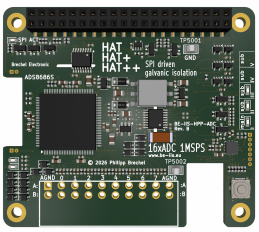
Isolated 16-Channel ADC HAT++ for Raspberry Pi

Brechel Electronic
Industrial Interface Solutions
Designed by Philipp Brechel (Germany)
www.be-iis.eu | www.github.com/be-iis



Document ID: BE-IIS-HPP-ADC_RevB_Datasheet
Revision: Rev. B
Date: October 24, 2026
Document Status: Draft
Product Status: Rev. B

16-channel analog input HAT for Raspberry Pi with 16-bit resolution. Two ADC cores simultaneously sample one channel from each of the two eight-channel input banks.

The ADS8686S from Texas Instruments provides software-selectable bipolar input ranges. An isolated SPI interface and onboard isolated supply separate the analog inputs from the Raspberry Pi. HAT++ selects the chip-select and BUSY connections.

Key Features

- 16-bit ADS8686S dual ADC
- 16 inputs in two banks of eight
- Simultaneous A/B sampling
- Software-selectable input ranges
- ±10 V, ±5 V and ±2.5 V
- Isolated SPI and power supply
- 20-pin analog connector
- HAT++ CS/BUSY selection
- GPIO23 conversion start
- 3.3 V and 5 V host supply

Product Description

The ADC HAT acquires single-ended voltage signals in ranges of ±10 V, ±5 V or ±2.5 V. All inputs share the isolated analog ground, GND_ISO.
The host controls acquisition over SPI. Rev. A requires exclusive use of the MISO line; details are given in the HAT++ section.

Applications

- Technology evaluation
- Analog sensor acquisition
- Prototyping
- Education and laboratory use
- Voltage measurement

Contents

1 Introduction **4**

1.1 General . . . 4
1.2 Raspberry Pi Compatibility . . . 4
1.3 HAT Compatibility Level . . . 5
1.4 Introduction: Optional HAT++ Stacking . . . 6
1.5 Design Resources . . . 7

2 Hardware Configuration **8**

2.1 Main Features . . . 8
2.2 Block Diagram . . . 9
2.3 Hardware Components . . . 10
2.3.1 ADS8686S Dual ADC . . . 10
2.3.2 HAT++ Control Logic . . . 10
2.4 Isolation . . . 11
2.5 Connectors . . . 12
2.5.1 J5001 Pin Assignment . . . 12
2.5.2 Connector System . . . 13
2.5.3 Wiring Instructions . . . 13
2.6 Indicators (LEDs) . . . 14

3 HAT++ Configuration **15**

3.1 Select the Board Configuration . . . 15
3.2 Combine Boards and Check Resources . . . 16
3.3 Install the Matching Software . . . 16

4 Software and System Configuration **17**

4.1 System Support . . . 17
4.2 Driver & Integration . . . 17
4.3 Hardware–Software Interaction . . . 18
4.4 System Inspection . . . 18
4.5 Interface Naming . . . 19

5 Delivery **20**

6 Mechanical **20**

6.1 Board Format . . . 20
6.2 3D Data . . . 20

7 Assembly **20**

7.1 Assemble 2x20-Pin Main Connector 20

7.2 Assemble Spacers 21

7.3 Assemble 3.5 mm Terminal Block Connector 21

7.4 Connect Analog Input Signals 21

7.5 Install ADC-HAT on Raspberry Pi 21

7.6 Power On 22

7.7 Board Overview 23

8 References **23**

9 Revision History **23**

Company Information **25**

1 Introduction

1.1 General

BE-IIS-HPP-ADC is an isolated analog input board for the Raspberry Pi. The ADS8686S [1] provides 16 single-ended inputs arranged in two banks of eight. One channel from each bank is sampled simultaneously, with the input range and channel sequence selected in software.

SPI, control signals and power are isolated from the analog input domain. HAT++ assigns the chip-select and BUSY connections, while GPIO23 starts conversion. The Rev. A MISO connection requires exclusive use of SPI0 or external output gating.

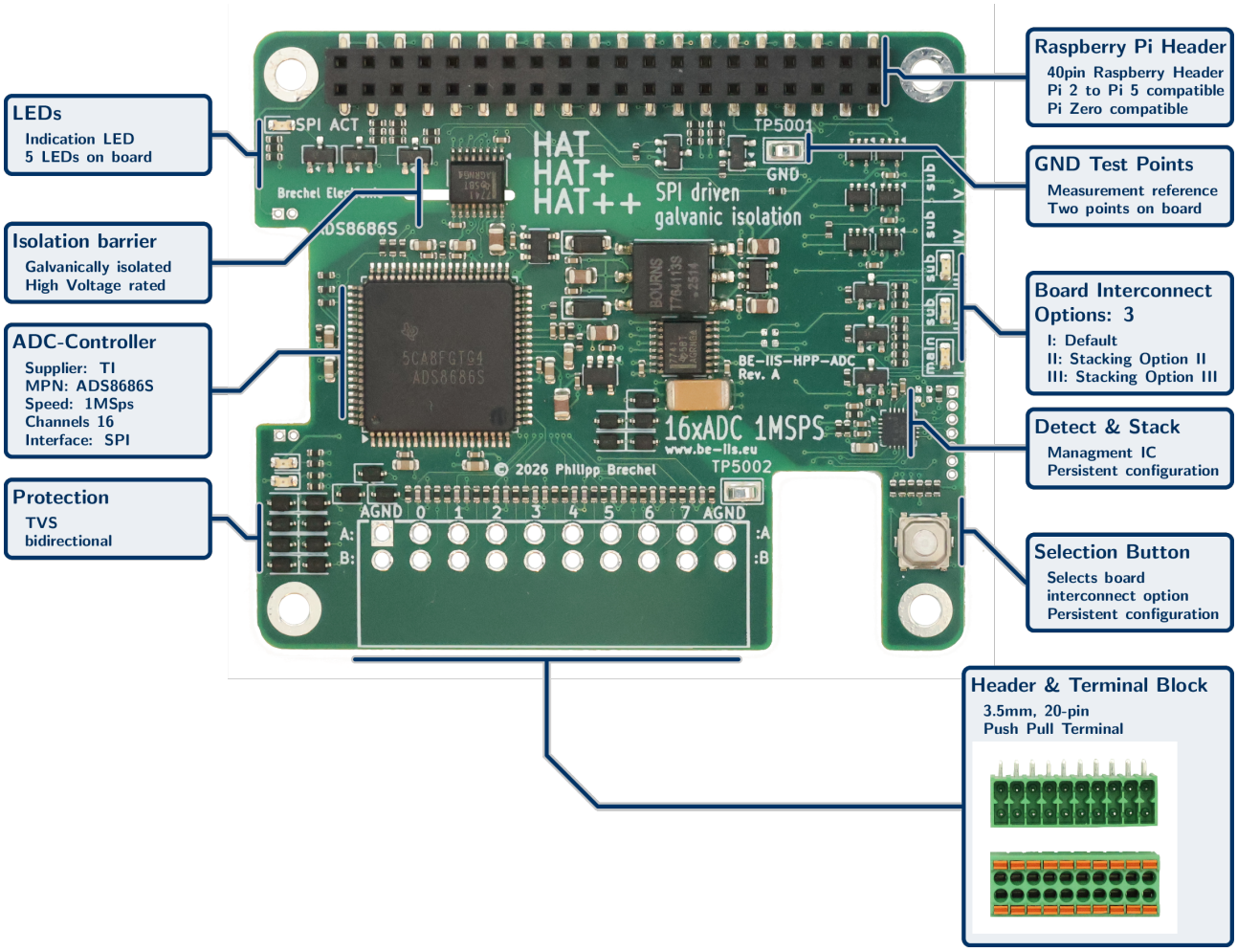


Figure 1: Annotated function overview

1.2 Raspberry Pi Compatibility

The board follows the Raspberry Pi HAT+ mechanical and electrical interface concept. Compatibility depends on the selected Raspberry Pi model, operating system architecture, enabled interfaces, and the installed BE-IIS software configuration.

Model	32-bit	64-bit
Raspberry Pi 2	✓	
Raspberry Pi 3	✓	✓
Raspberry Pi 4	✓	✓
Raspberry Pi 5	✓	✓
Raspberry Pi Zero	✓	
Raspberry Pi Zero 2 W	✓	✓

Table 1: Raspberry Pi compatibility

Note: – indicates a host and operating system combination awaiting validation.“‘

1.3 HAT Compatibility Level

Table 2 describes the HAT compatibility level of the board.

Level	Description	Support	Configuration
HAT	The board can be used as a standard Raspberry Pi HAT.	✓	3 CS/BUSY routes; MISO limitation applies
HAT+	The board supports the Raspberry Pi HAT+ auto-detection mechanism.	✓	Instance I; software support pending
HAT++	The board supports detection and stackable operation according to the BE-IIS HAT++ specification.	✓	3 CS/BUSY routes; MISO limitation applies

Table 2: HAT compatibility level

1.4 Introduction: Optional HAT++ Stacking

HAT++ is an extended function of the Raspberry Pi HAT+ concept. It enables stacking of up to five different HATs in one system. Depending on the supported instance modes, the same HAT type can be used up to three times in one stack.

Figure 2 gives a short overview of the required hardware configuration and the supported instance modes.

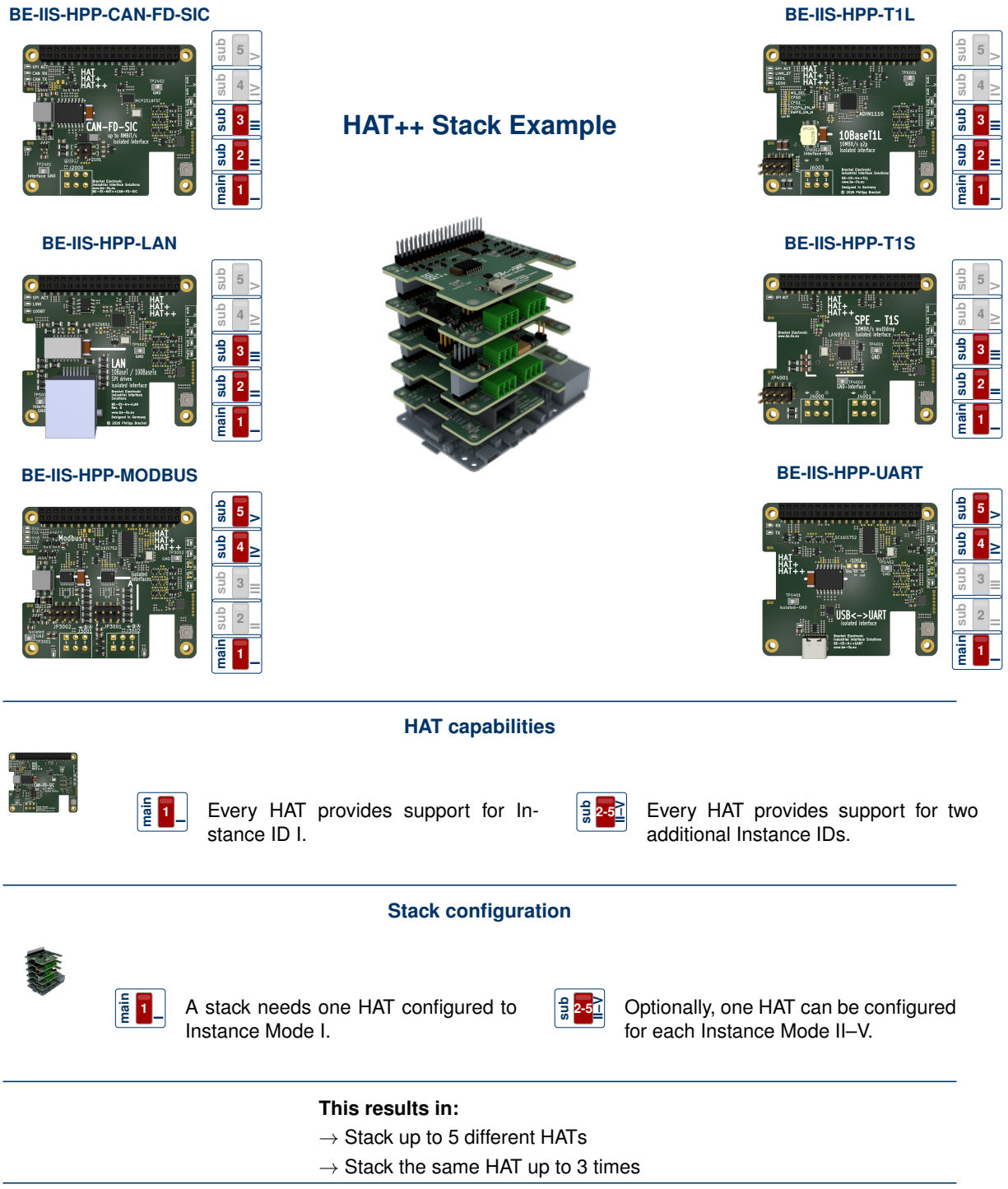


Figure 2: HAT++ stacking overview and instance mode configuration.

1.5 Design Resources

All design files and software resources are publicly available.

HTML **Product Page**
https://www.be-iis.eu/products/BE-IIS-HPP-ADC_B/

PDF **Datasheet (PDF)**
https://www.be-iis.eu/products/BE-IIS-HPP-ADC_B/datasheet.pdf

PDF **Schematic (PDF)**
https://www.be-iis.eu/products/BE-IIS-HPP-ADC_B/schematic.pdf

HTML **Layout & BOM (Interactive)**
https://www.be-iis.eu/products/BE-IIS-HPP-ADC_B/ibom.html

STL **3D Model (STEP/STL)**
https://www.be-iis.eu/products/BE-IIS-HPP-ADC_B/model.zip

GIT **GitHub Repository**
https://github.com/be-iis/be-iis-docs/tree/main/B-Sample/BE-IIS-HPP-ADC

GIT **Installer Script**
https://github.com/be-iis/be-iis-installer/blob/main/scripts/install/install-all.sh

DigiKey
Coming soon

SELL ON TINDIE
Coming soon

BUY IT ON LECTRONZ
Coming soon

2 Hardware Configuration

2.1 Main Features

The analog inputs are arranged in two multiplexed banks, each connected to one ADC core. The ADS8686S samples the selected A/B channel pair simultaneously and transfers the conversion results through the isolated SPI interface.

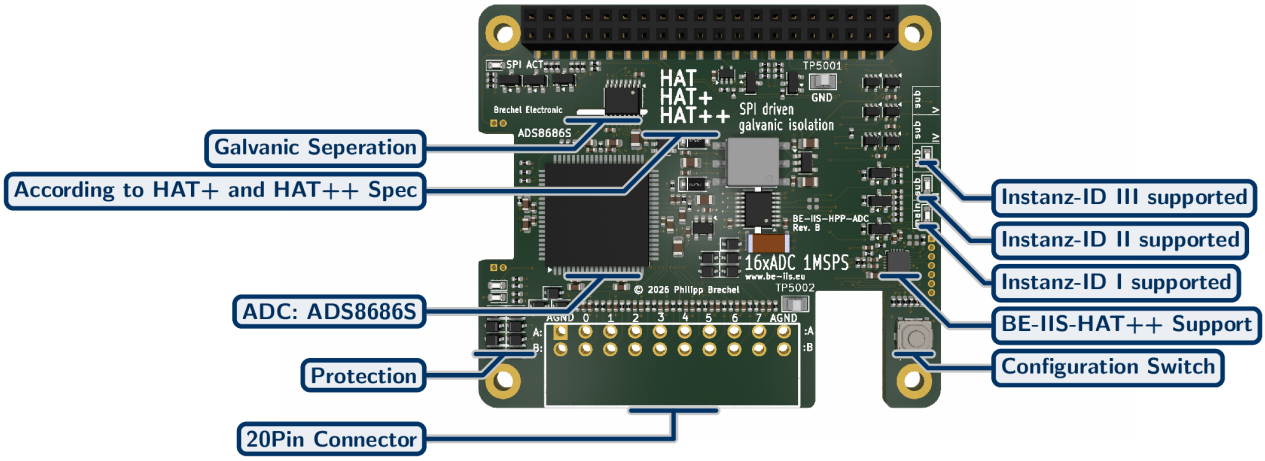


Figure 3: Annotated overview of the BE-IIS-HPP-ADC

ADC

- Texas Instruments ADS8686S
- 16-bit resolution; two ADC cores
- Eight input channels per bank
- Two input banks: A and B

SPI Interface

- SPI0 via the Raspberry Pi 40-pin GPIO header
- CS0, CS1 and CS2 for parallel operation of multiple boards
- GPIO23: CONVST; GPIO13: RESET

Galvanic Isolation

- Separate analog measurement ground
- Galvanic isolation between the measurement circuit and Raspberry Pi

Input Ranges

- ± 10 V, ± 5 V and ± 2.5 V
- Individually software-configurable per channel

Input Circuit

- 100 Ω series resistor and 1 nF filter capacitor
- Bidirectional transient voltage protection

Conversion

- Simultaneous sampling of one A/B channel pair
- Up to 1 MSPS per ADC core (device limit)
- Actual throughput depends on host performance and SPI configuration

HAT++

- Supports stacking up to three identical ADC-HATs
- Supports stacking up to five compatible HAT++ boards
- Up to 48 analog input channels per Raspberry Pi

2.2 Block Diagram

The analog signal path runs from the 20-pin terminal block through the input filters and transient voltage protection to the ADS8686S ADC.

Digital isolators provide galvanic isolation for the SPI interface and control signals between the ADC and the Raspberry Pi.

An isolated transformer-based power supply provides power to the analog measurement domain.

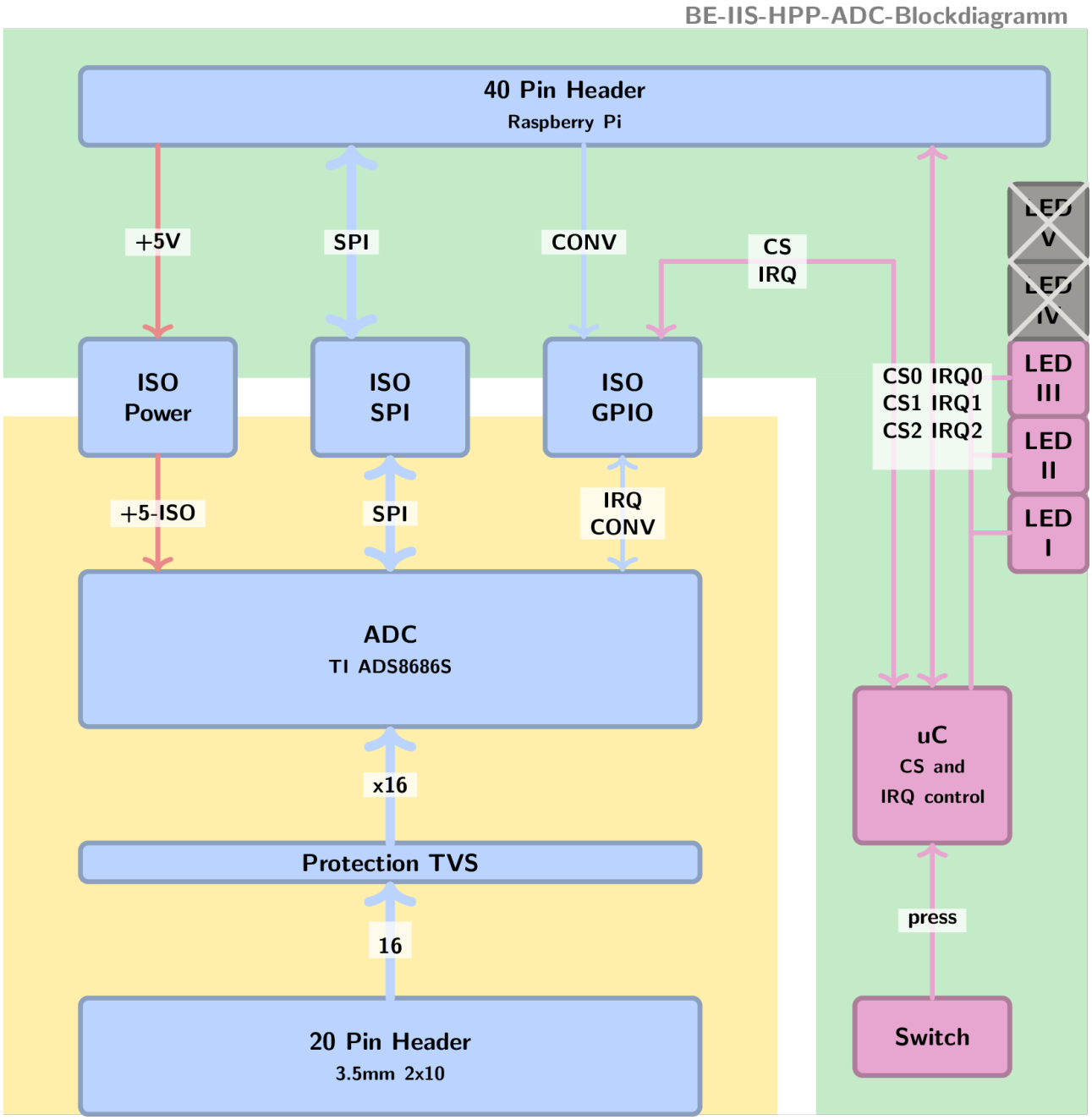


Figure 4: Block diagram of the BE-IIS-HPP-ADC

The ADC-HAT is designed for integration into the BE-IIS HAT++ ecosystem, supporting the parallel operation of up to five compatible boards on a single Raspberry Pi 40-pin GPIO interface.

Up to three ADC-HATs can be operated simultaneously, providing a total of 48 analog input channels per Raspberry Pi.

2.3 Hardware Components

The acquisition circuit is based on the ADS8686S (U5006).

U5004 and U5005 isolate its digital interface from the host.

The SN6501 driver (U5003) and transformer TR5001 supply the isolated domain, where separate regulators provide nominal 5 V rails for AVDD (U5002) and DVDD (U5001).

2.3.1 ADS8686S Dual ADC

The **ADS8686S** by *Texas Instruments* is a **16-channel, 16-bit successive approximation register (SAR) analog-to-digital converter** with an integrated analog front-end, designed for precision industrial data acquisition and measurement applications.

The device features two **simultaneously sampling ADCs**, each connected to eight analog input channels through a multiplexer. It supports a maximum conversion rate of **1 MSPS per ADC** and independently programmable bipolar input ranges of $\pm$**10 V**, $\pm$**5 V**, and $\pm$**2.5 V**, with an optional 20% overrange. Each input includes a high-impedance **1 MΩ analog front-end**, input protection, a programmable gain amplifier, and a configurable low-pass filter.

The ADS8686S integrates a precision voltage reference, configurable channel sequencing, digital averaging, and optional **CRC error checking**. Host communication is supported through **SPI**, parallel, or byte-wide digital interfaces, making it suitable for both microcontroller-based systems and embedded Linux platforms.

In Linux-based systems, the ADS8686S can be accessed through an **SPI interface** using a suitable device-specific driver or userspace application, enabling multichannel voltage measurements, data logging, and industrial signal monitoring. The achievable acquisition rate depends on the host interface and software implementation.

Electrical characteristics and register descriptions are provided in the device datasheet (Ref. 1).

2.3.2 HAT++ Control Logic

The control logic manages HAT+, HAT++, and application-specific functionality. A **Texas Instruments microcontroller** is used to implement the control logic. On the **I^{2}C0 bus**, it behaves like an **AT28-compatible EEPROM**, thereby enabling the **HAT++ functionality** while maintaining compatibility with the standard HAT+ detection mechanism.

Tri-state buffers are used to switch the control paths and to selectively connect the required control electronics depending on the active operating mode.

Functions:

- HAT+ detection (EEPROM interface)
- HAT++ Instance ID detection
- LED control for status and mode indication
- Instance mode selection via push button
- Board resource management based on the selected Instance ID

Board Resource Management:

- Control of I^{2}C addresses for I^{2}C-based HATs
- Selection of SPI chip select (CS) for SPI-based HATs

- IRQ routing and handling
- Enable I^2C pull-ups in Instance ID I
- HAT-specific control functions

For details on the HAT++ system, see section 3.

2.4 Isolation

Galvanic isolation is implemented between the field-side interface and the Raspberry Pi domain.

The interface and the Raspberry Pi are galvanically isolated. The isolation barrier provides a minimum clearance distance of $\geq 5\,\mathrm{mm}$. In areas where this spacing cannot be maintained, isolation slots are implemented to ensure a creepage distance of at least **5 mm**.

All components bridging the isolation barrier are specifically designed and specified for isolation applications. A detailed list of these components is provided in 3.

The isolation implemented on the board provides **functional galvanic isolation**. The boards are delivered without labeling or certified testing and must therefore be considered as providing **functional insulation only**.

Higher isolation ratings can be achieved by using an alternative BOM, application-specific validation, testing, labeling, and certification. This can be provided upon request (see **??**).

RefDes	Supplier	MPN	Description
U5004, U5005	TI	ISO7741DBQR	Digital isolation
TR5001	PAD	002-T764	Transformer

Table 3: Isolation Components



Figure 5: Isolation Barrier

2.5 Connectors

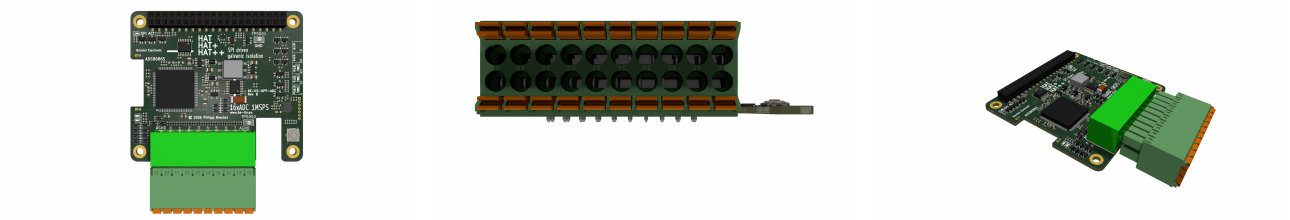
The BE-IIS-HPP-ADC provides 16 analog input channels and four isolated ground connections through J5001, a dual-row, 20-position connector with a 3.50 mm pitch.

The connector is divided into two banks:

- **Bank A (pins 1–10):** Eight analog inputs (AIN0A–AIN7A) on pins 2–9.
- **Bank B (pins 11–20):** Eight analog inputs (AIN0B–AIN7B) on pins 12–19.
- **Isolated Measurement Ground:** Pins 1, 10, 11 and 20 are connected to GND_ISO, the isolated analog measurement ground.

GND_ISO serves as the reference potential for all analog input channels and is galvanically isolated from the Raspberry Pi ground (GND).

The connector arrangement and pin assignments are illustrated below.

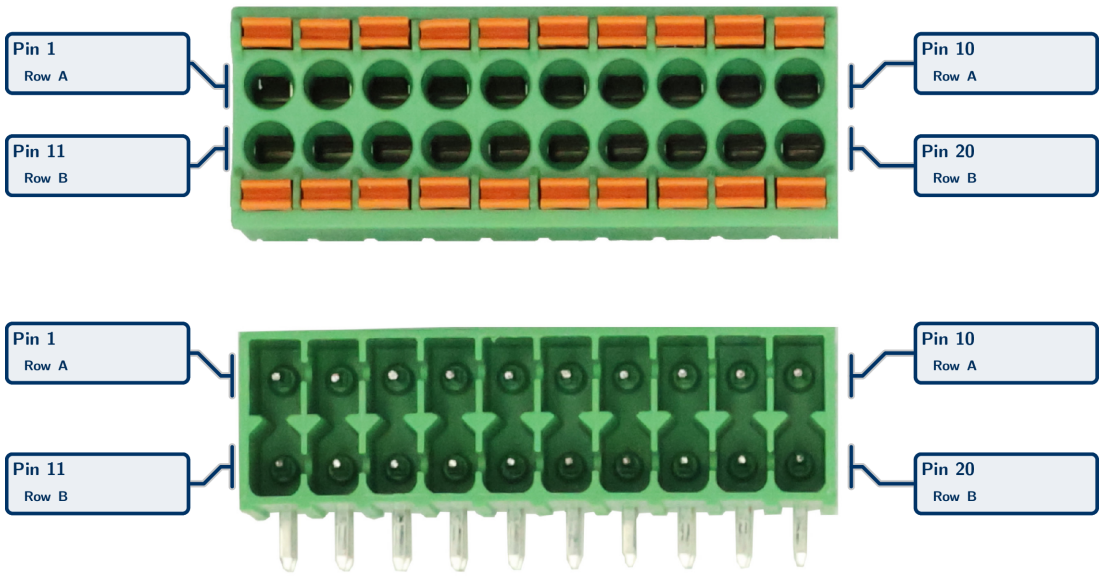


J5001 connector views: top, front and side.

2.5.1 J5001 Pin Assignment

Pin A	Signal A	Pin B	Signal B
1	GND_ISO*	11	GND_ISO*
2	AIN0A	12	AIN0B
3	AIN1A	13	AIN1B
4	AIN2A	14	AIN2B
5	AIN3A	15	AIN3B
6	AIN4A	16	AIN4B
7	AIN5A	17	AIN5B
8	AIN6A	18	AIN6B
9	AIN7A	19	AIN7B
10	GND_ISO*	20	GND_ISO*

* GND_ISO is the isolated analog measurement ground, galvanically isolated from the Raspberry Pi ground.



Connector Orientation

Pins 1, 10, 11 and 20 are marked on both the PCB header and the mating cable connector.

2.5.2 Connector System

J5001 uses the Würth Elektronik WR-TBL connector system.

The PCB is designed for the 20-position THT header **691378110020**.

The corresponding removable cable connector **691379100020** provides screwless push-in terminals for connecting external analog signals.

Component	Manufacturer P/N	Description
PCB Header	691378110020	20-position, dual-row, THT
Cable Connector	691379100020	20-position, dual-row, push-in

Further mechanical and electrical specifications are provided in the manufacturer's datasheets (Refs. 7 and 8).

2.5.3 Wiring Instructions

The removable push-in connector allows external signals to be connected without soldering.

According to the connector manufacturer, the following wiring specifications apply:

- **Connection type:** Screwless push-in terminal
- **Conductor cross-section:** 0.2–1.0 mm^2
- **Wire stripping length:** 9–10 mm
- **Contact pitch:** 3.50 mm

Important: The analog inputs are voltage measurement inputs. The electrical limits specified for the BE-IIS-HPP-ADC must be observed independently of the connector ratings.

GND_ISO must not be confused with the Raspberry Pi ground (GND).

2.6 Indicators (LEDs)

The side LED bar displays the selected Instance ID.

D5021 indicates chip-select activity, while D5001 and D5002 indicate the presence of the isolated supply rails.

3 HAT++ Configuration

HAT++ lets compatible expansion boards share one Raspberry Pi by assigning each board a defined set of hardware resources. Select the configuration with the on-board push button; the matching software uses that selection to configure the board. The board can operate individually or in a stack.

Quick start: one board

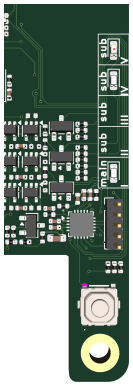
Select **Instance ID I**, leave the selection unchanged for at least **2 seconds**, then power-cycle the system. Use the BE-IIS installer to install the required drivers and overlays, and follow its reboot instructions.

3.1 Select the Board Configuration

An **Instance ID** selects the board's identification data and hardware routing, including the applicable chip-select, interrupt or peripheral address. In the HAT++ specification, the global position I–V is called the **Instance ID**. It describes resource use, not the board's physical position in the stack.

1. With power applied, press the button to cycle through the supported IDs.
2. Read the selected ID from the LED indication.
3. Wait at least **2 seconds** after the last change for it to be saved.
4. Switch the system off and on before using the new configuration.

The selection is retained without power. A button press does not reconfigure the running Linux system; a power cycle is required after a change.
The annotated image below shows this board's supported IDs, button sequence and resource assignments.



ID LEDs and selection button

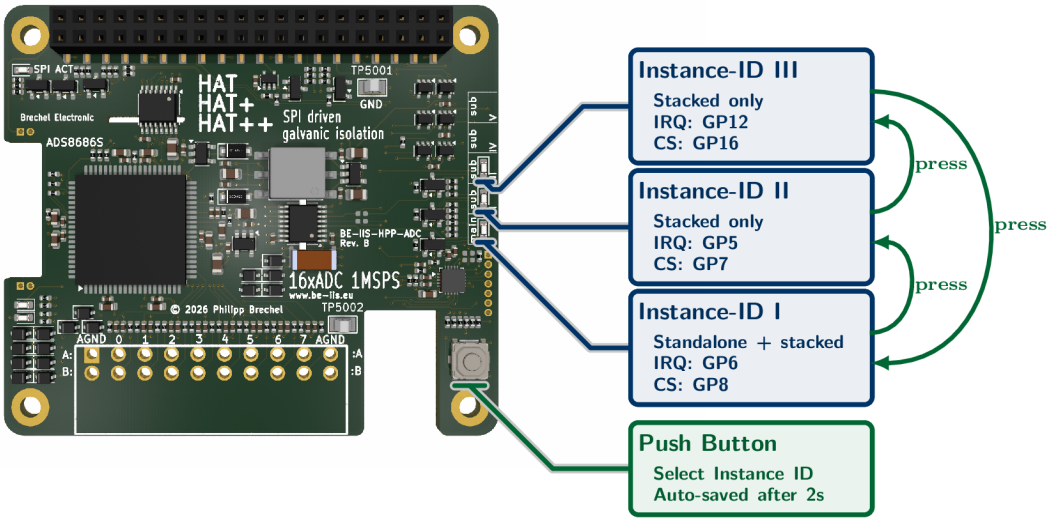


Figure 6: Board configuration: supported IDs, hardware routing and push button.

Overlay names: a suffix I/II/III may identify a board's local variant rather than its global Instance ID. For example, UART variant II uses Instance IV in the BE-IIS reference profile. Use the documented mapping or automatic detection; do not infer the Instance ID from the suffix alone.

3.2 Combine Boards and Check Resources

For automatic HAT++ operation, assign **Instance I to exactly one board**. It provides the identification-bus pull-ups and the primary configuration. Assign a different supported Instance ID to every additional board.

There are five global positions, I–V, and each board supports three of them, including I. A compatible stack can contain up to five boards; a product with three supported positions can be used at most three times. Check each product's supported positions before choosing a combination.

Resources used by this board

The tables below show the resources for each supported configuration. Exclusive signals must have only one owner. Shared signals require compatible electrical behaviour and coordinated use. GP/GPIO values are BCM GPIO numbers, not physical header-pin numbers.

Instance Mode	Signal	Pinname
I	CSN0	GP8
I	BUSY0	GP6
II	CSN1	GP7
II	BUSY1	GP5
III	CSN2	GP16
III	BUSY2	GP12

Table 4: Exclusive HW resources

Instance Mode	Signal	Pin
I–III	SCLK	GP11
I–III	MISO	GP9
I–III	MOSI	GP10
All	RESET	GP13
All	CONVST	GP23
All	SCL0	GP1
All	SDA0	GP0

Table 5: Shared HW resources

Unique Instance IDs coordinate the declared resource assignments; they do not replace checking additional GPIO use, supply requirements or mechanical clearance. Consult the other boards' datasheets when planning a stack.

3.3 Install the Matching Software

Automatic setup. Use the BE-IIS installer and the installation instructions in this datasheet. It deploys the supported overlays and driver setup. Raspberry Pi firmware handles Instance I; the BE-IIS Linux service applies the secondary-instance overlays. After boot, check that the required interfaces are present before starting the application.

Manual integration. Use the resource tables to select the matching Device Tree overlay and install any required kernel drivers. In Instance I, standard Raspberry Pi EEPROM-based detection can load `BE-IIS-HPP-ADC-I` when the overlay is installed. Secondary instances require HAT++ host integration or explicit manual configuration. When changing from automatic to manual setup, disable the automatic loader and remove duplicate overlay configuration.

HAT++ extends the Raspberry Pi HAT+ concept. Instance I is the HAT+-compatible configuration; secondary-instance discovery is provided by HAT++ host software.

Reference: HAT++ Specification

HAT++ Specification – PDF Version 1.0, Rev. A, 30 September 2026 (technical review draft). See Sections 2–4 for Instance IDs, selection and resources, and Sections 7–8 for boot integration and diagnostics. The specification defines the common architecture; this datasheet defines this board's assignments and controls. Specification page and publication status.

4 Software and System Configuration

The BE-IIS-HAT++ system provides a unified platform for fast system integration.

- Predefined drivers and kernel modules
- Support for prebuilt modules and custom kernel builds
- Ready-to-use build and configuration scripts
- Centralized software repository
- Typical setup time below a few minutes

After installation, the system can be used without further software modification.

4.1 System Support

The hardware is designed for use with Raspberry Pi platforms running a standard Raspberry Pi OS. All listed configurations have been validated or are expected to operate reliably with mainline Linux drivers.

- Supported Raspberry Pi platforms:
  - Raspberry Pi 2
  - Raspberry Pi 3B, 3B+, 3A+
  - Raspberry Pi 4B
  - Raspberry Pi 5
  - Raspberry Pi Zero, Zero W, Zero 2 W
- Supported Linux kernel versions:
  - $\geq$ **6.12**
  - BE-IIS aims to support Linux kernel 6.12 and newer across its product portfolio, provided that the required manufacturer drivers and kernel interfaces are available or can be adapted.
  - Older kernel versions may work but are not officially supported.
- Supported operating systems:
  - Raspberry Pi OS (32-bit and 64-bit)
  - Raspberry Pi OS Full and Lite
- Other Linux distributions:
  - Debian, Ubuntu, and other Linux distributions may work
  - Not tested or officially supported at this time

4.2 Driver & Integration

A standard Raspberry Pi OS installation is used as the base system. The provided installer configures all required components automatically, including kernel modules, Device Tree overlays, and systemd services.

- Prepare a Raspberry Pi hardware platform and operating system from the System Support list
- Run the provided Git-based installer
- Reboot the system to apply all configurations
- System supports the full BE-IIS-HAT++ portfolio

```
# install git
$ sudo apt install -y git

# clone installer
$ git clone https://github.com/be-iis/be-iis-installer.git

# enter directory
$ cd be-iis-installer

# run installer
$ ./scripts/install/install-all.sh
```

Drop-in CMD

```
sudo apt install -y git && cd ~ && git clone https://github.com/be-iis/be-iis-installer.git && cd be-iis-installer && ./scripts/install/install-all.sh
```

After running the installer, a summary is printed to indicate the installation status and applied system changes.

Example output (shortened):

```
[INFO] Installation complete.
[INFO] Total scripts : 6
[INFO] Successful    : 6
[INFO] Failed        : 0

[INFO] Changes active after reboot:
[INFO]   - systemd service
[INFO]   - udev rules
[INFO]   - module autoload / runtime setup
Press ENTER to reboot now or CTRL+C to cancel...
```

4.3 Hardware–Software Interaction

The Instance ID can be selected during normal operation using the on-board control interface. The selected configuration is stored persistently after approximately 2 seconds.

The new hardware configuration becomes active after a power cycle. Changing the Instance ID does not automatically reconfigure the running Linux system.

- Instance ID selection is possible during runtime
- Configuration is stored persistently after approximately 2 seconds
- A power cycle is required to activate the new configuration
- Exactly one board must be configured as Instance I for automatic HAT++ operation
- Each additional board must use a unique, supported Instance ID

For detailed configuration instructions, see Section **??**.

4.4 System Inspection

The system status and integration process can be inspected using standard Linux tools. All BE-IIS related services provide detailed runtime information via the system journal.

- View system integration logs:

```
# show BE-IIS system integration log
$ journalctl -b | grep BE-IIS
```

Example output (shortened):

```
BE-IIS Instance I   (0-0050): HAT detected -> BE-IIS-HPP-T1S-I
BE-IIS Instance II  (0-0060): HAT detected -> BE-IIS-HPP-CAN-SIC-II
BE-IIS Instance III (0-0070): HAT detected -> BE-IIS-HPP-LAN-III
BE-IIS Instance IV  (0-0074): HAT detected -> BE-IIS-HPP-UART-II
BE-IIS Instance V   (0-0076): HAT detected -> BE-IIS-HPP-MODBUS-III
BE-IIS HAT++ system integration complete.
```

Drop-in CMD

```
journalctl -b | grep BE-IIS
```

4.5 Interface Naming

All interfaces are assigned deterministic and persistent names using udev rules. This ensures stable device identification across reboots and different hardware configurations.

- Network interfaces are named based on function and instance index
- UART interfaces are exposed via symbolic links
- Naming is independent of kernel enumeration order
- udev rules location: `/etc/udev/rules.d/70-beiis-names.rules`

Example:

```
beiis-t1s0   # 10BASE-T1S interface
beiis-t1l0   # 10BASE-T1L interface
beiis-lan0   # Ethernet interface
beiis-can0   # CAN interface
beiis-uart0a # UART and MODBUS channel A
beiis-uart0b # UART and MODBUS channel B
```

```
$ cat /etc/udev/rules.d/70-beiis-names.rules
```

Drop-in CMD

```
cat /etc/udev/rules.d/70-beiis-names.rules
```

5 Delivery

The scope of delivery depends on the ordered assembly option. The kit configuration specifies the fitted connectors, host header and mechanical accessories.

Order Code	BE-IIS-HPP-ADC
Condition	As ordered
Status	Rev. B
Included Items	1 × BE-IIS-HPP-ADC PCBA 1 × 40-pin Raspberry Pi header 1 × 20-pin PCB header 1 × 20-pin push-pull terminal block 4 × spacers 4 × nuts
REACH & RoHS	Subject to supplier declaration

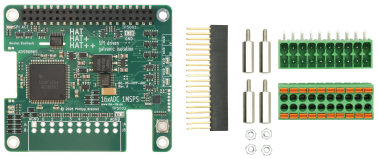


Figure 7: Delivery condition

DigiKey

Coming soon

SELL ON TINDIE

Coming soon

BUY IT ON LECTRONZ

Coming soon

6 Mechanical

6.1 Board Format

- Form factor: Raspberry Pi HAT+
- Mechanical dimensions: see the Rev. A PCB / 3D model
- Mounting-hole pattern: Raspberry Pi HAT interface
- Stack height: defined by the fitted header and spacers

6.2 3D Data

- Available on the BE-IIS product page [4]

7 Assembly

Kit versions require assembly of the connectors and mounting hardware before operation. Follow the assembly sequence below to ensure correct mechanical installation and electrical connection.

7.1 Assemble 2x20-Pin Main Connector

Solder the 40-pin Raspberry Pi GPIO header to the ADC-HAT PCB. Ensure that the connector is correctly aligned and fully seated.

The 2x20-pin connector provides the interface to the Raspberry Pi. For proper HAT functionality, the connector must be assembled carefully.

A stackable 2x20-pin header is included in the delivery and is recommended for most applications, especially when using the BE-IIS HAT++ stacking system.

The header must be inserted between the Raspberry Pi and the HAT: the header is first mounted onto the Raspberry Pi, and the HAT is then plugged onto the header.

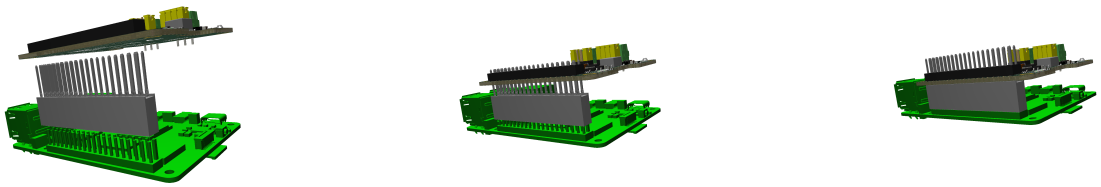


Figure 8: Assembly of the 2x20-pin stackable header between Raspberry Pi and HAT

7.2 Assemble Spacers

Install the supplied spacers and mounting hardware.

To ensure mechanical stability and correct stacking height, spacers must be installed.

- Recommended spacer height: see Section 6.1
- Fix the PCB using appropriate screws and spacers
- Ensure stable mechanical mounting to avoid stress on the connector

The spacers define the stacking distance and provide mechanical fixation of the HAT.

7.3 Assemble 3.5 mm Terminal Block Connector

Solder the specified 20-position terminal block connector to J5001.

Ensure correct alignment and verify that all solder joints are properly formed.

7.4 Connect Analog Input Signals

Connect the analog input signals to the 20-position terminal block according to the pin assignments described in Section 2.5.

Important: Use pins 1, 10, 11 and 20 as the GND reference connections for the analog input signals.

Ensure that all input voltages remain within the specified input voltage range of the ADC-HAT.

7.5 Install ADC-HAT on Raspberry Pi

With the Raspberry Pi powered off, carefully insert the ADC-HAT into the Raspberry Pi 40-pin GPIO connector.

Ensure that the connector is correctly aligned and that all 40 pins are properly seated.

Secure the ADC-HAT using the installed spacers.

7.6 Power On

After verifying all connections, switch on the Raspberry Pi.

The ADC-HAT is now ready for software initialization and operation.

7.7 Board Overview

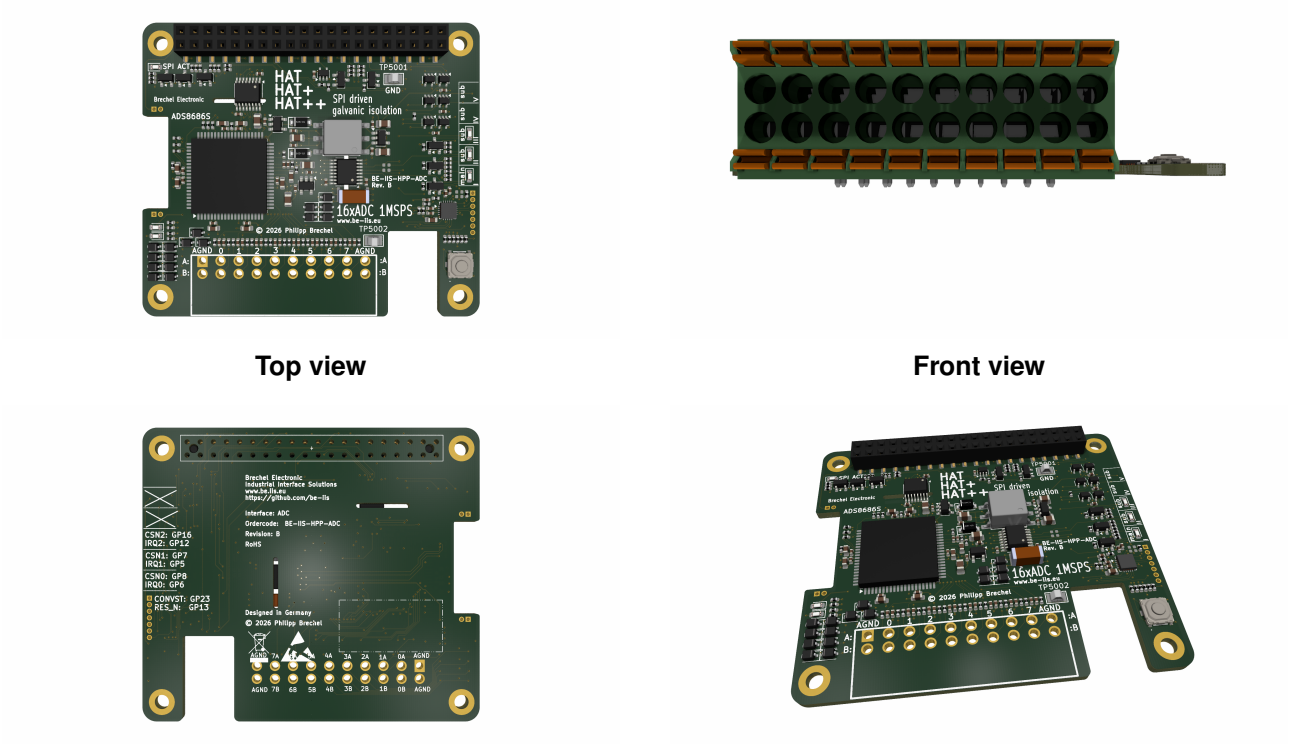


Figure 9: BE-IIS-HPP-ADC – mechanical overview

8 References

1. Texas Instruments ADS8686S datasheet
2. BE-IIS Installer (Software and Setup Tools)
3. Raspberry Pi HAT+ Specification
4. Schematic, PCB-Viewer, BOM, 3D-Model
5. Power over Dataline extension board
6. Texas Instruments ISO7741 datasheet
7. Würth Elektronik 691378110020 – 20-position PCB header datasheet
8. Würth Elektronik 691379100020 – 20-position push-in connector datasheet
9. BE-IIS-ACC-CON20 – Replacement Connector Set

9 Revision History

Revision	Date	Description
A.00	2026-09-30	Initial Rev. A draft
A.01	2026-09-30	Restore T1S template layout and figures
A.02	2026-09-30	Editorial revision; structure unchanged

Company Information

Manufacturer

BE-IIS welcomes technical feedback, suggestions and improvement ideas. Business inquiries, cooperation requests and distribution opportunities are welcome.

Name	Philipp Brechel
Company	Brechel Electronic
Business	Industrial Interface Solutions
Address	Hindenburgstraße 100/1
	73207 Plochingen, Germany

Contact

Email	`contact@be-iis.eu`
Web	`www.be-iis.eu`
GitHub	`github.com/be-iis`
LinkedIn	`linkedin.com/in/brechel`
Digikey	`https://www.digikey.de/de/supplier-centers/industrial-interface-solutions`
Languages	`German and English`