MCP23S17 - 16-Bit I/O Expander with Serial Interface

Page last updated 22 Oct 2014 (22 Oct 2014), by (/users/Shizler/) wes adams (/users/Shizler/). 0 replies (/users/Shizler/notebook/mcp23s17---16-bit-io-expander-with-serial-interfac/#commentform) I/O (/search/?q=I/O&type=Notebook), MCP23S17 (/search/?q=MCP23S17&type=Notebook), SPI (/search/?q=SPI&type=Notebook)

Chip Overview

The mbed LPC1768 chip has 26 pins used for digital input and output. All of these pins have other hardware associated with them, so space needed for all-purpose input and output pins can dwindle quickly.

From MCP23S17 datasheet http://ww1.microchip.com/downloads/en/DeviceDoc/21952b.pdf (http://ww1.microchip.com/downloads/en/DeviceDoc/21952b.pdf)

The MCP23S17 device family provides 16-bit, general purpose parallel I/O expansion for SPI applications. The MCP23X17 consists of multiple 8-bit configuration registers for input, output and polarity selection. The system master can enable the I/Os as either inputs or outputs by writing the I/O configuration bits (IODIRA/B). The data for each input or output is kept in the corresponding input or output register. The polarity of the Input Port register can be inverted with the Polarity Inversion register. All registers can be read by the system master. The 16-bit I/O port functionally consists of two 8-bit ports (PORTA and PORTB). . There are two interrupt pins, INTA and INTB, that can be associated with their respective ports, or can be logically OR'ed together so that both pins will activate if either port causes an interrupt. The interrupt output can be configured to activate under two conditions (mutually exclusive): 1. When any input state differs from its corresponding Input Port register state. This is used to indicate to the system master that an input state has changed. 2. When an input state differs from a preconfigured register value (DEFVAL register). The Interrupt Capture register captures port values at the time of the interrupt, thereby saving the condition that caused the interrupt. The Power-on Reset (POR) sets the registers to their default values and initializes the device state machine. The hardware address pins are used to determine the device address.

Registers

Register Name	Address (hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	POR/RST value
IODIRA	00	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IODIRB	01	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IPOLA	02	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
IPOLB	03	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
GPINTENA	04	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
GPINTENB	05	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
DEFVALA	06	DEF7	DEF6	DEF5	DEF4	DEF3	DEF2	DEF1	DEF0	0000 0000
DEFVALB	07	DEF7	DEF6	DEF5	DEF4	DEF3	DEF2	DEF1	DEF0	0000 0000
INTCONA	08	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	0000 0000
INTCONB	09	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	0000 0000
IOCON	0A	BANK	MIRROR	SEQOP	DISSLW	HAEN	ODR	INTPOL	—	0000 0000
IOCON	0B	BANK	MIRROR	SEQOP	DISSLW	HAEN	ODR	INTPOL	—	0000 0000
GPPUA	0C	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
GPPUB	0D	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
INTFA	0E	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INTO	0000 0000
INTFB	0F	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INTO	0000 0000
INTCAPA	10	ICP7	ICP6	ICP5	ICP4	ICP3	ICP2	ICP1	ICP0	0000 0000
INTCAPB	11	ICP7	ICP6	ICP5	ICP4	ICP3	ICP2	ICP1	ICP0	0000 0000
GPIOA	12	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
GPIOB	13	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
OLATA	14	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000
OLATB	15	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000

Register Descriptions

MCP23S17 - 16-Bit I/O Expander with Serial Interface

General information and API for MCP23S17 chip

I/O (/search/?q=I/O&type=Notebook), MCP23S17 (/search/?q=MCP23S17&type=Notebook), SPI (/search/?q=SPI&type=Notebook)

Page owner: (/users/Shizler/) wes adams (/users/Shizler/)

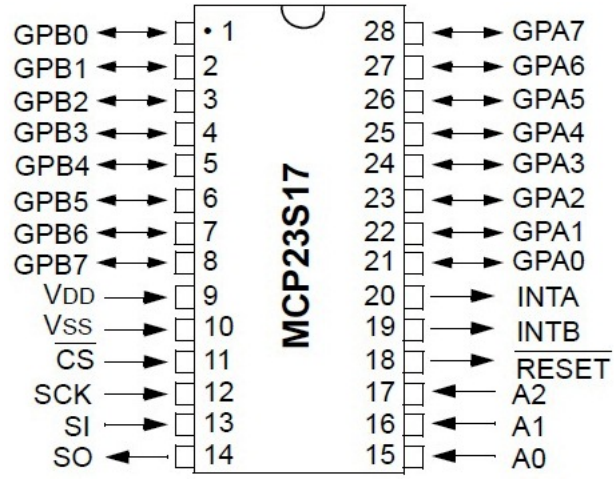
Created 21 Oct 2014 (21 Oct 2014).

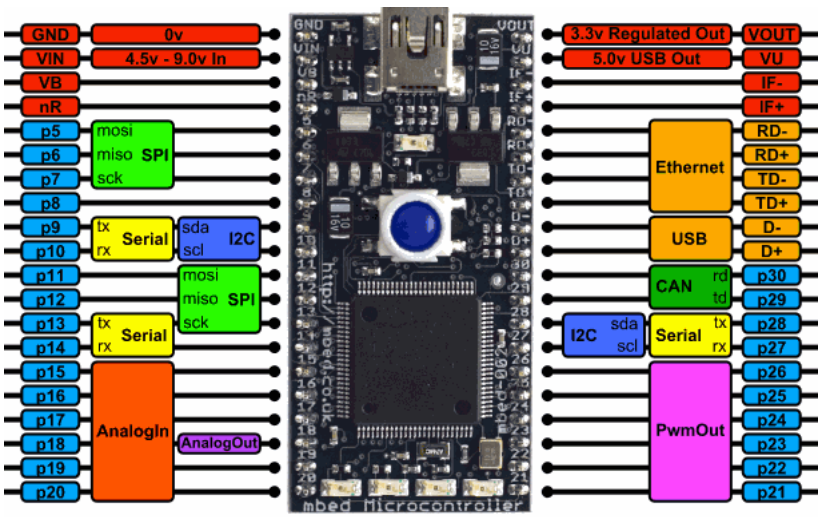
Last updated 22 Oct 2014 (22 Oct 2014)

IODIRA/B	Bits set '1' become inputs. Bits set '0' become outputs.
IPOLA/B	Bits set '1' will reflect opposite logic state of input pin. Bits set '0' will reflect same logic state of input pin.
GPINTENA/B	Bits set '1' cause corresponding pin to be enabled for interrupt-on-change(IOC). Bits set '0' disable IOC for corresponding pin. **DEFVAL and INTCON registers must be configured to enable IOC.
DEFVALA/B	This register holds value used for comparison if IOC is enabled.
INTCONA/B	Bits set '1' are compared against corresponding bit value in DEFVALA/B for IOC. Bits set '0' are compared against pin's previous value for IOC.
IOCON	IOCON[7] - Set '0' for all purposes of this demo and documentation.
	IOCON[6] - MIRROR - '1' - INTA and INTB are essentially "or'ed" change in one is change in both. - '0' - INTA corresponds only to PortA and vica versa.
	IOCON[5] - SEQOP - '1' - Address pointer does not increment - '0' - Address pointer does increment
	IOCON[4] - DISSLW - '1' - Slew rate disabled - '0' - Slew rate enabled
	IOCON[3] - HAEN - '1' - Enables address pins - '0' - Disables address pins
	IOCON[2] - ODR - '1' - Open-drain output - '0' - Active driver output
	IOCON[1] - INTPOL - '1' - Sets polarity of INT pin to active-high - '0' - Sets polarity of INT pin to active-low
	INTCON[0] - Unused - '0'
GPPUA/B	Bits set '1' are set as input, and corresponding port pin is internally pulled up with 100k resistor Bits set '0' disables pull up
INTFA/B	Read-only register - Pins read as '1' indicate that pin caused interrupt
INTCAPA/B	Read-only register - Updates only when interrupt occurs. Captures value of GPIO port at time of interrupt. Register remains unchanged until interrupt is cleared via read of INTCAP or GPIO
GPIOA/B	This register simply reflects the values on the port.
OLATA/B	Writing to this register updates pin values that are set as outputs.

Pinout Guide

From MCP23S17 datasheet http://ww1.microchip.com/downloads/en/DeviceDoc/21952b.pdf (http://ww1.microchip.com/downloads/en/DeviceDoc/21952b.pdf)





Wiring guide for demo code

```
MCP23S17 -> mbed

SI        -> p5
SO        -> p6
SCK       -> p7
CS        -> p20
Vss       -> 3V
Vdd       -> GND
A0        -> GND //A0, A1, A2 of the MCP23S17  are tied to ground on the breadboard, so the 8-bit address for
A1        -> GND
A2        -> GND

GPA0      -> GPB0 //This will allow testing by writing to A0 from the mbed and then reading B0

```

Program Files

MCP23S17

Import program (https://os.mbed.com/ide/import-program//users/romilly/code/MCP23S17/)

(/users/romilly/code/MCP23S17/)

v 0.4

Last commit 28 Aug 2010 (28 Aug 2010) by  *(/users/romilly/) Romilly Cocking (/users/romilly/)*

Sample Code

```
This code will ensure that the chip is working while giving you some insight into the use of the MCP23S17 clas
and several of the functions within.

This will alternate writing '1' and '0' to PORT_A[0]
PORT_A[0] is hardwired to PORT_B[0]
Therefore, reading in PORT_B[0] and letting that value control an LED on the mbed chip should result in a blir
```

This demo code taken from http://developer.mbed.org/users/4180_1/code/MCP23S17_Basic_IO_Demo/ (http://developer.mbed.org/users/4180_1/code/MCP23S17_Basic_IO_Demo/)

Simple demo

```
/* MCP23S17 - drive the Microchip MCP23S17 16-bit Port Extender using SPI
* Copyright (c) 2010 Romilly Cocking
* Released under the MIT License: http://mbed.org/license/mit (http://mbed.org/license/mit)
*
* version 0.4
*/
#include "mbed.h"
#include "MCP23S17.h"
// Create SPI bus
SPI spi(p5, p6, p7);
//

char Opcode = 0x40;

// Next create a MCP23S17
// mbed p20 is connected to ~chipSelect on the MCP23S17
MCP23S17 chip = MCP23S17(spi, p20, Opcode);

DigitalOut led1(LED1); // mbed LED1 is used for test status display

int main() {
//
//  Set all 8 Port A bits to output direction
    chip.direction(PORT_A, 0x00);
//  Set all 8 Port B bits to input direction
    chip.direction(PORT_B, 0xFF);
    led1=0;
//  Start Loopback test sending out and reading back values
//  loopback test uses A0 and B0 pins - so use a wire to jumper those two pins on MCP23S17 together
    while (1) {
        // write 0xAA to MCP23S17 Port A
        chip.write(PORT_A, 0xAA);
        wait(.5);
        // read back value from MCP23S17 Port B and display B0 on mbed led1
        led1 = chip.read(PORT_B)& 0x01;
        // write 0x55 to MCP23S17 Port A
        chip.write(PORT_A, 0x55);
        wait(.5);
        // read back value from MCP23S17 Port B and display B0 on mbed led1
        led1 = chip.read(PORT_B)& 0x01;
        // led1 should blink slowly when it is all working
    }
}
```

Class and Function Documentation

(/users/Shizler/code/MCP23S17/docs/a461ce77c9a0/classMCP23S17.html)

Import library (https://os.mbed.com/ide/import-program//users/Shizler/code/MCP23S17/;mode:lib)

(/users/Shizler/code/MCP23S17/docs/a461ce77c9a0/classMCP23S17.html)MCP23S17 - MCP23S17 Class Reference (/users/Shizler/code/MCP23S17/docs/a461ce77c9a0//classMCP23S17.html)

Public Member Functions

	MCP23S17 (classMCP23S17.html#a97c1e318d8cdd61f6fbf16985159f334) (SPI &spi, PinName ncs, char writeOpcode) MMCP23S17.
void	direction (classMCP23S17.html#a645c4d9bc84abf0d7af0a539e220ff5c) (Port port, char direction) Pass PORT_A or PORT_B in for port; direction is 8 bit value to set bits in port; '0' - Output, '1' - Input; Updates IODIRA/B.
void	configurePullUps (classMCP23S17.html#a93fd6e7d3db433199a35b46834c6056f) (Port port, char offOrOn) Pass PORT_A or PORT_B in for port; offOrOn is 8 bit value to set bit in port; '1' - Pullup enabled, '0' - Pullup disabled; Updates GPPUA/B.
void	interruptEnable (classMCP23S17.html#a4a993e1561504b3de2534d90c0698c55) (Port port, char interruptsEnabledMask) Pass PORT_A or PORT_B in for port; interruptsEnabledMask represents 8 bits in the GPINTEN register; each bit that is set to '1', then it is enabled for Interrupt-on-change; NOTE: defaultValue AND interruptControl MUST BE SETUP FOR THIS TO WORK.
void	interruptPolarity (classMCP23S17.html#a17e6f7175f80762edb03bd30177169a7) (Polarity polarity) polarity is 8-bit value used to update IPOLA/B.
void	mirrorInterrupts (classMCP23S17.html#a486c55a53b6874ac9fc5053c74489256) (bool mirror) Passing TRUE into this funtion causes an interrupt on either port to cause both pins on both ports to activate; passing in a FALSE causes an interrupt on any port to only activate its respective pin; Updates IOCON[6].
void	defaultValue (classMCP23S17.html#af628a3d725147437ca6103be29d56167) (Port port, char valuesToCompare) Pass PORT_A or PORT_B in for port; valuesToCompare represents 8 bits in either DEFVAL register; these registers will be used by their respective ports for an interrupt-on-change setting i.e.
void	interruptControl (classMCP23S17.html#af8ba701e13df9091a1f6a231a26856d3) (Port port, char interruptContolBits) Pass PORT_A or PORT_B in for port; interruptContolBits represents 8 bits in INTCON register; '1' - the corresponding I/O pin is compared against the associated bit in the DEFVAL register, '0' - bit value is compared against its previous value.
char	read (classMCP23S17.html#a9a9d8d05a34cdc783874975b2692bf44) (Port port) Pass PORT_A or PORT_B in for port; Returns 8 bits from port.
void	write (classMCP23S17.html#a143de248f59204f9433a56e3c2df7525) (Port port, char byte) Pass PORT_A or PORT_B in for port; Writes 8 bits of byte to port.

Please log in (/account/login/?next=/users/Shizler/notebook/mcp23s17---16-bit-io-expander-with-serial-interfac/) to post comments.