SILICON MICROSTRUCTURES INCORPORATED
Member of the ELMOS Group

SM5800 Amplified and Calibrated Pressure Sensor Family

Features

- Digital correction algorithm for nonlinearities of temperature, supply voltage and pressure
- Supply voltage range: 4.5 V to 5.5 V
- Embedded EEPROM for calibration coefficient storage
- Digital I²C interface
- Analog and digital output
- Operating temperature -40°C to 125°C

Application

- This Application Note applies to all amplified and calibrated SMI pressure sensor products from the SM5800 family.

Brief Functional Description

The sensor signal conditioning Application Specific Integrated Circuit (ASIC) used in the Silicon Microstructures, Inc. (SMI) amplified and calibrated products is designed for several fields of application. The ASIC has a built-in amplifier, an Analog-to-Digital Converter (ADC) to convert an amplified pressure sensor signal into a digital word, a digital signal processor to allow multi-order correction of the sensor output over both temperature and supply voltage, and a Digital-to-Analog Converter (DAC) and output amplifier to convert the digitally corrected signal back into an analog output. Digital communication by means of an Inter-Integrated Circuit (I²C) interface is possible as well. Coefficients for multi-order correction can be permanently stored inside the embedded Electrically Erasable Programmable Read-Only Memory (EEPROM). The I²C communication enables the usage of several identical pressure sensors on the same bus line.

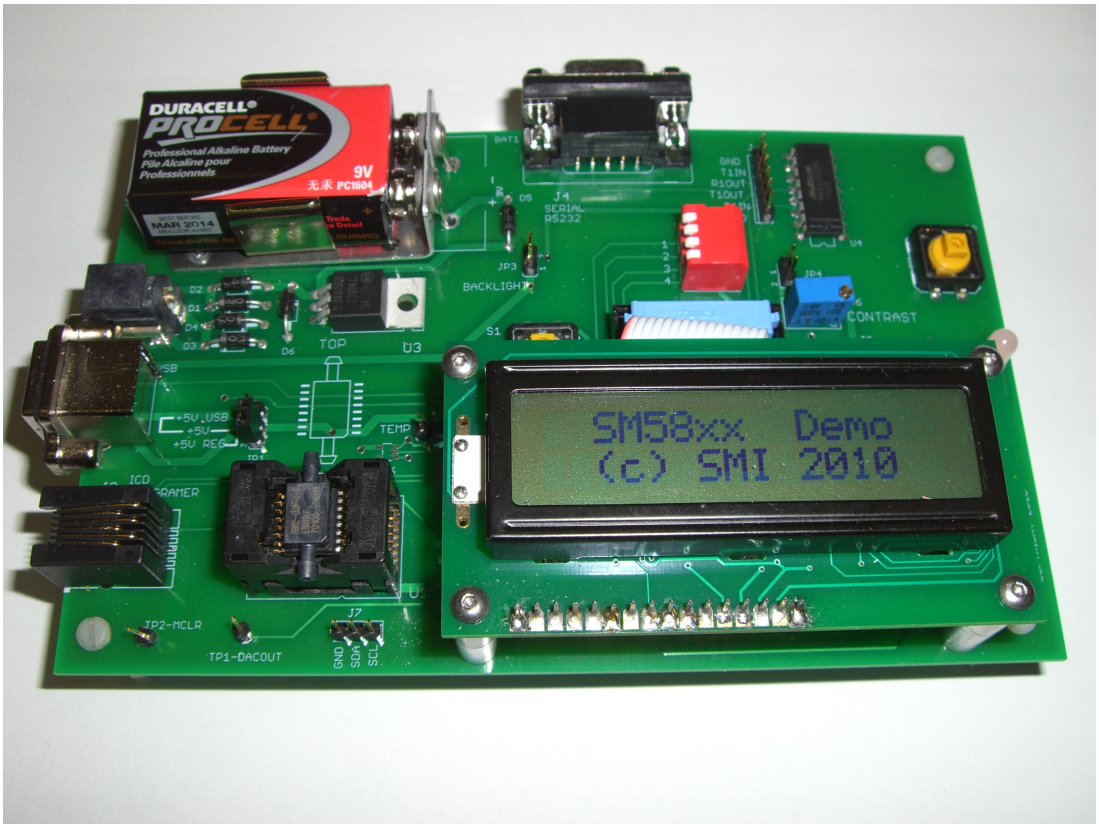

Fig.1: SM58xx Demonstrator Board

Phone: +1-(408) 577-0100 / **eMail:** sales@si-micro.com / **www:** http://www.si-micro.com

Table of Contents

Phone: +1-(408) 577-0100 / **eMail:** sales@si-micro.com / **www:** http://www.si-micro.com

1 Introduction

1.1 Functional Description

The sensor-signal conditioning ASIC is capable of operating at 5 V with ratiometric outputs. The ASIC is electronically programmable and provides an output correction for less than 1.5% total error from -40°C to +140°C. The digital part of the ASIC in this version communicates with an "outside" tester via separate pins. In normal use, these digital pins will not be accessible to the end customer.

The circuit calibrates and compensates the sensing element signal to minimize influence of temperature and improves signal-quality by reducing nonlinearities of pressure response and temperature coefficients.

Pressure sensing mode is the normal operating mode of the ASIC where the Vout pad provides an analog output and I²C communication can be used to read pressure values. In pressure sensing mode, the on-chip digital signal processor (DSP) performs the following typical calculation:

$$V_{out} = a_0 + a_1 \times V_T + a_2 \times V_T^2 + a_3 \times V_p + a_4 \times V_T^2 \times V_P + a_5 \times V_T^2 \times V_P + a_6 \times V_P^2 + a_7 \times V_T \times V_P^2 + a_8 \times V_T^2 \times V_P^2$$

where V_P is a pressure signal, V_T is a temperature signal, and a_0 to a_8 are constants. The coefficients a_0 to a_8 are determined during the calibration process and are stored in EEPROM.

The ASIC allows digital control of the output clipping, sensor response time (filter on or off) and storage of a unique part ID number.

The ASIC also allows random read and write access to all EEPROM cells including model coefficients, programmable gain and offset via the I²C interface. Write access is blocked permanently as soon as the write protection bit is set low (0).

Note: *SMI ships the EEPROM with the protection bit inactive high (1) to enable the end customer to make modifications before permanently locking EEPROM contents.*

Input Amplifier

The first stage after the sensor is a pre-amplifier block. Its purpose is to maximize the input sensitivity and minimize the total offset prior to the AD conversion. This has a coarse trim capability as outlined below. Final trim capability will take place in the digital correction block and final output amplifier.

The nominal sensor offset varies between +42 mV and -42 mV with the sensor driven at 5 V. Because the offset represents an imbalance in the bridge, the offset will be supply sensitive as well. Correction of this error term is performed in the input stage.

Analog-to-Digital Converter (ADC)

The ADC block has to convert the measured values to a digital word. To achieve the specified precision, the ADC has an accuracy of 11 bits.

The algorithm used by the DSP needs three different data values to correct the pressure value. These data values are fed into the DSP using multiplexer circuits. The three data values are

1. the pressure amplified by the input stage,
2. the temperature voltage generated by the current generator, and
3. the band-gap reference voltage.

Three bits are used to select the value that needs to be converted.

The ADC block receives a start-of-conversion signal from the amplifier in the input stage. This is also the case when it converts the temperature voltage or the band-gap reference voltage.

Digital Signal Processor (DSP)

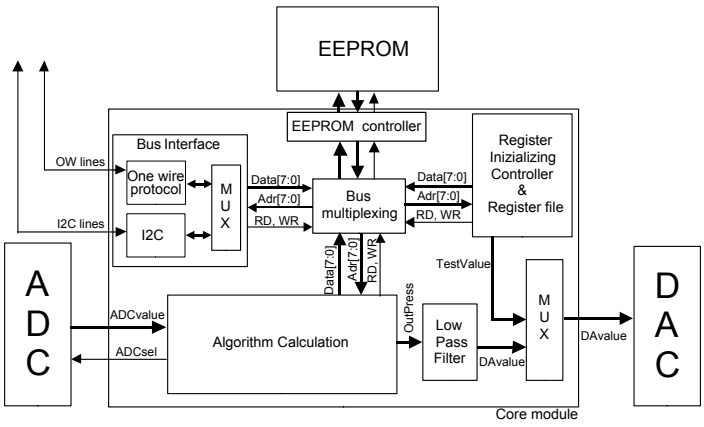

Fig.2: Blockdiagram of DSP

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

This digital signal processor performs the necessary calculations to do the signal conditioning. It is operating with a precision of up to 15 bits.

Output Stage

The output stage consisting of a DAC, an output buffer and an output clamp that has two functions. The first is to provide analog output drive from the ASIC, the second is in test mode to output internal voltages.

Digital-to-Analog Converter (DAC)

The DAC has a 12-bit resolution with an accuracy of 10-bit. The clock for the converter is generated by the digital part and has a frequency of 125 kHz.

Output Buffer

The output buffer is required to eliminate the digital signal processing noise to improve the equivalent resolution.

Note: *The analog output (DAC) must not be loaded with more than ±10 mA above 85°C.*

Output Clamp

The output clamp sets a limit range for the sensor in case of a failure (either high or low). The exact limit can be set by the customer and may vary by application. In general, the clamp needs to be programmed as 0 (no clamping), 2%, 5%, or 10% of supply with respect to both ground and supply. The DSP has a 12-bit register for both the upper and lower limit that allows setting the clamp point at any point between the rails as set by 2 digital words loaded into the on-board registers at the time of programming.

1.2 Block Diagram

The output of the amplifier is passed through a multiplexer (MUX) block to an 11-bit ADC. This ADC value is averaged four times to give a synthetic 13-bit word. Assuming that the circuit or sensor does contribute some noise, then the average will provide a higher resolution than is simply possible with the 11 bit ADC. This concept is often referred to as over sampling.

The DSP machine block is a digital signal processor which takes the digitized sensor information and processes it according to the look-up algorithms used in the DSP. The correction possible is defined by the amount of programmable EEPROM coefficients (maximum 20). This allows a number of corrections although the primary ones are for 3rd order of temperature and pressure correction. The DSP also contains storage registers for serializing the ASIC for traceability as well as programmed upper and lower limits for the ASIC output. In addition, output signal smoothing/filtering is programmable in this module.

The output of the DSP is fed to the 12-bit DAC which in turn drives an output buffer amplifier.

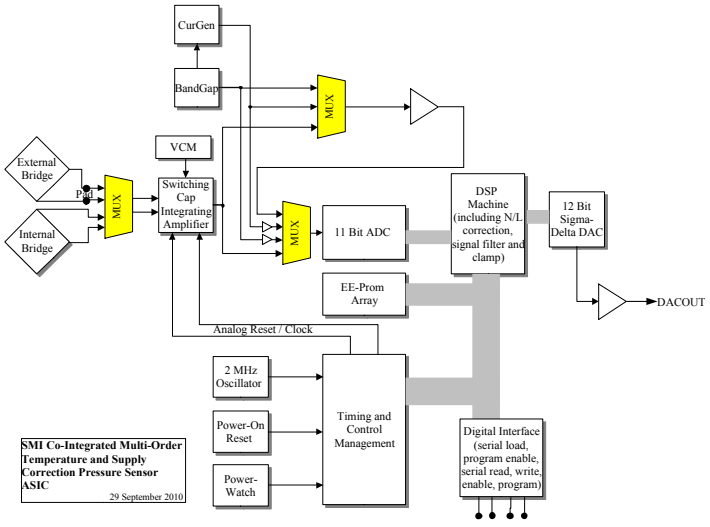

Fig.3: Functional Blockdiagram

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

Figure 4 shows a typical application schematic. Please note that in addition to the SM5800 family pressure sensor several passive components are required to ensure correct operation of the device. Resistors R1 and R2 are pull-up resistors for the I²C communication, C1a, C1b, and C3 are capacitors that improve quality of analog and digital signals.

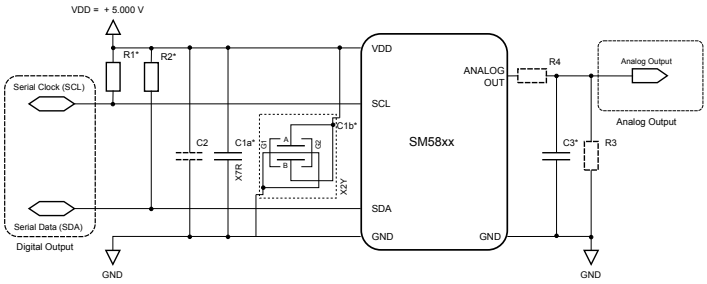

Fig. 4: Typical Application Schematic

VDD The optimum operation value for the power supply is 5 V.

R1/R2 Pull-up resistors for the digital data lines. Recommended resistor values are 10 to 100 kOhm.

C1a/C1b Buffer capacitor. For best performance of the sensor output signals it is obligatory to use a 100 nF buffer capacitor between the supply pins VDD and GND of the sensor device. SMI recommends high quality capacitors, such as X7R or X2Y. The maximum distance between the package supply pins and this capacitor should be no more than 5 mm. Usually this buffer capacitor is sufficient, but in connection with a poor power supply and to reduce power consumption a reload capacitor (C2) of min. 1 µF is advised. In this case, SMI recommends using ceramic capacitors of 47 µF.

C3 Capacitive load of the analog output. The capacitive load should be between 15 nF to 33 nF (recommended: 22 nF) with a resistor (optional: R3) of min. 3 kOhm. For a chosen lower capacitive load of 1 nF to 15 nF the resistor value should be at least 6.8 kOhm.

R4 Optional resistor for low-pass filtering (in lieu of R3) of the analog output. The recommended resistor value is: 4.7 kOhm.

* Mandatory components for best performance of pressure sensor device.

2 Timing and Algorithm Description

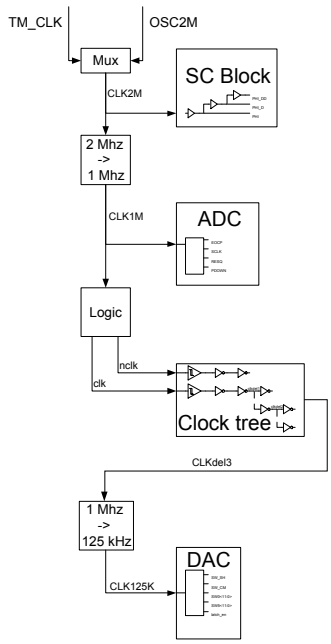

Fig. 5: Clock Tree

In order to reduce cross-talk noise from the digital to the analog part the clocking of the digital part of the ASIC is delayed.

Therefore, the analog circuit is the first part of the ASIC that is getting the clock. At this time no other part especially no digital circuit is clocking to minimize the spikes on the power supply and in the substrate. The clock tree illustrating the timing is shown in Figure 5.

The figure below shows a block diagram of the algorithm module. The DSP operates with a 15-bit word precision. Calculations for calibration and compensation of the sensor signal are performed in a sequence of steps.

In more detail the function of each algorithm module is described in the following sections:

(1) Averaging

AvTemp, *Avsupply*, *AvPress* (13-bit values) are obtained using four samples of Temperature Voltage Temp, Pressure Voltage Press and Band-gap Voltage Vbg.

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

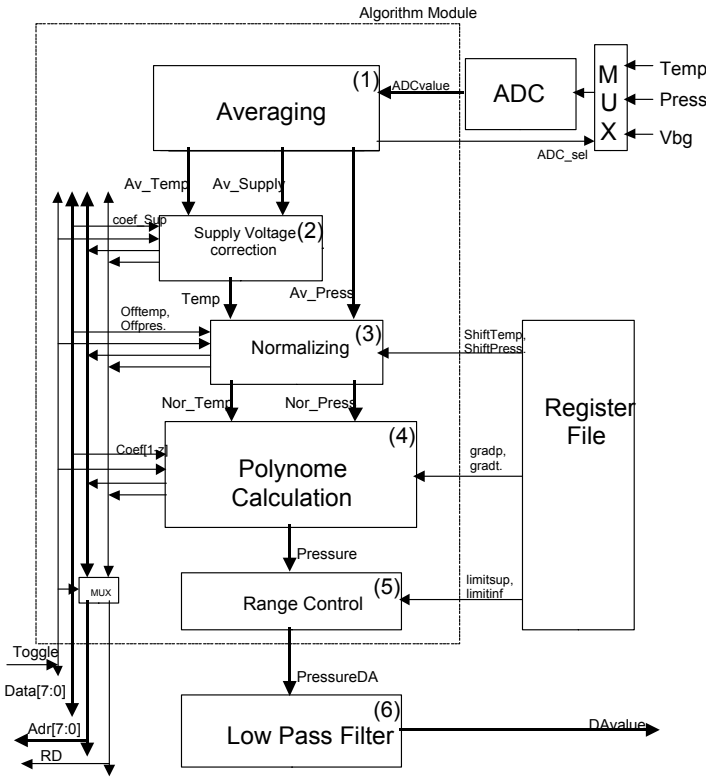

Fig. 6: Block Diagram of Algorithm Module

(2) Supply voltage correction

After the averaging step, the relationship between the AD-converted temperature and the supply voltage is:

$$Temp = \frac{AvTemp \times coefSup}{AvSupply}$$

where *coefsup* is the AD-converted *Vbg* value when the supply voltage has its nominal value.

(3) Normalizing

Next, *Temp* and *AvPress* values are normalized and stored as OffTemp, Offpres in registers R52 through R55

$$NorTemp = (Temp + OffTemp \times 4) \times 2^{ShiftTemp} - 2^{12}$$
$$NorPress = (AvPress + Offpres) \times 2^{ShiftPress} - 2^{12}$$

(4) Polynomial calculation

The output of the polynomial calculation module is Pressure:

$$\begin{aligned} P &= NorPress, \\ T &= NorTemp, \\ Pressure &= C_1 + C_2 \times P + \ldots + C_{gradp+1} \times P^{gradp} \\ &\quad + C_{gradp+2} \times T + \ldots + C_{2gradp+2} \times T \times P^{gradp} \\ &\quad + \ldots \\ &\quad + \ldots + C_{[(gradp+1)\times(gradt+1)]} \times T^{gradt} \times P^{gradp} \end{aligned}$$

The total number of coefficients is *z*:

$$z = (gradp + 1) \times (gradt + 1)$$

Example:

$$\begin{aligned} gradp &= 2, gradt = 3, \\ Pressure &= C_1 \times T^0 \times P^0 + C_2 \times T^0 \times P^1 + C_3 \times T^0 \times P^2 \\ &\quad + C_4 \times T^1 \times P^0 + C_5 \times T^1 \times P^1 + C_6 \times T^1 \times P^2 \\ &\quad + C_7 \times T^2 \times P^0 + C_8 \times T^2 \times P^1 + C_9 \times T^2 \times P^2 \\ &\quad + C_{10} \times T^3 \times P^0 + C_{11} \times T^3 \times P^1 + C_{12} \times T^3 \times P^2 \\ z &= (2+1) \times (3+1) = 3 \times 4 = 12 \end{aligned}$$

z, *gradp*, and *gradt* have these limits:

$$\begin{aligned} z &\leq 20 \\ gradp, gradt &\leq 7 \end{aligned}$$

num is the number of multiplications:

$$num = gradp \times gradt + gradp + gradt + 3$$

The calculation time (expressed in clock cycles) is obtained through this equation:

$$\text{Calculation time} \simeq 16 \times num + 30$$

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

(5) Range Control

Pressure is de-normalized to the DAC resolution of 12 bit.

$$limitinf \leq Pressure \leq limitsup$$

Next, there is the range-control check:

If $Pressure \leq limitinf$, then the output of Range control is $limitinf$.
If $Pressure \geq limitsup$, then the output of Range control is $limitsup$.

(6) Low pass filter

Finally, the output of the algorithm module is filtered:

$$Old \leftarrow Old - \frac{Old}{2^{shift}} + \frac{PressureDA}{2^{shift}}$$

$$DAvalue \leftarrow Old(12 \text{ most significant bits})$$

3 I²C Communication

I²C-Interface

The sensor-signal conditioning ASIC supports the I²C protocol. This protocol defines any device that sends data onto the bus as a transmitter and any device that reads the data as a receiver. The device that controls the data transfer is known as the master and the other as the slave. The master will always initiate a data transfer and will provide the serial clock for synchronizations. The sensor-signal conditioning ASIC is always a slave device in all communications.

Serial Clock (SCL)

The SCL input pin is used to synchronize all data in and out of the memory. It is always driven by an external master. This means a clock must be externally connected to the SMI pressure sensor. Typical clock frequencies are 100 kHz.

Serial Data (SDA)

The SDA pin is bi-directional and is used to transfer data in or out of the I²C interface. It is an open-drain output that may be wire-AND'ed with other open-drain or open-collector signals on the bus.

Start Condition

START is identified by a high to low transition of the SDA line while the clock SCL is stable in the high state. A START condition must precede any commands for data transfer.

Stop Condition

STOP is identified by a low to high transition of the SDA line while the clock SCL is stable in the high state. A STOP condition terminates communication between the sensor-signal conditioning ASIC and the bus master.

Acknowledge Bit (ACK)

An acknowledge signal is used to indicate a successful data transfer. The bus transmitter, either master or slave, will release the SDA bus after sending 8 bits of data. During the 9th clock pulse period the receiver pulls SDA bus low to acknowledge the receipt of 8 bits of data.

Data Input

During data input the ASIC samples the SDA bus signal during the high phase of the clock SCL. Note that for a correct device operation the SDA signal must be stable during the high phase of the clock SCL and the data must change ONLY when SCL line is low.

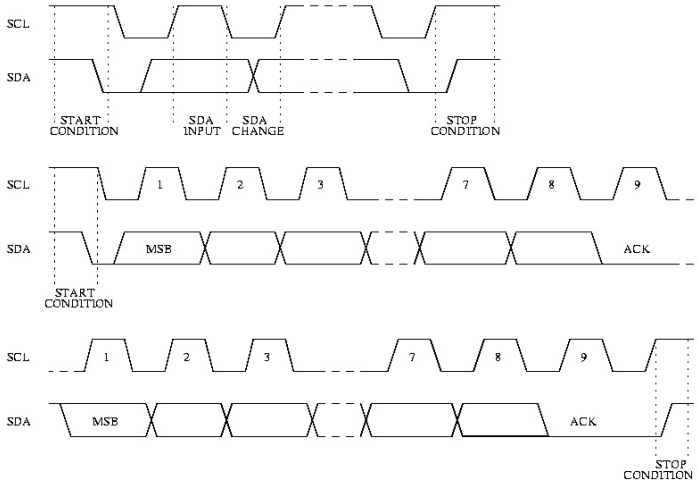

Fig. 7: I²C Timing

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

Memory Addressing

To start communication between the bus master and the slave, the master must initiate a START condition. Following this, the master sends onto the SDA bus line 8 bits (MSB first) corresponding to the device ID (7 bits) and a READ or WRITE bit. If a match is found, the slave will acknowledge the identification on the SDA bus during the 9th bit time.

Write Operation

Following a START condition the master sends a device ID with the R/W bit reset to '0'. The slave acknowledges this and waits for a byte address. The byte address of 8 bits provides access to 256 bytes of the memory (see Memory Mapping). After receipt of the byte address the device again responds with an acknowledgment.
In the Byte Write mode the master sends one data byte, which is acknowledged by the slave. The master terminates the transfer by generating a STOP condition.

Read Operation

Current Address Read

The interface has an internal byte address register, which stores the latest written address. For the Current Address Read mode, following a START condition, the master sends the device ID with the R/W bit set to '1'. The slave acknowledges this and outputs the byte addressed by the internally stored address. The master does not acknowledge the byte output, but terminates the transfer with a STOP condition.

Random Address Read

A dummy write is performed to load the address into the internal address register. This is followed by another START condition from the master and the byte address is repeated with the bit set to '1'. The interface acknowledges this and outputs the byte addressed. The master does not acknowledge the byte output, but terminates the transfer with a STOP condition.

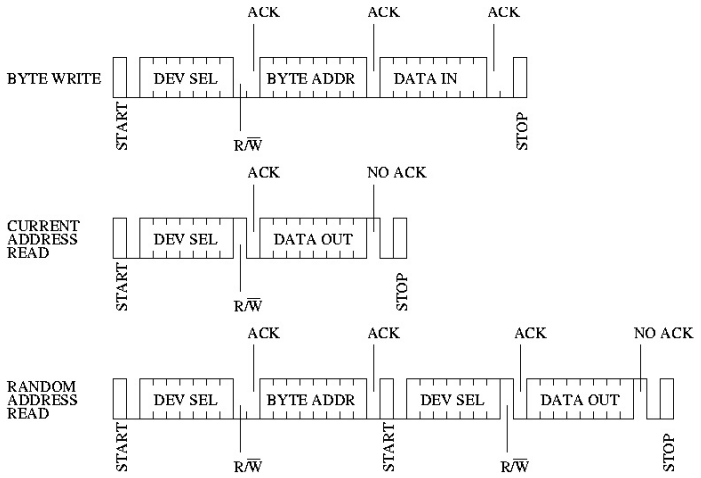

Fig. 8: I²C Read and Write Protocol

SCL clock frequencies for 1 MHz internal clock:
SCL min: > 0 kHz
SCL max: 125 kHz

Please see Section 6.2 in the appendix for I²C Pseudo Code.

4 Programming the EEPROM

General

This section describes how the EEPROM of signal-conditioning ASIC can be programmed.

In addition to the already discussed calibration coefficients, a number of bits of information need to be stored in the chip. These include:

- Oscillator frequency
- Offset set-points
- Bias supply coefficients as needed
- Device unique identifier (ID)

Memory mapping

There is an address space of 256 bytes available in the ASIC. It is divided as follows:

Address space	Description	Read/Write access
R0 to R63	EEPROM array	R+W
R64 to R127	unused	
R128 to R191	Digital registers	R+W (partially R only)
R192 to R255	EEPROM mirrored to digital registers at power-up	R+W

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

Please refer to the appendix for a detailed description of memory mapping.

Memory access

The memory access handling is managed by the ASIC, no matter if a read or write access to a digital register or read access to an EEPROM cell is performed. Only write access to the EEPROM bank requires special external support, which will be discussed in more detail further below.

The read and write operations are performed using the I²C port, with direct mapping of the 8-bit I²C address to internal memory address.

Some registers form internally wider registers and must be read at two separate address locations, those registers are:

- R128 & R129
- R130 & R131
- R132 & R133
- R134 & R135
- R140 & R141

where, e.g., R128 contains the least significant bits (LSB) and R129 contains the most significant bits (MSB). A temporary register will guarantee the consistency of the contents if read access is performed.

In general, read access should be performed in the form LSB/MSB or MSB/LSB on the corresponding registers. An exception takes place if the software skips the second read access, so that the interpretation of the value in the temporary register is wrong. Therefore, always perform an initial read access to a corresponding registers pair as follows:

- two sequential accesses to LSB (the temporary register is filled with the correct MSB on the second read)
- one access to MSB

or

- two sequential accesses to MSB (the temporary register is filled with the correct LSB on the second read)
- one access to LSB

Afterwards the software can follow the instruction for a general read access.

All access is performed with the help of the I2C bus interface. The write-protect bit must be disabled when writing or deleting of the EEPROM is performed:

R47[0] = 1: write protect disable
R47[0] = 0: write protect enable

The device ID is factory programmed by writing it to EEPROM register R63. The customer can change the device ID.

The device ID check is bypassed only if the DEVICE_SW pad is pulled high (1) externally. Bypassing means, that the device will answer to any ID used by a master on I²C bus. DEVICE_SW is pulled down internally, which means that in normal operation with this pin unconnected the device ID is always checked.

Note: *The device is shipped with a valid device ID programmed. The default ID value is decimal 95 (hexadecimal 0x5F).*

EEPROM Write-Sequence (For Single Register/EEPROM Locations)

The following procedure can be used to write data to single registers/EEPROM locations. This is used, e.g., to change the I²C-address of the device to an arbitrary value. The EEPROM programming voltage as specified must be applied to EEPROM_VPROG.

Note: *All time data is referring to the nominal clock frequency 1 MHz.*

1. Register 143, first bit set to 1: this enables the erase bit
2. Apply 16 V to pin 1 of the device
3. Write data 0xFF to EEPROM-cell
4. Wait minimum 13 ms
5. Register 143, first bit set to 0: this disables the erase bit
6. Write data, e.g. I²C device ID, to device
7. Wait minimum 13 ms
8. Turn off 16 V at pin 1 of the device
9. Reset the device: pull VDD to low or turn of VDD
10. Verify that data was programmed

Example

This section demonstrates how to change the default I2C-device ID 95 of a device to the I2C-device ID 96. This is done by writing the data value 0xC0 (= device ID 96 (decimal), shifted left by one bit) to the register 0x3F (register 63 (decimal)) of a device with value 0xBE (= device ID 95 (decimal), shifted left one bit).

1. Register 143, first bit set to 1:
   Write [0x5F] [0x8F] [0x01] to I²C bus
2. Apply 16 V to pin 1 of the device
3. Write data FF to EEPROM-cell
   Write [0x5F] [0x3F] [0xFF] to I²C bus
4. Wait minimum 13 ms
5. Register 143, first bit set to 0: this disables the erase bit
   Write [0x5F] [0x8F] [0x00] to I²C bus
6. Write data, e.g. I2C device ID, to device
   Write [0x5F] [0x3F] [0xC0] to I²C bus

Phone: +1-(408) 577-0100 / **eMail:** sales@si-micro.com / **www:** http://www.si-micro.com

7. Wait minimum 13 ms
8. Turn off 16 V at pin 1 of the device
9. Reset the device: pull VDD to low or turn of VDD
10. Verify that data was programmed
    Write [0x60] [0x3F] to I²C bus
    Read 1 byte from I²C bus: the result should be 0xC0 and not 0xBE (default)

Comments:

- The Totalphase Aardvark I²C/USB interface (http://www.totalphase.com) and an SMI demonstration board were used verify this procedure. The procedure has also been verified using a PIC18 Microchip microcontroller with an I²C interface on an SMI demonstration board.
- During the writing process the EEPROM cell bits can only be pulled down from 1 to 0. Thus, the cell has to be erased to value 0xFF before writing. Not properly deleting the cells still enables writing but can lead to wrong data in the cell.
- The device ID that is written to the EEPROM has to be shifted, e.g. to write address 0x60, left shift this value and thus write 0xC0 to the EEPROM.

5 FAQs

5.1 What are the features and how to use them?

I²C

I²C is a bidirectional digital interface using two lines for communication, Serial Data Line (SDA) and Serial Clock (SCL).

In the SM5800 series parts, the I²C interface is for example used to store information in the EEPROM during calibration but also to read out pressure and temperature information in the final application.

The I²C bus has two roles for nodes: master and slave. The master node issues the clock and addresses slaves; the slave node receives the clock line and address.

The SM5800 family parts can only be used as a slave on the I²C bus. The factory programmed address is decimal 95.

Please refer to Section 3 for details on the I²C communication.
Please refer to Section 4 for details on how to program the EEPROM using the I²C interface.

Analog output

All parts from the SM5800 family provide an analog output signal. A DAC and an output amplifier are used to convert the digitally corrected signal back into an analog output that can be accessed by connecting to the respective pin of the package.

For absolute and gauge type pressure sensors, the zero output is typically 0.5 V and full-scale output is 4.5 V. For differential type pressure parts, the negative full-scale output is typically at 0.5 V, zero output is typically 2.5 V, and positive full-scale output is 4.5 V for a span of 2.0 V.

Note: The analog output does not provide more accuracy than the digital output. It is the output of the DAC that converts the digital output into an analog value.

Please refer to Section 1 for details on functional description.

Digital output

The digital output of the SM5800 parts is stored in the EEPROM registers as follows:

Register (decimal)	Register (hexadecimal)	Register (binary)	Description
128	80	1000 0000	Corrected pressure, LSB [5:0]
129	81	1000 0001	Corrected pressure, MSB [11:6]
130	82	1000 0010	Temperature, LSB [5:0]
131	83	1000 0011	Temperature, MSB [11:6]

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

Register (decimal)	Register (hexadecimal)	Register (binary)	Description
132	84	1000 0100	Uncorrected pressure, LSB [5:0]
133	85	1000 0101	Uncorrected pressure, MSB [11:6]
134	86	1000 0110	Supply voltage, LSB [5:0]
135	87	1000 0111	Supply voltage, MSB [11:6]

The I²C interface can be used to read the values from the respective registers.

Please refer to Section 3 for details on the I²C communication. Please refer to Section 4 for details on how to program the EEPROM using the I²C interface.

Every value is represented by 12 bits. That corresponds to 4096 steps. However, not the whole number range is used to represent the pressure value.

Please refer to Section 5.2 for details on how to interpret the digital output.

5.2 How to interpret the digital output?

The corrected pressure (registers $R128_{10}$ and $R129_{10}$)

Starting at 410 corresponding to the low scale pressure value. The ideal value referring to the full scale value is 3686. That means that there are 3276 steps between zero and full scale pressure.

Example: *The 5882 S has a range from 0 to 0.6 psi. 0.6 divided by 3276 results in 1.8315×10^{-4} psi per digital step. Reading 2800 as a digital value from the register corresponds to:*

$$(2800 - 410) \times 1.8315 \times 10^{-4} \text{ psi} = 0.438 \text{ psi}$$

For a differential part, zero psi is:

$$410 + \frac{3276}{2} = 2048$$

So there are 1638 steps to full-scale in each direction.

Example: *The 5882 D has a range from 0 to 0.6 psi. 0.6 divided by 1638 results in 3.663×10^{-4} psi per digital step. Reading 2800 as a digital value from the register corresponds to*:

$$(2800 - 2048) \times 3.663 \times 10^{-4} \text{ psi} = 0.275 \text{ psi}$$

Temperature

There is no corrected temperature accessible. However, the dependence of the digital value to the temperature is quite linear. Therefore, linear interpolation can be used to determine temperature values.
To determine what temperature belongs to a certain digital value just use these two points to connect a line and the temperature can be calculated.

Example: *The two lookup temperatures are given with 2770 LSB belonging to 15°C and 2100 LSB belonging to 75°C. If you read a temperature of 2500 LSB that corresponds to*:

$$15°\mathcal{C} + (2500 - 2770) \times \frac{(75°\mathcal{C} - 15°\mathcal{C})}{2100 - 2770} = 39°\mathcal{C}$$

Uncorrected pressure

Since the corrected pressure is accessible there should be no need to use the uncorrected pressure.

Supply Voltage

A digital value of about 1000 corresponds to the designated supply voltage of 5V.

5.3 Can the SM5800 sensor be used to measure temperature?

There is no corrected calibrated temperature value accessible. However, the dependence of the digital values to the temperature is quite linear. Therefore, linear interpolation can be used to determine temperature values. For that purpose two reference data points are necessary where the known temperature is matched with the digital temperature value.

Please refer to Section 5.2 for more details on how to interpret the digital output

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

6 Appendix

6.1 Detailed Memory Mapping

Reg	Binary Addr.	Description
R0	0 000 0000	[5:3] : gradt [2:0] : gradp
R1	0 000 0001	reserved
R2	0 000 0010	[7:0] : C(Z)[7:0]
R3	0 000 0011	[5:0] : C(Z)[13:8]
:	:	[7/5:0] : LSBs and MSBs of C(Z-1) to C(2) 2nd complement notation: SMM,NNNNNNNNNNN
RX	0 010 1000	[7:0] : C(1)[7:0] RX depends on the number of Coefficients; RXmax=R40;
RY	0 010 1001	[5:0] : C(Z)[13:8] RY=RX+1;
R42	0 010 1010	reserved
R43	0 010 1011	reserved
R44	0 010 1100	1/coefsup [7:0]
R45	0 010 1101	1/coefsup [13:8]
R46	0 010 1110	reserved
R47	0 010 1111	[0] : EEPROM write protection (low activ); write '0' to activate write protection
R48	0 011 0000	[5:0] : maximal Pressure value[5:0]
R49	0 011 0001	[5:0] : maximal Pressure value[11:6]
R50	0 011 0010	[5:0] : minimal Pressure value[5:0]
R51	0 011 0011	[5:0] : minimal Pressure value[11:6]
R52	0 011 0100	[5:0] : Offset_temp[5:0] (reserved for SMI internal use only, must be 0)
R53	0 011 0101	[4:0] : Offset_temp[10:6] (reserved for SMI internal use only, must be 0)
R54	0 011 0110	[5:0] : Offset_press[5:0] (reserved for SMI internal use only, must be 0)
R55	0 011 0111	[4:0] : Offset_press[10:6] (reserved for SMI internal use only, must be 0)

Reg	Binary Addr.	Description
R56	0 011 1000	reserved (not usable for copy into register array)
R57	0 011 1001	reserved (not usable for copy into register array)
R58	0 011 1010	[3:0] : Clock adjust; calibrate clock to 1 Mhz ("0000" highest frequency)
R59	0 011 1011	[3:0] : shift (LowPassFilter) (valid values: "0000" to "1000")
R60	0 011 1100	[3:2] : normalizing Shift_temp [1:0] : normalizing Shift_press (reserved for SMI internal use only, must be 0)
R61	0 011 1101	[6] : EXTCsens, pad TM_EXTC is used as pressure signal for ADC [5:2] : Offset for analog [1] : Offset sign for analog [0] : E_Bridge for analog
R62	0 011 1110	[5:2] : Gain for analog [1] : Input (gain) polarity for analog [0] : nSym for analog
R63	0 011 1111	[7:1] : I2C_DEV_ID [0] : don't care (is I²C read/write)
Not mapped R64-R127		
R64	0 100 0000	
:	:	
R127	0 111 1111	
Algorithm-Registers R128-R135 (r)		
R128	1 000 0000	[5:0] : digital corrected Pressure value[5:0] (=DAC-value)
R129	1 000 0001	[5:0] : digital corrected Pressure value[11:6] (=DAC-value)
R130	1 000 0010	[5:0] : ADC Temp value[5:0]
R131	1 000 0011	[5:0] : ADC Temp value[11:6]

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

Reg	Binary Addr.	Description
R132	1 000 0100	[5:0] : ADC Press value[5:0]
R133	1 000 0101	[5:0] : ADC Press value[11:6]
R134	1 000 0110	[5:0] : ADC Supply value[5:0]
R135	1 000 0111	[5:0] : ADC Supply value[11:6]
R143	1 000 1111	EREG (EEPROM-ControlRegister); ResetValue : "0000000" [6] : INFO-Mode; EEPROM-Test-Mode must be enabled (rw) [5] : HIREFB; EEPROM-Test-Mode must be enabled; (rw) 0: low activ HIREFB-Signal is '1' 1: low activ HIREFB-Signal is '0' [4] : HIVRD; EEPROM-Test-Mode must be enabled (rw) [3] : VPP_on EEPROM-Test-Mode on: (rw) manually VPP_on pulse for EEPROM-Test-Mode; (make programming time adjustable;automatic 13 ms write is disabled;) EEPROM-Test-Mode off: (r) is high when VPP_on is automatically generated [2] : WR_on EEPROM-Test-Mode on : (rw) manually Write pulse for EEPROM-Test-Mode; (make programming time adjustable; automatic 13 ms write is disabled) EEPROM-Test-Mode off: (r) is high when writing or erasing EEPROM is automatically generated [1] : ALL; for erase and write (rw) [0] : Erase (rw) EEPROM-Test-Mode on: R143[1:0] must be set manually; for erase also set R143[3] (manually VPP_on) erase-signal is high during R143[0] is high EEPROM-Test-Mode off: (rw) R143[1:0]="11" : erase complete EEPROM immedently with 13 ms pulse; then clear R143[1:0] automatically with erase pulse start R143[1:0]="01" : when writing then erase addresse EEPROM cell R143[1:0]="10" : writing data in ALL EEPROM cells when writing in one;
R144	1 001 0000	TestRegister [2] : EEPROM-Test-Mode on 0: EEPROM full automatically mode 1: EEPROM Test Mode access to EEPROM test modes make programming time adjustable; automatic 13 ms write is disabled; [1] : ADC_Test on, pad TM_EXTC is connect to ADC input a high current load TM_EXTC, when ADC_Test_Value (R148,R149) > ADCvalue a low current load TM_EXTC, when ADC_Test_Value < ADCvalue (ADC_sel = "111") [0] : DAC_Test on; SCamplifier off, 0: DACvalue is calculated Polynom value 1: DAC_Test_Value (R148,R149)

Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

Reg	Binary Addr.	Description
R145	1 001 0001	[4:3] : MUXsc direct connect to TMPAD_SC "01" :SC_VCM, "10"=SC_OUT, others=NoConnect [2:0] : MUXbuf for buffered analog output TM_OUT "000": nTM_EXTC, "001": nTM_VBG, (resetValue) "010": nTM_VTEMP, "011": nTM_VSC, "100": TM_BR_INT, others: nTM_EXTC ResetValue : "001"
R146	1 001 0010	reserved
R147	1 001 0011	reserved
R148	1 001 0100	[5:0] : Test_Value[5:0] for DAC or ADC
R149	1 001 0101	[5:0] : Test_Value[11:6] for DAC [4:0] : Test_Value[10:6] for ADC
R150	1 001 0110	reserved
R151	1 001 0111	[0] : WatchRegister, was Power < 4V the register is set automatic reset when reading or POR
R152	1 001 1000	same as R136
R153	1 001 1001	same as R137
R168	1 010 1000	same as R136
R169	1 010 1001	same as R137

Reg	Binary Addr.	Description
R184	1 011 1000	same as R136
R185	1 011 1001	same as R137
Mirrored EEPROM Cells after reset (rw)		
R192	1 100 0000	Copy of R0
R200	1 100 1000	same as R136
R201	1 100 1001	same as R137
R216	1 101 1000	same as R136
R217	1 101 1001	same as R137
R232	1 110 1000	same as R136
R233	1 110 1001	same as R137
R239	1 110 1111	Copy of R47 (r)
R240	1 111 0000	Copy of R48
R241	1 111 0001	Copy of R49
R242	1 111 0010	Copy of R50
R243	1 111 0011	Copy of R51
R248	1 111 1000	same as R136
R249	1 111 1001	same as R137
R250	1 111 1010	Copy of R58
R251	1 111 1011	Copy of R59
R252	1 111 1100	Copy of R60
R253	1 111 1101	Copy of R61
R254	1 111 1110	Copy of R62
R255	1 111 1111	Copy of R63

6.2 I²C Pseudo Code

This Section contains pseudo code for the I²C random access read process. It can be used as a guide line for implementation of I²C communication when using microcontrollers.

```
//********** MAIN PROGRAM **********
START
  START_CONDITION;
  DEVICE_ADDRESS;  //Set the read/write bit to 0 (= write)
  ACKNOWLEDGEMENT;
  REGISER_ADDRESS;
  ACKNOWLEDGEMENT;
  RE-START_CONDITION:
  DEVICE_ADDRESS;  //Set the read/write bit to 1 (= read)
  ACKNOWLEDGEMENT;
  DATA_OUT;
  NO_ACKNOWLEDGEMENT;
  STOP_CONDITION;
END

//********** SUB-ROUTINES **********
START_CONDITION
  SDA(W)=0;                    //Set SDA bus line “low”
  DELAY;
  SCL(W)=1;                    //Set SCL bus line “high”
  DELAY;
```


Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

```
DEVICE_ADDRESS
  n=7;
  FOR i=0 to n-1
    SDA(W)=DEVICE_ADD(i);  //Writing ith bit of the Device
                           //Address
    DELAY;
    SCL(W)=0;              //Set SCL bus line "low"
    DELAY;
    SCL(W)=1;              //Set SCL bus line "high"
    DELAY
    i=i+1;
  END;
  SDA(W)=R_or_W            //Set the read/write bit
  DELAY;
  SCL(W)=0;                //Set SCL bus line "low"
  DELAY;
  SCL(W)=1;                //Set SCL bus line "high"
  DELAY;

ACKNOWLEDGEMENT
  SDA(W)=1;                //Set SDA bus line "high"
  DELAY;
  SCL(W)=0;                //Set SCL bus line "low"
  DELAY;
  ACK=SDA(R);              //Reading ACK bit
  DELAY;
  SCL(W)=1;                //Set SCL bus line "high"
  DELAY;

REGISTER_ADDRESS
  n=8;
  FOR i=0 to n-1
    SDA(W)=REGISTER_ADD(i); //Writing ith bit of the
                            //Register Address
    DELAY;
    SCL(W)=0;               //Set SCL bus line "low"
    DELAY;
    SCL(W)=1;               //Set SCL bus line "high"
    DELAY
    i=i+1;
  END;

RE-START_CONDITION
  SDA(W)=1;                //Set SDA bus line "high"
  DELAY;
  SCL(W)=0;                //Set SCL bus line "low"
  DELAY;
  SDA(W)=0;                //Set SDA bus line "low"
  DELAY;
  SCL(W)=1;                //Set SCL bus line "high"
  DELAY;

DATA_OUT
  n=8;
  FOR i=0 to n-1
    DATA(i)=SDA(R);        //Reading ith bit of Data Out
    DELAY;
    SCL(W)=0;              //Set SCL bus line "low"
    DELAY;
    SCL(W)=1;              //Set SCL bus line "high"
    DELAY
    i=i+1;
  END;

NO_ACKNOWLEDGEMENT
  SDA(W)=1;                //Set SDA bus line "high"
  DELAY;
  SCL(W)=0;                //Set SCL bus line "low"
  DELAY;
  SCL(W)=1;                //Set SCL bus line "high"
  DELAY;

STOP_CONDITION
  SDA(W)=0;    //Set SDA bus line "low"
  DELAY;
  SCL(W)=0;    //Set SCL bus line "low"
  DELAY;
  SDA(W)=1;    //Set SDA bus line "high"
  DELAY;
```


Phone: +1-(408) 577-0100 / eMail: sales@si-micro.com / www: http://www.si-micro.com

7 Change History

Revision	Date	Description of changes	Approval
00	20-Oct-2010	Initiate	BAL MDOELLE MMANZANO STERRY
01	10-Aug-2011	Removed table from section 5.2.	BAL MDOELLE MMANZANO PKURZAWS RMO
02	29-Sep-2011	Corrected calculation in section 5.2.	BAL MDOELLE MMANZANO PKURZAWS RMO

Phone: +1-(408) 577-0100 / **eMail:** sales@si-micro.com / **www:** http://www.si-micro.com

8 Warranty and Disclaimer

Silicon Microstructures Warranty and Disclaimer:

Information in this document is provided solely to enable software and system implementers to use Silicon Microstructures, Inc. (SMI) products and/or services. No express or implied copyright licenses are granted hereunder to design or fabricate any silicon-based microstructures based on the information in this document.

SMI makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does SMI assume any liability arising out of the application or use of any product or silicon-based microstructure, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in SMI’s data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. SMI does not convey any license under its patent rights nor the rights of others. SMI makes no representation that the circuits are free of patent infringement. SMI products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SMI product could create a situation where personal injury or death may occur. Should Buyer purchase or use SMI products for any such unintended or unauthorized application, Buyer shall indemnify and hold SMI and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that SMI was negligent regarding the design or manufacture of the products.

SMI warrants goods of its manufacture as being free of defective materials and faulty workmanship. SMI standard product warranty applies unless agreed to otherwise by SMI in writing. Please refer to your order acknowledgement or contact SMI directly for specific warranty details. If warranted goods are returned to SMI during the period of coverage, SMI will repair or replace, at its option, without charge those items it finds defective. The foregoing is buyer’s sole remedy and is in lieu of all warranties, expressed or implied, including those of merchantability and fitness for a particular purpose. In no event shall SMI be liable for consequential, special, or indirect damages.

While SMI may provide application assistance to aid its customers' design process, it is up to each customer to determine the suitability of the product for its specific application. The information supplied by SMI is believed to be accurate and reliable as of this printing. However, SMI assumes no responsibility for its use. SMI assumes no responsibility for any inaccuracies and/or errors in this publication and reserves the right to make changes to any products or specifications herein without further notice.

 Phone: +1-(408) 577-0100 / **eMail:** sales@si-micro.com / **www:** http://www.si-micro.com