ARM Glossary

AEG0014E (ID051812)

ARM Glossary

Release Information

The following changes have been made to this book.

Change history

Date	Issue	Confidentiality	Change
26 March 2010	A	Non-Confidential	New document.
27 September 2010	B	Non-Confidential	Second revision. Updated with new terms for the ARM architectures.
27 January 2011	C	Non-Confidential	Third revision. Updated with new terms for DS-5.
24 August 2011	D	Non-Confidential	Fourth revision. Updated by enhancements and closed defects.
17 May 2012	E	Non-Confidential	Fifth revision. Updated by enhancements and closed defects.

Proprietary Notice

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by ARM and the party that ARM delivered this document to.

Product Status

The information in this document is final, that is for a developed product.

Web Address

`http://www.arm.com`

Contents
ARM Glossary

Chapter 1
Using the ARM Glossary

This glossary is a collection of ARM-specific terminology from individual product documentation.

If you have comments on content, send an e-mail to `errata@arm.com`. Give:

- the title
- the number, AEG0014E
- the glossary item to which your comments apply
- a concise explanation of your comments.

ARM also welcomes general suggestions for additions and improvements.

Glossary

This glossary describes some of the technical terms used in ARM documentation.

AAPCS
See Procedure Call Standard for the ARM Architecture (AAPCS).

AArch32
The ARMv8 32-bit execution state, that uses 32-bit general purpose registers, and a 32-bit program counter (PC), stack pointer (SP), and link register (LR). AArch32 execution state provides a choice of two instruction sets, A32 and T32.

Operation in AArch32 state is compatible with ARMv7-A operation.

See also AArch64, A32, T32.

AArch64
The ARMv8 64-bit execution state, that uses 64-bit general purpose registers, and a 64-bit program counter (PC), stack pointer (SP), and exception link registers (ELR). AArch64 execution state provides a single instruction set, A64.

See also AArch32, A64.

A32
The instruction set used by an ARMv8 processor that is in AArch32 execution state. A32 is a fixed-width instruction set that uses 32-bit instruction encoding. It is compatible with the ARMv7 ARM instruction set.

See also AArch32, T32.

A64
The instruction set used by an ARMv8 processor that is in AArch64 execution state. A64 is a fixed-width instruction set that uses 32-bit instruction encoding.

See also AArch64, A32, T32.

ABI
See Application Binary Interface for the ARM Architecture (ABI).

Abort
Aborts occur when an illegal memory access causes an exception. The external memory system, or the hardware that manages the memory, can generate an abort. The hardware that generates the abort might be a *Memory Management Unit* (MMU) or a *Memory Protection Unit* (MPU).

See also External Abort, Data Abort and Prefetch Abort.

Abort model
Describes what happens to the processor state when a Data abort exception occurs. Different abort models behave differently with regard to load and store instructions that specify base register writeback.

Adaptive clocking
A technique where the debug interface hardware sends out a clock signal and then waits for the returned clock before generating the next clock pulse. This technique enables the run control unit in the debug hardware to adapt to differing signal drive capabilities and differing cable lengths.

Addressing mode
A method of generating the memory address that a load or store instruction uses.

The addressing modes mechanism can generate values for data-processing instructions to use as operands.

Advanced eXtensible Interface (AXI)
An AMBA bus protocol that supports:

- separate phases for address or control and data
- unaligned data transfers using byte strobes
- burst-based transactions with only start address issued
- separate read and write data channels
- issuing multiple outstanding addresses
- out-of-order transaction completion
- addition of register stages to provide timing closure.

The AXI protocol includes optional signaling extensions for low-power operation.

See also AXI Coherency Extensions (ACE).

Advanced High-performance Bus (AHB)
An AMBA bus protocol supporting pipelined operation, with the address and data phases occurring during different clock periods. This means the address phase of a transfer can occur during the data phase of the previous transfer. AHB provides a subset of the functionality of the AMBA AXI protocol.

See also Advanced Microcontroller Bus Architecture (AMBA) and AHB-Lite.

Advanced Microcontroller Bus Architecture (AMBA)
The AMBA family of protocol specifications is the ARM open standard for on-chip buses. AMBA provides solutions for the interconnection and management of the functional blocks that make up a *System-on-Chip* (SoC). Applications include the development of embedded systems with one or more processors or signal processors and multiple peripherals.

Advanced Peripheral Bus (APB)
An AMBA bus protocol for ancillary or general-purpose peripherals such as timers, interrupt controllers, UARTs, and I/O ports. It connects to the main system bus through a system-to-peripheral bus bridge that helps reduce system power consumption.

Advanced SIMD
An extension to the ARM architecture that provides *Single Instruction Multiple Data* (SIMD) operations on a bank of extension registers. If a floating-point extension is also implemented, the two extensions share a common extension register bank. The Advanced SIMD extension implements NEON technology, and is sometimes called NEON.

Advanced Trace Bus (ATB)
A bus used by trace devices to share CoreSight capture resources.

AEL *See* ARM Embedded Linux (AEL).

AHB *See* Advanced High-performance Bus (AHB).

AHB Access Port (AHB-AP)
An optional component of the DAP that provides an AHB interface to a SoC.

CoreSight supports access to a system bus infrastructure using the *AHB Access Port* (AHB-AP) in the *Debug Access Port* (DAP). The AHB-AP provides an AHB master port for direct access to system memory. Other bus protocols can use AHB bridges to map transactions. For example, you can use AHB to AXI bridges to provide AHB access to an AXI bus matrix.

See also Debug Access Port (DAP).

AHB Trace Macrocell (HTM)
A trace source that makes bus information visible. This information cannot be inferred from the processor using just a trace macrocell. HTM trace can provide:

- An understanding of multi-layer bus utilization.
- Software debug. For example, visibility of access to memory areas and data accesses.
- Bus event detection for trace trigger or filters, and for bus profiling.

See also Advanced High-performance Bus (AHB).

AHB-AP *See* AHB Access Port (AHB-AP).

AHB-Lite A subset of the full AMBA AHB protocol specification. It provides all of the basic functions required by the majority of AMBA AHB slave and master designs, particularly when used with a multi-layer AMBA interconnect.

Aligned A data item stored at an address that is divisible by the number of bytes that defines its data size is said to be aligned. Aligned doublewords, words, and halfwords have addresses that are divisible by eight, four, and two respectively. An aligned access is one where the address of the access is aligned to the size of each element of the access.

AMBA *See* Advanced Microcontroller Bus Architecture (AMBA).

APB *See* Advanced Peripheral Bus (APB).

APB Access Port (APB-AP)
An optional component of the *Debug Access Point* (DAP) that provides an APB interface to a SoC, usually to its main functional buses.

APB-AP *See* APB Access Port (APB-AP).

Application Binary Interface for the ARM Architecture (ABI)
A collection of specifications, some open and some specific to the ARM architecture, that regulate the inter-operation of binary code in a range of execution environments for ARM processors. The base standard specifies those aspects of code generation that must conform to a standard that supports inter-operation. It is aimed at authors and vendors of C and C++ compilers, linkers, and runtime libraries.

ARM Compiler for DS-5

ARM Compiler for DS-5 is a suite of tools, together with supporting documentation and examples, that you can use to write and build applications for the ARM family of processors. ARM Compiler for DS-5 supersedes RealView Compilation Tools.

See also armar, armasm, armcc, fromelf.

ARM Embedded Linux (AEL)
A version of embedded Linux OS ported to the ARM architecture.

ARM instruction
A word that specifies an operation for a processor in ARM state. ARM instructions must be word-aligned.

ARM profiler
A plug-in to the ARM Workbench Integrated Development Environment that provides non-intrusive analysis of embedded software over time, on targets running at frequencies which are typically as high as 250MHz. Targets can be *Real-Time System Models* (RTSMs) and hardware targets.

See also RealView Development Suite (RVDS).

ARM state
In ARM state the processor executes the ARM instruction set.

See also ARM instruction.

ARM TrustZone technology
The hardware and software that enable the integration of enhanced security features throughout a SoC. On an ARM processor, the Security Extensions implement the TrustZone hardware.

See also TrustZone Software.

ARM Workbench IDE (AWIDE)
ARM Workbench IDE is based around the Eclipse IDE, and provides additional features to support the ARM development tools provided in RVDS.

See also RealView Development Suite (RVDS).

armar
The ARM librarian which enables you to create libraries of files, such as object files.

See also Development Studio 5 (DS-5) and RealView Compilation Tools (RVCT).

armasm
The ARM assembler. This converts ARM assembly language into machine code.

See also Development Studio 5 (DS-5) and RealView Compilation Tools (RVCT).

armcc
The ARM compiler for C and C++ code.

See also Development Studio 5 (DS-5) and RealView Compilation Tools (RVCT).

Atomicity
Describes actions that appear to happen as a single operation. In the ARM architecture, atomicity refers to either single-copy atomicity or multi-copy atomicity. The *ARM Architecture Reference Manual* defines these forms of atomicity.

ArtiGrid
A power routing scheme, also referred to as Over The Cell.

See also Over The Cell (OTC).

ATB
See Advanced Trace Bus (ATB).

ATB bridge
A synchronous ATB bridge provides a register slice that helps timing closure by adding a pipeline stage. It also provides a unidirectional link between two synchronous ATB domains.

An asynchronous ATB bridge provides a unidirectional link between two ATB domains with asynchronous clocks. It supports connection of components with ATB ports in different clock domains.

See also Advanced Trace Bus (ATB).

ATPG
See Automatic Test Pattern Generation (ATPG).

Automatic Test Pattern Generation (ATPG)
The process of using a specialized software tool to automatically generate manufacturing test vectors for an ASIC design.

AWIDE *See* ARM Workbench IDE (AWIDE).

AXI *See* Advanced eXtensible Interface (AXI).

AXI Coherency Extensions (ACE)
The *AXI Coherency Extensions* (ACE) provide additional channels and signaling to an AXI interface to support system level cache coherency.

Back-annotation The process of applying timing characteristics from the implementation process onto a model.

Banked registers A register that has multiple instances. A property of the state of the device determines which instance is in use. For example the processor mode or security state might determine which instance is in use.

Base Platform Application Binary Interface (BPABI)
The base standard for the interface between executable files, such as dynamic shared objects and DLLs, and the systems that execute them.

Base porting layer A platform-dependent base driver software component that communicates with the Mali GPU. For example, the base porting layer controls the Mali GPU registers. You implement, or port, the base porting layer onto different target platforms.

Base register A register specified by a load or store instruction that is used as the base value for the address calculation for the instruction. Depending on the instruction, an offset can be added to or subtracted from the base register value to form the virtual address that is sent to memory.

Base register writeback
Writing back a modified value to the base register used in an address calculation.

Base Standard Application Binary Interface (BSABI)
See Application Binary Interface for the ARM Architecture (ABI).

BCD file *See* Board and Chip Definition (BCD) file.

Beat Alternative term for an individual transfer within a burst. For example, an INCR4 burst comprises four beats.

See also Burst.

Big-endian In the context of the ARM architecture, big-endian is defined as the memory organization in which:

- a byte or halfword at a word-aligned address is the most significant byte or halfword at that address
- a byte at a halfword-aligned address is the most significant byte in the halfword at that address.

See also Little-endian.

Board and Chip Definition (BCD) file
In the context of RealView Debugger, a BCD file enables you to define the memory map and memory-mapped registers for a target development board or processor. ARM provides various BCD files with RVDS for ARM development boards.

See also Board file and Debug configuration.

Board file
A debugger uses this term to refer to the top-level configuration file, normally called `rvdebug.brd`, that references one or more other configuration files. A board file contains:

- the Debug Configuration (connection-level) settings
- references to the Debug Interface configuration file that identifies the targets on the development platform
- references to any *Board and Chip Definition* (BCD) files assigned to a Debug Configuration.

See also Board and Chip Definition (BCD) file and Debug configuration.

Bounce
If a floating-point extension receives an instruction that it cannot process, it bounces the instruction. Bouncing the instruction causes an Undefined Instruction exception on the ARM processor. The extension can also use the bounce mechanism to handle some exceptions. The Undefined Instruction exception handler calls floating-point support code to respond to the bounce.

See also Trigger instruction and Exceptional state.

Boundary scan chain
A boundary scan chain is made up of serially-connected devices that implement boundary scan technology using a standard JTAG TAP interface. Each device contains at least one TAP controller containing shift registers that form the chain, connected between **TDI** and **TDO**, through which test data is shifted. A processor can contain several shift registers, enabling you to access selected parts of the device.

BPABI
See Base Platform Application Binary Interface (BPABI).

Branch folding
A technique where, on the prediction of a branch, the target instructions are completely removed from the instruction stream presented to the execution pipeline. Branch folding can significantly improve the performance of branches, and take the CPI for branches below one.

Branch phantom
Branch target instructions speculatively executed, in parallel with the main instruction stream, as a result of branch folding.

Branch prediction
The selection of a future execution path for instruction fetch. For example, after a branch instruction, the processor can choose to speculatively fetch either the instruction following the branch or the instruction at the branch target.

See also Prefetching.

Breakpoint
A debug event triggered by the execution of a particular instruction. It is specified by one or both of the address of the instruction and the state of the processor when the instruction is executed.

See also Watchpoint.

Breakpoint unit
In the context of an ARM debugger, a unit in a Chained breakpoint that combines with other breakpoint units to create a complex hardware breakpoint.

In an M-profile processor, a hardware debug component that can be part of the Flash Patch and Breakpoint unit.

See also Hardware breakpoint and Chained breakpoint.

BSABI
See Application Binary Interface for the ARM Architecture (ABI).

Burst
A group of transfers that form a single transaction. With AMBA protocols, only the first transfer of the burst includes address information, and the transfer type determines the addresses used for subsequent transfers.

See also Beat.

Byte lane strobe A signal that determines which byte lanes are active, or valid, in a data transfer. Each bit of this signal corresponds to eight bits of the data bus.

Byte swizzling Re-arranging the order of bytes in a word or halfword.

Byte-invariant In a byte-invariant system, the address of each byte of memory remains unchanged when switching between little-endian and big-endian operation. When a data item larger than a byte is loaded from or stored to memory, the bytes making up that data item are arranged into the correct order depending on the endianness of the memory access.

The ARM architecture supports byte-invariant systems in ARMv6 and later versions.

See also Word-invariant.

CADI The debug control and inspection API to a fast model.

Canonical Frame Address (CFA)
In *Debug With Arbitrary Record Format* (DWARF), this is an address on the stack specifying where the call frame of an interrupted function is located.

Captive thread Captive threads are all threads that can be brought under the control of RVDS. Special threads, called non-captive threads, are essential to the operation of *Running System Debug* (RSD) and so are not under debugger control.

See also Running System Debug (RSD).

Cast out *See* Victim.

CFA *See* Canonical Frame Address (CFA).

Chained breakpoint In the context of an ARM debugger, a complex breakpoint that comprises multiple hardware breakpoint units.

See also Breakpoint unit and Conditional breakpoint.

Chained tracepoint In the context of an ARM debugger, a complex tracepoint that comprises multiple tracepoint units.

See also Tracepoint unit and Tracepoint.

Characterized Designates a cell that includes timing data.

Clean A cache line that has not been modified while it is in the cache is said to be clean. To clean a cache is to write dirty cache entries into main memory.

See also Dirty.

Clock gating Gating a clock signal for a macrocell or functional block with a control signal and using the modified clock that results to control the operating state of the macrocell or block.

Clocks Per Instruction (CPI)
See Cycles Per Instruction (CPI).

CMM In the context of an ARM debugger, a scripting language provided for compatibility with other debuggers.

If you are writing new scripts, ARM recommends that you use the *GNU Debugger* (GDB) scripting commands because these offer more functionality in the ARM Debuggers.

Coherency *See* Memory coherency.

Cold reset
Also known as power-on reset. Starting the processor by turning power on. Turning power off and then back on again clears main memory and many internal settings. Some program failures can lock up the processor and require a cold reset to restart the system. In other cases, only a warm reset is required.

See also Warm reset.

Communications channel
The hardware used for communicating between the software running on the processor, and an external host, using the debug interface. When this communication is for debug purposes, it is called the *Debug Communications Channel* (DCC).

See also Debug Communications Channel (DCC).

Condition flags
The N, Z, C, and V bits of a *Program Status Register* (PSR).

See also Conditional execution, Current Program Status Register (CPSR), and Saved Program Status Register (SPSR).

Condensed Reference Format (CRF)
An ARM proprietary file format for specifying test vectors. Typically, ARM supplies a script to convert CRF format to *Verilog Reference Format* (VRF).

Condition code field
A four-bit field in an ARM instruction that specifies the condition under which the instruction executes.

See also Conditional execution and Condition flags.

Conditional breakpoint
A breakpoint that has one or more condition qualifiers assigned. The breakpoint is activated when all assigned conditions are met, and either stops or continues execution depending on the action qualifiers that are assigned. The condition normally references the values of program variables that are in scope at the breakpoint location.

See also Chained breakpoint and Software breakpoint.

Conditional execution
When a conditional instruction starts executing, if the condition flags indicate that the required condition is TRUE, the instruction executes normally. Otherwise, the instruction does nothing.

Context synchronization operation
A context synchronization operation is one of:

- the execution of an ISB instruction
- the taking of an exception
- the return from an exception.

The architecture requires a context synchronization operation to guarantee visibility of any change to a system control register.

Context switch
The saving and restoring of computational state when switching between different threads or processes.

Coprocessor
A processor, or conceptual processor, that supplements the main processor to carry out additional functions. The ARM architecture defines an interface to up to 16 coprocessors, CP0-CP15. Coprocessors CP8-CP15 are reserved for use by ARM and:

- CP15 instructions access the System Control processor
- CP14 instructions access control registers for debug, trace, and execution environment features

- CP10 and CP11 instruction space is for floating-point and Advanced SIMD instructions if supported.

Core module
In the context of an ARM Integrator development board, an add-on development board that contains an ARM processor and local memory. Core modules can run standalone, or can be stacked onto Integrator development boards.

See also Integrator.

Core register
The ARM core registers comprise:
- 13 general-purpose registers, R0 to R12, that software can use for processing
- SP, the Stack Pointer, that can also be referred to as R13
- LR, the Link Register, that can also be referred to as R14
- PC, the Program Counter, that can also be referred to as R15.

In some situations, software can use SP, LR, and PC for processing. The instruction descriptions include any constraints on the use of SP, LR, and PC.

Core reset
See Warm reset.

CoreSight
ARM on-chip debug and trace components, that provide the infrastructure for monitoring, tracing, and debugging a complete system on chip.

See also CoreSight ECT and CoreSight ETM.

CoreSight ECT
See Embedded Cross Trigger (ECT).

CoreSight ETB
See Embedded Trace Buffer (ETB).

CoreSight ETM
See Embedded Trace Macrocell (ETM).

CPI
Cycles or clocks per instruction.

See also Cycles Per Instruction (CPI).

CPSR
See Current Program Status Register (CPSR).

CRF
See Condensed Reference Format (CRF).

Cross Trigger Interface (CTI)
Part of an *Embedded Cross Trigger* (ECT) device. In an ECT, the CTI provides the interface between a processor or ETM and the CTM.

Cross Trigger Matrix (CTM)
In an ECT device, the CTM combines the trigger requests generated by CTIs and broadcasts them to all CTIs as channel triggers.

CTI
See Cross Trigger Interface (CTI).

CTM
See Cross Trigger Matrix (CTM).

Current Program Status Register (CPSR)
The register that holds the current processor status.

See also Program Status Register (PSR) and Saved Program Status Register (SPSR).

Cycles Per Instruction (CPI)
A measure of the number of computer instructions that can be performed in one clock cycle. It is also called clocks per instruction. You can use this value to compare the performance of different processors that implement the same instruction set. The lower the value, the better the performance.

DA
See Debug Agent.

DAP *See* Debug Access Port (DAP).

Data Abort An indication from a memory system to the processor of an attempt to access an illegal data memory location.

See also Abort, External Abort and Prefetch Abort.

Data breakpoint A hardware breakpoint that activates when an access to a specified location meets a set of specified conditions. The conditions can include a check for a specific data value being accessed at the given location.

See also Chained breakpoint and Conditional breakpoint.

Data Timing Module (DTM)
In the context of physical IP, a data timing module that synchronizes incoming and outgoing data. The DTM is a component of PHY to include I/Os and PLL.

Data-active write transaction
A transaction that has completed the address transfer or leading write data transfer, but has not completed all of its data transfers.

DBGTAP *See* Debug Test Access Port (DBGTAP).

DCC *See* Debug Communications Channel (DCC).

Debug Access Port (DAP)
A block that acts as a master on a system bus and provides access to the bus from an external debugger.

Debug Agent In RealView Debugger, the Debug Agent provides target-side support for *Running System Debug* (RSD). The Debug Agent can be a thread or be built into the RTOS. The Debug Agent and RealView Debugger communicate with each other using the DCC. This passes data between the debugger and the target using a hardware debug interface, without stopping the program or entering debug state.

See also Running System Debug (RSD) and Debug Communications Channel (DCC).

Debug Communications Channel (DCC)
A channel that an ARM debugger uses to transfer data to or from debug logic in the target. It can do this without stopping the program flow or entering Debug state, but can also use the DCC while the target is in Debug state. The channel is part of the debug register interface of the target.

Debug configuration In the context of an ARM debugger, a debug configuration defines a debugging environment for the development platform that is accessed through a particular Debug Interface. Multiple debug configurations can be created for a debug interface, each providing a separate debugging environment to different development platforms, or different debugging environments to the same development platform.

All debug configurations are stored in the main debugger board file. Each configuration might reference one or more BCD files.

See also Board file and Target.

Debug illusion The view of the software being debugged that a debugger presents to its user. The features of the debug illusion include:

- Mapping between assembler code and source code, including displaying assembler and source code simultaneously if required.
- Support for source-level stepping and breakpoints.
- Visibility of the source-level function call stack, even when called functions are generated inline.

- Display of variable values and structure field values, even when these values migrate between various locations. This includes displaying registers and the stack.

Debug interface
In the context of RealView Debugger, the Debug interface identifies the targets on your development platform, and provides the mechanism that enables RealView Debugger to communicate with those targets. The Debug interface corresponds directly to a piece of hardware or a software simulator.

See also Debug configuration and Target.

Debug Test Access Port (DBGTAP)
A debug control and data interface based on IEEE 1149.1 JTAG *Test Access Port* (TAP). The interface has four or five signals.

Debugger
A debugging system that includes a program, used to detect, locate, and correct software faults, together with custom hardware that supports software debugging.

Default NaN mode
In floating-point operation, a mode in which all operations that result in a NaN return the default NaN, regardless of the cause of the NaN result. This mode is compliant with the IEEE 754 standard but implies that all information contained in any input NaNs to an operation is lost.

See also NaN.

Denormalized value
See Subnormal value.

Design Simulation Model (DSM)
A functional simulation model of the device derived from the *Register Transfer Level* (RTL) but that does not reveal its internal structure. The DSM does not model any features added during synthesis such as internal scan chains.

Development platform
Contains the components, either hardware or simulated, that you use to develop your application. It can include:

- a development board, such as an Integrator/CP
- peripherals
- one or more processors that implement an ARM processor architecture
- CoreSight components
- one or more *Digital Signal Processors* (DSPs).

See also CoreSight.

Device
In the context of an ARM debugger, a component on a target containing the application that you want to debug.

See also Target.

Development Studio 5 (DS-5)
The suite of software development tools, together with supporting documentation and examples, that enable you to write and debug applications for the ARM family of processors. DS-5 supersedes RealView Development Suite.

See also RealView Development Suite (RVDS).

Device Validation Suite (DVS)
Use this set of tests to check the functionality of a device against that defined in the Technical Reference Manual.

Direct-mapped cache
A one-way set-associative cache. Each cache set consists of a single cache line, so cache look-up selects and checks a single cache line.

Dirty
A line in a write-back cache that has been modified while it is in the cache. Typically, a cache line is marked as dirty by setting the dirty bit to 1.

See also Clean.

DNM
See Do-Not-Modify (DNM).

Do-Not-Modify (DNM)
A value that must not be altered by software. DNM fields read as UNKNOWN values, and must only be written with the value read from the same field on the same processor.

Doubleword
A 64-bit data item. Doublewords are normally at least word-aligned in ARM systems.

Doubleword-aligned
A data item having a memory address that is divisible by eight.

Draw mode
In the context of graphics processing, one of the different ways to specify the primitives to draw. These different ways are called draw modes. The primitives can be specified individually or as a connected strip or fan. They can also be either:

- non-indexed, meaning that vertices are passed in a vertex array and processed in order
- indexed, meaning that vertices are passed as indices into a vertex array.

DS-5 Debugger
An ARM software development tool that enables you to make use of a debug agent to examine and control the execution of software running on a debug target. It is fully integrated into Eclipse for DS-5.

See also Eclipse for DS-5.

DSM
See Design Simulation Model (DSM).

DTM
See Data Timing Module (DTM).

DVS
See Device Validation Suite (DVS).

Early-Z
A Z-testing scheme that performs the actual Z-test before texturing or fragment shading when it is safe to do so. This increases the performance of the Mali GPU and reduces the required bandwidth.

Eclipse for DS-5
Eclipse for DS-5 is based around the Eclipse IDE, and provides additional features to support the ARM development tools provided in DS-5.

See also Development Studio 5 (DS-5).

ECT
See Embedded Cross Trigger (ECT).

EGL
See Embedded-System Graphics Library (EGL).

Embedded assembler
Embedded assembler is assembler code that is included in a C or C++ file, and is separate from other C or C++ functions.

Embedded Cross Trigger (ECT)
A modular system that supports the interaction and synchronization of multiple triggering events with an SoC. It comprises:

- *Cross Trigger Interface* (CTI)
- *Cross Trigger Matrix* (CTM).

Embedded Trace Buffer (ETB)
A Logic block that extends the information capture functionality of a trace macrocell.

Embedded Trace Macrocell (ETM)
A hardware macrocell that, when connected to a processor, outputs trace information on a trace port. The ETM provides processor driven trace through a trace port compliant to the ATB protocol. An ETM always supports instruction trace, and might support data trace.

EmbeddedICE logic An on-chip logic block that provides TAP-based debug support for an ARM processor. It is accessed through the DAP on the ARM processor.

EmbeddedICE-RT Hardware provided by an ARM processor to aid debugging in real-time.

Embedded-System Graphics Library (EGL)
A standardized set of functions that communicate between graphics software, such as OpenGL ES or OpenVG drivers, and the platform-specific windowing system that displays the image.

Emulator In the context of target connection hardware, an emulator provides an interface to the pins of a real processor. It emulates the pins to the external world, and enables you to control or manipulate signals on those pins.

Endianness The scheme that determines the order of successive bytes of a data word when it is stored in memory.

See also Big-endian and Little-endian.

ESSL *See* OpenGL ES Shading Language (ESSL).

ESSL compiler The compiler that translates shaders written in ESSL, into binary code for the shader units in the Mali GPU. There are two versions of the ESSL compiler:

- the on-target compiler
- the offline compiler.

ETB *See* Embedded Trace Buffer (ETB).

ETM *See* Embedded Trace Macrocell (ETM).

ETV *See* Extended Target Visibility (ETV).

Event In an ARM trace macrocell:

Simple An observable condition that a trace macrocell can use to control aspects of a trace.

Complex A boolean combination of simple events that a trace macrocell can use to control aspects of a trace.

Exception A mechanism to handle a fault, error event, or external notification. For example, exceptions handle external interrupts and undefined instructions.

Exception vector A fixed address that contains the address of the first instruction of the corresponding exception handler.

Exceptional state In floating-point operation, if the floating-point coprocessor detects an exceptional condition, the VFP extension sets the FPEXC bit and loads a copy of the exceptional instruction to the FPINST register. When in the exceptional state, the issue of a trigger instruction to the floating-point extension causes a bounce.

See also Bounce and Trigger instruction.

Execution vehicle A part of the debug target interface that processes requests from the client tools to the target.

See also Debug interface.

Execution view The address of regions and sections after the image is loaded into memory and started execution.

See also Scatter-loading and Load view.

Explicit access A read from memory, or a write to memory, generated by a load or store instruction executed by the processor. Reads and writes generated by hardware translation table accesses are not explicit accesses.

Extended Target Visibility (ETV)
Extended Target Visibility enables RealView Debugger to access features of the underlying target such as, for example, chip-level information provided by the hardware manufacturer or SoC designer.

eXtensible Verification Component (XVC)
A model that provides system or device stimulus and monitor responses.

See also XVC Test Scenario Manager.

External Abort An abort generated by the external memory system.

See also Abort and Data Abort.

Fast Context Switch Extension (FCSE)
An extension to the ARM architecture that modifies the behavior of the memory system. It enables multiple programs running on the processor to use identical address ranges, while ensuring that the addresses they present to the rest of the memory system differ.

From ARMv6, ARM deprecates use of the FCSE. The FCSE is optional in ARMv7, and obsolete from the ARMv7 Multiprocessing Extensions.

Fast Models from ARM
Instruction-accurate models that enable you to perform early software development on ARM systems. You can use the models, together with ARM Profiler and an ARM debugger, to optimize and debug your applications early in the development cycle.

See also ARM profiler *and* DS-5 Debugger.

Fault An abort generated by the memory system, for example by the *Memory Management Unit* (MMU) or *Memory Protection Unit* (MPU).

FCSE *See* Fast Context Switch Extension (FCSE).

FIQ FIQ interrupt. **nFIQ** is one of two interrupt signals on many ARM processors.

See also IRQ.

Fixed-function pipeline
In the context of graphics processors, a process that uses standard functions to draw graphics on fixed-function graphics processors, such as the Mali-55 GPU. The fixed-function pipeline is a requirement of OpenGL ES 1.1.

Flash Patch and Breakpoint unit (FPB)
In an ARM M-profile processor, an FPB can:

- remap sections of ROM, typically Flash memory, to regions of RAM
- set breakpoints on code in ROM.

It can be used for debug, and to provide a code or data patch to an application that requires field updates to a product ROM.

Flat address mapping A memory system where the physical address for every access is equal to its virtual address.

Flush-to-zero mode In floating-point operation, a special processing mode that optimizes the performance of some floating-point algorithms by replacing the denormalized operands and intermediate results with zeros, without significantly affecting the accuracy of their final results.

Formatter In an ETB or TPIU, an internal input block that embeds the trace source ID in the data to create a single trace stream.

FPB *See* Flash Patch and Breakpoint unit (FPB).

Fragment In the context of graphics processors, a fragment consists of all data, such as depth, stencil, texture, and color information, required to generate a pixel in the framebuffer. A pixel is usually composed of several fragments.

Fragment processor A programmable processor that performs rendering operations to produce a final image for display. The fragment processor receives completed vertex data from the vertex processor and then runs fragment shader programs.

The fragment processor was originally known as a pixel processor.

Fragment shader A program running on the fragment processor that calculates the color and other characteristics of each fragment.

fromelf The ARM image conversion utility. This accepts ELF format input files and converts them to a variety of output formats. fromelf can also generate text information about the input image, such as code and data size.

See also RealView Compilation Tools (RVCT).

Fully-associative cache
A cache that has only one cache set, that consists of the entire cache.

See also Direct-mapped cache.

G-POP A POP that includes components for an ARM MALI Graphics Processor.

See also POP.

General-purpose register
See Core register.

GPU *See* Graphics Processor Unit (GPU).

Graphics application A custom program that executes in the Mali graphics system and displays content in a framebuffer for transfer to a display.

Graphics driver A software library implementing OpenGL ES or OpenVG, using graphics accelerator hardware.

See also OpenGL ES driver and OpenVG driver.

Graphics Processor Unit (GPU)
A hardware accelerator for graphics systems using OpenGL ES and OpenVG. The Mali-200, Mali-300, and Mali-400 MP GPUs comprise of a vertex processor and one or more fragment processors. Mali-T600 series GPUs consist of one or more shader cores that can execute vertex or fragment shaders.

Half-rate clocking In an ARM trace macrocell, dividing the trace clock by two so that the TPA can sample trace data signals on both the rising and falling edges of the trace clock. The primary purpose of half-rate clocking is to reduce the signal transition rate on the trace clock of an ASIC for very high-speed systems.

Halfword A 16-bit data item. Halfwords are normally halfword-aligned in ARM systems.

Halfword-aligned A data item having a memory address that is divisible by 2.

Halted System Debug (HSD)
Means that a target can only be debugged when it is not running. With the target stopped, RealView Debugger presents OS awareness information by reading and interpreting target memory.

See also Running System Debug (RSD).

Halting debug-mode In ARM A-profile and R-profile processors, one of two mutually exclusive debug modes. In Halting debug-mode, processor execution halts when a breakpoint or watchpoint is encountered. You can use the debug interface to examine and alter all processor state, coprocessor state, memory, input and output locations.

See also Monitor debug-mode.

Hardware breakpoint A breakpoint that is implemented using non-intrusive hardware. Hardware breakpoints are the only method of halting execution on a specific instruction when the instruction is located in *Read Only Memory* (ROM) or Flash.

See also Chained breakpoint and Data breakpoint.

High registers Core registers R8-R12, SP, LR, and PC. Some Thumb instructions cannot access these registers..

See also Core register.

High vectors One of two possible locations for exception vectors. The high vector address range is near the top of the address space, rather than at the bottom.

Hint instruction A hint instruction provides information that the hardware can take advantage of.

Hit-Under-Miss (HUM) A buffer that enables a cache access to hit, even though there has been a data miss in the cache.

Host A computer that provides data and other services to another computer. Especially, a computer providing debugging services to a target being debugged.

HSD *See* Halted System Debug (HSD).

HTM *See* AHB Trace Macrocell (HTM).

HUM *See* Hit-Under-Miss (HUM).

ICE Extension Unit A hardware extension to the EmbeddedICE logic that provides more breakpoint units.

IEEE 1149.1 The IEEE Standard that defines TAP. Commonly referred to as JTAG.

See *IEEE Std 1149.1-1990 IEEE Standard Test Access Port and Boundary-Scan Architecture* specification available from the IEEE Standards Association, `http://standards.ieee.org`.

IGN An abbreviation for Ignore, when describing the behavior of a register or memory access.

Illegal instruction *See* Undefined.

Immediate values Values that are encoded directly in the instruction and used as numeric data when the instruction is executed. Many ARM and Thumb instructions can be used with an immediate argument.

IMPLEMENTATION DEFINED

Behavior that is not defined by the architecture, but is defined and documented by individual implementations.

When IMPLEMENTATION DEFINED appears in body text, it is always in SMALL CAPITALS.

See also IMPLEMENTATION SPECIFIC.

IMPLEMENTATION SPECIFIC

Behavior that is not architecturally defined, and might not be documented by an individual implementation. Used when there are a number of implementation options available and the option chosen does not affect software compatibility.

When IMPLEMENTATION SPECIFIC appears in body text, it is always in SMALL CAPITALS.

See also IMPLEMENTATION DEFINED.

Imprecise tracing
In an ARM trace macrocell, a filtering configuration where instruction or data tracing can start or finish earlier or later than expected. Most imprecise cases cause tracing to start or finish later than expected.

For example, if **TraceEnable** logic is configured to use a counter so that tracing begins after the fourth write to a location in memory, the instruction that caused the fourth write is not traced, although subsequent instructions are. This is because the use of a counter in the **TraceEnable** configuration always results in imprecise tracing.

In-Circuit Emulator
A device enabling access to and modification of the signals of a circuit while that circuit is operating.

Index register
A register specified in some load or store instructions. The value of this register is used as an offset to be added to or subtracted from the base register value to form the virtual address that is sent to memory. Some instruction forms permit the index register value to be shifted before the addition or subtraction.

Input section
Contains code or initialized data or describes a fragment of memory that must be set to zero before the application starts.

Instruction breakpoint
A location in the image containing an instruction that, if executed, activates a breakpoint. The breakpoint activation can be delayed by assigning condition qualifiers, and subsequent execution of the image is determined by any actions assigned to the breakpoint.

See also Conditional breakpoint and Software breakpoint.

Instruction Set System Model (ISSM)
In the context of RVDS, a set of models that simulate the ARM Cortex family of processors. These models are provided with RVDS.

See also Real Time System Model (RTSM) and Simulator.

Instruction Synchronization Barrier (ISB)
An operation to ensure that any instruction that comes after the ISB operation is fetched only after the ISB has completed.

Instrumentation trace
A component for debugging real-time systems through a simple memory-mapped trace interface. It providing `printf` style debugging.

Integrator
A range of ARM hardware development platforms. Core modules are available that contain the processor and local memory.

See also Core module.

Intelligent Energy Manager
An energy manager solution consisting of both software and hardware components that function together to prolong battery life in an ARM processor based device.

Intermediate Physical Address (IPA)
In an implementation of virtualization, the address to which a Guest OS maps a virtual address.

See also Virtual Address (VA) and Physical Address (PA).

Intermediate result
In floating-point operation, an internal format used to store the result of a calculation before rounding. This format can have a larger exponent field and fraction field than the destination format.

Internal scan chain
A series of registers connected together to form a path through a device, used during production testing to import test patterns into internal nodes of the device and export the resulting values.

Interworking
A method of working that permits branches between software using the ARM and Thumb instruction sets.

Invalidate Marking a cache line as being not valid. This must be done whenever the line does not contain a valid cache entry. For example, after a cache flush all lines are invalid.

IPA *See* Intermediate Physical Address (IPA).

IRQ IRQ interrupt. **nIRQ** is one of two interrupt signals on many ARM processors.

See also FIQ.

ISB *See* Instruction Synchronization Barrier (ISB).

IT block A block of up to four instructions following a Thumb If Then (`IT`) instruction. Each instruction in the block is conditional. The condition for each instruction is either the same as or the inverse of the condition specified by the `IT` instruction.

Jazelle architecture The Jazelle architecture extends the processor architecture to support operation of a *Java Virtual Machine* (JVM) that provides direct execution of some Java bytecodes.

Jazelle DBX An ARM technology for direct bytecode execution of Java bytecodes in hardware.

Jazelle Runtime Compilation Target (RCT)
On an ARM processor, A modification of the Thumb instruction set to make it a better target for code generated at runtime. This is also called the Thumb Execution Environment (ThumbEE).

See also ThumbEE instruction.

Jazelle state In Jazelle state the processor executes Java bytecodes as part of a *Java Virtual Machine* (JVM).

See also ARM state and Thumb state.

Jazelle Technology Enabling Kit (JTEK)
A kit containing source code for integration with a Java Virtual Machine to enable Jazelle DBX on an ARM-based host platform.

Job object In the context of graphics processing, a Mali job system component that provides jobs with required content for Mali GPU execution.

Job system back-end In the context of graphics processing, a job system component that shares some priority handling, but mainly requests jobs from queues and sends job requests to the Mali GPU.

JTAG *See* Joint Test Action Group (JTAG).

Joint Test Action Group (JTAG)
An IEEE group focussed on silicon chip testing methods. Many debug and programming tools use a *Joint Test Action Group* (JTAG) interface port to communicate with processors.

See *IEEE Std 1149.1-1990 IEEE Standard Test Access Port and Boundary-Scan Architecture* specification available from the IEEE Standards Association, `http://standards.ieee.org`.

JTAG Access Port (JTAG-AP)
An optional component of the DAP that provides debugger access to on-chip scan chains.

JTAG interface unit In the context of ARM RealView tools, a protocol converter that converts low-level commands from RVDS debuggers into JTAG signals to the processor, for example to the EmbeddedICE logic and the ETM.

See also Development Studio 5 (DS-5) and RealView ICE.

JTAG-AP *See* JTAG Access Port (JTAG-AP).

JTAG-DP *See* Debug Access Port (DAP).

JTEK *See* Jazelle Technology Enabling Kit (JTEK).

K Virtual Machine (KVM)
A small implementation of a Java Virtual Machine. It was originally derived from the Sun Spotless Virtual Machine.

KVM
See K Virtual Machine (KVM).

Little-endian
In the context of the ARM architecture, little-endian is defined as the memory organization in which the most significant byte of a word is at a higher address than the least significant byte.

See also Big-endian.

Load view
The address of regions and sections when the image has been loaded into memory but has not yet started execution.

See also Execution view and Scatter-loading.

Load/store architecture
A processor architecture where data-processing operations only operate on register contents, not directly on memory contents. The ARM architecture is a Load/Store architecture.

Mali DDK
A set of drivers, typically for AEL or Android, that enable communication with the Mali GPU. These drivers are available as normal or instrumented versions.

Mali MMU
A full-featured *Memory Management Unit* (MMU) that is present on Mali GPUs.

Memory coherency
A memory system is coherent if the value read by a data read or instruction fetch is always the value that was most recently written to that location. Memory coherency is difficult when the memory system includes multiple possible physical locations, such as main memory and at least one of a write buffer or one or more caches.

Memory hint
See Hint instruction.

Memory Management Unit (MMU)
Provides detailed control of the memory system. Most of the control uses translation tables that are held in memory.

Memory Protection Unit (MPU)
A hardware unit that controls a limited number of protection regions in memory.

MMU
See Memory Management Unit (MMU).

Model manager
A software control manager that handles the event transactions between the model and simulator.

Modified Virtual Address (MVA)
The address produced by the FCSE that is sent to the rest of the memory system to be used in place of the normal virtual address.

If the FCSE is absent or disabled, the MVA and the *Virtual Address* (VA) have the same value. From ARMv6, ARM deprecates any use of the FCSE. The FCSE is optional in the unextended ARMv7 architecture, and obsolete from the introduction of the Multiprocessing Extensions.

See also Fast Context Switch Extension (FCSE).

Monitor debug-mode
In ARM A-profile and R-profile processors, one of two mutually exclusive debug modes. In Monitor debug-mode, a debug event generates a debug exception, that is taken as a Prefetch Abort or Data Abort exception. Breakpoints and watchpoints are examples of debug events.

See also Halting debug-mode.

MPCore
An integrated *Symmetric Multiprocessor System* (SMP) or *Asymmetric Multiprocessor System* (AMP) with multiple processors in a single device.

MPU *See* Memory Protection Unit (MPU).

Multi-ICE A JTAG-based tool for debugging embedded systems.

Multi-layer interconnect
An interconnect scheme similar to a cross-bar switch. Each master on the interconnect has a direct link to each slave that is not shared with other masters. This means each master can process transfers in parallel with other masters. Contention in a multi-layer interconnect only occurs at a payload destination, typically a slave.

Multi-master AHB Typically a shared, not multi-layer, AHB interconnect scheme. More than one master connects to a single AMBA AHB link. In this case, the bus is implemented with a set of full AMBA AHB master interfaces. Masters that use the AMBA AHB-Lite protocol must connect through a wrapper to supply full AMBA AHB master signals to support multi-master operation.

MVA *See* Modified Virtual Address (MVA).

NaN Not a number. In floating-point operation, NaNs are special floating-point values that can be used when neither a numeric value nor an infinity is appropriate. NaNs can be either:

- quiet NaNs that propagate through most floating-point operations
- signaling NaNs that cause Invalid Operation floating-point exceptions.

For more information, see the IEEE 754 standard.

NEON The Advanced SIMD Extension implements this technology.

See also Advanced SIMD.

Normal and Secure Worlds
In software descriptions of ARM processor operation, effectively two virtual processors that run on a single physical processor. The Secure World processes operations that are security-critical, and passes non security-critical operations to the Normal World.

See also Secure monitor.

Normal World *See* Normal and Secure Worlds.

nSRST Abbreviation of *System Reset*. The electronic signal that causes the target system other than the TAP controller to be reset. This signal is known as **nSYSRST** in some documentation.

See also nTRST and Joint Test Action Group (JTAG).

nTRST Abbreviation of *TAP Reset*. The electronic signal that causes the target system TAP controller to be reset. This signal is known as **nICERST** in some documentation.

See also nSRST and Joint Test Action Group (JTAG).

Offline Compiler A command line tool that translates vertex shaders and fragment shaders written in the ESSL into binary vertex shaders and binary fragment shaders that you can link and run on the Mali GPU.

Offset addressing Addressing where the memory address is formed by adding an offset to, or subtracting an offset from, a base register value.

On-target compiler A component of the Mali GPU OpenGL ES 2.0 driver that translates shader source files provided by the graphics application, into binary shader code, at runtime.

OpenGL ES driver Part of a driver stack that translates OpenGL ES API commands into data and instructions for the Mali GPU. Only the device driver controls the Mali GPU directly.

OpenGL ES Shading Language (ESSL)
A programming language used to create custom shader programs that can be used in a programmable pipeline, on the Mali GPU. You can also use pre-defined library shaders, written in ESSL.

OpenVG driver Part of a driver stack that translates OpenVG API commands into data and instructions for the Mali GPU. Only the device driver controls the Mali GPU directly.

OS-awareness OS-awareness is a feature provided by RealView Debugger that enables you to:

- debug applications running on an embedded OS development platform, such as a *Real-Time Operating System* (RTOS)
- present thread information and scope some debugging operations to specific threads.

OTC *See* Over The Cell (OTC).

Output section A contiguous sequence of input sections that have the same RO, RW, or ZI attributes. The sections are grouped together in larger fragments called regions. The regions are grouped together into the final executable image.

See also Region.

Over The Cell (OTC) A power routing scheme, also referred to as ArtiGrid.

See also ArtiGrid.

PA *See* Physical Address (PA).

PCH *See* PreCompiled Header (PCH).

Penalty The number of cycles in which no useful Execute stage pipeline activity can occur because an instruction flow is different from that assumed or predicted.

Physical Address (PA) The address that identifies a location in physical memory.

PISMO *See* Platform Independent Storage Module (PISMO).

Platform Independent Storage Module (PISMO)
Memory specification for plug-in memory modules.

POP A performance optimization package for the implementation of an ARM processor using ARM Artisan optimized logic and memory physical IP.

See also G-POP.

Power Management Module (PMM)
In the context of graphics processors, a software routine that tracks the hardware blocks which can be enabled or disabled to reduce power. The PMM can control a specialized hardware unit, or a third-party power-management device, to power up or down each processor separately.

Power-on reset *See* Cold reset.

PreCompiled Header (PCH)
A header file that is precompiled. This avoids the compiler having to compile the file each time it is included by source files.

Prefetch Abort An indication from the internal or external memory system to the processor that an instruction has been fetched from an illegal memory location. An exception is taken only if the processor attempts to execute the instruction. No exception is taken if the processor does not execute an instruction prefetched from a faulting memory location.

See also Data Abort and Abort.

Prefetching The process of fetching instructions from memory before the instructions that precede them have finished executing. Prefetching an instruction does not mean that the instruction must be executed.

Primitive In the context of graphics processors, a basic element that the Mali GPU uses, with other primitives, to generate images.

See also Vertex.

Procedure Call Standard for the ARM Architecture (AAPCS)
Defines how registers and the stack are used for subroutine calls.

Profiling In the context of RealView Trace, the accumulation of statistics during execution of a program to measure performance or to determine critical areas of code.

Program Counter (PC)
In an ARM processor, core register R15.

Program Flow Trace (PFT)
The *Program Flow Trace* (PFT) architecture assumes that any trace decompressor has a copy of the program being traced, and generally outputs only enough trace for the decompressor to reconstruct the program flow. However, its trace output also enables a decompressor to reconstruct the program flow when it does not have a copy of parts of the program, for example because the program uses self-modifying code.

A *Program Flow Trace Macrocell* (PTM) implements the Program Flow Trace architecture.

Program Status Register (PSR)
Holds processor status and control information. The *Current Program Status Register* (CPSR) is the active PSR that affects processor operation. The *Saved Program Status Register* (SPSR) is a copy of the CPSR saved by the hardware.

Programming Language Interface (PLI)
For Verilog simulators, an interface by which foreign code can be included in a simulation. Foreign code is code written in a different language.

Project template A collection of configuration files for specific target development platforms. These templates enable you to create a target-specific development project in the ARM Workbench IDE.

See also ARM Workbench IDE (AWIDE) and RealView Compilation Tools (RVCT).

Protection region A memory region whose position, size, and other properties are defined by the Memory Protection Unit registers.

Protection Unit (PU) *See* Memory Protection Unit (MPU).

PSR *See* Program Status Register (PSR).

Quadword A 128-bit data item. Quadwords are normally at least word-aligned in ARM systems.

Quadword-aligned A data item having a memory address that is divisible by 16.

RAO *See* Read-As-One (RAO).

RAO/SBOP Read-As-One, Should-Be-One-or-Preserved on writes.

Hardware must implement the field as Read-as-One, and must ignore writes to the field.

Software can rely on the field reading as all 1s, but must use an SBOP policy to write to the field.

This description can apply to a single bit that reads as 1, or to a field that reads as all 1s.

See also Read-As-One (RAO), Should-Be-One-or-Preserved (SBOP).

RAO/WI Read-As-One, Writes Ignored.

Hardware must implement the field as Read-as-One, and must ignore writes to the field.

Software can rely on the field reading as all 1s, and on writes being ignored.

This description can apply to a single bit that reads as 1, or to a field that reads as all 1s.

See also Read-As-One (RAO).

RAZ *See* Read-As-Zero (RAZ).

RAZ/SBZP Read-As-Zero, Should-Be-Zero-or-Preserved on writes.

Hardware must implement the field as Read-as-Zero, and must ignore writes to the field.

Software can rely on the field reading as all 0s, but must use an SBZP policy to write to the field.

This description can apply to a single bit that reads as 0, or to a field that reads as all 0s.

See also Read-As-Zero (RAZ), Should-Be-Zero-or-Preserved (SBZP).

RAZ/WI Read-As-One, Writes Ignored.

Hardware must implement the field as Read-as-Zero, and must ignore writes to the field.

Software can rely on the field reading as all 0s, and on writes being ignored.

This description can apply to a single bit that reads as 0, or to a field that reads as all 0s.

See also Read-As-Zero (RAZ).

Read A memory operation that has the semantics of a load. See the *ARM Architecture Reference Manual* for more information.

Read-As-One (RAO) Hardware must implement the field as reading as all 1s.

Software can rely on the field reading as all 1s.

This description can apply to a single bit that reads as 1, or to a field that reads as all 1s.

Read-As-Zero (RAZ) Hardware must implement the field as reading as all 0s.

Software can rely on the field reading as all 0s.

This description can apply to a single bit that reads as 0, or to a field that reads as all 0s.

Read, modify, write In a read, modify, write sequence, a value is read to a general-purpose register, the relevant fields updated in that register, and the new value written back.

Read-Only Position Independent (ROPI)
In the context of the ARM architecture, code or read-only data that can be placed at any address.

Real Time System Model (RTSM)
A software model of a development system, for example, the Emulation Baseboard. The model can run applications at almost full speed. This enables applications and operating systems to be written and debugged without a requirement for actual hardware.

RealView Development Suite (RVDS)
The suite of software development tools, together with supporting documentation and examples, that enable you to write and debug applications for the ARM family of processors.

See also ARM Compiler for DS-5, ARM profiler, Eclipse for DS-5, Development Studio 5 (DS-5), RealView ICE, and RealView Trace and RealView Trace 2.

Read Write Position Independent (RWPI)
In the context of the ARM architecture, read-write code or data that can be placed at any address.

RealMonitor
A small program that, when integrated into your target application or *Real-Time Operating System* (RTOS), enables you to observe and debug your target while parts of your application continue to run.

RealView Instruction Set Simulator (RVISS)
One of the ARM simulators supplied with RVDS. RVISS is a collection of programs that simulate the instruction sets and architecture of various ARM processors. This provides instruction-accurate simulation and enables ARM and Thumb executable programs to be run on non-native hardware. RVISS provides modules that model:

- the ARM processor
- the memory used by the processor.

There are alternative predefined models for each of these parts. However, you can create your own models if a supplied model does not meet your requirements.

See also Instruction Set System Model (ISSM) and Real Time System Model (RTSM).

RealView Compilation Tools (RVCT)
A suite of tools that, together with supporting documentation and examples, enables you to write and build applications for the ARM family of processors.

See also armcc and armasm.

RealView Debugger
An ARM debugger that enables you to examine and control the execution of software running on a debug target. RealView Debugger is supplied as part of RVDS in both Windows and Red Hat Linux versions.

RealView Debugger Trace
Part of RVDS that extends the debugging capability with the addition of real-time program and data tracing. It is available from the RealView Debugger Code window.

See also RealView Debugger Trace and RealView ICE.

RealView Development Suite (RVDS)
The suite of software development applications, together with supporting documentation and examples, that enable you to write and debug applications for ARM processors. RVDS supersedes ARM Developer Suite.

RealView ICE
An ARM JTAG interface unit for debugging embedded processor cores that uses a DBGTAP or Serial Wire interface.

RealView Trace and RealView Trace 2
Work in conjunction with RealView ICE to provide real-time trace functionality for software running in System-on-Chip devices with deeply embedded ARM processors. RealView Trace 2 also supports data streaming directly to ARM Profiler, providing real-time hardware platform profiling.

See also RealView Debugger Trace and RealView ICE.

Region
In an image, a region is a contiguous sequence of one to three output sections (RO, RW, and ZI). A region typically maps onto a physical memory device, such as ROM, RAM, or peripherals.

See also Root region.

Remapping
Changing the address of physical memory or devices after an application has started executing. This might be done to permit RAM to replace ROM when the initialization has completed.

Replicator
In an ARM trace macrocell, enables two trace sinks to be wired together and to operate independently on the same incoming trace stream. The input trace stream is output onto two independent ATB ports.

Reserved
Unless otherwise stated in the architecture or product documentation, reserved:
- instruction and 32-bit system control register encodings are UNPREDICTABLE
- 64-bit system control register encodings are Undefined
- register bit fields are UNK/SBZP.

Root region
In an image, regions having the same load and execution address. A non-root region is a region that must be copied from its load address to its execution address.

See also Region.

ROPI
See Read-Only Position Independent (ROPI).

Round to Nearest (RN) mode
In floating-point operation, the rounded result is the nearest representable number to the un-rounded result.

See also Rounding mode and Rounding error.

Round towards Minus infinity (RM) mode
In floating-point operation, the rounded result is the nearest representable number that is less than or equal to the exact result.

See also Rounding mode and Rounding error.

Round towards Plus infinity (RP) mode
In floating-point operation, the rounded result is the nearest representable number that is greater than or equal to the exact result.

See also Rounding mode and Rounding error.

Round towards Zero (RZ) mode
In floating-point operation, results are rounded to the nearest representable number that is no greater in magnitude than the un-rounded result. This rounding mode discards any bits to the right of the significand, and therefore always rounds down. It is used by the C, C++, and Java languages in integer conversions.

See also Rounding mode and Rounding error.

Rounding error
Is defined to be the value of the rounded result of an arithmetic operation minus the exact result of the operation.

See also Rounding mode.

Rounding mode
In floating-point operation, specifies how the exact result of a floating-point operation is rounded to a value that is representable in the destination format.

See also Round to Nearest (RN) mode, Round towards Minus infinity (RM) mode, Round towards Plus infinity (RP) mode and Round towards Zero (RZ) mode.

RSD
See Running System Debug (RSD).

RTSM
See Real Time System Model (RTSM).

Running System Debug (RSD)
Means that a target can be debugged while it is running. RSD gives access to the application using a *Debug Agent* (DA) that resides on the target. The Debug Agent is scheduled with other tasks in the system.

See also Halted System Debug (HSD) and Debug Agent.

RVCT *See* RealView Compilation Tools (RVCT).

RVDS *See* RealView Development Suite (RVDS).

RVI *See* RealView ICE.

RWPI *See* Read Write Position Independent (RWPI).

Saved Program Status Register (SPSR)
A register used to save the CPSR value on taking an exception.

SBO *See* Should-Be-One (SBO).

SBOP *See* Should-Be-One-or-Preserved (SBOP).

SBZ *See* Should-Be-Zero (SBZ).

SBZP *See* Should-Be-Zero-or-Preserved (SBZP).

Scatter-loading Assigning the address and grouping of code and data sections individually rather than using single large blocks.

SDF *See* Standard Delay Format (SDF).

Section In the context of applications targeted at processors that implement the ARM architecture, a block of software or data for an image.

In the context of an MMU when using the short-descriptor table format, a 1MB region of virtual address space that a translation table descriptor assigns to a 1MB block of physical address or intermediate physical address space.

See also Input section and Output section.

Secure monitor In software descriptions, the module that switches the ARM processor between Normal World and Secure World execution environments. The Secure monitor is transparent to TrustZone Software developers.

Secure World *See* Normal and Secure Worlds.

Stack Pointer (SP) ARM core register R13.

Security hole A mechanism that bypasses system protection.

Semihosting A mechanism to communicate *Input/Output* (I/O) requests from application code to a host workstation running a debugger. For example, you can use semihosting to enable functions in the C library, such as `printf()` and `scanf()`, to use the screen and keyboard on the host workstation instead of having a screen and keyboard on the target system.

Serial Wire Debug (SWD)
A debug implementation that uses a serial connection between the SoC and a debugger. This connection normally requires a bidirectional data signal and a separate clock signal, rather than the four to six signals required for a JTAG connection.

Serial Wire Debug Port (SWDP)
The interface for Serial Wire Debug.

Shared layer In general, contains functions used by more than one Mali GPU driver. It contains math functions, texture processing and list utilities.

Should-Be-One (SBO) Hardware must ignore writes to the field.

Software should write the field as all 1s. If software writes a value that is not all 1s, it must expect an UNPREDICTABLE result.

This description can apply to a single bit that should be written as 1, or to a field that should be written as all 1s.

Should-Be-One-or-Preserved (SBOP)
The Large Physical Address Extension modifies the definition of SBOP for register bits that are reallocated by the extension, and as a result are SBOP in some but not all contexts. For more information see the *ARM Architecture Reference Manual, ARMv7-A* and *ARMv7-R* edition. The generic definition of SBOP given here applies only to bits that are not affected by this modification.

Hardware must ignore writes to the field.

If software has read the field since the processor implementing the field was last reset and initialized, it should preserve the value of the field by writing the value that it previously read from the field. Otherwise, it should write the field as all 1s.

If software writes a value to the field that is not a value previously read for the field and is not all 1s, it must expect an UNPREDICTABLE result.

This description can apply to a single bit that should be written as its preserved value or as 1, or to a field that should be written as its preserved value or as all 1s.

See also Should-Be-Zero-or-Preserved (SBZP) and Should-Be-One (SBO).

Should-Be-Zero (SBZ) Hardware must ignore writes to the field.

Software should write the field as all 0s. If software writes a value that is not all 0s, it must expect an UNPREDICTABLE result.

This description can apply to a single bit that should be written as 0, or to a field that should be written as all 0s.

Should-Be-Zero-or-Preserved (SBZP)
The Large Physical Address Extension modifies the definition of SBZP for register bits that are reallocated by the extension, and as a result are SBZP in some but not all contexts. For more information see the *ARM Architecture Reference Manual*, *ARMv7-A* and *ARMv7-R* edition. The generic definition of SBZP given here applies only to bits that are not affected by this modification.

Hardware must ignore writes to the field.

If software has read the field since the processor implementing the field was last reset and initialized, it must preserve the value of the field by writing the value that it previously read from the field. Otherwise, it must write the field as all 0s.

If software writes a value to the field that is not a value previously read for the field and is not all 0s, it must expect an UNPREDICTABLE result.

This description can apply to a single bit that should be written as its preserved value or as 0, or to a field that should be written as its preserved value or as all 0s.

See also Should-Be-One-or-Preserved (SBOP) and Should-Be-Zero (SBZ).

Signaling NaN
In floating-point operation, the floating-point coprocessor causes an Invalid Operation exception whenever any floating-point operation receives a signaling NaN as an operand. You can use signaling NaNs in debugging, to track down some uses of uninitialized variables.

Sign-Off Model (SOM)
An opaque, compiled simulation model generated from a technology-specific netlist of an ARM processor, derived after gate level synthesis and timing annotation, that you can use in back-annotated gate-level simulations to prove the function and timing behavior of the device. A SOM provides accurate timing simulation of a SoC, and supports simulation using production test vectors from the *Automatic Test Pattern Generation* (ATPG) tool. It only supports back-annotation using SDF files. The SOM includes timing information but provides slower simulation than a DSM.

SIMD
See Single Instruction, Multiple Data (SIMD).

Simple tracepoint
A type of tracepoint that enables you to set trigger points, trace start and end points, or trace ranges for memory and data accesses.

See also Tracepoint.

Simulator
In the context of the ARM tools, a simulator executes non-native instructions in software, simulating a core.

See also Real Time System Model (RTSM) and Instruction Set System Model (ISSM).

Single Instruction, Multiple Data (SIMD)
From ARM v6, the ARM instruction sets include SIMD instructions. These comprise:

- Instructions that perform parallel operations on the bytes or halfwords of the ARM core registers.
- Instructions that perform vector operations. That is, they perform parallel operations on vectors held in multiword registers.

Different versions of the ARM architecture support and recommend different instructions for vector operations. See the appropriate version of the ARM Architecture Reference Manual for more information.

Software breakpoint
A breakpoint that is implemented by replacing an instruction in memory with one that causes the processor to take an exception. Because instruction memory must be altered, software breakpoints cannot be used where instructions are stored in read-only memory.

See also Instruction breakpoint and Data breakpoint.

SOM
See Sign-Off Model (SOM).

SPICE
Simulation Program with Integrated Circuit Emphasis. An accurate transistor-level electronic circuit simulation tool that can predict how an equivalent real circuit behaves for given circuit conditions.

SPSR
See Saved Program Status Register (SPSR).

Standard Delay Format (SDF)
A file format that contains timing information to the level of individual bits of buses and is used in SDF back-annotation. An SDF file can be generated in a number of ways, but most commonly from a delay calculator.

Subnormal value
In the IEEE 754 standard format for single-precision and double-precision operands, a subnormal value has a zero exponent and a nonzero fraction field. The IEEE 754 standard requires that the generation and manipulation of subnormal operands be performed with the same precision as normal operands.

Plus or minus 0 have zero exponent fields, but are not subnormals because there is no loss of accuracy.

Supervisor Call (SVC) An instruction that causes the processor to take a Supervisor Call exception.

Used by the ARM standard C library to handle semihosting. This was previously called SoftWare Interrupt (SWI).

Support code In a floating-point implementation, system software that complements the hardware VFP implementation. The support code can provide a library of routines that perform operations beyond the scope of the hardware. The support code includes a set of exception handlers to process exceptional conditions in compliance with the IEEE 754 standard.

SVC *See* Supervisor Call (SVC).

SWD *See* Serial Wire Debug (SWD).

SWDP *See* Serial Wire Debug Port (SWDP).

SWI *See* Supervisor Call (SVC).

Synchronization primitive

An instruction that is used to ensure memory synchronization, for example `LDREX` or `STREX`. See the ARM *Architecture Reference Manual* for more information.

System Control Space On Cortex-M series processors, a memory-mapped region from `0xE000E000` to `0xE000EFFF` that provides system control and configuration registers, including control of the *Nested Vectored Interrupt Controller* (NVIC) and debug functions.

TAP Controller Logic on a device that enables access to some or all of that device for test purposes. The circuit functionality is defined in IEEE1149.1.

See also Joint Test Action Group (JTAG).

Target In the context of an ARM debugger, the part of your development platform to which the debugger can connect, and on which debugging operations can be performed. A target can be:

- A runnable target, such as a processor that implements the ARM architecture. When connected to a runnable target, you can perform execution-related debugging operations on that target, such as stepping and tracing.
- A non-runnable CoreSight component. CoreSight components provide a system wide solution to real-time debug and trace.

See also Debug interface.

Target Vehicle Target vehicles provide RVDS with a standard interface to disparate targets so that the debugger can connect easily to new target types without having to make changes to the debugger core software. The interface can be a hardware or software interface.

See also RealView ICE and Real Time System Model (RTSM).

TCD *See* Trace Capture Device (TCD).

TCM *See* Tightly Coupled Memory (TCM).

TCK The electronic clock signal that times data on the TAP data lines **TMS**, **TDI**, and **TDO**.

See also Test Data Input (TDI) and Test Data Output (TDO).

Test Access Port (TAP)

The collection of four mandatory and one optional terminals that form the input/output and control interface to a JTAG boundary-scan architecture. The mandatory terminals are **TDI**, **TDO**, **TMS**, and **TCK**. The optional terminal is **nTRST**. This signal is mandatory in ARM processors because it is used to reset the debug logic.

See also Joint Test Action Group (JTAG), TAP Controller, TCK, Test Data Input (TDI), Test Data Output (TDO), and TMS.

Test Data Input (TDI)

Test Data Input (TDI) is the electronic signal input to a TAP controller from the data source (upstream). Usually this is seen connecting the RealView ICE run control unit to the first TAP controller.

See also Joint Test Action Group (JTAG), RealView ICE, *and* TAP Controller.

Test Data Output (TDO)

Test Data Output (TDO) is the electronic signal output from a TAP controller to the downstream data sink. Usually this connects the last TAP controller to the RealView ICE run control unit.

See also Joint Test Action Group (JTAG), RealView ICE, *and* TAP Controller.

Texture Descriptor

Data structure used by the Mali GPU to describe one texture map.

Thumb instruction

One or two halfwords that specify an operation for a processor in Thumb state to perform. Thumb instructions must be halfword-aligned.

See also Thumb state and ThumbEE state.

Thumb-2

The technology, introduced in ARMv6T2, that extends the Thumb instruction set to a variable-length instructions set that includes both 16-bit and 32-bit instructions.

See also Thumb instruction *and* ThumbEE instruction.

Thumb state

In Thumb state the processor executes the Thumb instruction set.

See also ARM state, Jazelle state, and ThumbEE state.

ThumbEE instruction

One or two halfwords that specify an operation for a processor in ThumbEE state to perform. Thumb-EE is the Thumb Execution Environment and the ThumbEE instruction set is based on the Thumb instruction set, with some changes and additions to make it a better target for dynamically generated code, that is, code compiled on the device either shortly before or during execution.

See also ARM instruction and Thumb instruction.

ThumbEE state

In ThumbEE state the processor executes the ThumbEE instruction set.

See also ARM state, Thumb state, and Jazelle state.

Tightly Coupled Memory (TCM)

An area of low latency memory that provides predictable instruction execution or data load timing, for cases where deterministic performance is required. TCMs are suited to holding:

- critical routines such as for interrupt handling
- scratchpad data
- data types whose locality is not suited to caching
- critical data structures, such as interrupt stacks.

Tile buffer

A memory buffer inside the GPU that holds the framebuffer contents for the tile that is currently being rendered. The tile buffer can be accessed without using the memory bus.

TLB

See Translation Lookaside Buffer (TLB).

TLB lockdown Prevents specific translation table walk results being removed from the TLB. This ensures that accesses to the associated memory areas never cause a translation table walk.

TMS Test Mode Select.

TPA *See* Trace Port Analyzer (TPA).

TPIU *See* Trace Port Interface Unit (TPIU).

Trace Capture Device (TCD)
A generic term to describe Trace Port Analyzers, logic analyzers, and on-chip trace buffers.

Trace driver A remote debug interface target that controls a piece of trace hardware. That is, the trigger macrocell, trace macrocell, and trace capture tool.

Trace funnel In an ARM trace macrocell, a device that combines multiple trace sources onto a single bus.

See also AHB Trace Macrocell (HTM) and CoreSight.

Trace hardware A term for a device that contains an ARM trace macrocell.

Trace port A port on a device, such as a processor or ASIC, used to output trace information.

Trace Port Analyzer (TPA)
A hardware device that captures trace information output on a trace port. This can be a low-cost product designed specifically for trace acquisition, or a logic analyzer.

Trace Port Interface Unit (TPIU)
Drains trace data and acts as a bridge between the on-chip trace data and the data stream captured by a TPA.

Tracepoint A tracepoint can be set on a line of source code, a line of assembly code, or a memory address. In an ARM debugger, you can set a variety of tracepoints to determine exactly what program information is traced.

See also Chained tracepoint and Tracepoint unit.

Tracepoint unit In the context of an ARM debugger, a unit within a Chained tracepoint that combines with other tracepoint units to create a complex tracepoint.

See also Chained tracepoint and Tracepoint.

Translation Lookaside Buffer (TLB)
A memory structure containing the results of translation table walks. TLBs help to reduce the average cost of memory accesses.

Translation table A table, held in memory, that contains descriptors that define the properties of regions of memory.

Translation table walk A full translation table lookup. It is performed automatically by hardware.

Trap enable bits In a floating-point coprocessor, determines whether trapped or untrapped exception handling is selected. If trapped exception handling is selected, the way it is carried out is IMPLEMENTATION DEFINED.

Triangle setup unit A component of a fragment processor. The triangle setup unit prepares primitives for rendering by calculating the data required to rasterize and shade the primitive.

Trigger In the context of tracing, a trigger is an event that instructs the debugger to stop collecting trace and display the trace information around the trigger position, without halting the processor. The exact information that is displayed depends on the position of the trigger within the buffer.

See also Bounce.

Trigger instruction A floating-point instruction that causes a bounce when it is issued.

See also Bounce.

TrustZone Software A secure software framework that uses the ARM architecture Security Extensions.

T32 An instruction set that can be used by an ARMv8 processor that is in AArch32 execution state. T32 is a variable-length instruction set that uses both 16-bit and 32-bit instruction encodings. It is compatible with the ARMv7 Thumb instruction set.

See also AArch32, A32.

UMP *See* Unified Memory Provider (UMP).

Unaligned An unaligned access is an access where the address of the access is not aligned to the size of the elements of the access.

Unconditional breakpoint
A breakpoint that does not have a conditional qualifier assigned. The breakpoint activates immediately it is hit, but subsequent image execution is determined by any actions assigned to the breakpoint.

See also Conditional breakpoint and Hardware breakpoint.

Undefined Indicates an instruction that is not architecturally defined. It generates an Undefined Instruction exception. See the *ARM Architecture Reference Manual* for more information.

Unified Assembler Language (UAL)
A common assembler language for the ARM and Thumb instruction sets in ARMv7, or the A32 and T32 instruction sets in ARMv8.

See the appropriate ARM Architecture Reference Manual for more information.

Unified Memory Provider (UMP)
Provides a safe way to share memory across processes, drivers and hardware components, possibly using an MMU or MPU for memory protection. The Mali driver stack uses the UMP API for certain optional functionality.

UNK An abbreviation indicating that software must treat a field as containing an UNKNOWN value.

In any implementation, the bit must read as 0, or all 0s for a bit field. Software must not rely on the field reading as zero.

See also UNKNOWN.

UNK/SBOP Hardware must implement the field as Read-As-One, and must ignore writes to the field.

Software must not rely on the field reading as all 1s, and except for writing back to the register it must treat the value as if it is UNKNOWN. Software must use an SBOP policy to write to the field.

This description can apply to a single bit that should be written as its preserved value or as 1, or to a field that should be written as its preserved value or as all 1s.

See also Read-As-One (RAO), Should-Be-One-or-Preserved (SBOP), UNKNOWN.

UNK/SBZP Hardware must implement the field as Read-As-Zero, and must ignore writes to the field.

Software must not rely on the field reading as all 0s, and except for writing back to the register it must treat the value as if it is UNKNOWN. Software must use an SBZP policy to write to the field.

This description can apply to a single bit that should be written as its preserved value or as 0, or to a field that should be written as its preserved value or as all 0s.

See also Read-As-Zero (RAZ), Should-Be-Zero-or-Preserved (SBZP), UNKNOWN.

UNKNOWN An UNKNOWN value does not contain valid data, and can vary from moment to moment, instruction to instruction, and implementation to implementation. An UNKNOWN value must not be a security hole or documented or promoted as having a defined value or effect.

When UNKNOWN appears in body text, it is always in SMALL CAPITALS.

UNP *See* UNPREDICTABLE.

UNPREDICTABLE For an ARM trace macrocell, means that the behavior of the macrocell cannot be relied on. Such conditions have not been validated. When applied to the programming of an event resource, only the output of that event resource is UNPREDICTABLE. UNPREDICTABLE behavior can affect the behavior of the entire system, because the trace macrocell can cause the processor to enter debug state, and external outputs can be used for other purposes.

For a processor means the behavior cannot be relied on. UNPREDICTABLE behavior must not represent a security hole. UNPREDICTABLE behavior must not hang the processor, or any parts of the system.

When UNPREDICTABLE appears in body text, it is always in SMALL CAPITALS.

VA *See* Virtual Address (VA).

VDMA Video Direct Memory Access. The VDMA transfers data in a burst efficient way to and from system memory.

Vectorization Using multiword registers to hold multiple values of the same type for SIMD processing. For example, software might use doubleword registers to hold four 16-bit unsigned integers. Vectorization also describes the process of adapting software to use SIMD processing.

Vector operations are provided by:

- The VFP instructions in ARMv6.
- The Advanced SIMD extension in ARMv7.

Veneer A small block of code used with subroutine calls when there is a requirement to change processor state or branch to an address that cannot be reached in the current processor state.

Vertex A set of data defining the properties of one point of a primitive. For example, a point primitive, an endpoint of a line primitive, or a corner of a triangle primitive.

Vertex attributes The data provided by the application, to define a vertex.

Vertex loader A component of the vertex processor that loads vertex attributes from memory and inputs them to the vertex shader unit.

Vertex processor A programmable processor that executes vertex shaders with typical transform and lightning calculations, and generates lists of primitives for a fragment processor to draw.

The vertex processor was originally known as a geometry processor.

Vertex shader A program running on a vertex processor or shader core, that calculates the position and other characteristics, such as color and texture coordinates, for each vertex.

Vertex shader unit A programmable component of the vertex processor that runs vertex shaders.

VFP A coprocessor extension to the ARM architecture that provides floating-point arithmetic. For ARMv7, more accurately described as the Floating-Point Extension.

Victim
A cache line, selected to be discarded to make room for a replacement cache line that is required because of a cache miss. How the victim is selected for eviction is processor-specific. A victim is also known as a cast out.

Virtual Address (VA)
An address used in an instruction as a data or instruction address. The PC, LR, and SP always hold virtual addresses. For a *Protected Memory System Architecture* (PMSA) implementation, the virtual address is identical to the physical address.

Warm reset
Also known as a core reset. Initializes most of the processor functionality, excluding the debug controller and debug logic. This type of reset is useful if you are using the debugging features of a processor.

See also Cold reset.

Watch
In an ARM debugger, a watch is a variable or expression that you require the debugger to display at every step or breakpoint so that you can see how its value changes.

Watchpoint
A debug event triggered by an access to memory, specified in terms of the address of the location in memory being accessed.

In DS-5, this is a hardware breakpoint.

See also Breakpoint.

Word
A 32-bit data item. Words are normally word-aligned in ARM systems.

Word-aligned
A data item having a memory address that is divisible by four.

Word-invariant
In a word-invariant system, the address of each byte of memory changes when switching between little-endian and big-endian operation, in such a way that the byte with address A in one endianness has address A EOR 3 in the other endianness. As a result, each aligned word of memory always consists of the same four bytes of memory in the same order, regardless of endianness. The change of endianness occurs because of the change to the byte addresses, not because the bytes are rearranged.

The ARM architecture supports word-invariant systems in ARMv3 and later versions. When word-invariant support is selected, the behavior of load or store instructions with unaligned addresses is instruction-specific, and is in general not the expected behavior for an unaligned access. ARM strongly recommends that word-invariant systems use the endianness that produces the required byte addresses at all times, apart possibly from very early in their reset handlers before they have set up the endianness, and that this early part of the reset handler uses only aligned word memory accesses.

See also Byte-invariant.

Write
Operations that have the semantics of a store. See the *ARM Architecture Reference Manual* for more information.

Write completion
The memory system indicates to the processor that a write is complete at a point in the transaction where the memory system can guarantee that the write is observable by all processors in the system.

In addition, for Strongly-ordered memory, a write to a memory-mapped peripheral is complete when it reaches that memory-mapped peripheral and therefore can trigger any side effects caused by the memory-mapped peripheral. Write completion is not required to ensure that all side effects are globally visible, although some peripherals might define this as a required property of completed writes.

Write interleave capability
The number of data-active write transactions for which the interface can transmit data. This is counted from the earliest transaction.

Write interleave depth The number of data-active write transactions for which the interface can receive data.

XTSM *See* XVC Test Scenario Manager.

XVC *See* eXtensible Verification Component (XVC).

XVC Test Scenario Manager
This coordinates the operation of multiple XVCs.

See also eXtensible Verification Component (XVC).

Appendix A
Revisions

This appendix describes the technical changes from the previous issue, Issue D, of this book.

Table A-1 Summary of changes

Term	Change
AArch32	New
AArch64	New
A32	New
A64	New
CADI	New
Explicit access	Updated
G-POP	New
POP	New
Quadword	New
Quadword-aligned	New
RAO/SBOP	Updated
RAO/WI	Updated
RAZ/SBZP	Updated
RAZ/WI	Updated

Table A-1 Summary of changes (continued)

Term	Change
Read-As-One (RAO)	Updated
Read-As-Zero (RAZ)	Updated
Should-Be-One (SBO)	Updated
Should-Be-One-or-Preserved (SBOP)	Updated
Should-Be-Zero (SBZ)	Updated
Should-Be-Zero-or-Preserved (SBZP)	Updated
T32	New
UNK/SBOP	Updated
UNK/SBZP	Updated