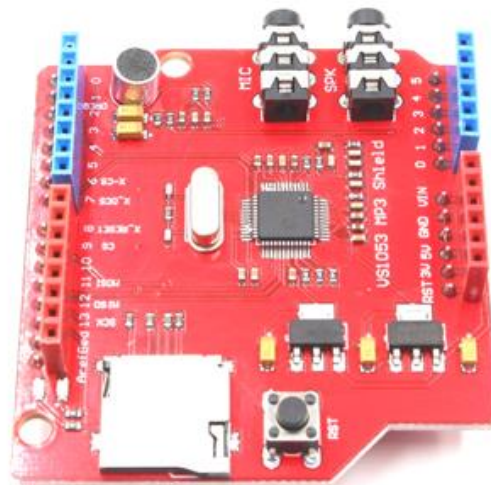


VS1053B MP3 Audio Shield for Arduino

VS1053B Arduino Shield is a versatile MP3 codec module that is capable of decoding a variety of music formats, including OggVorbis, MP3, AAC, WMA, FLAC, WAV (PCM and ADPCM) and MIDI audio. For the best headphone listening experience, the VS1053 includes Ear Speaker spatial processing which accurately simulates how a room with stereo loudspeakers would sound. With the TF card slot on the other side of the board, you can play mp3 files from TF card. Support Ogg Vorbis recording open file format that allows for very high sound quality with low to medium bitrates.

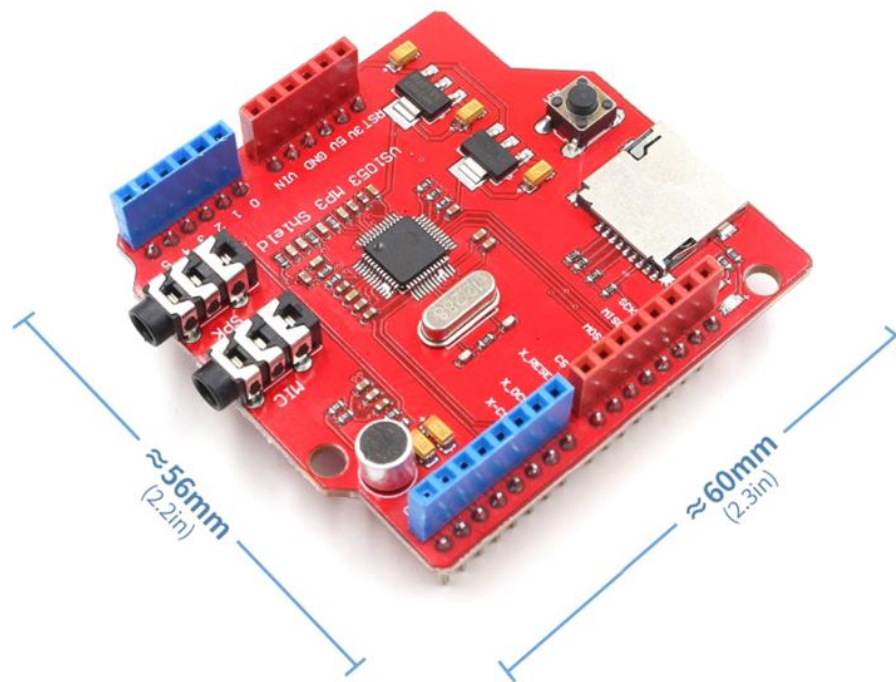


SKU: MDU1031

Brief Data:

- Direct plug-in to Arduino Uno or Mega board.
- Real-time recording.
- SPI interface.
- A headphone and stereo output.
- A microphone for recording.
- Line-in input interface.
- Power indicator.
- 3.3V and 2.8V of LDO chip AMS-1117 on board, provides up to 800mA current.
- A single power supply: +5 VDC.
- SD card slot.

Mechanical Dimension:

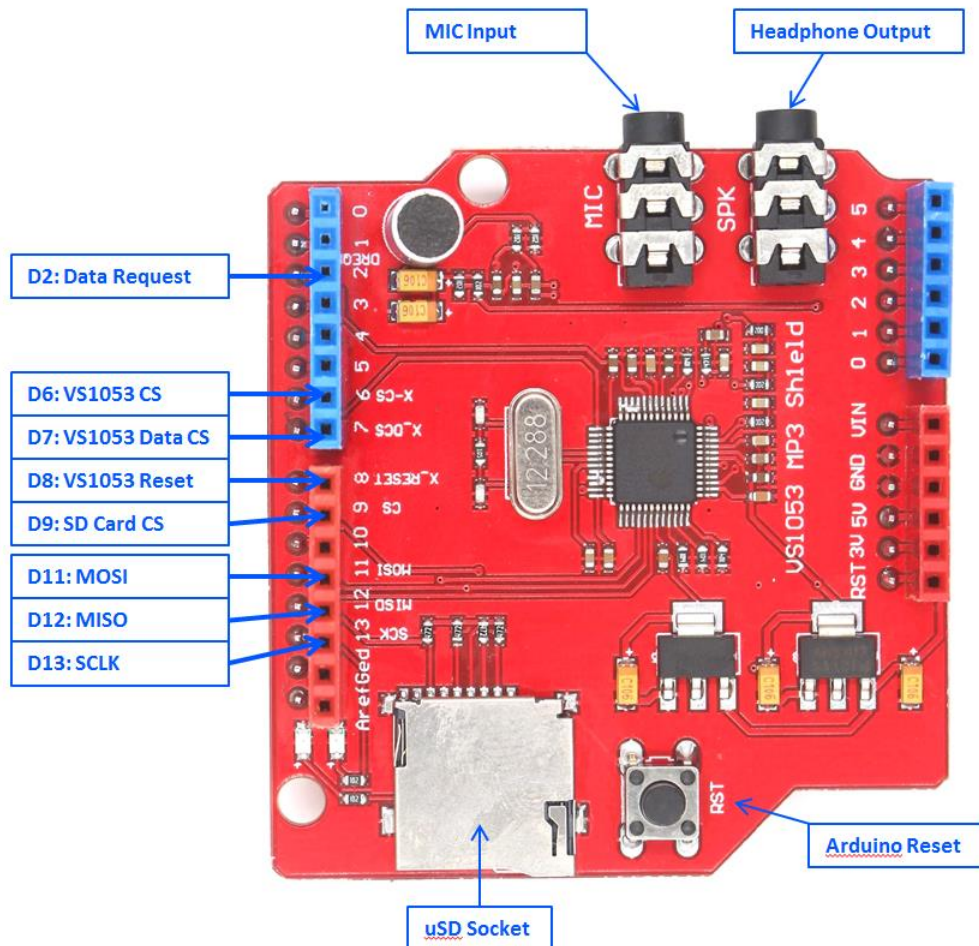


Hardware Overview:

The centerpiece of the MP3 Player Shield is a VS1053B Audio Codec IC. The VS1053B is a multitasking little chip. On top of MP3's, it can also decode Ogg Vorbis, AAC, WMA, and MIDI. (It's also capable of encoding audio, although that's outside the scope of the MP3 Shield.)

Supporting the VS1053 is a μ SD card socket, which you can use to store MP3 files on. Using the Arduino SD library, it's simple to read music files off an SD card, and stream them to the VS1053B. There's additional circuitry on-board to level shift signals down to the 3.3V maximum allowable by SD cards.

Here's a quick visual overview of the important connectors and other components on the MP3 Player Shield:



The MP3 Player Shield requires exclusive use of a handful of pins. These pins can't be used to interface with other devices:

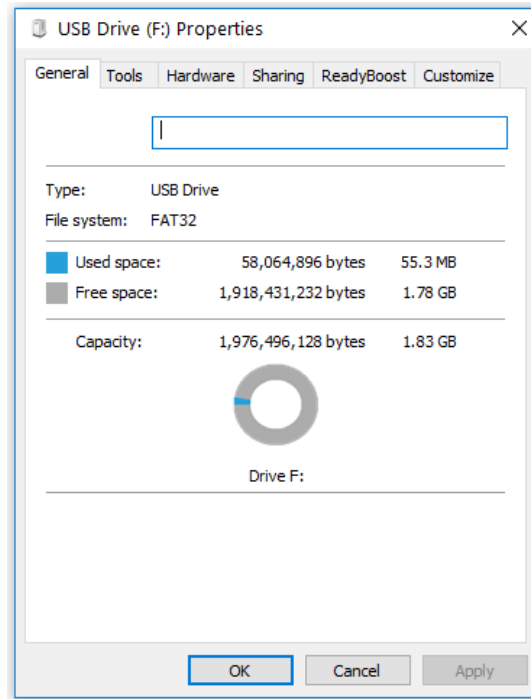
- D2 is connected to the data request output of the VS1053B. This pin is an interrupt, which tells the Arduino that the IC needs more music data.
- D6 is connected to the chip select input of the VS1053B. This active-low pin tells the chip when data is being sent to it.
- D7 is connected to the data chip select input of the VS1053B, which tells the chip when music data is being sent.
- D8 is connected to the reset input of the VS1053B.
- D9 is connected to the chip select input of the μ SD card.

The Arduino's three SPI data and clock pins -- D11, D12, and D13 -- can be used to interface to other SPI components. They can't, however, be used for any purpose other than SPI.

MP3 File and μ SD Card Setup:

You may also want to start preparing your μ SD card. First, make sure it's formatted correctly. The SDFat Arduino library we'll be using only supports cards formatted for FAT16 or FAT32. Your card is probably already formatted to one of these standards, but it doesn't hurt to double check.

SD card formatted to FAT32:



You may also want to prepare your music files. The VS1053B is capable of playing MP3, AAC, WMA, MIDI, and Ogg Vorbis audio files. The VS1053B supports a variety of sample rates and bitrates for each file type. Check out the datasheet (beginning in section 8 -- page 26), to make sure your audio files are supported. MP3's, for example, are supported at up to a 320 kbps bitrate and a 48 kHz sample rate.

Finally, before loading the audio files onto your SD card, you'll need to modify their names. The SDFat library only supports "8.3" file names - that's eight characters before the '.' and three characters after (e.g. "track001.mp3"). Further, some of the example code we'll be using later on requires that the audio files be named using special conventions. In the MP3 trigger example the files will need to be named "TRACK001.mp3", "TRACK002.mp3", etc.

Using the SFEMP3Shield Library:

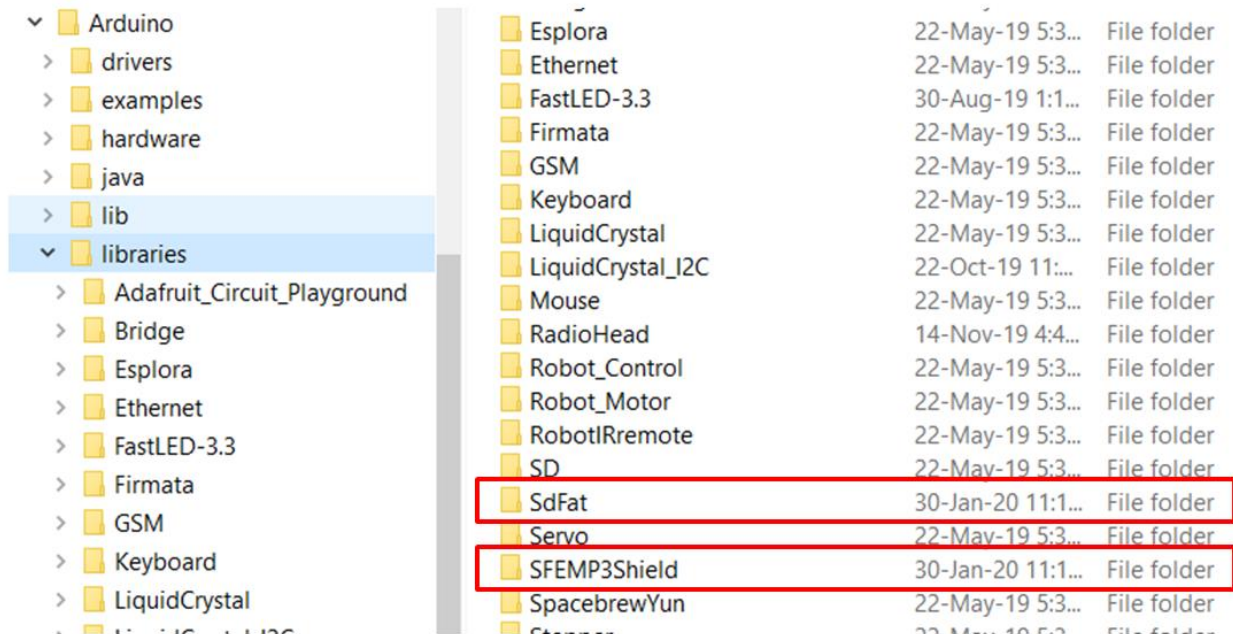
The [SFEMP3Shield Arduino library](#) - written collaboratively by Bill Porter, Michael Flaga, ddz, and Wade Brainerd, is an AMAZING resource for the MP3 Player Shield. Combined with the equally awesome SdFat library, SFEMP3Shield greatly simplifies the task of interfacing with the VS1053 and using the MP3 Player Shield.

We recommend using the SFEMP3Shield library with this shield. On this page we'll go over how to install and use the library. On the next page, we'll make a fun example sketch using it.

Install the SFEMP3Shield Library:

To download the latest version of the SFEMP3Shield, click the "Download ZIP" link on the library's [GitHub](#) page. This GitHub repo already includes a copy of the SdFat library, so you're covered there.

The "Sparkfun-MP3-Player-Shield-Arduino-Library-master" folder you extract should have a handful of folders within. The "**SFEMP3Shield**" and "**SdFat**" folders in particular need to be installed as Arduino libraries. You'll need to place those two folders within your Arduino sketch book (by default in your home/Arduino folder). Your file structure should look something like this once installed:



Now restart Arduino (if it was open), and check under the "Sketch" > "Import Library" menu to make sure "SFEMP3Library" and "SdFat" are both listed there.

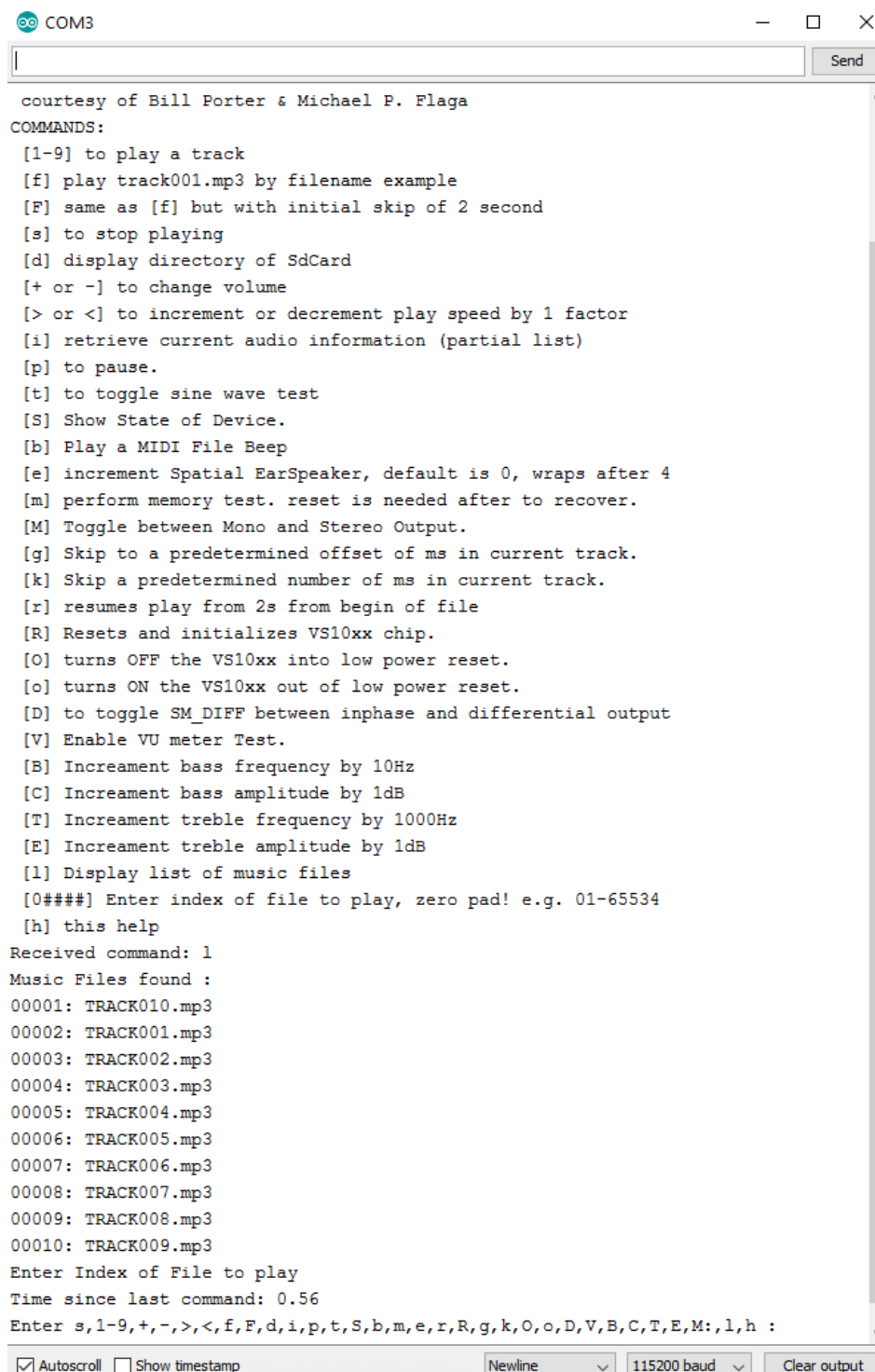
Upload an Example Sketch:

The SFEMP3Shield library includes a few fun example sketches that show off all of its awesome abilities. To begin, try loading up the "FilePlayer" example, by going to:

"File" > "Examples" > "SFEMP3Shield" > "Examples" > "FilePlayer".

Make sure the MP3 Player Shield is sitting comfortably on top of your Arduino, and upload away!

Once uploaded, open up the Serial Monitor and set the baud rate to 115200 bps. Once the sketch initializes, it should present you with a navigable menu:

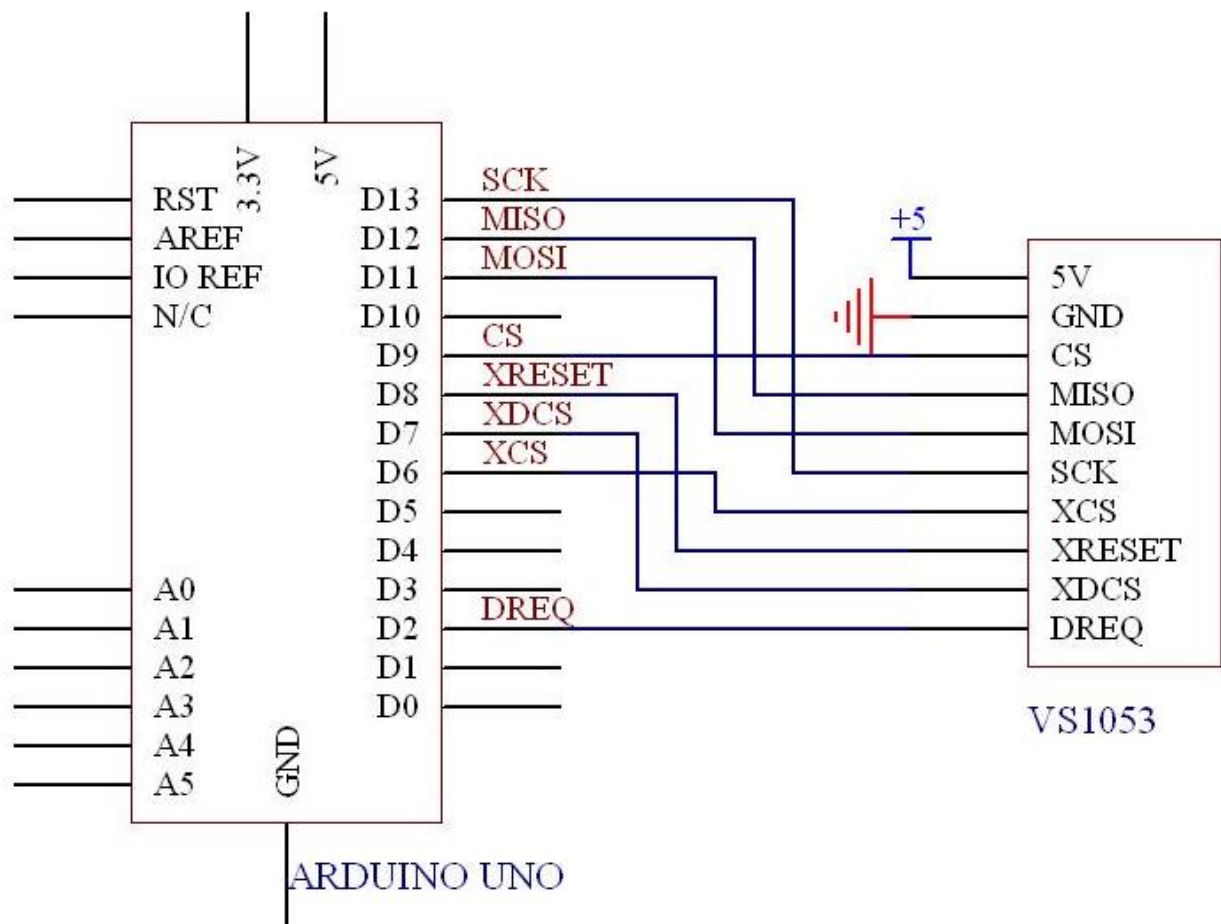


```
COM3
[1-9] to play a track
[f] play track001.mp3 by filename example
[F] same as [f] but with initial skip of 2 second
[s] to stop playing
[d] display directory of SdCard
[+ or -] to change volume
[> or <] to increment or decrement play speed by 1 factor
[i] retrieve current audio information (partial list)
[p] to pause.
[t] to toggle sine wave test
[S] Show State of Device.
[b] Play a MIDI File Beep
[e] increment Spatial EarSpeaker, default is 0, wraps after 4
[m] perform memory test. reset is needed after to recover.
[M] Toggle between Mono and Stereo Output.
[g] Skip to a predetermined offset of ms in current track.
[k] Skip a predetermined number of ms in current track.
[r] resumes play from 2s from begin of file
[R] Resets and initializes VS10xx chip.
[O] turns OFF the VS10xx into low power reset.
[o] turns ON the VS10xx out of low power reset.
[D] to toggle SM_DIFF between inphase and differential output
[V] Enable VU meter Test.
[B] Increment bass frequency by 10Hz
[C] Increment bass amplitude by 1dB
[T] Increment treble frequency by 1000Hz
[E] Increment treble amplitude by 1dB
[l] Display list of music files
[0####] Enter index of file to play, zero pad! e.g. 01-65534
[h] this help
Received command: 1
Music Files found :
00001: TRACK010.mp3
00002: TRACK001.mp3
00003: TRACK002.mp3
00004: TRACK003.mp3
00005: TRACK004.mp3
00006: TRACK005.mp3
00007: TRACK006.mp3
00008: TRACK007.mp3
00009: TRACK008.mp3
00010: TRACK009.mp3
Enter Index of File to play
Time since last command: 0.56
Enter s,1-9,+,-,>,<,f,F,d,i,p,t,S,b,m,e,r,R,g,k,O,o,D,V,B,C,T,E,M:,l,h :
```

Hopefully, towards the bottom of the menu, you'll see a list of MP3 files that the sketch found on your SD card. Try sending a 3-digit number with padding zeros (e.g. 001, 003, etc.) to make one of the listed files start playing.

If it's too quiet, try turning up the volume using the '+' command, or go down with '-'. There are all sorts of other fun options to try out too.

Schematic Diagram:



Web Resource:

- [TDA7498 2X100W Class-D Audio Amplifier](#)
- [GPD2856C-MP3 Player Board](#)
- [Class-D Audio Amplifier Basic](#)



Handsontec.com

We have the parts for your ideas

HandsOn Technology provides a multimedia and interactive platform for everyone interested in electronics. From beginner to diehard, from student to lecturer. Information, education, inspiration and entertainment. Analog and digital, practical and theoretical; software and hardware.



open source
hardware

HandsOn Technology support Open Source Hardware (OSHW) Development Platform.

Learn : Design : Share

handsontec.com



The Face behind our product quality...

In a world of constant change and continuous technological development, a new or replacement product is never far away – and they all need to be tested.

Many vendors simply import and sell without checks and this cannot be the ultimate interests of anyone, particularly the customer. Every part sold on Handsontec is fully tested. So when buying from Handsontec products range, you can be confident you're getting outstanding quality and value.

We keep adding the new parts so that you can get rolling on your next project.



www.handsontec.com

[Breakout Boards & Modules](#)



[Connectors](#)



[Electro-Mechanical Parts](#)



[Engineering Material](#)



[Mechanical Hardware](#)



[Electronics Components](#)

P



www.handsontec.com

[Power Supply](#)



[Arduino Board & Shield](#)

[Tools & Accessory](#)



www.handsontec.com

[Tools & Accessory](#)