

AIEC / ARM966E-S™ Microprocessor Core (AIEC9)

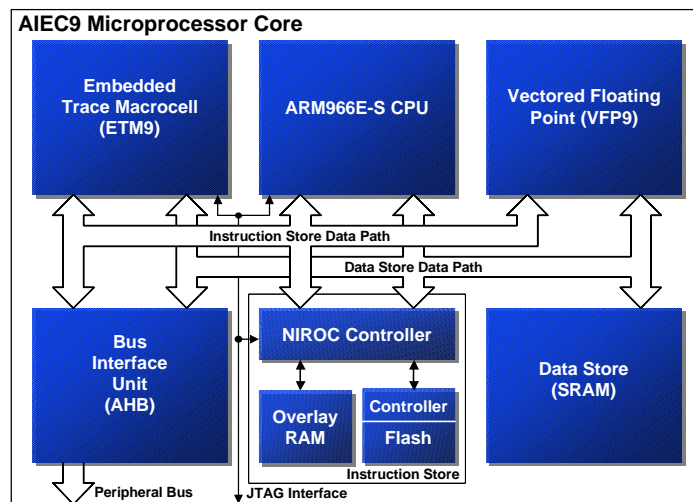
Product Overview

The AIEC / ARM966E-S™ Microprocessor Core (AIEC9) is a high-performance, low-cost 32-bit microprocessor core that is intended as a platform for custom automotive System on Chip (SoC) developments. The high-performance of the AIEC9 is in part due to the memory architecture, which has been optimized for real time control systems where deterministic performance is achieved through execution of instructions directly from flash. An interleaved flash controller allows zero wait state execution of sequential code even with comparatively slow flash. This memory architecture provides the highest degree of deterministic performance while system cost is reduced through elimination of a cache and associated cache controller. Additionally, the instruction store and data store memory sizes can be adjusted for optimal balance of cost and capacity. The memory architecture also supports modification of calibration tables stored in flash through the Non-Intrusive RAM Overlay Calibration (NIROC) circuit. Calibration tables are modified via the JTAG interface completely independent of the ARM966E-S™ CPU, leaving operation of a system under calibration completely unaltered from production operating conditions. A Vectored Floating Point unit (VFP9) is contained within the AIEC9 to accommodate high-performance floating-point algorithms, which enables the use of automatic code generation tools for model based system control methodologies. The AIEC9 also contains an Embedded Trace Macrocell (ETM9) that allows true, real-time software debug of embedded control systems where halting the processor to observe data and program flow is not possible. Finally, the AIEC9 uses the industry standard Advanced High-performance Bus (AHB) for interfacing to peripherals.

The AIEC9 is part of AIEC's modular library of components and it can be developed as a stand-alone microprocessor core or it can be integrated with peripherals and memory as part of a complete custom System on Chip (SoC).

Features

- ARM966E-S™ 32-Bit RISC CPU, 100MHz Operation Over Automotive Temperature Range (-40°C to 125°C)
- ARM966E-S™ as a Member of the ARM® Family of Microprocessors is Supported by the Industry's Largest Selection of Development Tools
- Code Compatibility With All ARM® Microprocessor Cores Safeguards Software Investment
- Code Compression On-The-Fly for High Speed / High Code Density Selection (Thumb® Instruction Set Compatible)
- Embedded ICE Software Debug Facilities through JTAG Interface
- DSP Extensions to Instruction Set Include Single Cycle MAC and Saturating Arithmetic
- IEEE 754 Compliant Floating Point (VFP9) Uses Vectored Instructions that Overlap Load/Store Operations for Superscalar Performance (200 FIR Filter MFlops @ 100MHz)
- Embedded Trace Macrocell Monitors Internal Busses and Passes Compressed Data to Trace Port for Real Time Hardware/Software Debug
- Embedded Flash Instruction Store and High Speed Static RAM Data Store User Configurable for Optimal Balance of Cost and Capacity
- Flash Memory Controller Pipelines Instructions for In-Line Code Execution at CPU Clock Frequency
- Non-Intrusive RAM Overlay Calibration (NIROC) Completely Independent of CPU
- Advanced High-performance Bus (AHB) Industry Standard Peripheral Interface
- AIEC9 is Technology Independent Allowing True Second Sourcing from Multiple Semiconductor Suppliers to Ensure Competitive Cost and Quality



AIEC9 Microprocessor Core Block Diagram

Functional Overview

The AIEC9 microprocessor core is designed specifically for real-time automotive embedded control applications where deterministic performance and low cost are of utmost importance. Deterministic performance is achieved through execution of instructions directly from flash as opposed to a cached system where page swaps introduce latencies. Low cost is maintained since high-speed static cache and the associated cache memory controller are not part of the memory architecture.

The AIEC9 is made up of the following functional blocks:

- ARM966E-S™ CPU,
- Vectored Floating Point (VFP9),
- Embedded Trace Macrocell (ETM9),
- Instruction Store,
- Data Store,
- Advanced High-performance Bus (AHB).

ARM966E-S™ CPU

At the heart of the AIEC9 is the ARM966E-S™ 32-bit RISC CPU. The ARM966E-S™ CPU implements the ARMv5TE instruction set and features an enhanced 16 x 32-bit multiplier capable of single cycle MAC operations, and 16-bit fixed point DSP instructions to accelerate signal processing algorithms and applications. In addition to the single cycle 16 x 32-bit multiply instruction, the DSP instruction set includes a 32 x 32-bit multiply, four fractional saturating arithmetic instructions, and a count leading zeros instruction for faster normalization and divide performance. Typical DSP throughput reaches 1MMAC/MHz. By combining DSP instructions with a traditional microprocessor instruction set, the ARM966E-S™ CPU provides numerous benefits over that of a two-processor solution including:

- Fast time to market with off-the-shelf solution (does not require system architecture and design as would a two processor system),
- Single processor eliminates inter-processor communications and synchronization,
- Application code does not need to be segmented between microprocessor and DSP,
- Single unified software development and debug environment.

Additional performance is gained using Harvard architecture with separate connections to the instruction and data memories. Both interfaces are capable of operating at the processor clock speed, which allows performance levels to reach 1.1MIPS/MHz.

The ARM966E-S™ CPU also includes a coprocessor interface that is used to connect the Vectored Floating Point (VFP9). The coprocessor

interface can also be used to connect application specific acceleration hardware.

From a system debug standpoint, the ARM966E-S™ offers Embedded In Circuit Emulation (EmbeddedICE™), which provides classical ICE functions such as real time address and data dependent breakpoints, single stepping, full control of the CPU, and access to the peripherals connected to the AHB interface. The EmbeddedICE™ function is implemented within the target silicon and is accessed via a JTAG port. The EmbeddedICE™ provides several benefits over that of traditional ICE systems (e.g., classical silicon emulators or resident monitor programs), which includes:

- Low cost solution that does not require dedicated emulation or “bond-out” microprocessor
- No target resources required (does not require internal RAM or ROM),
- No special hardware requirements such as ICE pod connectors or dedicated serial port,
- Uses existing boundary scan pins and does not require pin count overhead,
- Debug is performed at full processor speed.

Lastly, the ARM966E-S™ CPU supports code compression through the Thumb® instruction set. Thumb® instructions are compressed down to 16-bit op-codes, which allow code density improvements as high as 30% over that of the standard 32-bit op-codes. The trade-off between code space and performance can be made sub-routine by sub-routine with size critical routines compiled using Thumb® instructions and performance critical routines using standard 32-bit instructions.

Vectored Floating Point (VFP9)

The Vectored Floating Point (VFP9) is well suited to applications that have wide dynamic ranges, which are greater than the range of a standard microprocessor core. This is particularly true for automotive embedded control systems where large data ranges and small data values are predominant. The VFP9 also allows rapid development using technical computing tools (e.g., MatLab® and MATRIXx®) to directly model the system and quickly derive the application code.

Compared to standard integer emulation routines, the VFP9 provides fast processing owing to its use of low-latency, single-cycle throughput, single and double precision operations. The VFP9 includes novel short vector operations that allow performance processing with reduced instruction bandwidth. This delivers high-processing throughput with overlapping load and store operations, which allows a single precision floating-point Finite Impulse Response (FIR) tap to be computed in less than 1.5 cycles.

Embedded Trace Macrocell (ETM9)

The Embedded Trace Macrocell (ETM9) provides comprehensive real-time tracing and debugging capabilities for situations where traditional start/stop debug technology and logic analyzers are not capable of locating bugs. This allows developers to quickly locate difficult system bugs thus reducing time to market and development costs. The ETM9 facilitates debug by tracing the internal instruction and data buses in real-time at full clock speeds. Information is then compressed and transmitted to the Trace Port Analyzer. Once the trace has been captured, the debugger extracts the information from the trace port analyzer and de-compresses it to provide a full disassembly (with symbols) of the code that was executed. The debugger can also link this back to the original high-level source code, providing a visualization of how the code was executed on the target system. The features of the ETM9 include:

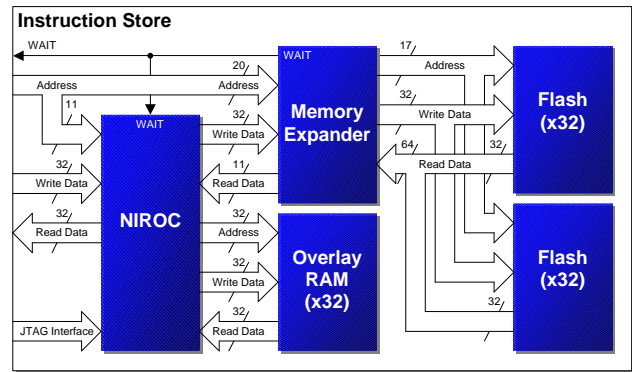
- Non-intrusive trace and debug,
- Trace capture of target system running at speed,
- Traces both instructions and data,
- Cycle-accurate trace,
- Time-stamping data for later analysis and profiling,
- Comprehensive trigger facility, allowing trace to be captured on complex sequential conditions,
- Comprehensive filter conditions for data capture,
- Debug foreground tasks while interrupts continue,
- Access memory without stopping the application,
- Very small memory footprint (approx 2K).

Instruction Store

The AIEC9 avoids a memory caching architecture, which introduces un-predictable latencies during page swapping. Instead, the AIEC9 executes instructions directly from flash avoiding the high cost of multi-layered memory systems and the non-determinism of cache misses.

Even though relatively slow flash is used as the instruction store, the AIEC9 still achieves a high level of performance, incurring no wait states while executing sequential code. For a branch, the AIEC9 incurs two wait states as compared to a cached system where tens of wait states are inserted during a page swap.

The Instruction Store also includes the Non-Intrusive RAM Overlay Calibration (NIROC) circuit to provide a real-time calibration solution. The NIROC is particularly useful in powertrain applications where large amounts of data are used to define the operating parameters of the engine. These data are stored in "calibration tables" in the instruction space. The values to be programmed into these tables are chosen during calibration. Traditional approaches to calibration have used parallel memory emulation technology, however with the integration of memory directly on chip, parallel memory emulation is no longer practical.



Instruction Store Block Diagram

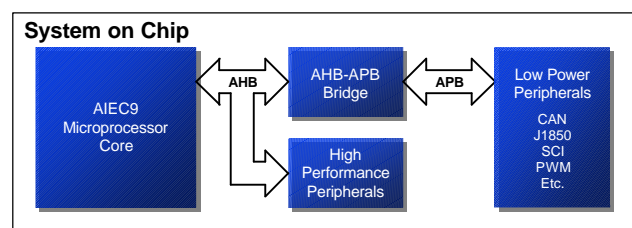
Other calibration approaches include serial port calibration protocols (CAN, SCI, etc.), but these require additional software to support the calibration activity and affect the operation of the system by using code space and processor throughput to support calibration. The NIROC macrocell provides a hardware solution for calibration, when used in conjunction with the JTAG interface and ETM9. Calibration using NIROC requires zero support code and incurs zero microprocessor overhead, leaving operation of a system under calibration completely unaltered from production operating conditions.

Data Store

The AIEC9 Data Store is implemented using a high-speed 32-bit wide (byte addressable) static RAM to support the maximum throughput capability of the ARM966E-S™ CPU. Up to 64MBytes of Data Store may be specified for a particular SoC, however smaller values are typically used to achieve the optimum balance between capacity and cost. Built In Self Test (BIST) is incorporated within the ARM966E-S™ for automatic testing of the data memory. In fact, background BIST of the Data Store can occur while application code is running.

Advanced High-performance Bus (AHB)

The AIEC9 microprocessor core has an Advanced High-performance Bus (AHB) interface for connecting peripherals. The AHB is normally used for connecting high-performance peripherals that require high-speed and high-throughput such as interrupt controllers, DMA controllers, and high-speed interfaces. The lower throughput peripherals that are accessed less frequently are bridged from the AHB to the Advanced Peripheral Bus (APB), which helps reduce system power consumption.



Peripheral Busing Block Diagram

Automotive Integrated Electronics Corporation (AIEC)
9034 N. 23rd Avenue Suite 13
Phoenix, AZ 85021
Phone: (602) 943-7499 Facsimile: (602) 861-1777
Email: sales@aiec.com <http://www.aiec.com>

Information furnished is believed to be accurate and reliable. However, AIEC assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties, which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of AIEC. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. AIEC products are not authorized for use as critical components in life support devices or systems without express written approval of AIEC.

© 2002 AIEC All Rights Reserved