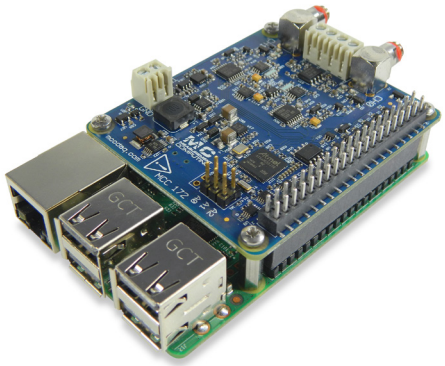
MCC 172

IEPE Measurement DAQ HAT for Raspberry Pi®



The MCC 172 is a 24-bit DAQ HAT for making sound and vibration measurements from IEPE sensors. The MCC 172 is shown connected to a Raspberry Pi (not included).

Overview

The MCC 172 is a voltage HAT (Hardware Attached on Top) board designed for use with Raspberry Pi, the most popular single-board computer on the market today.

A HAT is an add-on board with a 40W GPIO (general purpose input/output) connector that conforms to the Raspberry Pi HAT specification.

The MCC 172 HAT provides two analog inputs for sound or vibration measurements. Up to eight MCC HATs can be stacked onto one Raspberry Pi.

Raspberry Pi Interface

The MCC 172 header plugs into the 40-pin general purpose I/O (GPIO) connector on a user-supplied Raspberry Pi. The MCC 172 was tested for use with all Raspberry Pi models with the 40-pin GPIO connector.

HAT Configuration

HAT configuration parameters are stored in an on-board EEPROM that allows the Raspberry Pi to automatically set up the GPIO pins when the HAT is connected.

Stackable HATs

Up to eight MCC HAT boards can be stacked onto a single Raspberry Pi.

Users can mix and match MCC HAT models in the stack.

Analog Input

The two 24-bit differential analog input channels simultaneously acquire data at rates up 51.2 kS/s. Users can turn IEPE excitation current on or off.

Each channel has a dedicated A/D converter. Both ADCs share the same clock and are synchronized to start conversions at the same time for synchronous data.

Multiple HAT Synchronization

Multiple MCC 172 HATs can be synchronized to a single sampling clock. The clock is programmable for sampling rates between 51.2 kS/s to 200 S/s.

Sample Rates

- Single-board: max throughput is 102.4 kS/s (51.2 kS × 2 channels)
- Stacked boards: max throughput is 307.2 kS/s aggregate[1].

Digital Trigger

The trigger input (terminal TRIG) is used to delay an input scan until a specified condition is met at the trigger input.

The trigger input signal may be a 3.3V or 5V TTL or CMOS logic signal. The input condition may be edge or level sensitive, rising or falling edge, or high or low level. This terminal may be used to trigger the start of an acquisition on multiple synchronized MCC 172 HATs.

Power

The MCC 172 is powered with 5 V provided by the Raspberry Pi through the GPIO header connector.

1 Dependent on the load on the Raspberry Pi and the SPI interface.

Features

- **Two IEPE inputs**
  - **Two 24-bit, 51.2 kS/s A/D converters (one per channel)**
  - **AC coupled at ±5 V**
  - **10-32 and screw terminal connections for OEM support**
- **Synchronous ADC conversions between multiple boards**
- **Onboard sample buffers allow high-speed acquisition**
- **External digital trigger input**
- **Stack up to eight MCC HATs onto a single Raspberry Pi**

Software

- **MCC DAQ HAT Library; available on GitHub**

Supported Operating Systems

- **Linux®/Raspbian**

Programming API

- **C, C++, Python**

OEM Support

Users can connect analog input signals to either the 10-32 coaxial inputs or to the screw terminals. Only one source may be connected to a channel at a time.

MCC DAQ HAT Library

The open-source MCC DAQ HAT Library of commands in C/C++ and Python allows users to develop applications on the Raspberry Pi using Linux.

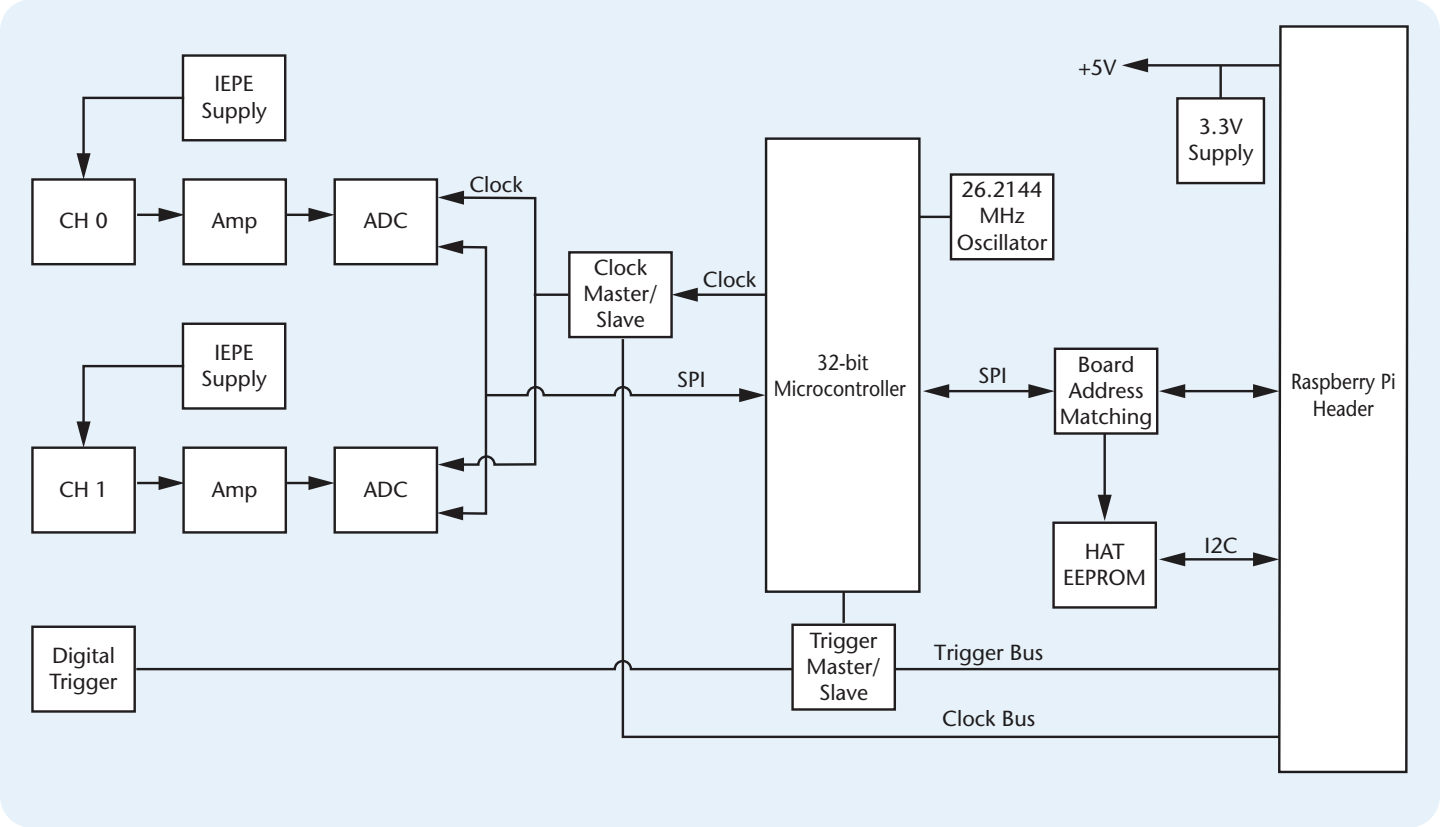
The library is available to download from GitHub. Comprehensive API and hardware documentation is available.

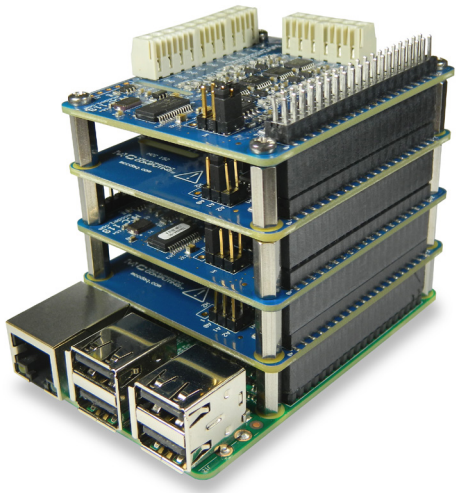
The MCC DAQ HAT Library supports operation with multiple MCC DAQ HATs running concurrently.

Console-based and user interface (UI) example programs are available for each API.

MCC 172

Block Diagram





Stackable

Connect up to eight MCC DAQ HATs onto a single Raspberry Pi.

Onboard jumpers identify each board in the stack.

MCC 172

Example Programs

MCC DAQ HAT Examples

The MCC DAQ HAT Library includes example programs developed in C/C++ and Python that users can run to become familiar with the DAQ HAT library and boards; source code is included.

Console-Based (C/C++ and Python)

Console-based examples are provided that demonstrate how to perform FFT on a block of data, acquire synchronous data from multiple MCC 172 HATs using shared clock and trigger options, trigger a finite scan, and synchronously acquire data from multiple DAQ HATs. Source code is included.

The fft_scan example is shown here.

```
File  Edit  Tabs  Help
pi@janet_pi:~/daqhats_dev/examples/c/mcc172/fft_scan $ ./fft_scan

Selected MCC 172 device at address 0
Enable IEPE power [y or n]?  y

MCC 172 FFT example
    Functions demonstrated:
        mcc172_iepe_config_write
        mcc172_a_in_clock_config_write
        mcc172_a_in_clock_config_read
        mcc172_a_in_scan_start
        mcc172_a_in_scan_read
        mcc172_a_in_scan_stop
        mcc172_a_in_scan_cleanup
    IEPE power: on
    Channel 0
    Samples per channel: 12800
    Requested scan rate: 51200.00
    Actual scan rate: 51200.00
    Options: OPTS_DEFAULT

Press ENTER to continue

Scanning input...

Peak: -4.2 dBFS at 188.5 Hz
2nd harmonic: -52.1 dBFS at 376.9 Hz
3rd harmonic: -14.3 dBFS at 565.4 Hz
4th harmonic: -52.8 dBFS at 753.8 Hz
5th harmonic: -19.4 dBFS at 942.3 Hz
6th harmonic: -52.6 dBFS at 1130.7 Hz
7th harmonic: -21.5 dBFS at 1319.2 Hz
Data and FFT saved in fft_scan.csv.
pi@janet_pi:~/daqhats_dev/examples/c/mcc172/fft_scan $
```

Peak frequency and harmonics display in a terminal window

	A	B	C	D
1	Time data (V)	Frequency (Hz	Spectrum (dBFS)	
2	-2.420162	0	-67.119626	
3	-2.419815	4	-66.998816	
4	-2.419353	8	-96.851257	
5	-2.418839	12	-108.361359	
6	-2.418419	16	-103.65905	
7	-2.418002	20	-101.043952	
8	-2.417593	24	-98.230112	
9	-2.417056	28	-103.009634	
10	-2.416751	32	-106.578171	
11	-2.416245	36	-114.967032	

fft_scan

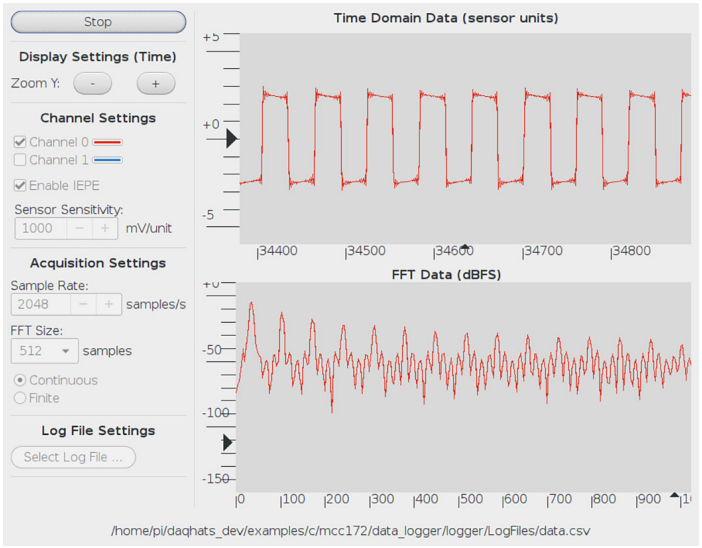
FFT data is saved to a csv file

User Interface

Example programs featuring a user interface are provided in different formats. Examples of each are shown here.

Data Logger (C/C++)

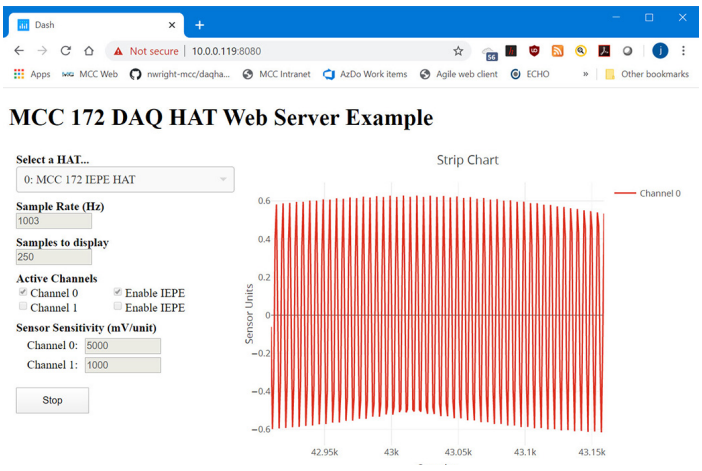
The data logger example shows how to acquire data from the MCC 172, display the data on a strip chart, and log the data to a CSV file. This example can be run from the terminal.



Configure options, plot data on a strip chart, and log data to a file

Web Server (Python)

The web server example lets users configure acquisition options and view acquired data from a browser window. This example is written for Python (source included).



Configure options and view strip chart data from your browser

MCC 172

Specifications

All specifications are subject to change without notice.
Typical for 25 ˚C unless otherwise specified.

Analog input

Number of channels: 2
ADC Resolution: 24 bits
A/D converter type: Delta sigma
Sampling mode: Simultaneous
Master timebase (f_M):
Frequency: 26.2144 MHz
Accuracy: ±50 ppm max
Master timebase sources
Internal clock
Shared clock from another MCC 172
Data rates (fS)
(f_M / 512) / n, n = 1, 2, …, 256
51.2 kS/s max
200 S/s min
Input coupling: AC
AC cutoff frequency
-3 dB: 0.78 Hz
-0.1 dB: 5.2 Hz max
Input voltage range: ±5 V
Common-mode voltage range
CHx to AGND: ±2 V max
Overvoltage protection
CHx+ to CHx-: ±35 V
CHx- to ground: ±2.5 V
IEPE compliance voltage: 23 V max
IEPE excitation current: 4.0 mA min, 4.1 mA typ
Input delay
1 kHz to 23 kHz input frequency: 4.5 µs + 39 / f_S
Channel-to-channel matching
Phase (200 Hz to 23 kHz): (f_{in} * 0.022°) max
Gain (20 Hz to 23 kHz): 0.19 dB typ
Passband
Frequency: 0.453 * f_S
Flatness (20 Hz to 23 kHz): 52 mdB (pk-to-pk max)
Phase nonlinearity
f_S = 51.2 kS/s, 200 Hz to 23 kHz input frequency: ±0.36° max
Stopband
Frequency: 0.547 * f_S
Rejection: 99 dB min
Alias-free bandwidth: 0.453 * fS
Alias rejection: 100 dB @ 51.2 kS/s
Oversample rate: 128 * f_S
Crosstalk
1 kHz: -122 dB
SFDR
f_{in} = 1 kHz, -60 dBFS: 120 dB
Dynamic range
f_{in} = 1 kHz, -1 dBFS: 100 dB
Input impedance
Differential: 202 kΩ
AI- (shield) to ground: 50 Ω
Throughput
Single board: 102.4 kS/s max (51.2 kS/s × 2 channels)
Multiple boards: Up to 307.2 kS/s aggregate†
† Depends on the load on the Raspberry Pi processor and the SPI interface.

Note: For best results, connect the signal source and the Raspberry Pi to a common ground. If a floating source is required, connect the MCC 172 to earth ground via the DGND screw terminal to minimize common mode noise.

Accuracy

Analog input AC voltage measurement accuracy (all values are (±) and apply to calibrated readings)			
Gain error, max:	**Offset error, max:**	**Gain temperature coefficient, max:**	**Gain temperature coefficient, max:**
0.43%	5.10 mV	88 ppm/°C	184 µV/°C

Noise performance

Idle Channel	51.2 kS/s
Noise	33 µVrms
Noise density	207 nV/√Hz

Total harmonic distortion (THD)

Input Amplitude	1 kHz	8 kHz
-1 dBFS	-93 dB	-91 dB
-10.96 dBFS	-87 dB	-87 dB

External digital trigger

Trigger source: TRIG input
Trigger mode: Software configurable for rising or falling edge, or high or low level
Trigger latency: 1 µs + 1 sample period (1/fS) max
Trigger pulse width: 100 ns min
Input type: Schmitt trigger, 100K pull-down to ground
Input high voltage threshold: 1.48 V min
Input low voltage threshold: 1.2 V max
Input hysteresis: 0.51 V min
Input voltage limits: 6.5 V absolute max, –0.5 V absolute min, 0 V recommended min

Memory

Data FIFO: 48 K (49,152) analog input samples
Non-volatile memory: 4 KB (ID and calibration storage, no user-modifiable memory)

Power

Supply current, 5 V supply
Typical: 100 mA
Maximum: 140 mA

Interface

Raspberry Pi GPIO pins used:
GPIO 8, GPIO 9, GPIO 10, GPIO 11 (SPI interface)
ID_SD, ID_SC (ID EEPROM)
GPIO 12, GPIO 13, GPIO 26 (Board address)
GPIO 5, 6, 19, 16, 20 (Clock / trigger sharing, Reset, IRQ)
Data interface type: SPI slave device, CE0 chip select
SPI mode: 1
SPI clock rate: 18 MHz, max

Environment

Operating temperature: 0 °C to 55 °C
Storage temperature: –40 ˚C to 85 °C max
Relative humidity: 0% to 90% non-condensing

Mechanical

Dimensions (L × W × H): 65 × 56.5 × 12 mm (2.56 × 2.22 × 0.47 in.) max

MCC 172

Ordering

Signal connectors

Connector types: 10-32 coaxial / screw terminal (in parallel; only one source may be connected to a channel at a time)
Coaxial input signals
CH0: channel 0 input
CH1: channel 1 input
Screw terminal wire gauge range: 16 AWG to 30 AWG

Analog input screw terminal pinout (Connector J2)		
Pin	**Signal name**	**Pin description**
1	CH0+	Channel 0 positive input
2	CH0-	Channel 0 negative input
3	CH1+	Channel 1 positive input
4	CH1-	Channel 1 negative input

Trigger input screw terminal pinout (Connector J5)		
Pin	**Signal name**	**Pin description**
1	TRIG	Digital trigger input
2	GND	Digital ground

Order Information

Hardware

Part No.	Description
MCC 172	24-bit, 2-channel IEPE measurement DAQ HAT. Raspberry Pi with the 40-pin GPIO connector required.

Accessories

Part No.	Description
ACC-172	Coaxial cable, 3 ft, with 10-32 plug to BNC jack (2 qty)

Software

Part No.	Description
MCC DAQ HAT Library	Open-source library for developing applications in C, C++, and Python on Linux for MCC DAQ HAT hardware. Available for download on GitHub at https://github.com/mccdaq/daqhats.

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Modules Accessories ***category:***

Click to view products by Digilent ***manufacturer:***

Other Similar products are found below :

7010-0001 AX98219 A1UL8RISER F1UJPMICRISER FHW1U16RISER 20-101-0440 MBCDROM AX61221TM VM-105 EA CARREDIPTFT02 RK-210E-B E226171106 88606200030E 8816K6400A0E SI-HDMI-EDID-EM MIC-75M13-00A1E FPM-1000T-SMKE AMK-R004E 96FMCF-ST2ADAPTER1 AHWKPTP12GBGB AXXSTCPUCAR FPK-07-R10 Mini Din 6P to 6P HARNESS 881261510A0E AXXP3SWX08080 conga-B7XD/CSP-Cu-B 881281021A0E HFT for mounting KIT FN928X_FN929X 15100600 9-5000-1116 BKCMCR1ABB 70763 98R3612003E 881261910A0E 106897 48222R 4D ARDUINO ADAPTOR SHIELD II 20926110901 PYCASE GREEN PYCASE BLUE FP15072_ZORYA-SC-HEKLA 20952000004 20953000007 DP-DVI-R10 575-BBIS RACK-220GW/A130B 492-BBKM IP411 70760 PICONYMPHGL