A product Line of
Diodes Incorporated

PI7C1401

I²C-bus/SPI to Quad Port Expander

Features

- Support Low-speed signals management and I²C aggregation across four ports
- Support multiple PI7C1401s to control high-port-count through a single host interface
- Selectable I²C(up to 1Mhz) or SPI(up to 33Mhz) host control interface
- Automatic pre-fetching of critical, user-specified data from the modules and stored into internal 32 Byte FIFO
- Advanced LED features for port status indication, including programmable blinking and dimming
- Support standard SFF-8431, SFF-8472, SFF-8436, SFF-8679, INF-8074 and other such low-speed electrical interfaces, including a dedicated 100/400Khz I²C interface to each port
- Customizable events which trigger an interrupt to the host
- Broadcast Mode allows writes to all ports simultaneously
- Separate host-side I/O voltage: 1.8V to 3.3V
- Industrial Temperature Range: -40°C to 85°C
- ESD: (HBM) 4KV, (CDM) 2KV
- Totally Lead-Free & Fully RoHS Compliant (Notes 1 & 2)
- Halogen and Antimony Free. "Green" Device (Note 3)
- Packaging (Pb-free & Green):
  - 56-ZF (TQFN) 5x11mm, 0.5mm pin pitch

Software Support

- PI7C1401 Programming Guide
- PI7C1401 Sample Code
- PI7C1401 Implementation Guide

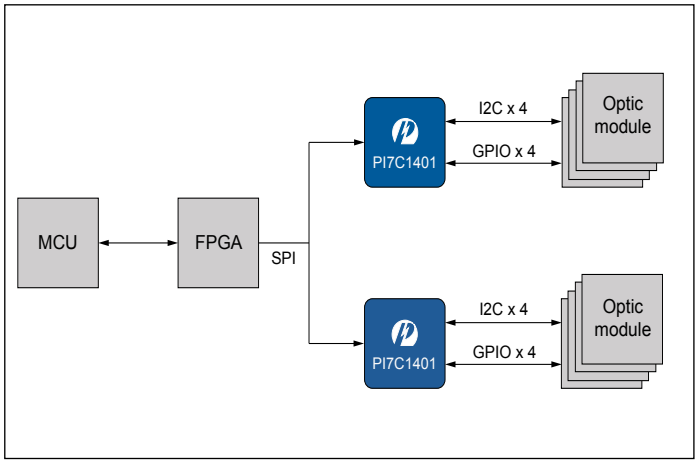
Application

- Telecom for Wireless Base Station
- Enterprise Switch/Router
- Embedded System for Video
- Storage for Data Center

Description

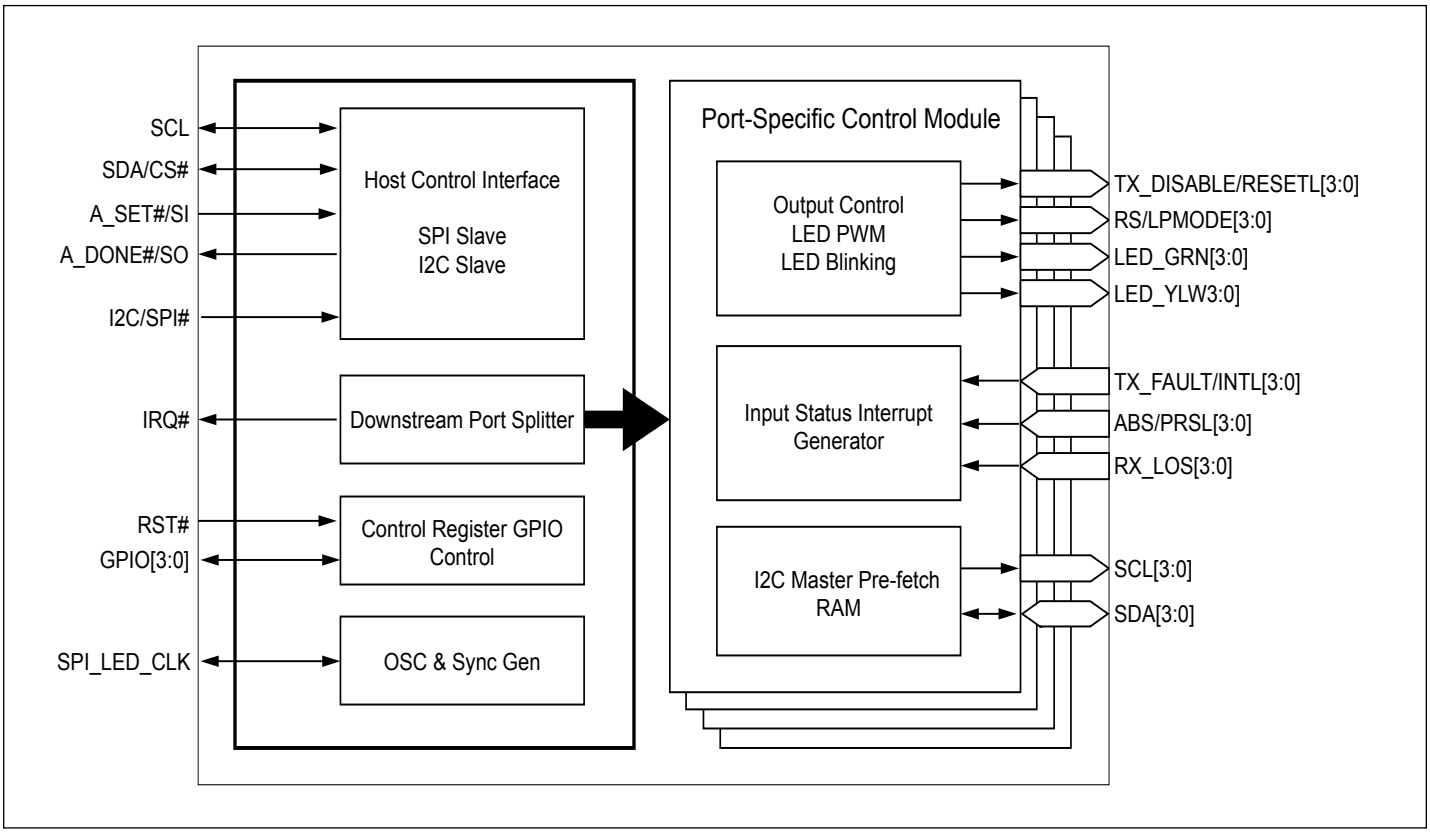
The PI7C1401 is a SPI/I²C Quad Port Expander, it's specifically designed to manage common port types such as SFP+ and QSFP+. The PI7C1401 provides an I²C/SPI interface, an interrupt output to host and four I²C interfaces, general purpose pins for low-speed control signals to the modules. The PI7C1401 can be used a wide range of applications such as Networking, Telecom, Storage, Embedded. The typical usage for this device is high volume applications requiring an "expander" to aggregate all low-speed control and I²C signals across four ports and presents a single I²C or SPI interface to host. Multiple PI7C1401s can be used in high-port-count applications with one I²C or SPI to host. The system BOM cost can be cut by using smaller IO count control devices(FPGAs, CPLDs, MCUs) and by reducing routing layer congestion.

Application Diagram

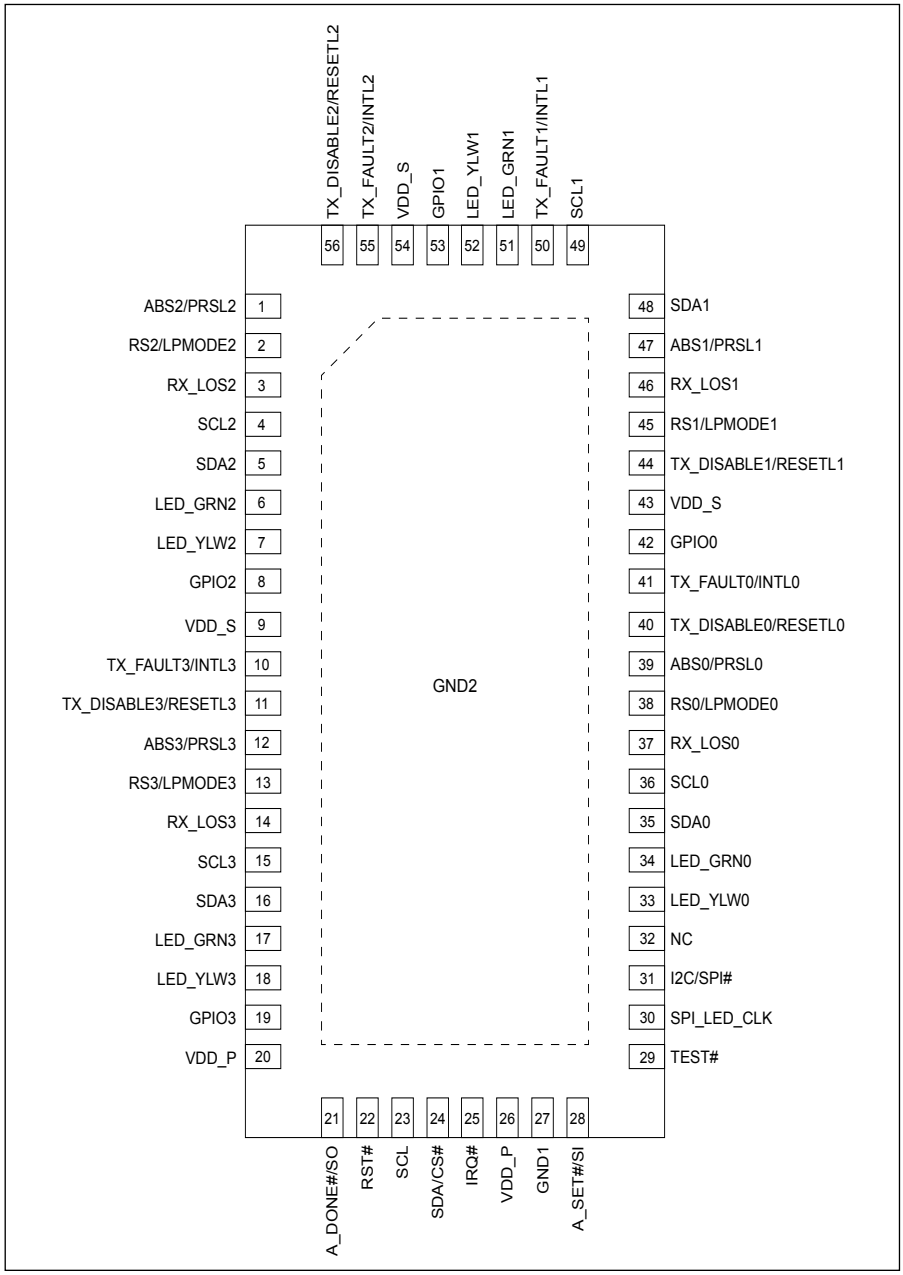


Notes:
1. No purposely added lead. Fully EU Directive 2002/95/EC (RoHS), 2011/65/EU (RoHS 2) & 2015/863/EU (RoHS 3) compliant.
2. See https://www.diodes.com/quality/lead-free/ for more information about Diodes Incorporated's definitions of Halogen- and Antimony-free, "Green" and Lead-free.
3. Halogen- and Antimony-free "Green" products are defined as those which contain <900ppm bromine, <900ppm chlorine (<1500ppm total Br + Cl) and <1000ppm antimony compounds.

Block Diagram



Pin Configuration



Pin Description

Pin Name	Pin#	Type	Description
IRQ#	25	OD	Interrupt output, active low, the events which trigger an interrupt are configurable
SCL	23	IOD	Primary I^2C-bus or SPI serial input clock
SDA/CS#	24	IOD	Primary I^2C-bus data input/output(open-drain) if I^2C bus configuration is selected by I^2C/SPI# pin. Chip select pin, active low if SPI configuration is selected
A_SET#/SI	28	IU	Primary I^2C-bus salve address assignment enable pin if I^2C bus configuration is selected by I^2C/SPI# pin. SPI data input pin if SPI configuration is selected
A_DONE#/SO	21	O	Primary I^2C-bus slave address assignment complete pin if I^2C bus configuration is selected by I^2C/SPI# pin. SPI data output pin if SPI configuration is selected
RST#	22	IU	When set to low, PI7C1401 is held in reset state
GPIO0	42	B	Programmable I/O pin, general purpose IO
GPIO1	53	B	Programmable I/O pin, general purpose IO
GPIO2	8	B	Programmable I/O pin, general purpose IO
GPIO3	19	B	Programmable I/O pin, general purpose IO
I^2C/SPI#	31	IU	I^2C bus or SPI interface select. I^2C bus interface is selected if this pin is at logic high, SPI interface is selected if this pin is at logic low
SPI_LED_CLK	30	BU	LED sync pin for primary SPI mode only, connect all PI7C1401's SPI_LED_CLK pins together in order to ensure LED sync for all PI7C1401 devices
TEST#	29	IU	Test pin is for internal test purpose, high: normal operation, low: test mode
SCL0	36	IOD	Secondary I^2C-bus output clock for port0
SCL1	49	IOD	Secondary I^2C-bus output clock for port1
SCL2	4	IOD	Secondary I^2C-bus output clock for port2
SCL3	15	IOD	Secondary I^2C-bus output clock for port3
SDA0	35	IOD	Secondary I^2C-bus data input/output, open-drain for port0
SDA1	48	IOD	Secondary I^2C-bus data input/output, open-drain for port1
SDA2	5	IOD	Secondary I^2C-bus data input/output, open-drain for port2
SDA3	16	IOD	Secondary I^2C-bus data input/output, open-drain for port3
TX_FAULT0/INTL0	41	IU	TX_FAULT0 if secondary I^2C bus is connected to SFP+ optic module, INTL0 if secondary I^2C bus is connected to QSFP+ optic module
TX_FAULT1/INTL1	50	IU	TX_FAULT1 if secondary I^2C bus is connected to SFP+ optic module, INTL1 if secondary I^2C bus is connected to QSFP+ optic module
TX_FAULT2/INTL2	55	IU	TX_FAULT2 if secondary I^2C bus is connected to SFP+ optic module, INTL2 if secondary I^2C bus is connected to QSFP+ optic module
TX_FAULT3/INTL3	10	IU	TX_FAULT3 if secondary I^2C bus is connected to SFP+ optic module, INTL3 if secondary I^2C bus is connected to QSFP+ optic module

Pin Description Cont.

Pin Name	Pin#	Type	Description
ABS0/PRSL0	39	IU	ABS0 if secondary I^2C bus is connected to SFP+ optic module, PRSL0 if secondary I^2C bus is connected to QSFP+ optic module
ABS1/PRSL1	47	IU	ABS1 if secondary I^2C bus is connected to SFP+ optic module, PRSL1 if secondary I^2C bus is connected to QSFP+ optic module
ABS2/PRSL2	1	IU	ABS2 if secondary I^2C bus is connected to SFP+ optic module, PRSL2 if secondary I^2C bus is connected to QSFP+ optic module
ABS3/PRSL3	12	IU	ABS3 if secondary I^2C bus is connected to SFP+ optic module, PRSL3 if secondary I^2C bus is connected to QSFP+ optic module
RX_LOS0	37	IU	RX_LOS0 if secondary I^2C bus is connected to SFP+ optic module, NC if secondary I^2C bus is connected to QSFP+ optic module
RX_LOS1	46	IU	RX_LOS1 if secondary I^2C bus is connected to SFP+ optic module, NC if secondary I^2C bus is connected to QSFP+ optic module
RX_LOS2	3	IU	RX_LOS2 if secondary I^2C bus is connected to SFP+ optic module, NC if secondary I^2C bus is connected to QSFP+ optic module
RX_LOS3	14	IU	RX_LOS3 if secondary I^2C bus is connected to SFP+ optic module, NC if secondary I^2C bus is connected to QSFP+ optic module
TX_DISABLE0/RESETL0	40	O	TX_DISABLE0 if secondary I^2C bus is connected to SFP+ optic module, RESETL0 if secondary I^2C bus is connected to QSFP+ optic module
TX_DISABLE1/RESETL1	44	O	TX_DISABLE1 if secondary I^2C bus is connected to SFP+ optic module, RESETL1 if secondary I^2C bus is connected to QSFP+ optic module
TX_DISABLE2/RESETL2	56	O	TX_DISABLE2 if secondary I^2C bus is connected to SFP+ optic module, RESETL2 if secondary I^2C bus is connected to QSFP+ optic module
TX_DISABLE3/RESETL3	11	O	TX_DISABLE3 if secondary I^2C bus is connected to SFP+ optic module, RESETL3 if secondary I^2C bus is connected to QSFP+ optic module
RS0/LPMODE0	38	O	RS0 if secondary I^2C bus is connected to SFP+ optic module, LPMODE0 if secondary I^2C bus is connected to QSFP+ optic module
RS1/LPMODE1	45	O	RS1 if secondary I^2C bus is connected to SFP+ optic module, LPMODE1 if secondary I^2C bus is connected to QSFP+ optic module
RS2/LPMODE2	2	O	RS2 if secondary I^2C bus is connected to SFP+ optic module, LPMODE2 if secondary I^2C bus is connected to QSFP+ optic module
RS3/LPMODE3	13	O	RS3 if secondary I^2C bus is connected to SFP+ optic module, LPMODE3 if secondary I^2C bus is connected to QSFP+ optic module
LED_GRN0	34	O	Green LED output for port0
LED_GRN1	51	O	Green LED output for port1
LED_GRN2	6	O	Green LED output for port2
LED_GRN3	17	O	Green LED output for port3
LED_YLW0	33	O	Yellow LED output for port0
LED_YLW1	52	O	Yellow LED output for port1
LED_YLW2	7	O	Yellow LED output for port2

Pin Description Cont.

Pin Name	Pin#	Type	Description
LED_YLW3	18	O	Yellow LED output for port3
VDD_S	9, 43, 54	P	3.3V Power supply for down stream interface
VDD_P	20, 26	P	1.8V to 3.3V Power supply for primary interface I/Os (pin 21-25, 28)
GND1	27	G	Connect to ground
GND2	Center Pad	G	Connect to ground
NC	32	–	Not Connection

Pin type: B=Bi-directional, I=Input, IU=Input with pull-up, ID=Input with pull-down, IOD=Bi-directional with open drain output, OD=Open drain output, O=Output, P=Power, G=Ground.

Functional Description

The PI7C1401 can interface with four ports and aggregate the I^2C and low-speed control and status signals associated with these ports into a single host-side interface (SPI or I^2C). Multiple PI7C1401s can be combined to support up to 56 total ports, all of which are controlled via the same host-side interface. This greatly reduces the number of signals which route to the host controller.

Functionally, the PI7C1401 is organized as shown in Block Diagram on page 2. Two types of host-side control interfaces are supported (SPI or I^2C) for controlling and monitoring the downstream ports. The PI7C1401 has two special outputs per downstream port (LED_GRN,LED_YLW) which can be used to drive port status LEDs.

1. Host Interface

PI7C1401 host interface can be set as either I^2C or SPI bus depending on the strap value of the pin I^2C/SPI#.

1.1 I^2C

When pin I^2C/SPI# is set high, I^2C is selected. PI7C1401 can support up to 1Mhz Fast-mode. The maximum number of PI7C1401 devices which can share the bus is 14. PI7C1401 does not need dedicated address pins for unique addressing since it has an auto-addressing scheme which is accomplished by connecting one PI7C1401's A_DONE# pin to the subsequent PI7C1401's A_SET# pin.

The first PI7C1401 will connect A_SET# to GND,and the last PI7C1401 will connect A_DONE# to GND. A_SET# pin has internal pullup resistor, PI7C1401 will not respond to any I^2C transaction until this pin is low. A_DONE# pin is high-Z at default and Host controller should first configure each PI7C1401 devices to have a unique address using the default I^2C address(0x1E), once the device is programmed to a new address, PI7C1401 will drive low to A_DONE# and it will not respond to I^2C transaction using default address. After address programming is done. Pin A_SET# and A_DONE# are used to transfer the LED clock for blinking synchronization. Power cycling the device or toggling the RST# pin will restore the device to the default address. Each PI7C1401, each port behind the device and each logical device address within each port is accessible to the Host controller via a unique I^2C address.

Below is an example of I^2C address map:

Table 1. I^2C 8-Bit Address Map

PI7C1401 Instance#	PI7C1401 Self-Address	Port0		Port1		Port2		Port3	
		Device 0xA0	Device 0xA2	Device 0xA0	Device 0xA2	Device 0xA0	Device 0xA2	Device 0xA0	Device 0xA2
ALL	0x02	—	—	—	—	—	—	—	—
0	0x04	0x20	0x22	0x24	0x26	0x28	0x2A	0x2C	0x2E
1	0x06	0x30	0x32	0x34	0x36	0x38	0x3A	0x3C	0x3E
2	0x08	0x40	0x42	0x44	0x46	0x48	0x4A	0x4C	0x4E
3	0x0A	0x50	0x52	0x54	0x56	0x58	0x5A	0x5C	0x5E
4	0x0C	0x60	0x62	0x64	0x66	0x68	0x6A	0x6C	0x6E
5	0x0E	0x70	0x72	0x74	0x76	0x78	0x7A	0x7C	0x7E
6	0x10	0x80	0x82	0x84	0x86	0x88	0x8A	0x8C	0x8E
7	0x12	0x90	0x92	0x94	0x96	0x98	0x9A	0x9C	0x9E
8	0x14	0xA0	0xA2	0xA4	0xA6	0xA8	0xAA	0xAC	0xAE
9	0x16	0xB0	0xB2	0xB4	0xB6	0xB8	0xBA	0xBC	0xBE
10	0x18	0xC0	0xC2	0xC4	0xC6	0xC8	0xCA	0xCC	0xCE
11	0x1A	0xD0	0xD2	0xD4	0xD6	0xD8	0xDA	0xDC	0xDE
12	0x1C	0xE0	0xE2	0xE4	0xE6	0xE8	0xEA	0xEC	0xEE
13	0x1E	0xF0	0xF2	0xF4	0xF6	0xF8	0xFA	0xFC	0xFE

1.2 SPI

If I^2C/SPI# is low, SPI is selected. PI7C1401 can support up to 33Mhz of SPI speed. The maximum number of PI7C1401 devices which can share a single SPI bus is unlimited. All PI7C1401 devices are connected in a daisy-chain fashion, the first PI7C1401 will connect SI to the host controller's MOSI pin, and SO of the first PI7C1401 will connect to the subsequent PI7C1401's SI pin and so on until the last PI7C1401's SO pin connects back to the host controller's MISO pin. All PI7C1401 will connect SCL and CS# to the same pin on the host controller. SI input is ignored and SO output is high-Z whenever CS# is high to synchronize LED blinking across multiple PI7C1401 devices, the SPI_LED_CLK pin should be connected across all PI7C1401 devices in SPI mode.

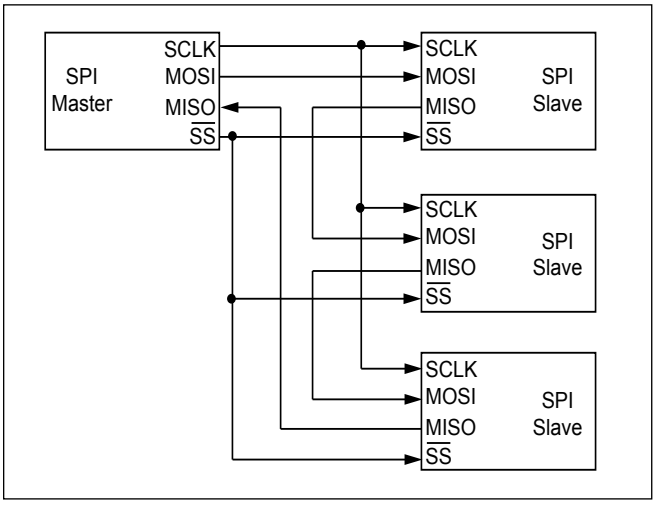
The SPI address is 12 bit and it is unique for each port and each logical devices. Refer to below table:

Table 2. SPI Address Map

PI7C1401 Instance#	Address Range								
	Port0		Port1		Port2		Port3		PI7C1401 Regs
	Device 0xA0	Device 0xA2	Device 0xA0	Device 0xA2	Device 0xA0	Device 0xA2	Device 0xA0	Device 0xA2	
0	0x000 to 0x0FF	0x100 to 0x1FF	0x200 to 0x2FF	0x300 to 0x3FF	0x400 to 0x4FF	0x500 to 0x5FF	0x600 to 0x6FF	0x700 to 0x7FF	0x800 to 0x8FF
1									
2									
–									
N									

1.2.1 SPI Frame Structure for Daisy Chain

Figure 1. Daisy Chained SPI Bus



By default, daisy chain is used for SPI operation.

Each SPI transaction to a single PI7C1401 device is 29 bits long, please see below table:

Table 3. SPI Frame Structure

Bit	Field	Description
28	R/W	0: Write command 1: Read Command This is the first bit shifted in on the MOSI input.

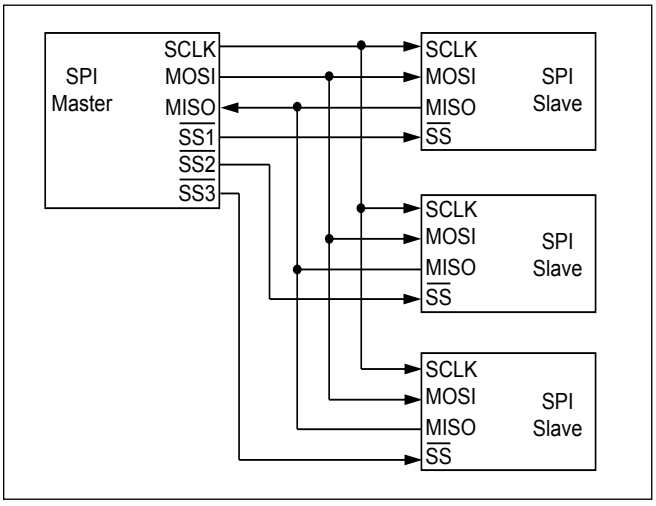
Bit	Field	Description
27:16	ADDR[11:0]	12-bit address field. See Table 2.
15	DATA[15]	Busy flag. For read operations, a '1' means the downstream port is buys. For write operations, DATA[15] is a don't care.
14	DATA[14]	Don't care.
13	DATA[13]	NACK received flag. A '1' means the PI7C1401 has received a NACK from the downstream port.
12	DATA[12]	Reject flag. A '1' means the PI7C1401 has rejected the previous command because it is busy servicing a prior command.
11:8	DATA[11:8]	Don't care.
7:0	DATA[7:0]	8-bit data field. DATA[0] is the last bit shifted in on the MOSI input.

When daisy-chain N PI7C1401 devices, there are 29 x N bit in one SPI transaction (one CS# assertion), When CS# transit from low to high, each PI7C1401 device in the SPI chain will capture the command in its 29 bits shift registers.

The previous SPI command, address, and data are shifted out on SO as the current SPI command, address, and data are shifted in on SI. SPI read operation requires two SPI transactions, in between these two transactions, PI7C1401 fetches the requested data from wither local PI7C1401 registers or from the downstream port. And PI7C1401 shifted the data on SO in the 2nd SPI transaction. The 2nd SPI transaction can be dummy frame composed of all ones. SPI write operation requires only one SPI transaction.

1.2.2 SPI Burst Transaction

Figure 2. Independent Configuration SPI Bus



If SPI devices are connected in independent configuration mode, that is, each PI7C1401 has separated connection to host controller for CS#, but only share SI, SO(wired-or) and SCL. Host controller can burst read to local PI7C1401 registers (including prefetch Data registers) after set a burst read enable register.

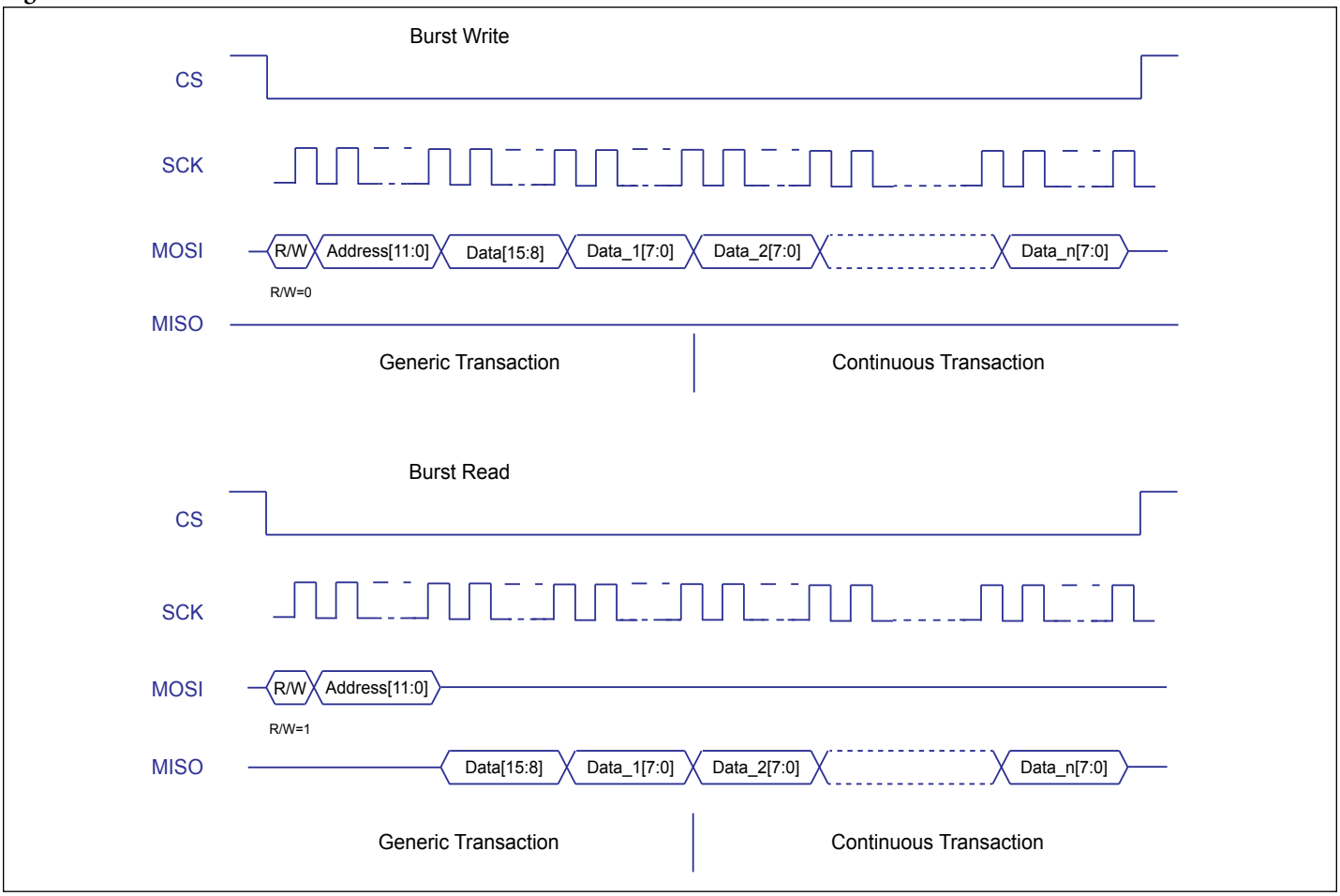
For frame of burst read, it requires only one SPI transaction and more data byte can be read out after the first 29 bits in one SPI transaction. Host controller can stop the burst read by de-assert CS#.

Table 4. SPI Burst read/write frame structure at independent configuration

Bit	Field	Description
28	R/W	0: Write, 1:Read
27:16	ADDR	12-bit Address
15	DATA[15]	Busyflag

Bit	Field	Description
14	DATA[14]	Don't care
13	DATA[13]	NACK flag
12	DATA[12]	Reject flag
11:8	DATA[11:8]	Don't care
7:0	DATA[7:0]	The first byte 8-bit data
7:0	DATA[7:0]	Second byte 8-bit data (continuer/w)
:	:	:
7:0	DATA[7:0]	The end of byte 8-bit data

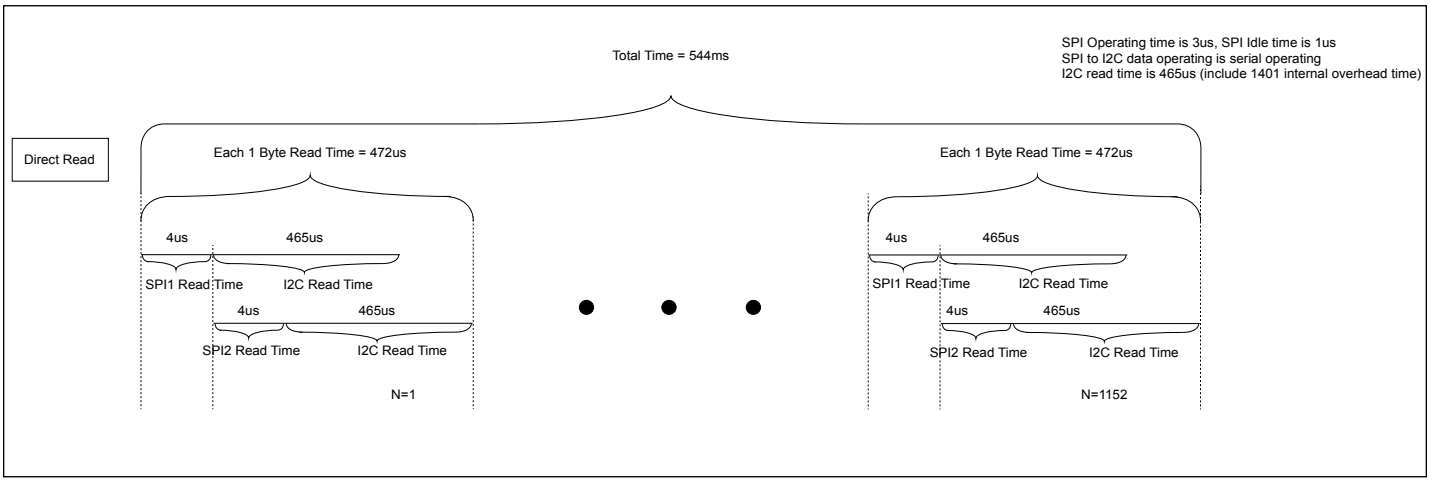
Figure 3. SPI Burst Transaction Waveform

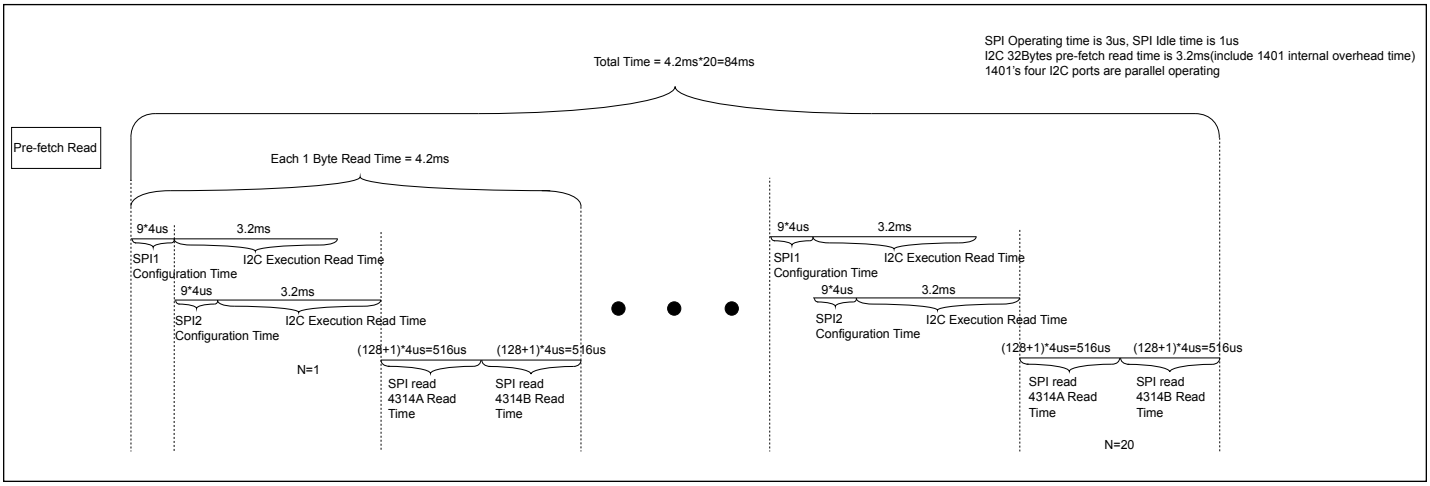


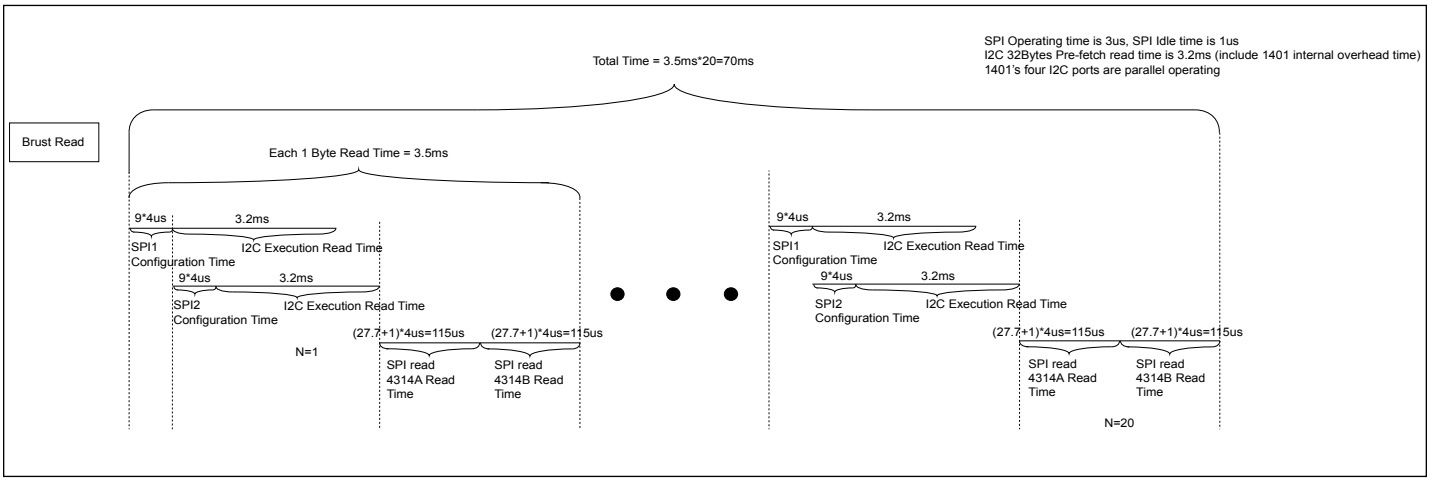
Typical SPI Bus Burst read/write Configuration

1. Connect PI7C1401 SPI bus in independent configuration.
2. Set SPI Burst Enable Register -- B1h[0] = 1 (default value 0)
3. Follow generic transaction continuous read (or write) local register 8-bit data.
4. Or follow generic transaction continuous read pre-fetch RAM's 8-bit data.

1.2.3 SPI Read Latency







2. LED Control

The PI7C1401 uses two sets of outputs, LED_GRN[3:0] and LED_YLW[3:0], to drive LEDs associated with the ports under its control. Most SFP+ and QSFP+ applications use one yellow and one green LED per port to indicate different link status such as link up, link down, and other link states.

LEDs should be connected to the PI7C1401 in an active-low fashion. When the LED_GRN or LED_YLW pin drives a low voltage (V_{OL}), the LED is illuminated. When drives a high voltage (V_{OH}), the LED is off. Bi-color LEDs can be connected in a similar fashion, and each LED should have its own current-limiting resistor. The current-limiting resistor value is selected by choosing the desired maximum current through the LED and the corresponding voltage drop from the LED's current vs. voltage plot. The sum of forward voltage drop of the LED, the voltage drop across the series resistor, and the maximum V_{OL} (0.5V maximum for currents between 2 and 18mA) is equal to the LED supply voltage. Note that LED_GRN and LED_YLW are tri-stated while the device is held in reset (during POR or while the RST# pin is low), and are enabled during normal operation and drive a high voltage by default.

Each port under the PI7C1401's control has a set of registers that allow the user to configure each LED into one of the following states:

- ON
- OFF
- PWM (ON with programmable intensity)
- BLINK (with programmable blink duty cycle, frequency, and ON intensity)

LED blinking is configured by setting an on and an off time using control registers (see programming guide)

LED blinking can be synchronized across all four ports under the PI7C1401's control, and it can be synchronized across all ports in the system which are under the control of an PI7C1401. For SPI, cross-device synchronization utilizes the SPI_LED_CLK pin. One device is configured to forward its internal LED clock to this pin, and all other devices are configured to receive an external LED clock on this pin. For I^2C, the first device in the A_DONE# to A_SET# pin daisy chain is configured to output its internal LED clock to the A_DONE# pin. All other devices are configured to receive an external LED clock from the A_SET# pin and to output the clock to the A_DONE# pin.

I2C mode daisy chain LED synchronization sequence

1. Set LED counter registers for each devices (same value recommend)
2. Set forward LED clock enable for each devices
3. Configure all device to receive external LED clock
4. All of device's LED blinking counter reset
5. Set the first device (nearest the host) in the daisy chain to use internal LED clock
   (LED sync clock will output from A_DONE# pin)

SPI mode daisy chain LED synchronization sequence

1. Set LED counter registers for each devices (same value recommend)
2. Configure all device to receive external LED clock
3. All of device's LED blinking counter reset
4. Set any one device to enable LED clock output (only one device acceptable)
5. Set above device to use internal LED clock
   (LED sync clock will output from SPI_LED_CLK pin)

3. Low Speed downstream Outputs Control

The PI7C1401 has two general-purpose outputs per port (TX_DISABLE/RESETL and RS/LPMODE) which can be used to drive the low-speed inputs to the module. The host controller can change the state of these outputs for each port individually, for all ports connected to a given PI7C1401 device simultaneously, or for all ports in the system simultaneously.

There are two configuration registers for these output pins (see detail in programming guide). A 10-kΩ pull-up or pull-down resis-

tor is recommended (note that SFP+ and QSFP+ modules have internal pull-up and pull-downs on certain inputs). Note that if the VDD_S rail does not have power and there is an externally powered pull-up resistor connected to an output pin, the output pin will be pulled low until VDD_S is supplied.

4. Low Speed Input Status and Interrupt

The PI7C1401 has three general-purpose inputs per port which can be used to monitor the low-speed outputs from the module. The host controller can monitor the status of these signals for each port by reading the appropriate registers in the PI7C1401. In addition, the PI7C1401 can be configured to generate an interrupt to the host via the IRQ# signal whenever one or more of the low-speed input signals change state. Multiple PI7C1401s' IRQ# pins can be connected together in a wired-or fashion.

The interrupt can be configured to trigger on the falling edge, the rising edge, or both the falling and rising edges. A single register stores flags for which inputs and edges are responsible for the trigger (detail see programming guide)

The PI7C1401 is also able to generate an interrupt based on pre-fetched data. This is known as a data-driven interrupt. The PI7C1401 monitors up to four bytes within the pre-fetched range for each port. For each of the bytes, the register offset address is programmed to a local PI7C1401 register as well as the enable bit fields which will trigger the interrupt. When one of the enabled bits of the four monitored bytes changes state from a '0' to a '1' and stays a '1' for two consecutive periodic pre-fetch cycles (0→1→1), the interrupt is generated and the periodic pre-fetch operation is halted. The PI7C1401 has four port-specific registers which contain the sampled data from the bytes being monitored after the interrupt is triggered. To clear the interrupt, the trigger source byte's sampled data register is read. The periodic pre-fetch must be restarted after the interrupt is cleared with an I^2C command. Because it takes two periodic pre-fetch cycles to trigger this interrupt, it may take up to 10ms for the host to see the trigger after the downstream module's monitored bit field changes for the fastest periodic pre-fetch setting.

The PI7C1401 also has the ability to generate an interrupt if there is a abnormal behavior in the downstream I^2C bus. The SDA bus and the SCL bus each have timers that will trigger an interrupt if they are held in a low state too long due to excessive clock stretching or a port error. Once the interrupt is triggered, it is cleared by issuing a port reset on the relevant port. These interrupts are known as SCL Stuck and SDA Stuck interrupts and can be configured individually for each port. By default, the SCL Stuck interrupt will trigger after the SCL bus is held low for 35ms (typical). This value is configurable individually by port. The SDA Stuck interrupt will trigger after the SDA is held low for 1 s (typical). The user may issue a port reset sequence (9 consecutive SCL clock cycles with the last being an I^2C stop condition) or module reset to restore the module to a known state.

When a host-side interrupt is triggered, the host must determine the source and cause of the interrupt. The recommended procedure for identifying the source and cause of an interrupt is as follows:

1. Read the PI7C1401 aggregated port interrupt flags of the first PI7C1401 instance to see which, if any, downstream port triggered the interrupt.

2. If this instance of the PI7C1401 has any aggregated port interrupts flagged, read all of the status registers to determine the source of the interrupt and clear it. If an SCL Stuck or SDA Stuck interrupt is triggered, a port reset must be issued and the periodic pre-fetch must be restarted. The host may also perform other housekeeping activities based on the interrupt, such as change the state of the LEDs after a module is no longer present.

3. Repeat steps 1 and 2 for the next PI7C1401 instance, until the IRQ# bus is cleared.

This procedure applies to every PI7C1401 device which is wire-or'ed to the host-side interrupt signal. The total time required for the host to identify the source and cause of the interrupt for an implementation consisting of N total PI7C1401's, where all N IRQ# outputs are wire-or'ed together, is as follows:

$T_{interrupt}$ = Delay between the input pin changing state and the corresponding PI7C1401 device triggering an interrupt (50µs max).

T_{read} = Time required to read a single register from N PI7C1401 devices.

For I^2C mode, $T_{read} = (9*4*N)/F_{I2C}$, where F_{I2C} is the SCL clock frequency.

For SPI mode, $T_{read} = (29*2*N)/F_{SPI} + T_{OFF-CSN}$, where F_{SPI} is the SCK clock frequency, and $T_{OFF-CSN}$ is the CS#

off time.

$T_{total} = T_{interrupt} + 4*T_{read}$

See the Programming Guide for more details on how to configure the interrupts.

5. Downstream I²C Master

The PI7C1401 has four master I²C interfaces for managing up to four ports, referred to as “downstream” ports. Each downstream I²C interface can be configured to operate with an SCL clock frequency between 100kHz and 400kHz. The downstream I²C master supports clock stretching.

The SFF-8472 and SFF-8431 specifications define up to two logical device addresses per SFP+ port: 0xA0 and 0xA2. The SFF-8436 specification defines one logical device address per QSFP+ port: 0xA0. Both 0xA0 and 0xA2 are directly addressable by the upstream host controller. Refer to Table 1 (I²C) and Table 2 (SPI). The PI7C1401 uses this address mapping scheme to decode the port and device address and perform a downstream I²C read or write operation. This is known as a direct access. Direct accesses have the highest priority when accessing the downstream module. If there is an on-going periodic pre-fetch or indirect write, these operations will be stopped at the next byte boundary and the direct access will be executed. The periodic pre-fetch or schedule write operation will be resumed after the direct access finishes. Note that the periodic pre-fetch will begin from the starting register offset of the pre-fetch range rather than where it left off during the interruption. If a direct access is attempted during an interrupt-driven pre-fetch, the interrupt-driven pre-fetch will finish and the direct access will be executed afterwards. If an autonomous access (pre-fetch or indirect write) occurs during a direct access, the autonomous access will be executed after the direct access is completed.

6. Pre-Fetch Read From Downstream Modules

The PI7C1401 can be configured to pre-fetch data from each downstream port’s module. The pre-fetched data is stored locally in the PI7C1401's memory, allowing any downstream read operations in the pre-fetch range to be directly read from the PI7C1401 rather than waiting for the PI7C1401 to read from the downstream device through I²C. The PI7C1401 can pre-fetch data from the ports on a one-time basis, a regular basis (periodic pre-fetch), or upon the occurrence of certain events (interrupt-driven pre-fetch).

For periodic pre-fetching, the pre-fetched range is determined by two settings, the pre-fetch length and the pre-fetch offset address. The PI7C1401 will pre-fetch beginning at the offset address for a length of bytes between 1 and 32. The target device address is set to either 0xA0 or 0xA2. Once configured, the start bit is set to begin periodic pre-fetching and the stop bit is set to stop pre-fetching. After a pre-fetch is completed, the gate bit is reset to '0', and any attempted read operation in the pre-fetched range will return data from the PI7C1401's memory containing the last pre-fetched data. To modify the pre-fetched range or to stop the PI7C1401 from returning the data from memory, the gate bit must be set to '1'. If the PI7C1401 receives a NACK during a pre-fetch attempt, the gate bit will automatically be set to '1'. Each port has its own gate bit and separate memory and settings.

For interrupt-driven pre-fetch, the interrupt event can be configured for either the rising- or falling-edge of one of the low speed input signals of a port. The pre-fetch range and target device address is configured similarly but independently of the periodic pre-fetch settings. Interrupt-driven pre-fetch also has a gate bit and memory independent of the periodic pre-fetch. Once an interrupt-driven pre-fetch occurs successfully, an interrupt is triggered on the IRQ# pin and the aggregated interrupt flag for that port will be set. For the interrupt to be cleared and for another interrupt pre-fetch to occur, it must be re-armed with a register write. If the pre-fetch attempt is NACK'd, the gate bit will not be set, the interrupt will not be generated, and the interrupt-driven pre-fetch does not need to be re-armed. Note that the pre-fetched data from the interrupt-driven pre-fetch has precedence over the data from a periodic pre-fetch if they have overlapping pre-fetch ranges. The PI7C1401 will return data from the interrupt-driven pre-fetch even if the periodic pre-fetch data is more recent. When an interrupt-driven pre-fetch occurs, it is recommended that it is dealt with immediately by reading the pre-fetched data and re-arming it.

See the Programming Guide for more details on how to configure data pre-fetch.

7. Indirect Write

The PI7C1401 has the ability to schedule a write operation on one or more downstream modules simultaneously by writing to local PI7C1401 registers. This operation, known as a indirect write, allows for quicker writing by utilizing the faster host-side I²C rate. The host-side I²C bus is not held while the write occurs in the downstream I²C. This command may be broadcasted to all PI7C1401 to write to any combination of ports concurrently.

indirect writes can be directed to an individual port (port Indirect Write) or to a group of two or more ports simultaneously (common indirect write). The status of the port indirect write or common indirect write may be checked in a local PI7C1401 register. This register will reflect if the operation completed successfully, or if it was NACKed by the downstream module.

indirect write operations have a higher priority than periodic pre-fetch operations. This means that if a schedule write is sent while a

periodic pre-fetch is on-going, the periodic pre-fetch will be stopped at the next byte boundary and the indirect write will be executed. The periodic pre-fetch will resume on the next period. Note that it will begin reading at the start of the pre-fetch range rather than where the indirect write occurred.

See the Programming Guide for more details on how to configure indirect write.

8. I²C Bus Timeouts

The PI7C1401 has a watchdog timer to ensure that the I²C buses do not become permanently stuck. For example, if the host is performing a direct access on a downstream module, the PI7C1401 will clock stretch the host-side I²C while the downstream I²C transaction occurs. If the downstream module clock stretches for a very long time or any other error occurs that prevents the transaction from finishing, the host-side I²C will not become stuck. The watchdog timer is what prevents this from happening by setting a maximum time for the downstream transaction to complete; and if it does not complete, the timer expires and the PI7C1401 will NACK the host to terminate the transaction. By default, the timer is set to 3ms and is programmable in steps of 1ms up to 127ms. This timer may also be disabled, but this is not recommended as the I²C bus may become permanently stuck and a device reset will be necessary. Each port's I²C master also has a programmable watchdog timer which operates similarly to the host-side I²C watchdog timer.

When the host attempts a direct access transaction through I²C, after the I²C device ID has been ACKed, the PI7C1401 waits for the host to send a register offset address or a read/write command before downplaying it on the downstream port I²C. If the host becomes busy with something else and does not finish the I²C transaction, the PI7C1401 state machine will be stuck. There is a I²C Bus Timeout timer for each port to prevent this from happening. If the host does not finish the I²C transaction within this timer, the PI7C1401 will timeout and return to the idle state. This counter is 10ms (typical) by default and is configurable in steps of 1ms up to 255ms.

See the Programming Guide for more details on how to configure I²C Bus Timeouts.

9. General-Purpose Inputs/Outputs

The PI7C1401 has multiple general purpose input/output pins which can be used to control auxiliary functions on the board through the same host-side control interface which is used to manage the ports. The GPIO pins can be configured as inputs or outputs through the PI7C1401 registers.

If ringing is a concern a series resistor may be placed near the GPIO pin. A good rule of thumb for sizing the resistor is the difference of the transmission line characteristic impedance minus the driver impedance. For example in the case of a 60Ω transmission line impedance a 50Ω series resistor may be used to minimize ringing. Cases such as these may be simulated using the provided PI7C1401

IBIS model.

10. Register Map

Register Type	Definition
RO	Read Only
RW	Read/Write
RWS	Read/Write and Self-clear
ROC	Read only and Clear at read

Offset 00H(default=00)—Reset Register

Bit	Type	Description
[7]	RW	Reset local register when set
[6:4]	RO	Reserved
[3:0]	RW	Reset port3-0 when set, it clear port logic including prefetch

Offset 01H(default=3F)—I2C Device ID Register

Bit	Type	Description
[7:1]	RW	When bit[0] is set, Device ID can be programmed by host at I2C mode
[0]	RW	This bit must be cleared when bit[7:1] is programmed This bit can't be set except hardware reset

Offset 02H(default=00)—External LED Clock Enable Register

Bit	Type	Description
[7:4]	RO	Reserved
[3]	RW	When set to 1, the external LED clock will be used for LED counter
[2:0]	RO	Reserved

Offset 03H(default=00)—Reserved

Bit	Type	Description
[7:0]	RO	Reserved

Offset 04H(default=46)—I2C Slave Watchdog Timer Register

Bit	Type	Description
[7:1]	RW	Host side I2C watchdog timer[6:0], the unit is ms and default is 35ms The counter is armed when receives Start, disarmed when receives Stop, and cleared when see ACK. When it timeout, I2C slave is set to idle state
[0]	RW	When set, disable the I2C slave watchdog timer

Offset 05H(default=00)—Reserved

Bit	Type	Description
[7:0]	RO	Reserved

Offset 06H(default=00)—TX_FAULT/INTL Inputs Status and Aggregated Interrupt Status Register

Bit	Type	Description
[7:4]	RO	Input status of TX_FAULT/INTL for port 3~0
[3:0]	RO	Aggregated interrupt status for port 3~0. It will be cleared once all of the interrupts are cleared

Offset 07H(default=00)—ABS/PRSL and RX_LOS Inputs Status Register

Bit	Type	Description
[7:4]	RO	Input status of pin RX_LOS for port 3~0
[3:0]	RO	Input status of pin ABS/PRSL for port 3~0

Offset 08H(default=00)—TX_DISABLE/RESET and RS/LPMODE Output Enable Register

Bit	Type	Description
[7:4]	RW	Output enable for outputs RS/LPMODE of port 3~0
[3:0]	RW	Output enable for outputs TX_DISABLE/RESET of port 3~0

Offset 09H(default=FF)—LED_YLW and LED_GRN Output Enable Register

Bit	Type	Description
[7:4]	RW	Output enable for outputs LED_YLW of port 3~0
[3:0]	RW	Output enable for outputs LED_GRN of port 3~0

Offset 0AH(default=0F)—TX_DISABLE/RESET and RS/LPMODE Output Value Register

Bit	Type	Description
[7:4]	RW	Output value for outputs RS/LPMODE of port 3~0
[3:0]	RW	Output value for outputs TX_DISABLE/RESET of port 3~0

Offset 0BH(default=FF)—Prefetch Gate Register

Bit	Type	Description
[7:4]	RO	Interrupt driven prefetch gate register of port 3~0.When this bit is 1, host cannot access the interrupt driven prefetching range and any access falling into this range is directed to downstream port. this bit is reset to 0 when interrupt driven prefetching has completed successfully. and host can directly access the memory if it fall into the range. This bit is set to 1 when interrupt driven prefetching is re-armed or disabled. if the interrupt driven prefetch range is overlaps the periodic prefetch range,host will get data from the interrupt driven prefetching memory.
[3:0]	RW	Periodic prefetch gate register of port 3~0. When this bit is 1, host cannot access the periodic prefetching range and any access falling into this range is directed to downstream port. This bit is reset to 0 when prefetching has completed successfully. and host can directly access the memory if it fall into the range. This bit should be set to 1 when pre-fetching range is modified or periodic prefetch is stopped.

Offset 0CH(default=00)—Interrupt Driven Prefetch Source Register

Bit	Type	Description
[7:6]	RW	Interrupt driven prefetch source register of port 3 0h = interrupt driven prefetch is disabled 1h = input pin TX_FAULT3 is selected 2h = input pin ABS/PRSL3 is selected 3h = input pin RX_LOS3 is selected
[5:4]	RW	Interrupt driven prefetch source register of port 2 0h = interrupt driven prefetch is disabled 1h = input pin TX_FAULT2 is selected 2h = input pin ABS/PRSL2 is selected 3h = input pin RX_LOS2 is selected

[3:2]	RW	Interrupt driven prefetch source register of port 1 0h = interrupt driven prefetch is disabled 1h = input pin TX_FAULT1 is selected 2h = input pin ABS/PRSL1 is selected 3h = input pin RX_LOS1 is selected
[1:0]	RW	Interrupt driven prefetch source register of port 0 0h = interrupt driven prefetch is disabled 1h = input pin TX_FAULT0 is selected 2h = input pin ABS/PRSL0 is selected 3h = input pin RX_LOS0 is selected

Offset 0DH(default=00)—Downstream Address 0xA2 Absence Register

Bit	Type	Description
[7:4]	RO	Reserved
[3:0]	RW	Downstream address 0xA2 absence control for port 3~0. When this bit is set to 1, the device address 0xA2 is absent and access to 0xA2 from host will be Nacked without being sent to the downstream port.

Offset 0EH(default=00)—I2C LED Clock Output Enable Register

Bit	Type	Description
[7:5]	RW	3'b000 : pin A_DONE# is set to default functionality 3'b001 : pin A_DONE# outputs LED clock.
[4:0]	RO	Reserved

Offset 0FH(default=--)—GPIO and Pin Status Register

Bit	Type	Description
[7]	RO	status of pin I2C/SPI#
[6]	RO	status of pin SPI_LED_SYNC
[5]	RO	status of pin A_DONE#
[4]	RO	status of pin A_SET#
[3]	RO	status of pin GPIO3
[2]	RO	status of pin GPIO2
[1]	RO	status of pin GPIO1
[0]	RO	status of pin GPIO0

Offset 10H/30H/50H/70H(default=00)—Reserved

Bit	Type	Description
[7:0]	RO	Reserved

Offset 11H/31H/51H/71H(default=98)—Port 0~3 SCL High Time Register

Bit	Type	Description
[7:0]	RW	Port I2C SCL high time in number of main clock(27Mhz +/- 10%) together with SCL Low Time register to define the port I2C bit rate. set to 0x98 gives 100K at most. set to 0x26 gives 400K at most.

Offset 12H/32H/52H/72H(default=98)—Port 0~3 SCL Low Time Register

Bit	Type	Description
[7:0]	RW	Port I2C SCL low time in number of main clock(27Mhz +/- 10%) together with SCL High Time register to define the port I2C bit rate. set to 0x98 gives 100K at most. set to 0x26 gives 400K at most. This register can only be programmed to be laess than 0xAA

Offset 13H/33H/53H/73H(default=A0)—Port 0~3 Downstream I2C Device ID Modifier Register

Bit	Type	Description
[7:3]	RW	Upper five bit of I2C Device ID.
[2]	RW	When set to 1, it disable the port I2C watchdog timer(A9H/AAH/ABH/ACH).
[1]	RW	Downstream I2C Device ID bit 2
[0]	RO	Reserved

Offset 14H/34H/54H/74H(default=00)—Port 0~3 LED_GRN PWM Control Register

Bit	Type	Description
[7:0]	RW	Control the PWM on time of LED_GRN output for LED brightness. FFh is the brightnest setting, and 01h is the dimmest setting, 00h is completely off. On time value = This register value x 10 us Off time value = 2.55ms - On time.

Offset 15H/35H/55H/75H(default=00)—Port 0~3 LED_YLW PWM Control Register

Bit	Type	Description
[7:0]	RW	Control the PWM on time of LED_YLW output for LED brightness. FFh is the brightnest setting, and 01h is the dimmest setting, 00h is completely off. On time value = This register value x 10 us Off time value = 2.55ms - On time.

Offset 16H/36H/56H/76H(default=00)—Port 0~3 LED_GRN Blink On Control Register

Bit	Type	Description
[7:0]	RW	When Blinking mode is set, control the BLINKING on time of LED_GRN output, the unit is 2.5ms. When 4x long on/off time blink mode is set, the unit is 10ms setting to 0 is not allowed.

Offset 17H/37H/57H/77H(default=00)—Port 0~3 LED_GRN Blink Off Control Register

Bit	Type	Description
[7:0]	RW	Control the BLINKING off time of LED_GRN output When Blinking mode is set, the unit is 2.5ms. When 4x long on/off time blink mode is set, the unit is 10ms setting to 0 is not allowed.

Offset 18H/38H/58H/78H(default=00)—Port 0~3 LED_YLW Blink On Control Register

Bit	Type	Description
[7:0]	RW	When Blinking mode is set, control the BLINKING on time of LED_YLW output, the unit is 2.5ms. When 4x long on/off time blink mode is set, the unit is 10ms setting to 0 is not allowed.

Offset 19H/39H/59H/79H(default=00)—Port 0~3 LED_YLW Blink Off Control Register

Bit	Type	Description
[7:0]	RW	Control the BLINKING off time of LED_YLW output When Blinking mode is set, the unit is 2.5ms. When 4x long on/off time blink mode is set, the unit is 10ms setting to 0 is not allowed.

Offset 1AH/3AH/5AH/7AH(default=30)—Port 0~3 LED Mode Select Register

Bit	Type	Description
[7:6]	RW	LED long time blink mode select: 0h = OFF 1h = LED_GRN/YLW 4x long on/off time mode enable 2h = Reserved 3h = LED_GRN/YLW 125ms interminttent blinking mode enable
[5]	RW	When set, inverts the LED_YLW output for active low LED
[4]	RW	When set, inverts the LED_GRN output for active low LED
[3:2]	RW	LED mode select for LED_YLW: 0h = OFF 1h = ON(no PWM) 2h = PWM(for dimming) 3h = BLINK
[1:0]	RW	LED mode select for LED_GRN: 0h = OFF 1h = ON(no PWM) 2h = PWM(for dimming) 3h = BLINK

Offset 1BH/3BH/5BH/7BH(default=00)—Port 0~3 Interrupt Driven Prefetch Control Register

Bit	Type	Description
[7:3]	RW	Number of bytes of interrupt driven prefetch range is equal to this register plus one. between single byte to 32 bytes.
[2]	RW	When set, enable modified down port ID of device full features
[1]	RW	Define which edge of the input pin will trigger the prefetch 0h = falling edge 1h = rising edge

[0]	RW	Define the I2C device address for interrup driven prefetching: 0h = 0xA0 1h = 0xA2

Offset 1CH/3CH/5CH/7CH(default=00)—Port 0~3 Interrupt Driven Prefetch Offset Address Register

Bit	Type	Description
[7:0]	RW	Define the starting address of interrupt driven prefetching.

Offset 1DH/3DH/5DH/7DH(default=00)—Port 0~3 Periodic Prefetch Control Register

Bit	Type	Description
[7:3]	RW	Number of bytes of periodic prefetch range is equal to this register plus one. between single byte to 32 bytes.
[2]	RWS	Set to 1 will stop the ongoing periodic prefetch. This bit is self cleared when the prefetching is stopped on the port.
[1]	RW	Set to 1 will start the periodic prefetch. This bit is cleared when bit 2(Stop) is set to 1 or self cleared if the prefetch period is set to 0(one-time).
[0]	RW	Define the I2C device address for periodic prefetching: 0h = 0xA0 1h = 0xA2

Offset 1EH/3EH/5EH/7EH(default=00)—Port 0~3 Periodic Prefetch Offset Address Register

Bit	Type	Description
[7:0]	RW	Define the starting address of periodic prefetching.

Offset 1FH/3FH/5FH/7FH(default=00)—Port 0~3 Periodic Prefetch Period Register

Bit	Type	Description
[7:0]	RW	Define the period of periodic prefetching in unit of 5ms Maximum of 1.3 seconds can be set. Setting to 0 will trigger one-time prefetch.

Offset 20H/40H/60H/80H(default=00)—Port 0~3 Input Pin Interrupt Enable Register

Bit	Type	Description
[7]	RW	When set, ABS/PRSL 2s hot-plug interrupt delay enable
[6]	RW	When set, TX_FAULT/INTL 300ms input window gate enable
[5]	RW	Setting 1 enable interrupt at falling edge of pin ABS/PRSL
[4]	RW	Setting 1 enable interrupt at rising edge of pin ABS/PRSL
[3]	RW	Setting 1 enable interrupt at falling edge of pin RXLOS
[2]	RW	Setting 1 enable interrupt at rising edge of pin RXLOS
[1]	RW	Setting 1 enable interrupt at falling edge of pin TX_FAULT/INTL
[0]	RW	SSetting 1 enable interrupt at rising edge of pin TX_FAULT/INTL

Offset 21H/41H/61H/81H(default=80)—Port 0~3 Interrupt Status and Interrupt Driven Prefetch Re-Arm Register

Bit	Type	Description
[7]	RW	This bit is cleared when an interrupt-driven prefetching is completed successfully on the port. Setting to 1 will clear the interrupt on IRQ# caused by interrupt-driven prefetching completeness and re-arm the interrupt driven prefetching.
[6]	RO	Data driven interrupt status from periodic prefetch. It is cleared when sampled data register from the triggered byte is cleared.
[5]	ROC	Interrupt status of falling edge of input ABS/PRSL. It is cleared when read.
[4]	ROC	Interrupt status of rising edge of input ABS/PRSL. It is cleared when read.
[3]	ROC	Interrupt status of falling edge of input RXLOS. It is cleared when read.
[2]	ROC	Interrupt status of rising edge of input RXLOS. It is cleared when read.
[1]	ROC	Interrupt status of falling edge of input TX_FAULT/INTL It is cleared when read.
[0]	ROC	Interrupt status of rising edge of input TX_FAULT/INTL It is cleared when read.

Offset 22H/42H/62H/82H(default=00)—Port 0~3 Data Driven Interrupt Index 0 Offset Address Register

Bit	Type	Description
[7:0]	RW	Offset register for data-driven interrupt byte(Index 0) This offset must be within the periodic prefetch range. When specific bits(defined in Data Driven Interrupt Index 0 Bit Enable Register) in the prefetched data of this specific offset changes from 0 to 1 and stay at 1 for two prefetch cycles, the data driven interrupt will be triggered and periodic prefetch will be halted.

Offset 23H/43H/63H/83H(default=00)—Port 0~3 Data Driven Interrupt Index 1 Offset Address Register

Bit	Type	Description
[7:0]	RW	Offset register for data-driven interrupt byte(Index 1) This offset must be within the periodic prefetch range. When specific bits(defined in Data Driven Interrupt Index 1 Bit Enable Register) in the prefetched data of this specific offset changes from 0 to 1 and stay at 1 for two prefetch cycles, the data driven interrupt will be triggered and periodic prefetch will be halted.

Offset 24H/44H/64H/84H(default=00)—Port 0~3 Data Driven Interrupt Index 2 Offset Address Register

Bit	Type	Description
[7:0]	RW	Offset register for data-driven interrupt byte(Index 2) This offset must be within the periodic prefetch range. When specific bits(defined in Data Driven Interrupt Index 2 Bit Enable Register) in the prefetched data of this specific offset changes from 0 to 1 and stay at 1 for two prefetch cycles, the data driven interrupt will be triggered and periodic prefetch will be halted.

Offset 25H/45H/65H/85H(default=00)—Port 0~3 Data Driven Interrupt Index 3 Offset Address Register

Bit	Type	Description
[7:0]	RW	Offset register for data-driven interrupt byte(Index 3) This offset must be within the periodic prefetch range. When specific bits(defined in Data Driven Interrupt Index 3 Bit Enable Register) in the prefetched data of this specific offset changes from 0 to 1 and stay at 1 for two prefetch cycles, the data driven interrupt will be triggered and periodic prefetch will be halted.

Offset 26H/46H/66H/86H(default=00)—Port 0~3 Data Driven Interrupt Index 0 Sampled Data Register

Bit	Type	Description
[7:0]	ROC	Sampled data for data-driven interrupt byte(Index 0) Reading this register clean itself and also clear the IRQ# caused by data-driven of Index 0 byte.

Offset 27H/47H/67H/87H(default=00)—Port 0~3 Data Driven Interrupt Index 1 Sampled Data Register

Bit	Type	Description
[7:0]	ROC	Sampled data for data-driven interrupt byte(Index 1) Reading this register clean itself and also clear the IRQ# caused by data-driven of Index 1 byte.

Offset 28H/48H/68H/88H(default=00)—Port 0~3 Data Driven Interrupt Index 2 Sampled Data Register

Bit	Type	Description
[7:0]	ROC	Sampled data for data-driven interrupt byte(Index 2) Reading this register clean itself and also clear the IRQ# caused by data-driven of Index 2 byte.

Offset 29H/49H/69H/89H(default=00)—Port 0~3 Data Driven Interrupt Index 3 Sampled Data Register

Bit	Type	Description
[7:0]	ROC	Sampled data for data-driven interrupt byte(Index 3) Reading this register clean itself and also clear the IRQ# caused by data-driven of Index 3 byte.

Offset 2AH/4AH/6AH/8AH(default=00)—Port 0~3 Data Driven Interrupt Index 0 Bit Enable Register

Bit	Type	Description
[7:0]	RW	If any bit is 1, the corresponding bits of the index 0 byte will trigger the interrupt when it changes (0->1->1) if all bits are 0, the index 0 byte driven interrupt is disabled.

Offset 2BH/4BH/6BH/8BH(default=00)—Port 0~3 Data Driven Interrupt Index 1 Bit Enable Register

Bit	Type	Description
[7:0]	RW	If any bit is 1, the corresponding bits of the index 1 byte will trigger the interrupt when it changes (0->1->1) if all bits are 0, the index 1 byte driven interrupt is disabled.

Offset 2CH/4CH/6CH/8CH(default=00)—Port 0~3 Data Driven Interrupt Index 2 Bit Enable Register

Bit	Type	Description
[7:0]	RW	f any bit is 1, the corresponding bits of the index 2 byte will trigger the interrupt when it changes (0->1->1) if all bits are 0, the index 2 byte driven interrupt is disabled.

Offset 2DH/4DH/6DH/8DH(default=00)—Port 0~3 Data Driven Interrupt Index 3 Bit Enable Register

Bit	Type	Description
[7:0]	RW	If any bit is 1, the corresponding bits of the index 3 byte will trigger the interrupt when it changes (0->1->1) if all bits are 0, the index 3 byte driven interrupt is disabled.

Offset 2EH/4EH/6EH/8EH(default=00)—Port 0~3 Indirect Write Offset Register

Bit	Type	Description
[7:0]	RW	This register defined the target offset in the indirect write transaction.

Offset 2FH/4FH/6FH/8FH(default=00)—Port 0~3 Indirect Write Data Register

Bit	Type	Description
[7:0]	RW	This register defined the write data in the indirect write transaction. whenever this register is written, the indirect write is performed.

Offset 90H(default=00)—Indirect Write Status Register

Bit	Type	Description
[7:4]	RO	NACK flag for port 3~0. This flag is set indicating the indirect write sees NACK and was not successful. this bit stay until a new indirect write is started.
[3:0]	RO	DONE flag for port 3~0. This flag is set indicating the indirect write has completed successfully. this bit stay until a new indirect write is started. write transaction.

Offset 91H(default=00)—Common Indirect Write Select Register

Bit	Type	Description
[7:4]	RO	Reserved
[3:0]	RW	Specify which ports the common indirect write will access. for example, '1111' will set to all ports simultanously. '0101' will only write to prots 2 and 0.

Offset 92H(default=00)—Common Indirect Write Offset Register

Bit	Type	Description
[7:0]	RW	This register defined the target offset in the common indirect write transaction.

Offset 93H(default=00)—Common Indirect Write Data Register

Bit	Type	Description
[7:0]	RW	This register defined the write data in the common indirect write transaction. whenever this register is written, the common indirect write is performed.

Offset 94H(default=00)—Indirect Write Device Address Register

Bit	Type	Description
[7]	RW	Defines the I2C device ID for common indirect write. 1 = 0xA2; 0 = 0xA0
[6:4]	RO	Reserved
[3:0]	RW	Defines the I2C device ID for port(3~0) indirect write. 1 = 0xA2; 0 = 0xA0

Offset 95H(default=00)—I2C Bus Clear Register

Bit	Type	Description
[7:4]	RO	Reserved
[3:0]	RW	I2C bus clear for port 3~0. when set to 1, a clear sequence(9 clock cycles of SCL and a STOP condition) is started on the port. Bus is cleared when the sequence is done.

Offset 96H(default=00)—GPIO0 and GPIO1 Control Register

Bit	Type	Description
[7:4]	RW	GPIO1 control register: 0h = GPIO1 is set to input 1h = GPIO1 drives a 0 2h = GPIO1 drives a 1 3h = Not used.
[3:0]	RW	GPIO0 control register: 0h = GPIO0 is set to input 1h = GPIO0 drives a 0 2h = GPIO0 drives a 1 3h = Not used.

Offset 97H(default=00)—GPIO2 and GPIO3 Control Register

Bit	Type	Description
[7:4]	RW	GPIO3 control register: 0h = GPIO3 is set to input 1h = GPIO3 drives a 0 2h = GPIO3 drives a 1 3h = Not used.
[3:0]	RW	GPIO2 control register: 0h = GPIO2 is set to input 1h = GPIO2 drives a 0 2h = GPIO2 drives a 1 3h = Not used.

Offset 98H(default=00)—LED_CLOCK output Control Register

Bit	Type	Description
[7:2]	RO	Reserved
[1:0]	RW	LED clock output control: 0h = Output disabled 1h = Output LED clock 2h = Output a 0 3h = Output a 1

Offset 99H(default=00)—LED Sync Control Register

Bit	Type	Description
[7:4]	RWS	LED_YLW sync control for port 3~0: write F to reset the LED_YLW counters for all ports. it is self cleared after one clock.
[3:0]	RWS	LED_GRN sync control for port 3~0: write F to reset the LED_GRN counters for all ports. it is self cleared after one clock.

Offset 9AH(default=00)—Port SCL and SDA Stuck Timeout Control Register

Bit	Type	Description
[7:4]	RW	When set,enable port 3~0 fast mode for simulation only

[3:0]	RW	SCL and SDA stuck timeout control for port 3~0: setting 1 disable the SCL and SDA stuck watchdog timer for the port. This register cannot be programmed to 0xFF.

Offset 9BH(default=00)—Port SCL Stuck Interrupt Control Register

Bit	Type	Description
[7:4]	RO	Port SCL stuck indicator for port 3~0. This bit is cleared when bus reset is executed. (00H[3:0])
[3:0]	RW	Set to 1 to enable interrupt from SCL stuck for port 3~0.

Offset 9CH(default=00)—Port SDA Stuck Interrupt Control Register

Bit	Type	Description
[7:4]	RO	Port SDA stuck indicator for port 3~0. This bit is set when SDA stuck at low for 1s. it is cleared when bus reset is executed.(00H[3:0])
[3:0]	RW	Set to 1 to enable interrupt from SDA stuck for port 3~0.

Offset 9DH/9EH/9FH/A0H(default=23)—Port 0~3 Protocol Timeout Register

Bit	Type	Description
[7:0]	RW	During direct access,when I2C slave is waiting for host I2C to send register address or read/write command and this timer expires, the I2C master will enter idle state. This timer is in ms. This register cannot be programmed to 0xFF.

Offset A1H/A2H/A3H/A4H(default=23)—Port 0~3 SCL Stuck Timeout Register

Bit	Type	Description
[7:0]	RW	When port's SCL is stretched to low and this timer expires, it will generate a interrupt if is enabled by register 0x9B. This timer is in ms. This register cannot be programmed to 0xFF.

Offset A5H/A6H/A7H/A8H(default=00)—Port 0~3 NACK Counter Register

Bit	Type	Description
[7:0]	ROC	Counter the number of received NACKs at the port. It saturates at FFh and clear when read.

Offset A9H/AAH/ABH/ACH(default=23)—Port 0~3 Port I2C Watchdog Timer Register

Bit	Type	Description
[7:0]	RW	Port side I2C watchdog timer, if the I2C transaction does not complete within this timer, I2C master will enter idle state. This timer is in ms. This register cannot be programmed to 0xFF

Offset ADH/AEH/AFH/B0H(default=00)—Port 0~3 Prefetch NACK Counter Register

Bit	Type	Description
[7:0]	ROC	Counter the number of received NACKs at the port during prefetching. It saturates at FFh and clear when read.

Offset B1H(default=00)—SPI Burst Mode Control Register

Bit	Type	Description
[7:4]	RO	Reserved

[3]	RW	When set to 1, 1/250 division is used for LED base clock, when set to 0, 1/255 division is used for LED base clock
[2:1]	RW	Reserved
[0]	RO	When set to 1, it enables the SPI continous access feature.

Offset C0H(default=00)—OSC Status Register 0

Bit	Type	Description
[7:0]	RO	OSC status byte-0

Offset C1H(default=00)—OSC Status Register 1

Bit	Type	Description
[7:0]	RO	OSC status byte-1

Offset C2H(default=00)—OSC Control Register 0

Bit	Type	Description
[7:0]	RW	OSC control byte-0

Offset C3H(default=00)—OSC Control Register 1

Bit	Type	Description
[7:0]	RW	OSC control byte-1

Offset C4H(default=00)—OSC Control Register 2

Bit	Type	Description
[7:0]	RW	OSC control byte-2

Offset D0H(default=19)—Port 0~3 Debounce Register Lower Byte

Bit	Type	Description
[7:0]	RW	Port-0~3 input debounce timer bit [7:0] This timer is in 2us.

Offset D1H/D3H/D5H/D7H(default=00)—Port 0~3 Debounce Register Higher Byte

Bit	Type	Description
[7:0]	RW	Port-0 input debounce timer bit [15:8] This timer is in 2us.

Offset D8H/DAH/DCH/DEH(default=0A)—Port 0~3 I2C Bus Idle Register Lower Byte

Bit	Type	Description
[7:0]	RW	Port-0~3 I2C bus idle timer bit [7:0] This timer is in 2us.

Offset D9H/DBH/DDH/DFH(default=00)—Port 0~3 I2C Bus Idle Register Higher Byte

Bit	Type	Description
[7:0]	RW	Port-0~3 I2C bus idle timer bit [15:8] This timer is in 2us.

Offset F0H(default=00)—Revision Register

Bit	Type	Description
[7:0]	RO	The revision of this version.

Offset F1H(default=01)—Device ID Low Byte Register

Bit	Type	Description
[7:0]	RO	The device ID (combined with 0xF2) is 0x1401

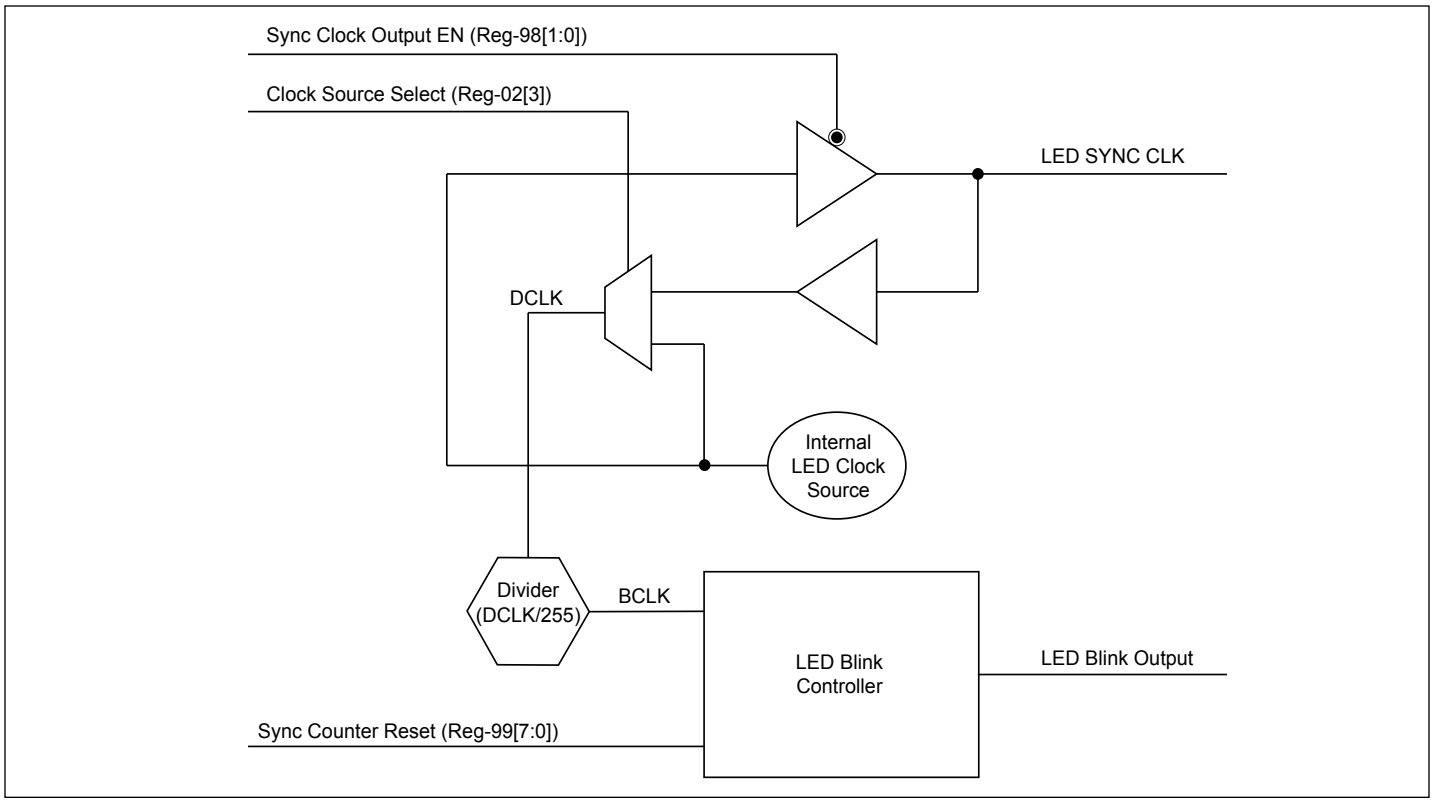
Offset F2H(default=14)—Device ID High Byte Register

Bit	Type	Description
[7:0]	RO	The device ID (combined with 0xF1) is 0x1401

11 Programming Guide

11.1 LED SYNC

11.1.1 Block Diagram



500ms LED On/Off Blinking Program Example: (DCLK = 102KHz, BCLK = 400Hz)

Set Reg-16/36/56/76 = 200; Reg-17/37/57/77 = 200;

LED on time = 200 x 2.5ms = 500ms

LED off time = 200 x 2.5ms = 500ms

11.1.2 SPI Mode LED Synchronization Sequence

(1) All devices' LED turns off

Register 1AH/3AH/5AH/7AH = 0x30

(2) Set external LED sync clock as source clock for each devices

Register 02H = 0x08

(3) Set LED counter registers for each device (same values recommend)

Register 14H/34H/54H/74H LED_1 PWM counter

Register 15H/35H/55H/75H LED_2 PWM counter

Register 16H/36H/56H/76H LED_1 Blink On counter

Register 17H/37H/57H/77H LED_1 Blink Off counter

Register 18H/38H/58H/78H LED_2 Blink On counter

Register 19H/39H/59H/79H LED_2 Blink Off counter

(4) All of devices' LED blinking counter reset

Register 99H = 0xFF

(5) Set all devices as LED BLINK mode

Register 1AH/3AH/5AH/7AH = 0x33

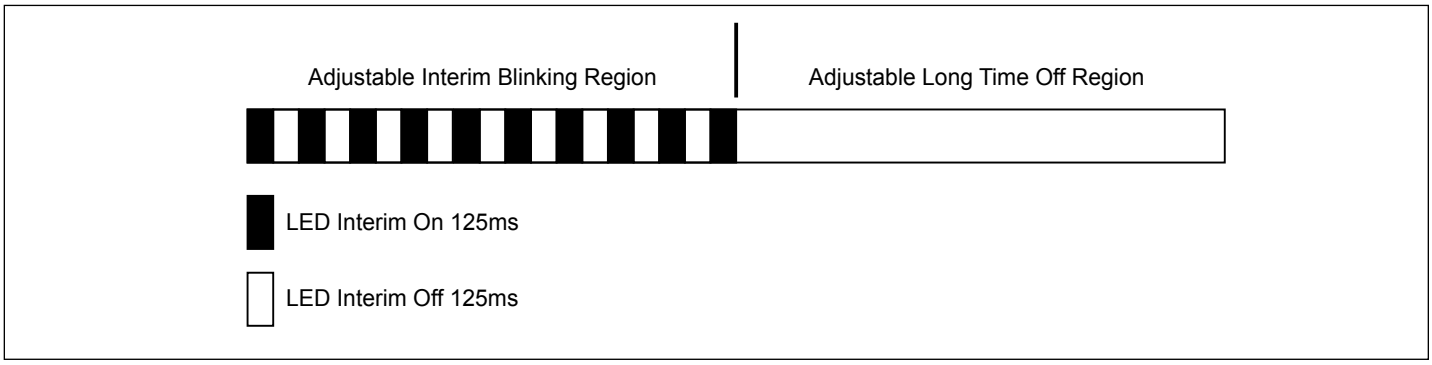
(6) Set any one device to enable LED clock output

Register 98H = 0x01

Note: Only one device acceptable, LED sync clock will output from SPI_LED_CLK pin

11.2 LED Long Time Interim Blinking

11.2.1 Blink Pattern



11.2.2 Configuration

(1) LED Mode Control Registers

1AH/3AH/5AH/7AH [7:6] Port 0~3 LED_2/1 long-time interim blink mode enable

1AH/3AH/5AH/7AH [5:4] Port 0~3 LED_2/1 LED output active signal invert

1AH/3AH/5AH/7AH [3:2] Port 0~3 LED_2 mode select

1AH/3AH/5AH/7AH [1:0] Port 0~3 LED_1 mode select

(2) LED On Time Control Registers

16H/36H/56H/76H Port 0~3 LED_1 On time set, unit 10ms on long time mode

18H/38H/58H/78H Port 0~3 LED_2 On time set, unit 10ms on long time mode

(3) LED Off Time Control Registers

17H/37H/57H/77H Port 0~3 LED_1 Off time set, unit 10ms on long time mode

19H/39H/59H/79H Port 0~3 LED_2 Off time set, unit 10ms on long time mode

(4) LED Brightness Control Registers

14H/34H/54H/74H Port 0~3 LED_1 PWM dimmer

15H/35H/55H/75H Port 0~3 LED_2 PWM dimmer

11.2.3 Programming Examples

(1) Slower Blinking (0.5Hz blinking: 1s on, 1s off)

Set 1AH = 0x73 for Port-0 LED-GRN long time blink mode

Set 14H = 0xE5 for Port-0 LED-GRN PWM 90% brightness

Set 16H = 0x64 for Port-0 LED-GRN 1000ms On

Set 17H = 0x64 for Port-0 LED-GRN 1000ms Off

(2) Faster Blinking (4Hz blinking: 125ms on, 125ms off) Set 1AH = 0x33 for Port-0 LED-GRN normal blink mode

Set 14H = 0xE5 for Port-0 LED-GRN PWM 90% brightness

Set 16H = 0x32 for Port-0 LED-GRN 125ms On

Set 17H = 0x32 for Port-0 LED-GRN 125ms Off

(3) Interim Blinking (2s in 4Hz blinking, 2s off)

Set 1AH = 0xF3 for Port-0 LED-GRN long time interim blink mode

Set 14H = 0xE5 for Port-0 LED-GRN PWM 90% brightness

Set 16H = 0xC8 for Port-0 LED-GRN 2000ms On

Set 17H = 0xC8 for Port-0 LED-GRN 2000ms Off

(4) Short Blinking (0.05s on, 1.95s off)

Set 1AH = 0x73 for Port-0 LED-GRN long time blink mode

Set 14H = 0xE5 for Port-0 LED-GRN PWM 90% brightness

Set 16H = 0x05 for Port-0 LED-GRN 50ms On

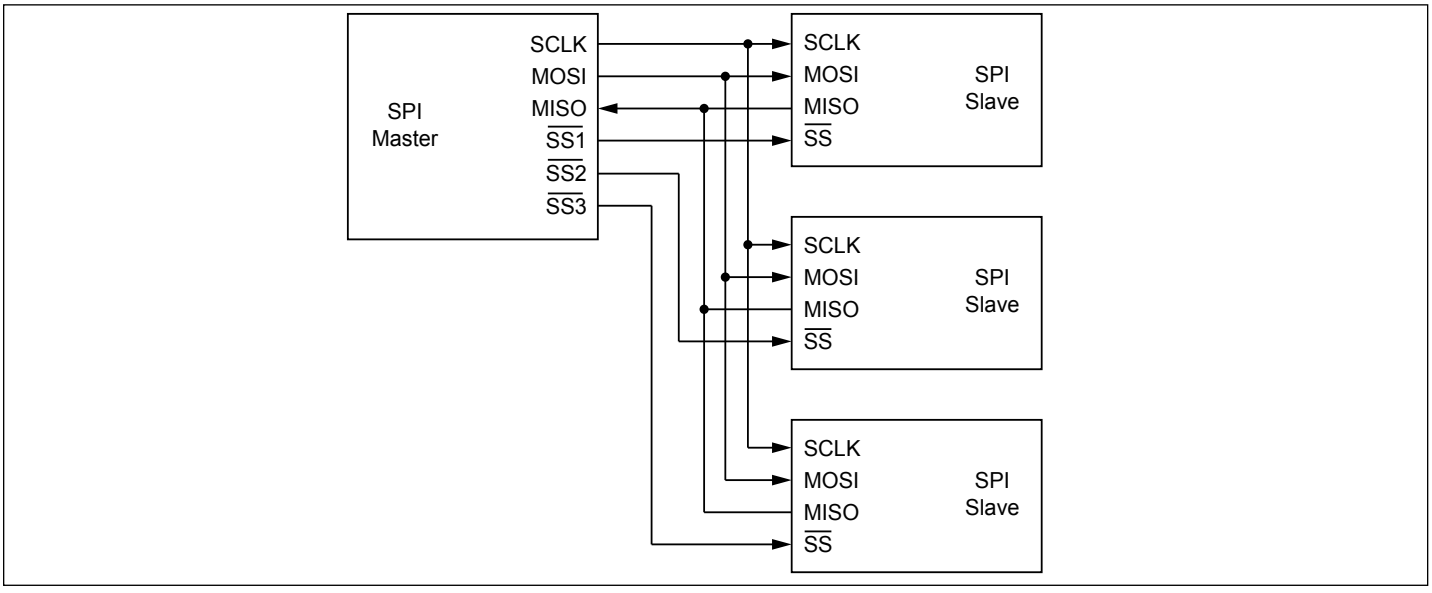
Set 17H = 0xC3 for Port-0 LED-GRN 1950ms Off

11.3 SPI Burst Read/Write

SPI burst read/write mode is for independent SPI bus system only.

SPI burst mode support the fastest process to continue register range by one transaction

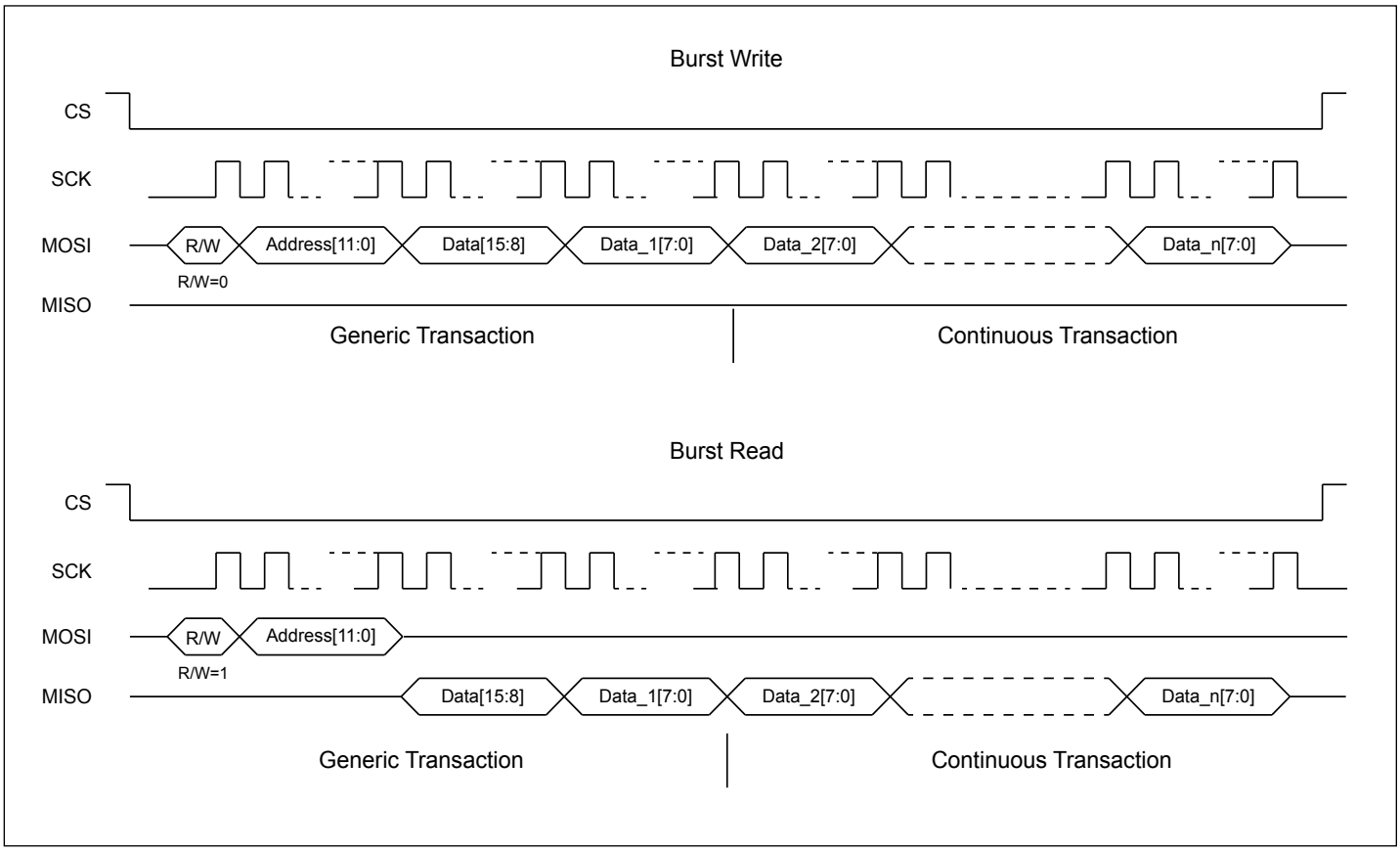
11.3.1 Typical independent SPI Bus configuration



11.3.2 SPI Burst Read/ Write Frame Structure

Bit	Field	Description
28	R/W	0: Write, 1:Read
27:16	ADDR	12-bit Address
15	DATA[15]	Busyflag
14	DATA[14]	Don't care
13	DATA[13]	NACK flag
12	DATA[12]	Reject flag
11:8	DATA[11:8]	Don't care
7:0	DATA[7:0]	The first byte 8-bit data
7:0	DATA[7:0]	Second byte 8-bit data (continue R/W)
:	:	:
7:0	DATA[7:0]	The end of byte 8-bit data

11.3.3 SPI Burst Read/Write Transaction Waveform



11.3.3 SPI Burst Read/Write Transaction Waveform

The following example read 32-bytes pre-fetch data in burst mode.

(1) Pre-fetch downstream port-0 0xA0 device's 32-bytes data from register address 0x00 to 0x1F

Set device register 1DH = 0xFA

(w 0x81d: 0xfa)

(2) Enable SPI burst mode

Set device register B1H = 0x01

(w 0x8b1: 0x01)

(3) Continuers read out 32-bytes data form pre-fetch RAM in burst mode

Read downstream port-0 0xA0 pre-fetch RAM's data from 0x00 to 0x1F

(r 0x000 from 0x00 to 0x1f)

(4) Disable SPI burst mode

Set device register B1H = 0x00

(w 0x8b1: 0x00)

11.4 Full Range Downstream Port's Device ID Support

The downstream ports' 0xA0 device ID can be modified by "Port 0~3 Downstream I2C Device ID Modifier Register". When "full rang function enable" bit be set, all of the "in-direct write", "common in-direct write", "interrupt pre-fetch" and "periodic pre-fetch" etc. downstream ports' operation can support full rang ID also.

11.4.1 Full Range Device ID Enable Control Registers

13H/33H/53H/73H [7:3] Port 0~3 downstream port 0~3 Device ID [7:3]

13H/33H/53H/73H [1] Port 0~3 downstream port 0~3 Device ID [2]

Note: 13H/33H/53H/73H [0] Reserved, should be 0

1BH/3BH/5BH/7BH [2] Port 0~3 full range function enables select

(1) When "full range function enable": (1BH/3BH/5BH/7BH [2] = 1)

Direct read/write down port device ID: Device ID [7:0]

In-direct write down port device ID: Device ID [7:0]/0xA2

Common in-direct write down port device ID: Device ID [7:0]/0xA2

Interrupt pre-fetch down port device range: Device ID [7:0]/0xA2

Periodic pre-fetch down port device ID: Device ID [7:0]/0xA2

(2) When "full range function disable": (1BH/3BH/5BH/7BH [2] = 0)

Direct read/write down port device ID: Device ID [7:0]

In-direct write down port device ID: 0xA0/0xA2

Common in-direct write down port device ID: 0xA0/0xA2

Interrupt pre-fetch down port device range: 0xA0/0xA2

Periodic pre-fetch down port device ID: 0xA0/0xA2

11.5 Optical Module Hot-Plug Interrupt Trigger Delay

When “hot-plug interrupt delay” enable, (20H/40H/60H/80H [7] = 1) the downstream port_0~3’s hot-plug interrupt trigger signal will be delay 2 seconds.

11.6 Tx-FAULT Input Window Gate

When “Tx-FAULT window gate” enable, (20H/40H/60H/80H [6] = 1) forces the downstream port_0~3’s Tx-FAULT input signal to go through a 300ms window gate. Once the Tx-DISABLE output form high to low, the window gate will turns off immediately to stop (ignore) Tx-FAULT input, and then after 300ms delay, the gate turns on automatically to allow the Tx-FAULT signal pass through.

11.7 Downstream Input Ports’ De-bouncer

Each of downstream input port has independently de-bounced circuit.

The timer could be set from 2us to 130ms:

D0H/D2H/D4H/D6H Port_0~3 de-bounce timer register [7:0]

D1H/D3H/D5H/D7H Port_0~3 de-bounce timer register [15:8]

11.8 Downstream Ports’ I2C Bus Idle Time

Each of downstream I2C master has independently bus idle timer.

The timer could be set from 2us to 130ms:

D8H/DAH/DCH/DEH Port_0~3 bus idle timer register [7:0]

D9H/DBH/DDH/DFH Port_0~3 bus idle timer register [15:8]

Electrical Specification

12 Maximum Ratings

(Above which useful life may be impaired. For user guidelines, not tested.)

Power Supply Range	4V
Voltage at Any Pin	GND-0.3V to 5.5V
Storage Temperature	–65°C to +150°C
Junction Temperature	125°C
Package Dissipation	500 mW

Note: Stresses greater than those listed under MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

12.1 DC Electrical Characteristics

(T_A = -40C° to + 85°C, V_{DD} = 1.62V - 3.63V)

Symbol	Parameter	V_{DD} = 1.8V ± 10%		V_{DD} = 3.3V± 10%		Unit	Conditions
		Min.	Max.	Min.	Max.		
V_{IL}	Input low voltage	-0.3	0.2	-0.3	0.8	V	
V_{IH}	Input high voltage	1.4	5.5	2.0	5.5	V	
V_{OL}	Output low voltage		0.4		0.4	V V V	I_{OL} = 4 mA I_{OL} = 2 mA I_{OL} = 1.5 mA
V_{OH}	Output high voltage	1.4		2.0		V V V	I_{OH} = -1 mA I_{OH} = -400 uA I_{OH} = -200 uA
I_{IL}	Input low leakage current		10		10	uA	
I_{IH}	Input high leakage current		10		10	uA	
C_{IN}	Input pin capacitance		5		5	pF	
I_{CC}	Power supply current		0.02@VDD_P		15@VDD_S	mA	

Note: 5.5V steady voltage tolerance on inputs and outputs is valid only when the supply voltage is present.

12.2 AC Electrical Characteristics

(T_A = -40C° to + 85°C, V_{DD} = 1.62V - 3.63V)

Symbol	Parameter	Test Condidtion	Min	Typ	Max	Unit
General Timing Requirements						
TPOR	Internal power-on reset (PoR) time	Time between stable VDD_S power supply (VDD_S ≥ 3.3V-5%) and deassertion of internal PoR. The port-side and host-side control interfaces (I2C and/or SPI) are not operational during this time.			50	ms
Host-Side SPI Timing Requirements (I2C/SPI# = GND)(1)(2)						
f_{SPI}			0.1		50	MHz
f_{HI-SCK}				0.4÷f_{SPI}		ns
f_{LO-SCK}				0.4÷f_{SPI}		ns
$t_{HD-MOSI}$				1		ns
$t_{SU-MOSI}$				1		ns
tHD-SSN				4		ns
t_{SU-SSN}				1.2		ns
$t_{OFF-SSN}$		For writes and local PI7C1401 register reads		1		µs
		For consecutive downstream (remote) register reads on the same port, assuming 400KHz I2C		125		
		For consecutive downstream (remote) register reads on the same port, assuming 100KHz I2C		465		
$t_{ODZ-MISO}$	MISO driven-to-TRI_STATE time			20		ns
$t_{OZD-MISO}$	MISO TRI_STATE-to-driven time			5		ns
t_{OD}	MISO output delay time			7		ns
Host-Side I2C Timing Requirements (I2C/SPI# = Float or High)(2)(3)(4)						
f_{SCL}	Host-side I2C clock frequency in I2C mode		100		1000	kHz
t_{BUF}	Bus free time between STOP and START condition		0.5			µs
t_{HD-STA}	Hold time after (repeated) START condition. After this period, the first clock is generated.	After this period, the first clock can be gernerated by the master.	0.3			µs
t_{SU-STA}	Repeated START condition setup time		0.3			µs
t_{SU-STO}	STOP condition setup time		0.3			µs

AC Electrical Characteristics Cont.

Symbol	Parameter	Test Condidtion	Min	Typ	Max	Unit
t_{HD-DAT}	SDA hold time		0			ns
t_{SU-DAT}	SDA setup time	Applies to standatd-mode I2C, 100kHz	250			ns
	SDA setup time	Applies to fast-mode I2C, 400kHz	100			ns
	SDA setup time	Applies to fast-mode plus I2C, 1000kHz	50			ns
t_{LOW}	SCL clock low time		0.5			µs
t_{HIGH}	SCL clock high time		0.3			µs
t_R	SDArise time, read	Applies to standard-mode I2C, 100kHz			1000	ns
	SDA rise time, read	Applies to fast-mode I2C, 400kHz	20		300	ns
	SDA rise time, read	Applies to fast-mode plus I2C, 1000kHz			120	ns
t_F	SDA fall time, read	Applies to standard-mode I2C, 100kHz			300	ns
	SDA fall time, read	Applies to fast-mode I2C, 400kHz	4.4		300	ns
	SDA fall time, read	Applies to fast-mode plus I2C, 1000kHz	4.4		120	ns
Downstram Master I2C Switching Characteristics						
f_{SCL}	SCL clock frequency	Applies to standard-mode I2C, 100kHz	88	93	100	kHz
		Applies to fast-mode I2C, 400kHz	357	372	400	kHz
$t_{LOW-SCL}$	SCL clock pluse width low period		1.3			µs
$t_{HIGH-SCL}$	SCL clock pluse width high period		0.6			µs
t_{BUF}	Time bus free before new transmission starts	Between STOP and START and between ACK and RESTART	20			µs
t_{HD-STA}	Hold time START operation		0.6			µs
t_{SU-STA}	Setup time START operation		0.6			µs
t_{HD-DAT}	Data hold time		0			µs
t_{SU-DAT}	Data setup time		0			µs
tR	SCL and SDA rise time	100KHz operation. From V_{IL}(Max) - 0.15V to V_{IH}(Min) + 0.15V			300	ns
	SCL and SDA rise time	100KHz operation. From V_{IL}(Max) - 0.15V to V_{IH}(Min) + 0.15V			300	

AC Electrical Characteristics Cont.

Symbol	Parameter	Test Condidtion	Min	Typ	Max	Unit
tF	SCL and SDA fall time	100KHz operation. From V_{IH}(Max) + 0.15V to V_{IL}(Min) - 0.15V			300	ns
	SCL and SDA fall time	100KHz operation. From V_{IH}(Max) + 0.15V to V_{IL}(Min) - 0.15V			300	
t_{SU-STO}	STOP condition setup time		0.6			µs
t_{SP-I2C}(5)	Pulse wwidth of spikes that are suppressed by PI7C1401		0		50	ns

Note:

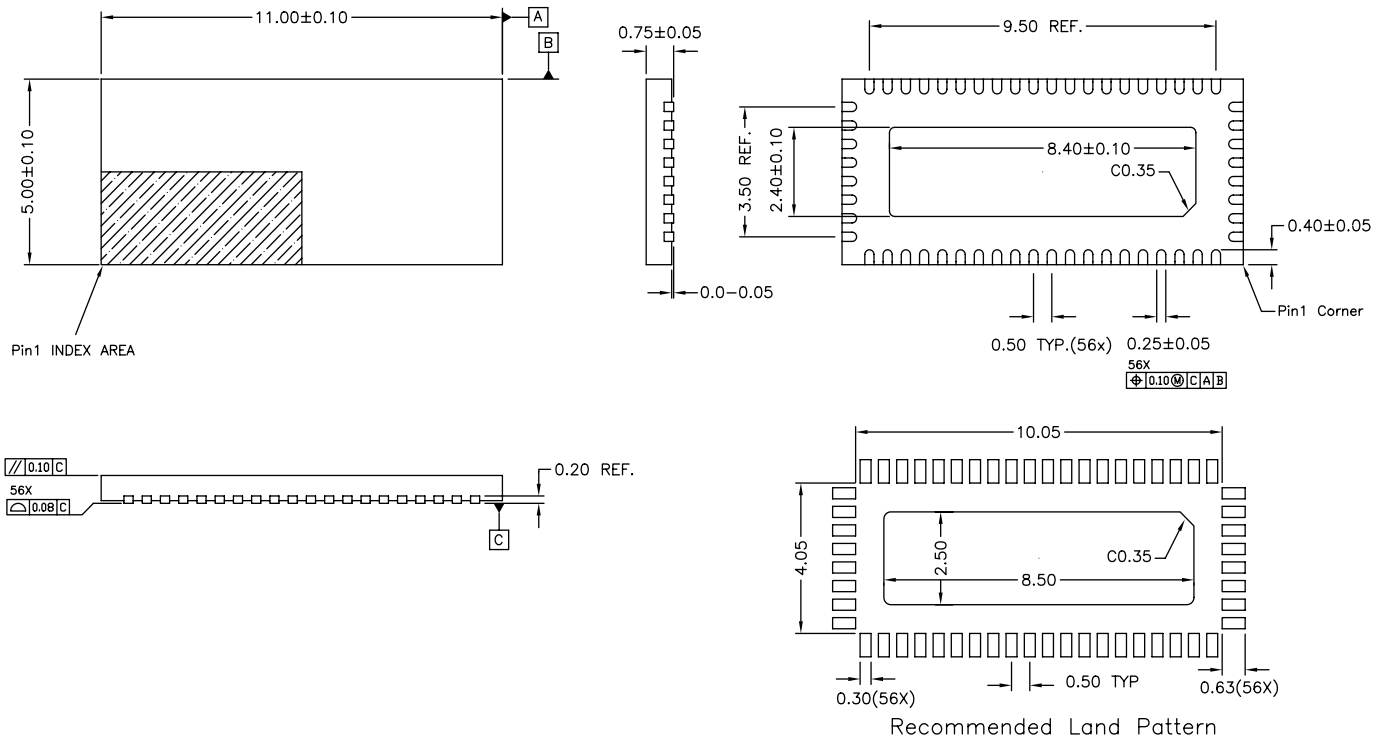
1. SPI operation is available T_{POR} milliseconds after VDD_S power up, provided RST# = high or float and VDD_P is stable.
2. These parameters are not production tested.
3. I2C operation is available T_{POR} milliseconds after VDD_S power up, provided RST# = high or float and VDD_P is stable.
4. These specifications support I2C Rev 6 specifications.
5. These parameters are not production tested.

Part Marking

ZF Package

YY: Year
WW: Workweek
1st X: Assembly Code
2nd X: Fab Code

Packaging Mechanical: 56-TQFN (ZF)



NOTE :
1. ALL DIMENSION ARE IN mm. ANGLES IN DEGREES.
2. BILATERAL COPLANARITY ZONE APPLIES TO THE EXPOSED HEAT SINK SLUG AS WELL AS THE TERMINALS.
3. REFER JEDEC MO−220 MODIFIED.
4. Thermal Via Diameter. Recommended 0.2~0.33mm
5. Thermal Via Pitch. Recommended 1.27mm

PERICOM Semiconductor Corporation — DATE: 05/15/08

DESCRIPTION: 56-contact, Thin Fine Pitch Quad Flat No-lead (TQFN)

PACKAGE CODE: ZF56

DOCUMENT CONTROL #: PD-2024 — REVISION: C

08-0208

For latest package info.

please check: http://www.diodes.com/design/support/packaging/pericom-packaging/packaging-mechanicals-and-thermal-characteristics/

Ordering Information

Ordering Number	Package Code	Package Description
PI7C1401AZFEX	ZF	56-contact, Thin Fine Pitch Quad Flat No-lead (TQFN)

Notes:

1. No purposely added lead. Fully EU Directive 2002/95/EC (RoHS), 2011/65/EU (RoHS 2) & 2015/863/EU (RoHS 3) compliant.
2. See https://www.diodes.com/quality/lead-free/ for more information about Diodes Incorporated's definitions of Halogen- and Antimony-free, "Green" and Lead-free.
3. Halogen- and Antimony-free "Green" products are defined as those which contain <900ppm bromine, <900ppm chlorine (<1500ppm total Br + Cl) and <1000ppm antimony compounds.
4. E = Pb-free and Green
5. X suffix = Tape/Reel

IMPORTANT NOTICE

DIODES INCORPORATED MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARDS TO THIS DOCUMENT, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION).

Diodes Incorporated and its subsidiaries reserve the right to make modifications, enhancements, improvements, corrections or other changes without further notice to this document and any product described herein. Diodes Incorporated does not assume any liability arising out of the application or use of this document or any product described herein; neither does Diodes Incorporated convey any license under its patent or trademark rights, nor the rights of others. Any Customer or user of this document or products described herein in such applications shall assume all risks of such use and will agree to hold Diodes Incorporated and all the companies whose products are represented on Diodes Incorporated website, harmless against all damages.

Diodes Incorporated does not warrant or accept any liability whatsoever in respect of any products purchased through unauthorized sales channel.

Should Customers purchase or use Diodes Incorporated products for any unintended or unauthorized application, Customers shall indemnify and hold Diodes Incorporated and its representatives harmless against all claims, damages, expenses, and attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized application.

Products described herein may be covered by one or more United States, international or foreign patents pending. Product names and markings noted herein may also be covered by one or more United States, international or foreign trademarks.

This document is written in English but may be translated into multiple languages for reference. Only the English version of this document is the final and determinative format released by Diodes Incorporated.

LIFE SUPPORT

Diodes Incorporated products are specifically not authorized for use as critical components in life support devices or systems without the express written approval of the Chief Executive Officer of Diodes Incorporated. As used herein:

A. Life support devices or systems are devices or systems which:

1. are intended to implant into the body, or

2. support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in significant injury to the user.

B. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or to affect its safety or effectiveness.

Customers represent that they have all necessary expertise in the safety and regulatory ramifications of their life support devices or systems, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of Diodes Incorporated products in such safety-critical, life support devices or systems, notwithstanding any devices- or systems-related information or support that may be provided by Diodes Incorporated. Further, Customers must fully indemnify Diodes Incorporated and its representatives against any damages arising out of the use of Diodes Incorporated products in such safety-critical, life support devices or systems.

www.diodes.com

Revision History

Date	Revision Number	Description
2017/08/30	1	Revision numbering system changed to whole number
2018/03/12	2	Updated Pin Description UPdated 1.2.3 SPI Read Latency
2018/07/04	3	Updated Feature Updated Section 10 Register Map Added Section 11 Programming Guide Change Section 11 Maximum Ratings to Section 12 Updtaed Ordering Information, added PI7C1401A
2019/01/22	4	Added Part Marking Web Released