ZMCP23017 Library

Generated by Doxygen 1.8.14

Contents

1 Deprecated List

Global ZMCP23017::check ()

2 Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

   PinExtender

3 Data Structure Index

3.1 Data Structures

Here are the data structures with brief descriptions:

4 File Index

4.1 File List

Here is a list of all files with brief descriptions:

5 Data Structure Documentation

5.1 ZMCP23017 Class Reference

```
#include <ZMCP23017.h>
```

Inheritance diagram for ZMCP23017:

```
┌─────────────────────┐
│     PinExtender      │
├─────────────────────┤
│                     │
├─────────────────────┤
│                     │
└─────────────────────┘
           △
           │
┌─────────────────────┐
│      MCP23017       │
├─────────────────────┤
│                     │
├─────────────────────┤
│ + begin()           │
│ + begin()           │
│ + begin()           │
│ + acceptlocal()     │
│ + test()            │
│ + check()           │
│ + setHardAddress()  │
│ + setWire()         │
│ + pullUp()          │
│ + digitalWordRead() │
│ + digitalWordWrite()│
│ + inputPolarityMask()│
│ + inputOutputMask() │
│ + internalPullupMask()│
│ + analogRead()      │
│ + pinMode()         │
│ + digitalWrite()    │
│ + digitalRead()     │
│ + analogWrite()     │
│ + analogWriteResolution()│
│ * test()            │
│ * check()           │
│ * setHardAddress()  │
│ * setWire()         │
│ * pullUp()          │
│ * digitalWordRead() │
│ * digitalWordWrite()│
│ * inputPolarityMask()│
│ * inputOutputMask() │
│ * internalPullupMask()│
│ * analogRead()      │
│ * pinMode()         │
│ * digitalWrite()    │
│ * digitalRead()     │
│ * analogWrite()     │
│ * analogWriteResolution()│
└─────────────────────┘
```

Collaboration diagram for ZMCP23017:

```
                    ┌─────────────────┐
                    │   PinExtender    │
                    ├─────────────────┤
                    │                  │
                    ├─────────────────┤
                    │                  │
                    └─────────────────┘
                             △
                             │
                    ┌─────────────────┐
                    │    MCP23017     │
                    ├─────────────────┤
                    │                  │
                    ├─────────────────┤
                    │ + begin()        │
                    │ + begin()        │
                    │ + begin()        │
                    │ + acceptlocal()  │
                    │ + test()         │
                    │ + check()        │
                    │ + setHardAddress()│
                    │ + setWire()      │
                    │ + pullUp()       │
                    │ + digitalWordRead()│
                    │ + digitalWordWrite()│
                    │ + inputPolarityMask()│
                    │ + inputOutputMask()│
                    │ + internalPullupMask()│
                    │ + analogRead()   │
                    │ + pinMode()      │
                    │ + digitalWrite() │
                    │ + digitalRead()  │
                    │ + analogWrite()  │
                    │ + analogWriteResolution()│
                    │ * test()         │
                    │ * check()        │
                    │ * setHardAddress()│
                    │ * setWire()      │
                    │ * pullUp()       │
                    │ * digitalWordRead()│
                    │ * digitalWordWrite()│
                    │ * inputPolarityMask()│
                    │ * inputOutputMask()│
                    │ * internalPullupMask()│
                    │ * analogRead()   │
                    │ * pinMode()      │
                    │ * digitalWrite() │
                    │ * digitalRead()  │
                    │ * analogWrite()  │
                    │ * analogWriteResolution()│
                    └─────────────────┘
```

Public Member Functions

- void begin (TwoWire *MyWire, uint8_t addr)
- void begin (uint8_t addr)
- void begin (void)
- bool acceptlocal (uint32_t p)

hardware API

- bool test ()
- bool check ()
- void setHardAddress (uint8_t A210)
- void setWire (TwoWire ∗MyWire)
- void pullUp (uint32_t p, uint8_t d)
- word digitalWordRead ()
- void digitalWordWrite (word w)
- void inputPolarityMask (word mask)
- void inputOutputMask (word mask)
- void internalPullupMask (word mask)

arduino like API

- uint32_t analogRead (uint32_t pin)
- void pinMode (uint32_t p, uint8_t d)
- void digitalWrite (uint32_t p, uint8_t d)
- uint8_t digitalRead (uint32_t p)
- void analogWrite (uint32_t ulPin, uint32_t ulValue)
- void analogWriteResolution (int res)

5.1.1 Detailed Description

Definition at line 69 of file ZMCP23017.h.

5.1.2 Member Function Documentation

5.1.2.1 acceptlocal()

```
bool ZMCP23017::acceptlocal (
            uint32_t p )
```

this test if the arduino pin is a pin for this object a pin of this object is a ulPin value where the value is equal to this.getPin(channel) for a channel=0..15

Definition at line 306 of file ZMCP23017.cpp.

Referenced by analogWrite(), digitalRead(), digitalWrite(), and pinMode().

Here is the caller graph for this function:
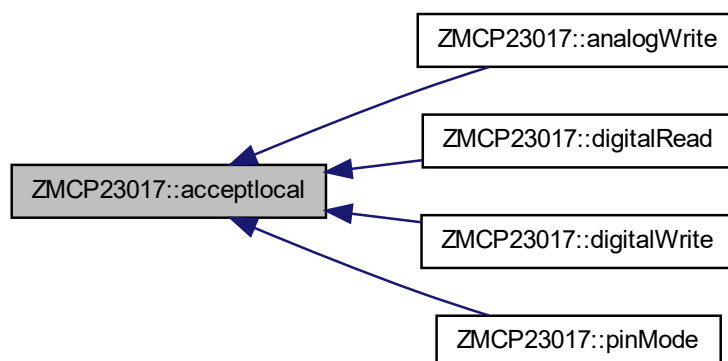
5.1.2.2   analogRead()

```
uint32_t ZMCP23017::analogRead (
            uint32_t pin )
```

dummy function for compatibility with arduino API, always return 0

Parameters

pin	the pin requested, it is the instance number.

Definition at line 62 of file ZMCP23017.cpp.

5.1.2.3   analogWrite()

```
void ZMCP23017::analogWrite (
            uint32_t ulPin,
            uint32_t ulValue )
```

define the analog output value of the pin like arduino API ulValue can be 0..4096...

ulPin is a pin from this.getPin(x) where x=0..15 here not PWM are avalable so a value below _writeResolution/2 generate a LOW state else an HIGH state will be apply

Parameters

ulPin	the pin requested, it is the instance number.

Definition at line 248 of file ZMCP23017.cpp.

References acceptlocal().

Here is the call graph for this function:

5.1.2.4 analogWriteResolution()

```
void ZMCP23017::analogWriteResolution (
            int res )
```

define the analog Write Resolution of the components like arduino API res can be 1..16

Definition at line 235 of file ZMCP23017.cpp.

5.1.2.5 begin() [1/3]

```
void ZMCP23017::begin (
            TwoWire * My_i2c,
            uint8_t addr )
```

initialise the board, the Wire interface must be initialize before. see wire.begin()

Initializes the ZMCP23017 given its HW selected address, see datasheet for Address selection. or use setHard↩
Address to over write it.

Parameters

My_i2c	the Wire interface like &Wire for board that handle several one.
addr	the I2C address of MCP23017

Definition at line 123 of file ZMCP23017.cpp.

References ZMCP23017_ADDRESS, ZMCP23017_ADDRESS_MASK, and ZMCP23017_IODIR.

5.1.2.6 begin() [2/3]

```
void ZMCP23017::begin (
            uint8_t addr )
```

initialise the board, the Wire interface must be initialize before. see wire.begin()

Initializes the ZMCP23017 given its HW selected address, see datasheet for Address selection.

Parameters

addr	the I2C address of MCP23017

Definition at line 152 of file ZMCP23017.cpp.

References begin().

Here is the call graph for this function:

5.1.2.7 begin() [3/3]

```
void ZMCP23017::begin (
            void  )
```

initialise the board, the Wire interface must be initialize before. see wire.begin()

Initializes the default ZMCP23017, with 000 for the configurable part of the address

Definition at line 159 of file ZMCP23017.cpp.

Referenced by begin().

Here is the caller graph for this function:

5.1.2.8 check()

```
bool ZMCP23017::check ( )
```

check the board

Deprecated

Definition at line 65 of file ZMCP23017.cpp.

Referenced by test().

Here is the caller graph for this function:
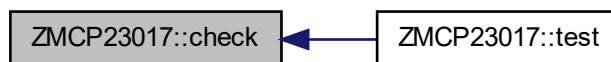
5.1.2.9 digitalRead()

```
uint8_t ZMCP23017::digitalRead (
            uint32_t ulPin )
```

return the digital value of the pin like arduino API

ulPin is a pin from this.getPin(channel) where channel=0..15 returns LOW or HIGH, in case of no pin it returns LOW

Parameters

ulPin	the pin requested, it is the instance number.

Definition at line 291 of file ZMCP23017.cpp.

References acceptlocal(), and ZMCP23017_GPIO.

Here is the call graph for this function:

5.1.2.10 digitalWordRead()

```
word ZMCP23017::digitalWordRead ( )
```

These provide a more advanced mapping of the chip functionality See the data sheet for more information on what they do

Returns

Returns a word with the current pin states (ie contents of the GPIO register)

this function return a word that is the image to of the bus GPIOB[7..0]:GPIOA[7..0]

Definition at line 317 of file ZMCP23017.cpp.

References ZMCP23017_GPIO.

5.1.2.11 digitalWordWrite()

```
void ZMCP23017::digitalWordWrite (
            word w )
```

Allows you to write a word to the GPIO register

this function write a word that is applied to the bus GPIOB[7..0]:GPIOA[7..0]

Definition at line 323 of file ZMCP23017.cpp.

References ZMCP23017_GPIO.

5.1.2.12 digitalWrite()

```
void ZMCP23017::digitalWrite (
            uint32_t ulPin,
            uint8_t val )
```

define the output value of the pin like arduino API val can be LOW, HIGH ulPin is a pin from this.getPin(x) where x=0..15 if input mode is set, a High active pull up.

Parameters

ulPin	the pin requested, it is the instance number.

Definition at line 205 of file ZMCP23017.cpp.

References acceptlocal(), ZMCP23017_GPIO, and ZMCP23017_GPPU.

Here is the call graph for this function:

5.1.2.13 inputOutputMask()

```
void ZMCP23017::inputOutputMask (
            word mask )
```

Sets which pins are inputs or outputs (1 = input, 0 = output) NB Opposite to arduino's definition for these

this function define the direction of the bus GPIOB[7..0]:GPIOA[7..0], one bit for each pin 1 means input 0 means output

Definition at line 336 of file ZMCP23017.cpp.

References ZMCP23017_IODIR.

5.1.2.14 inputPolarityMask()

```
void ZMCP23017::inputPolarityMask (
            word mask )
```

Sets up the polarity mask that the MCP23017 supports if set to 1, it will flip the actual pin value.

Definition at line 328 of file ZMCP23017.cpp.

References ZMCP23017_IPOL.

5.1.2.15 internalPullupMask()

```
void ZMCP23017::internalPullupMask (
            word mask )
```

Allows enabling of the internal 100k pullup resisters (1 = enabled, 0 = disabled)

this function define the pull up activation of the bus GPIOB[7..0]:GPIOA[7..0], one bit for each pin 1 means enabled 0 means disabled

Definition at line 344 of file ZMCP23017.cpp.

References ZMCP23017_GPPU.

Referenced by pullUp().

Here is the caller graph for this function:
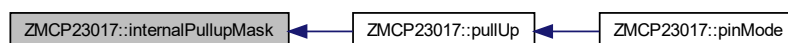
5.1.2.16 pinMode()

```
void ZMCP23017::pinMode (
            uint32_t ulPin,
            uint8_t mode )
```

Sets the pin mode to either INPUT or OUTPUT

void ZMCP23017::pinMode(uint32_t p, uint8_t d) { if(acceptlocal(p)) { updateRegisterBit(p,(d==INPUT),ZMC↩P23017_IODIRA,ZMCP23017_IODIRB); } else if (_next) return _next->pinMode(p,d);

} define the mode of the pin like arduino API mode can be INPUT_PULLUP, INPUT,OUTPUT ulPin is a pin from this.getPin(x) where x=0..15

Parameters

ulPin	the pin requested, it is the instance number.

Definition at line 180 of file ZMCP23017.cpp.

References acceptlocal(), pullUp(), and ZMCP23017_IODIR.

Here is the call graph for this function:
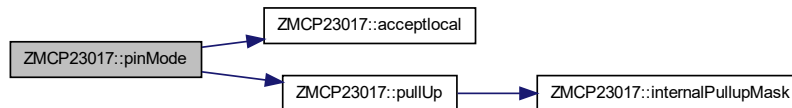
5.1.2.17 pullUp()

```
void ZMCP23017::pullUp (
            uint32_t p,
            uint8_t d )
```

define the pull upof the pin d can be 0 for nio pull up and 1 to enable pull up p is a pin from 0..15

Definition at line 267 of file ZMCP23017.cpp.

References internalPullupMask().

Referenced by pinMode().

Here is the call graph for this function:

Here is the caller graph for this function:

5.1.2.18 setHardAddress()

```
void ZMCP23017::setHardAddress (
            uint8_t A210 )
```

setup the custom address

define the address based of PIN A2 , A1,A0 value

Parameters

A210	value of pin A2..A0

Definition at line 71 of file ZMCP23017.cpp.

5.1.2.19 setWire()

```
void ZMCP23017::setWire (
            TwoWire * MyWire )
```

set the Wire object in case of several avalable.

Definition at line 86 of file ZMCP23017.cpp.

5.1.2.20 test()

```
bool ZMCP23017::test ( )
```

Test the Hardware to be sure that the connection is good. else it return false.

Returns

true : if the communication work well and it look like that it is the good chip behing I2C interface

Definition at line 92 of file ZMCP23017.cpp.

References check(), ZMCP23017_IOCONA, and ZMCP23017_IOCONB.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- ZMCP23017.h
- ZMCP23017.cpp

6 File Documentation

6.1   ZMCP23017.cpp File Reference

```
#include "ZMCP23017.h"
#include <WireUtility.h>
```
Include dependency graph for ZMCP23017.cpp:
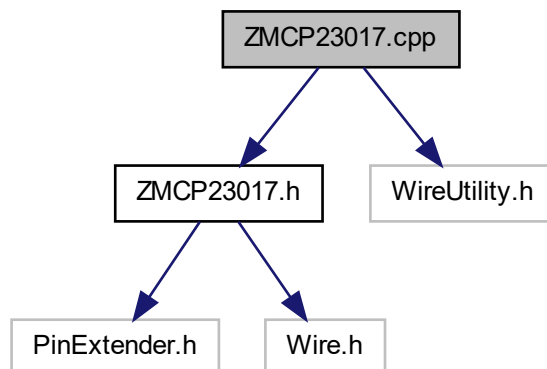
Macros

- #define ZMCP23017_ADDRESS 0x20
- #define ZMCP23017_ADDRESS_MASK 0x78
- #define ZMCP23017_IODIRA 0x00
- #define ZMCP23017_IPOLA 0x02
- #define ZMCP23017_GPINTENA 0x04
- #define ZMCP23017_DEFVALA 0x06
- #define ZMCP23017_INTCONA 0x08
- #define ZMCP23017_IOCONA 0x0A
- #define ZMCP23017_GPPUA 0x0C
- #define ZMCP23017_INTFA 0x0E
- #define ZMCP23017_INTCAPA 0x10
- #define ZMCP23017_GPIOA 0x12
- #define ZMCP23017_OLATA 0x14
- #define ZMCP23017_IODIRB 0x01
- #define ZMCP23017_IPOLB 0x03
- #define ZMCP23017_GPINTENB 0x05
- #define ZMCP23017_DEFVALB 0x07
- #define ZMCP23017_INTCONB 0x09
- #define ZMCP23017_IOCONB 0x0B
- #define ZMCP23017_GPPUB 0x0D
- #define ZMCP23017_INTFB 0x0F
- #define ZMCP23017_INTCAPB 0x11
- #define ZMCP23017_GPIOB 0x13
- #define ZMCP23017_OLATB 0x15

- #define ZMCP23017_INT_ERR 255
- #define ZMCP23017_IODIR 0x00
- #define ZMCP23017_IPOL 0x2
- #define ZMCP23017_GPPU 0x0C
- #define ZMCP23017_GPIO 0x12

6.1.1 Macro Definition Documentation

6.1.1.1 ZMCP23017_ADDRESS

```
#define ZMCP23017_ADDRESS 0x20
```

Definition at line 18 of file ZMCP23017.cpp.

Referenced by ZMCP23017::begin().

6.1.1.2 ZMCP23017_ADDRESS_MASK

```
#define ZMCP23017_ADDRESS_MASK 0x78
```

Definition at line 19 of file ZMCP23017.cpp.

Referenced by ZMCP23017::begin().

6.1.1.3 ZMCP23017_DEFVALA

```
#define ZMCP23017_DEFVALA 0x06
```

Definition at line 27 of file ZMCP23017.cpp.

6.1.1.4 ZMCP23017_DEFVALB

```
#define ZMCP23017_DEFVALB 0x07
```

Definition at line 40 of file ZMCP23017.cpp.

6.1.1.5 ZMCP23017_GPINTENA

```
#define ZMCP23017_GPINTENA 0x04
```

Definition at line 26 of file ZMCP23017.cpp.

6.1.1.6 ZMCP23017_GPINTENB

```
#define ZMCP23017_GPINTENB 0x05
```

Definition at line 39 of file ZMCP23017.cpp.

6.1.1.7 ZMCP23017_GPIO

```
#define ZMCP23017_GPIO 0x12
```

Definition at line 57 of file ZMCP23017.cpp.

Referenced by ZMCP23017::digitalRead(), ZMCP23017::digitalWordRead(), ZMCP23017::digitalWordWrite(), and ZMCP23017::digitalWrite().

6.1.1.8 ZMCP23017_GPIOA

```
#define ZMCP23017_GPIOA 0x12
```

Definition at line 33 of file ZMCP23017.cpp.

6.1.1.9 ZMCP23017_GPIOB

```
#define ZMCP23017_GPIOB 0x13
```

Definition at line 46 of file ZMCP23017.cpp.

6.1.1.10 ZMCP23017_GPPU

```
#define ZMCP23017_GPPU 0x0C
```

Definition at line 56 of file ZMCP23017.cpp.

Referenced by ZMCP23017::digitalWrite(), and ZMCP23017::internalPullupMask().

6.1.1.11 ZMCP23017_GPPUA

```
#define ZMCP23017_GPPUA 0x0C
```

Definition at line 30 of file ZMCP23017.cpp.

6.1.1.12 ZMCP23017_GPPUB

```
#define ZMCP23017_GPPUB 0x0D
```

Definition at line 43 of file ZMCP23017.cpp.

6.1.1.13 ZMCP23017_INT_ERR

```
#define ZMCP23017_INT_ERR 255
```

Definition at line 49 of file ZMCP23017.cpp.

6.1.1.14 ZMCP23017_INTCAPA

```
#define ZMCP23017_INTCAPA 0x10
```

Definition at line 32 of file ZMCP23017.cpp.

6.1.1.15 ZMCP23017_INTCAPB

```
#define ZMCP23017_INTCAPB 0x11
```

Definition at line 45 of file ZMCP23017.cpp.

6.1.1.16 ZMCP23017_INTCONA

```
#define ZMCP23017_INTCONA 0x08
```

Definition at line 28 of file ZMCP23017.cpp.

6.1.1.17 ZMCP23017_INTCONB

```
#define ZMCP23017_INTCONB 0x09
```

Definition at line 41 of file ZMCP23017.cpp.

6.1.1.18 ZMCP23017_INTFA

```
#define ZMCP23017_INTFA 0x0E
```

Definition at line 31 of file ZMCP23017.cpp.

6.1.1.19 ZMCP23017_INTFB

```
#define ZMCP23017_INTFB 0x0F
```

Definition at line 44 of file ZMCP23017.cpp.

6.1.1.20 ZMCP23017_IOCONA

```
#define ZMCP23017_IOCONA 0x0A
```

Definition at line 29 of file ZMCP23017.cpp.

Referenced by ZMCP23017::test().

6.1.1.21 ZMCP23017_IOCONB

```
#define ZMCP23017_IOCONB 0x0B
```

Definition at line 42 of file ZMCP23017.cpp.

Referenced by ZMCP23017::test().

6.1.1.22 ZMCP23017_IODIR

```
#define ZMCP23017_IODIR 0x00
```

Definition at line 54 of file ZMCP23017.cpp.

Referenced by ZMCP23017::begin(), ZMCP23017::inputOutputMask(), and ZMCP23017::pinMode().

6.1.1.23 ZMCP23017_IODIRA

```
#define ZMCP23017_IODIRA 0x00
```

Definition at line 24 of file ZMCP23017.cpp.

6.1.1.24 ZMCP23017_IODIRB

```
#define ZMCP23017_IODIRB 0x01
```

Definition at line 37 of file ZMCP23017.cpp.

6.1.1.25 ZMCP23017_IPOL

```
#define ZMCP23017_IPOL 0x2
```

Definition at line 55 of file ZMCP23017.cpp.

Referenced by ZMCP23017::inputPolarityMask().

6.1.1.26 ZMCP23017_IPOLA

```
#define ZMCP23017_IPOLA 0x02
```

Definition at line 25 of file ZMCP23017.cpp.

6.1.1.27 ZMCP23017_IPOLB

```
#define ZMCP23017_IPOLB 0x03
```

Definition at line 38 of file ZMCP23017.cpp.

6.1.1.28 ZMCP23017_OLATA

```
#define ZMCP23017_OLATA 0x14
```

Definition at line 34 of file ZMCP23017.cpp.

6.1.1.29 ZMCP23017_OLATB

```
#define ZMCP23017_OLATB 0x15
```

Definition at line 47 of file ZMCP23017.cpp.

6.2   ZMCP23017.h File Reference

```
#include "PinExtender.h"
#include <Wire.h>
```
Include dependency graph for ZMCP23017.h:
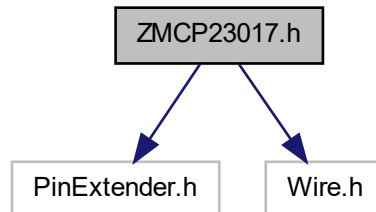
This graph shows which files directly or indirectly include this file:
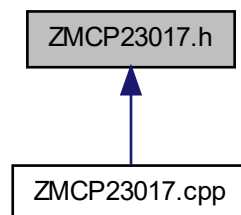
Data Structures

- class ZMCP23017

Macros

- #define NO_CHANNEL 0xfe

channel of the component

- #define MCP23017_GPA7 7
- #define MCP23017_GPA6 6
- #define MCP23017_GPA5 5
- #define MCP23017_GPA4 4
- #define MCP23017_GPA3 3
- #define MCP23017_GPA2 2

- #define MCP23017_GPA1 1
- #define MCP23017_GPA0 0
- #define MCP23017_GPB7 15
- #define MCP23017_GPB6 14
- #define MCP23017_GPB5 13
- #define MCP23017_GPB4 12
- #define MCP23017_GPB3 11
- #define MCP23017_GPB2 10
- #define MCP23017_GPB1 9
- #define MCP23017_GPB0 8

6.2.1 Detailed Description

This library manage the component MCP23017, this is a 16 pin I/O extention driver by I2C protocol

dependency :

this library use the folowing ones : PinExtender you can find it on `https://github.com/zoubworld↩ Arduino/`

Author

strongly inspired from MCP23017 library of Adafruit

This is a library for the MCP23017 i2c port expander

These displays use I2C to communicate, 2 pins are required to
interface Adafruit invests time and resources providing this open source code, please support Adafruit and open-source hardware by purchasing products from Adafruit!

Written by Limor Fried/Ladyada for Adafruit Industries.
BSD license, all text above must be included in any redistribution

description

6.2.2 Macro Definition Documentation

6.2.2.1 MCP23017_GPA0

```
#define MCP23017_GPA0 0
```

Definition at line 56 of file ZMCP23017.h.

6.2.2.2 MCP23017_GPA1

```
#define MCP23017_GPA1 1
```

Definition at line 55 of file ZMCP23017.h.

6.2.2.3 MCP23017_GPA2

`#define MCP23017_GPA2 2`

Definition at line 54 of file ZMCP23017.h.

6.2.2.4 MCP23017_GPA3

`#define MCP23017_GPA3 3`

Definition at line 53 of file ZMCP23017.h.

6.2.2.5 MCP23017_GPA4

`#define MCP23017_GPA4 4`

Definition at line 52 of file ZMCP23017.h.

6.2.2.6 MCP23017_GPA5

`#define MCP23017_GPA5 5`

Definition at line 51 of file ZMCP23017.h.

6.2.2.7 MCP23017_GPA6

`#define MCP23017_GPA6 6`

Definition at line 50 of file ZMCP23017.h.

6.2.2.8 MCP23017_GPA7

`#define MCP23017_GPA7 7`

Definition at line 49 of file ZMCP23017.h.

6.2.2.9 MCP23017_GPB0

`#define MCP23017_GPB0 8`

Definition at line 65 of file ZMCP23017.h.

6.2.2.10 MCP23017_GPB1

```
#define MCP23017_GPB1 9
```

Definition at line 64 of file ZMCP23017.h.

6.2.2.11 MCP23017_GPB2

```
#define MCP23017_GPB2 10
```

Definition at line 63 of file ZMCP23017.h.

6.2.2.12 MCP23017_GPB3

```
#define MCP23017_GPB3 11
```

Definition at line 62 of file ZMCP23017.h.

6.2.2.13 MCP23017_GPB4

```
#define MCP23017_GPB4 12
```

Definition at line 61 of file ZMCP23017.h.

6.2.2.14 MCP23017_GPB5

```
#define MCP23017_GPB5 13
```

Definition at line 60 of file ZMCP23017.h.

6.2.2.15 MCP23017_GPB6

```
#define MCP23017_GPB6 14
```

Definition at line 59 of file ZMCP23017.h.

6.2.2.16 MCP23017_GPB7

```
#define MCP23017_GPB7 15
```

Definition at line 58 of file ZMCP23017.h.

6.2.2.17 NO_CHANNEL

```
#define NO_CHANNEL 0xfe
```

Definition at line 67 of file ZMCP23017.h.

Index