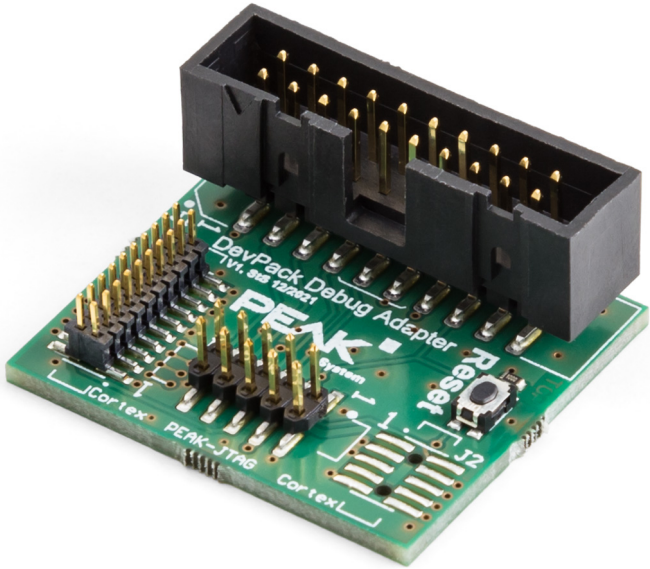
PEAK-DevPack Debug Adapter

User Manual



Relevant Product

Product name	Part number
PEAK-DevPack Debug Adapter	IPEK-003026

Imprint

PEAK-System Technik GmbH
Otto-Röhm-Straße 69
64293 Darmstadt
Germany

Phone: +49 6151 8173-20
Fax: +49 6151 8173-29

www.peak-system.com
info@peak-system.com

Document version 1.3.0 (2023-12-06)

Contents

1 Introduction

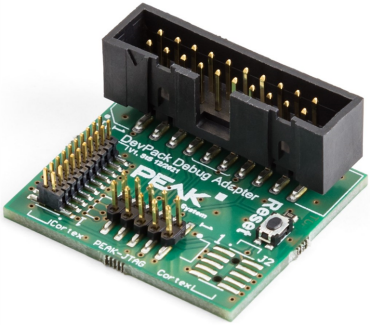
The PEAK-DevPack Debug Adapter allows the uncomplicated connection of a hardware debugger to different ARM-based, programmable hardware products from PEAK-System. For this purpose, the adapter has different connectors to connect the hardware debugger to specific JTAG connectors of the hardware to be programmed.

1.1 Properties at a Glance

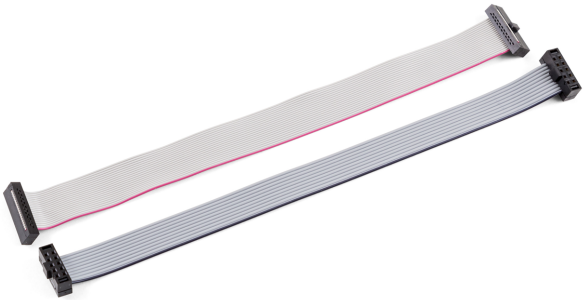
- Adapter for connecting a hardware debugger to the JTAG connector of a PCAN hardware
- Supported PCAN hardware products:
  - PCAN-Router Pro FD IPEH-002220 / IPEH-002222
  - PCAN-Router FD IPEH-002214 / IPEH-002215
  - PCAN-MicroMod FD IPEH-003080
  - PCAN-GPS IPEH-002110
  - PCAN-GPS FD IPEH-003110
- Reset button for restarting the PCAN hardware to be programmed

1.2 Scope of Supply

- PEAK-DevPack Debug Adapter



- Ribbon cables with colored pin 1
  - 1 x 20 pin with 1.27 mm grid spacing
  - 1 x 10 pin with 2 mm grid spacing



1.3 Intended Use

The PEAK-DevPack Debug Adapter is designed as a connection adapter between ARM-based, programmable PCAN hardware and a hardware debugger.

The PEAK-DevPack Debug Adapter has been tested with the hardware debugger ST-Link V2, the software Visual Studio Code, and the PEAK-DevPack Debug add-on. Using other hardware debuggers and/or software is possible, but not part of this manual.

2 Prerequisites

2.1 Hardware

Hardware debugger

For debugging a PCAN hardware, a hardware debugger is required. The PEAK-DevPack Debug Adapter was tested with the hardware debugger ST-Link V2. This hardware debugger is not included in the scope of supply.

More information about the ST-Link V2 is available on the website:
https://www.st.com/en/development-tools/st-link-v2.html

PCAN hardware

Various ARM-based, programmable hardware products from PEAK-System support debugging with a hardware debugger and the PEAK-DevPack Debug Adapter. Commission your PCAN hardware as described in the corresponding manual. The PCAN hardware must be supplied via a separate power source.

PCAN interface

A CAN interface from PEAK-System is required to transfer firmware via CAN. Commission the CAN interface as described in the corresponding manual.

2.2 Software

For debugging, the following software is required:

- Visual Studio Code with the following plug-ins installed:

  C/C++ plug-in
  https://marketplace.visualstudio.com/items?itemName=ms-vscode.cpptools

  Cortex-Debug plug-in
  https://marketplace.visualstudio.com/items?itemName=marus25.cortex-debug

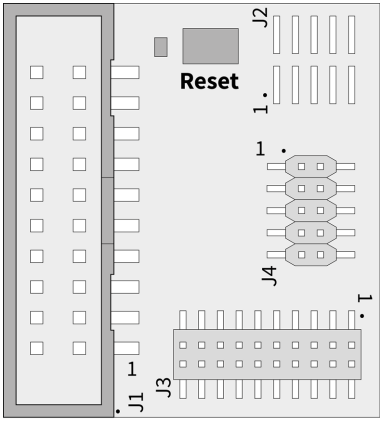
- Installed ST-Link V2 driver:
  https://www.st.com/en/development-tools/stsw-link009.html#overview

- PEAK-DevPack with included Debug add-on:
  https://www.peak-system.com/quick/DLP-DevPack

1. Save and unpack the *PEAK-DevPack* archive.
2. Execute the script `SetPath_for_VSCode.vbs`.
3. Change to the subdirectory `PEAK-DevPack/Debug`.
4. Execute the script `SetDebug_for_VSCode.vbs`.

3 Connectors

3.1 PEAK-DevPack Debug Adapter



Connector	Pins and pitch	Connector for
J1	20 pins 2.54 mm	St-Link V2 Debugger
J2	10 pins 1.27 mm	Not equipped
J3	20 pins 1.27 mm	PCAN-Router Pro FD PCAN-MicroMod FD PCAN-GPS FD
J4	10 pins 2.00 mm	PCAN-Router FD PCAN-GPS

Pin 1 of the connector fields is marked with a 1 on the circuit board.
Reset button is directly connected to the reset pin of the PCAN hardware CPU.

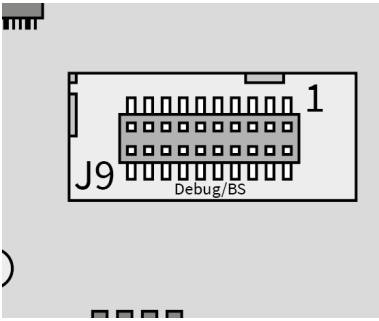
3.2 Programmable PCAN Hardware

Debugging is done via the JTAG connector of the PCAN hardware. On most PCAN hardware products, this connector is not equipped on delivery. Users must solder on the pin header themselves.

Risk of short circuit! Soldering on the PCAN hardware may only be performed by qualified electrical engineering personnel.

Attention! Electrostatic discharge (ESD) can damage or destroy components on the card. Take precautions to avoid ESD.

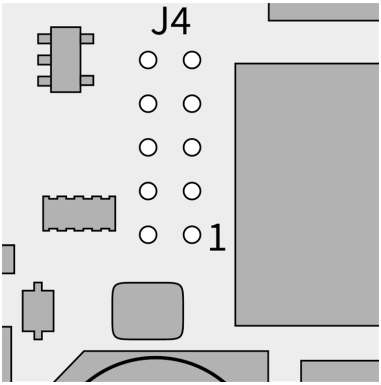
PCAN-Router Pro FD



JTAG connector	Pins and pitch	Pin 1 identifier
J9	20 pins, 1.27 mm	1

The JTAG connector of the PCAN-Router Pro FD is equipped on delivery.

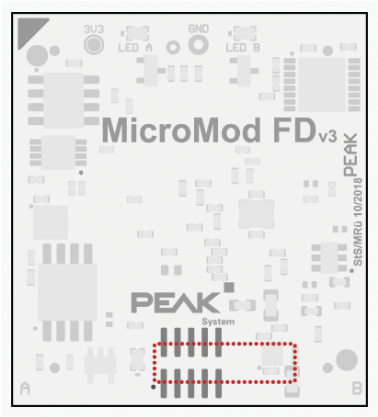
PCAN-Router FD



JTAG connector	Pins and pitch	Pin 1 identifier
J4	10 pins, 2.00 mm	1

The JTAG connector of the PCAN-Router FD is not equipped on delivery.

PCAN-MicroMod FD

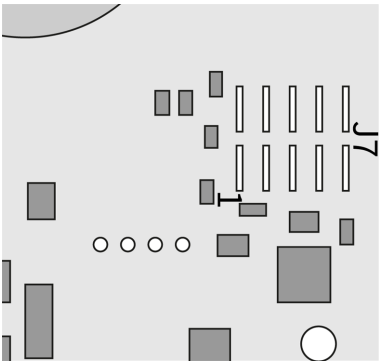


JTAG connector	Pins and pitch	Pin 1 identifier
without label	10 pin 1.27 mm	white dot

The JTAG connector of the PCAN-MicroMod FD is not equipped on delivery.

When using the 20-pin ribbon cable with 1.27 mm pitch, the connector will overlap as marked in red in the picture. Make sure that pin 1 of the cable and the circuit board are connected.

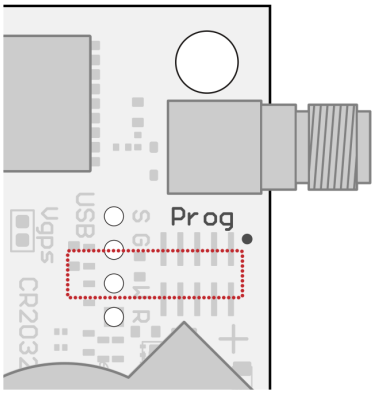
PCAN-GPS



JTAG connector	Pins and pitch	Pin 1 identifier
J7	10 pin 2.00 mm	1

The JTAG connector of the PCAN-GPS is not equipped on delivery.

PCAN-GPS FD



JTAG connector	Pins and pitch	Pin 1 identifier
Prog	10 pin 1.27 mm	white dot

The JTAG connector of the PCAN-GPS FD is not equipped on delivery.

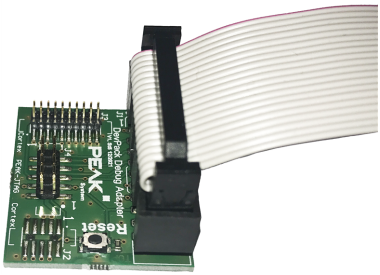
When using the 20-pin ribbon cable with 1.27 mm pitch, the connector will overlap as marked in red in the picture. Make sure that pin 1 of the cable and the circuit board are connected.

4 Connecting Hardware

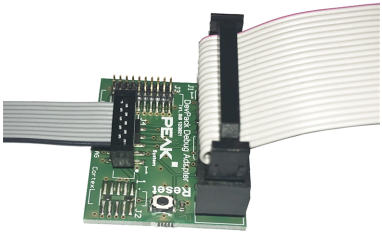
1. Connect the ST-Link to your computer. The LED lights up red.

2. Connect the 20 pin ribbon cable to the STM32 connector of the ST-Link.

3. Connect the other end of the 20 pin ribbon cable to the PEAK-DevPack Debug Adapter.

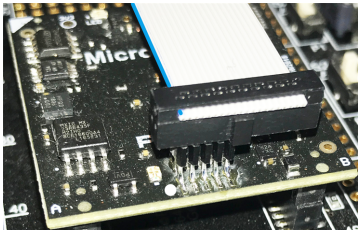


4. Connect the ribbon cable that is compatible with your PCAN hardware to the PEAK-DevPack Debug Adapter.



Make sure that pin 1 of the ribbon cable is connected to pin 1 of the PEAK-DevPack Debug Adapter.

5. Connect the ribbon cable to the JTAG connector of your PCAN hardware.



Shown with the example PCAN-MicroMod FD

Make sure that pin 1 of the ribbon cable is connected to pin 1 of the PCAN hardware.

Note: Pin 1 is colored on the ribbon cables and marked with a 1 or a dot on the PCAN hardware circuit boards and on the adapter.

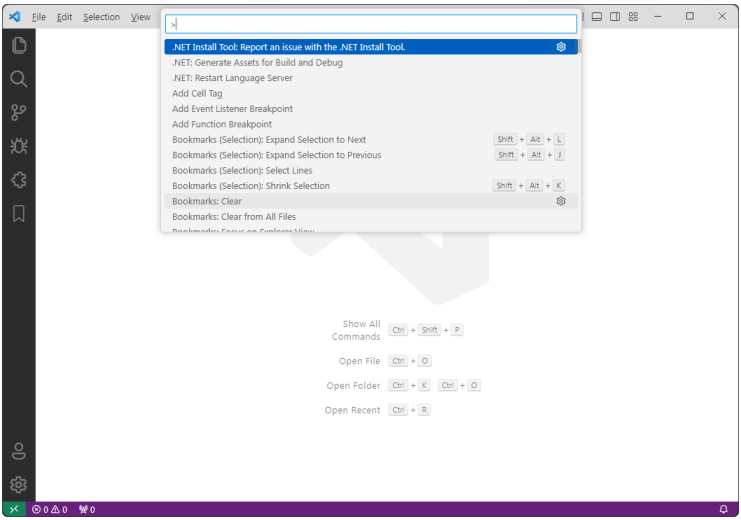
Note: For the PCAN-MicroMod FD and PCAN-GPS FD, the 20-pin ribbon cable with 1.27 mm pitch is intended for use. The connector of the ribbon cable will overlap, as the JTAG connector only has 10 pins shown in the example image above.

5 Debugging

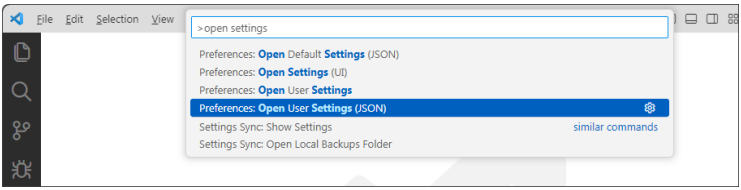
This chapter demonstrates the use of Visual Studio Code with the ST-Link V2 hardware debugger.

5.1 Set up Visual Studio Code

1. Launch Visual Studio Code.
2. Open the command line with F1 or alternatively with Ctrl + Shift + P.



3. Enter *open settings*.



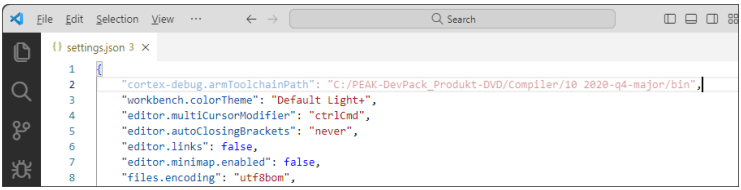
Visual Studio Code offers the following command, among others, for selection: *Preferences: Open User Settings (JSON)*

4. Confirm the selection with a click. The file `settings.json` is opened.

5. Insert the path to the file `arm-none-eabi-gdb.exe` in the source code.

```
"cortex-debug.armToolchainPath":
".../PEAK-DevPack/Compiler/10 2020-q4-major/bin/",
```

The placeholder `"..."` must be replaced by your directory path. This ensures that the Cortex Debug Plug-in knows the path to the compiler.

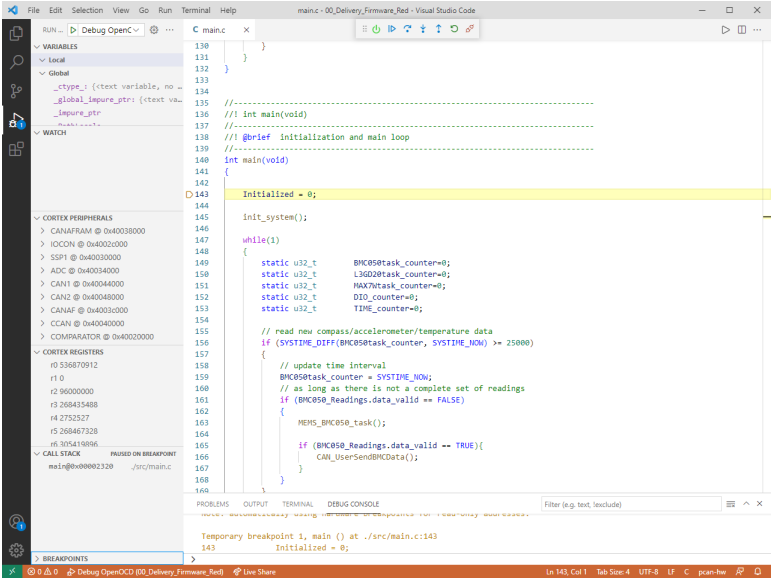


6. Save and close the file.

> **Note:** The file `settings.json` is usually saved in the path:
> `C:\Users\User\AppData\Roaming\Code\User\`
> The placeholder `User` must be replaced by your username.

5.2 PCAN Hardware Debugging

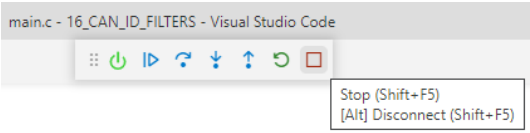
1. Open a sample project for your PCAN hardware in Visual Studio Code.
2. Compile the code before execution with *Make Debug* (Ctrl + Shift + B).
3. Start the debugger with F5. In the process, the previously compiled firmware is transferred to your PCAN hardware.



Keyboard shortcuts are available for the following commands:

Continue	F5
Step Over	F10
Step Into	F11
Step Out	Shift + F11
Restart	Crtl + Shift + F5
Stop	Shift + F5
Disconnect	Alt + Shift + F5

With *Disconnect* the debugger is quit.



5.3 Special Case: PCAN-GPS Debugging

Risk of damage! Attempting to upload the firmware with the Visual Studio Code Debug Toolchain can irreversibly damage the PCAN-GPS.

Uploading the PCAN-GPS firmware cannot be done directly with Visual Studio Code. The corresponding function is already disabled in the `launch.json` file for the PCAN-GPS. The upload of the firmware you have compiled must be done according to the PCAN-GPS user manual with PEAK-Flash. After the upload is done, the PCAN-GPS can be debugged with Visual Studio Code.

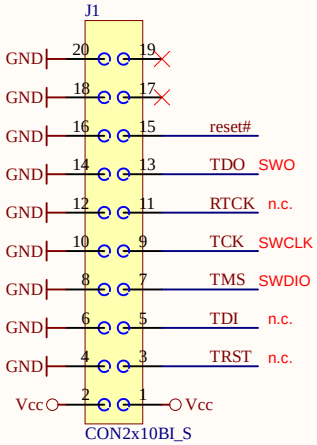
To do this, switch back to your Visual Studio Code project and start debugging by pressing the F5 key twice. The first press starts the debug environment and stops the PCAN hardware at a random point in the software. The second press causes a jump to the breakpoint, which was automatically set to the beginning of the main function.

Appendix A Pin Assignment

Minimum pin assignment required for SWD debugging on the respective connectors:

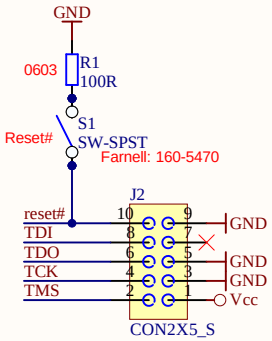
	J1 **ARM JTAG 20** **2.54 mm**	**J2** **Cortex 2x5** **1.27 mm**	**J3** **Cortex 2x10** **1.27 mm**	**J4** **PEAK-JTAG 2x5** **2.00 mm**
Hardware	ST-Link V2 Debugger	Not equipped	PCAN-Router Pro FD PCAN-MicroMod FD PCAN-GPS FD	PCAN-Router FD PCAN-GPS
Single Wire Signals				
SWO	13	6	6	7
SWDIO	7	2	2	6
SWCLK	9	4	4	5
V_{cc}	1/2	1	1	4
GND	4/6/8/.../20	3/5/9	3/5	1/2
SRST	15	10	10	3

The pin assignment of the PEAK-DevPack Debug Adapter connectors is shown in the following schematic.



ARM JTAG 20 2.54mm
(for. e.g.: ST-Link V2)

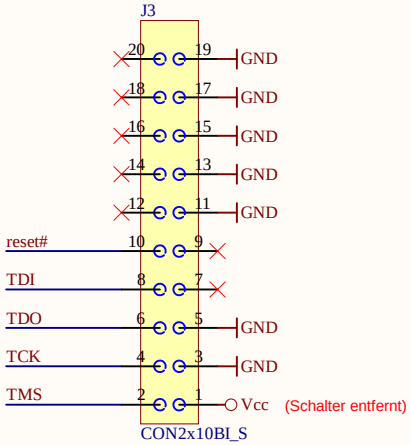
Amphenol 52601-S20-4LF
Mouser: 649-52601-S20-4LF



Cortex 2x5 1.27mm

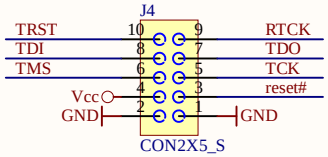
Samtec FTSH-105-01-L-DV-A-P
Mouser: 200-FTSH10501LDVAP

pin header SMT for debug target (if not populated):
Samtec FTSH-105-01-L-DV-P
Mouser 200-FTSH10501LDVP



Cortex 2x10 1.27mm

Samtec FTSH-110-01-L-DV-A-P
Mouser: 200-FTSH11001LDVAP



PEAK-JTAG 2x5 2mm

Amphenol 98424-F52-10ULF
Mouser: 649-98424-F52-10ULF

pin header THT for debug target (if not populated):
Hirose A3C-10P-2DSA(30)
Mouser 798-A3C-10P-2DSA30

PEAK-System Technik GmbH
Otto-Röhm-Str. 69
D-64293 Darmstadt

Engineer: StS
Date: 26.06.2022

Title: DevPack-DebugAdaptor
Sheet: Main
Customer: *
Version: 1.1
Variant: [No Variations]
File name: PEAK-DevPack-DebugAdapter_V1.SchDoc
Page 1 of 1