AN1584

"Very Low-Pressure" Smart Sensing Solution with Serial Communications Interface

Jeffery Baum and Eric Jacobsen
Systems Engineering Group
Sensor Products Division
Motorola Semiconductor Products Sector
Phoenix, Arizona

INTRODUCTION

This paper is an update on the recent progress that has been made in using local intelligence to improve both functionality and performance for low–pressure smart sensing applications. As the name implies, the following text, figures, and tables provide a follow–up overview of the system enhancements that build upon the work documented in a previous paper entitled "Low–Pressure Smart Sensing Solution with Serial Communications Interface" (presented at Sensors Expo Boston '95).

As alluded to in the previous paper, the lowest pressure devices in the Motorola portfolio are rated at a full–scale pressure of 10 kPa (40″ of H^2O). The calibrated and temperature compensated, 10 kPa device (MPX2010) is specified to operate at a 10 Vdc supply voltage and produce 25 mV (nominal) at the full–scale pressure of 10 kPa. This translates to a 0.25 mV/(V*kPa) pressure sensitivity. At the specified operating supply voltage, this sensitivity is not conducive to measuring pressures below several kPa with the previously established performance goal of 1–2% of full–scale pressure accuracy. In addition, increasing the dc supply excitation to the device's absolute maximum only produces limited improvement in the output signal level.

The original low–pressure smart sensing solution had been developed to demonstrate a system solution capable of measuring 0–10″ H^2O (0–2.5 kPa) with the above–mentioned accuracy. This solution was based around the MPX2010 pressure sensor. This system is reviewed in the next section. While this system provides an accurate solution for a range spanning several kPa, it cannot maintain this performance for sub–kPa pressure ranges. Considering the market opportunities for low–cost, high–performance sensing solutions in the range of less than 1 kPa (4″ H^2O and below) and the sensor/system design modifications required to address this very low pressure range, it was decided to develop a solution for full–scale pressure ranges as low as 1.5″ H^2O with 1–2% overall accuracy. The majority of applications demanding such performance are typically related to either liquid–level or gas–flow sensing. The key design changes for this next generation system are:

- use of a more sensitive piezoresistive transducer element.
- applying the full excitation voltage directly across the piezoresistive transducer element.
- providing an on–chip temperature sensing circuit.
- performing span temperature compensation in software.

These enhancements to the original system solution will be discussed following a brief review of the MPX2010–based low–pressure solution. For the purposes of clarity, the original low–pressure solution employing the MPX2010 will be referred to as the "MPX2010–based solution/system", while the new system will be referred to as the "Very Low–Pressure solution/system".

MPX2010–BASED SYSTEM REVIEW

Using a "high–voltage", low duty–cycle, relatively low–frequency, pulsed excitation, much greater gains in sensor output sensitivity can be obtained (compared to limited improvement with elevated dc excitation). For the MPX2010 pulsed at 24 V, we obtain 15 mV of output for an applied pressure of 10″ H^2O (2.5 kPa). This same sensor device will only produce 6.25 mV at its normally specified supply of 10 V and 2.5 kPa; thus, not meeting the signal–to–noise ratio criteria for a 1–2% accuracy performance (Table 1 shows the operating characteristics of the MPX2010 at its specified 10 V dc excitation and in terms of its rated full–scale pressure of

10 kPa). While pulsed excitation is a fundamental advantage of this system solution, there are many other intelligent features that contribute to the milestone low–cost performance obtained. The prior paper described a smart sensing solution intended to sense full–scale pressures below 10 inches of water with 1% of full scale pressure resolution and better than 2% of full–scale accuracy. This solution contained the following hardware sub–systems (see Figure 1, MPX2010–based Smart Sensing Block Diagram):

- MPX2010 pressure sensor.
- high–side switch pulsing circuitry.
- signal conditioning amplifier interface with resistors to adjust the sensor's amplified, full–scale span and zero pressure offset.
- on–chip resources of a complete 8–bit microcontroller (MCU).
- MCU oscillator circuitry (4 MHz).
- 5 V ±5% linear voltage regulator.
- low–voltage inhibit (LVI) supervisory voltage monitoring circuit.
- resistor divider connected to the sensor's power supply bias to sense the excitation voltage across the sensor.

The above sub–systems, as employed by the MPX2010 solution, are explained in detail in the previous paper. The relevant sub–systems, with necessary modifications, for the Very Low–Pressure system are included in the System Design section of this paper. (Figure 2, MPX2010–based System Schematic for MPX2010–based solution is repeated here for reference).

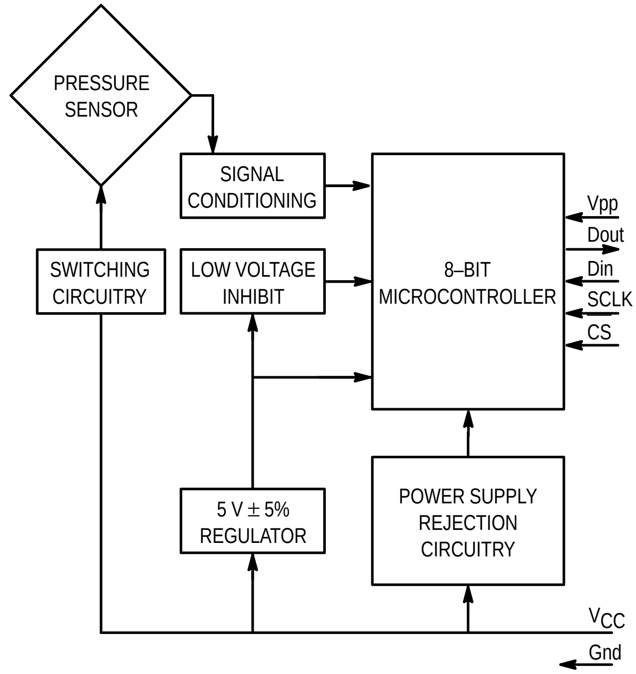

Figure 1. MPX2010–based Smart Sensing Block Diagram

Characteristic	Min	Typ	Max	Unit
Pressure Range	0	—	10	kPa
Supply Voltage	—	10	16	Vdc
Supply Current	—	6	—	mAdc
Full–Scale Span (FSS)	24	25	26	mV
Zero–Pressure Offset	–1.0	—	1.0	mV
Sensitivity	—	2.5	—	mV/kPa
Linearity	–1.0	±0.15	1.0	%FSS
Pressure Hysteresis (0 – 10 kPa)	–0.1	—	0.1	%FSS
Temperature Hysteresis (–40°C to +125°C)	—	±0.5	—	%FSS
Temperature Effect on Full–Scale Span	–1.0	—	1.0	%FSS
Temperature Effect on Offset (0°C to 85°C)	–1.0	—	1.0	mV
Input Impedance	1300	—	2500	Ω
Output Impedance	1400	—	3000	Ω
Response Time (10% to 90%)	—	1.0	—	ms
Temperature Error Band	0	—	85	°C
Stability	—	±0.5	—	%FSS

Table 1. MPX2010 Operating Characteristics
(Supply Voltage = 10 Vdc, T_A = 25°C unless otherwise noted)

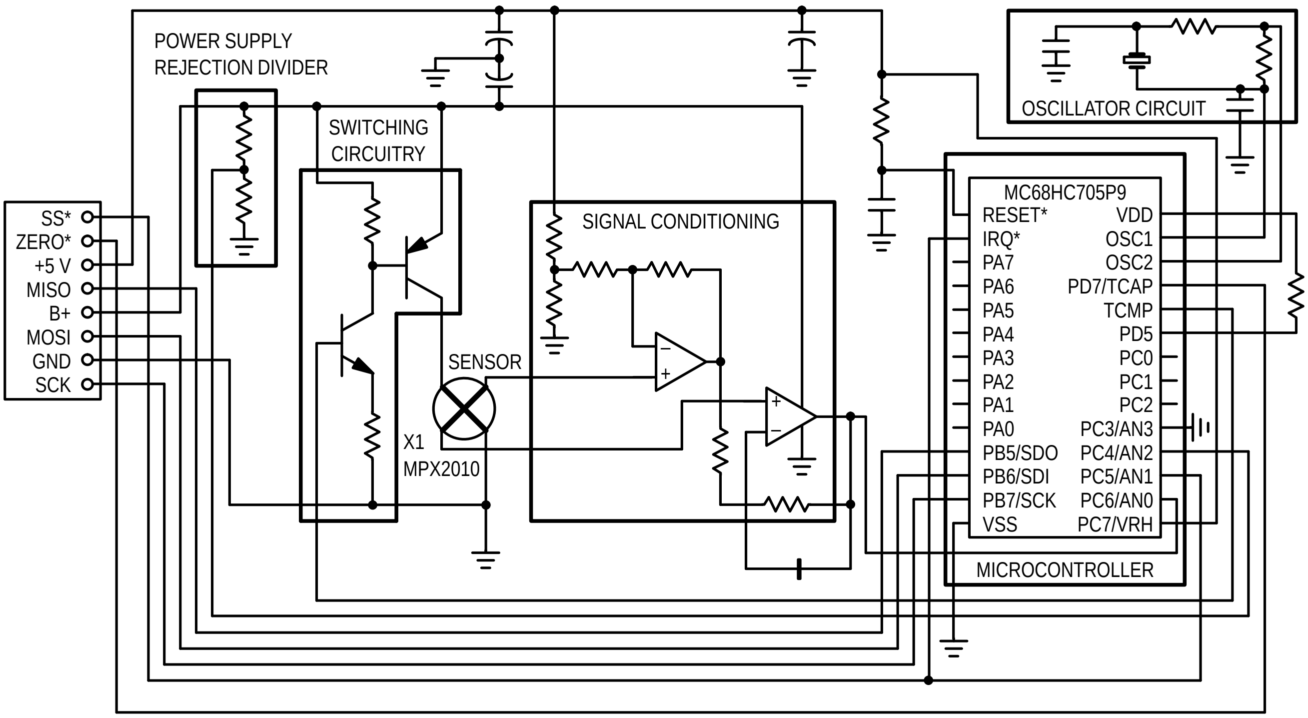

Figure 2. MPX2010–based System Schematic

VERY LOW–PRESSURE SOLUTION DEVELOPMENT HISTORY

In considering the circuit design of the MPX2010 sensor, it becomes readily apparent that great gains in sensitivity can be achieved by making some relatively simple changes. These changes and their system implications (both hardware and software) are the basis for the Very Low–Pressure system. The first aspect of the sensor design targeted for producing enhanced sensitivity is the removal of the resistances that are in series with the piezoresistive transducer element and the supply voltage and ground connections. (See Figure 3 for MPX2010 internal schematic.) These resistances are used for temperature compensating the drift in the dynamic signal span/sensitivity over temperature (TC of span). The circular transducer symbol (circle with an “X” in the center) is used here to denote the piezoresistive element.

The arrows on the resistor symbols indicate that these are laser–trimmed resistances. At the final trimmed values that establish the proper temperature coefficient for compensating the inherent TC of span of the piezoresistive element, the total series resistance (sum of both resistors) is approximately twice the resistance of the piezoresistive element. The remaining resistances in the circuit are parallel to the piezoresistive element and high enough in value, compared to the piezoresistive element, to not significantly change the resistor divider ratio formed by the so–called TC of span series resistors and the piezoresistive element. Thus, approximately one–third of the MPX2010 supply voltage is actually provided to the piezoresistive element for excitation. In other words, if one could find a way to eliminate the series resistance, while preserving the span temperature compensation via other non–sensitivity reducing means, then an almost 70% sensitivity increase results.

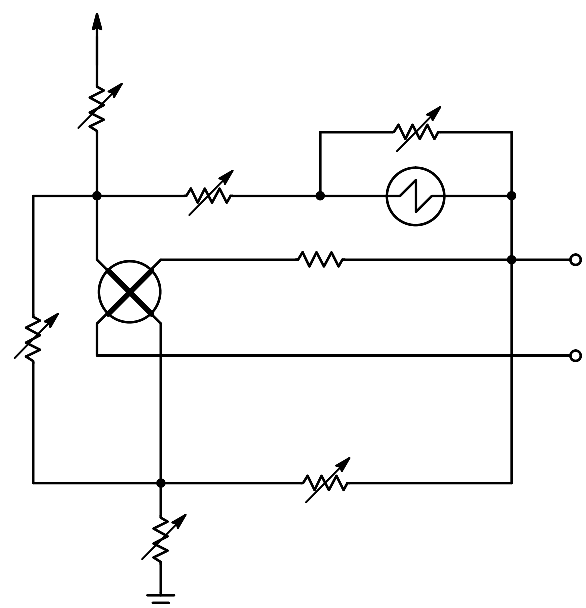

Figure 3. MPX2010 Internal Circuit Schematic

While removing a couple of components is a simple re–design effort proposal, redesigning the temperature compensation of offset (TC of offset) circuitry and the design of the alternative means of span compensating is not. At first glance, it seemed straight forward to eliminate the sensitivity–reducing resistors and replace these with a hardware/software solution for span compensation. It also appeared that the conventional laser–trimmed offset temperature compensation scheme could be directly re–used (with modified trim procedure). As Murphy would have it, both of the above presumptions were false. When considering the tight tolerance needed for TC of offset in such low–pressure applications and the manufacturing changes required for the alternative TC of span scheme, there is some challenge to making such modifications.

Although it is not a simple "drop–in–the–bucket" solution, it is possible to provide a different TC of offset compensation circuit and an on–chip temperature sensing circuit (to be used in conjunction with a software algorithm) for TC of span compensation. These changes, plus a higher sensitivity piezoresistive element, are indeed the design changes that were implemented for this Very Low–Pressure system solution. The solution sub–systems and modifications/additions to the prior MPX2010–based system solution are discussed in detail below.

SYSTEM DESIGN

The following sub–systems of the Very Low–Pressure Smart Sensing Solution are either directly reused as in the MPX2010–based Low–Pressure Smart Sensing Solution, modified versions of an MPX2010 solution sub–system, or a newly added sub–system for this development effort. Since the major difference between systems is the sensing device, only the new analog sensing section is shown in the schematic of Figure 5. The digital/MCU portion of the system is basically the same as that of Figure 2, with the addition of another A/D converter channel to read the temperature sense circuit output.

Sensor

The sensor device design changes (compared to an MPX2010 sensor) are: the elimination of the series resistors that are laser–trimmed to tune in a compensating temperature coefficient for span temperature drift, replacement of the conventional piezoresistive sensing element with a higher sensitivity geometry element, the addition of a linear output temperature sensing circuit consisting of a string of diode–connected bipolar transistors and a current source, and a redesigned offset calibration and temperature compensation network. For comparison purposes, a schematic representation of this new sensor is shown below in Figure 4. The circular transducer symbol with the concentric inner circle is used here to denote the higher sensitivity geometry piezoresistive element.

Pulsing Circuitry

As previously mentioned, the sensor's output is ratiometric to the excitation voltage across the sensing element, the sensor's sensitivity increases with increasing supply voltage. Thus, to detect low pressures and minute changes in pressure, it is desirable to operate the sensor at the highest possible excitation voltage. The maximum supply voltage at which the sensor can reliably operate is determined by one or both of the following two limitations: a) maximum allowable sensor die temperature, b) maximum supply voltage available in the sensing application/system.

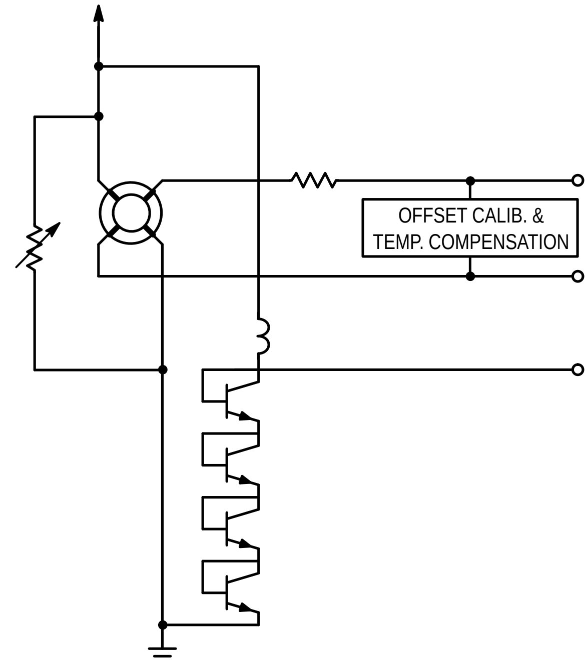

Figure 4. Higher Sensitivity Sensor

In terms of the thermal/power dissipation issue, the maximum voltage that can be supplied to the sensor on a continuous basis is relatively low compared to that which can be pulsed on the sensor at a low duty cycle. The average power that is dissipated in the sensor is the square of the average sensor excitation voltage divided by the input resistance of the sensor. When the sensor's supply bias is operated in a pulsed fashion, the average excitation voltage is simply the product of the dc supply voltage used and the percent duty–cycle that the dc voltage is "on".

The pulsing circuitry is a high–side switch (two small–signal switching transistors with associated bias resistors) that is controlled via the output compare (TCMP) pin of the MCU. The output compare timer function of the MCU provides a logic–level pulse waveform to the switch that has a 2 ms period and a 200 µs on–time (note: this is user–programmable).

Signal Conditioning

Even with pulsing at a relatively high supply voltage, the pressure sensing element still has a full–scale output that is only on the order of tens of millivolts. To input this signal to the A/D converter of the MCU, the sensing element output must be amplified to allow adequate digital resolution. A basic two operational amplifier signal conditioning circuit is used to provide the following desired characteristics of an instrumentation amplifier interface:

- high input impedance
- low output impedance
- differential to single–ended conversion of the pressure sensor signal
- moderate gain capability

Both the nominal gain and offset reference pedestal of this interface circuit can be adjusted to fit a given distribution of sensor devices. Varying the gain and offset reference pedestal is desirable since pressure sensors' full–scale span and zero pressure offset voltages will vary somewhat from lot–to–lot and unit–to–unit. During software calibration, each sensor device's specific offset and full–scale output characteristics will be stored. Nonetheless, a variable gain amplifier circuit is desirable to coarsely tune the sensor's full–scale span, and a positive or negative dc level shift (offset pedestal adjustment) of the pressure sensor signal is needed to translate the pressure sensor's signal conditioned output span to a specific level (e.g. within the high and low reference voltages of the A/D converter).

Microcontroller

The microcontroller performs all the necessary tasks to give the smart sensor system the specified performance and intelligent features. The following describes its responsibilities:

- Creates the control signal to pulse the sensor.
- Samples the pressure sensor's output.
- Signal averages a programmable number of amplified pressure sensor samples for noise reduction.
- Samples the output of the temperature sensing circuit. Monitoring the relative die temperature allows the microcontroller to compensate for the variations in the pressure sensor's span with temperature.
- Samples a scaled–down version of the pressure sensor supply voltage. Monitoring the power supply voltage allows the microcontroller to reject sensor output changes resulting from power supply variations.
- Uses serial communications interface (SPI) to receive commands from and to send sensor information to a master MCU.

Resistor Divider for Rejection of Supply Voltage Variation

Since the pressure sensor's output voltage is ratiometric to its supply voltage, any variation in supply voltage will result in variation of the pressure sensor's output voltage. By attenuating the supply voltage (since the supply voltage may exceed the 5 V range of the A/D) with a resistor divider, this scaled voltage can be sampled by the microcontroller's A/D converter. By sampling the scaled supply voltage, the microcontroller can compensate for any variances in the pressure sensor's output voltage that are due to supply variations. This technique allows correct pressure determination even when the pressure sensor is powered with an unregulated supply.

5 V Regulator

A 5 V ±5% voltage regulator is required for the following functions:

- To provide a stable 5 V for the high voltage reference (VRH) of the microcontroller's A/D converter. A stable voltage reference is crucial for sampling any analog voltage signals.
- To provide a stable 5 V for the resistor divider that is used to level shift the amplified zero pressure offset voltage.

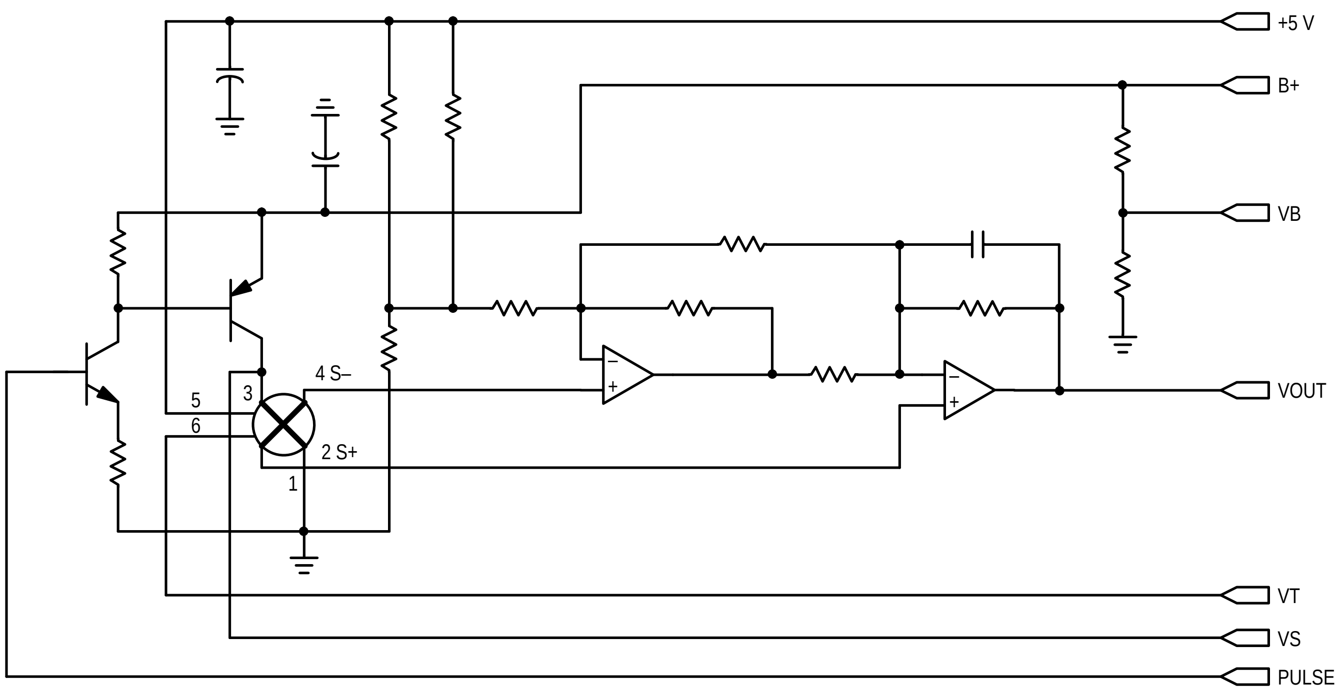

Figure 5. Very Low–Pressure System Analog Sensing Schematic

Low Voltage Inhibit (LVI) Circuitry

Low voltage inhibit circuitry is required to ensure proper power–on–reset (POR) of the microcontroller and to put the MCU in a known state when the supply voltage is decreased below the MCU supply voltage threshold.

SOFTWARE DESCRIPTION

The smart sensor system's EPROM resident code provides the control pulse for the sensor's excitation voltage and performs calibration with respect to a wide range of excitation voltages (e.g., 20 ~ 28 V for HVAC applications). Pressure measurement sampling and averaging is also incorporated to reduce both signal error and noise. In response to the temperature sense input, the MCU performs temperature compensation of the sensor's sensitivity/span. In addition, the availability of a serial communications interface allows a variety of software commands to be sent to the smart sensor system. The following brief outline provides more detailed description about the software features included in the smart sensor system.

Calibration, Temperature Compensation, and Power Supply Rejection

Only twelve 8–bit words of information are stored in order to calibrate the smart sensor system for a given pressure sensor device, to store the relationship between the pressure output and power supply voltage, and to store the relationship between the temperature sensor output and temperature. This information is used to reduce errors due to device–to–device variations, to reject variations in power supply voltage that can introduce error into the pressure measurement, and to compensate for temperature drift errors of the pressure sensor's sensitivity. The pressure sensor's amplified output at zero–pressure and full–scale pressure are stored at each of two different supply voltages and two different temperatures. In addition, the scaled and digitized representation of the applied supply voltages and the output of the temperature sensing circuit at the two applied temperatures are stored.

Compensating for power supply variation in software allows higher performance with lower tolerance, or even unregulated, supply voltages. For HVAC applications, where a 24 Vac line voltage will be simply rectified and filtered to provide a crude 24 Vdc supply, this approach has major performance benefits. The impact on applications where a regulated supply is available is that a lower cost regulator or dc–to–dc converter can be used without compromising system accuracy significantly. The benefit of temperature compensating via the on–chip temperature sensing circuit and a software algorithm is that temperature errors are minimized without introducing the laser–trimmed series resistive components mentioned above. This prevents precious piezoresistive element excitation voltage from being dropped across such series resistances. A secondary benefit is that the number of laser–trimmed resistances is reduced by two. Thus, manufacturing throughput of sensor devices is increased. Also, in certain scenarios, software temperature compensation can actually yield better temperature performance compared to that of its hardware counterpart.

A/D Sample Averaging

Noise inherent to the 8 bit A/D successive approximation conversion method used by the smart sensor accounts for ±1 bit resolution. Signal noise, which exhibits a measured peak to peak range larger in magnitude than 1 bit of A/D resolution, can be minimized by a sample averaging technique.

The current technique uses 16 A/D converted pressure samples, sums the result, and divides by 16 (the number of samples) to get the average (as shown below).

$$\text{Avg} = \sum_{1}^{n} \frac{(an)}{n}; \text{ where } n = 16$$

Assuming a gaussian distribution of noise, this averaging technique improves the signal to noise ratio (SNR).

Smart Sensor Unit ID & Software Revision Level

This solution may be implemented as a single sensing system using a non–dedicated MCU to provide the sensing function and smart features or as a slaved smart sensor (with dedicated sensing MCU) that communicates over a serial bus to a master controller or microprocessor (Host). Part identification and software revision level can also be read on request from the master MCU. This information is utilized by the master MCU to determine what the full–scale pressure range of a given smart sensor unit is. This allows for multiple sensor units with different pressure ranges to be controlled and sensed from a single master MCU.

Communication

The serial peripheral interface (SPI) is used to communicate to a master/host MCU. The master MCU initiates all I/O control, and sends commands to the slave regarding data requests, calibration, etc. The command codes are parsed at the slave in a look–up table, at which time the corresponding request is serviced via subroutine. Table 2 lists the Master/Slave commands.

Request Pressure

Returns the percent of full scale pressure applied to the sensor in the form of $00 (0) through $FF (255) and is equivalent to:

Pressure Range {from 0 to 255},

$$\text{where } \frac{\{0 - 255\}}{255} \times \text{FS} = \text{Measured Pressure.}$$

Request Temperature

Returns the percent of full scale temperature of the sensor in the form of $00 (0) through $FF (255) and is equivalent to:

Temperature Range {from 0 to 255},

$$\text{where } \frac{\{0 - 255\}}{255} \times \text{FS} = \text{Measured Temperature.}$$

Dynamic Zero

Assigns a nonzero pressure as offset reference.

Undo Dynamic Zero

Resets offset to the original stored offset.

Pressure Range

Returns a value representing the sensor's full–scale pressure range.

Function (Command Codes)	Command From Host	Data From Smart Sensor
Request Pressure	$01	$00~$FF
Dynamic Zero	$02	—
Undo Dynamic Zero	$03	—
Pressure Range	$04	TBD
Request Temperature	$05	$00~$FF

Table 2. Software Command Codes

SOFTWARE EXAMPLES

The following example listings show how a user may communicate with the smart sensor via a master MCU. The software example shown below assumes that the master MCU is an MC68HC11. Any MCU with the proper I/O functionality will operate similarly with the smart sensor system.

When using parallel I/O instead of an SPI port to interface the smart sensor, the user must “bit bang” the clock and data out of the parallel I/O, so as to simulate the SPI port. As long as the timing relationships of data and clock follow those of Figure 6 (see Table 3 as well), the smart sensor will function properly when interfaced to a processor with a parallel type interface. In the two code examples to follow, the sensor unit is interfaced to the master MCU via the SPI port, and the sensor’s input is connected to the HC11’s Port D pin 5.

This example is coded in ’C’ for the MC68HC11:

```
/* FIRST INITIALIZE THE I/O (INCLUDE A HEADER FILE TO INCLUDE I/O DEFINITIONS) */
void init_io(void)
{
PORTD = 0X29; /* SS* PD5 = 1, PD3 = 1, PD0 = 1 */
DDRD = 0X3B; /* SS* PD5 = 1, PD3 = 1, PD1 = 1, PD0 = 1 */
SPCR = 0X5E; /* ENABLE THE SPI, MAKE MCU THE MASTR, SCK = E CLK /4 */
/* I/O INITIALIZATION IS COMPLETE */
}

/* WE NEED A FUNCTION TO WRITE TO AND READ FROM THE SPI */
write_spi(char data)
{
SPDR = data;  /* WRITE THE DATA TO THE SPI DATA PORT */
   while( ! (SPSR & 0x80 )); /* WAIT UNTIL DATA HAS SHIFTED OUT OF AND
                                                            BACK INTO INTO THE SPI */
return(SPDR): /* RETRIEVE THE RESULTS OF THE LAST COMMAND TO
                                  THE SENSOR AND RETURN */
}

/* NOW WE NEED TO CALL THE ABOVE */
void main(void)
{
char rtn_data; /* rtn_data IS THE RETURNED DATA FROM THE SENSOR */

init_io();

while(1)  /* JUST LOOP FOREVER */

rtn_data = write_spi(0x01); /* 0x01 IS THE COMMAND TO THE SENSOR
                                                                              THAT REQUESTS PRESSURE. THE VALUE IN
```

```
                                rtn_data WILL BE IN THE RANGE OF
                                0..0XFF = 0..100% FULL SCALE PRESSURE THE
                                SECOND TIME THROUGH THE LOOP. THE INITIAL
                                TIME THROUGH THE LOOP, THE DATA
                                RETURNED IS INDETERMINATE  */
}
```

The next example is coded in assembly for the MC68HC11:

```
"* PORT OFFSETS INTO THE I/O MAP
PORTS    EQU     $1000          ASSUME THE I/O STARTS AT $1000
PORTD    EQU     $8
DDRD     EQU     $9
SPCR     EQU     $8
SPSR     EQU     $29
SPDR     EQU     $2A

         ORG     $E000
* FIRST INITIALIZE THE I/O
INITIO   LDX     #PORTS         BASE ADDRESS OF THE I/O
         LDAA    #$29
         STAA    PORTD,X        SS* PD5 = 1, PD3 = 1, PD0 = 1
         LDAA    #$3B
         STAA    DDRD,X         SS* PD5 = 1, PD3 = 1, PD1 = 1, PD0 = 1
         LDAA    #$5E
         STAA    SPCR,X         ENABLE THE SPI, MAKE MCU THE MASTR,
*                               SCK = E CLK /4
         RTS                    I/O INITIALIZATION IS COMPLETE

*WE NEED A SUBROUTINE TO WRITE TO AND READ FROM THE SPI
*TO CALL THIS ROUTINE LOAD ACCUMULATOR A WITH THE COMMAND DATA
*AND JSR WRITSPI. WHEN THE ROUTINE RETURNS, ACCUMULATOR A
*CONTAINS THE DATA RETURNED FROM THE SENSOR

WRITSPI  LDX     #PORTS         BASE ADDRESS OF THE I/O
         STAA    SPDR,X         SEND THE COMMAND TO THE SENSOR
WRLOOP   BRCLR   7,SPSR,WRLOOP LOOP UNTIL THE DATA HAS SHIFTED
                                OUT OF AND BACK INTO THE SPI
         LDAA    SPDR,X         RETRIEVE THE RESULTS OF THE LAST
                                COMMAND
*                               TO THE SENSOR
         RTS

* NOW WE NEED TO CALL THE ABOVE */
START    JSR     INITIO         SET–UP THE I/O
LOOP     LDAA    #$1            1 IS THE COMMAND TO THE SENSOR THAT
*                               REQUESTS PRESSURE
         JSR     WRITSPI        SEND THE COMMAND TO THE SENSOR.
*        ...                    THE VALUE RETURNED IN ACCUMULATOR A
*                               WILL BE IN THE RANGE 0..0XFF = 0..100%
*                               FULL SCALE PRESSURE THE SECOND TIME
*                               THROUGH THE LOOP. THE INITIAL TIME
*                               THROUGH THE LOOP, THE DATA RETURNED
                                IS INDETERMINATE DATA FROM THE SENSOR
         BRA     LOOP
```

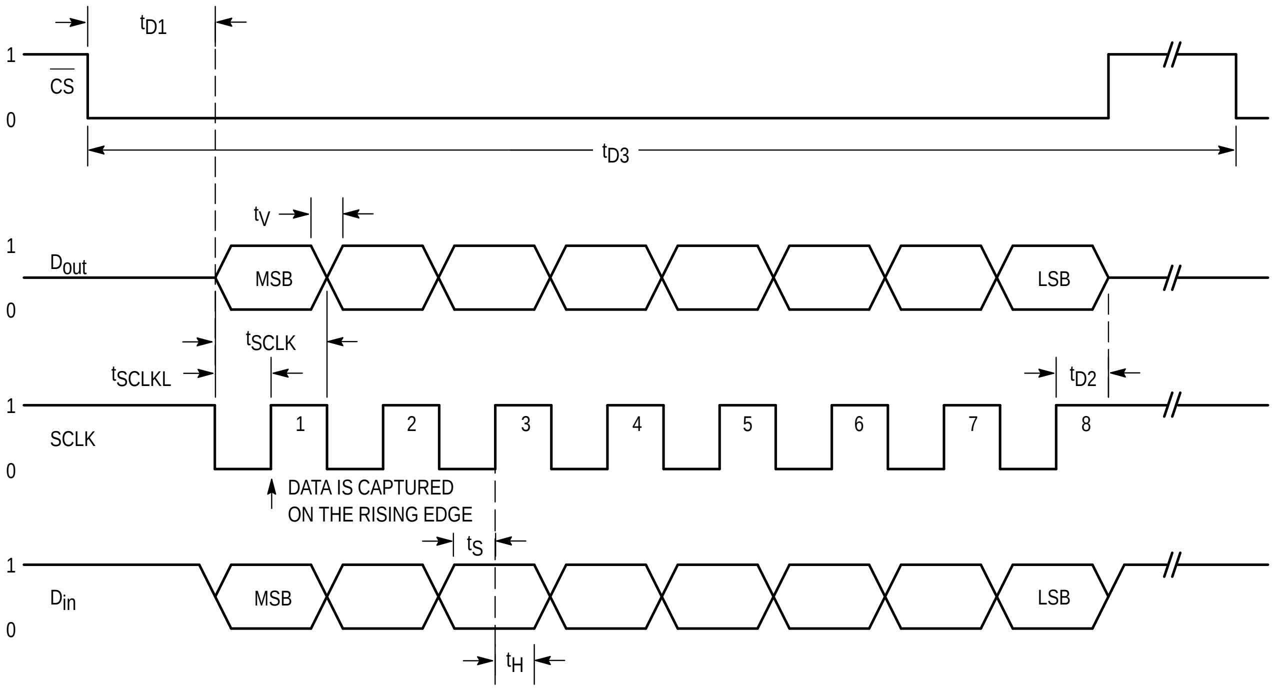

Figure 6. SPI Timing Diagram

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation	f_{OP}	dc	525	kHz
Cycle Time	t_{SCLK}	—	1920	ns
Clock (SCLK) Low Time	t_{SCLKL}	932	—	ns
D_{out} Data Valid Time	t_V	—	200	ns
D_{in} Setup Time	t_S	100	—	ns
D_{in} Hold time	t_H	100	—	ns
On–Bus Delay Time	t_{D1}	1	—	ms
Off–Bus Delay Time	t_{D2}	—	50	µs
Chip Select Period	t_{D3}	TBD	—	ms

Table 3. SPI Timing Characteristics

SERIAL DATA OUTPUT FORMAT

The primary serial data output is an 8–bit number of value 0–255. This number represents the current applied pressure as a percentage of the full–scale pressure rating of the smart sensor system. The master MCU can simply consider an output of “0” to be zero pressure and “255” to be full–scale pressure. To convert this number to engineering units, such as inches of water (″ H_2O), the master MCU must multiply the smart sensor output (0–255) by the full–scale pressure of the smart sensor system in ″ H_2O, and then divide (normalize) by 255. See equation in Software Description–Command Codes–Request Pressure.

The master MCU can either use an absolute number for the full–scale pressure of the smart sensor system (as indicated above) or can query each smart sensor that is connected to the serial bus for its rated pressure range. The latter technique allows multiple smart sensors of various full–scale pressure ranges to be communicating with a single master MCU, without the need for an absolute addressing scheme that contains full–scale pressure information for each sensor.

CONCLUSION

A smart sensing system which achieves high performance for very low–pressure applications has been presented here. The key performance advantage of the smart sensor system is taking advantage of the fact that the output of the piezoresistive sensing element is ratiometric (linearly proportional) to the excitation voltage applied to the sensing element. A pressure sensor device is pulsed at a much higher–than–normally–specified voltage and a low duty–cycle for the purpose of increased sensitivity. While some of the sensor’s parasitic drawbacks are increased in magnitude, some of the sensor’s negative characteristics are lessened and other sources of error and noise in the system are reduced. The net effect is that a better signal–to–noise ratio is obtained. To maximize the benefit of pulsed–excitation sensing technology, the span temperature compensation series resistances of the previous MPX2010 system design have been removed so that 100% of the excitation voltage is applied to the actual piezoresistive element. The span temperature compensation is accomplished via an on–chip temperature sensing circuit whose output is used by a software temperature compensation algorithm that resides in the system MCU. These schemes, combined with several other performance enhancing smart features, provide better pressure resolution and accuracy than inherent in conventional semiconductor pressure sensor devices alone.

Besides the sensor excitation pulsing, alternative span temperature compensation technique, and output sampling functions, a low–cost MCU provides the performance enhancing features of signal averaging, software calibration, and software power supply rejection. The added–functionality of intelligent communications capability, serial digital output flexibility, and local control and decision making capability are also at the user's disposal. The development history, system design, software functions, example communications routines, and serial output format have been detailed to provide the reader with an understanding of how low–pressure capability can be greatly enhanced via a smart sensor system approach.

ACKNOWLEDGEMENTS

We wish to acknowledge our colleagues Bill Lucas and Warren Schultz for their outstanding efforts and major contributions to the pursuit of very low–pressure sensing technology.

How to reach us:
USA / EUROPE / Locations Not Listed: Motorola Literature Distribution; P.O. Box 5405, Denver, Colorado 80217. 303–675–2140 or 1–800–441–2447

Mfax™: RMFAX0@email.sps.mot.com – TOUCHTONE 602–244–6609
– US & Canada ONLY 1–800–774–1848
INTERNET: http://motorola.com/sps

JAPAN: Nippon Motorola Ltd.; Tatsumi–SPD–JLDC, 6F Seibu–Butsuryu–Center, 3–14–2 Tatsumi Koto–Ku, Tokyo 135, Japan. 81–3–3521–8315

ASIA / PACIFIC: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park, 51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852–26629298

MOTOROLA