Erriez MCP23017 library for Arduino

1.0.0

Generated by Doxygen 1.8.13

Contents

Chapter 1

MCP23017 16-pin I2C IO-expander library for Arduino

This is a MCP23017 16-pin I2C IO-expander library for Arduino with interrupt change/edge support and extensive examples.

Library features

- I2C interface
- Input/output/read/write/toggle/mask control per pin or per 16-pins
- Configurable pullup per pin
- Interrupt change/falling/rising per pin
- Interrupt edge handled by the library, because the chip supports pin change / level only.
- Low-power support
- Generic examples / AVR / ESP8266 / ESP32 support

Hardware

The following targets are supported:

- AVR: UNO, MINI, Pro Mini 8/16 MHz, ATMega2560, Leonardo
- ARM: DUE
- ESP8266: Mini D1 & D2, NodeMCU
- ESP32: Lolin D32

Examples

Extensive examples are located `here`.

Documentation

Doxygen API documentation is located here.

Getting started LED blink

```
{c++}
#include <Arduino.h>
#include <Wire.h>
#include <ErriezMCP23017.h>

// PORTA: pins 0..7
// PORTB: pins 8..15
#define LED_PIN                 8 // Pin B0

// Default I2C Address 0x20
#define MCP23017_I2C_ADDRESS    0x20

// Create MCP23017 object
ErriezMCP23017 mcp = ErriezMCP23017(MCP23017_I2C_ADDRESS);

void setup()
{
    // Initialize Wire
    Wire.begin();
    Wire.setClock(400000);

    // Initialize MCP23017
    while (!mcp.begin()) {
        // MCP23017 not detected
        delay(3000);
    }

    // LED pin output
    mcp.pinMode(LED_PIN, OUTPUT);
}

void loop()
{
    mcp.digitalWrite(LED_PIN, HIGH);
    delay(1000);
    mcp.digitalWrite(LED_PIN, LOW);
    delay(1000);
}
```

Library installation

Please refer to the Wiki page.

Other Arduino Libraries and Sketches from Erriez

Erriez Libraries and Sketches

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 4

Class Documentation

4.1 ErriezMCP23017 Class Reference

Erriez MCP23017 I2C IO-Expander class.

```
#include <ErriezMCP23017.h>
```

Public Member Functions

- ErriezMCP23017 (uint8_t i2cAddress=MCP23017_I2C_ADDRESS, TwoWire *twoWire=&Wire)

  ErriezMCP23017 Constructor.
- bool begin (bool reset=true)

  Initialize MCP23017.
- void pinMode (uint8_t pin, uint8_t mode)

  Set direction of a single pin.
- void digitalWrite (uint8_t pin, uint8_t level)

  Set state of a single pin.
- int digitalRead (uint8_t pin)

  Get state of a single pin.
- void setPortDirection (uint16_t outputPins)

  Set PORT direction all pins.
- uint16_t getPortDirection ()

  Get PORT direction all pins.
- void setPortPullup (uint16_t pullupPins)

  Set PORT pullup all pins.
- uint16_t getPortPullup ()

  Get PORT pullup all pins.
- void pinWrite (uint8_t pin, bool level)

  Set pin state.
- void pinToggle (uint8_t pin)

  Toggle state of a single pin (only for output pins)
- bool pinRead (uint8_t pin)

  Read state of a single pin (input and output pins)
- void portWrite (uint16_t value)

  Set all pin states.

- void portToggle (uint16_t value)

  Toggle pin states (output pins only)
- void portMask (uint16_t maskSet, uint16_t maskClear)

  Clear and set pin states.
- uint16_t portRead ()

  Read PORT of all pins (input and output pins)
- void setInterruptPolarityINTA (bool activeHigh)

  Set interrupt polarity INTA.
- uint16_t getPortInterruptMask ()

  Get interrupt mask all pins.
- void setPortInterruptEnable (uint16_t pins)

  Enable interrupt change on pins.
- void setPortInterruptDisable (uint16_t pins)

  Disable interrupt on pins.
- bool interruptINTA ()

  MCP23017 INTA pin changed.
- uint16_t registerRead (uint8_t reg)

  MCP23017 I2C read register.
- void registerWrite (uint8_t reg, uint16_t value)

  MCP23017 I2C write register.
- uint8_t getI2CStatus ()

  Return status of the last I2C write, returned by Wire endTransfer()
- void dumpRegisters (HardwareSerial ∗serial)

  Print I2C registers on serial port.

Public Attributes

- uint16_t portState

  Port state since last portRead() call.
- uint16_t pinsChanged

  Pins change on interrupt enabled pins since last intPinChanged() call.
- uint16_t pinsFalling

  Falling edge on interrupt enabled pins since last intPinChanged() call.
- uint16_t pinsRising

  Rising edge on interrupt eanbled pins since last intPinChanged() call.

4.1.1 Detailed Description

Erriez MCP23017 I2C IO-Expander class.

Definition at line 135 of file ErriezMCP23017.h.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 ErriezMCP23017()

```
ErriezMCP23017::ErriezMCP23017 (
            uint8_t i2cAddress = MCP23017_I2C_ADDRESS,
            TwoWire * twoWire = &Wire )
```

ErriezMCP23017 Constructor.

The constructor initializes internal variables and does not call I2C functions

Parameters

i2cAddress	7-bit MCP23017 I2C device address
twoWire	Default is Wire object to select I2C bus.

Definition at line 38 of file ErriezMCP23017.cpp.

4.1.3 Member Function Documentation

4.1.3.1 begin()

```
bool ErriezMCP23017::begin (
             bool reset = true )
```

Initialize MCP23017.

Parameters

reset	Perform resetting registers to default values (Default enabled)

Return values

true	MCP23017 detected
false	MCP23017 not detected

Definition at line 55 of file ErriezMCP23017.cpp.

4.1.3.2 digitalRead()

```
int ErriezMCP23017::digitalRead (
             uint8_t pin )
```

Get state of a single pin.

Parameters

pin	Pin number 0..15 (PORTA = 0..7, PORTB = 8..15)

Return values

HIGH	= 1
LOW	= 0

Definition at line 145 of file ErriezMCP23017.cpp.

4.1.3.3 digitalWrite()

```
void ErriezMCP23017::digitalWrite (
            uint8_t pin,
            uint8_t state )
```

Set state of a single pin.

Parameters

pin	Pin number 0..15 (PORTA = 0..7, PORTB = 8..15)
state	HIGH = 1 LOW = 0

Definition at line 132 of file ErriezMCP23017.cpp.

4.1.3.4 dumpRegisters()

```
void ErriezMCP23017::dumpRegisters (
            HardwareSerial * serial )
```

Print I2C registers on serial port.

This function is optimized away by the compiler when not used

Parameters

serial	Serial port

Definition at line 494 of file ErriezMCP23017.cpp.

4.1.3.5 getI2CStatus()

```
uint8_t ErriezMCP23017::getI2CStatus ( )
```

Return status of the last I2C write, returned by Wire endTransfer()

Return values

0	Success
1	Data too long to fit in transmit buffer
2	Received NACK on transmit of address
3	Received NACK on transmit of data
4	Other error

Definition at line 480 of file ErriezMCP23017.cpp.

4.1.3.6 getPortDirection()

```
uint16_t ErriezMCP23017::getPortDirection ( )
```

Get PORT direction all pins.

Returns

PORT direction pins 0..15, Arduino compatible:
Bit value '0': INPUT
Bit value '1': OUTPUT

Definition at line 174 of file ErriezMCP23017.cpp.

4.1.3.7 getPortInterruptMask()

```
uint16_t ErriezMCP23017::getPortInterruptMask ( )
```

Get interrupt mask all pins.

Returns

Interrupt enabled

Definition at line 345 of file ErriezMCP23017.cpp.

4.1.3.8 getPortPullup()

```
uint16_t ErriezMCP23017::getPortPullup ( )
```

Get PORT pullup all pins.

Returns

PORT pull-up pins 0..15:
Bit value '0': Pull-up disabled
Bit value '1': Pull-up enable

Definition at line 203 of file ErriezMCP23017.cpp.

4.1.3.9 interruptINTA()

```
bool ErriezMCP23017::interruptINTA ( )
```

MCP23017 INTA pin changed.

The application should call this function when the MCP23017 INTA pin changed. Default: Falling edge This function re

Return values

true	At least one pin changed
false	No pins changed (Pin pulse was too short, or INTA edge did not match)

Returns

portState: PORT state since last portRead() call
pinsChanged: Changed pins on interrupt pins since last call
pinsFalling: Falling edge on interrupt pins since last call
pinsRising: Rising edge on interrupt pins since last call

Definition at line 401 of file ErriezMCP23017.cpp.

4.1.3.10 pinMode()

```
void ErriezMCP23017::pinMode (
            uint8_t pin,
            uint8_t mode )
```

Set direction of a single pin.

Parameters

pin	Pin number 0..15 (PORTA = 0..7, PORTB = 8..15)
mode	OUTPUT: Configure pin as output INPUT: Configure pin as input INPUT_PULLUP: Configure pin with input pull-up

Definition at line 95 of file ErriezMCP23017.cpp.

4.1.3.11 pinRead()

```
bool ErriezMCP23017::pinRead (
            uint8_t pin )
```

Read state of a single pin (input and output pins)

Parameters

pin	Pin number 0..15 (PORTA = 0..7, PORTB = 8..15)

Return values

HIGH	= 1

Return values

LOW	= 0

Definition at line 248 of file ErriezMCP23017.cpp.

4.1.3.12 pinToggle()

```
void ErriezMCP23017::pinToggle (
            uint8_t pin )
```

Toggle state of a single pin (only for output pins)

Parameters

pin	Pin number 0..15 (PORTA = 0..7, PORTB = 8..15)

Definition at line 233 of file ErriezMCP23017.cpp.

4.1.3.13 pinWrite()

```
void ErriezMCP23017::pinWrite (
            uint8_t pin,
            bool level )
```

Set pin state.

Parameters

pin	Pin number 0..15 (PORTA = 0..7, PORTB = 8..15)
level	HIGH = 1, LOW = 0

Definition at line 217 of file ErriezMCP23017.cpp.

4.1.3.14 portMask()

```
void ErriezMCP23017::portMask (
            uint16_t maskSet,
            uint16_t maskClear )
```

Clear and set pin states.

Parameters

maskSet	Bit value '1': Pins to HIGH
maskClear	Bit value '1': Pins to LOW

Definition at line 293 of file ErriezMCP23017.cpp.

4.1.3.15 portRead()

```
uint16_t ErriezMCP23017::portRead ( )
```

Read PORT of all pins (input and output pins)

Returns

State of all 16 pins

Definition at line 305 of file ErriezMCP23017.cpp.

4.1.3.16 portToggle()

```
void ErriezMCP23017::portToggle (
            uint16_t value )
```

Toggle pin states (output pins only)

Parameters

value	16 pins, bit value '0' = unchanged, '1' = toggle

Definition at line 278 of file ErriezMCP23017.cpp.

4.1.3.17 portWrite()

```
void ErriezMCP23017::portWrite (
            uint16_t value )
```

Set all pin states.

Parameters

value	16 pins, bit value '0' = LOW, '1' = HIGH

Definition at line 265 of file ErriezMCP23017.cpp.

4.1.3.18 registerRead()

```
uint16_t ErriezMCP23017::registerRead (
            uint8_t reg )
```

MCP23017 I2C read register.

Parameters

reg	MCP23017 register

Returns

MCP23017 register value

Definition at line 441 of file ErriezMCP23017.cpp.

4.1.3.19 registerWrite()

```
void ErriezMCP23017::registerWrite (
            uint8_t reg,
            uint16_t value )
```

MCP23017 I2C write register.

Parameters

reg	MCP23017 register
value	MCP23017 value

Definition at line 461 of file ErriezMCP23017.cpp.

4.1.3.20 setInterruptPolarityINTA()

```
void ErriezMCP23017::setInterruptPolarityINTA (
            bool activeHigh )
```

Set interrupt polarity INTA.

Parameters

activeHigh	HIGH = 1: Active high, LOW = 0: Active low (default)

Definition at line 320 of file ErriezMCP23017.cpp.

4.1.3.21 setPortDirection()

```
void ErriezMCP23017::setPortDirection (
            uint16_t outputPins )
```

Set PORT direction all pins.

Parameters

outputPins	PORT direction pins 0..15, Arduino compatible: Bit value '0': INPUT Bit value '1': OUTPUT

Definition at line 159 of file ErriezMCP23017.cpp.

4.1.3.22 setPortInterruptDisable()

```
void ErriezMCP23017::setPortInterruptDisable (
            uint16_t pins )
```

Disable interrupt on pins.

Parameters

pins	Pins to disable

Definition at line 375 of file ErriezMCP23017.cpp.

4.1.3.23 setPortInterruptEnable()

```
void ErriezMCP23017::setPortInterruptEnable (
            uint16_t pins )
```

Enable interrupt change on pins.

The MCP23017 does not support edge interrupts. This is handled by software.

Parameters

pins	Pins to enable interrupt change

Definition at line 357 of file ErriezMCP23017.cpp.

4.1.3.24 setPortPullup()

```
void ErriezMCP23017::setPortPullup (
             uint16_t pullupPins )
```

Set PORT pullup all pins.

Parameters

pullupPins	Set PORT pull-up pins 0..15: Bit value '0': Pull-up unchanged Bit value '1': Pull-up enable

Definition at line 188 of file ErriezMCP23017.cpp.

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

- src/ErriezMCP23017.h
- src/ErriezMCP23017.cpp

Chapter 5

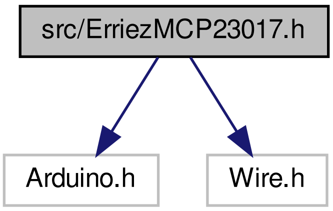
File Documentation

5.1 src/ErriezMCP23017.h File Reference

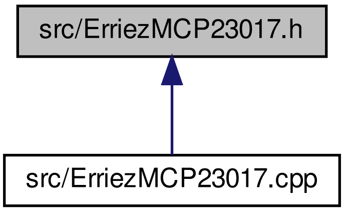
MCP23017 I2C IO expander library for Arduino.

```
#include <Arduino.h>
#include <Wire.h>
```

Include dependency graph for ErriezMCP23017.h:



This graph shows which files directly or indirectly include this file:



Classes

- class ErriezMCP23017

  Erriez MCP23017 I2C IO-Expander class.

Macros

- #define MCP23017_I2C_ADDRESS 0x20

  Default MCP23017 I2C address.
- #define MCP23017_REG_IODIR 0x00

  Controls the direction of the data I/O for port A.
- #define MCP23017_REG_IPOL 0x02

  Configures the polarity on the corresponding GPIO port bits for port A.
- #define MCP23017_REG_GPINTEN 0x04

  Controls the interrupt-on-change for each pin of port A.
- #define MCP23017_REG_DEFVAL 0x06

  Controls the default comparaison value for interrupt-on-change for port A.
- #define MCP23017_REG_INTCON 0x08

  Controls how the associated pin value is compared for the interrupt-on-change for port A.
- #define MCP23017_REG_IOCON 0x0A

  Configuration register A.
- #define MCP23017_REG_GPPU 0x0C

  Controls the pull-up resistors for the port A pins.
- #define MCP23017_REG_INTF 0x0E

  Reflects the interrupt condition on the port A pins.
- #define MCP23017_REG_INTCAP 0x10

  Captures the port A value at the time the interrupt occured.
- #define MCP23017_REG_GPIO 0x12

  Reflects the value on the port A.
- #define MCP23017_REG_OLAT 0x14

  Provides access to the port A output latches.
- #define MCP23017_NUM_REGS 0x16

  Total number of registers.
- #define MCP23017_NUM_PINS 16

  Total number of pins port A + B.
- #define MCP23017_MASK_ALL_PINS 0xFFFF

  All 16-pins mask.
- #define MCP23017_MASK_REG_A 0x1E

  Address mask to select A registers on even addresses.
- #define IOCON_BANK 7

  Controls how the registers are addressed.
- #define IOCON_MIRROR 6

  INT Pins Mirror bit.
- #define IOCON_SEQOP 5

  Sequential Operation mode bit.
- #define IOCON_DISSLW 4

  Slew Rate control bit for SDA output.
- #define IOCON_ODR 2

  Configures the INT pin as an open-drain output.
- #define IOCON_INTPOL 1

  This bit sets the polarity of the INT output pin.
- #define REG_IOCON_VALUE

  Default MCP23017 configuration.

5.1.1 Detailed Description

MCP23017 I2C IO expander library for Arduino.

Source: https://github.com/Erriez/ErriezMCP23017 Documentation: https://erriez.github.io/ErriezMCP23017

Design notes:

1 This library is designed for MCP23017 with I2C interface.

2 This library does not support the MCP23S017 with SPI interface. Workaround: None, use another library, or add SPI support to this library.

3 The INTB is not enabled in this library, because. INTA and INTB interrupts are ORed to INTA with configuration bit MIRROR=1 in IOCON register. Workaround: The application shall only use INTA.

4 Port, direction and pull-up states are cached in variables for speed: No additional register reads are needed.

MCP23017 limitations:

1 The MCP23017 does not support rising or falling edge interrupts. Workaround: None: The MCP23017 support only the following interrupts:

- Pin change (Recommended: Generates interrupt on pin level change)
- Level high (Generates continues interrupts when pin is high)
- Level low (Generates continues interrupts when pin is low)

2 Note: The register IODIR bits are reversed: 0: Output 1: Input Workaround: Be careful with interpreting register IODIR.

MCP23017 major bugs:

1 The INTA pin is released when reading from register GPIO or INTCAP. This happens when the application calls function pinRead() or portRead(). This is a documented chip limitation. Workaround: None.

2 Register INTF captures only one pin change. The MCP23017 does not update register INTF when multiple interrupts occurs at the same time. Workaround:

1. The application shall read the INTF register with function getPortIntertuptStatus() as fast as possible after INTA is asserted. otherwise multiple pin interrupts are lost.
2. The application shall poll pin INTA and read GPIO pin manually to detect pin changes.

5.1.2 Macro Definition Documentation

5.1.2.1 REG_IOCON_VALUE

```
#define REG_IOCON_VALUE
```

Value:

```
((0<<IOCON_BANK) |      /* 1 = The registers associated with each port are separated into
       different banks. */ \
     (1<<IOCON_MIRROR) |    /* 1 = INTA and INTB pins are OR'ed to INTA (INTB disabled) */ \
     (0<<IOCON_SEQOP) |     /* 1 = Sequential operation disabled, address  pointer does not
       increment. */ \
     (0<<IOCON_DISSLW) |    /* 1 = Slew rate disabled */ \
     (0<<IOCON_ODR) |       /* 1 = Open-drain output (overrides the INTPOL bit.) */ \
     (0<<IOCON_INTPOL))     /* 1 = Active-high */
```

Default MCP23017 configuration.

Definition at line 124 of file ErriezMCP23017.h.

Index