4-KEY Capacitive Touch Module

BMK52M134
Arduino Library V1.0.1 Description

Revision: V1.10 Date: December 20, 2023

www.bestmodulescorp.com

Contents

Introduction

The Best Modules BMK52M134 is a 4-KEY capacitive touch module, which uses the I²C communication method. This document provides the description of the BMK52M134 Arduino Lib functions and how to install the Arduino Lib. The example demonstrates the function of obtaining the touch key value.

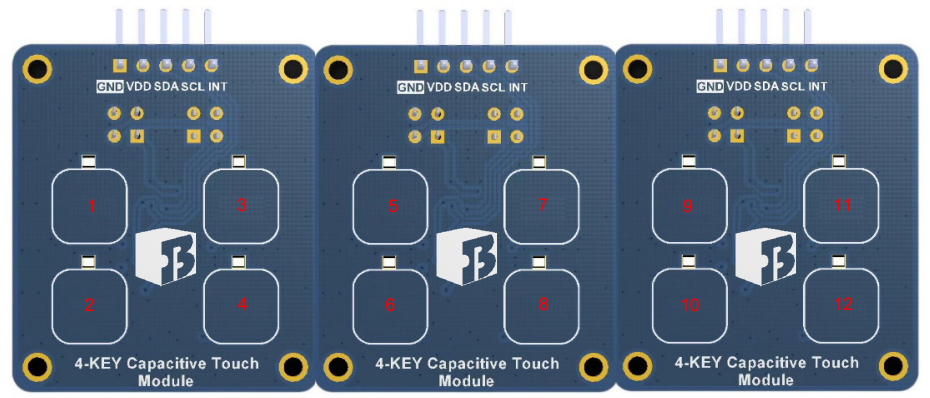
Arduino Lib Functions

Arduino Lib Name: BMK52M134		Lib Version: V1.0.1
Constructors & Initialisation		
1	BMK52M134(uint8_t intPin, TwoWire *thewire=&Wire)	
	Description	Constructor
	Parameter	intPin: INT pin, connect to the INT pin of the module *theWire: Select the I²C interface
	Return Value	—
	Note	—
2	void begin(uint8_t i2c_addr=BMK52M134_ADDRESS)	
	Description	Module initialisation
	Parameter	i2c_addr: I²C address, 0x71
	Return Value	void
	Note	—
Performance Functions		
3	uint8_t getINT()	
	Description	Obtain the INT pin level
	Parameter	—
	Return Value	INT pin level: 0: Low level, touch action occurs 1: High level, no touch action occurs
	Note	—
4	uint8_t getNumber()	
	Description	Obtain the cascade module number
	Parameter	—
	Return Value	Cascade module number
	Note	—
5	void getKeyValueArray(uint8_t key_value[])	
	Description	Obtain the key status of all cascade modules
	Parameter	key_value[]: The key state of the ID=1~N modules key_value[0]: Store the key value of the ID=1 modules (Bit0~Bit3 corresponds to KEY1~KEY4) Bit=0: No key is pressed Bit=1: A key is pressed key_value[N-1]: Store the key value of the ID=N module (Bit0~Bit3 corresponds to KEY1~KEY4) Bit=0: No key is pressed Bit=1: A key is pressed
	Return Value	void
	Note	ID: Refer to the Multi-board Cascade chapter of the user guide. This function supports pressing multiple keys at the same time.

6	uint8_t getKeyValue()	
	Description	Obtain the key status
	Parameter	—
	Return Value	Key value(1)
	Note	For example, if the ID=2 module KEY2 is pressed, the key value=6. This function does not support pressing multiple keys at the same time; If multiple keys are pressed at the same time, the most recently pressed key will be recognized.
7	uint8_t getThresholdSingle(uint8_t sensor_number)	
	Description	Obtain the key trigger threshold for the specified module
	Parameter	sensor_number: Module ID
	Return Value	Key trigger threshold
	Note	ID: Refer to the Multi-board Cascade chapter of the user guide.
8	uint8_t getSleepENSingle(uint8_t sensor_number)	
	Description	Obtain the specified module sleep mode state
	Parameter	sensor_number: Module ID
	Return Value	Sleep state: 0x00: Sleep disable 0x01: Sleep enable
	Note	ID: Refer to the Multi-board Cascade chapter of the user guide.
Parameter Configuration		
9	uint8_t setThresholdAll(uint8_t Threshold)	
	Description	Set the key trigger threshold for all cascade modules
	Parameter	Threshold: Key trigger threshold, range: 10~64
	Return Value	Execution result: 0x00: Succeeded 0x40: Failed
	Note	—
10	uint8_t setThresholdSingle(uint8_t sensor_number, uint8_t Threshold)	
	Description	Set the key trigger threshold for the specified module
	Parameter	sensor_number: Module ID Threshold: Key trigger threshold, range: 10~64
	Return Value	Execution result: 0x00: Succeeded 0x40: Failed
	Note	ID: Refer to the Multi-board Cascade chapter of the user guide.
11	uint8_t setSleepENAll(uint8_t sleepen)	
	Description	Set the sleep mode state for all cascade modules
	Parameter	sleepen: 0x00: Sleep disable 0x01: Sleep enable
	Return Value	Execution result: 0x00: Command sent successful 0x40: Checksum wrong 0x80: Instruction is not supported 0xA0: Cascade slaves do not respond
	Note	The module will automatically enter sleep state when no touch action occurs for 8s.

12	uint8_t setSleepENSingle(uint8_t sensor_number, uint8_t sleepen)	
	Description	Set the sleep state for the specified module
	Parameter	sensor_number: Module ID sleepen: 0x00: Sleep disable 0x01: Sleep enable
	Return Value	Execution result: 0x00: Command sent successful 0x40: Checksum wrong 0x80: Instruction is not supported 0xA0: Cascade slaves do not respond
	Note	ID: Refer to the Multi-board Cascade chapter of the user guide. The module will automatically enter sleep state when no touch action occurs for 8s.

Note 1: The “getKeyValue” function obtains the key value as follows:

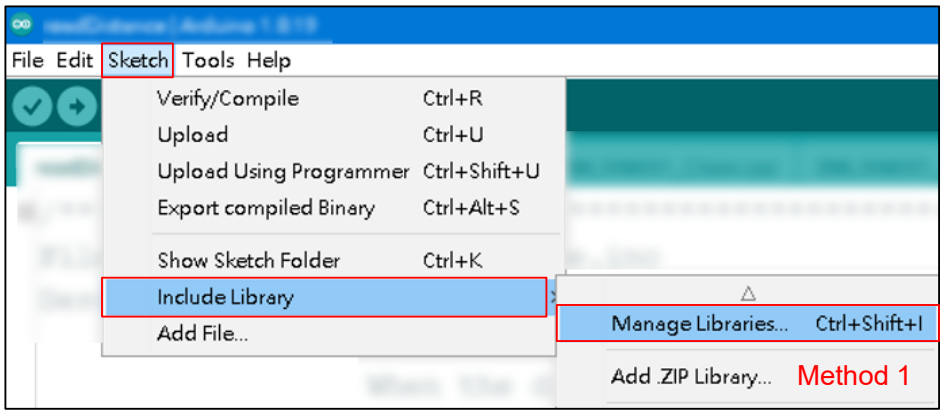


Arduino Lib Download and Installation

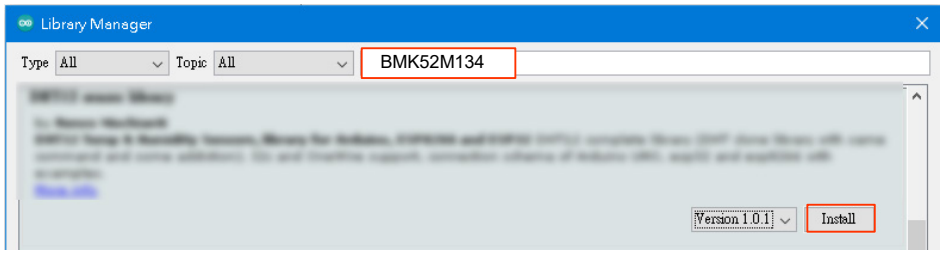
BMK52M134 Library: Refer to the following two methods to install the BMK52M134 Arduino Library.

Method 1: Search for installation

Arduino IDE→Sketch→Include Library→Manage Libraries...→Search BMK52M134→Install



Search for Installation Step 1

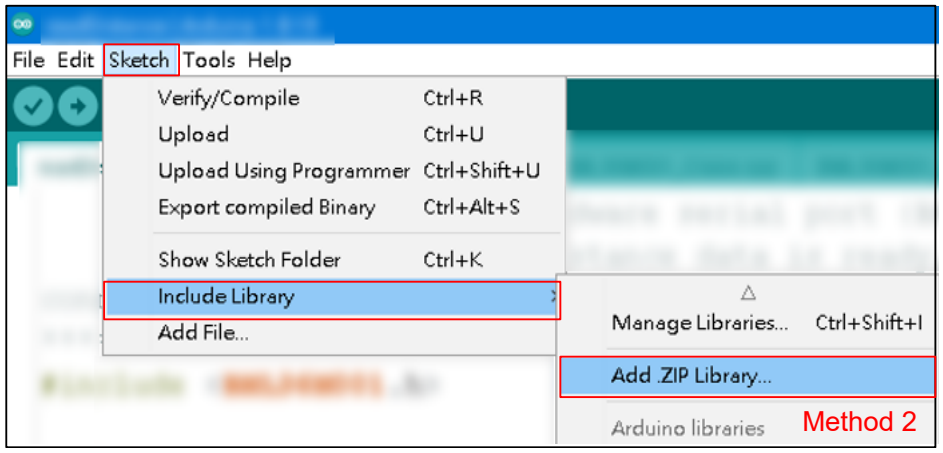


Search for Installation Step 2

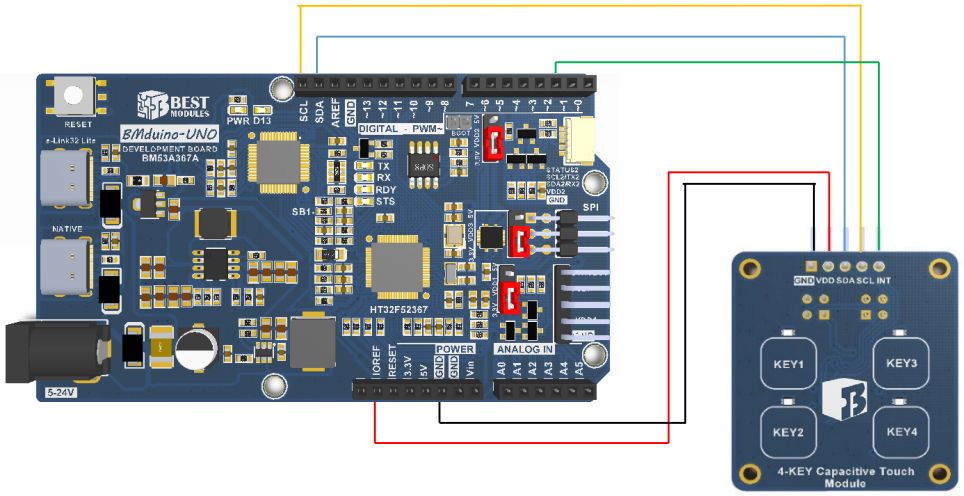
Method 2: Download the .ZIP library before adding it

Download the Arduino example (BMK52M134 Library) under the DOCUMENTS menu from the Best Modules website (https://www.bestmodulescorp.com/bmk52m134.html)

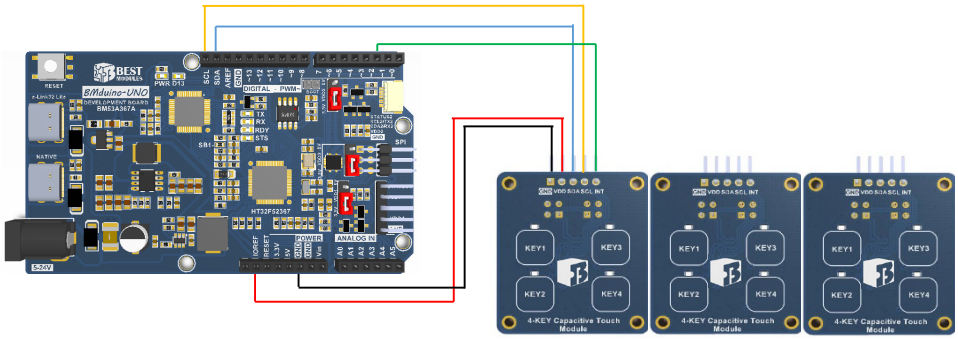
Add .ZIP library: Arduino IDE→Sketch→Include Library→Add .ZIP Library...



Arduino Example



Physical Connection Diagram (Single module)



Physical Connection Diagram (Cascade Module)

Example 1: BasicReading

Example implemented function: Obtain the key value of the module and display it on the serial monitor.

1. Open the example: Arduino IDE→File→Examples→Select Lib (BMK52M134)→Select example (BasicReading)
2. Example Description:

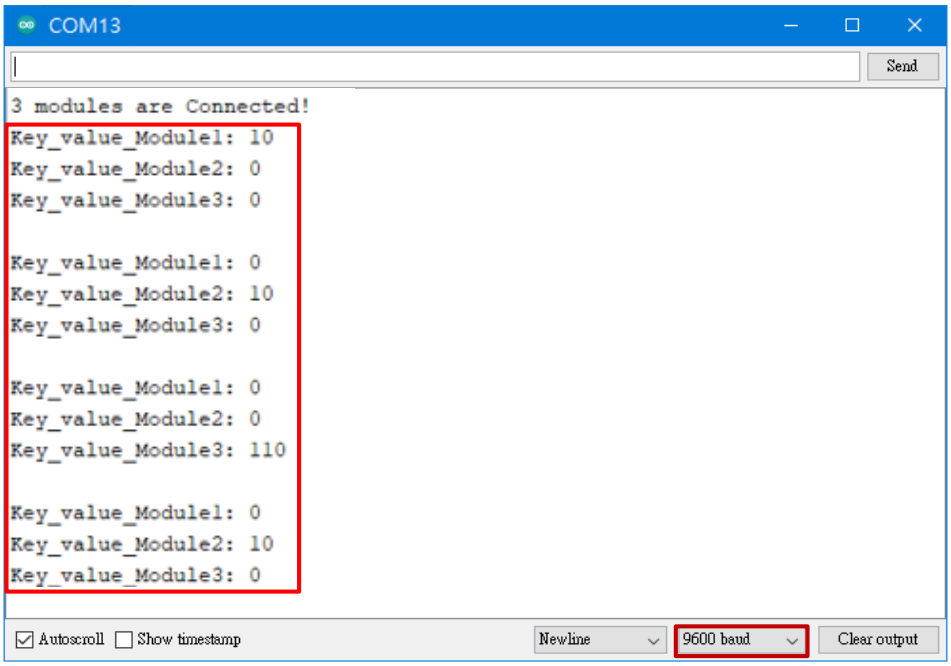
a. Create object & initialise module

```
#include <BMK52M134.h>
BMK52M134 sensor(2, &Wire);   // Create object
uint8_t sumofmodules;
void setup()
{
  Serial.begin(9600);         // Configure the serial monitor
  sensor.begin();             // Module initialisation
  sumofmodules = sensor.getNumber();
  if (sumofmodules != 0)
  {
    Serial.print(sumofmodules);
    Serial.print(" modules are ");
    Serial.println("Connected!");
  }
  else
  {
    Serial.println("Not connected. Please check connections.");
  }
}
```

b. Obtain the key value and display it in the serial monitor

```
void loop()
{
  uint8_t key_value[5] = {0}; // An array that stores key values
  if (sensor.getINT() == 0)
  {
    sensor.getKeyValueArray(key_value);
    for (uint8_t i = 0; i < sumofmodules; i++)
    {
      Serial.print("Key_value_Module");
      Serial.print(i+1);
      Serial.print(": ");
      Serial.print(key_value[i], BIN);
      Serial.println(" ");
    }
    Serial.println();
  }
}
```

3. Open the serial monitor and set the baud rate to be 9600. The serial monitor will display as follows.



Example 2: detectCurrentTouch

Example implemented function: Obtain the the pressed key serial number and display it on the serial monitor, if multiple modules are cascaded, the pressed key serial number will increase. For example, if the second module KEY1 is pressed, the key serial number will be 5, and so on.

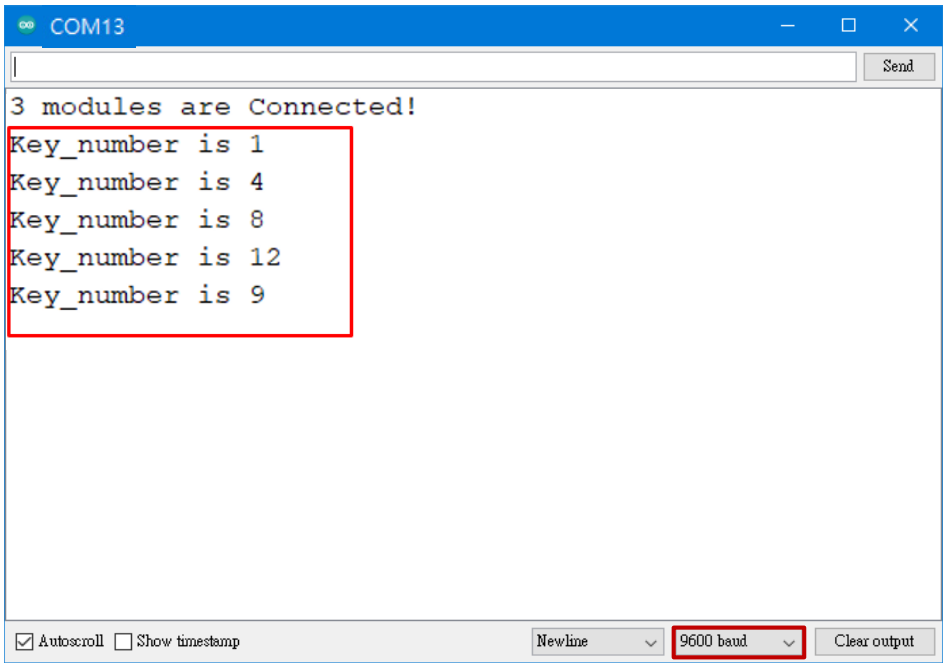
1. Open the example: Arduino IDE→File→Examples→Select Lib (BMK52M134)→Select example (detectCurrentTouch)
2. Example Description:
   a. Create object & initialise module

```
#include <BMK52M134.h>
BMK52M134 sensor(2, &Wire); // Create object
uint8_t sumofmodules;
void setup()
{
  Serial.begin(9600);       // Configure the serial monitor
  sensor.begin();           // Module initialisation
  sumofmodules = sensor.getNumber();
  if (sumofmodules !=0)
  {
    Serial.print(sumofmodules);
    Serial.print(" modules are ");
    Serial.println("Connected!");
  }
  else
  {
    Serial.println("Not connected. Please check connections.");
  }
}
```

b. Obtain the pressed key serial number and displays it on the serial monitor

```
void loop()
{
  static uint8_t key_number_last;
  static uint32_t time_out;
  if (sensor.getINT() == 0)
  {
    time_out = millis();
    if (sensor.getKeyValue() != key_number_last)
    {
      key_number_last = sensor.getKeyValue();
      Serial.print("Key_number is ");
      Serial.print(sensor.getKeyValue());
      Serial.print(" ");
      Serial.println();
    }
  }
  else
  {
    if (millis() - time_out > 200)
    {
      key_number_last = 0;
    }
  }
}
```

3. Open the serial monitor and set the baud rate to be 9600. The serial monitor will display as follows.



Example 3: LongPress

Example implemented function: Use the function of pressing the key for a certain time and display it on the serial monitor

1. Open the example: Arduino IDE→File→Examples→Select Lib (BMK52M134)→Select example (LongPress)
2. Example Description:

a. Create object & initialise module

```
#include <BMK52M134.h>
BMK52M134 sensor(2, &Wire); // Create object
uint8_t sumofmodules;
void setup()
{
  Serial.begin(9600);       // Configure the serial monitor
  sensor.begin();           // Module initialisation
  sumofmodules = sensor.getNumber();
  if (sumofmodules == 1)
  {
    Serial.print(sumofmodules);
    Serial.print(" modules are ");
    Serial.println("Connected!");
  }
  else
  {
    Serial.println("Not connected. Please check connections.");
  }
}
```

b. Obtain the key value and long press time and display them in the serial monitor

```
void loop()
{
  static uint32_t key_start_time, key_current_time, time_out;
  static uint8_t  key_number_last, count;
  if(sensor.getINT()!=0)
  {
    time_out = millis();
    if(sensor.getKeyValue() != key_number_last)
    {
      key_start_time = millis();
      key_current_time = millis();
      count = 0;
      key_number_last = sensor.getKeyValue();
      Serial.print("Key_number is ");
      Serial.print(sensor.getKeyValue());
      Serial.println();
    }
    else if(key_number_last == sensor.getKeyValue())
    {
      key_current_time = millis();
    }
    if( key_current_time - key_start_time > 3000)
    {
      Serial.print(" and Keep ");
      Serial.print(((key_current_time - key_start_time) / 1000) *
                   (count + 1));
```

```
        Serial.println("s");
        key_start_time = millis();
        count++;
      }
    }
    else
    {
      if( millis() - time_out > 300)
      {
        key_number_last = 0;
      }
    }
}
```

3. Open the serial monitor and set the baud rate to be 9600. The serial monitor will display as follows.