INDUSTRIAL CONTROL COMMUNICATIONS, INC.

SPI Slave Driver Manual

October 3, 2019

TABLE OF CONTENTS

1 SPI Slave

1.1 Overview

This driver allows the PicoPort to act like a memory device on the SPI bus. The internal database is exposed as addressable memory and may be directly accessed using any of three different data types: byte (8-bit), short (16-bit), and long (32-bit). All operation instructions use a common, 5-byte format (refer to Table 1). However, because the SPI functionality is implemented in software instead of hardware, data cannot be read and written in a single instruction cycle. For this reason, the SPI driver must transition through different states to read and write data. The SPI operation instructions and state machine are described below.

1.2 Operation Instructions

Table 1: SPI Operation Instructions

Bus Cycle[1]	1		2		3		4		5	
Operation	**S_{IN}**	**S_{OUT}**	**S_{IN}**	**S_{OUT}**	**S_{IN}**	**S_{OUT}**	**S_{IN}**	**S_{OUT}**	**S_{IN}**	**S_{OUT}**
GS (Get Status)	0x01	STATUS[6]	X	D_{31}-D_{24}[7]	X	D_{23}-D_{16}[7]	X	D_{15}-D_{8}[7]	X	D_{7}-D_{0}[7]
SA (Set Address)[2,3,4]	0x11	STATUS[6]	0	X	0	X	A_{15}-A_{8}	X	A_{7}-A_{0}	X
RB (Read Byte)[2,3,5]	0x21	STATUS[6]	X	X	X	X	X	X	X	X
RS (Read Short)[2,3,5]	0x22	STATUS[6]	X	X	X	X	X	X	X	X
RL (Read Long)[2,3,5]	0x24	STATUS[6]	X	X	X	X	X	X	X	X
WB (Write Byte)[2,3,5]	0x41	STATUS[6]	0	X	0	X	0	X	D_{7}-D_{0}	X
WS (Write Short)[2,3,5]	0x42	STATUS[6]	0	X	0	X	D_{15}-D_{8}	X	D_{7}-D_{0}	X
WL (Write Long)[2,3,5]	0x44	STATUS[6]	D_{31}-D_{24}	X	D_{23}-D_{16}	X	D_{15}-D_{8}	X	D_{7}-D_{0}	X

Notes

1. One bus cycle is comprised of eight clock periods.
2. This asynchronous operation is started when ACK = 1.
3. During this operation, the device will be busy and no new operations can be started (ACK = 0, STATE = 1). The device status may be polled by performing the Get Status operation until the device is no longer busy and can accept commands (ACK = 1).
4. This operation is complete when the device enters the Ready state (ACK = 1, STATE = 2).
5. This operation is complete when the device enters the Operation Complete state (ACK = 1, STATE = 3).
6. Refer to Table 2 for an overview of the STATUS register.
7. This data is invalid and don't care except when ACK = 1 and STATE = 3. If ERR = 0, this data will be the value from the completed operation. If ERR = 1, this data will be the error code (refer to Table 3).

Table 2: SPI STATUS Register

Bit	Name	Function	Default on Reset
0	ACK[1]	1 = The command was accepted 0 = The command was ignored, the device is processing a previous command	1
1	ERR	1 = An error occurred while performing the operation, the error code is encoded in the output data 0 = No errors have occurred	0
5:2	RESERVED	Reserved for future use	0
7:6	STATE[2]	0 = Reset 1 = Busy 2 = Ready 3 = Operation Complete	0

Notes

1. The ACK bit indicates whether or not the PicoPort is currently accepting commands, and therefore, whether or not the sent command was acknowledged. If a command was not acknowledged, it should be sent again until it is acknowledged.
2. The STATE indicates the state in which the PicoPort was in prior to receiving the command.

Table 3: SPI Error Codes

Error Code[1]	Error	Description
0xF0	Invalid Data Address	The address set in the Set Address command is invalid.
0xF1	Data Error	There is an error in a data field.
0xF2	Write To Read-Only	A write was performed to a read-only address.
0xF3	Read From Write-Only	A read was performed to a write-only address.
0xFB	Invalid Function	The command used is not understood by the device.
0xFC	Invalid Packet	The packet received is invalid.

Notes

1. The error code will appear in D_7-D_0. D_{31}-D_8 will have a value of 0.

1.3 State Machine

1.3.1 State Transition Diagram

All SPI transactions occur in accordance with the state transition diagram shown in Figure 1.

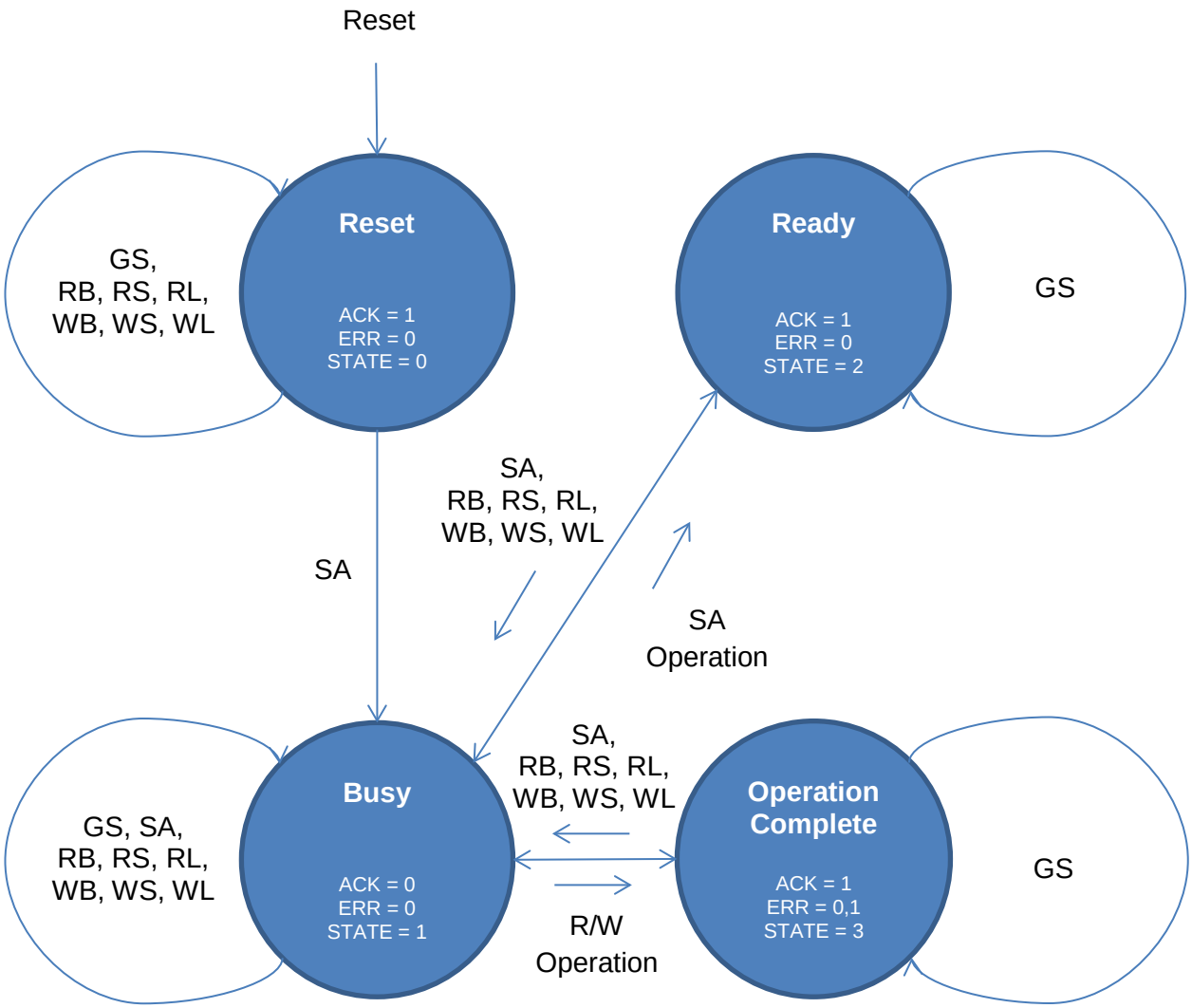

Figure 1: SPI State Transition Diagram

1.3.2 State Descriptions

This section describes each state the PicoPort can transition through during SPI communications.

1.3.2.1 Reset

The Reset state is entered upon power up. The STATUS register will always be 0x01 (ACK = 1, ERR = 0, STATE = 0) while in this state. The Set Address command is the only command that will cause the PicoPort to leave the Reset state and enter the Busy state. All other commands have no effect and the PicoPort will stay in the Reset state.

1.3.2.2 Busy

The PicoPort enters the Busy state after accepting any command except the Get Status command. The STATUS register will always be 0x40 (ACK = 0, ERR = 0, STATE = 1) and all additional commands are ignored while in this state. When the PicoPort finishes processing the Set Address command, it will enter the Ready state. When the PicoPort finishes processing any other command, it will enter the Operation Complete state.

1.3.2.3 Ready

The Ready state is entered when the PicoPort has completed the Set Address command and is accepting new commands. The STATUS register will always be 0x81 (ACK = 1, ERR = 0, STATE = 2) while in this state. Any command except the Get Status command will cause the PicoPort to leave the Ready state and enter the Busy state.

1.3.2.4 Operation Complete

The PicoPort enters the Operation Complete state when it has finished processing a read or write command. In this state, the ERR bit in the STATUS register may be set to indicate an error in processing the command. The STATUS register will be 0xC1 (ACK = 1, ERR = 0, STATE = 3) if no errors have occurred, or 0xC3 (ACK = 1, ERR = 1, STATE = 3) if an error has occurred. While the PicoPort is in the Operation Complete state, the data bytes will be the result of the completed command (if ERR = 0) or the error code (if ERR = 1). Any command except the Get Status command will cause the PicoPort to leave the Operation Complete state and enter the Busy state.

1.4 Instruction Transaction Examples

Table 4: SPI Read after Reset Example

SPI Transfer	Command	MOSI Data	STATUS Register			MISO Data
			ACK	ERR	STATE	
1	Get Status	X	1	0	Reset	X
2	Set Address	Address	1	0	Reset	X
3	Get Status	X	0	0	Busy	X
4	Get Status	X	1	0	Ready	X
5	Read Byte	X	1	0	Ready	X
6	Get Status	X	0	0	Busy	X
7	Get Status	X	1	0	Operation Complete	Read Data

Table 5: SPI Write after Read Example

SPI Transfer	Command	MOSI Data	STATUS Register			MISO Data
			ACK	ERR	STATE	
1	Get Status	X	1	0	Operation Complete	Read Data
2	Set Address	Address	1	0	Operation Complete	Read Data
3	Get Status	X	0	0	Busy	X
4	Get Status	X	1	0	Ready	X
5	Write Byte	Write Data	1	0	Ready	X
6	Get Status	X	0	0	Busy	X
7	Get Status	X	1	0	Operation Complete	Write Data

Table 6: SPI Read Error after Read Example

SPI Transfer	Command	MOSI Data	STATUS Register			MISO Data
			ACK	ERR	STATE	
1	Get Status	X	1	0	Operation Complete	Read Data
2	Set Address	Address	1	0	Operation Complete	Read Data
3	Get Status	X	0	0	Busy	X
4	Get Status	X	1	0	Ready	X
5	Read Byte	X	1	0	Ready	X
6	Get Status	X	0	0	Busy	X
7	Get Status	X	1	1	Operation Complete	Error Code

1.5 SPI Timing

This section shows timing diagrams for SPI transactions and timing parameters which must be observed by the SPI master. Figure 2 shows the typical transaction timing when performing an SPI operation. Note that each SPI operation takes 5 bus cycles and each bus cycle corresponds to 8 clock periods.

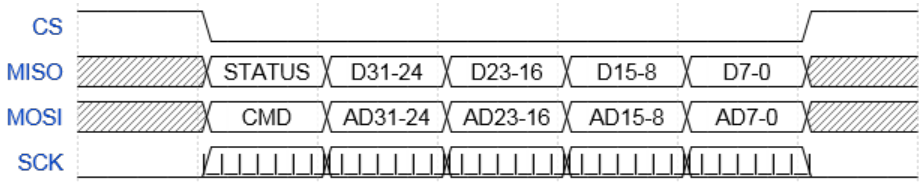

Figure 2: SPI Operation Transaction

Figure 3 shows the timing of character transactions and the corresponding timing parameters which must be observed. The timing parameters are described in section 1.5.1.

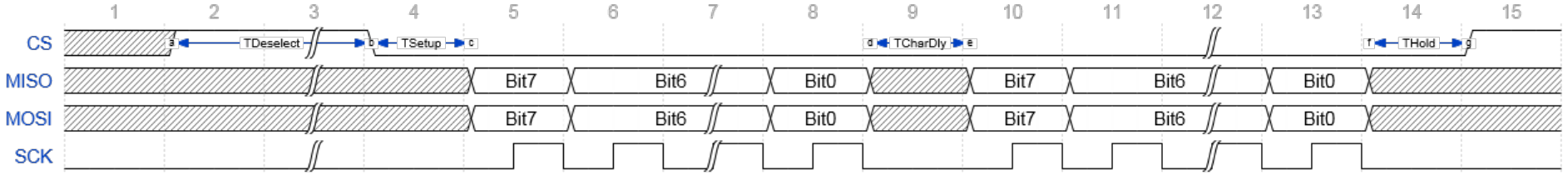

Figure 3: SPI Character Transaction Timing

1.5.1 SPI Timing Parameters

Clock Frequency ≤ 5MHz
This is the maximum supported frequency of the clock signal on the SCK pin.

$T_{Deselect}$ ≥ 1 Clock Period and $T_{Deselect}$ ≥ 150µs
This is the minimum time the chip select signal must be held high (deselected) before being asserted low (selected).

T_{Setup} ≥ 1 Clock Period
This is the minimum time the chip select signal must be held low (selected) before the start of a transaction.

$T_{CharDly}$ ≥ 1 Clock Period
This is the minimum time the SPI master must delay between transmitting characters.

T_{Hold} ≥ 1 Clock Period
This is the minimum time the chip select signal must be held low (selected) after the end of a transaction.

ICC

INDUSTRIAL CONTROL COMMUNICATIONS, INC.

230 Horizon Drive, Suite 100
Verona, WI USA 53593
Tel: [608] 831-1255

http://www.iccdesigns.com

Printed in U.S.A