System Debugging

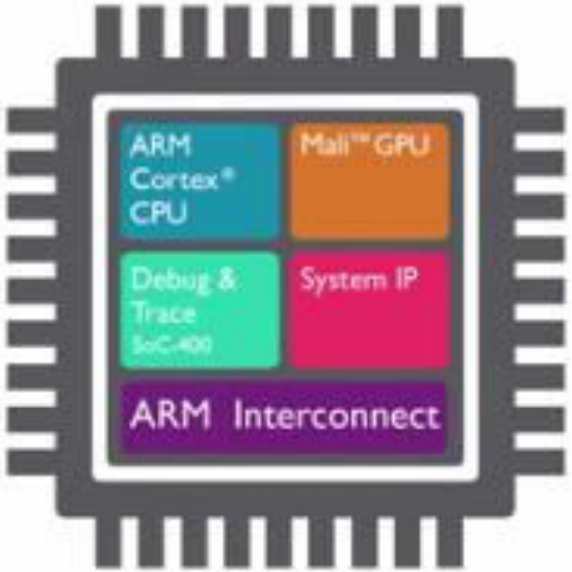
Learning Outcomes

At the end of this module, you will be able to:

- Describe the purpose of a debugging system, including the Arm CoreSight debugging system.
- Outline the components and their function in an Arm CoreSight debugging and trace system.
- Identify the main steps to build an Arm debugging system.
- Arm Cortex-A9 debug interface and Debug registers and CMSIS-DAP.

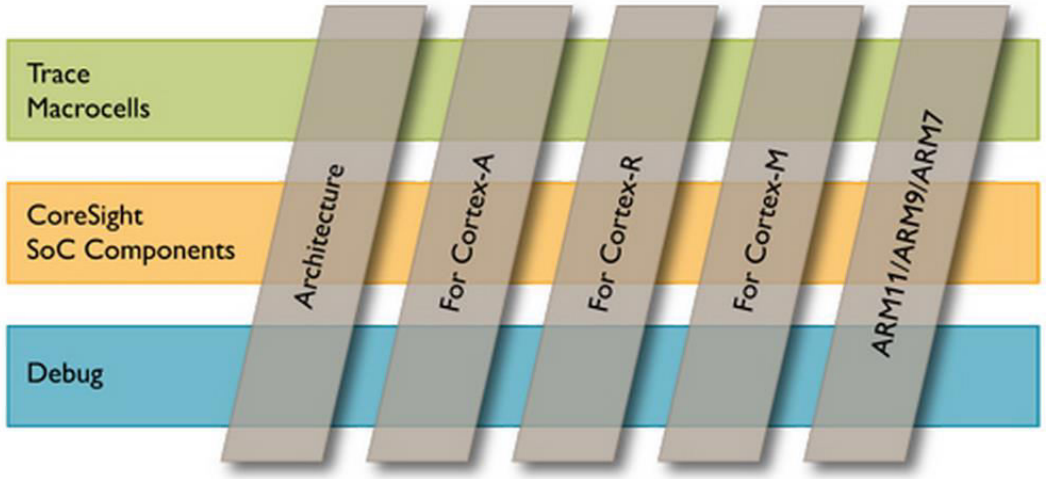
Arm CoreSight Debugging System

- **Arm CoreSight**
  - Arm debug and trace technology.
  - Most complete on-chip debug and real-time trace solution for the entire System on Chip (SoC).
  - Making Arm processor-based SoCs the easiest to debug and optimize.



Arm CoreSight Debugging System

- Arm CoreSight architecture
  - Arm processors real-time trace macrocells (ETM & PTM) architecture.
  - Arm CoreSight SoC component architecture .
  - Arm Debug Interface (ADI) architecture.



Arm CoreSight Debugging System

- **Arm Debug Interface** (**ADI**) Architecture
  - ADI defines a standard debug interface for debug components in an embedded System on Chip (SoC)
  - Includes Debug Access Ports (DAP), which contain Access Ports (AP) and Debug Ports (DP).

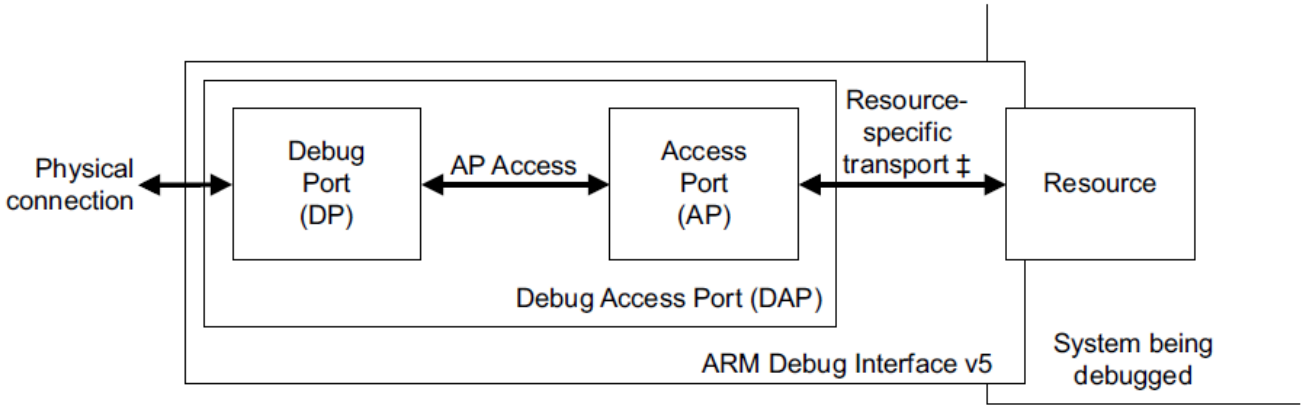
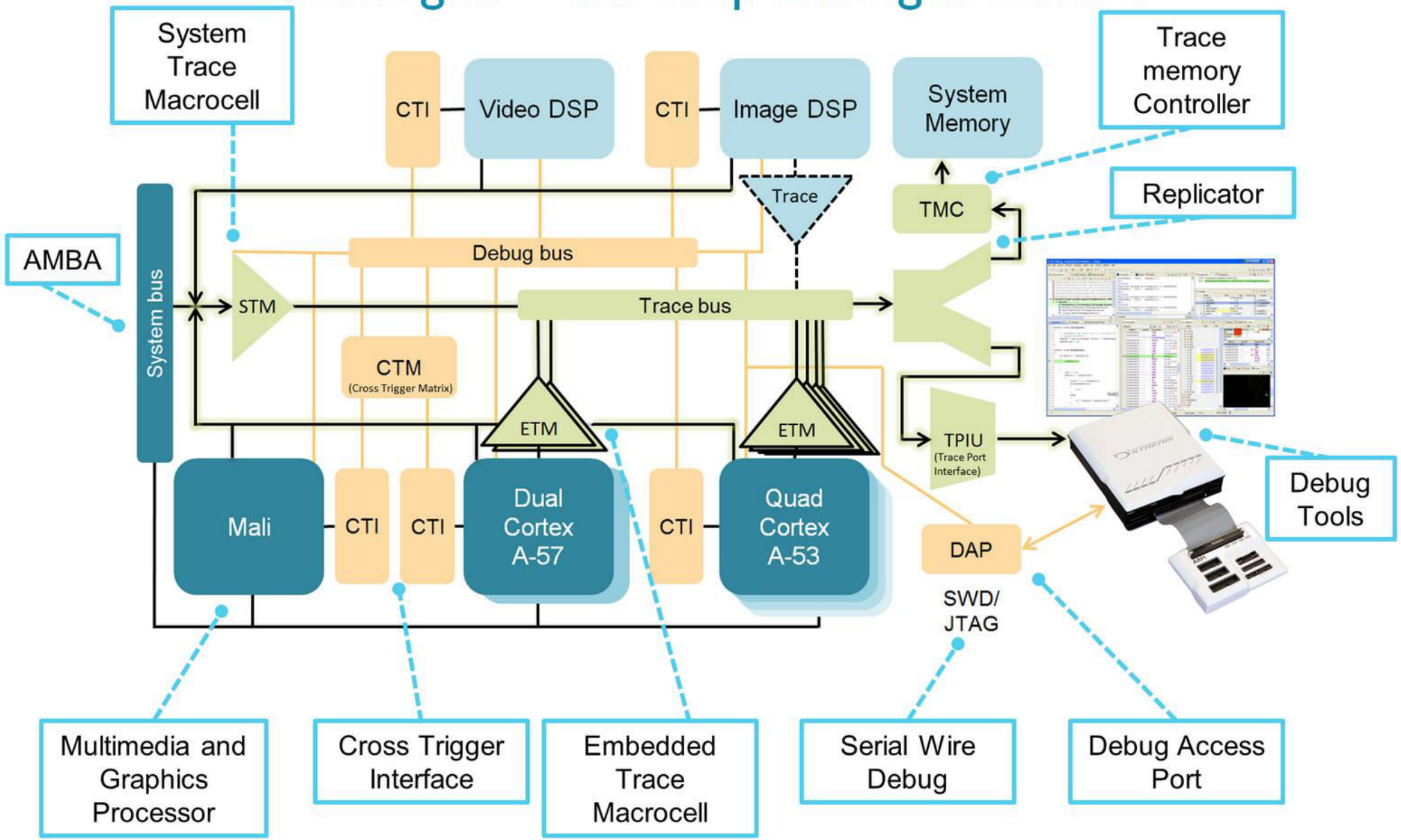


Figure source: Arm Debug Interface Architecture Specifications ADIv5 to ADIv5.2

- **Trace Macrocell** Architecture
  - **Embedded Trace Macrocell** (ETM) is a real-time trace module providing instruction and data tracing of a processor. ETM is an integral part of an Arm CoreSight debug and real-time trace solution.
    - **Program Trace Macrocell** (PTM) for Program Flow Trace of Cortex-A9
    - **Instrumentation Trace Macrocell** (ITM) for high level software view.
    - **AHB Trace Macrocell** (HTM) for performance and functional debug.

CoreSight™ On Chip Debug & Trace IP



arm

Example1: Single Processor Debug

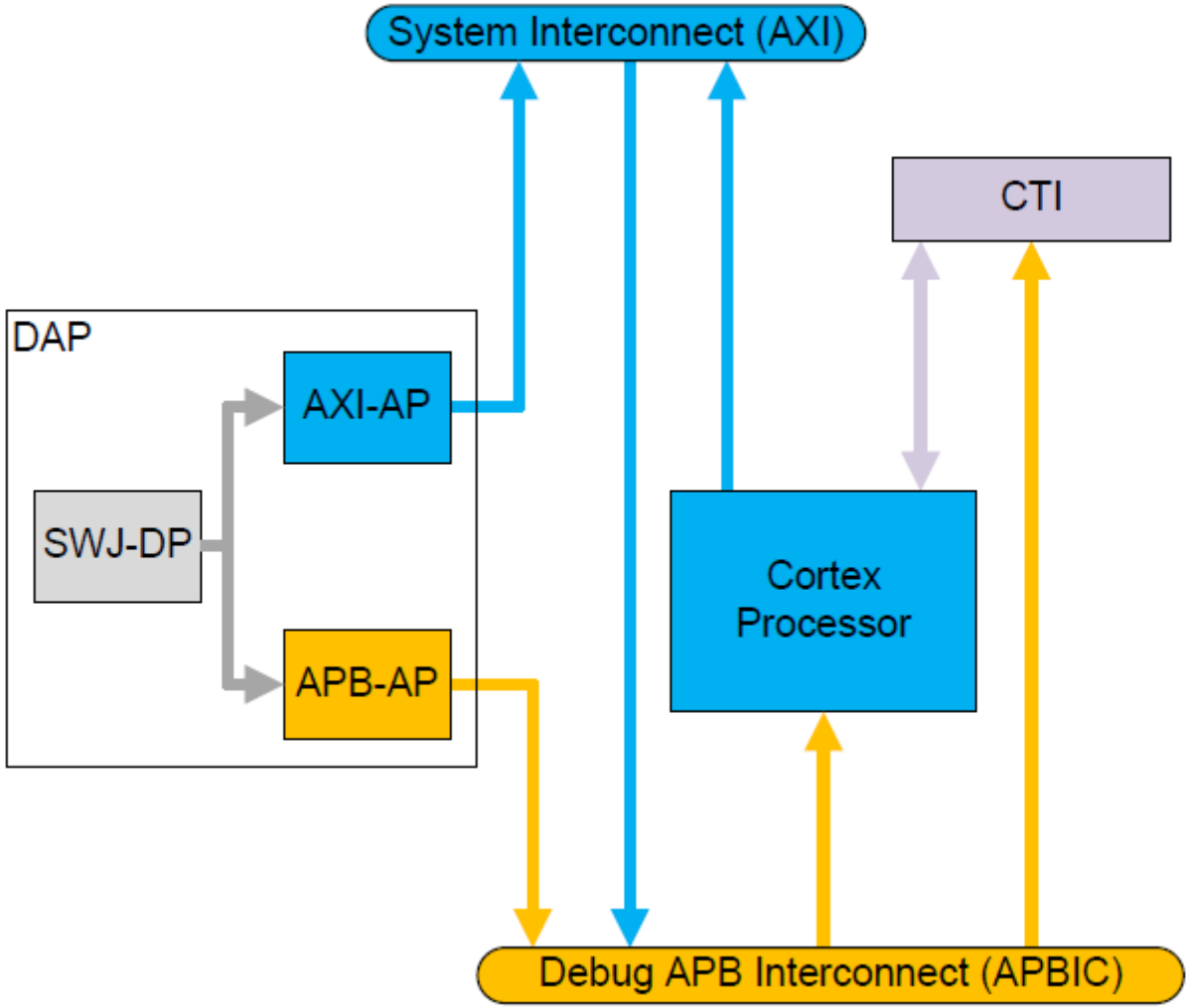


Figure source: CoreSight Technical Introduction. A quick start for designers. White Paper, Arm

Example 2: Single Processor Trace

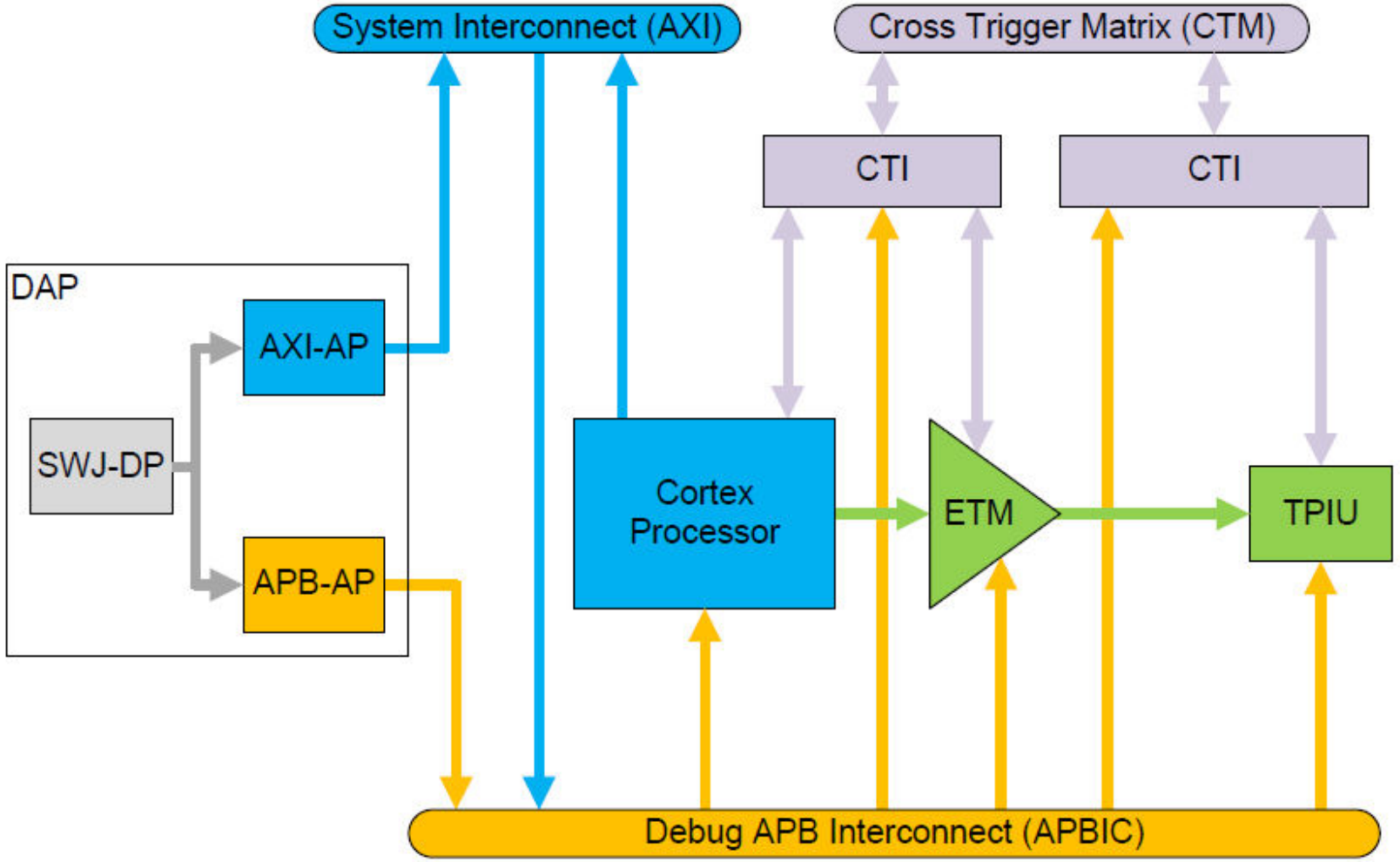


Figure source: CoreSight Technical Introduction. A quick start for designers. White Paper, Arm

Example 3: Multi Source Trace

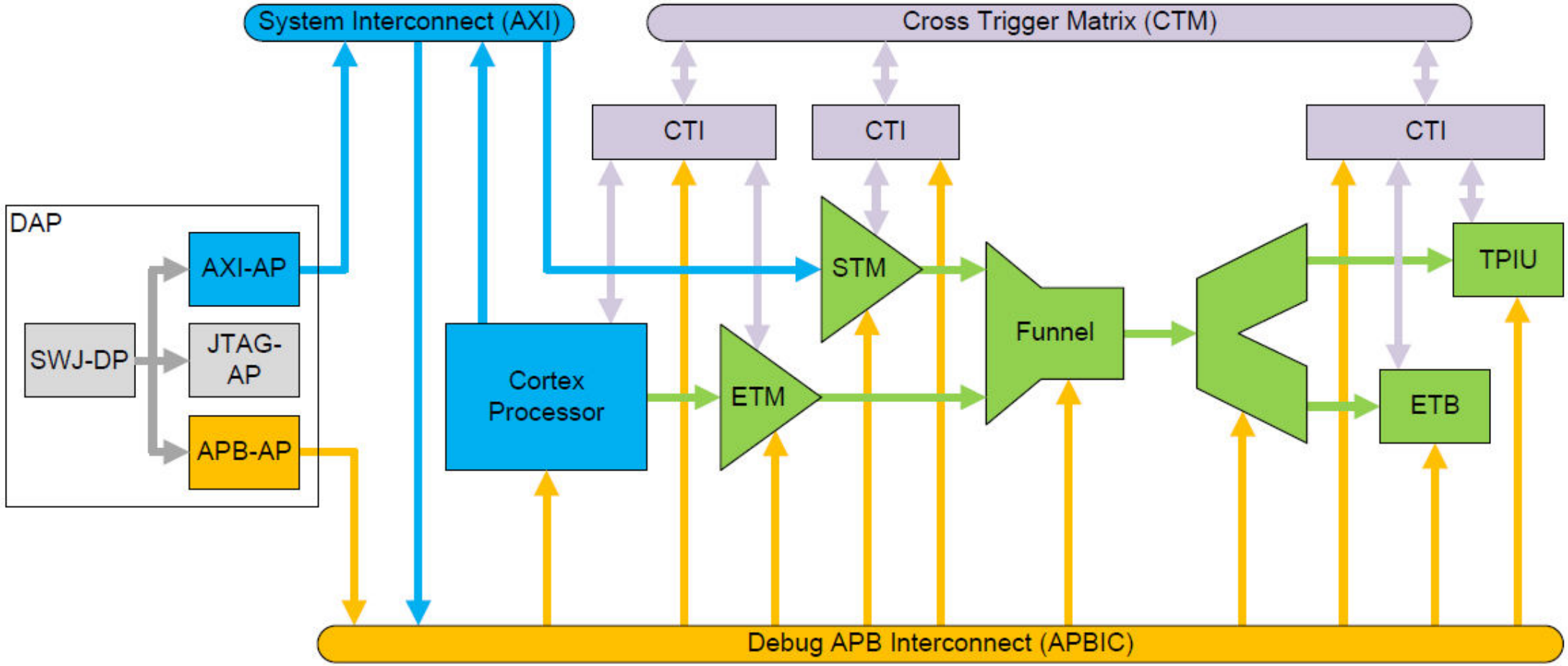
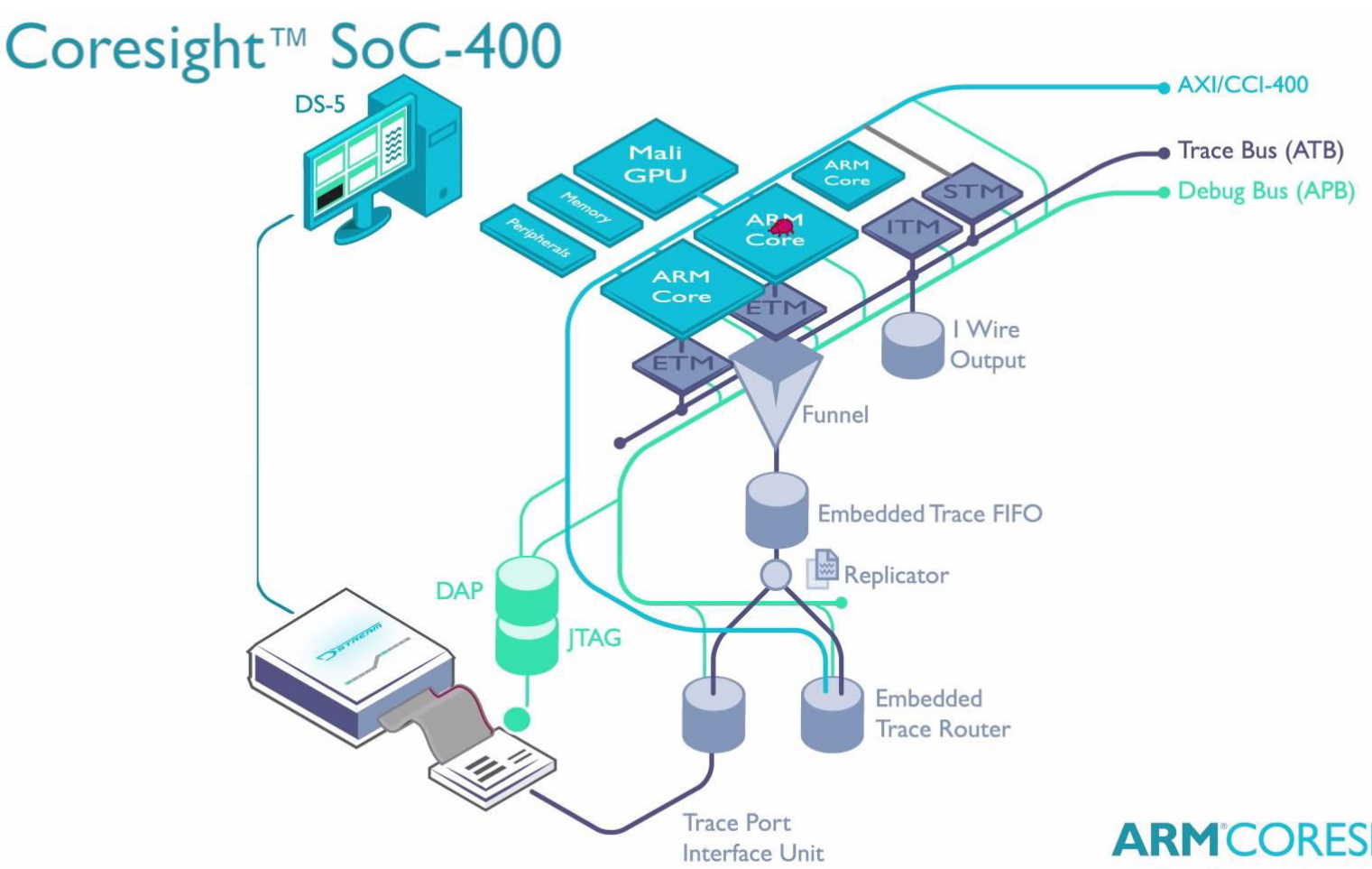


Figure source: CoreSight Technical Introduction. A quick start for designers. White Paper, Arm

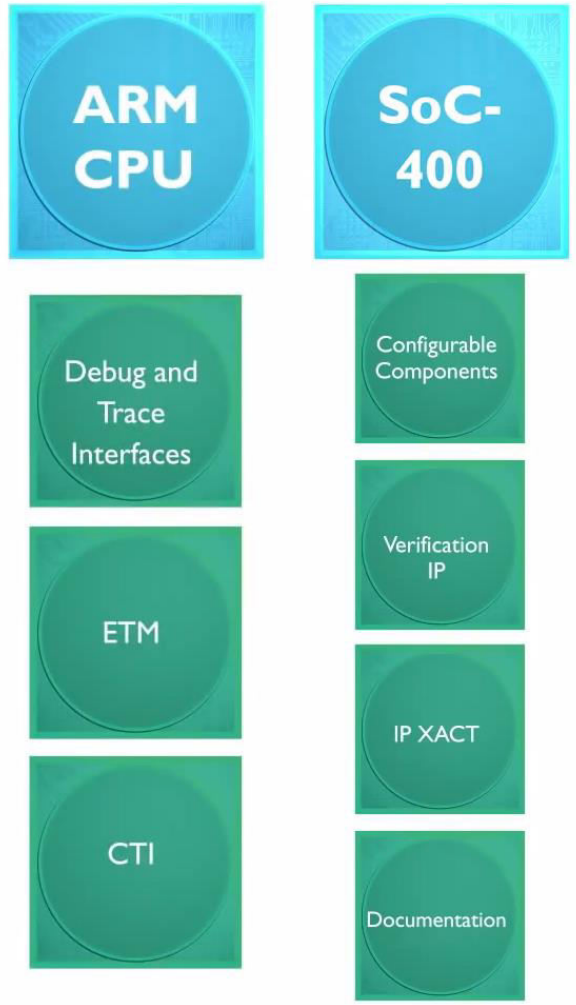
Arm CoreSight Debugging System

- CoreSight SoC Components Architecture
  - CoreSight™ SoC components in conjunction with the CoreSight Trace macrocells provide all the infrastructure required to debug, monitor, and optimize the performance of a complete System on Chip (SoC) design.
  - **CoreSight SoC-400** provides fully configurable versions of all of the CoreSight components together with AMBA Designer support for the entire range of CoreSight debug & trace logic

CoreSight SoC-400



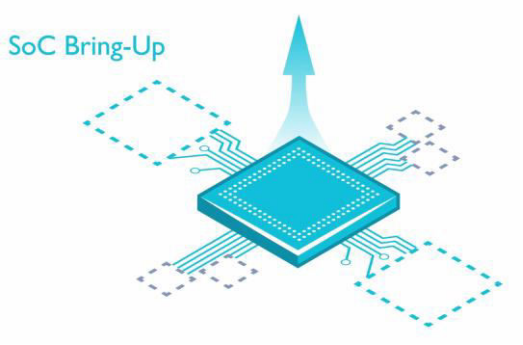
ARM CORESIGHT™
Processor Debug & Trace IP

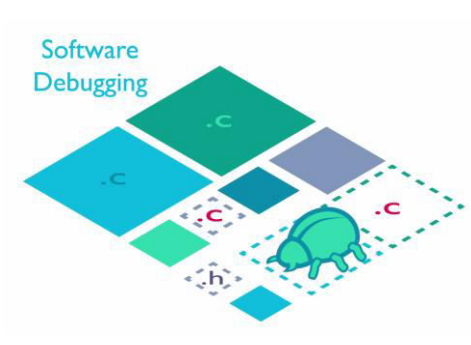


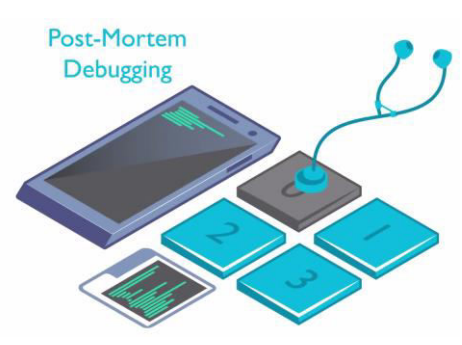
arm

CoreSight SoC-400

- SoC Bring-up
  - Fast bring-up SoC design, reduce time to market.
- Software Debugging
  - Help improve the reliability of your product.
- Post-mortem Debugging
  - Find bugs earlier in the product lifecycle.
- System Optimization
  - For both hardware and software help create best Arm solutions.



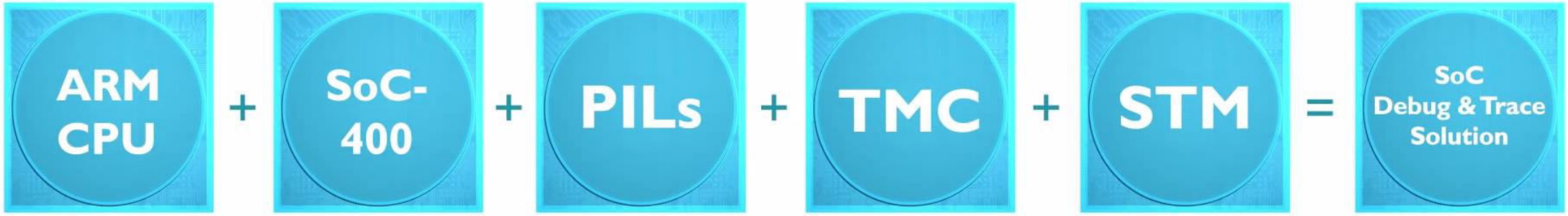




arm

Steps to build an Arm debugging subsystem

1. Start with an Arm CPU.
2. Use SoC-400 product and configurable components to build the infrastructure for debug trace sub-system.
3. Use Processor Integration Layer (PILs) to ease integration of Arm processors in SoC-400 debugging subsystem.
4. Use Trace Memory Controller (TMC) and System Trace Macrocell (STM) to build debug trace solution.



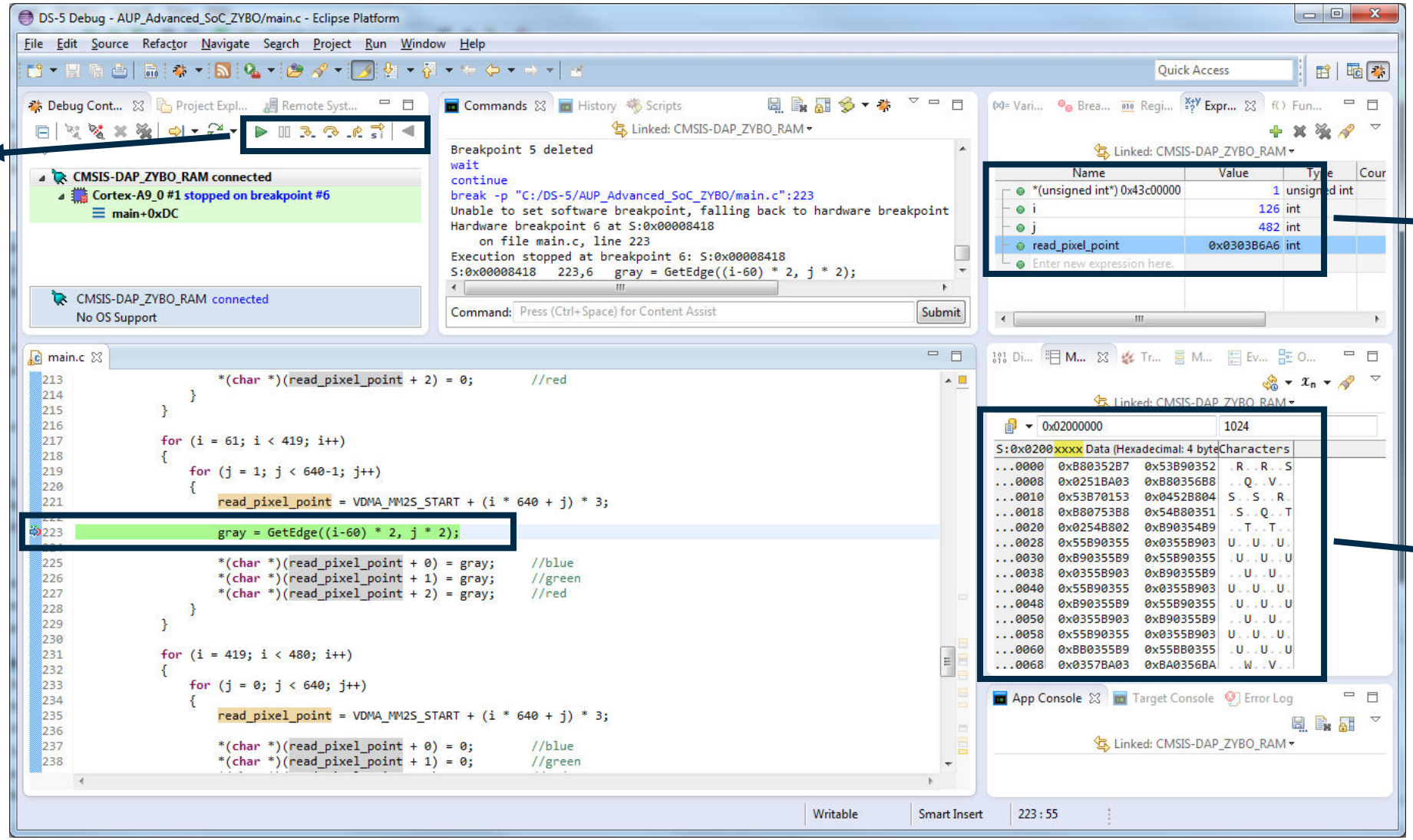
Supported tools

- Supported by over 25 industry-leading software and hardware debug tools companies, across all markets and regions
  - Debug of symmetric multi-processing and asymmetric multicore systems.
  - Powerful interactive debugging with real-time visibility.
  - Performance optimization.
  - Line and path code coverage of assembler and C/C++.
  - High-level system views with OS and RTOS context.
  - Real-time data monitoring, common to MCU and automotive applications.

Debug using Arm Development Studio

Debug control

Breakpoints

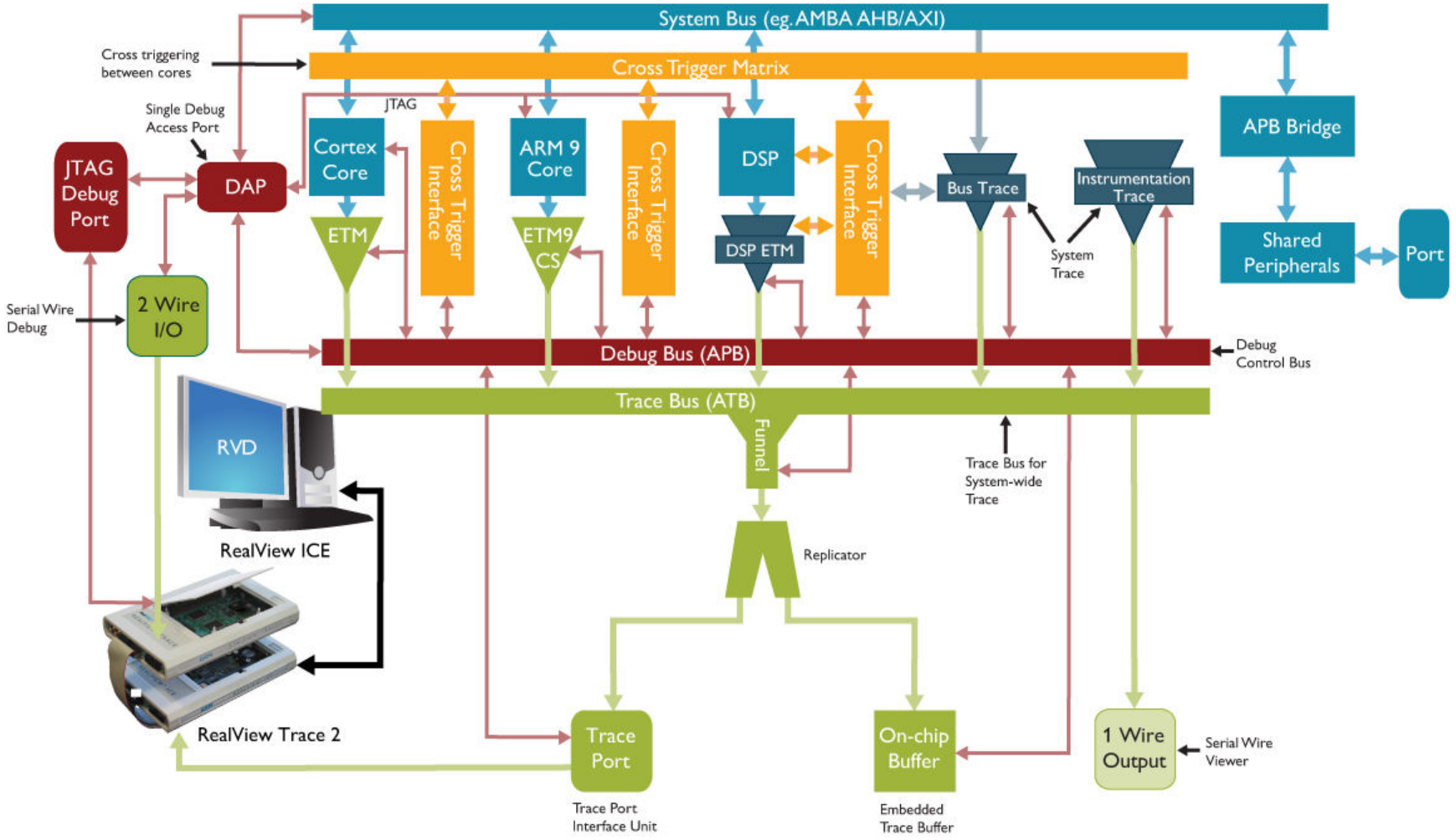


Variables watching

Framebuffer in “Memory”

Debugging Cotex-A9 processor using CoreSight

- CoreSight™ for Arm Cortex-A series processors
  - Provides embedded software and application developers with all the on-chip debug.
  - Real-time trace resources .
  - Optimization and debug of Cortex-A application processor platforms.



Debugging Cotex-A9 processor using CoreSight

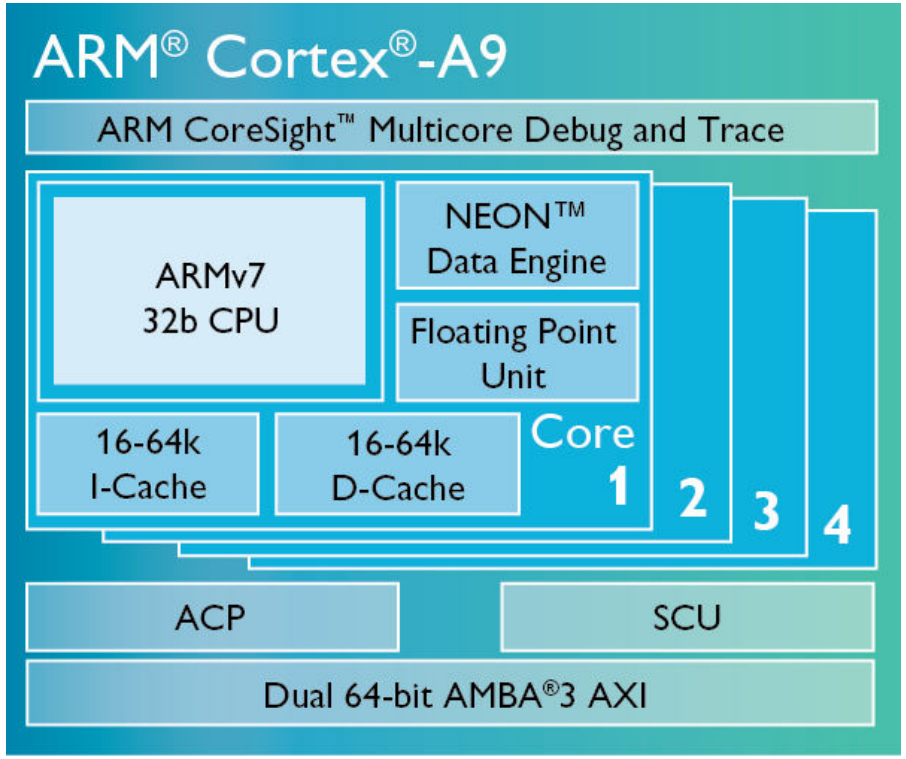
Component	Overview
Debug Access Port	Provide debugger access to the cores and buses in a SoC, across multiple power and clock islands, enabling exceptionally high download speeds direct to memory.
Embedded Cross Trigger	Synchronize debug and trace across multiple cores.
Program Trace Macrocells (Cortex-A9)	Noninvasively generate cycle-accurate, instruction trace of Arm processors running at full speed.
Embedded Trace Macrocells (Cortex-A5,Cortex-A8)	Noninvasively generate cycle-accurate, instruction and data trace of Arm processors running at full speed.
Trace Funnel	Combine multiple trace sources together.

Debugging Cotex-A9 processor using CoreSight

Component	Overview
Embedded Trace Buffer	Store trace data on-chip at high rates at 32-bit data width, eliminating the need for dedicated trace port pins or an external trace collection unit.
Trace Port Interface Unit	Transmit trace data off-chip via 2-34 pins at frequencies asynchronous to the core. Instrumentation Trace Macrocell for high level, low bandwidth, software generated trace.
Serial Wire Debug	High performance 2-pin debug port that replaces the 5/6-pin JTAG interface with multi-drop support.
Serial Wire Viewer	Single pin output for Instrumentation Trace.
Integration Kit	Contain RTL testbench, test vectors and full documentation for easy validation of a designer's own CoreSight subsystem

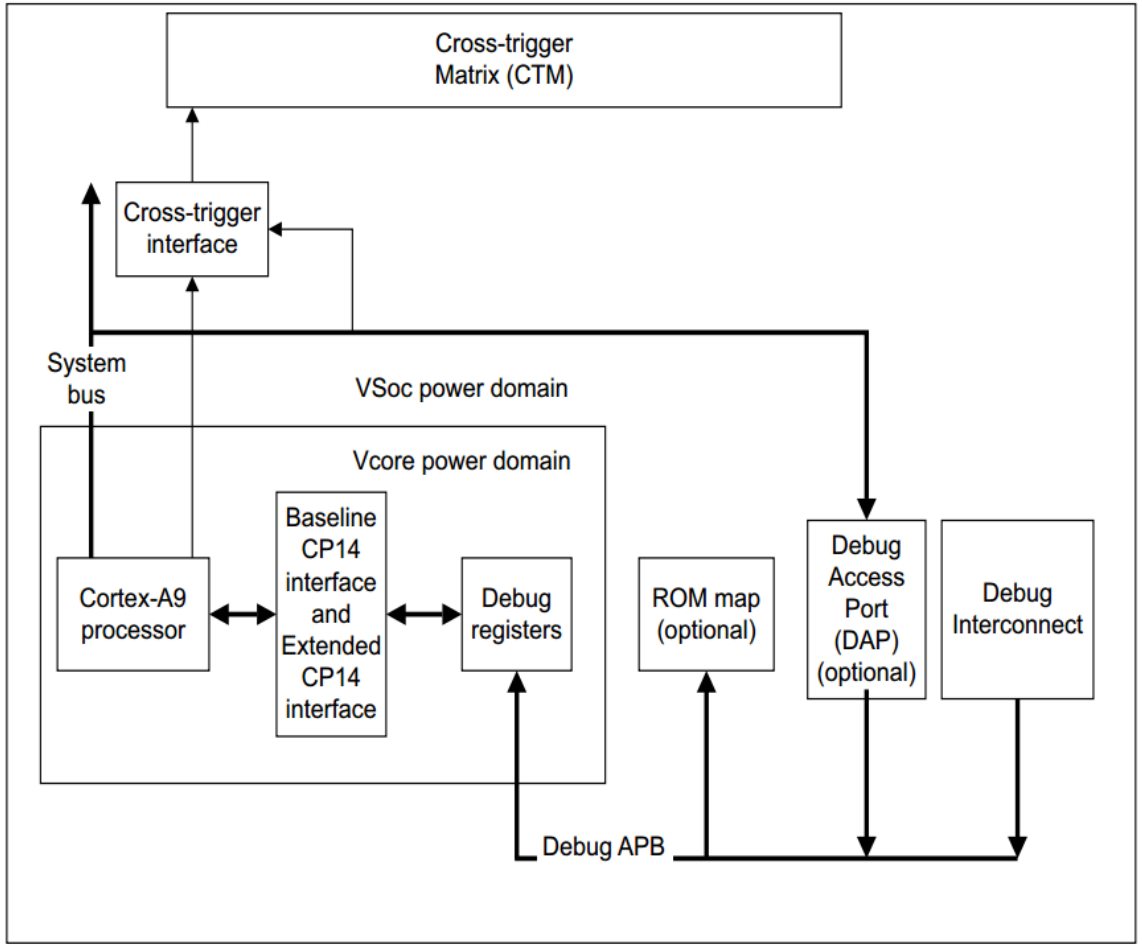
Arm Cortex-A9 debug interface

- Debug interface
  - The Cortex-A9 processor implements the ARMv7 debug architecture.
- Debugging modes:
  - **Halt mode** debugging in Secure user mode.
  - **Monitor mode** debugging in Secure user mode.
- Breakpoints and watchpoints
  - Six breakpoints.
  - Four watchpoints.
- Asynchronous aborts



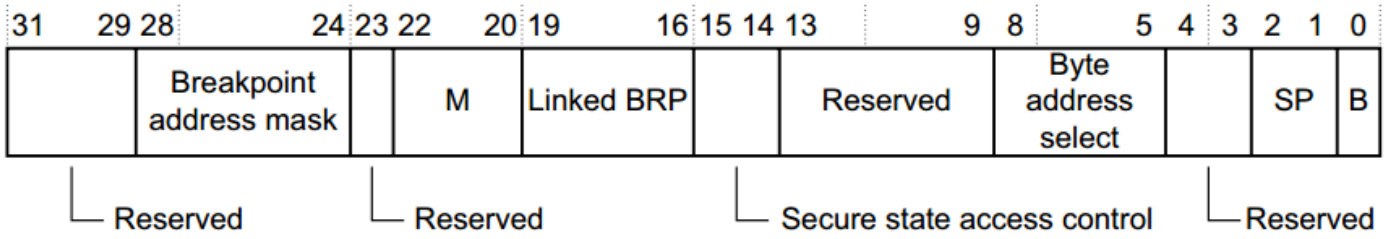
Arm Cortex-A9 Debug Interface

- The debug interface consists of:
  - A Baseline CP14 interface
  - An Extended CP14 interface
  - An external debug interface connected to the external debugger through a Debug Access Port (DAP).



Debug Register Descriptions

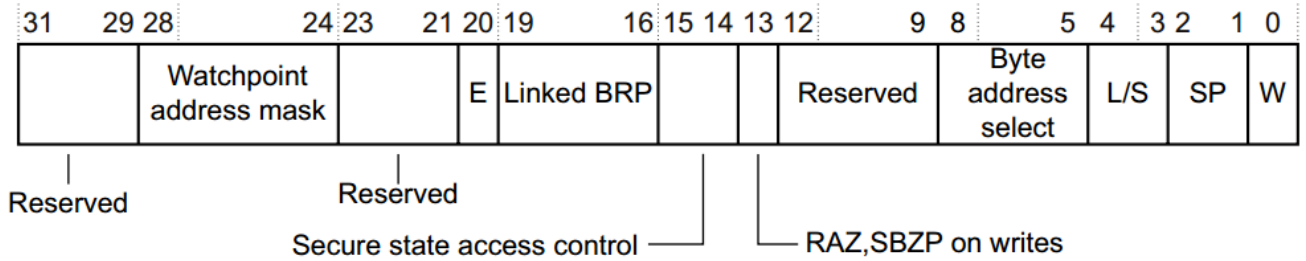
- Debug State Cache Control Register (DBGDSCCR)
  - Controls cache behavior while the processor is in Debug state.
  - Cortex-A9 processor does not implement any of the features of the DBGDSCCR, and it reads as zero.
- Breakpoint Value Registers (BVRs)
  - Registers 64-68, at offsets 0x100-0x114.
  - Each BVR is associated with a Breakpoint Control Register (BCR)
- Breakpoint Control Registers (BCRs)
  - Read and write registers that contain the necessary control bits for setting:
    - breakpoints
    - linked breakpoints



arm

Debug Register Descriptions

- Watchpoint Value Registers (WVRs)
  - Registers 96-99, at offsets 0x180-0x18C.
  - Each WVR is associated with a Watchpoint Control Register (WCR).
- Watchpoint Control Registers (WCRs)
  - Contain the necessary control bits for setting:
    - watchpoints
    - linked watchpoints

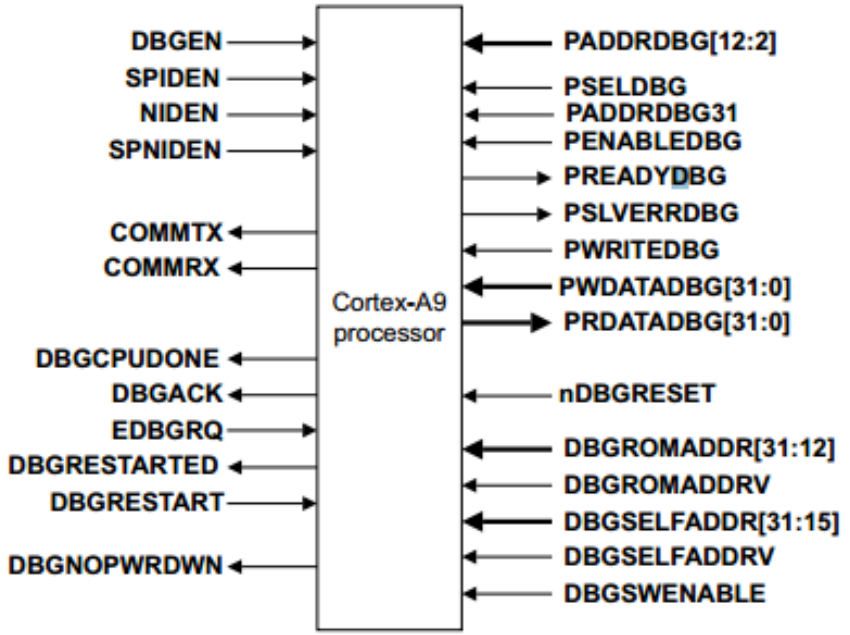


- Debug management registers
  - Define the standardized set of registers that is implemented by all CoreSight components.

arm

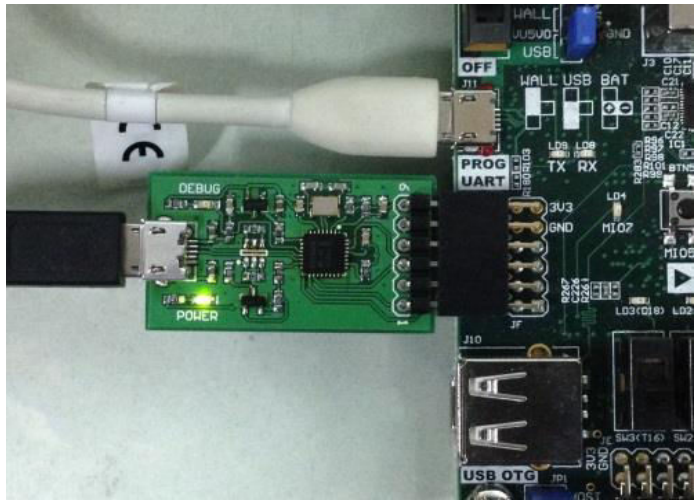
Debug Register Descriptions

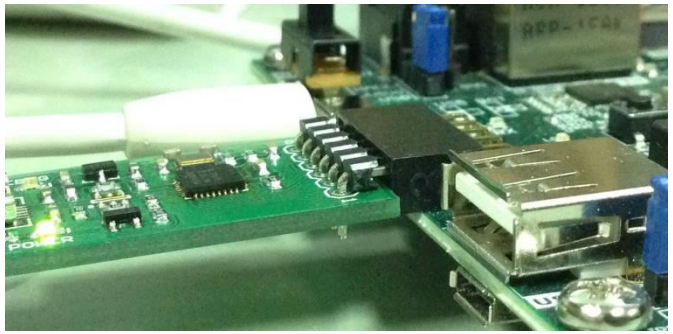
- Processor ID Registers
  - Read-only registers that return the same values as the corresponding CP15 ID Code Register and Feature ID Register
- CoreSight Identification Registers (Debug Management Registers)
  - Read-only registers that consist of the Peripheral Identification Registers and the Component Identification Registers.
- External debug interface
  - The system can access memory-mapped debug registers through the Cortex-A9 APB Completer port.
  - This APB Completer interface supports 32-bits wide data, stalls, completer-generated aborts, and eleven address bits [12:2] mapping 2x4KB of memory.
- More information on Cortex-A9 Technical Reference Manual



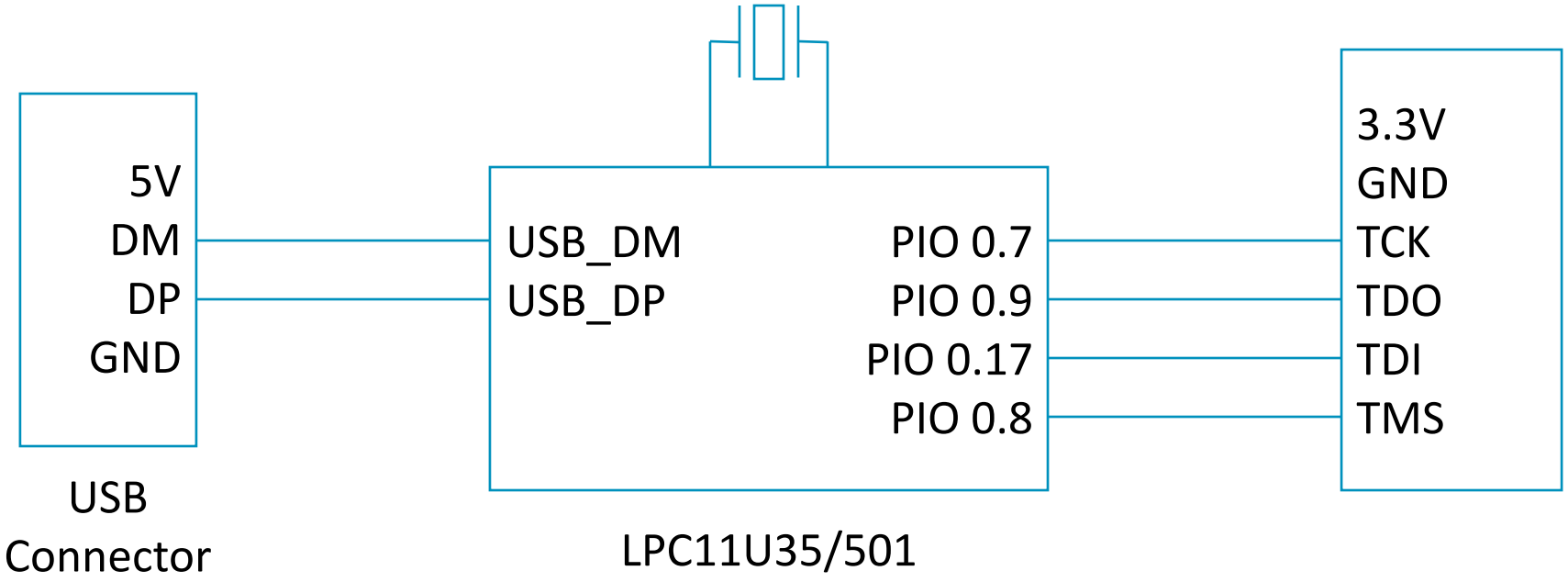
CMSIS-DAP Debugger Probe

- CMSIS-DAP Debugger Probe
  - Provides a standardized way to access the CoreSight Debug Access Port (DAP) of an Arm Cortex microcontroller via USB.
  - USB connection uses HID (Human Interface Device) driver class.
  - Based on NXP LPC11U35 MCU.
  - Low-cost and open-source.





CMSIS-DAP Debugger Probe



Bibliography

- Cortex-A9 Technical Reference Manual, Arm, 2012.
- Arm® Architecture Reference Manual. Armv7-A and Armv7-R edition, Arm, 2012.
- Arm CoreSight Architecture Specification, Arm, 2013.
- Arm CoreSight SoC-400 Technical Reference Manual, Arm, 2015.
- Arm Debug Interface Architecture Specification. ADIv5.0 to ADIv5.2. ARM, 2013.
- CoreSight technical introduction. A Quick Start for Designers. White Paper, ARM-EPM-039795, Arm, 2013.