit (MDU)

From a signal processing perspective, the multiply/divide unit is a key feature of the 74K core, because signal processing workloads often require frequent multiply and multiply-accumulate (MAC) operations.

The MDU supports 32 × 32-bit multiplies and MACs with single-cycle throughput, and also supports SIMD (single instruction, multiple data) multiplications or MACs of two sets of 16-bit data operands. (16-bit data is commonly

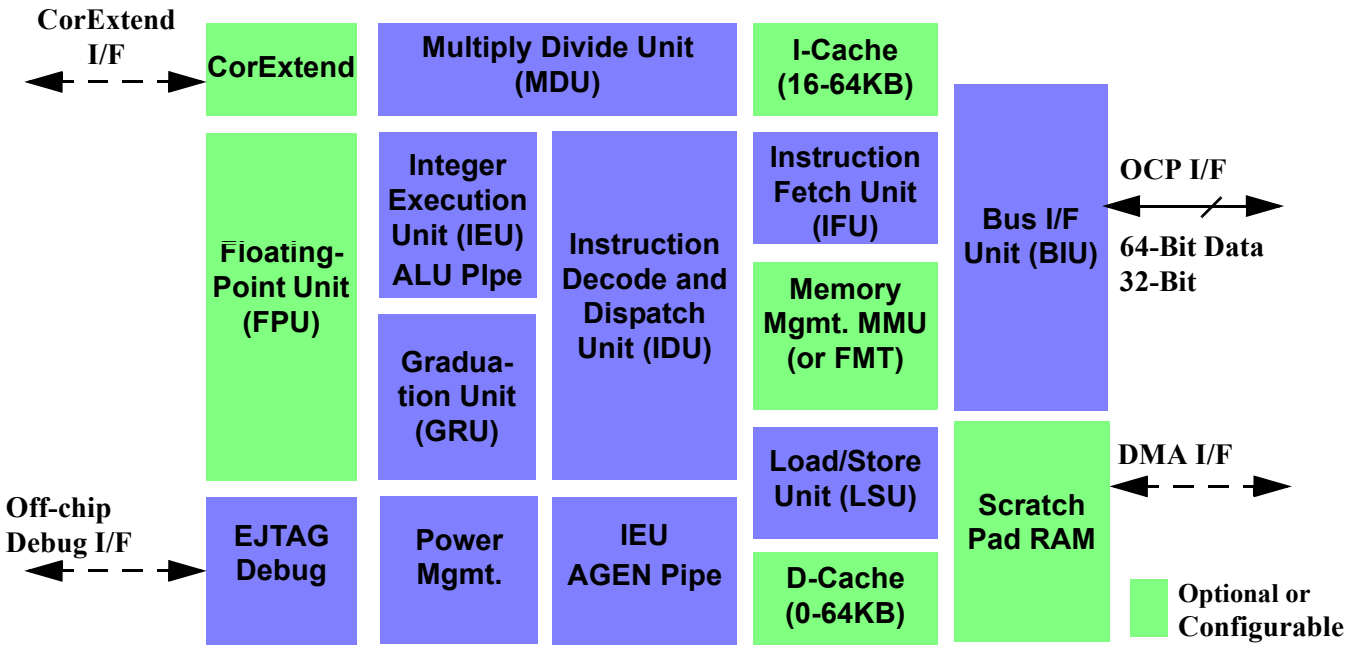

FIGURE 1. MIPS32® 74K® Processor Core Block Diagram

used in signal processing applications.) The multiplication operations are supported by the four 64-bit accumulators.

The 74K core's 16-bit multiplication capabilities (in terms of the number of multiplications that can be executed in parallel per cycle) are comparable to those of medium-performance DSP chips, such as Texas Instruments TMS30C55x and Analog Devices' Blackfin, and to mid-range licensable DSP cores, such as the Ceva Teaklite-III. They are not as powerful as the MAC capabilities of some high-performance processors, which can perform four (or more) 16-bit multiplications per cycle.

Arithmetic Logic Unit (ALU)

The ALU supports a variety of common 32-bit arithmetic and logic operations. It also supports a range of SIMD capabilities, including dual 16-bit adds, subtracts, shifts, and compares and quad 8-bit adds, subtracts, shifts, and compares. The SIMD arithmetic operations are useful for a wide range of signal processing algorithms, including FFTs, video/graphics processing, and Viterbi decoding, and are similar to those found on mid-range to high-end DSP processors. The 74K core ALU also supports specialized instructions that facilitate efficient SIMD processing, such as packing and unpacking of 16- or 8-bit operands within 32-bit registers. Many of the 74K core arithmetic operations include saturation or rounding, which are useful for maintaining signal fidelity in many DSP applications and are not always included in the instruction sets of DSP-enhanced general-purpose processors.

PIPELINE

To support superscalar execution, the 74K core has dual asymmetric pipelines. One pipeline is used for computations (the ALU pipeline) and one is used for address generation and loads/stores (the AGEN pipeline). This design differs from superscalar processors that use symmetric pipelines, in which any instruction can execute in either pipe. Asymmetric pipelines can be a good match for signal processing tasks, which typically require computations to be executed in parallel with data loads and stores. The asymmetric approach yields a noticeable performance improvement over single-issue execution without requiring as much silicon area (and power) as fully symmetric pipelines.

Both of the 74K core pipelines are quite deep—14 stages for the ALU pipeline, and 15 stages for the address generation pipeline. The primary advantage of using deep pipelines is that they enable high clock speeds, and indeed, the 74K core's 1 GHz speed is impressive, particularly for a synthesizable core. One trade-off is that deep pipelines can cause long delays when the pipeline is flushed (such as during a branch). On the 74K core, this penalty is 12 cycles. To mitigate this effect and reduce branching penalties, the 74K core includes branch prediction hardware that uses three 256-entry branch history tables.

Deep pipelines can also lead to long latencies for operations other than branches. For example, on the 74K core, 32 × 32-bit multiply instructions have either 5- or 7-cycle latencies (depending on the instruction variant), though all have single-cycle throughput. Such instruction latencies can create performance bottlenecks. In some cases, the latencies can be concealed by software pipelining and instruction reordering, reducing their effect on performance. The 74K core also supports out-of-order instruction execution (described further, below), which can automatically reorder instructions to help mitigate the effect of multi-cycle latencies and improve performance.

According to MIPS, the 74K core compiler is capable of software pipelining and is designed to reduce the performance penalties due to long latencies. The compiler will not always be able to identify and implement optimal code, however, and in some cases the programmer will need to hand-optimize assembly language to achieve the 74K core's maximum signal processing performance potential.

With the 74K core, the long instruction latencies will make the optimization process more challenging (since they will make it more difficult to understand the software flow), though probably not more difficult than optimizing assembly code for today's high-performance DSP processors.

OUT-OF-ORDER EXECUTION

The 74K core supports out-of-order instruction execution, in which the processor dynamically reorders the instruction stream to more efficiently pair up instructions for parallel execution and help hide instruction and cache latencies. According to MIPS, the processor analyzes a window of up to eight instructions per pipeline and reorders them in a way that maximizes performance without affecting the functionality of the code. To ensure that the correct functionality is preserved, a completion buffer holds the results of instructions until they are "graduated" in program order.

The 74K core is binary compatible with the 24KE core; software written for the earlier core can run on the 74K core without being recompiled. This is a benefit for embedded designers who don't necessarily have access to source code or don't want to recompile their code. 24KE binaries that are run on the new core will typically require fewer clock cycles to execute because of the out-of-order superscalar execution.

Out-of-order execution is commonly used in high-performance computer CPUs, but is relatively rare in embedded processors because it requires additional hardware to analyze the instruction stream and juggle the instructions. This hardware comes at the price of additional silicon area

and power consumption. Furthermore, in the context of implementing demanding signal-processing applications, out-of-order execution can make it more difficult to understand and accurately predict how many cycles a given section of software will take to execute—thus making it hard to ensure real-time performance and optimize software.

MIPS has compromised on these trade-offs by constraining the core's reordering capabilities. Some high-end computer CPUs examine a hundred or more instructions; MIPS has limited the window to eight instructions in order to balance the additional complexity against the potential performance benefits.

The effect of out-of-order execution on the performance of software will vary depending on the specifics of the software being executed. In some cases, out-of-order execution may help meet performance targets and eliminate (or reduce) the need to hand-optimize the software in assembly language, and it should help speed up software that was compiled for the 24KE core.

MEMORY SYSTEM

The 74K core supports separate level-1 instruction and data caches. The instruction cache is 4-way set associative and can be configured as 16, 32, or 64 KBytes. Up to four instructions can be fetched per cycle. (Though the processor without the FPU can only issue two instructions per cycle, the additional fetches are still useful for supporting out-of-order execution and mitigating instruction cache miss penalties.)

The data cache is optional, and if present it can be configured as 16, 32, or 64 KBytes. Like the instruction cache, it is 4-way set associative. The core uses a 128-bit bus to transfer instructions and a 64- or 128-bit bus to transfer data from L1 memory. However, the 74K core only supports 64-bit data transfers to the optional floating-point unit. Otherwise, a maximum of one 32-bit data word can be read per cycle. Thus, the core can not provide sufficient data bandwidth to keep the dual multipliers fed with four new 16-bit operands per cycle. In comparison, most DSP processors and many DSP-enhanced general-purpose processors match the data bandwidth to the multiplication bandwidth.

The 74K core's data bandwidth limitation may become a bottleneck in some applications, though algorithm transformations (such as "zipping" in filter algorithms) can sometimes be used to circumvent the bottleneck.

Like many DSP processors with caches, the core supports instruction and data cache locking on a per-line basis. This feature enables critical portions of software and the associated data to be locked into the cache, helping the programmer to achieve consistent real-time performance.

The core supports an optional data scratchpad RAM, independent of the caches. This RAM can be configured as 4 KBytes to 1 MByte. The 74K core accesses off-core memory via a 32-bit address bus and 64-bit data bus. The core also includes a programmable memory management unit (MMU) that maps virtual addresses to physical addresses. The MMU can be configured (at synthesis time) to be used as a translation look-aside buffer or for fixed mapping translation.

Instruction Set

The 74K core supports the baseline MIPS32 instruction set, the DSP ASE Rev 2 extensions, and the MIPS16e 16-bit compressed instruction set extensions.

DSP ASE REV 2 INSTRUCTIONS

The DSP ASE instructions are designed to accelerate common signal processing algorithms. The DSP ASE Rev 1 instructions included in the 24KE and 34K support a range of SIMD signal processing-oriented operations, including dual 16-bit integer and fractional multiplications, and quad add/subtract of eight 8-bit data operands. Also included are typical DSP addressing capabilities, like support for modulo (circular) addressing and bit-reversal (useful for radix-2 FFTs). Many DSP ASE instructions support rounding or saturation.

In general, the DSP ASE instruction-set extensions provide capabilities that are similar to those found on mid-range DSP processors.

DSP ASE Rev 2 adds a number of new instruction variants that are geared towards multimedia processing, VoIP, video processing, and Viterbi decoding, as well as being intended to improve the orthogonality of the instruction set and facilitate development of more efficient compilers. Many of the new instructions are similar to those included in DSP ASE Rev 1, but offer additional flexibility. For example, they offer more options for source and target registers, and more options for fractional vs. integer data types. (None of the DSP ASE instructions operate on floating-point data, even if the core includes the optional floating-point unit. This is sensible, because most performance-critical DSP algorithms rely on fixed-point data.)

The Rev 2 additions don't increase the processor's multiplication bandwidth relative to Rev 1 (the maximum is still two 16 × 16 bit multiplications in parallel), but will help to improve the 74K core's performance on some signal processing algorithms, such as filters that process complex-valued data and FFTs. For example, the following instruction stream performs a complex FIR filter calculation using DSP ASE Rev 1 instructions. This algorithm requires a dot-product of complex-valued vectors (i.e., with real and imaginary components), and is implemented on the 74K core using the DSP ASE Rev 1 as follows:

```
rotr $t2, $t1, 16 #swap real/imag in t1
mulsaq_s.w.ph   $ac0, $t0, $t1 #compute
real part
dpaq_s.w.ph     $ac1, $t0, $t2 #compute
imaginary part
```

DSP ASE Rev 2 adds a cross-multiply instruction ("dpaqx") that eliminates the need to swap the real and imaginary components, enabling the code to be rewritten using fewer instructions:

```
mulsaq_s.w.ph   $ac0, $t0, $t1 #compute
real part
dpaqx_s.w.ph    $ac1, $t0, $t1 #compute
imaginary part
```

The new instructions help minimize the data packing, unpacking, and shuffling that SIMD arithmetic often requires. The additional flexibility in choice of operand data types and registers improves the instruction set orthogonality (i.e., its regularity and clarity), which should help enable more effective compilers.

A few other notable additions in DSP ASE Rev 2 include:

- Increased support for bit-field manipulation, such as the ability to append or prepend two data values in a single register. These instructions are useful for encoding and decoding bitstreams that contain arbitrary length bit fields (such as in audio and video codecs). Revision 1 of the ASE included a set of instructions to speed up bitstream decoding; these new instructions are primarily useful for encoding. They may also be useful for saving the transition history during the add-compare-select loop of a Viterbi decoder.
- More flexible multiply, multiply-add, and multiply-subtract instructions for 32-bit operands, with more options for operand selection and choices of integer or fractional modes. 32-bit multiplication operations are useful, for example, in some speech and audio applications where precision higher than 16 bits is needed.

As mentioned earlier, the 74K core is backwards compatible with the 24KE core, and can execute 24KE core binaries without requiring recompilation. To take advantage of the additional instructions described above and achieve the maximum performance potential of the new core, however, software written for the 24KE core will need to be modified.

MIPS16E INSTRUCTIONS

Like the 24KE, the 74K core supports the MIPS16e compressed instruction set extensions. These extensions provide 16-bit instruction encodings of many standard MIPS32 instructions to reduce program memory requirements. (The MIPS16e extensions do not include compressed versions of DSP ASE instructions.) When loaded into the 74K core, MIPS16e instructions are expanded to 32 bits for execution. While the16-bit instruction encoding provides restricted functionality compared to 32-bit instruction encoding, it has the advantage that code memory requirements are reduced significantly, since twice as many instructions can be stored in the same space.

COREXTEND CUSTOM INSTRUCTIONS

In addition to the built-in instructions provided in the 74K core, the CorExtend build-time configuration option allows designers to add their own, custom instructions to the processor. These instructions can access the core registers (including the four accumulators), and can complete in one or more processor cycles. This capability allows designers to add functionality to the core to accelerate operations that present a performance bottleneck in the target application. ASIC designers using the 74K core for DSP applications may find this a useful way to accelerate a few key operations in critical inner loops.

Performance

The 74K core is a significant step up in performance from the 24KE, due to a combination of higher clock rates and a more powerful architecture. According to MIPS, the 74K core is expected to execute at a 25% higher clock rate than the 24KE core when implemented in the same fabrication process. MIPS estimates that the 74K core's out-of-order superscalar execution will provide an additional 20-30% performance boost, which BDTI believes is a reasonable estimate, with the speed-up primarily coming from the newer core's ability to execute load/stores in parallel with arithmetic operations.

According to MIPS, the combination of increased clock speed and out-of-order superscalar execution means that, in the same fabrication process, the 74K core will be approximately 60% faster than the 24KE on DSP inner loops. As mentioned above, the new instructions in DSP ASE Rev 2 may provide a modest additional speed-up; the amount of the speed-up will depend on the specific algorithm and implementation.

BDTI has not yet benchmarked the 74K core. However, BDTI has benchmarked its predecessor, the 24KE, using BDTI's suite of DSP algorithm kernel benchmarks, the BDTI DSP Kernel Benchmarks. Based on the 24KE's benchmark results and MIPS' projected 60% speed-up, BDTI estimates that a 1 GHz 74K core will offer signal processing speed that is comparable to that provided by mid-range DSP chips operating in the range of 600-700 MHz, such as Analog Devices Blackfin, or by more powerful DSPs operating in the range of 300 MHz, such as Freescale's SC1400. Because the 74K core has to operate at a higher clock speed to achieve performance similar to that of the DSP processors, however, it is likely to be less energy efficient when executing signal-processing code.

At this level of performance, the 74K core should be able to eliminate the need for a separate DSP processor in applications that have moderate signal processing demands; for example, the 74K core can almost certainly subsume the audio processing in some products (like set-top boxes) and may be able to subsume some of the video processing, depending on the resolution required and the codec being used.

Eliminating a separate DSP processor can help to reduce the system development effort and ease software development, since programmers will only need to learn and use one set of development tools. However, it may introduce other challenges. It can be tricky, for example, to ensure the robust real-time behavior of signal processing software when it runs on the same processor that's running a full-featured OS and a large body of other software. Designers will need to weigh the benefits and challenges of using a single core for both DSP and non-DSP workloads.

Development and Debug Tools

MIPS provides tools and hardware features to enable software development and in-system application debug for the 74K core. The MIPS Software Toolkit is MIPS' premium development environment, consisting of the usual array of development tools an application developer might expect, and is common to all MIPS cores. It supports C and assembly language software development using a GNU-based compiler-assembler-linker tool chain. The Toolkit includes the MIPS DSP libraries and the MIPSsim instruction set simulator, both described below. Solaris, Linux, and Windows development platforms are all supported.

According to MIPS, the 74K core C compiler does not currently use the DSP ASE instructions; instead, MIPS provides a library of DSP functions that have been optimized for the 74K core. These include a range of filters and transforms, along with several specialized functions used in H.264 video codecs, such as quarter-pixel motion compensation.

For high-performance DSP software development, a cycle-accurate simulator is very important. It allows performance-critical code to be accurately profiled and optimized. Cycle accuracy is particularly important in simulators for complex processors that have dynamic features such as superscalar execution, instruction reordering, branch prediction, and caches—all of which are present on the 74K core. According to MIPS, the current instruction set simulator for the 74K core is accurate to within 1-10%, depending on the specifics of the software being executed. MIPS expects the production version of the simulator to be within 1-3%. A 3% level of accuracy should be sufficient for developing optimized signal processing code, and is comparable to the accuracy of simulators provided for high-performance DSP processors with caches.

The 74K core includes build-time configuration options for different degrees of hardware debug support in the core, ranging from no debug support whatsoever, to support for complex hardware breakpoints and instruction and data tracing capabilities. These build-time configuration options provide the designer with some flexibility in making trade-offs between die size and power consumption on the one hand, and debug visibility and control on the other.

Basic hardware debug is enabled by choosing the EJTAG build-time configuration option. EJTAG is a proprietary JTAG interface extension that allows the designer to control debug events and single-stepping operations in the core. Associated with the EJTAG interface are build-time configuration options for hardware breakpoint support. Three different build-time options are available for hardware breakpoints, providing support for different numbers and combinations of events related to instruction and data memory addressing and pre-conditions for breakpoints. Breakpoint events can be used in different ways, such as initiating an exception handler for software-only debug handling, or for triggering instruction and data trace events.

Conclusions

The 74K core is significantly more powerful than earlier MIPS cores. Based on MIPS' performance estimates and BDTI's own analysis, we believe that the 74K core has sufficient signal processing horsepower to eliminate the need for a separate DSP processor in applications with moderately demanding signal processing requirements. While the core probably isn't powerful enough to perform high-resolution, computationally-demanding video processing, it should be capable of executing most audio processing and some less-demanding video processing.

The ability to execute legacy 24KE binaries without recompiling is an advantage, and MIPS' estimates of the performance improvements due to out-of-order superscalar execution are credible. The lack of sufficient data bandwidth to support the SIMD multiplications is a disadvantage, but in some cases this can be mitigated by algorithmic transformations. Overall, we expect that the 74K core's combination of CPU and signal processing capabilities will be attractive to SoC designers hoping to reduce the number of processors in their chips or add DSP-oriented capabilities to their existing MIPS-based products.