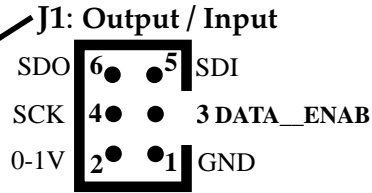




Application Note A59 - SPI Version 32

2015SPI-1, 4, 8,13

SPI connector



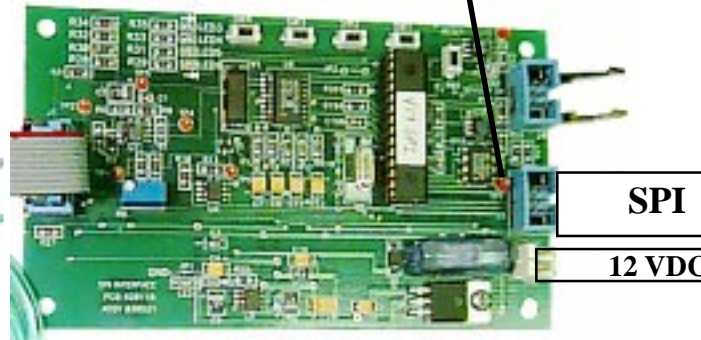
0-1 Volt out TP7

2005SPI board

2005SPI-1 Remote diffusion gas cell on 30 inch long cable



400 ppm (0.04%) CO₂ in gas cell (fresh air)

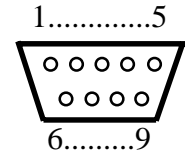


SPI
12 VDC

Standard 9 pin "D" RS232 connector

Pin#	Signal Name
1	DCD (data carrier detect)
2	RX (receive data)
3	TX (transmit data)
4	DTR (data terminal ready)
5	GND (signal ground)
6	DSR (data set ready)
7	RTS (request to send)
8	CTS (clear to send)
9	RI (ring indicator)
Shell	FG (frame ground)

9 pin "D" RS232



RS232 TEST board



null modem switch allows you to reverse pins 2 & 3

9 pin "D" female RS232 conn.
pin 2 - transmit
pin 3 - receive
pin 5 - digital ground

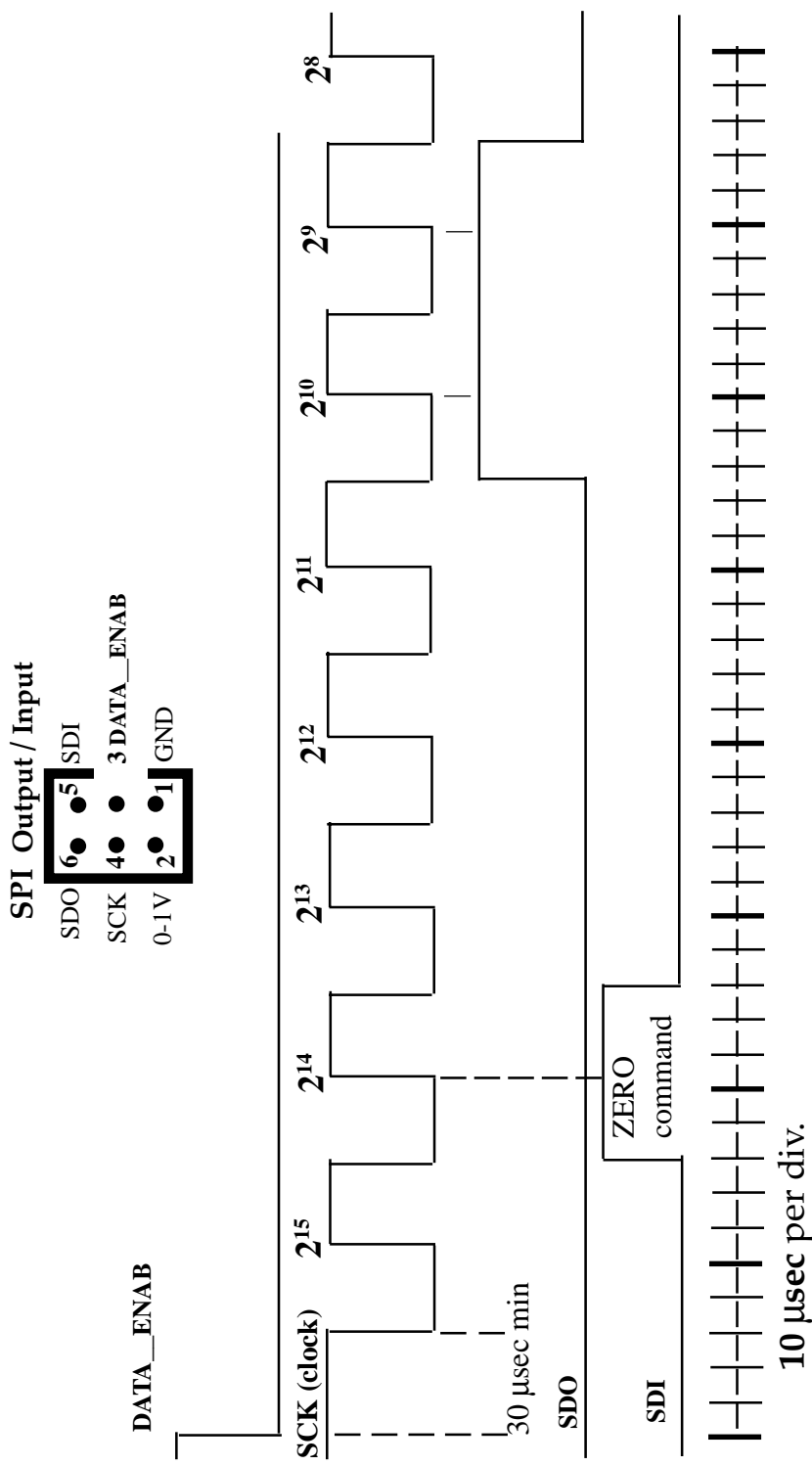
Note: If JP2 is installed on the SPI board then JP1 needs to be installed on the RS232 board

The Model **2015SPI-3** uses an on board gas cell for the higher full scale of 3 to 20% CO₂. The Model **2015SPI-1** has the longer on board gas cell for a full scale anywhere from 0.2 to 2.0% CO₂. The Model **2005SPI-2** uses a shorter remote gas cell for the higher full scale of 3 to 20% CO₂. The Model **2005SPI-1** has the longer remote gas cell shown for a full scale anywhere from 0.2 to 2.0% CO₂. The **2015SPI-4** is a 4 to 100% CH₄ unit.

Getting Started: Refer to the individual specification sheet that is shipped with each unit. A print out of the stored **calibration parameters** is shipped with each unit. This can be duplicated using the **RS232 test board** & a PC with **HyperTerminal** program that comes with Windows. **DO NOT** press the calibration switches on the SPI board until you verify proper operation using the **RS232** board. Connect 12 VDC as shown in the specification sheet to both boards. Connect the ribbon cable between the **SPI** connector J1 and the **RS232** board as shown. Connect the **RS232** 9-pin "D" connector to one of your **COM ports** on your PC. Set up **HyperTerminal** to talk to that port at 9600 baud, no parity, 8 bits and 1 stop bit, Xon/Xoff under Port Settings in **HyperTerminal** Properties. The **RESET** switches on each board perform a power on RESET. They do not clear the calibration data. The Print out shipped with each sensor shows the response to keyboard commands **1, 1 1, 2, 3,** and **4** (see page 4). Keyboard command **2** will show you the detectors peak to peak response to nitrogen when it was calibrated like 3.9541 volts and the temperature it was calibrated at like 27.3°C. Command **1 1** will give you the present ambient air detectors Vpp response. Pressing the **SPAN** calibration switch without the **STAR value of span gas** in the cell will result in an incorrect calibration. A **SETS SPAN** should **NOT** be done without first doing a **SETZ ZERO** with nitrogen in the gas cell. The on-board switches perform **one point** field calibrations. See page 2 for **SLAVE Mode** operation.

2005SPI and 2015SPI timing diagram for Slave Mode Operation

Version 27 firmware & later



The high to low transition of the clock (**SCK**) must occur at least 30 microseconds after the high to low transition of the **DATA_ENAB** control for that sensor. In **SLAVE Mode** the clock is controlled by you, the Master microprocessor. The **SDO** data output changes on the high to low transition of the clock (**SCK**) and the data should be read on the low to high transition of the clock (**SCK**). The example in the timing diagram above shows a % gas reading of **15.36%** which is equal to a binary output of **0000 0110 0000 0000** (only bits **2¹⁰** and **2⁹** are high "1"). The last eight bits (least significant byte LSB) are not shown on the diagram but they are all low "0". You must wait (stop the clock) a minimum of 100 microseconds between the end of the **MSB** (Most Significant 8 bits) and the beginning of the **LSB** (Least Significant 8 bits).

The Serial Data Input (**SDI**) may initiate a **SETZ command** by sending **C000 hex** (binary **1100 0000 0000 0000**). You need to make sure that there is zero gas in the gas cell and that the gas concentration is stable when you do this. This is the first part of a two point calibration and should be followed by a **SETS** command using a known concentration of gas in the cell. This is the first command of a two point calibration and should be followed by a **SETS** command.

The Serial Data Input (**SDI**) may initiate a **SETS command** by sending **AXXX hex** (binary **1010 XXXX XXXX XXXX**) where the least significant **12 bits** is the **STAR gas value**. You need to make sure that the correct value **SPAN** gas in the gas cell and that the gas concentration is stable when you do this. An example of a **SPAN** command at **5.0% gas** would be an **SDI** of binary **1010 0000 0011 0010** with the decimal point implied. That would make the **STAR** value = 5.00

The Serial Data Input (**SDI**) may initiate a **RESET command** by sending **1000 hex** (binary **0001 0000 0000 0000**). This command resets the SPI board and should be initiated whenever the master microprocessor resets.

Sample "C" Source Code for Digital SPI Interface

```
YourFunctionName() // Slave mode control to emulate customer equipment
{
    #define NORMAL_OP_FIELD 0 // MUST agree with Gas board!
    #define ZERO_OFF_FIELD 64 // 64d Ver 25 (SPI Board) change
    #define SPAN_OFF_FIELD 128 // 128d

    unsigned char byte2send[2];
    int gasread;
    unsigned int span_target;

    port.bim &= 0xfd; // dummy port where changes are made
    PORTB = port.bim; // RB1=0 for Data_Enable (OUTPUT)

    MilliSeconds(1); // Delay added to ensure MSB is first

    // typing 'ZERO' = 'field zero' in Master mode
    // cal.command is a Global Variable we use to control the commands to be sent
    if ( cal.command == FIELD_ZERO )
    {
        byte2send[0] = ZERO_OFF_FIELD; //command = ZERO (0100 0000)
        byte2send[1] = 0;
    }

    else if (cal.command == FIELD_SPAN)
    {
        span_target = (unsigned int)(cal.span_target * 10); // Double value to int value
        byte2send[1] = (unsigned char)( span_target & 0x00FF ); // load last 8 bits
        span_target = (span_target >> 8);
        byte2send[0] = (unsigned char)( span_target & 0x00FF ); // load 1st 4 bits
        byte2send[0] = (byte2send[0] | SPAN_OFF_FIELD); // Load Command, = 128 (1000 0000b)
    }

    else // normal op, get reading
    {
        byte2send[0] = NORMAL_OP_FIELD;
        byte2send[1] = 0;
    }

    // ALWAYS receive gas reading
    gasread = SendByteSPI(byte2send[0]); // msb loaded into SPI buffer and sent
    gasread = (gasread << 8);

    //make sure to leave at least 100usec between sending of bytes :
    gasread |= SendByteSPI(byte2send[1]); // lsb loaded into SPI buffer and sent

    calc_gas_percent = gasread; //latest data from gas board

    port.bim |= 0x02; // dummy port where changes are made
    PORTB = port.bim; // RB1=1 for Data_Enable (OUTPUT)

    if ( cal.command != FALSE )
    {
        cal.command = FALSE;
        SendRS232 ( DONE );
    }

    // end of Slave Command function
}
```

RS232 Test Board Terminal Emulation Interfaces

For test and troubleshooting you may connect an RS232 test board to the SPI connector as shown on page 1 and the RS232 9pin D connector to your computer's COM port. A terminal emulation program (VT100 or VT220 type dumb terminal) **HyperTerminal** program that comes with Windows or a program like **PCPLUS** in DOS may be used to talk to the RS232 Test Board. If you have properly configured your terminal program and connected the correct COM port to the RS232 connector you should get a response on your screen of "LOCKED" when you press the RESET switch on the RS232 Test Board. Type **VTI** and it will unlock. This will allow you to type the commands listed below: See **Application Note A66**.

<u>Command</u>	<u>Action</u>	<u>Command</u>	<u>Action</u>
1	DD.DD% gas Continuous	VTI	Unlocks for changing set-up, time out 5 min
1	Toggles all calculated values: --- Measurements ---	CX DDD.DDDDD	Enter Curve Coefficient X =0-7 & value D
(first "1" scrolls % gas data, the next "1" gives measurement table to right, next "1" goes back to scroll)	0.00 % gas 3.8818 volts 27.45 Deg C 0.0010 Atten	STC D.DDDDD ZTC D.DDDDD STAR DDD.DD	Enter Span Temperature Coefficient D Enter Zero Temperature Coefficient D Enter Span Target value D
	20.0 Range	VS D.DDDDD VZ D.DDDDD	Enter Span voltage D (measured by SPAN) Enter Zero voltage D (measured by ZERO)
	0.000 0-1 V out	SETZ	Perform auto ZERO & store VZ & TZ
	3.8855 VZ TC'd	SETS	Perform auto SPAN to STAR & store VS & TS
	1.9708 VSTC'd	OFF D.DDDDD	Enter new AgingFactor, effects Field Calib.
	ZERO switch	FILTER D	Enter new filter value D 1-10 lamp cycles ave
	SPAN switch	D CCCCCCCC	Enter Cal Date (9 alphanumeric characters)
2	Displays all calibration data ----- Cal -----	RT	RESETS the RS232 Test Board
	VZ 3.8801 Volts	RS	RESETS the SPI board
	VS 1.9683 Volts (full scale)	Note:	All values are stored on EEPROM so they are not lost during power failures.
	TZ 28.43 Deg C (zero cal temp)		
	TS 28.43 Deg C (span cal temp)		
	ZTC -0.005486 V/degC		
	STC 0.002904 V/degCs		
	STAR 20.0 % gas (target calibration gas)		
	OFF 1.0000 Volts (aging factor as a result of field1 point calibration)		
3	Displays all response curve coefficients: ----- Curve -----	<u>Command</u>	<u>Action</u>
	CO -0.030000	5	Change % Gas reading & STAR values to 4 decimal places in SLAVE mode only. Press RESET or type RT to reset back to two decimal places.
	C1 8.438299		
	C2 33.528999		
	C3 66.509003		
	C4 16.270000		
	C5 -24.007000		
	C6 0.000000 (if coefficient is "0" the term will drop out. Hence this is an example of a 5th)		
	C7 0.000000		
4	Display miscellaneous data ----- Sys Info -----		
	UNV.16 RS232 Rev (read from e-prom)		
	20.30 SPI Rev (read from e-prom)		
	DATE 9904243 (factory calibration date & serial number: April 1999 # 243)		
	Filter 15 Secs (variable 1 to 10 seconds, factory set at 15 seconds)		