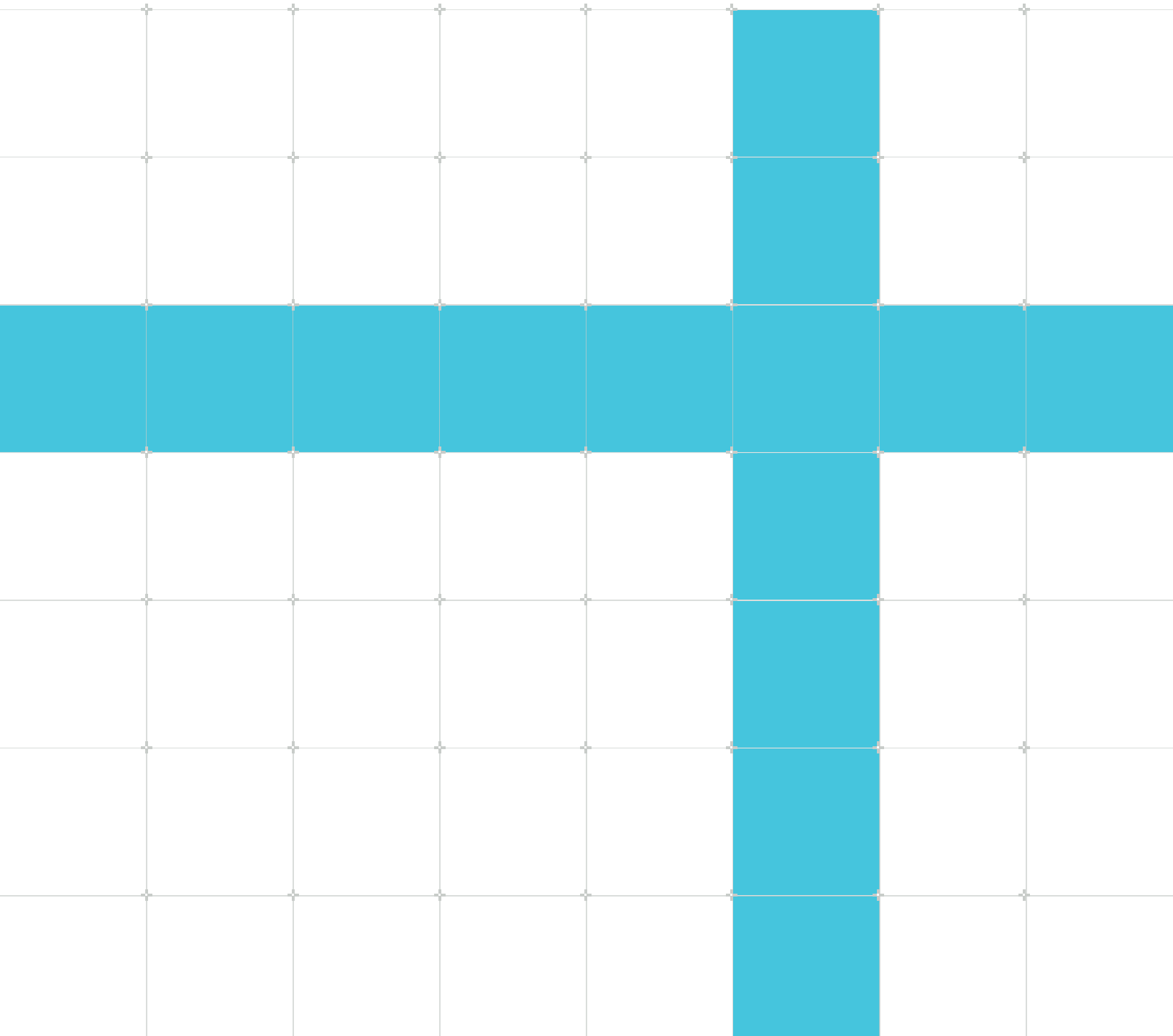
KAN347 - Debug Probe Firmware Update Information

Version 1.0

Application Note

Issue
109243_1.0_en



KAN347 - Debug Probe Firmware Update Information

Application Note

Release information

Document history

Issue	Date	Confidentiality	Change
1.0	7 July 2023	Non-Confidential	Initial release

Proprietary Notice

and regulations to assure that this document or any portion thereof is not exported, directly 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. Abstract

Keil Studio (Keil Studio Cloud and Keil Studio Visual Studio Code extensions) in MDK v6 supports a wide variety of debug probes based on CMSIS-DAP v2.x and ST-LINK/V2 and above. Some development boards require an update of the on-board firmware to work with Keil Studio.

This application note points to update information for on-board debug probes.

2. CMSIS-DAP 2.x

Keil Studio supports debug probes that run the CMSIS-DAP protocol.

See the CMSIS-DAP documentation for general information.

The following implementations are available:

- Arm Keil ULINKplus (firmware version 2)
- Arm Mbed DAPLink: boards form various vendors are supported.
- Nuvoton Nu-Link2
- NXP development boards

2.1 NXP development boards

This section explains the different CMSIS-DAP-based options for various NXP development boards.

MCU-Link

The latest debug probe from NXP is the MCU-Link. It is based on the LPC55S69 microcontroller to provide high-performance debug over high-speed USB. Download the latest firmware from the link above and make sure that you flash a version 3.x onto the development board.

Refer to MCU-Link.

i.MXRT10xx development boards

i.MXRT10xx-based development boards contain an on-bard debug adapter that is based on the LPC4322 device.

The application note Modifying Debug Firmware on i.MX RT10xx Boards Featuring An LPC4322-based Debug Probe explains how to update the on-board debug adapter firmware for i.MXRT10xx-based development boards.

OpenSDA

Many NXP boards can be flashed with a CMSIS-DAP 2.x-based OpenSDA firmware (OpenSDA2.x is required). The page for each board explains how to flash the firmware image onto it. Note that additional command line tools might be required.

Board list:

FRDM-K20D50M, FRDM-K22F, FRDM-K28F, FRDM-K32L2A4S, FRDM-K32L2B3, FRDM-K32L3A6, FRDM-K64F, FRDM-K66F, FRDM-K82F, FRDM-KE02Z, FRDM-KE04Z, FRDM-KE06Z, FRDM-KE15Z, FRDM-KL05Z, FRDM-KL25Z, FRDM-KL26Z, FRDM-KL27Z, FRDM-KL28Z, FRDM-KL43Z, FRDM-KL46Z, FRDM-KL82Z, FRDM-KLO2Z, FRDM-KLO3Z, FRDM-KV10Z, FRDM-KV31F, FRDM-KW24D512, FRDM-KW36, FRDM-KW38, FRDM-KW40Z, FRDM-KW41Z, FRDM-KW019032, HVP-KE18F, MIMXRT1010-EVK, MIMXRT1015-EVK,

MIMXRT1020-EVK, MIMXRT1024-EVK, MIMXRT1050-EVK, MIMXRT1060-EVK, MIMXRT1064-EVK, MIMXRT1170-EVK, TRK-KEA8, TRK-KEA64, TRK-KEA128, TWR-K22F120M, TWR-K64F120M, TWR-K65F180M, TWR-K80F150M, TWR-KE18F, TWR-KL25Z48M, TWR-KL28Z72M, TWR-KL43Z48M, TWR-KL46Z48M, TWR-KL82Z72M, TWR-KM34Z50M, TWR-KM35Z75M, TWR-KV10Z32, TWR-KV11Z75M, TWR-KV31F120M, TWR-KV46F150M, TWR-KV58F220M, TWR-KW21D256, TWR-KW24D512, USB-KW38, USB-KW40Z, USB-KW41Z, USB-KW019032

2.2 Linux

The Linux subsystem `udev` supplies the computer with device events. In order to access USB devices, it is common to specify `udev` rules.

Here are some sample rules files for different CMSIS-DAp implementations:

- DAPLink:

```
SUBSYSTEM=="usb", ATTR{idVendor}=="0d28", ATTR{idProduct}=="0204", MODE:="666"
```

- LPC-LinkII:

```
SUBSYSTEM=="usb", ATTR{idVendor}=="1fc9", ATTR{idProduct}=="0090", MODE:="666"
```

- Nu-Link:

```
ACTION!="add|change", GOTO="openocd_nuvoton_rules_end"
SUBSYSTEM!="usb|tty|hidraw", GOTO="openocd_nuvoton_rules_end"

# Nuvoton NuLink VID:PID
ATTRS{idProduct}=="511b", ATTRS{idVendor}=="0416", MODE="660", GROUP="plugdev",
 TAG+="uaccess"
ATTRS{idProduct}=="511c", ATTRS{idVendor}=="0416", MODE="660", GROUP="plugdev",
 TAG+="uaccess"
ATTRS{idProduct}=="511d", ATTRS{idVendor}=="0416", MODE="660", GROUP="plugdev",
 TAG+="uaccess"
ATTRS{idProduct}=="5200", ATTRS{idVendor}=="0416", MODE="660", GROUP="plugdev",
 TAG+="uaccess"

LABEL="openocd_nuvoton_rules_end"
```

- ULINKplus:

```
SUBSYSTEM=="usb", ATTR{idVendor}=="c251", ATTR{idProduct}=="2750", MODE:="666"
```

If you share your Linux system with other users, or just don't like the idea of write permission for everybody, you can replace `MODE:="0666"` with `OWNER:="yourusername"` to create the device owned by you, or with `GROUP:="somegroupname"` and mange access using standard unix groups.

3. STMicroelectronics ST-LINK

Keil Studio supports ST-LINK debug adapters starting with V2. Some issues might arise around drivers and host operating system specific rules.

The recommended versions of the ST-LINK firmware are:

- For ST-LINK/V2 and ST-LINK/V2-1 probes: J36 and later.
- For STLINK-V3 probes: J6 and later.

See “Firmware naming rules” in Overview of ST-LINK derivatives for more details on naming conventions.

3.1 Firmware update

Before using an ST-LINK with Keil Studio, consider upgrading the probe’s firmware.

Use the STSW-LINK007 firmware upgrade application to update the debug adapter firmware on your development board to the latest version.

3.2 Windows

STMicroelectronics devices may require a driver to be installed on Windows before they can be discovered.

Please find the drivers here:

- STSW-LINK009

3.3 Linux

The Linux subsystem `udev` supplies the computer with device events. In order to access USB devices, it is common to specify `udev` rules.

Here are some sample rules files for different ST-LINK hardware:

- STM32-Discovery boards with on-board ST-LINK/V2:

```
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="3748",
 MODE:="0666", SYMLINK+="stlinkv2_%n"
```

- STM32-NUCLEO boards with on-board ST-LINK/V2-1:

```
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="374b",
 MODE:="0666", SYMLINK+="stlinkv2-1_%n"
```

```
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="3752",
 MODE:="0666", SYMLINK+="stlinkv2-1_%n"
```

- STLINK-V3-based boards (standalone and embedded) in usbloader mode and standard (debug) mode:

```
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="374d",
 MODE:="0666", SYMLINK+="stlinkv3loader_%n"
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="374e",
 MODE:="0666", SYMLINK+="stlinkv3_%n"
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="374f",
 MODE:="0666", SYMLINK+="stlinkv3_%n"
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="3753",
 MODE:="0666", SYMLINK+="stlinkv3_%n"
```


Note

If you share your Linux system with other users, or just don't like the idea of write permission for everybody, you can replace `MODE:="0666"` with `OWNER:="yourusername"` to create the device owned by you, or with `GROUP:="somegroupname"` and mange access using standard unix groups.