

BMduino-Shield
Creative Music Touch

BMV56T123
Arduino Library V1.0.1 Description

Revision: V1.00 Date: August 10, 2023

www.bestmodulescorp.com

Contents

Introduction	3
Arduino Library Functions.....	3
Arduino Lib Download and Installation	5
Arduino Example	6
Example: getTouchValue.....	6

Introduction

The BMV56T123 is a creative music touch shield from Best Modules, which uses the UART communication method to implement the touch music and volume settings. This document provides the description of the BMV56T123 Arduino Lib functions and how to install the Arduino Lib. The example demonstrates the voice source setting and tone setting operations.

Arduino Library Functions

Arduino Lib Name: BMV56T123		Lib Version: V1.0.1
Constructors & Initialisation		
1	BMV56T123(uint16_t intPin=4, uint16_t rxPin=2, uint16_t txPin=3)	
	Description	Constructor
	Parameter	intPin: INT pin, connects to the shield INT pin. Defaults to D4 rxPin: RX pin, connects to the shield mTX pin. Defaults to D2 txPin: TX pin, connects to the shield mRX pin. Defaults to D3
	Return Value	—
	Note	—
2	void begin()	
	Description	Shield initialisation
	Parameter	—
	Return Value	void
	Note	—
Performance Functions		
3	bool setPlayMode(uint8_t mode)	
	Description	Set the touch playback voice source
	Parameter	mode: Voice source selection 0 (INTERNAL_SOUND_SOURCE): Integrated MIDI voice source 1 (EXTERNAL_SOUND_SOURCE): Customised voice source
	Return Value	Execution result: true: Succeeded false: Failed
	Note	—
4	bool setTimbreGroup(uint8_t group)	
	Description	Set the MIDI tone group
	Parameter	group : Tone group selection, ranging from 0~12
	Return Value	Execution result: true: Succeeded false: Failed
	Note	Refer to the Voice Source Introduction section of the User Guide for MIDI tone group
5	bool setVolume(uint8_t volume)	
	Description	Set the volume
	Parameter	volume :Volume level, ranging from 0~15
	Return Value	Execution result: true: Succeeded false: Failed
	Note	There are 15 levels of volume adjustment, 0 is the minimum volume, i.e. mute.

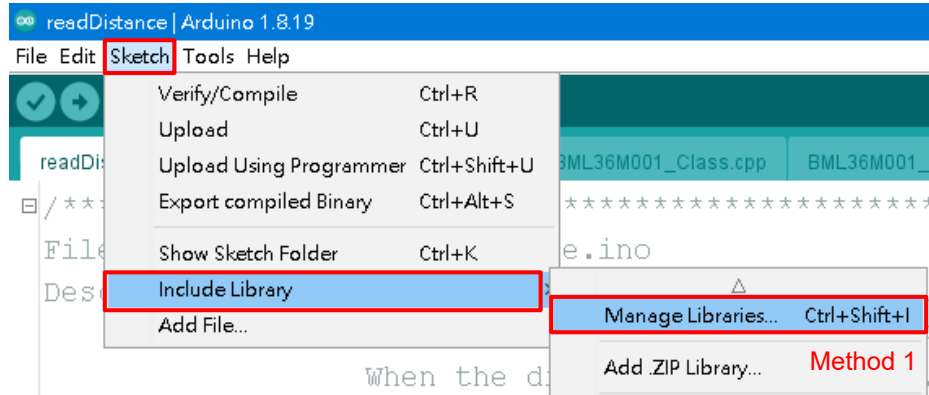
6	bool playVoice(uint8_t voice_number)	
	Description	Play the sound effects
	Parameter	voice_number: Touch detection pin, ranging from 1~16
	Return Value	Execution result: true: Succeeded false: Failed
	Note	Refer to the Touch Detection Pins section of the User Guide for touch location
7	bool openLED(uint8_t led_number)	
	Description	Turn on the touch indicator
	Parameter	led_number: LED number, ranging from 1~16
	Return Value	Execution result: true: Succeeded false: Failed
	Note	Refer to the Hardware Overview section of the User Guide for LED location
8	bool closeLED(uint8_t led_number)	
	Description	Turn off the touch indicator
	Parameter	led_number: LED number, ranging from 1~16
	Return Value	Execution result: true: Succeeded false: Failed
	Note	Refer to the Hardware Overview section of the User Guide for LED location
9	uint16_t getTouchValue(void)	
	Description	Read the touch value
	Parameter	—
	Return Value	bit0~bit15: Corresponding to the 1~16 touch detection pins status bit=0: No pressed bit=1: Pressed
	Note	Refer to the Touch Detection Pins f the User Guide for touch location
10	uint8_t getINT(void)	
	Description	Get the INT pin level
	Parameter	—
	Return Value	INT pin level 0x00: Low, no touch action 0x01: High, touch action occurs
	Note	—

Arduino Lib Download and Installation

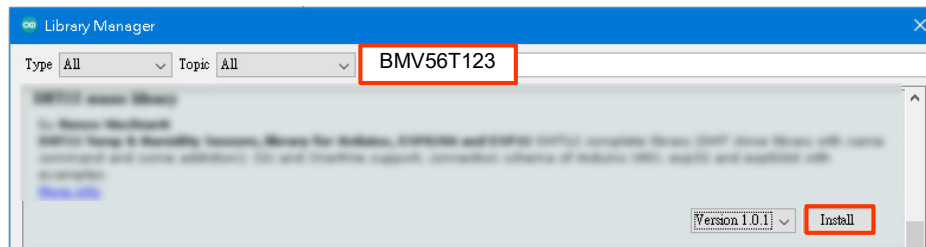
BMV56T123 Library: Refer to the following two methods to install the BMV56T123Arduino Library

Method 1: Search for installation

Search for installation: Arduino IDE → Sketch → Include Library → Manage Libraries... → Search BMV56T123 → Install



Search for Installation Step 1

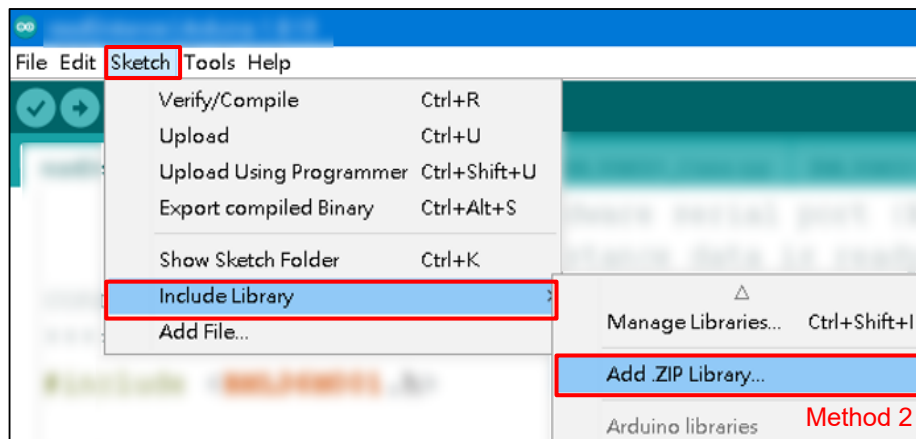


Search for Installation Step 2

Method 2: Download before adding a ZIP library

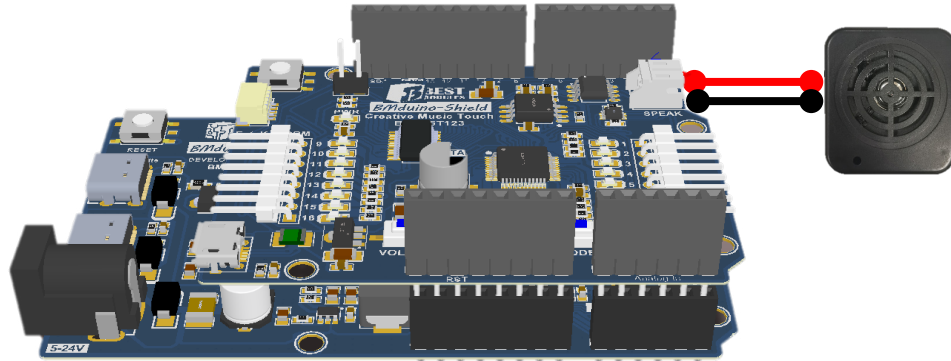
Download method: Open the Best Modules official website (<https://www.bestmodulescorp.com/bmv56t123.html#tab-product2>) and download the BMV56T123 Library from “Arduino example program” under the “DOCUMENTS” menu.

Add .ZIP library: Arduino IDE → Sketch → Include Library → Add .ZIP Library....



Arduino Example

Example: getTouchValue



Physical Connection Diagram

Example function: Demonstrate how to read the touch key values using BMduino.

1. Open the example: Arduino IDE → File → Examples → Select Lib (BMV51T123) → Select example (getTouchValue)
2. Example description:
 - a. Create object & Initialise shield and configuration

```
#include <BMV56T123.h>

#define INTPin          4

BMV56T123 BM_MusicTouch(INTPin, 2, 3); // Create object

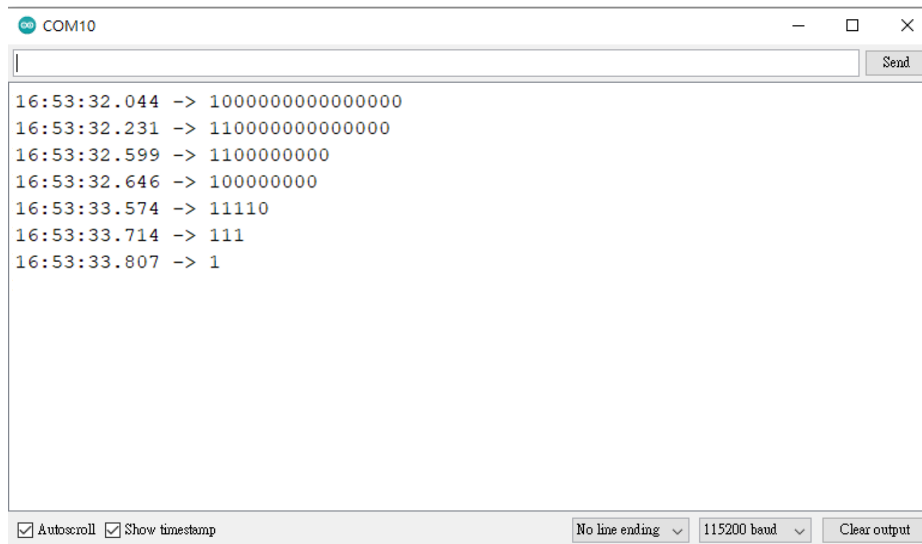
void setup()
{
  Serial.begin(9600); // Configure the serial monitor
  BM_MusicTouch.begin(); // Initialise the shield, the internal default
                        // software serial communication rate is 115200
  BM_MusicTouch.setPalyMode(INTERNAL_SOUND_SOURCE); // Play the MIDI
                                                    // voice source
  BM_MusicTouch.setTimbreGroup(8); // Set as the 8th tone group
  BM_MusicTouch.setVolume(10);    // Set the volume level to 10
}
```

b. Read the touch key value

```
void loop()
{
  uint32_t key_value;

  if(BM_MusicTouch.getINT() == 0) // Low represents a key has been
                                  // pressed
  {
    if(BM_MusicTouch.getTouchValue(&key_value) == true) // Read succeeded
    {
      Serial.println(key_value, BIN); // Print 16-bit binary result,
                                      // where 1 represents touch
                                      // action occurs in this bit
    }
  }
}
```

3. Output result: Open the serial monitor and select the baud rate to be 115200. The serial monitor will display as follows:



The screenshot shows the Serial Monitor window for COM10. The output displays a series of binary values over time, indicating touch events. The values are: 1000000000000000, 1100000000000000, 1100000000, 1000000000, 11110, 111, and 1. The window includes a 'Send' button and settings for 'Autoscroll', 'Show timestamp', 'No line ending', '115200 baud', and 'Clear output'.

```
16:53:32.044 -> 1000000000000000
16:53:32.231 -> 1100000000000000
16:53:32.599 -> 1100000000
16:53:32.646 -> 1000000000
16:53:33.574 -> 11110
16:53:33.714 -> 111
16:53:33.807 -> 1
```

Copyright© 2023 by BEST MODULES CORP. All Rights Reserved.

The information provided in this document has been produced with reasonable care and attention before publication, however, BEST MODULES does not guarantee that the information is completely accurate. The information contained in this publication is provided for reference only and may be superseded by updates. BEST MODULES disclaims any expressed, implied or statutory warranties, including but not limited to suitability for commercialization, satisfactory quality, specifications, characteristics, functions, fitness for a particular purpose, and non-infringement of any third-party's rights. BEST MODULES disclaims all liability arising from the information and its application. In addition, BEST MODULES does not recommend the use of BEST MODULES' products where there is a risk of personal hazard due to malfunction or other reasons. BEST MODULES hereby declares that it does not authorise the use of these products in life-saving, life-sustaining or safety critical components. Any use of BEST MODULES' products in life-saving/sustaining or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold BEST MODULES harmless from any damages, claims, suits, or expenses resulting from such use. The information provided in this document, including but not limited to the content, data, examples, materials, graphs, and trademarks, is the intellectual property of BEST MODULES (and its licensors, where applicable) and is protected by copyright law and other intellectual property laws. No license, express or implied, to any intellectual property right, is granted by BEST MODULES herein. BEST MODULES reserves the right to revise the information described in the document at any time without prior notice. For the latest information, please contact us.