

SmartElex Raspberry Pi High-Precision AD/DA Expansion Board Manual

The SmartElex Raspberry Pi High-Precision AD/DA Expansion Board is a multifunction data acquisition HAT designed for the Raspberry Pi Foundation Raspberry Pi, providing both high-resolution analog-to-digital (ADC) and digital-to-analog (DAC) conversion for precise measurement and signal generation applications. The High-Precision AD/DA Board allows you to add high-precision AD/DA functions to the Raspberry Pi.

Features

- Onboard ADS1256, 8ch 24bit high-precision ADC (4ch differential input), 30ksps sampling rate
- Onboard DAC8532, 2ch 16bit high-precision DAC
- Onboard input interface via pin headers, for connecting analog signal
- the pinout is compatible with different sensor interface standard, easy to connect various analog sensor modules
- Onboard input/output interface via screw terminals, for connecting analog/digital signal
- Features AD/DA detect circuit, easy for signal demonstration

Supported Pi

- Raspberry Pi 4
- Raspberry Pi 5

Pin Function	BCM	WiringPi	Description
DRDY	P17	P0	ADS1256 data output, active-low
RESET	P18	P1	ADS1256 reset input
PDWN	P27	P2	ADS1256 Sync/Shutdown power input, active-low
CS0	P22	P3	ADS1256 chip select, active-low

C1	P23	P4	DAC8532 chip select, active-low
DIN	P10	P12	SPI data input
DOUT	P9	P13	SPI data output
SCK	P11	P14	SPI clock signal

Specifications

Provides both high-precision ADC and DAC functionality on a single board

- Supports multi-channel analog input for sensor interfacing
- Includes high-resolution DAC output for waveform generation
- Communicates via SPI interface with Raspberry Pi
- Compatible with Raspberry Pi 3, 4, and 5
- Integrated low-noise reference and filtering circuitry
- Supports differential and single-ended input modes
- Compact HAT form factor for direct stacking

◆ 1. Overview

A **High-Precision AD/DA Expansion Board** allows the Raspberry Pi to:

- Read **analog signals (ADC – Analog to Digital Conversion)**
- Output **analog voltages (DAC – Digital to Analog Conversion)**

👉 Used in:

- Industrial measurement
 - Sensor interfacing
 - Data acquisition systems
 - Control systems
-

◆ 2. Key Features

Typical specifications (varies by board model):

ADC (Analog Input)

- Resolution: **16-bit / 24-bit**
- Channels: **4 / 8 differential or single-ended**
- Input range: $\pm 5V / \pm 10V$
- High accuracy & low noise

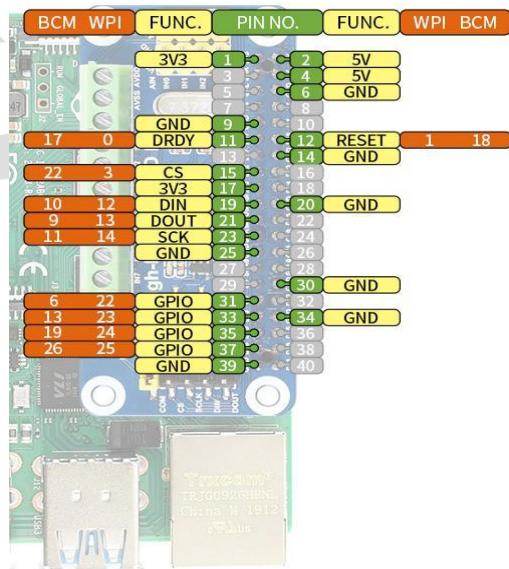
DAC (Analog Output)

- Resolution: **12-bit / 16-bit**
- Channels: 1–4
- Output: 0–3.3V / 0–5V

Interface

- Communication: **SPI / I2C**
- Compatible with:
 - Raspberry Pi 3 / 4 / 5

◆ 3. Hardware Layout



3V3	Power (3.3V power input)
5V	Power (5V power input)
GND	Ground
DRDY	ADS1263 data ready output low active
RESET	ADS1263 reset input
CS	ADS1263 chip select, low active
DIN	SPI data input
DOUT	SPI data output
SCK	SPI clock
GPIO	Extended to sensor interface

Typical Pins

Pin	Function
VCC	3.3V / 5V power
GND	Ground
DIN	SPI MOSI
DOUT	SPI MISO
SCLK	SPI Clock
CS	Chip Select
DRDY	Data Ready (ADC)
AIN0–AIN7	Analog Inputs
DAC_OUT	Analog Output

◆ 4. Hardware Connection

✦ SPI Connection (Most Common)

Board Pin	Raspberry Pi Pin
VCC	3.3V (Pin 1)
GND	GND (Pin 6)
DIN	MOSI (Pin 19)
DOUT	MISO (Pin 21)
SCLK	SCLK (Pin 23)
CS	CEO (Pin 24)

Board Pin	Raspberry Pi Pin
DRDY	GPIO (e.g., Pin 11)

◆ 5. Enable SPI on Raspberry Pi

Run:

```
sudo raspi-config
```

Go to:

Interface Options → SPI → Enable

Reboot:

```
sudo reboot
```

◆ 6. Symbol descriptions

- AD/DA input/output AD0-AD7:
 - Analog input AGND: Analog ground GND: Digital ground VCC: Power supply (3.3V and 5V optional, can be switched by setting the Power selection jumper)
 - DA0-DA1: Analog output
- Analog input AD0-AD7:
 - ADS1256 analog input
 - D0-D3: GPIO of ADS1256 (See ADS1256 datasheet)
 - P22-P25: GPIO of Raspberry Pi
 - AGND: Analog ground
- 3) LDR: Light-dependent resistor, i.e. Photo resistor
 - By connecting the jumper between AD1 and LDR, the MCU can read the voltage output of LDR from the pin AD1.
- LEDA/LEDB: LED output indicator
 - By connecting the jumper between LEDA/LEDB and DAC0/DAC1, the voltage output of DAC0/DAC1 can be estimated roughly by the brightness of LEDA/LEDB.

- ADJ: The adjustable resistor of the 10K potentiometer
 - By connecting the jumper between AD0 and ADJ, the MCU can read the voltage output of the potentiometer from the pin AD0
- 6) PWR LED: Power indicator
- 7) Power selection jumper VCC:
 - Power supply selection VREF: Reference input voltage 3V3: 3.3V output 5V: 5V output
- 8) JMP_AGND: Analog ground jumper
 - For single-ended measurements, use AINCOM (Analog input common) as common input, which can be connected to AGND or external reference voltage. For differential measurements, do not use AINCOM.

◆ 7. Enable SPI on Raspberry Pi

Run:

```
sudo raspi-config
```

Go to:

Interface Options → SPI → Enable

Reboot:

```
sudo reboot
```

◆ 8. Software Setup

Install dependencies

```
sudo apt update
```

```
sudo apt install python3-pip
```

```
pip3 install spidev RPi.GPIO
```

BCM2835

#Open the Raspberry Pi terminal and run the following commands:

```
sudo wget http://www.airspayce.com/mikem/bcm2835/bcm2835-1.71.tar.gz
```

```
sudo tar zxvf bcm2835-1.71.tar.gz
```

```
cd bcm2835-1.71/
```

```
sudo ./configure && sudo make && sudo make check && sudo make install
```

For more information, please refer to the official website:

```
http://www.airspayce.com/mikem/bcm2835/
```

wiringPi

#Open the Raspberry Pi terminal and run the following commands:

```
cd
```

```
sudo apt-get install wiringpi
```

#For Raspberry Pi systems after May 2019 (those earlier may not require execution), an upgrade may be necessary:

```
Wget https://files.waveshare.com/wiki/common/wiringpi-latest.deb
```

```
sudo dpkg -i wiringpi-latest.deb
```

```
gpio -v
```

Run gpio -v and version 2.52 will appear. If it does not appear, there is an installation error

#Bullseye branch system uses the following command:

```
sudo git clone https://github.com/WiringPi/WiringPi
```

```
cd WiringPi
```

```
sudo ./build
```

```
gpio -v
```

Run gpio -v and version 2.70 will appear. If it does not appear, there is an installation error

Lgpio

`sudo su`

`wget https://github.com/joan2937/lg/archive/master.zip`

`unzip master.zip`

`cd lg-master`

`sudo make install`

python

`sudo apt-get update`

`sudo apt-get install ttf-wqy-zenhei`

`sudo apt-get install python-pip`

`sudo pip install RPi.GPIO`

`sudo pip install spidev`

Download Demo

Run in raspberry pi terminal:

`sudo apt-get install p7zip-full`

`wget https://files.waveshare.com/upload/5/5e/High-Precision-AD-DA-Board-Code.7z`

`7z x High-Precision-AD-DA-Board-Code.7z -r -o./High-Precision-AD-DA-Board-Code`

`cd High-Precision-AD-DA-Board-Code/RaspberryPI/`

AD Demo

cd ADS1256

Provides four demos: bcm2835, python2, python3, wiringpi.

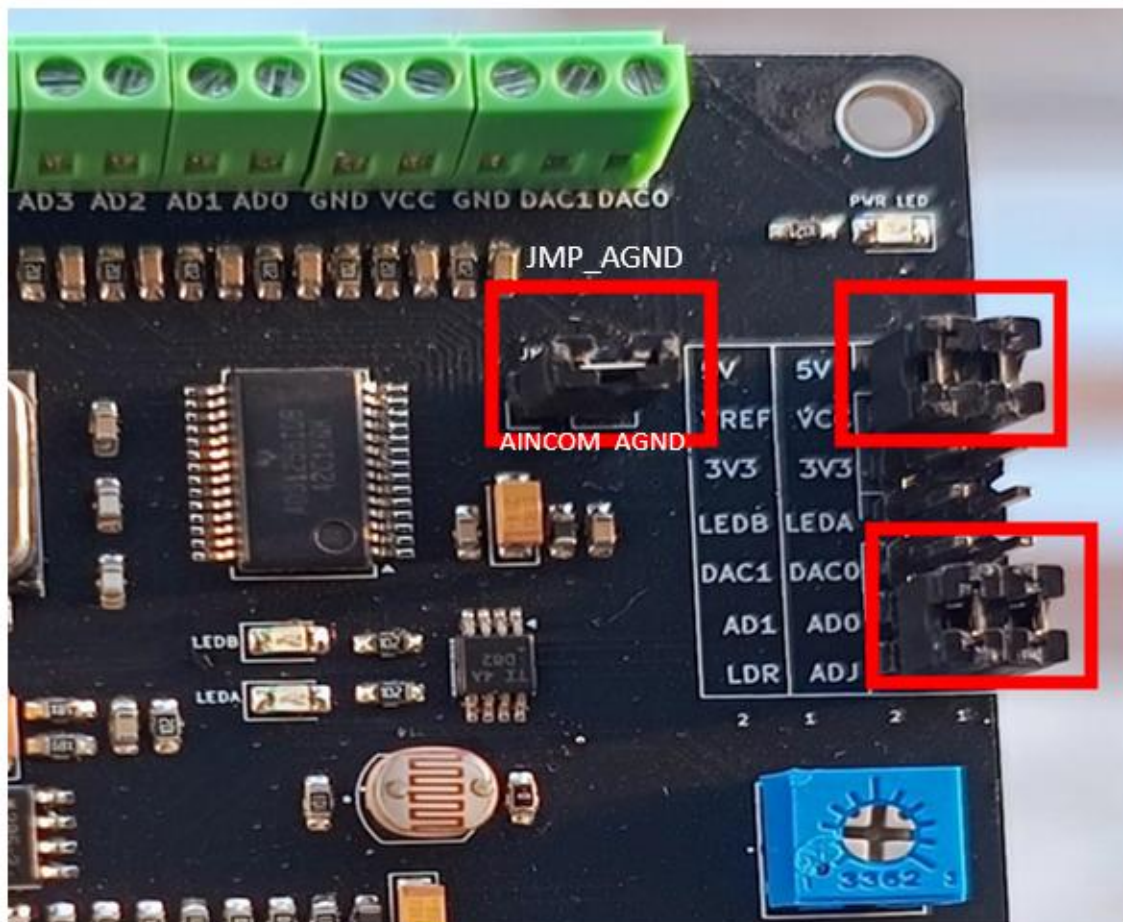
- Hardware Settings

Set the Power Supply to 5V: connect 5V to VCC.

Set the Reference Input Voltage to 5V: connect 5V to VREF.

Set the Potentiometer output as an Analog Input: connect the pin ADJ and AD0. Make sure the left side Sensor Interface AD0 is disconnected.

Set the LDR output as an Analog Input: connect the pin LDR and AD1. Make sure the left-side Sensor Interface AD1 is disconnected. Connect AINCOM to AGND. When using AD for differential measurements, the common input AINCOM does not need to be tied to the ground.



1. bcm2835

```
cd bcm2835/  
make  
sudo ./ads1256_test
```

Notice: If you have run BCM2835 libraries, you need to reboot Raspberry Pi so as to run other libraries successfully.

2. wiringpi

```
cd wiringpi/  
make  
sudo ./ads1256_test
```

3. python2

```
cd python2/  
sudo python main.py
```

4. python3

```
cd python3/  
sudo python3 main.py
```

- Experiment Phenomenon:

Turn the potentiometer, the AD0 channel voltage will change accordingly.

Block the photoresistor, and the AD1 channel voltage changes accordingly.

Since other channels are not connected to the load, the related pins float around 1.17V.

DA Demo

```
cd DAC8532/
```

Provides four demos: bcm2835, python2, python3, wiringpi.

- Hardware setting

Set the Power Supply to 5V: connect 5V to VCC.

Set the Reference Input Voltage to 5V: connect 5V to VREF.

Connect DA0 to LEDA and DA1 to LEDB. Then the brightness of the indicator LEDA will change according to the DA0 output voltage. The brightness of the indicator LEDB will vary according to the DA1 output voltage.

1. bcm2835

```
cd bcm2835/  
make  
sudo ./dac8532_test
```

Notice: If you have run BCM2835 libraries, you need to reboot Raspberry Pi so as to run other libraries successfully.

2. wiringpi

```
cd wiringpi/  
make  
sudo ./dac8532_test
```

3. python2

```
cd python2/  
sudo python main.py
```

4. python3

```
cd python3/  
sudo python3 main.py
```

- Experiment Phenomenon:

Breathing light effect in LEDA and LEDB.

AD/DA synthetical demo

```
cd AD-DA/
```

Provide demo: bcm2835, python.

1. bcm2835

```
cd bcm2835/
```

```
make
```

```
sudo ./main
```

Notice: If you have run BCM2835 libraries, you need to reboot Raspberry Pi so as to run other libraries successfully.

2. python

```
cd python/
```

```
sudo python main.py
```

- Experiment Phenomenon

After rotating the potentiometer, the two LEDs will light up alternately.

◆ 9. Calibration

For high precision:

- Use **known reference voltage**
- Apply correction:

Corrected Value = Raw × Gain + Offset

◆ 10. Applications

- Industrial sensors (pressure, temperature)
- Audio signal processing
- Data logging systems
- Robotics feedback systems
- Lab instruments

◆ 11. Troubleshooting

Issue	Solution
No SPI response	Check SPI enabled
No data	Verify wiring
Noise in signal	Add shielding / filtering
Wrong values	Check reference voltage

◆ 12. Safety Notes

- Do NOT exceed input voltage limits
- Use proper grounding
- Add protection (voltage divider / op-amp buffer)
- Avoid floating inputs

Important Notes & Disclaimer

1. Code Customization

The provided code is for reference/demo purposes only. You may modify it according to your project requirements. We are **not responsible for any issues, malfunctions, or damages** caused by code modifications.

2. Handle with Care

This board contains sensitive electronic components. Please handle it carefully to avoid **physical damage, electrostatic discharge (ESD), or component failure**.

3. External Wiring

Ensure all external connections are done properly. Incorrect wiring or short circuits may damage the board. We are **not responsible for any damage caused due to improper connections or short circuits**.

Notes for Raspberry Pi 4 & Raspberry Pi 5

• Compatibility

This board is compatible with both Raspberry Pi 4 and Raspberry Pi 5 via the standard 40-pin GPIO header.

• SPI Interface Requirement

Ensure that the **SPI interface is enabled** in the Raspberry Pi settings before using this board.

• GPIO Voltage Levels

Both Raspberry Pi 4 and 5 operate at **3.3V logic levels**. Do not connect 5V signals directly to GPIO pins, as this may permanently damage the board.

• Power Supply

Use a **stable and sufficient power supply**:

- Raspberry Pi 4: Recommended 5V / 3A

- Raspberry Pi 5: Recommended 5V / 5A

Insufficient power may cause unstable ADC/DAC performance.

- **Pin Conflicts**

The board uses SPI and specific GPIO pins. Avoid using the same pins for other peripherals simultaneously.

- **Heat Management (Pi 5)**

Raspberry Pi 5 may generate more heat under load. Use proper cooling (heatsink/fan) for stable operation.

- **OS & Driver Support**

Use the latest Raspberry Pi OS. Some libraries or demo codes may require updates for Raspberry Pi 5 compatibility.

SmartElex