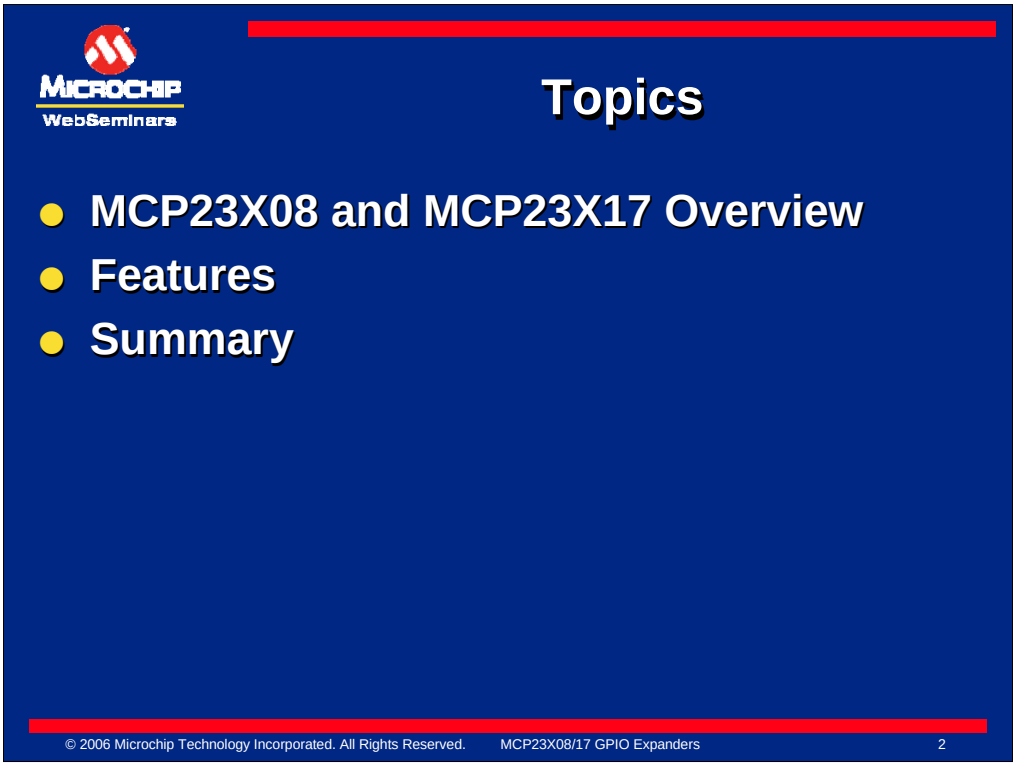
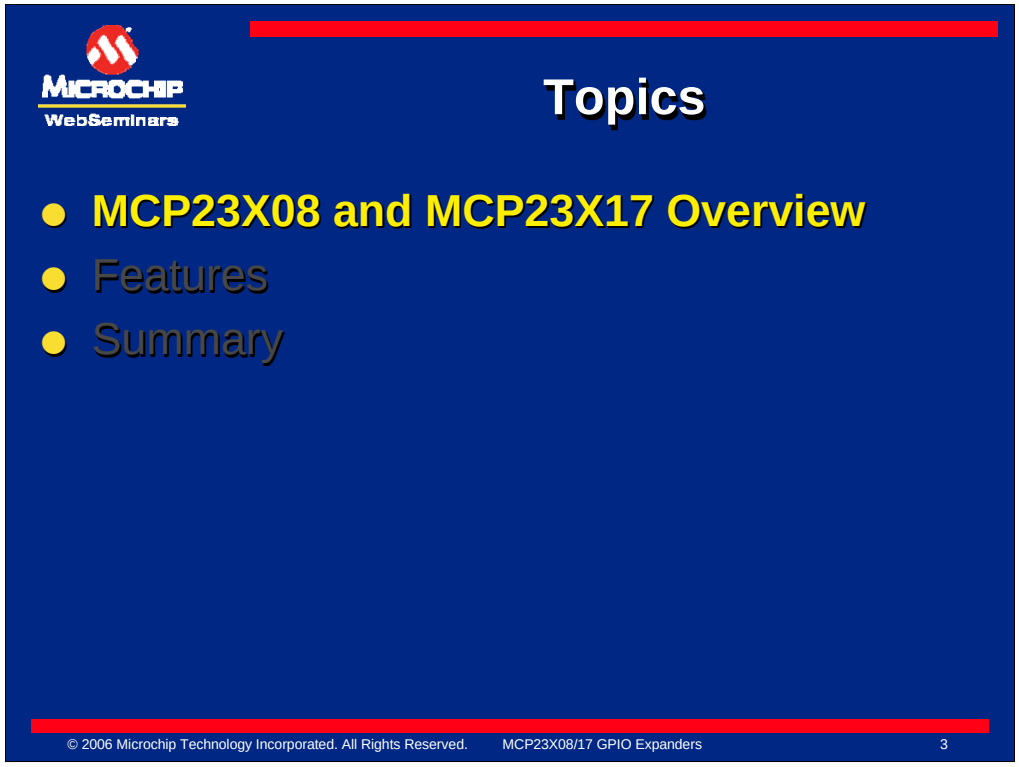
Hello and welcome the WebSeminar on the unique features of Microchip's GPIO Expanders. My name is Pat Richards. I am an applications engineer in the Analog and Interface Products Division. So let's get started.

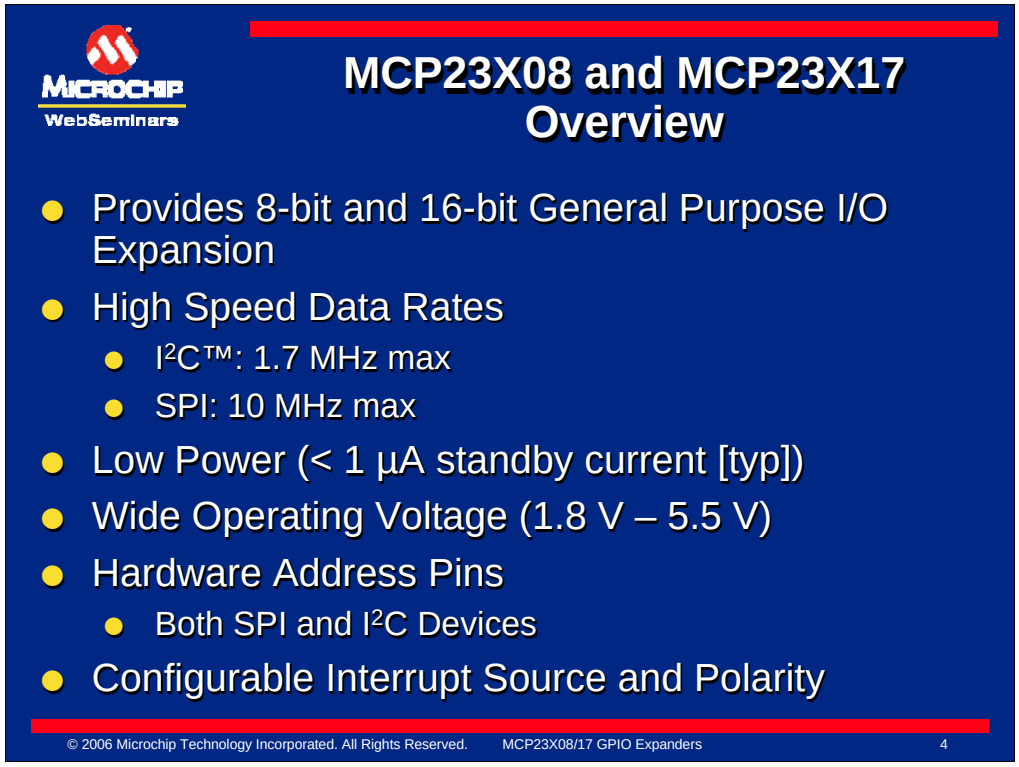


Today we will be discussing the unique features of the MCP23X08 and the MCP23X17 GPIO Expanders.

This family consists of 8-bit and 6-bit devices both having I^2C™ and SPI interface offerings.

The discussion will begin with an overview of the family, we will then discuss the features of the family including some unique features which are designed to make the devices easy to use and add flexibility so that they can be customized for streamlined systems integration. Some of the features we will discuss are port description, 8-bit and 16-bit modes of operation, the interrupt features such as polarity, sources, capturing, mirroring, etc. and using or controlling the internal address pointer, and finally, address pins on the SPI devices.


MICROCHIP
WebSeminars
Topics
MCP23X08 and MCP23X17 Overview
Features
Summary
© 2006 Microchip Technology Incorporated. All Rights Reserved.
MCP23X08/17 GPIO Expanders
3



The MCP23X08 family, consists of the MCP23008 which has an I^2C interface, and the MCP23S08 which has an SPI interface, which provide 8-bit I/O expansion.

The MCP23X17, consists of the MCP23017 which has an I^2C interface, and the MCP23S17 which has an SPI interface, and these provide 16-bit I/O expansion.

Both families are otherwise basically the same except where noted in later slides. The MCP23X17 family has features which apply only to 16-bit devices.

The I^2C version as denoted by the zero (0) in the part number includes the MCP23008 and the MCP23017, and it's specified for 1.7 MHz max. I^2C.

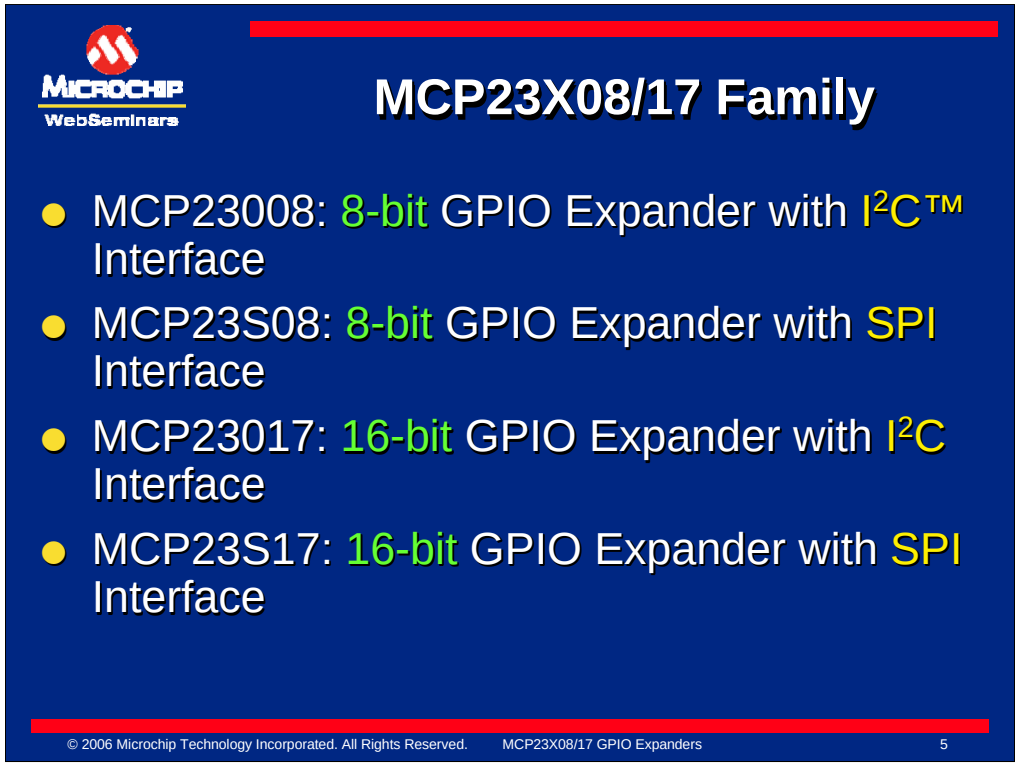
The SPI version as denoted by an "S" in the part number is specified for 10 MHz max.

The standby current is very low with a typical current of <1 µA, and the voltage range is from 1.8 V to 5.5 V covering the vast majority of applications.

It is well known that I^2C slave devices many times have hardware address pins which allow for multiple devices with the same opcode to share the bus. The SPI versions also have address pins to allow multiple devices to share the bus using only one chip select pin.

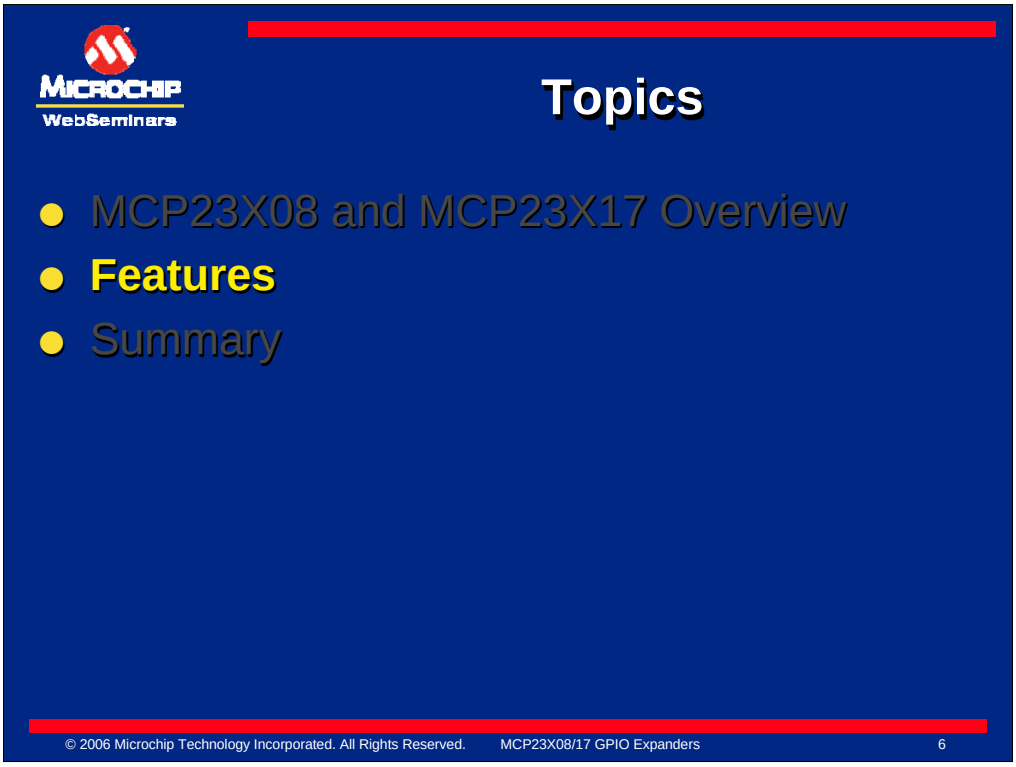
Each I/O pin (configured as an input) can be individually enabled and configured to interrupt either when the pin changes state from a previous value, or when the state of the pin does not match a preconfigured value in a register (DEFVAL).

The polarity of the INT pin is configurable for push-pull active high or push-pull active low, or as open drain (active low).

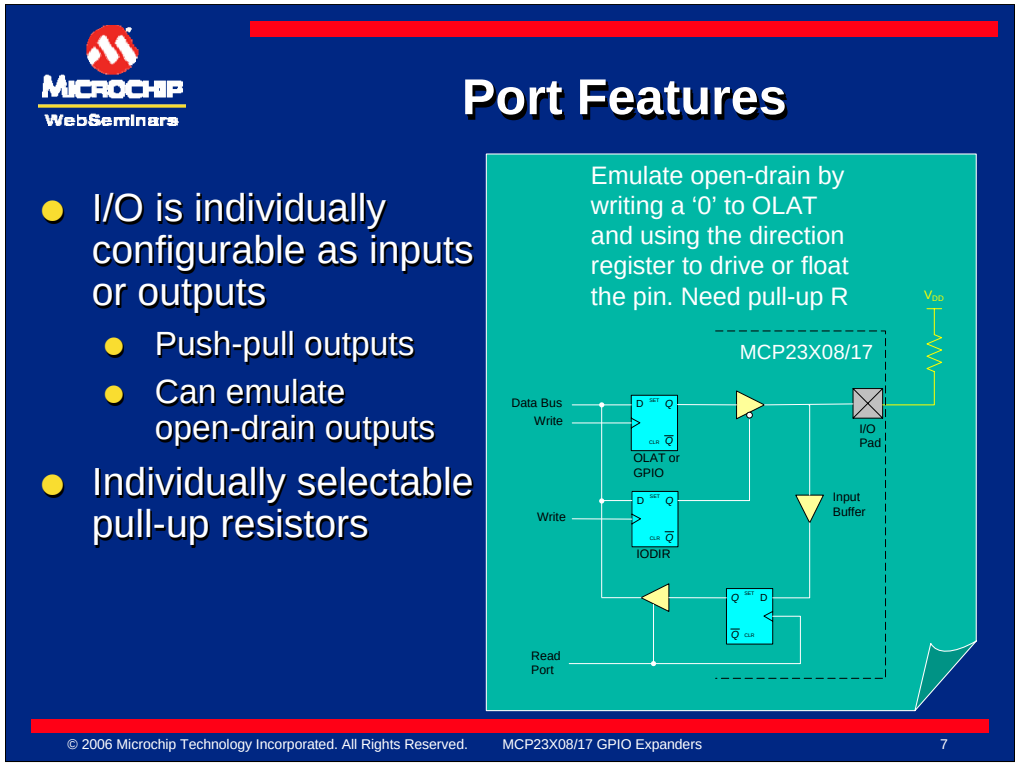


This slide shows the devices in the family.

The MCP23X08 devices are 8-bit GPIO expanders and the MCP23X17 devices are 16-bit GPIO expanders. The “0” in the part number denotes the device having an I^2C interface and the “S” in the part number denotes the device having an SPI interface.



Now let’s take a look at some of the features of Microchip’s GPIO expanders.



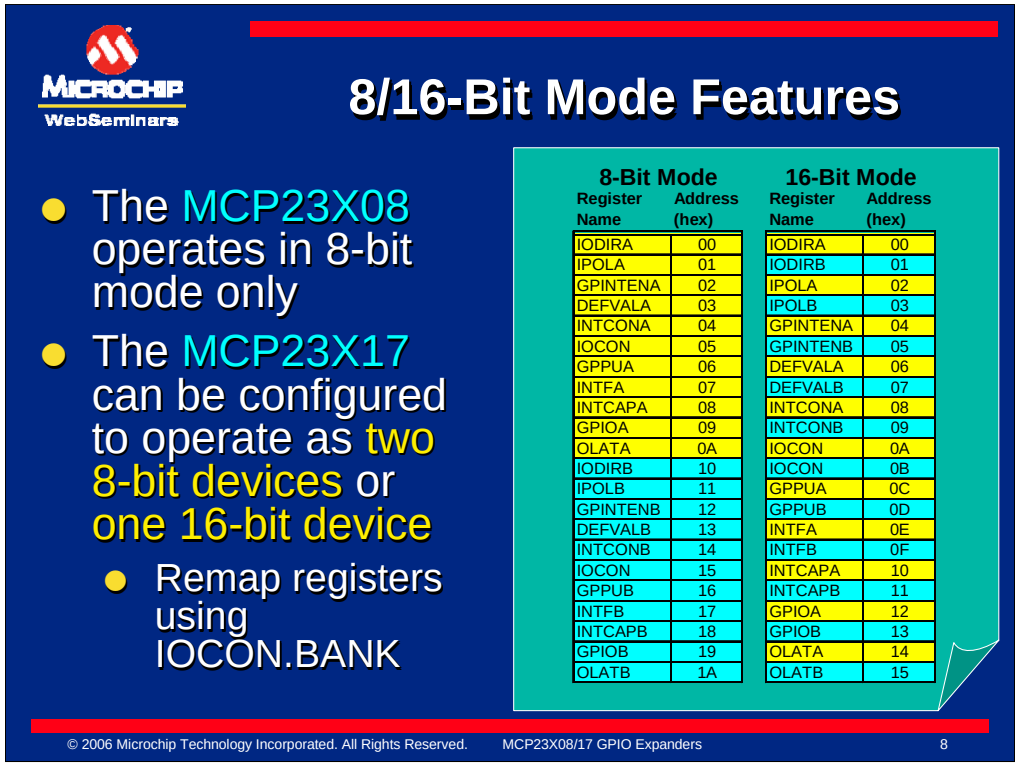
Each I/O consists of a pad, weak pull-up resistor, input buffer and output latch. Each I/O can be individually configured as input or output – the default on power up or reset is input.

Writing to GPIO register actually writes to the OLAT register.

The GPIO register always reflects the state on the pin, regardless of whether it is configured as an input or output. And when configured as an output, the port is in a push-pull configuration.

However, the device can also emulate open-drain configuration by clearing the output latch bit and using the pin direction register (IODIR) to put the logic states on the pin. This requires a pull-up resistor on the pin. Clear latch pin so a '0' will be driven when pin is configured as a output, and the pin will float high when pin is an input and drive low when pin is an output. That's because the output latch is cleared to a zero.

Each port has individually selectable weak pull-up resistors which are not shown in the drawing, but there are actually weak pull-up resistors internally on this device also.



This feature only applies to the MCP23X17. The MCP23X08 is an 8-bit device only and does not apply to this feature.

The MCP23X17 has the unique ability to appear to the MCU as either two (2) 8-bit GPIO expanders, or as a single 16-bit GPIO expander.

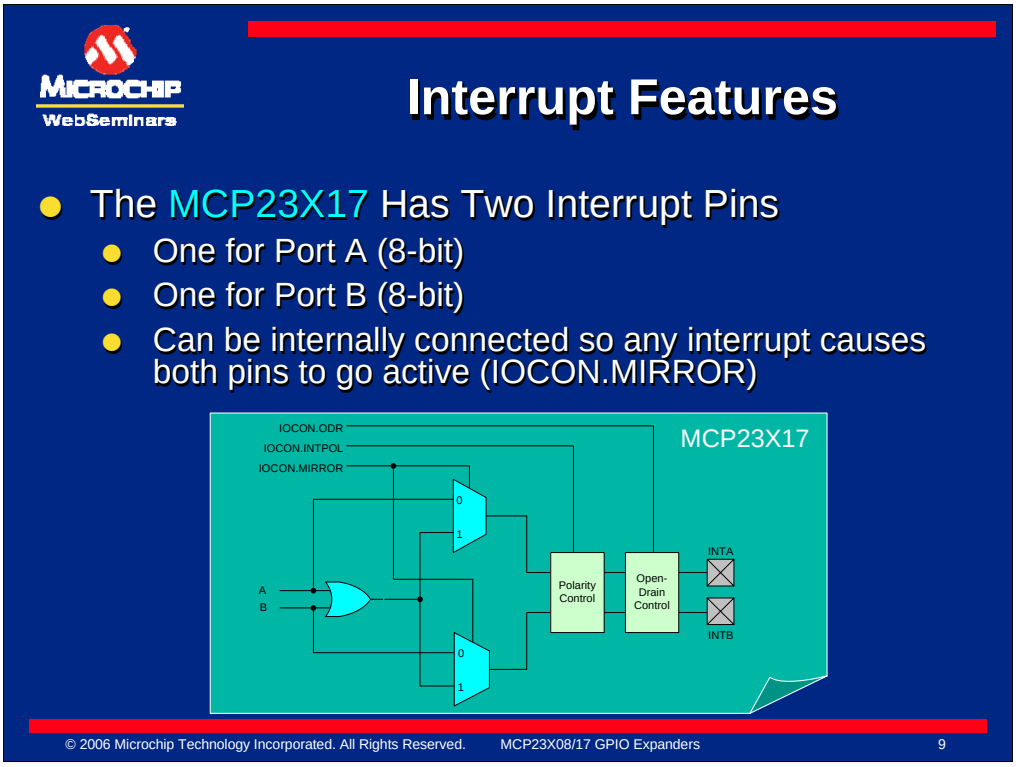
This is accomplished by splitting the 16 I/O into two separate 8-bit I/O ports (Port A and Port B).

Each port has a group of dedicated registers. The table shows how the registers are organized into two modes. The yellow registers are associated with Port A, and the cyan or blue registers are associated with Port B.

When in 8-bit mode, the ports' registers are separated : Port A register addresses range from 00h – 0Ah; and Port B register addresses range from 10h – 1Ah

When in 16-bit mode, the ports' registers are interleaved to emulate 16-bit wide registers: Port A and Port B register addresses range from 00h – 15h.

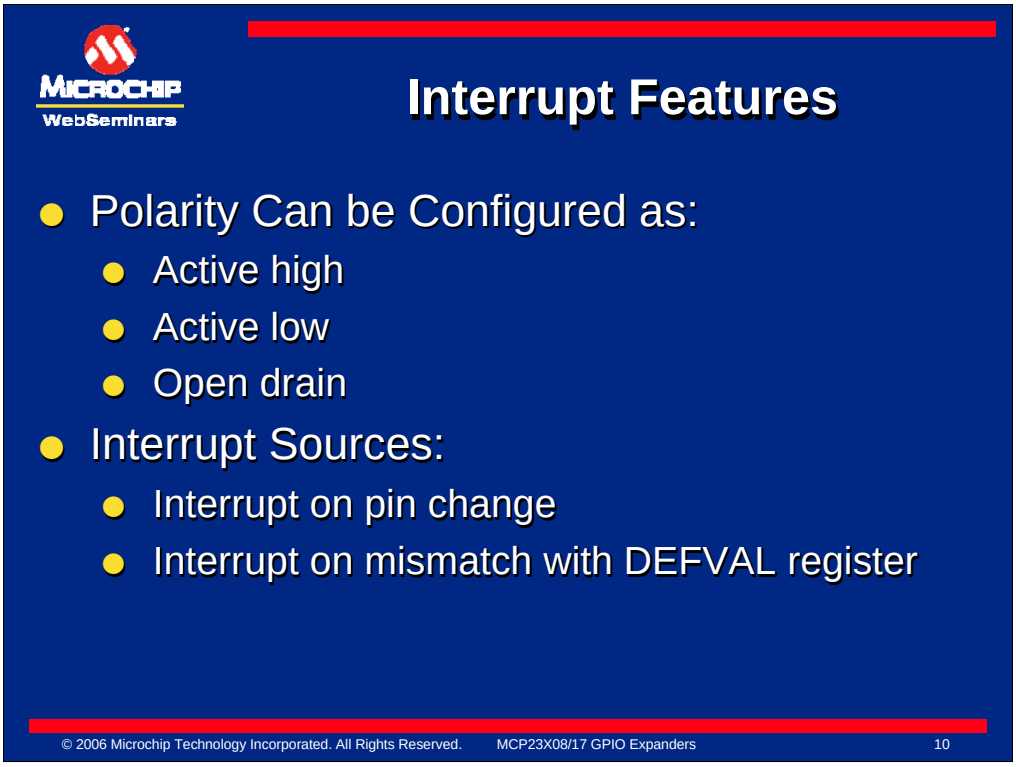
The registers associated with PortA are at even registers and the registers associated with PortB are at odd registers. Therefore, to address a port pair, the master device would send an even address to the slave.



The MCP23X17 which is the 16-bit device has two interrupt pins, one for PortA and one for PortB.

The two interrupt pins allow the application to easily differentiate between input conditions on either port.

The device can also be configured so that both interrupt pins go active when any interrupt condition occurs on either port.



The devices have very flexible interrupt features.

The interrupt polarity can be configured to one of three active states. Active High, Active Low or Open Drain. Note, the polarity is the same for both interrupt pins on the MCP23X17 device.

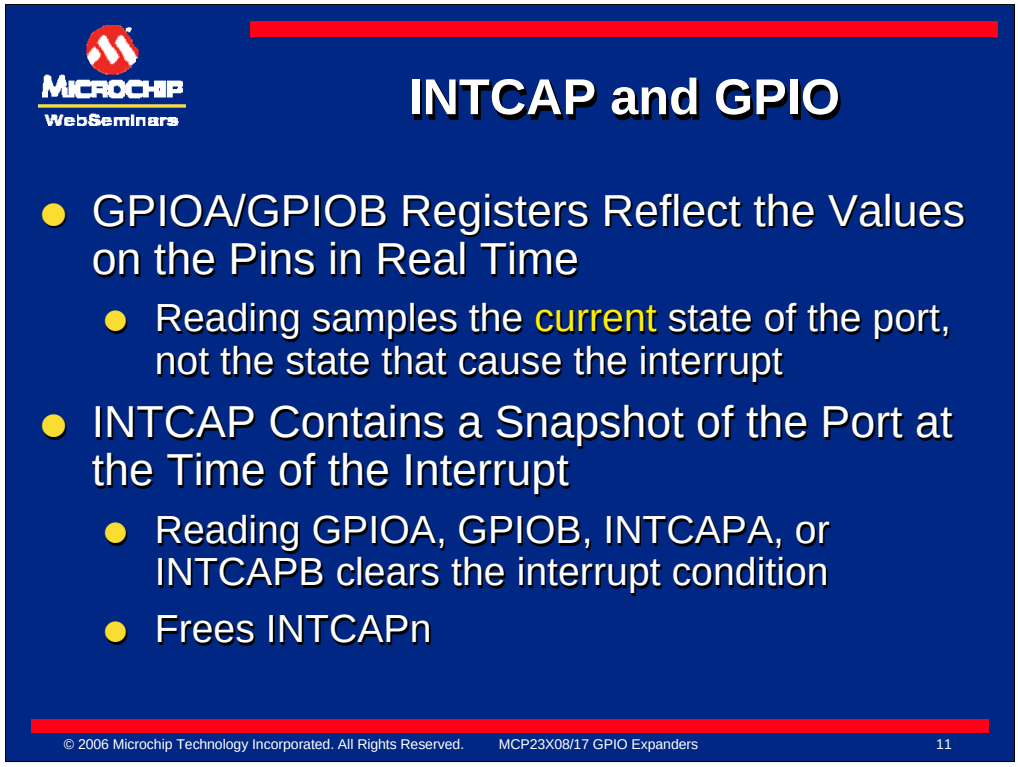
Each pin configured as an input can be individually configured to cause interrupt for two different interrupt conditions.

The first interrupt condition is an interrupt when the pin changes states. For example, if no interrupt is pending and an input changes states, an interrupt will occur. At this point, the interrupt enable is disabled until the condition that caused the interrupt is cleared. This implies that further changes on input(s) will not cause subsequent interrupts.

The other one is an interrupt when input level is opposite from the bit value in associated register which is internal in the device. This is the DEFVAL register. Again, the interrupt enable is disabled until the interrupt condition is cleared.

To clear the interrupt condition. In case number 1, interrupt when the pin changes states, reading either GPIO register, which is the Port, or INTCAP register which is the Capture register, will cause the interrupt to clear and the new pin default to be set. For example, if an input was in logic 1 and a logic 0 was applied causing an interrupt, if either GPIO or INTCAP is read while the pin is low, the interrupt will clear and the new default value on the pin is logic 0.

In the other case, interrupt when input level is opposite from the default value in the DEFVAL register, reading either GPIO or the INTCAP will also clear the interrupt, however, only if the pin that caused the interrupt changed back to its initial state. That is, if it is the same as what was in the DEFVAL register.



There are two methods to determine the state of the ports.

The first, the GPIO registers always reflect the current state on the port in real-time. And second, the INTCAP registers reflect the state of the port when the interrupt occurred.

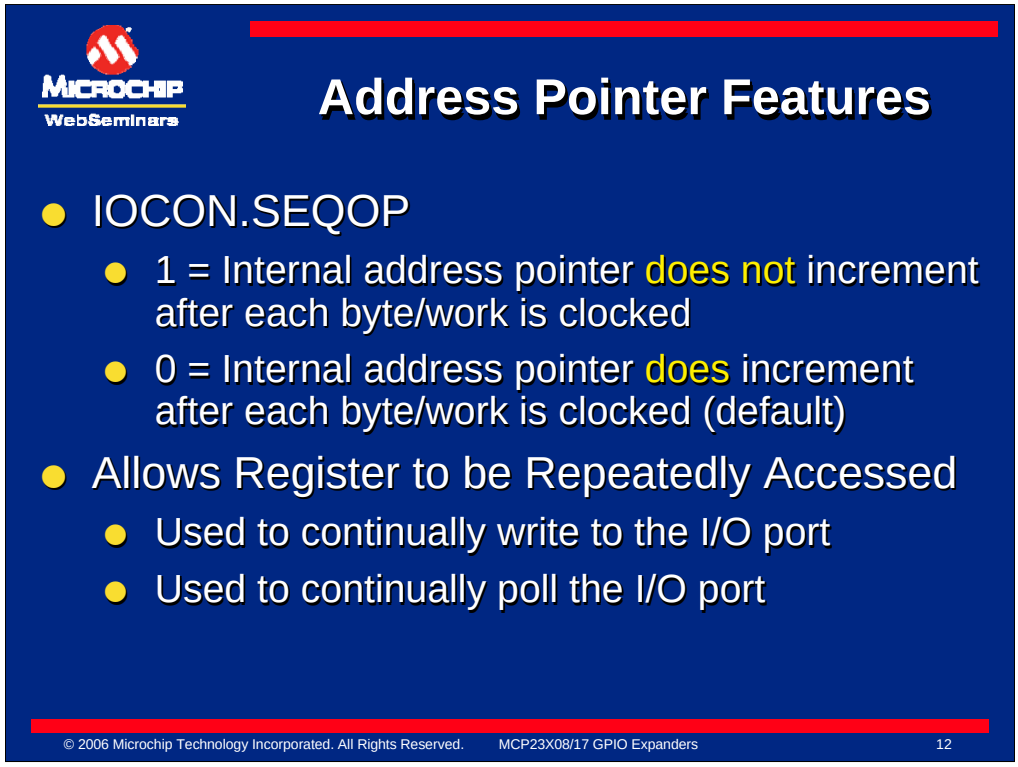
The Port registers always reflect the current value on the pin and the interrupt capture (INTCAP) registers contains a snapshot of the port at the time of the interrupt.

The guidelines on which register to read.

- Read the GPIO register(s), which is the Port register, when interested in the port value in real-time.
- Read the capture register (INTCAP) when interested in the state of the port at the time of the interrupt.

As mentioned earlier, the interrupt condition is cleared by reading either the port (GPIO) or the capture register (INTCAP).

Clearing the interrupt also frees INTCAP to capture another interrupt.



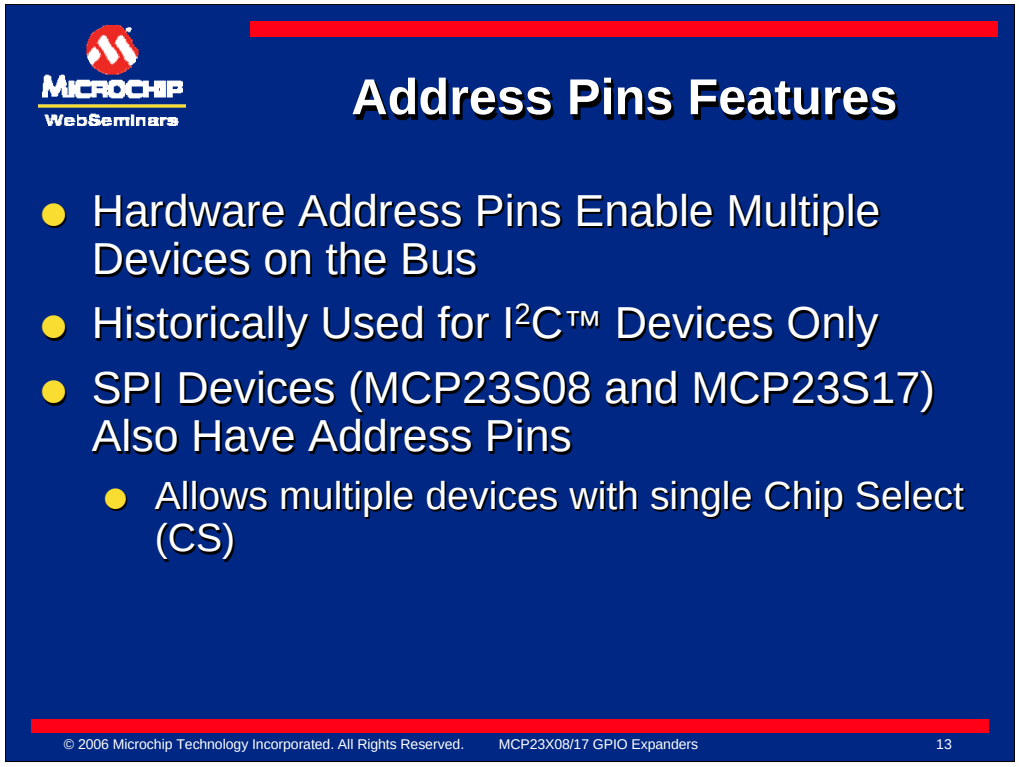
Some slave serial devices automatically increment their internal address pointer after each byte is clocked by the master. This allows the master to sequentially access multiple registers without resending the write or read command.

Other slave devices do not automatically increment their internal address pointer.

The MCP23x08 and the MCP23x17 devices have the ability to do either by configuring a control bit.

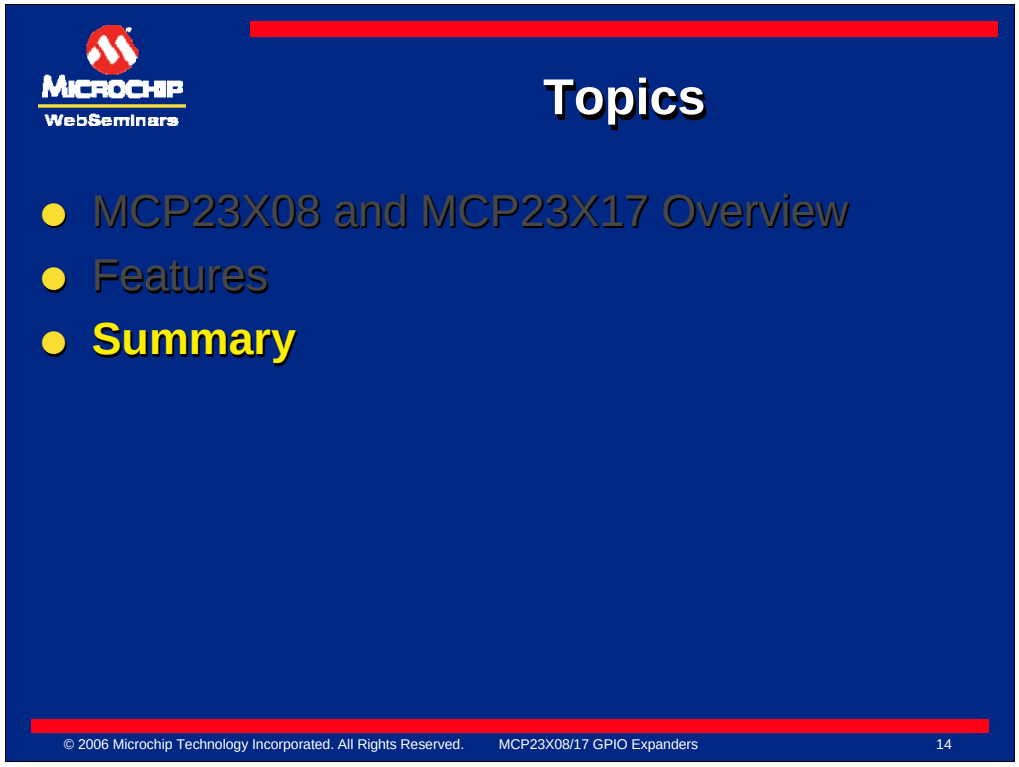
On power-up, the device may be quickly configured by sequential writing to all of the registers, and then during normal operation, it may be desirable to turn off the automatic address incrementing so the I/O port can be continually written or read.

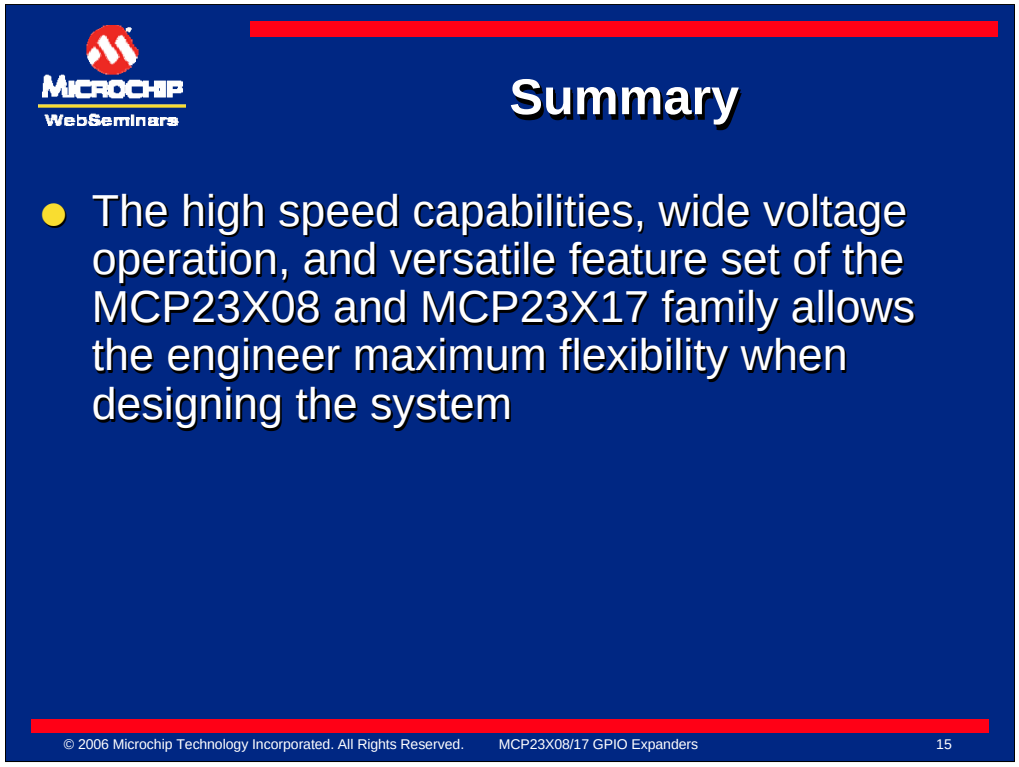
Note, when the MCP23X17 (or the 16-bit device) is in 16-bit mode, that is the A-B addresses are paired. PortA and PortB are paired, and the address incrementing is turned off, the address pointer actually points to the two paired registers in a ping-pong manner. For example, the address will alternate between GPIOA and GPIOB as bytes are clocked, thereby maintaining the continual access to the full 16-bit.



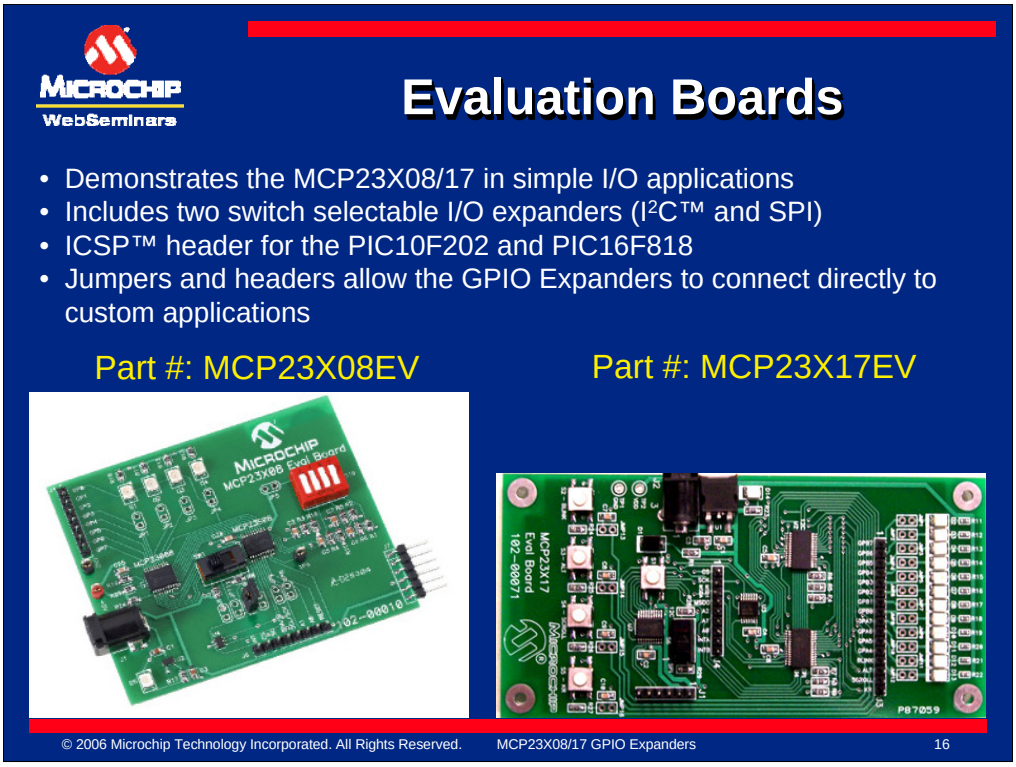
Address pins are typically only used on I^2C devices to allow multiple devices with the same base slave address to be on the same bus.

Microchip's GPIO expanders with SPI interfaces also have address pins to allow multiple devices on the same bus while using one chip select.


MICROCHIP
WebSeminars
Topics
MCP23X08 and MCP23X17 Overview
Features
Summary
© 2006 Microchip Technology Incorporated. All Rights Reserved.
MCP23X08/17 GPIO Expanders
14



The high speed capabilities, wide voltage operation, and versatile feature set of the MCP23x08 and MCP23x17 family allows the engineer maximum flexibility when designing the system.



There are two evaluation boards available at the time of this presentation.

One for the MCP23x08 family or the 8-bit family and one for the MCP23x17 or the 16-bit family. Check the Microchip website for the latest list of evaluation boards.

This concludes the presentation on Microchip's 8-bit and 16-bit GPIO expanders.

Thank you.