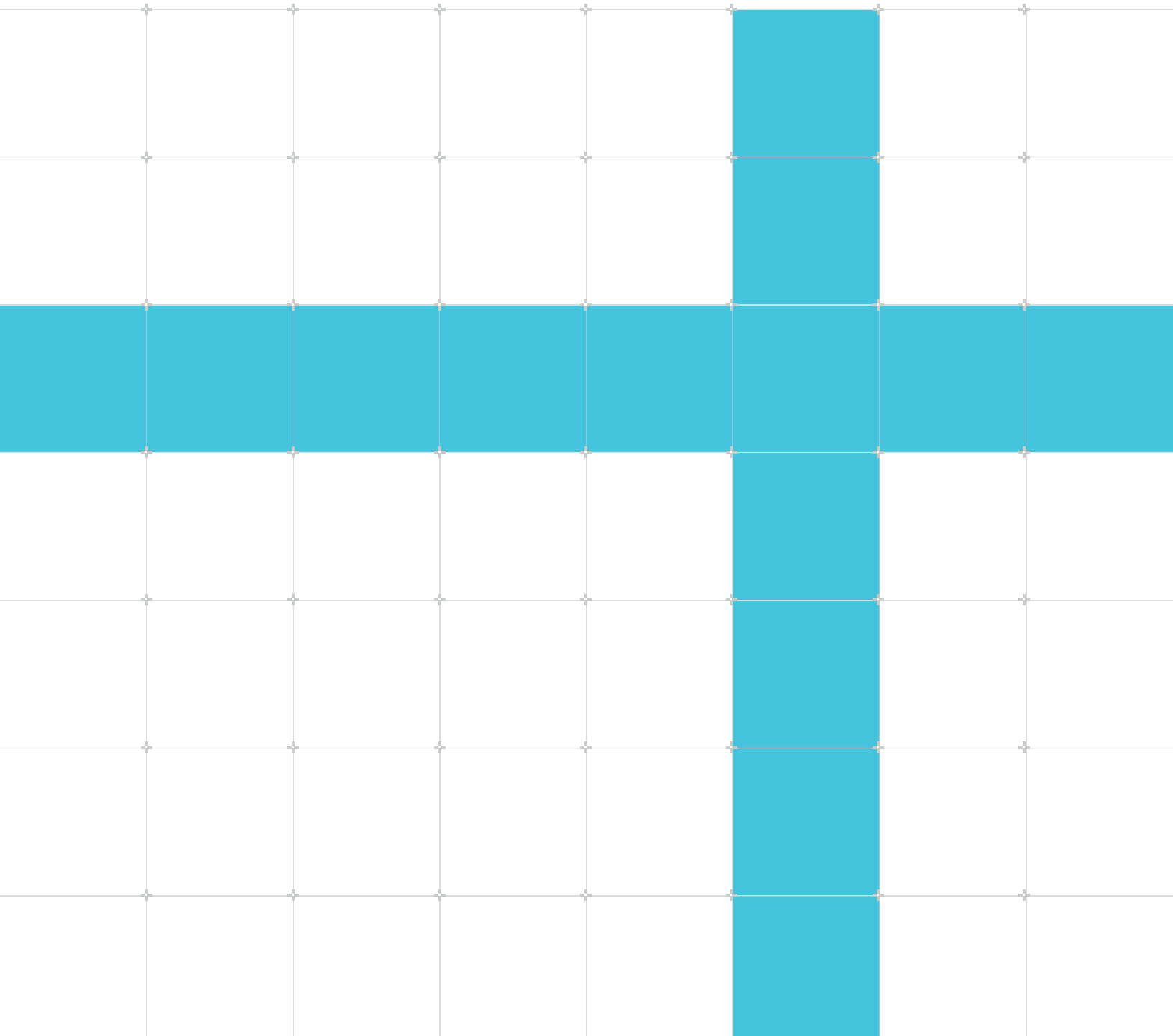
Debug connections to your hardware

Version 1.0

Issue 02
102435_0100_02_en



Debug connections to your hardware

Release information

Document history

Issue	Date	Confidentiality	Change
0100-02	25 February 2020	Non-Confidential	Initial release

Proprietary Notice

or indirectly, in violation of such export laws. Use of the word “partner” in reference to Arm’s customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349|version 21.0)

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.

Unrestricted Access is an Arm internal classification.

Product Status

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

Feedback

Arm® welcomes feedback on this product and its documentation. To provide feedback on the product, create a ticket on https://support.developer.arm.com

To provide feedback on the document, fill the following survey: https://developer.arm.com/documentation-feedback-survey.

Inclusive language commitment

Arm values inclusive communities. Arm recognizes that we and our industry have used language that can be offensive. Arm strives to lead the industry and create change.

We believe that this document contains no offensive language. To report offensive language in this document, email terms@arm.com.

Contents

1. Overview

For embedded system development, you will usually want to connect your target hardware to a host to perform debugging. Arm provides both JTAG and Serial Wire Debug (SWD) connections through a range of debug probes which are tuned to the needs of the system:

- DSTREAM and DSTREAM-ST enable powerful software debug and optimization of any Arm processor-based hardware target.
- The ULINK family provides uncomplicated debug, enabling cost efficient run-control debug on devices ranging from microcontrollers to multicore application processors.
- CMSIS-DAP provides simple debug over driverless USB.

For application development on Linux or Android, it is common to use GDB (the GNU Project Debugger) or ADB (Android Debug Bridge).

Debug Probes and Adapters gives more information, including a comprehensive overview and comparison of the available solutions.

2. Debugging with DSTREAM and DSTREAM-ST

DSTREAM and DSTREAM-ST enable powerful software debug and optimization of any Arm processor-based hardware target.

The following resources will help you get started debugging with DSTREAM:

- Arm DS-5 Debugger Linux kernel debug example with DSTREAM and BeagleBoard setup. This short video guides you through debugging a sample Linux kernel using the DS-5 Debugger, a DSTREAM unit and a BeagleBoard.
- Video Tutorial on ARM Cortex-M Series - Debug and Trace. This is a technical tutorial detailing the key aspects of the debug and trace features available in the Arm Cortex-M series of processors.
- DSTREAM knowledge articles provide answers to real-world questions regarding DSTREAM.

3. Debugging with the ULINK family

The ULINK family of debug probes provides uncomplicated debug, enabling cost efficient run-control debug on devices ranging from microcontrollers to multicore application processors.

The following resources will help you get started debugging with the ULINK family:

- These introductory videos cover the key ULINKplus features:
  - Part One: Debug and Trace
  - Part Two: Power Measurement
  - Part Three: Test Automation and IO pin control
- The Introducing ULINKplus webinar demonstrates how to use ULINKplus to provide better insight into the overall execution of the embedded system.
- The Getting Started with ULINKpro trace video shows how to connect to a target and capture trace information.
- The ULINKpro Debug Adapter video describes the unique features of the Keil ULINKpro Debug Adapter.

4. Debugging with CMSIS-DAP

CMSIS-DAP provides simple debug over driverless USB.

The following resources will help you get started debugging with the ULINK family:

- The Enabling CMSIS-DAP debug on the Tower System tutorial shows how to update the OpenSDA firmware on your NXP Semiconductors Vybrid Controller Tower System module to enable multicore debug with Arm DS-5 via USB using CMSIS-DAP (Cortex Microcontroller Software Interface Standard - Debug Access Port).
- The CMSIS-DAP Debugger User's Guide describes the configuration options of the CMSIS-DAP Debugger driver implemented in µVision. The driver can be used to flash and debug applications on Cortex-M processor-based devices.

5. Debugging with GDB

For application development on Linux or Android, it is common to use GDB (the GNU Project Debugger) for debugging.

The following resources will help you get started debugging with GDB:

- The Linux Application Debug Tutorial shows how to debug the open source fractal rendering application Xaos running on Linux on an Arm Cortex-A8 processor-based development board using Arm Development Studio 5 (DS-5) Community Edition.
- The Learn how to debug an ARM Linux application on a Pandaboard video demonstrates debugging on the the free toolchain ARM DS-5 Community Edition.
- The Debugging Linux applications on the Altera SoC with ARM DS-5 video shows how to use ARM DS-5 Altera Edition, part of the Altera SoC EDS toolkit, to debug a Linux application running on an Altera Cyclone V SoC-based board.