MCP23017 IO Expansion Board

User Manual

OVERVIEW

This is an I/O expansion module uses I2C interface, gives you 16 GPIO. Supports using 8 expansion modules at the same for up to 128 general purpose pins. Compatible with 3.3V and 5V voltage.

FEATURES

- Use two i2c pins, expands up to 16 pins I/O
- I2C address is hardware configurable by changing A0/A1/A2
- PH2.0 terminal and weld pad, supports parallelly connecting multiple I2C modules.
- Integrates level conversion circuit, compatible with 3.3V and 5V.
- Comes with development resources and manual (for RaspberryPi/micro:bit/Arduino/STM32)

SPECIFICATIONS

Operating voltage:	5V/3.3V
Interface:	I2C
Interrupt pin:	INTA, INTB
Expansion I/O:	16
Dimension:	38mmx23mm
Holes size:	2.0mm

INTERFACE

PIN	Description	Arduino Uno	STM32F407	Raspberry Pi
Vcc	Voltage	3.3V/5V	3.3V/5V	3.3V/5V
GND	Ground	GND	GND	GND
SDA	I2C data line	SDA	PB7	SDA
SCL	I2C clock line	SCL	PB6	SCL
INTA[1]	Interrupt pin	3	PA0	0(WiringPi)
INTB[2]	Interrupt pin	3	PA0	0(WiringPi)

[1] You can just connect one interrupt pin when using.

[2] You can just connect one interrupt pin when using.

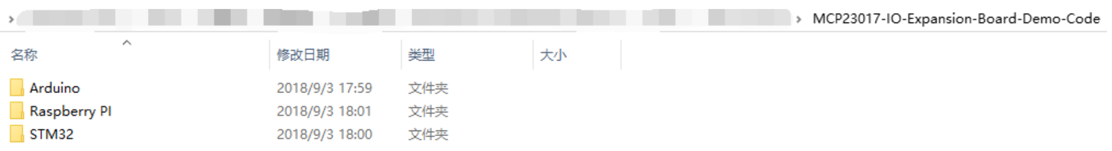
HOW TO USE

WORKING WITH RASPBERRY PI

LIBRARIES INSTALLATION

To use the Raspberry Pi demo code, you need to install WiringPi libraries. Please refer to the page: https://www.waveshare.com/wiki/Libraries_Installation_for_RPi

Download demo code from Waveshare Wiki and extract it to get the demo code.



› [redacted] › MCP23017-IO-Expansion-Board-Demo-Code

名称	修改日期	类型	大小
Arduino	2018/9/3 17:59	文件夹	
Raspberry PI	2018/9/3 18:01	文件夹	
STM32	2018/9/3 18:00	文件夹	

Copy Raspberry PI folder to your Raspberry Pi. (generally copy to boot of the SD card which has Raspbian installed)

RUNNING THE CODE

Setting

1. Configuration

```
sudo raspi-config
```

2. Choose Interfacing Options -> I2C ->yes to enable I2C interface

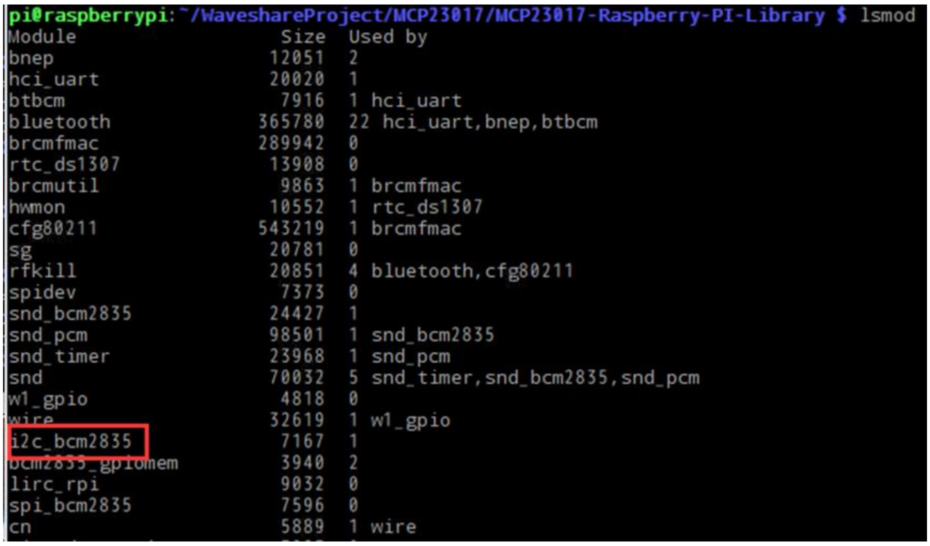
3. Save and reboot Raspberry Pi

```
sudo reboot
```

4. After rebooting, you can check if I2C module is enabled by command:

```
lsmod
```

I2C module is enabled normally if you get the information as below,



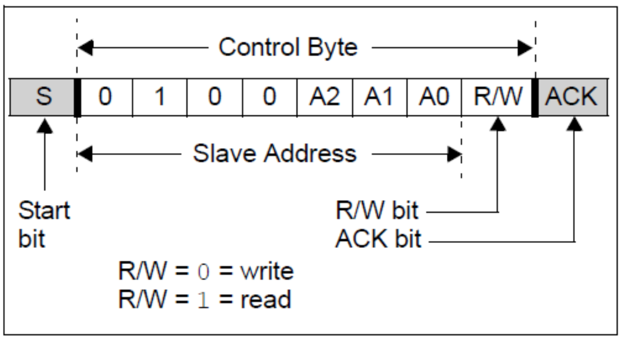
```
pi@raspberrypi:~/WaveshareProject/MCP23017/MCP23017-Raspberry-PI-Library $ lsmod
Module                  Size  Used by
bnep                   12051  2
hci_uart               20020  1
btbcm                   7916  1 hci_uart
bluetooth             365780  22 hci_uart,bnep,btbcm
brcmfmac              289942  0
rtc_ds1307             13908  0
brcmutil                9863  1 brcmfmac
hwmon                  10552  1 rtc_ds1307
cfg80211              543219  1 brcmfmac
sg                     20781  0
rfkill                 20851  4 bluetooth,cfg80211
spidev                  7373  0
snd_bcm2835            24427  1
snd_pcm                98501  1 snd_bcm2835
snd_timer              23968  1 snd_pcm
snd                    70032  5 snd_timer,snd_bcm2835,snd_pcm
w1_gpio                 4818  0
wire                   32619  1 w1_gpio
i2c_bcm2835             7167  1
bcm2835_gpiomem         3940  2
lirc_rpi                9032  0
spi_bcm2835             7596  0
cn                      5889  1 wire
```

5. Connecting MCP23017 module to Raspberry Pi according to interfaces

6. You can just connect one of the two interrupt pins to 0(wiringPi) as INTA and INTB are configured to interconnecting by code

Default address of module is 0x27, you can also change its device address by welding A2, A0, A1. By default, A2, A1, A0 are all High without welded. If you want to turn them to Low, you should short these welding points. According to datasheet, slave address can be configured as below:

FIGURE 3-4: I^2C CONTROL BYTE FORMAT

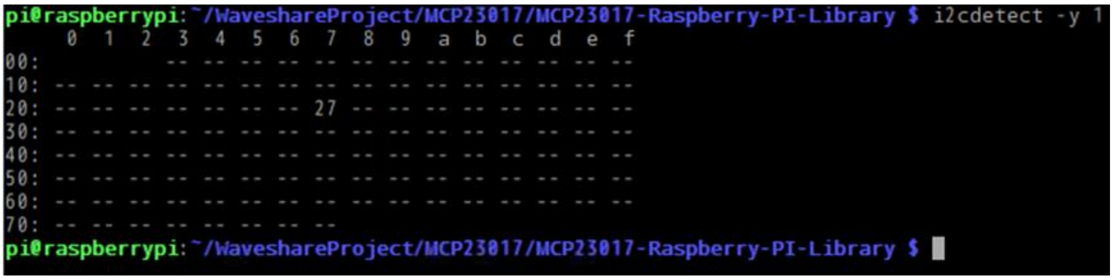


7. Install i2c-tools for inquiring slave address of i2c modules

```
sudo apt-get install i2c-tools
```

8. Inquiring i2c devices connected

```
i2cdetect -y 1
```



```
pi@raspberrypi:~/WaveshareProject/MCP23017/MCP23017-Raspberry-PI-Library $ i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- 27 -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
pi@raspberrypi:~/WaveshareProject/MCP23017/MCP23017-Raspberry-PI-Library $ 
```

If you didn’ t change the slave address of MCP23017 module, you will get the information as above, the slave address is 0x27 inquired as we expected. You can also change the address by welding A2/A1/A0 points, however, you need to make sure there are not other modules which have the same address first.

Running

1. Copy the demo code (we have copied to boot) to /home/pi/ directory.

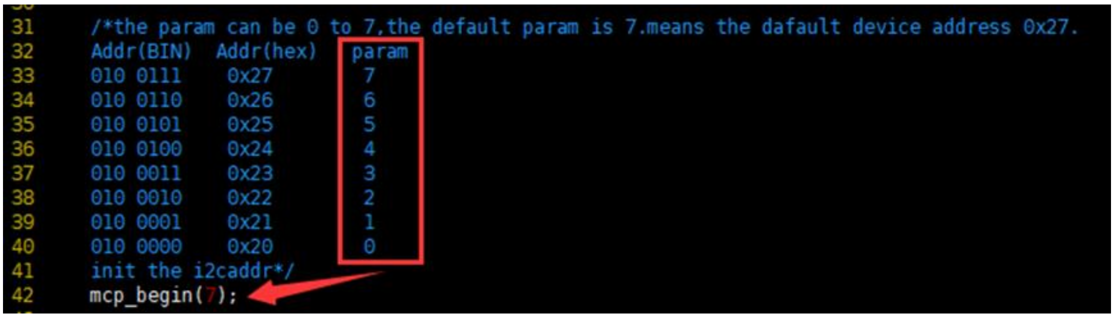
2. Enter the directory of demo code

```
cd MCP23017-Raspberry-PI-Library
```

3. Open main.c file

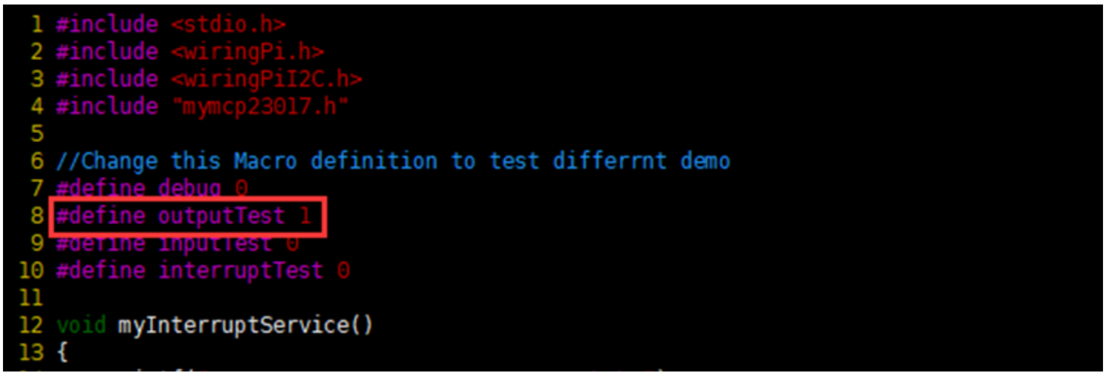
```
vim main.c
```

4. Check the slave address first. If you once changed the address, you should change the parameter of mcp_begin().



```
	/*the param can be 0 to 7,the default param is 7.means the dafault device address 0x27.
	Addr(BIN)  Addr(hex)   param
	010 0111    0x27       7
	010 0110    0x26       6
	010 0101    0x25       5
	010 0100    0x24       4
	010 0011    0x23       3
	010 0010    0x22       2
	010 0001    0x21       1
	010 0000    0x20       0
	init the i2caddr*/
	mcp_begin(7);
```

Then, enable macro for corresponding testing.



```
#include <stdio.h>
#include <wiringPi.h>
#include <wiringPiI2C.h>
#include "mymcp23017.h"

//Change this Macro definition to test differnt demo
#define debug 0
#define outputTest 1
#define inputTest 0
#define interruptTest 0

void myInterruptService()
{
```

#define debug: 1: print registers’ value to console

#define outputTest: 1: Output testing

#define inputTest: 1: Input testing

#define interruptTest: 1; Interrupt output testing

【Note】

Output testing: set outputTest to 1; inputTest and interruptTest to 0;

Input testing: set inputTest to 1; outputTest and interruptTest to 0;

Interrupt testing: set interruptTest to 1; outputTest and inputTest to 0

5. After modifying, save and exit.

6. Compiling demo code

```
make
```

7. running the code

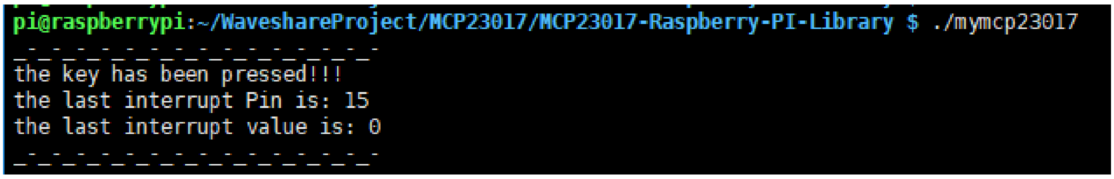
```
./mymcp23017
```

EXPECTED RESULT

Output Testing: MCP23017 output High level to 16 I/O for 500ms, and then Low level for 500ms. You can connect LED to these pins for testing.

Input Testing: Set PA0[3] of MCP23017 as output and PA1 as input. PA0 will output level states which is read by PA1. You can connect PA1 to Vcc or GND, and then connect PA to LED for testing.

Interrupt Testing: Set PA7 and PB7 pins of MCP23017 as interrupt pins. interrupt information will be printed to console as below:



```
pi@raspberrypi:~/WaveshareProject/MCP23017/MCP23017-Raspberry-PI-Library $ ./mymcp23017
_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-
the key has been pressed!!!
the last interrupt Pin is: 15
the last interrupt value is: 0
_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-
```

[3] PA0~PA7 are printed as 0~7 and PB0~PB7 are printed as 8~15.

WORKING WITH ARDUINO

1. Connect MCP23017 module to Arduino UNO/UNO PLUS according to interfaces

2. Download demo code and extract it. Copy MCP23017-Arduino-Library folder to libraries directory of IDE.

Output Testing:

1. Open Arduino IDE. Click File->Examples->MCP23017-Arduino-Library->output

2. Configure i2c slave address, default 0x27. You can refer to Running the code about slave address configuration

3. Compile and upload the code to Arduino board

Expected result: Connect LED to one pin of MCP2317 module, the LED will flicker.

Input Testing:

1. Open Arduino IDE. Click File->Examples->MCP23017-Arduino-Library->input

2. Configure i2c slave address, default 0x27. You can refer to Running the code about slave address configuration

3. Compile and upload the code to Arduino board

Expected result: Connect PA0 of modules to LED, connect PA1 to Vcc or GND, the LED will light on/off according to PA1.

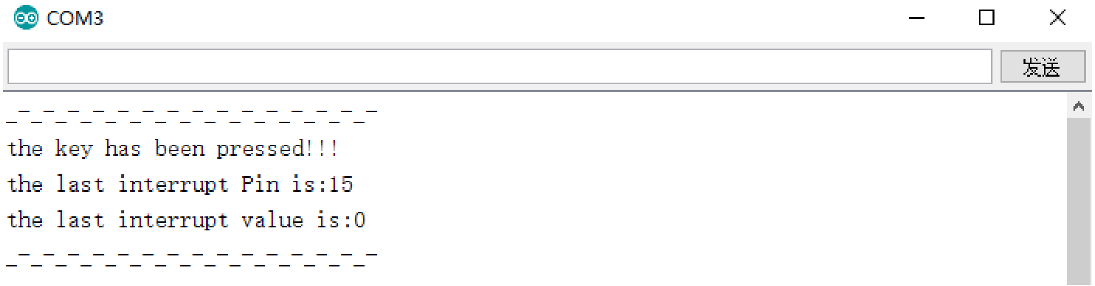
Interrupt Testing:

1. Open Arduino IDE. Click File->Examples->MCP23017-Arduino-Library->interrupt

2. Configure i2c slave address, default 0x27. You can refer to Running the code about slave address configuration

3. Compile and upload the code to Arduino board

Expected result: Open serial monitor, interrupt information will be printed



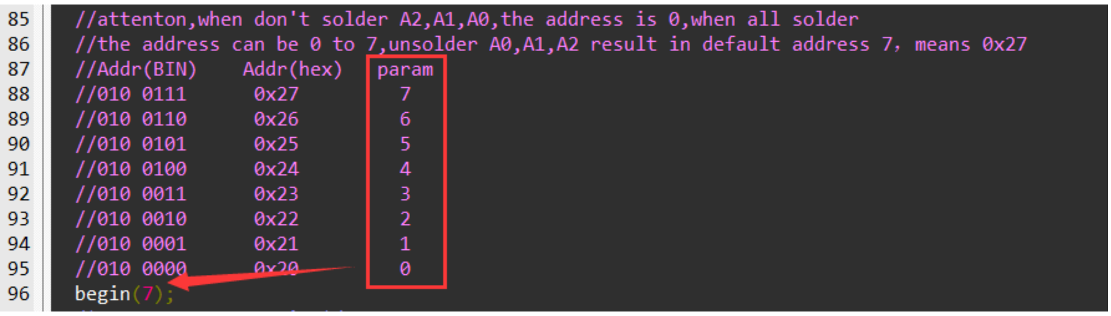
```
_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_
the key has been pressed!!!
the last interrupt Pin is:15
the last interrupt value is:0
_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_
```

WORKING WITH STM32

1. Connect MCP23017 module to STM32F407 development board

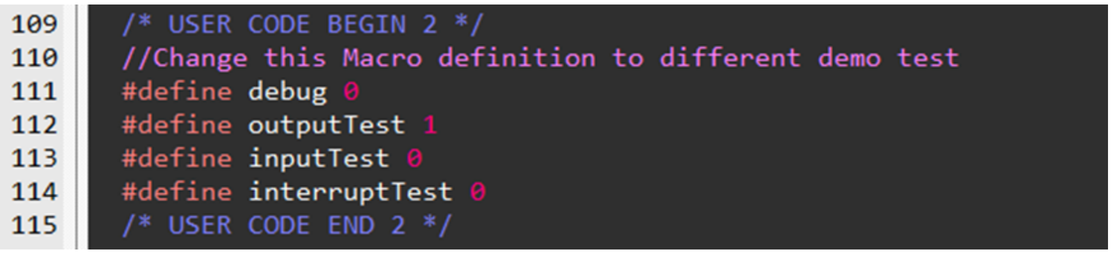
2. Download demo code form Waveshare Wiki and extract it. Open STM32 project with software keil5:STM32->MCP23017-STM32F4-Library->MDK-ARM.

Open file main.c to configure i2c slave address. Default address is 0x27. About how to change it. You can refer to Running the code about slave address configuration



```
//attenton,when don't solder A2,A1,A0,the address is 0,when all solder
//the address can be 0 to 7,unsolder A0,A1,A2 result in default address 7, means 0x27
//Addr(BIN)     Addr(hex)   param
//010 0111       0x27         7
//010 0110       0x26         6
//010 0101       0x25         5
//010 0100       0x24         4
//010 0011       0x23         3
//010 0010       0x22         2
//010 0001       0x21         1
//010 0000       0x20         0
begin(7);
```

Change macros:



```
/* USER CODE BEGIN 2 */
//Change this Macro definition to different demo test
#define debug 0
#define outputTest 1
#define inputTest 0
#define interruptTest 0
/* USER CODE END 2 */
```

【Note】

Output testing: set outputTest to 1; inputTest and interruptTest to 0;

Input testing: set inputTest to 1; outputTest and interruptTest to 0;

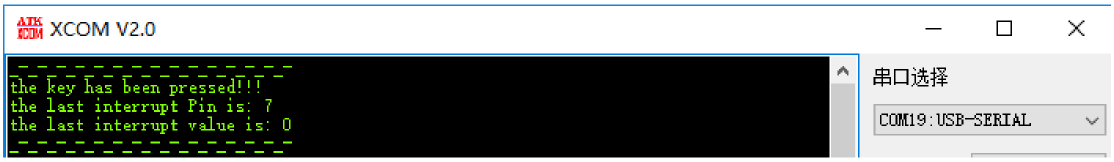
Interrupt testing: set interruptTest to 1; outputTest and inputTest to 0

3. Compile and download demo code to STM32 board.

Output Testing: MCP23017 output High level to 16 I/O for 500ms, and then Low level for 500ms. You can connect LED to these pins for testing.

Input Testing: Set PA0[4] of MCP23017 as output and PA1 as input. PA0 will output level states which is read by PA1. You can connect PA1 to Vcc or GND, and then connect PA to LED for testing.

Interrupt Testing: Set PA7 and PB7 pins of MCP23017 as interrupt pins. interrupt information will be printed to serial as below:



COM19:USB-SERIAL

```
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _
the key has been pressed!!!
the last interrupt Pin is: 7
the last interrupt value is: 0
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _
```

[4] PA0~PA7 are printed as 0~7 and PB0~PB7 are printed as 8~15.