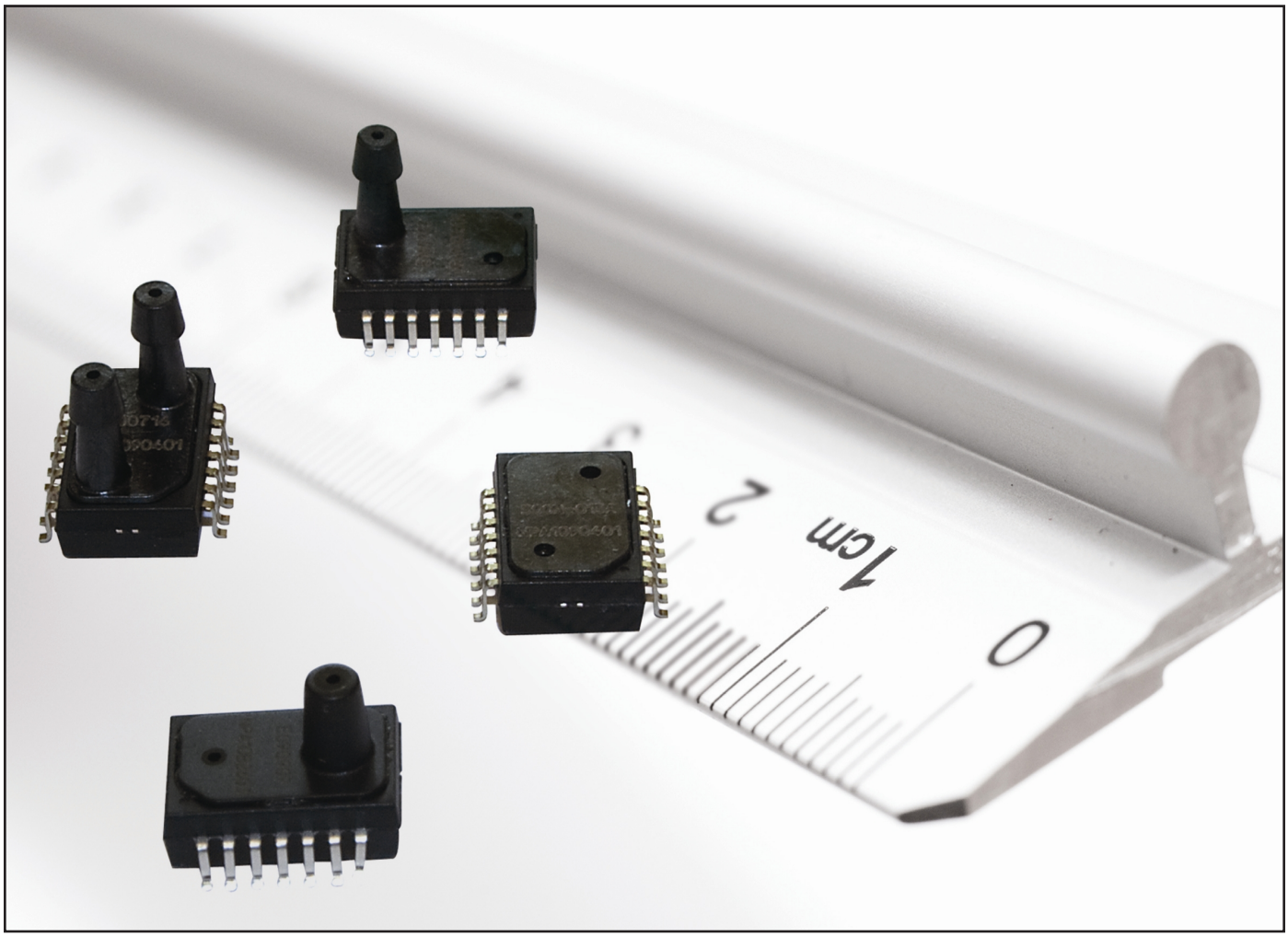
NOVA® SENSOR

# Pressure Sensor Type NPA

## Application Guide




Amphenol

Advanced Sensors

AAS-910-289 Rev. E
July 2014

1. General Description . . . . . 1
2. Output Characteristics . . . . . 1
2.1 Pressure Output Options . . . . . 1
2.2 Pressure Output Characteristics (Calibrated Types) . . . . . 2
2.3 Pressure Accuracy (Calibrated Types) . . . . . 3
2.4 Temperature Output Characteristics (Digital Types) . . . . . 4
3. Analog Output (Amplified and Compensated) . . . . . 5
3.1 General . . . . . 5
3.2 Block Diagram . . . . . 5
3.3 Pinout . . . . . 6
3.4 Typical Application Circuits . . . . . 7
3.5 Output Impedance . . . . . 8
3.6 Output Short Protection . . . . . 8
4. Digital Output (I2C) . . . . . 9
4.1 General . . . . . 9
4.2 Block Diagram . . . . . 9
4.3 I2C Timing . . . . . 10
4.4 Data Read Operations . . . . . 11
4.5 Pinout . . . . . 12
4.6 Typical Application Circuit . . . . . 12
4.7 Changing I2C Slave Address . . . . . 13
4.8 Diagnostic Features . . . . . 15
4.9 NPA-700 Sleep Mode . . . . . 15
5. Digital Output (Manchester Code) . . . . . 16
5.1 General . . . . . 16
5.2 Block Diagram . . . . . 16
5.3 Pinout . . . . . 17
5.4 Typical Application Circuits . . . . . 17
5.5 Bit Encoding . . . . . 18
5.6 Mode 1 - Pressure Only . . . . . 19

5.7 Mode 2 - Pressure With Temperature . . . . . 20
5.8 Data Transmission . . . . . 20
5.9 How to Read a Data Packet . . . . . 21
5.10 How to Read a Data Packet Using a Microcontroller . . . . . 21
5.11 8051 C++ Code Example . . . . . 22
5.12 PIC1 Assembly Code Example . . . . . 25
6. Uncalibrated Versions . . . . . 28
6.1 General . . . . . 28
6.2 Block Diagram . . . . . 28
6.3 Pinout . . . . . 29
7. Package Dimensions . . . . . 30
7.1 Manifold Fitting . . . . . 30
7.2 Barbed Fitting . . . . . 31
7.3 No Port . . . . . 33
7.4 Dead Volume . . . . . 34
8. Pressure Port Connections . . . . . 34
9. Suggested PCB Pad Layout . . . . . 35
10. Soldering . . . . . 35
10.1 Standard Reflow Soldering . . . . . 35
10.2 Manual Soldering . . . . . 35
11. Device Branding . . . . . 36

# 1. General Description

The NPA series of pressure sensors combines Amphenol Advanced Sensors' SenStable® silicon fusion bonded pressure die technology with packaged electronics to provide highly stable, amplified and calibrated pressure measurement in a cost effective surface mount package.

The NPA is offered in a range of pressure ratings. Various port configurations are available to measure absolute, gauge and differential pressures. Versions are offered with either analog or digital outputs.

This application guide should be read in conjunction with product data sheet 920-477.

# 2. Output Characteristics

## 2.1 Pressure Output Options

The NPA series is available in a range of pressure ratings, pressure configurations and output formats. The nominal output ranges for standard product ranges are detailed in Tables 1 and 2 below.

**Table 1: Available Pressure Output Ranges for Standard Products**

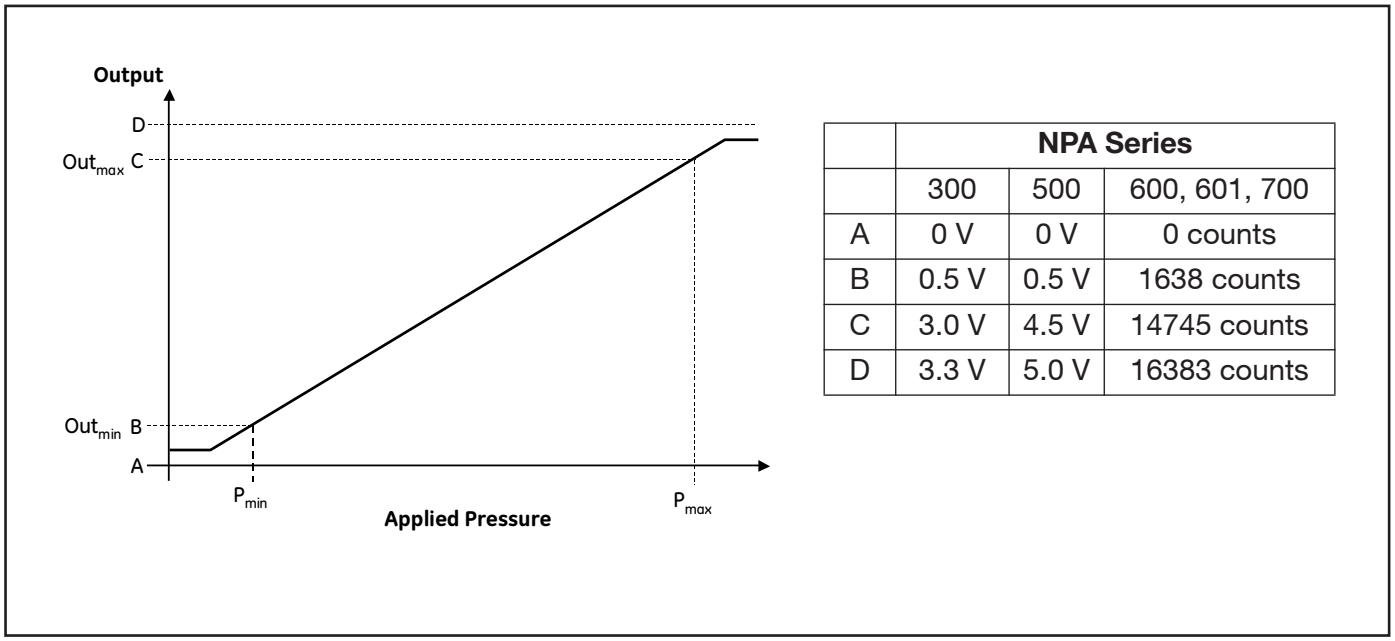
| Device series | Units | Absolute/Gauge | | Differential | | | Excitation |
|---|---|---|---|---|---|---|---|
| | | Applied pressure | | | | | |
| | | 0 | + Rated | - Rated | 0 | +Rated | |
| | | | | | | | |
| NPA-100 (10"H2O) | mV | 0 | 80 | -80 | 0 | 80 | 1.5 mA |
| NPA-100 (1 psi) | mV | 0 | 105 | -105 | 0 | 105 | 1.5 mA |
| NPA-100 (5 to 30 psi) | mV | 0 | 133 | -133 | 0 | 133 | 1 mA |
| | | | | | | | |
| NPA-300 | V | 0.50 | 3.00 | 0.50 | 1.75 | 3.00 | 3.3 V |
| NPA-500 | V | 0.50 | 4.50 | 0.50 | 2.50 | 4.50 | 5 V |
| | | | | | | | |
| NPA-600, NPA-601 | Counts | 1638 | 14745 | 1638 | 8192 | 14745 | 2.7 to 5.5 V |
| NPA-700 | Counts | 1638 | 14745 | 1638 | 8192 | 14745 | 2.7 to 5.5 V |

**Table 2: Example – Output Range of NPA-500M-001D**

| Applied Pressure (psid) | Output (V) |
|---|---|
| -1 | 0.5 |
| 0 | 2.5 |
| +1 | 4.5 |

## 2.2 Pressure Output Characteristics (Calibrated Types)

All NPA types, excluding NPA-100 series, are factory calibrated to give a defined linear change in output over a specified range of pressures. Outside this range, the output varies with pressure but with unspecified accuracy. The output is clipped at low and high pressures as the output approaches the supply rails.




| | NPA Series | | |
|---|---|---|---|
| | 300 | 500 | 600, 601, 700 |
| A | 0 V | 0 V | 0 counts |
| B | 0.5 V | 0.5 V | 1638 counts |
| C | 3.0 V | 4.5 V | 14745 counts |
| D | 3.3 V | 5.0 V | 16383 counts |

**Figure 1: Output Characteristic of Calibrated Types**

Pressure can be calculated from the sensor output using the following formula:

$$P = P_{min} + \left(\frac{Out - Out_{min}}{Out_{max} - Out_{min}}\right).(P_{max} - P_{min})$$

where P = calculated pressure

Out = measured sensor output

## 2.3 Pressure Accuracy (Calibrated Types)

The measurement accuracy of calibrated NPA types is defined in terms of ±% of full scale (FS) over a specified compensated temperature range. Standard parts are specified to be ±1.5%. This accuracy figure is a total error band and includes all errors due to offset, span, linearity and temperature, as illustrated in Figure 2 below.

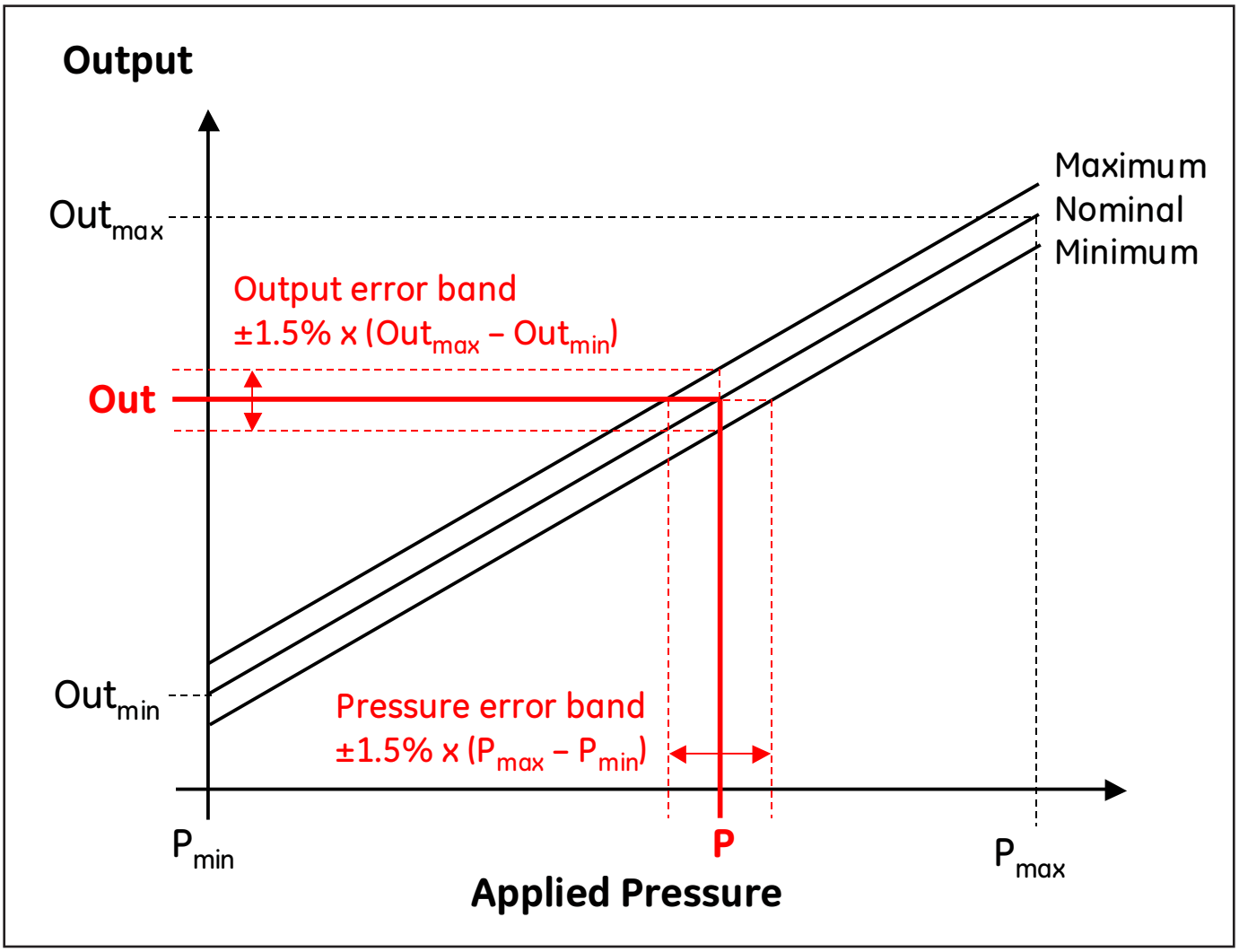



**Figure 2: Accuracy of Calibrated NPA Types**

**Table 3: Pressure Errors**

| | Absolute / Gauge | | Differential | |
|---|---|---|---|---|
| **Rated pressure** | **FS** | **Error** | **FS** | **Error** |
| 10 ”H2O | 10 ”$H_2O$ | ± 0.15” $H_2O$ | 20” $H_2O$ | ± 0.30” $H_2O$ |
| 1 psi | 1 psi | ± 0.015 psi | 2 psi | ±0.03 psi |
| 5 psi | 5 psi | ± 0.075 psi | 10 psi | ±0.15 psi |

**Table 3: Pressure Errors**

| | | | | |
|---|---|---|---|---|
| 15 psi | 15 psi | ± 0.225 psi | 30 psi | ±0.45 psi |
| 30 psi | 30 psi | ± 0.45 psi | 60 psi | ±0.90 psi |

## 2.4 Temperature Output Characteristics (Digital Types)

The NPA-601 and NPA-700 series output digital temperature data as shown in Table 4 below.

Note: Accuracy of the temperature reading is not guaranteed. The calculated value should only be used as an approximate indication.

**Table 4: Digital Temperature Output**

| Series | Bits | Conversion from Count Value to Temperature °C |
|---|---|---|
| NPA-601 | 8 | Temperature = (Counts x 200 / $2^8$) - 50 |
| NPA-700 | 8 | Temperature = (Counts x 200 / $2^8$) - 50 |
| NPA-700 | 11 | Temperature = (Counts x 200 / $2^{11}$) - 50 |

# 3. Analog Output (Amplified and Compensated)

## 3.1 General

The NPA-300 and NPA-500 series output pressure data as analog voltage:

- NPA-300: 3.3 V dc supply, 0.5 to 3.0 V output
- NPA-500: 5 V dc supply, 0.5 to 4.5 V output

Standard parts are configured for ratiometric operation; i.e., the output varies linearly with supply voltage. For absolute operation (output independent of supply voltage), special versions of the NPA are available with an additional pin configured to control the gate of an external FET.

## 3.2 Block Diagram

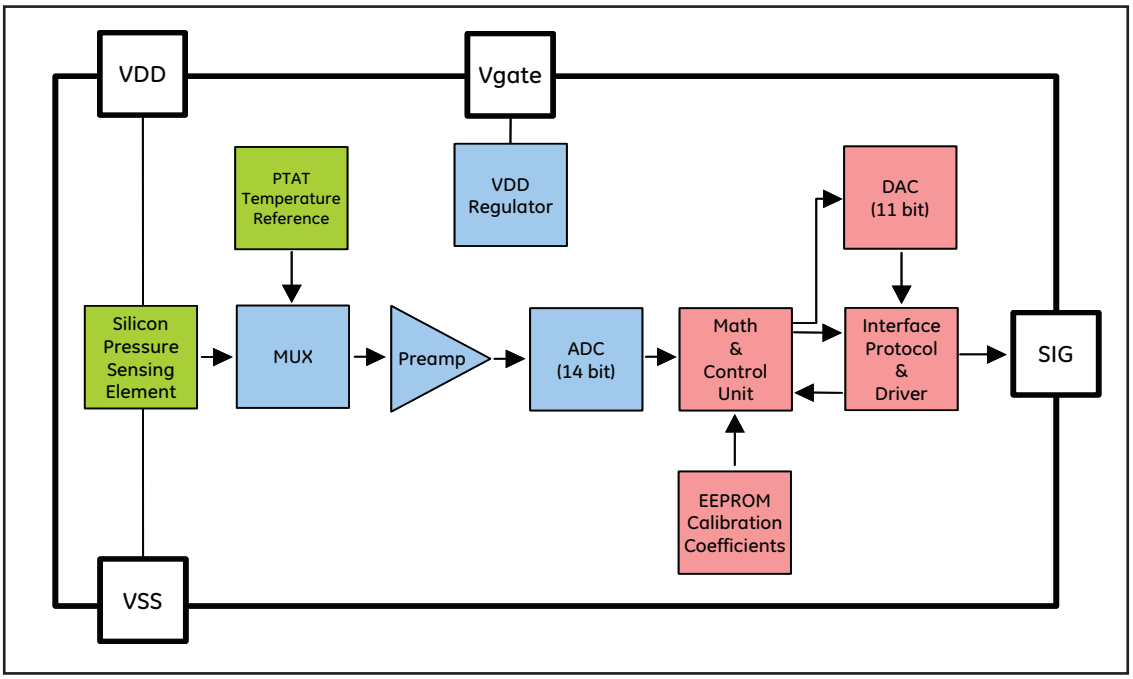



**Figure 3: Block Diagram of NPA-300, NPA-500 Series**

## 3.3 Pinout

**Table 5: Pinout of NPA-300, NPA-500 Series**

| Pin # | | Function |
|---|---|---|
| 1 | | NC |
| 2 | | NC |
| 3 | | NC |
| 4 | | NC |
| 5 | | NC |
| 6 | VSS | Ground |
| 7 | | NC |
| 8 | SIG | Output |
| 9 | VDD | Supply Voltage |
| 10 | Vgate | FET Gate Control (special order) |
| 11 | | NC |
| 12 | | NC |

Note:
NC= No connection should be made to these pins

## 3.4 Typical Application Circuits

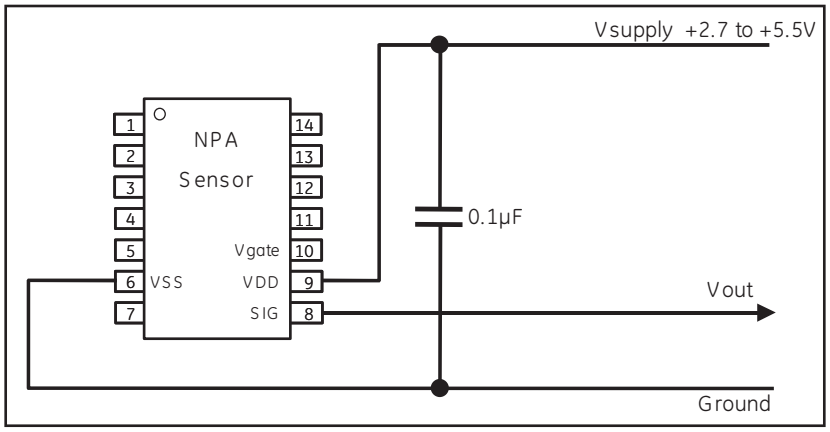



**Figure 4: Ratiometric Voltage Output**

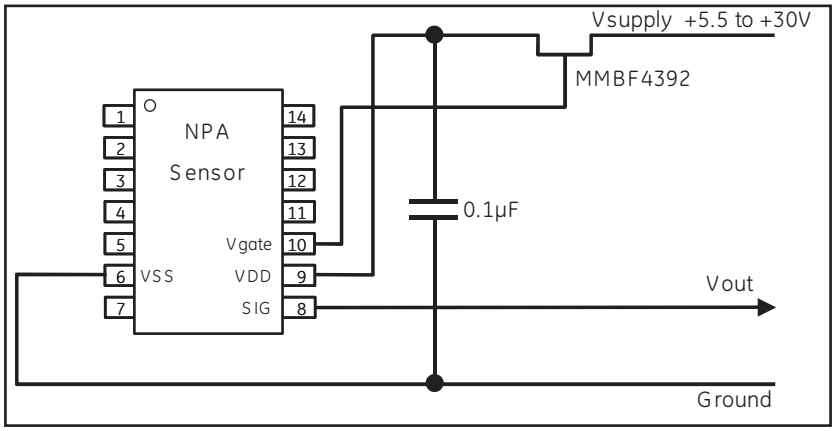



**Figure 5: Absolute Voltage Output**

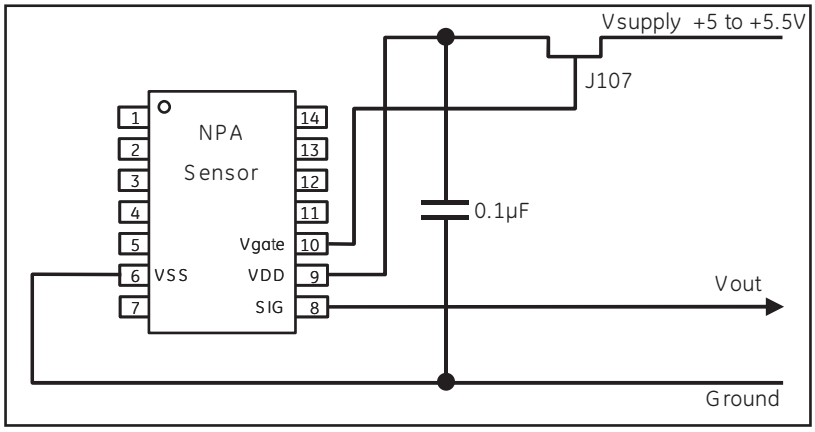



**Figure 6: Ratiometric Voltage Output with Overvoltage Protection**

**Note:** *Standard analog versions of NPA are configured for ratiometric operation as shown in Figure 4. For the circuits in Figures 5 and 6, a special version of the NPA with pin 10 (Vgate) activated should be used.*
*Typical output load resistor $R_L$ (to $V_{supply}$ or Ground) = 10 kΩ (minimum 2.5 kΩ)*
*Typical output load capacitance $C_L$ = 10 nF (maximum 15 nF)*

## 3.5 Output Impedance

The source impedance defined by voltage drop due to load current is not specified as such for the NPA. The sensor incorporates a unity gain buffer at the output to control the output voltage within a range of load current/resistances. An error compensation circuit tracks and reduces the amplifier offset voltage to <1 mV.

## 3.6 Output Short Protection

The NPA does not incorporate short protection. A resistor $R_{SP}$, as specified in *Table 6* below, should therefore be connected in series with the output.

**Table 6: Short Protection Resistor**

| Ambient Temp. (°C) | Resistor $R_{SP}$ (ohm) |
|---|---|
| Up to 85 | 51 |
| Up to 125 | 100 |

To minimize the effect on measurement accuracy caused by this resistance in analog mode, the load impedance $R_L$ should be chosen to ensure that $R_L >> R_{SP}$.

# 4. Digital Output ($I^2C$)

## 4.1 General

The NPA-700 series provides digital output data using the industry standard $I^2C$ protocol. Bit rates up to 400 kHz are supported, compatible with the Standard-mode (Sm) and Fast-mode (Fm) standards. Devices are supplied with a slave address of 0x28 as standard.

## 4.2 Block Diagram

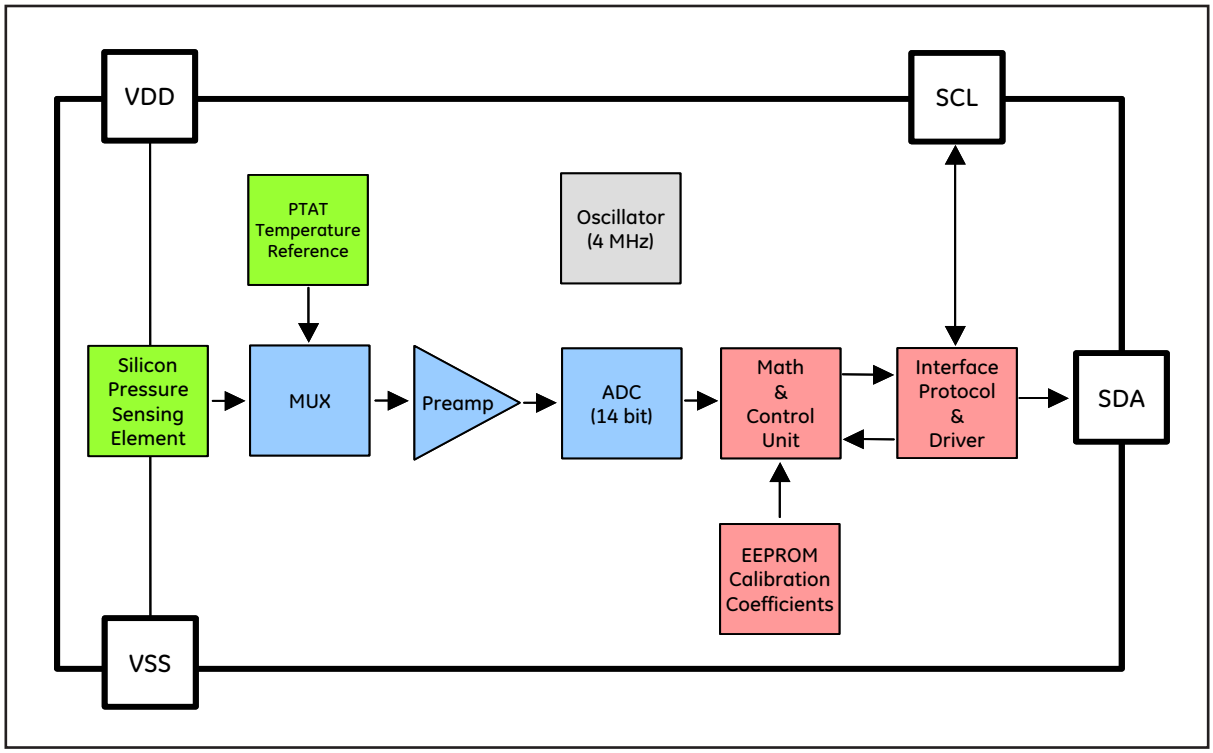



**Figure 7: Block Diagram of NPA-700 Series**

## 4.3 I²C Timing

**Table 7: I²C Timing Parameters**

| Parameter | Symbol | Min | Max | Units |
|---|---|---|---|---|
| SCL clock frequency | $f_{SCL}$ | 100 | 400 | kHz |
| Start condition hold time relative to SCL edge | $t_{HDSTA}$ | 0.1 | | µs |
| Minimum SCL clock low width* | $t_{LOW}$ | 0.6 | | µs |
| Minimum SCL clock high width* | $t_{HIGH}$ | 0.6 | | µs |
| Start condition setup time relative to SCL edge | $t_{SUSTA}$ | 0.1 | | µs |
| Data hold time on SDA relative to SCL edge | $t_{HDDAT}$ | 0 | | µs |
| Data setup time on SDA relative to SCL edge | $t_{SUDAT}$ | 0.1 | | µs |
| Stop condition setup time on SCL | $t_{SUSTO}$ | 0.1 | | µs |
| Bus free time between stop and start condition | $t_{BUS}$ | 2 | | µs |

** Combined low and high widths must equal or exceed minimum SCL period.*

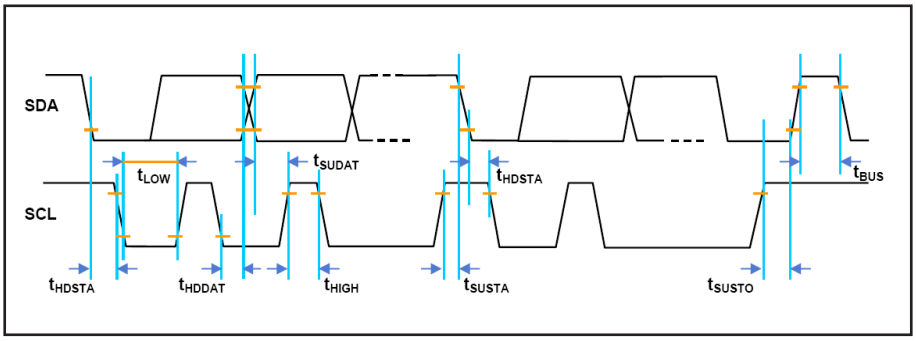



**Figure 8: I²C Timing Diagram**

Two conditions can cause communication errors:

- Sending a start-stop condition without any transitions on the SCL line (no clock pulses in between) creates a communication error for the following communication, even if the next start condition is correct and the clock pulse is applied.
- Restart condition – a falling SDA edge during data transmission when the SCL line is still high will cause the next communication to fail.

In both situations, an additional start condition should be sent to restore communication.

## 4.4 Data Read Operations

To read data from the NPA sensor, the I$^2$C master device sends 8 bits — the 7 bit slave address (0x28 for standard devices) and the 8th bit = 1 to designate a read request. The NPA sensor then sends acknowledge (ACK) to indicate success.

The NPA has three I$^2$C read commands:

Read_DF2: Pressure (2 bytes)
Read_DF3: Pressure (2 bytes) + Temperature (1 byte)
Read_DF4: Pressure (2 bytes) + Temperature (2 bytes)

The number of data bytes returned by the NPA sensor is determined by when the I$^2$C master device sends the NACK and stop conditions as shown in Figure 9 below.

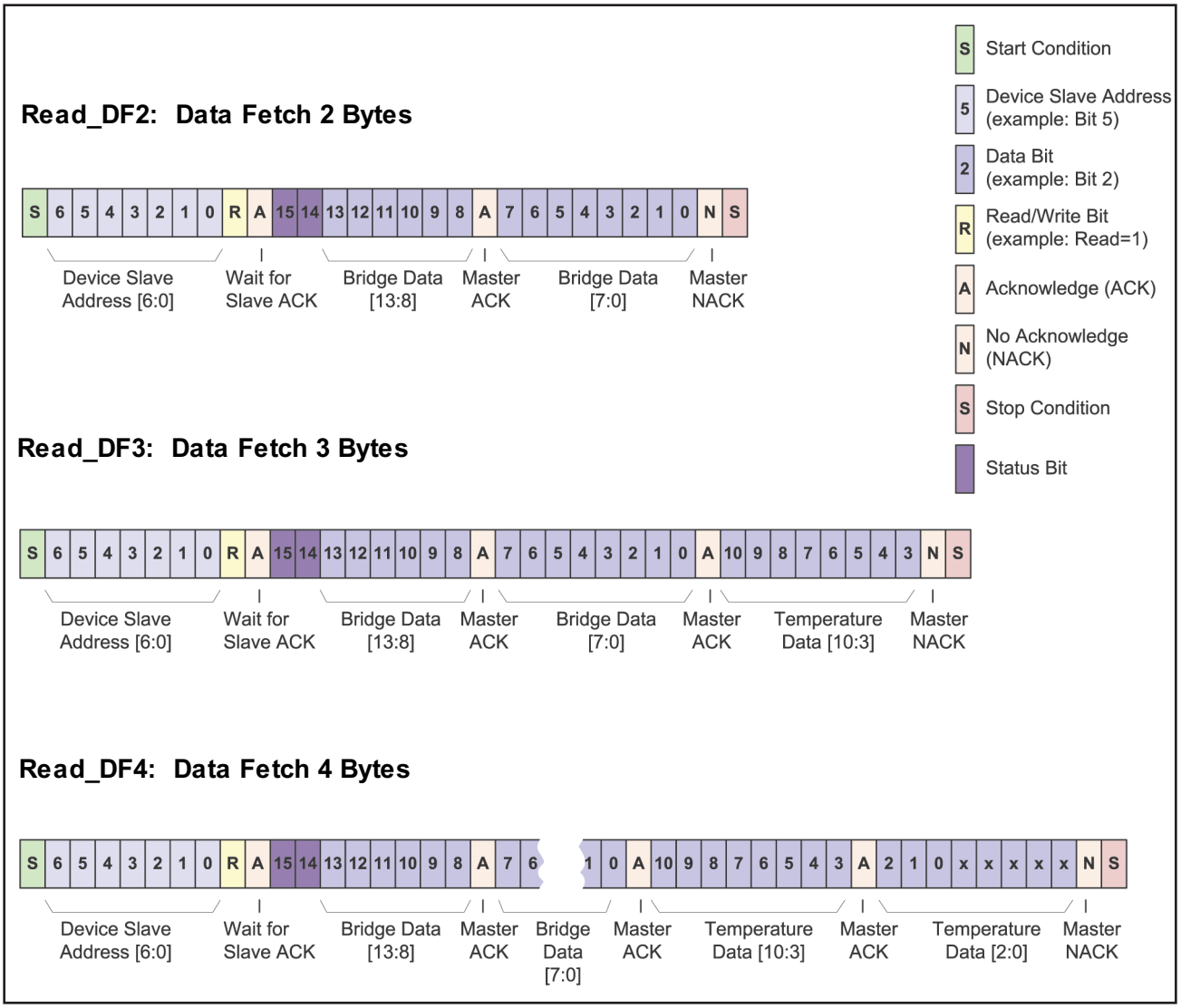



**Figure 9: I$^2$C Data Packet Reads**

## 4.5 Pinout

**Table 8: Pinout of NPA-700 Series**

| Pin # | | Function |
|---|---|---|
| 1 | | NC |
| 2 | | NC |
| 3 | | NC |
| 4 | | NC |
| 5 | | NC |
| 6 | VSS | Ground |
| 7 | VDD | Supply Voltage |
| 8 | SDA | Serial Data |
| 9 | SCL | Serial Clock |
| 10 | | NC |
| 11 | | NC |
| 12 | | NC |

Note:
NC = No connection should be made to these pins

## 4.6 Typical Application Circuit

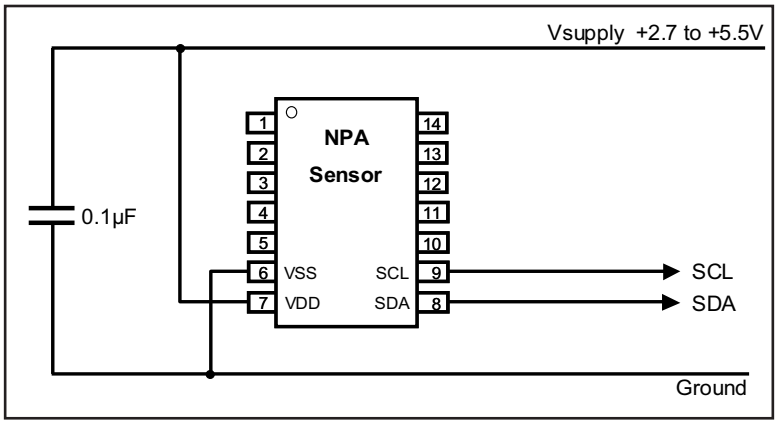



**Figure 10: $I^2C$ Digital Output**

**Note:** *Standard versions are calibrated for 5V supply voltage; 3.3V is available to special order.*

## 4.7 Changing I²C Slave Address

When multiple devices are connected to the I²C bus, each device must be assigned a unique address to ensure correct operation. NPA-700 types are supplied with the I²C slave address set to 0x28 as standard. It is possible to change the address by modifying specific data bits in the sensor EEPROM. The sensor is put into a programming mode and the I²C master device then sends special 4-byte commands.

**IMPORTANT:***1. Be sure that only the specified data bits are changed. Writing data to other locations may cause the sensor to become permanently unusable.*
*2.There must not be any other devices on the data bus sharing the default or proposed address of the NPA sensor. This can be achieved by one of the following methods:*
- *Sensor programmed individually before connection to application circuitry.*
- *Application hardware configured to allow isolation of individual sensor for programming.*

The new address will be effective after the next power-on sequence.

## 4.7 Changing I²C Slave Address (cont.)

The procedure for changing the address is in Table 9 below:

**Table 9: Changing I²C Slave Address**

| | | Data on I²C Bus (hex values) | | | | |
|---|---|---|---|---|---|---|
| | **Action** | **Byte 1** | **Byte 2** | **Byte 3** | **Byte 4** | **Notes** |
| 1 | Put sensor into command mode | [7 bit address *] + [Write bit = 0] | A0 | 00 | 00 | Data must be sent within 6ms of power up |
| 2 | Command to read EEPROM word 02 from sensor | [7 bit address *] + [Write bit = 0] | 02 | 00 | 00 | |
| 3 | Fetch EEPROM word 02 | [7 bit address *] + [Read bit = 1] | 5A (response byte) | Word 02 [bits 15:8] | Word 02 [bits 7:0] | |
| 4 | Modify Word 02 in user software | | | | | Bits [9:3]: I²C address required Bits [12:10]: 011 (communication lock) |
| 5 | Write new version of Word 02 to sensor EEPROM | [7 bit address *] + [Write bit = 0] | 42 | Word 02 [bits 15:8] | Word 02 [bits 7:0] | |
| 6 | Exit command mode & start normal operating mode | [7 bit address *] + [Write bit = 0] | 80 | 00 | 00 | |

*I²C address = 0x28 for standard parts

## 4.8 Diagnostic Features

The NPA-700 incorporates a range of diagnostic features to detect internal faults. The result of the check is sent as part of the digital output data. The first 2 bits of the first pressure byte (labeled 15 & 14 in Figure 9 on page 11) are status bits and are set to 00 if no errors are detected.

| Bit [15] | Bit [14] | Meaning |
|---|---|---|
| 0 | 0 | Normal operation (good data). |
| 0 | 1 | The sensor is in a special mode used for programming. |
| 1 | 0 | The data present at the output has already been read since the last internal measurement cycle i.e. a repeat of previous reading.<br>This will occur if the sensor is polled at a faster rate than the sensor's internal measurement process. It is recommended that polling should be slower than 1.67kHz (0.6ms interval). |
| 1 | 1 | Internal fault exists. |

## 4.9 NPA-700 Sleep Mode

Standard signal-conditioned NPA sensors operate in a continuous measurement mode with current consumption of approximately 3mA.

For applications where reduced power consumption is required, special versions of NPA-700 are available with 'sleep mode' enabled. In 'standby' mode these sensors sit in a powered-down mode, consuming approximately 2µA at room temperature.

When a measurement is required, a special Read_MR command is sent over the $I^2C$ bus as shown in Figure 11 below.

**Read_MR: Measurement Request**

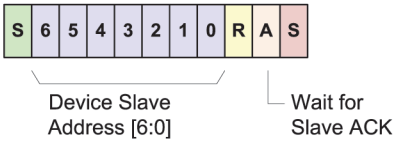



**Figure 11: $I^2C$ Measurement Request Command**

After the sensor responds with the slave ACK, the bus master should create a stop condition.

This command wakes up the sensor and starts an internal measurement cycle. When the measurements are made and the associated calculations are completed, the corrected values of pressure and temperature are written to the output register. The sensor then returns to 'sleep mode'. The values in the output register can then be read using the Data Fetch commands shown in section 4.4.

The same wake up function as the Read_MR command can also be accomplished by using the Read_DF2 or Read_DF3 commands described in section 4.4 and ignoring the "stale" data that will be returned.

# 5. Digital Output (Manchester Code)

## 5.1 General

The NPA sensor is available in two versions with digital output interface:

- Pressure data only: NPA-600 series
- Pressure & temperature data: NPA-601 series

The bit encoding is similar to Manchester in that clocking information is embedded into the signal (falling edges of the signal occur at regular periods). This allows the protocol to be largely insensitive to baud rate differences between the two ICs communicating. In user applications, the NPA sensor will be transmitting information and another IC in the system (typically a microcontroller) will read the data.

## 5.2 Block Diagram

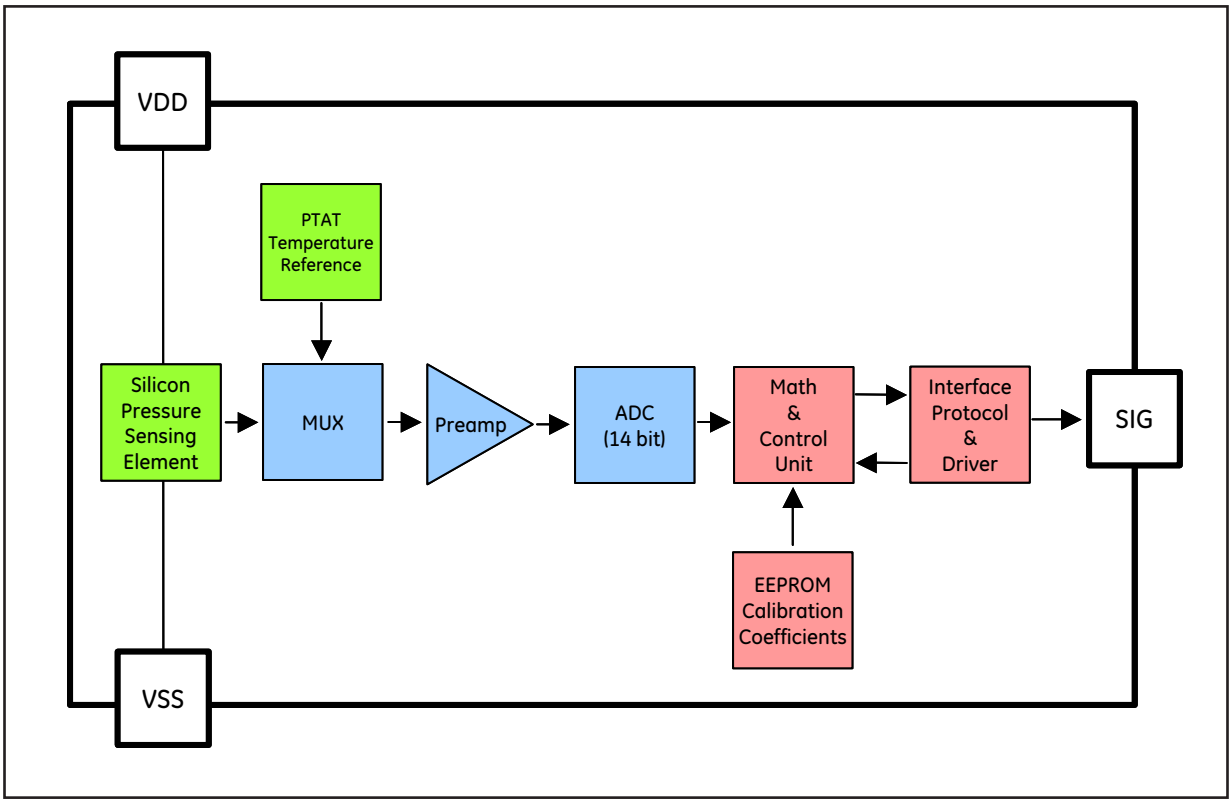



**Figure 12: Block Diagram of NPA-600, NPA-601 Series**

## 5.3 Pinout

**Table 10: Pinout of NPA-600, NPA-601 Series**

| Pin # | | Function |
|---|---|---|
| 1 | | NC |
| 2 | | NC |
| 3 | | NC |
| 4 | | NC |
| 5 | | NC |
| 6 | VSS | Ground |
| 7 | | NC |
| 8 | SIG | Serial Data |
| 9 | VDD | Supply Voltage |
| 10 | | NC |
| 11 | | NC |
| 12 | | NC |

Note:
NC = No connection should be made to these pins

## 5.4 Typical Application Circuits

The circuits in Figures 4,5 & 6 on page 6 can also be used for digital outputs with the following conditions:

- The load resistor and capacitor are not required.
- No pull-down resistor is allowed.
- If a line resistor or pull-up resistor is used, the requirement for rise time ($\leq$5 $\mu$s) must be met.
- Standard versions are calibrated for 5V supply voltage; 3.3V is available to special order.

## 5.5 Bit Encoding

Data bits are transmitted as a Manchester duty-cycle encoded signal:

Start bit 50% duty cycle used to set up strobe time

Logic 1 75% duty cycle

Logic 0 25% duty cycle

Stop Bit High signal for half a bit width

There is a half stop bit time between bytes in a packet.

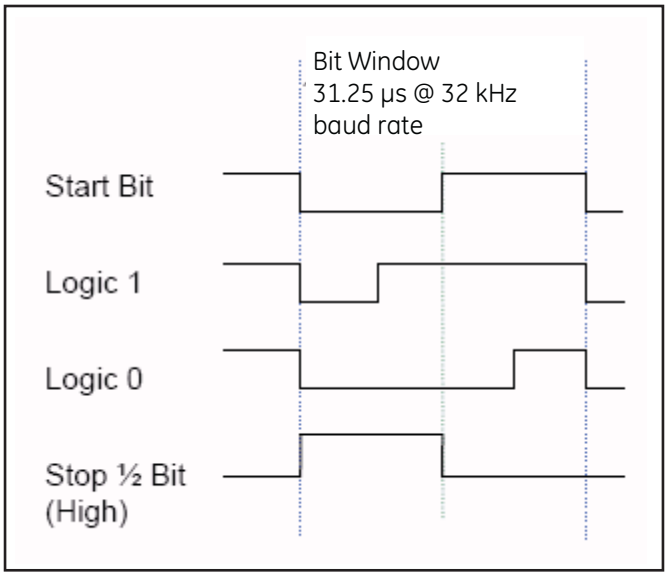



**Figure 13: Manchester Duty Cycle**

An oscilloscope trace of the data transmission demonstrates the bit encoding. Figure 14 on the next page shows a single packet of value 96 hex (= 10010110 bin = 150 dec) being transmitted. Because 96Hex is already even parity, the parity bit P is set to 0.

## 5.5 Bit Encoding (cont.)

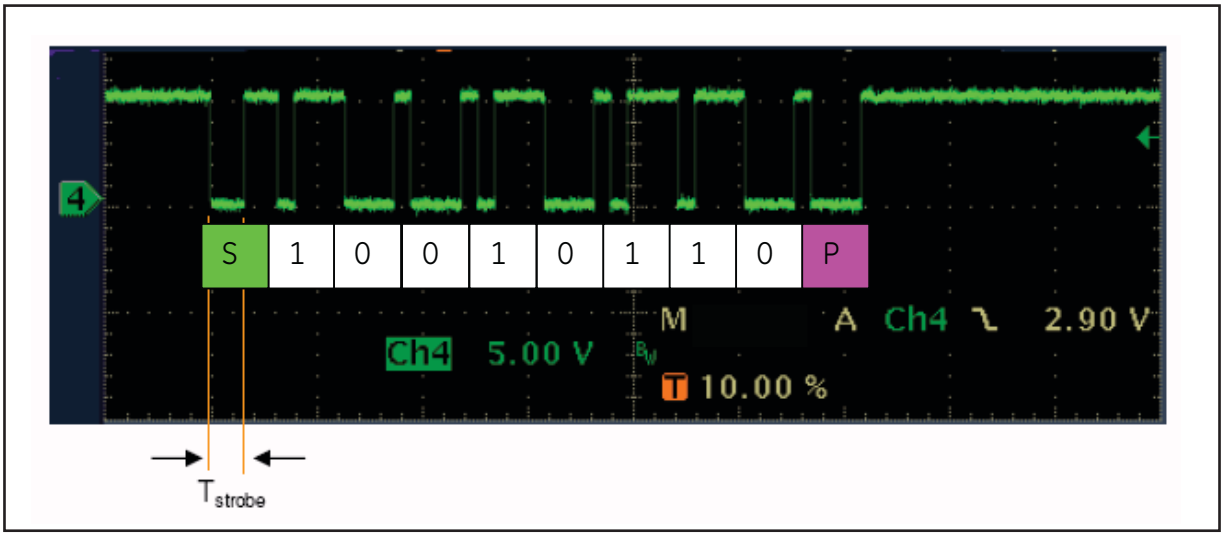



**Figure 14: Oscilloscope Trace of Data Packet Transmission**

## 5.6 Mode 1 - Pressure Only

The sensor first transmits the high byte of *pressure data*, followed by the low byte. The data resolution is 14-bits, so the upper two bits of the high byte are always zero padded. There is a half stop bit time between bytes in a packet. That means that, for the time of half a bit width, the signal level is high. Combining the high and low data bytes provides a 14-bit number corresponding to the pressure reading (see Figure 15 below).

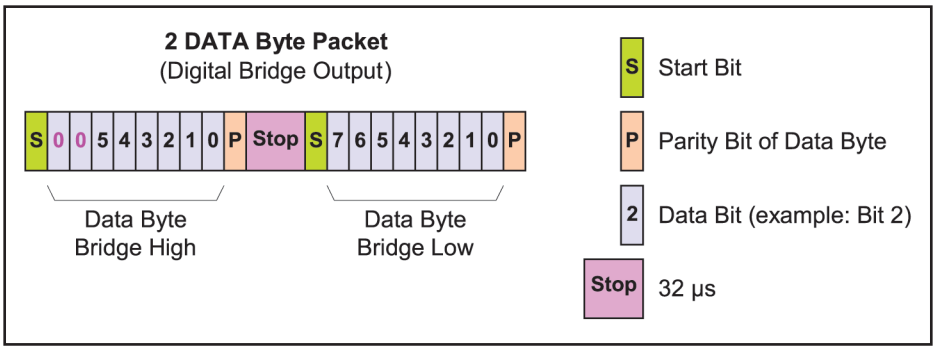



**Figure 15: Digital Output Bridge Readings**

## 5.7 Mode 2 - Pressure With Temperature

The second digital output mode is a *digital pressure reading* with a *temperature reading*, which is transmitted as a 3-data-byte packet (see Figure 16 below). The temperature byte represents an 8-bit value ranging from -50°C to +150°C.

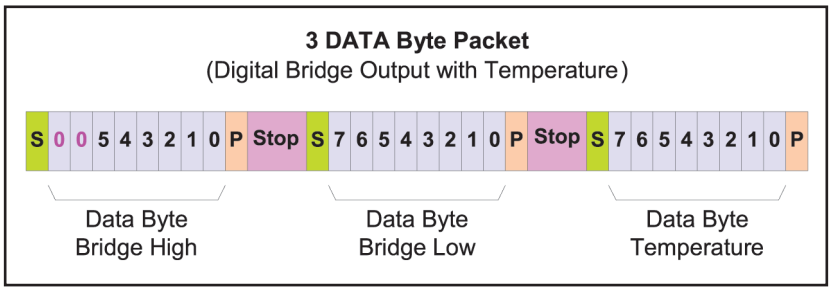



**Figure 16: Digital Output Bridge Readings with Temperature**

## 5.8 Data Transmission

Data is transmitted in packets as shown in Figure 17 below.

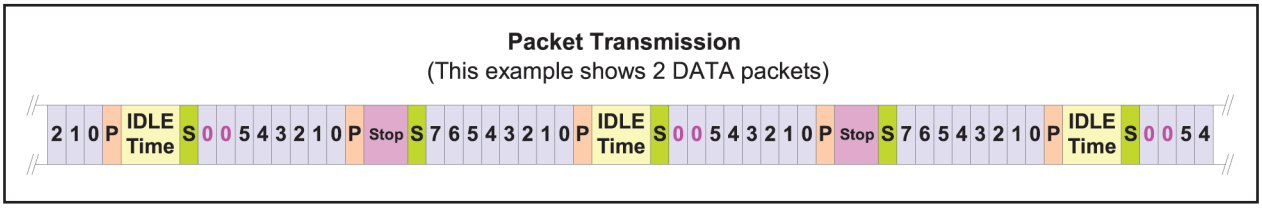



**Figure 17: Transmission of data packets (NPA-600 pressure only mode)**

The total transmission time for both of digital output modes is shown in Table 11 below:

**Table 11: Data Transmission Times**

| Baud Rate | Bit Length | Idle Time* | Transmission Time (Pressure Only) | | Transmission Time (Pressure&Temperature) | |
|---|---|---|---|---|---|---|
| | | | # Bits | Total Time (inc. Idle) | # Bits | Total Time (inc. Idle) |
| 32 kHz | 31.25 µs | 1.0 ms | 20.5 | 1.64 ms | 31.0 | 1.97 ms |
| *The idle time between packets can vary by a nominal ±15% between parts, and over a temperature range of -50 to +150°C. | | | | | | |

## 5.9 How to Read a Data Packet

When the falling edge of the start bit occurs, measure the time until the rising edge of the start bit. This time (Tstrobe) is the strobe time. When the next falling edge occurs, wait for time period equal to Tstrobe, and then sample the signal. The data present on the signal at this time will be the bit being transmitted. Because every bit starts with a falling edge, the sampling window is reset with every bit transmission. This means errors will not accrue for bits downstream from the start bit, as it would with a protocol like RS232. It is recommended that when acquiring the start bit, the sampling rate should be at least 16x the nominal baud rate i.e. 16 x 32 kHz = 512 kHz.

## 5.10 How to Read a Data Packet Using a Microcontroller

It is best to connect the digital signal to a pin on the microcontroller that is capable of causing an interrupt on a falling edge. When the falling edge of the start bit occurs, it causes the microcontroller to branch to its ISR. The ISR enters a counting loop incrementing a memory location (Tstrobe) until it sees a rise on the digital signal. When Tstrobe has been acquired, the ISR can simply wait for the next nine falling edges (8-data, 1-parity). After each falling edge, it will wait for Tstrobe to expire and then sample the next bit.

The digital line is driven by a strong CMOS push/pull driver. The parity bit is intended for error checking when the digital signal is driving long (>2m) interconnects to the microcontroller in a noisy environment. For systems in environments without noise interference, the user can choose to have the microcontroller ignore the parity bit.

## 5.11 8051 C++ Code Example

The following code reads the digital transmission of pressure data (2 bytes). The code also includes functionality to switch the sensor power to reduce consumption in battery-powered applications.

It is assumed that the NPA digital output is connected to the PORT 0 (0x80hex) of the 8051 microcontroller. This code is for a microcontroller running at 24.5 MHz. However, frequencies from 8 to 24.5 MHz can also be used, in which case the number of nop (No Operation) commands in the wait routine should be adjusted accordingly.

```
Hi#define PWR_PIN 0x40
#define SIG_PIN 0x80
#define PORT P2

/*****************************************************************************
* FUNCTION MACROS
*****************************************************************************/

#define NPA_INIT()

{

                SFRPAGE = CONFIG_PAGE;
                PORT_CONFIG |= PWR_PIN;
                PORT &= ~PWR_PIN; /* power */\
                PORT_CONFIG &= ~SIG_PIN;
                PORT |= SIG_PIN; /* signal */

}

#define NPA_ON()    SFRPAGE = CONFIG_PAGE;    PORT |= PWR_PIN;

#define NPA_OFF()    SFRPAGE = CONFIG_PAGE;    PORT &= ~PWR_PIN;

#define NPA_SIGNAL()    (PORT & SIG_PIN)

/*****************************************************************************
* Blocking wait function
* Assuming MCU runs at 24.5MHz, 1 nop = 1/(24.5MHz ÷ 8) µs = ~0.33µs
* Number of nops for 15 µs = 15 ÷ 0.33 = 45
*****************************************************************************/

#define WAIT_15_US()
_nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_() _nop_();
_nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_() _nop_();
_nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_() _nop_();
_nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_() _nop_();
_nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_(); _nop_() _nop_();
```

```
/*******************************************************************************
* Function : getNPAPressure
* Description : reads from the NPA its output value
* Parameters : pointer for return value
* Returns : read value
*******************************************************************************/

UINT16 getNPAPressure (UINT16 *Pressure_value16)
{
                UINT16 Pressure_value1 = 0;
                UINT16 Pressure_value2 = 0;
                UINT8 i;
                UINT16 Pressure;
                UINT8 parity;
                NPA_ON();
                sleep(200); // wait for stabilization
                SFRPAGE = CONFIG_PAGE;
                while (NPA_SIGNAL()); // wait until start bit starts
                // wait, TStrobe
                while (NPA_SIGNAL() == 0x00);
                // first data byte
                // read 8 data bits and 1 parity bit
                for (i = 0; i < 9; i++)
                {
                        while (NPA_SIGNAL()); // wait for falling edge
                        WAIT_15_US();
                if (NPA_SIGNAL())
                Pressure_value1 |= 1 << (8-i); // get the bit
                else
                while (NPA_SIGNAL() == 0x00); // wait until line comes high again

}
// second byte
while (NPA_SIGNAL());
// wait, TStrobe
while (NPA_SIGNAL() == 0x00);
// read 8 data bits and 1 parity bit
for (i = 0; i < 9; i++)
{
                while (NPA_SIGNAL()); // wait for falling edge
                WAIT_15_US();
                if (NPA_SIGNAL())
                Pressure_value2 |= 1 << (8-i); // get the bit
                else
                while (NPA_SIGNAL() == 0x00); // wait until line comes high again
}\
NPA_OFF(); // switch NPA off
```

```
// check parity for byte 1
parity = 0;

for (i = 0; i < 9; i++)
if (Pressure_value1 & (1 << i))
parity++;
if (parity % 2)
return FALSE;
// check parity for byte 2
parity = 0;
for (i = 0; i < 9; i++)
if (Pressure_value2 & (1 << i))
parity++;
if (parity % 2)
return FALSE;
Pressure_value1 >>= 1; // delete parity bit
Pressure_value2 >>= 1; // delete parity bit
Pressure = (Pressure_value1 << 8) | Pressure_value2;
*Pressure_value16 = Pressure;
return TRUE; // parity is OK
}

/*****************************************************************************
* Function : cmdGetNPAValue
* Description : converts digital pressure value to pressure in real units
* Parameters : none
* Returns : none
* Notes : none
*****************************************************************************/
void cmdGetNPAValue (void)

{
                UINT16 Pressure_value;
                float Pressure_float;
                float pressure_min = -1.25; // pressure corresponding to output value 0 dec
                float pressure_max = 11.25; // pressure corresponding to output value 16383 dec
                // values dependant on specific sensor rating
                // values shown are for sensor with full range of -1.25 to +11.25 inH2O
                printf("cmdGetNPAValue\n");
                NPA_INIT(); // init the I/O pins used for the NPA
                NPA_OFF(); // switch the NPA off until use
                if (getNPAPressure(&Pressure_value))
                {
                        Pressure_float = ((float)Pressure_value * (pressure_max - pressure_min) / 16383
                + pressure_min; // conversion to real pressure units
                        SFRPAGE_UART();
                        printf("Pressure %u, %2.1f\n", Pressure_value, Pressure_float);
                }
}
```

## 5.12 PIC1 Assembly Code Example

The following code reads the digital transmission of pressure data (2 bytes). It is assumed that the NPA digital output is connected to the interrupt pin (PORTB,0) of the PIC and that the interrupt is configured for falling edge interruption. This code would work for a PIC running between 3 and 20 MHz.

```
Pressure_high EQU 0x24          ;; memory location reserved for pressure high byte
Pressure_low EQU 0x25           ;; memory location reserved for pressure low byte
                                ;; this byte must be consecutive from Pressure_high
LAST_LOC EQU 0x26               ;; this byte must be consecutive from Pressure_low
Tstrobe EQU 0x26                ;; location to store start bit strobe time.
ORG 0x004                       ;; ISR location

;;;
```

```
                bcf STATUS,C            ;; clear the carry
                btfsc PORTB,0           ;; sample the NPA signal
                bsf STATUS,C            ;; if signal was high then set the carry
                rlf INDF,1              ;; rotate carry=NPA into LSB of register
                                        ;; that FSR currently points to
                clrf time_out           ;; clear the edge timeout counter

WAIT_RISE:      btfsc PORTB,0           ;; if rise has occurred then done
                goto NEXT_BIT
                incfsz time_out,1       ;; increment the edge time out counter
                goto WAIT_RISE
                goto RTI                ;; edge time out occurred.

NEXT_BIT:       incf bit_cnt,1          ;; increment bit counter
                movlw 0x08              ;; there are 8 bits of data
                subwf bit_cnt,0         ;; test if bit counter at limit
                btfss STATUS,Z          ;; if not zero then get next bit
                goto BIT_LOOP
                clrf time_out           ;; clear the edge time out counter

WAIT_PF:        btfss PORTB,0           ;; wait for fall of parity
                goto P_RISE
                incfsz time_out,1       ;; increment time_out counter
                goto WAIT_PF
                goto RTI                ;; edge timeout occurred

P_RISE:         clrf time_out           ;; clear the edge time out counter

WAIT_PR:        btfsc PORTB,0           ;; wait for rise of parity
                goto NEXT_BYTE
                incfsz time_out,1       ;; increment edge time out counter
                goto WAIT_PR
                goto RTI                ;; Edge time out occurred

NEXT_BYTE:      incf FSR,1              ;; increment the INDF pointer
                movlw LAST_LOC
                subwf FSR,0             ;; compare FSR to LAST_LOC
                btfss STATUS,Z          ;; if equal then done
                goto WAIT_Plow

;;;
```

```
WAIT_PLF:   btfss PORTB,0                   ; wait for fall of PORTB,0 indicating
            goto GET_Plow                   ; start of pressure low byte
            incfsz time_out
            goto WAIT_PLF
            goto RTI                        ; edge timeout occurred

RTI:        ;;;
```

# 6. Uncalibrated Versions

## 6.1 General

The NPA-100 series provides the raw bridge output from the MEMS pressure sensing die (uncalibrated and uncompensated) so that the user can add custom signal conditioning circuitry as required.

## 6.2 Block Diagram

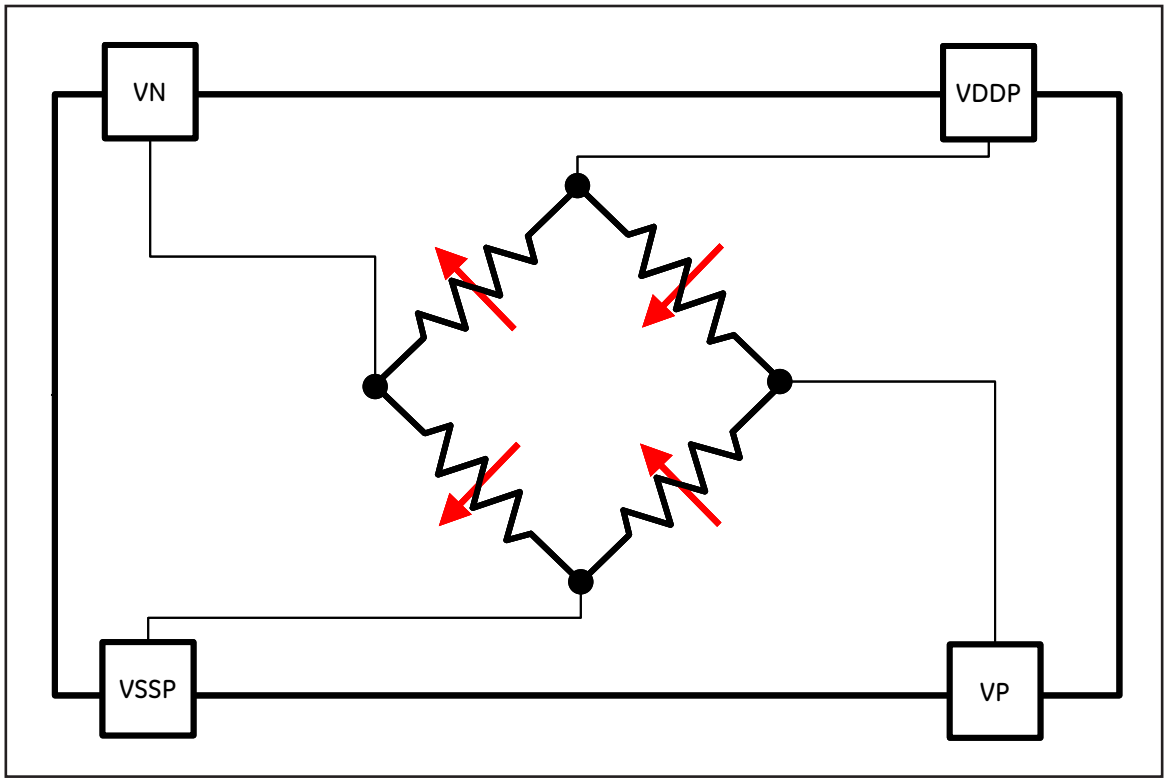



**Figure 18: Block Diagram of NPA-100 Series**

## 6.3 Pinout

**Table 12: Pinout of NPA-100 Series**

| Pin # | | Function |
|---|---|---|
| 1 | | NC |
| 2 | | NC |
| 3 | VN | Negative Output |
| 4 | VSSP | Negative excitation |
| 5 | | NC |
| 6 | | NC |
| 7 | | NC |
| 8 | | NC |
| 9 | | NC |
| 10 | | NC |
| 11 | VP | Positive Output |
| 12 | VDDP | Positive excitation |

Note: NC = No Connection should be made to these pins.

# 7. Package Dimensions

## 7.1 Manifold Fitting

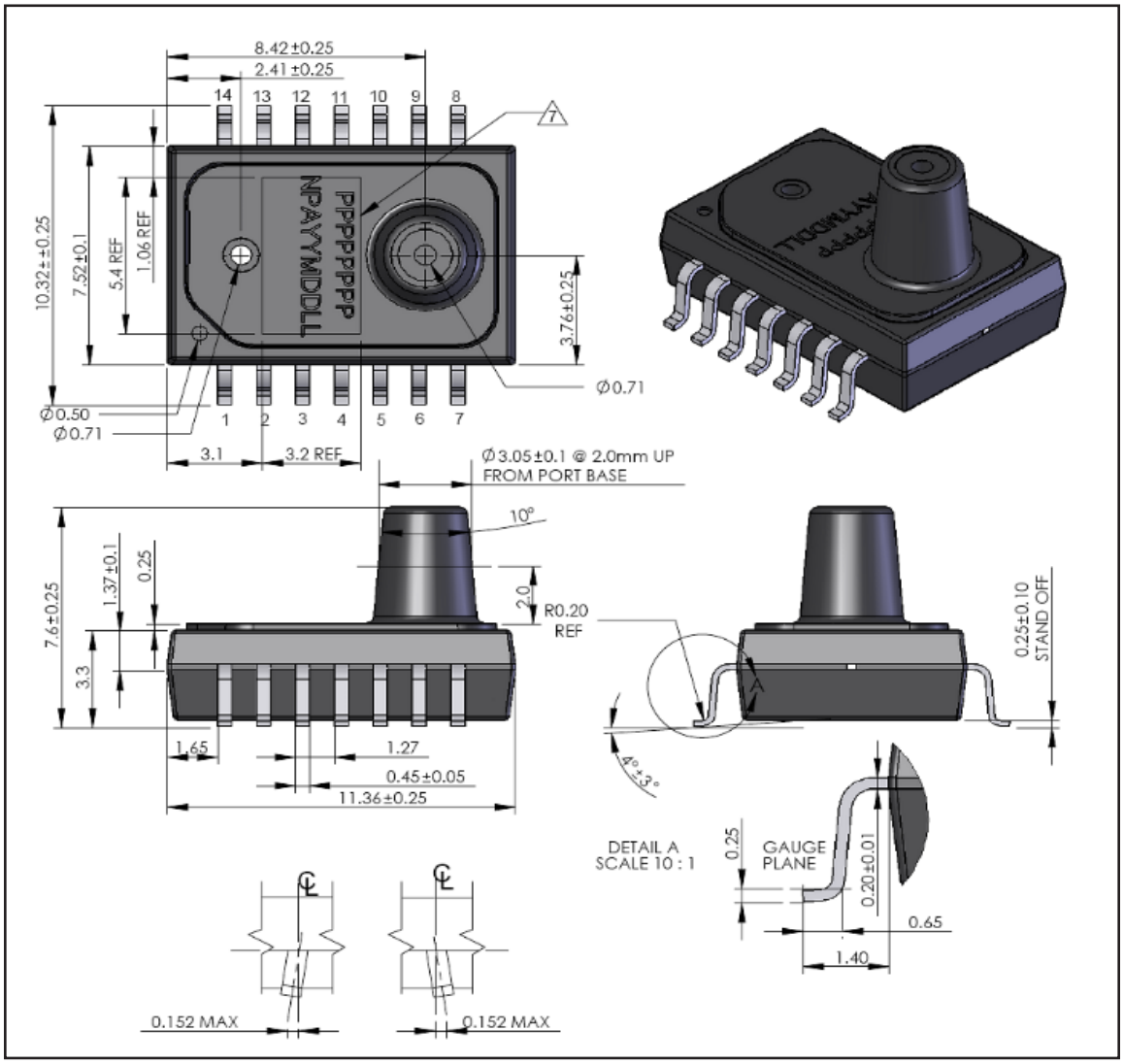



**Figure 19: Package Dimensions (mm) - Manifold Fitting**

## 7.2 Barbed Fitting

Versions with barbed fittings are available for use with flexible tubing. Recommended tube size is 3/32" I.D. x 7/32" O.D. (1/16" wall thickness).

Barbed fittings can also be used for manifold connections with appropriate sealing features.

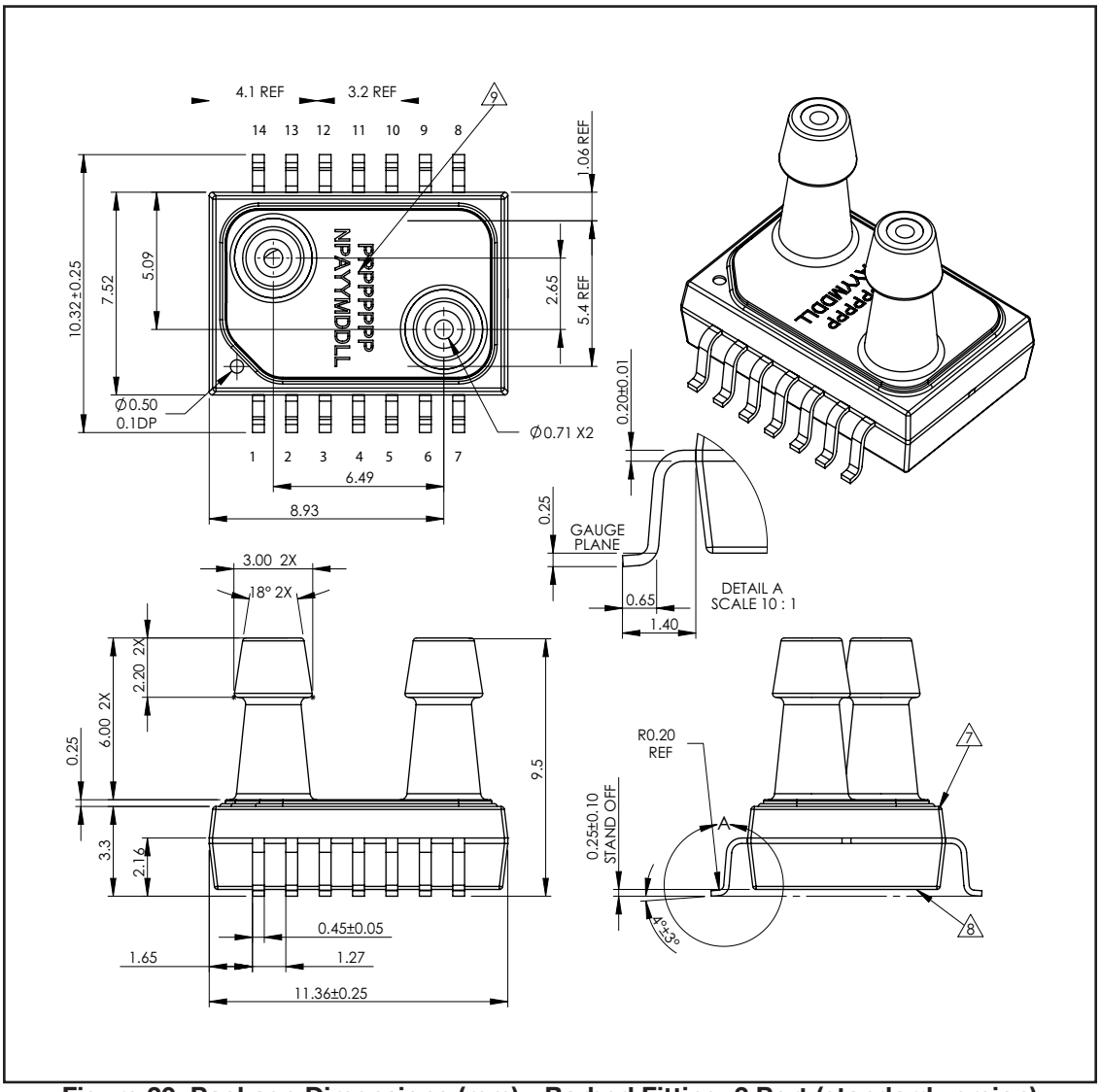



**Figure 20: Package Dimensions (mm) - Barbed Fitting, 2 Port (standard version)**

## 7.2 Barbed Fitting (cont.)

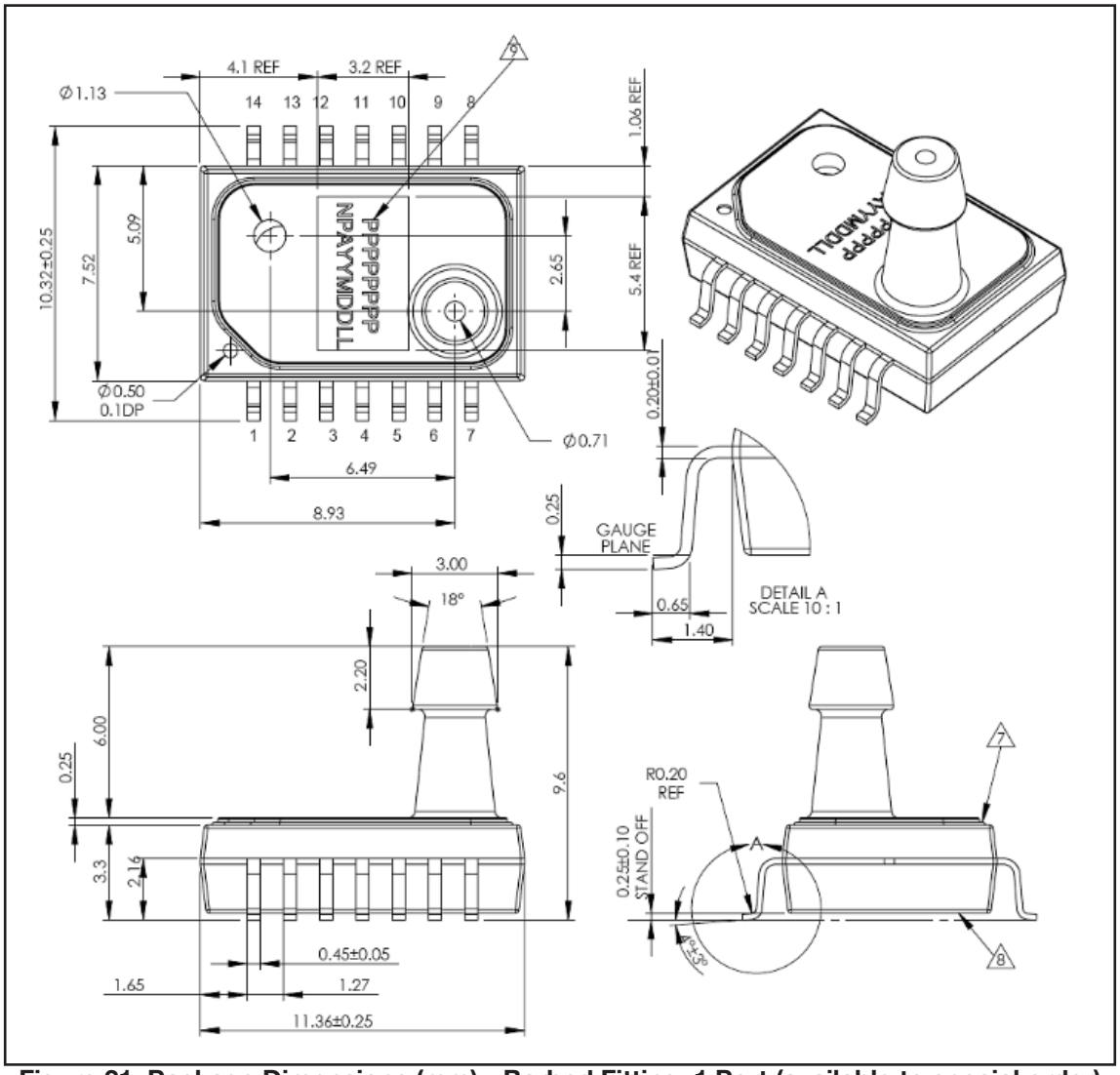



**Figure 21: Package Dimensions (mm) - Barbed Fitting, 1 Port (available to special order)**

## 7.3 No Port

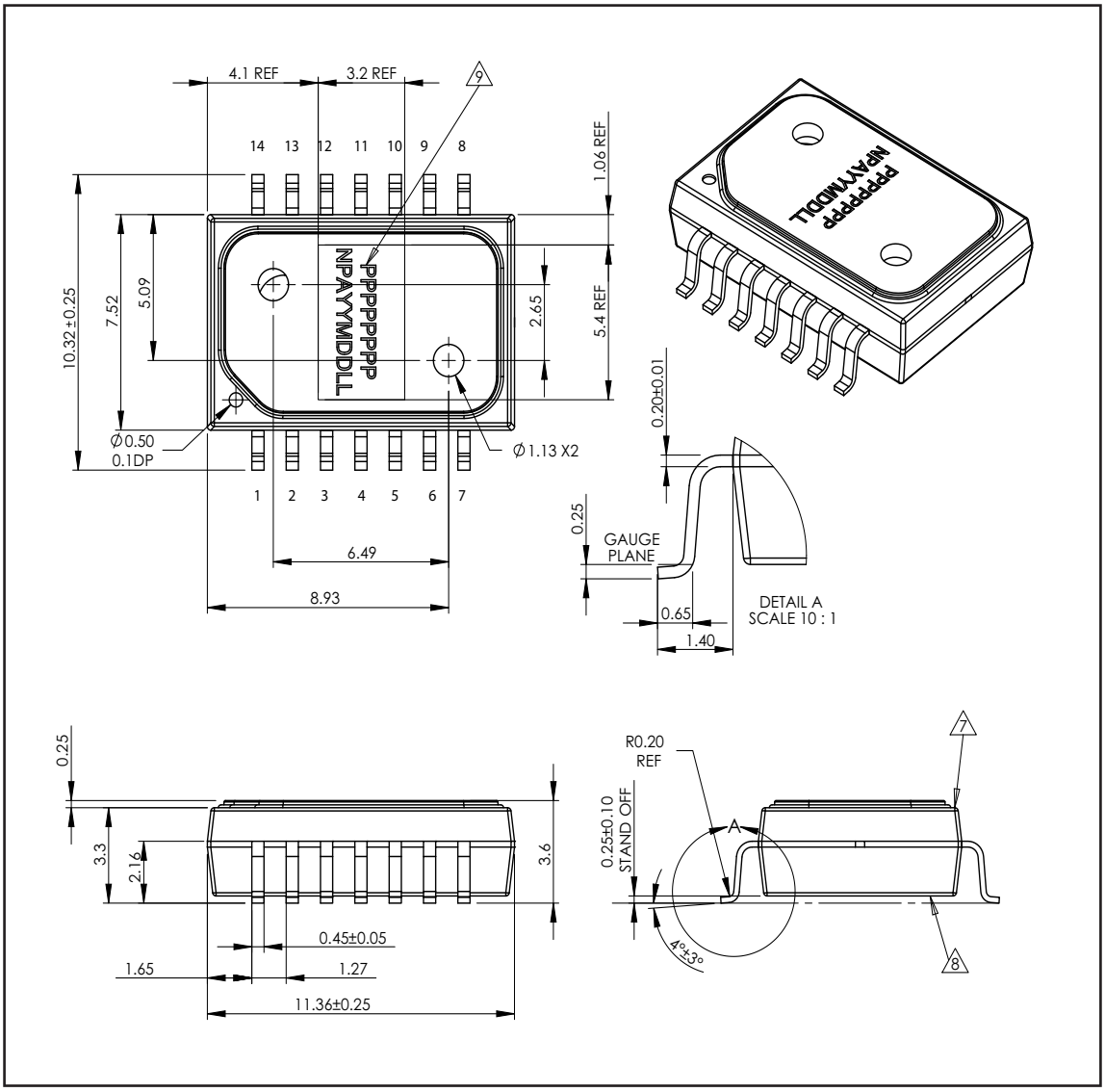



**Figure 22: Package Dimensions (mm) - No Port**

### 7.4 Dead Volume

For some applications it is important to know the volume of gas held within the interior volume of the sensor package (‘dead volume’). Table 13 gives approximate values for the standard combinations of port style and pressure rating.

**Table 13: Estimated Dead Volumes**

| | Dead Volume ($mm^{3)}$ | | |
|---|---|---|---|
| **Pressure Rating** | **Dual Barb** | **1 Manifold** | **No Ports** |
| 10 “H2O, 1 psi | 22.0 | 22.0 | 17.8 |
| 5 psi | 22.6 | 22.6 | 18.5 |
| 15 psi | 22.6 | 22.5 | 18.4 |
| 30 psi | 22.5 | 22.5 | 18.4 |

## 8. Pressure Port Connections

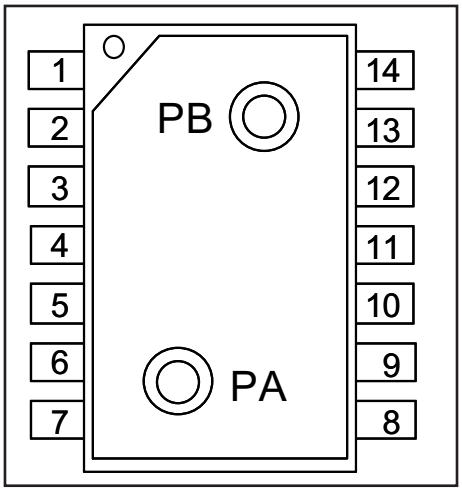



**Figure 23: Pressure Port Locations**

**Table 14: Pressure Port Connections**

| Pressure Type | PA (primary) | PB (secondary) |
|---|---|---|
| Absolute | Pressure | No connection |
| Gauge | Pressure | Vent to atmosphere |
| Differential | + Pressure | - Pressure |

# 9. Suggested PCB Pad Layout

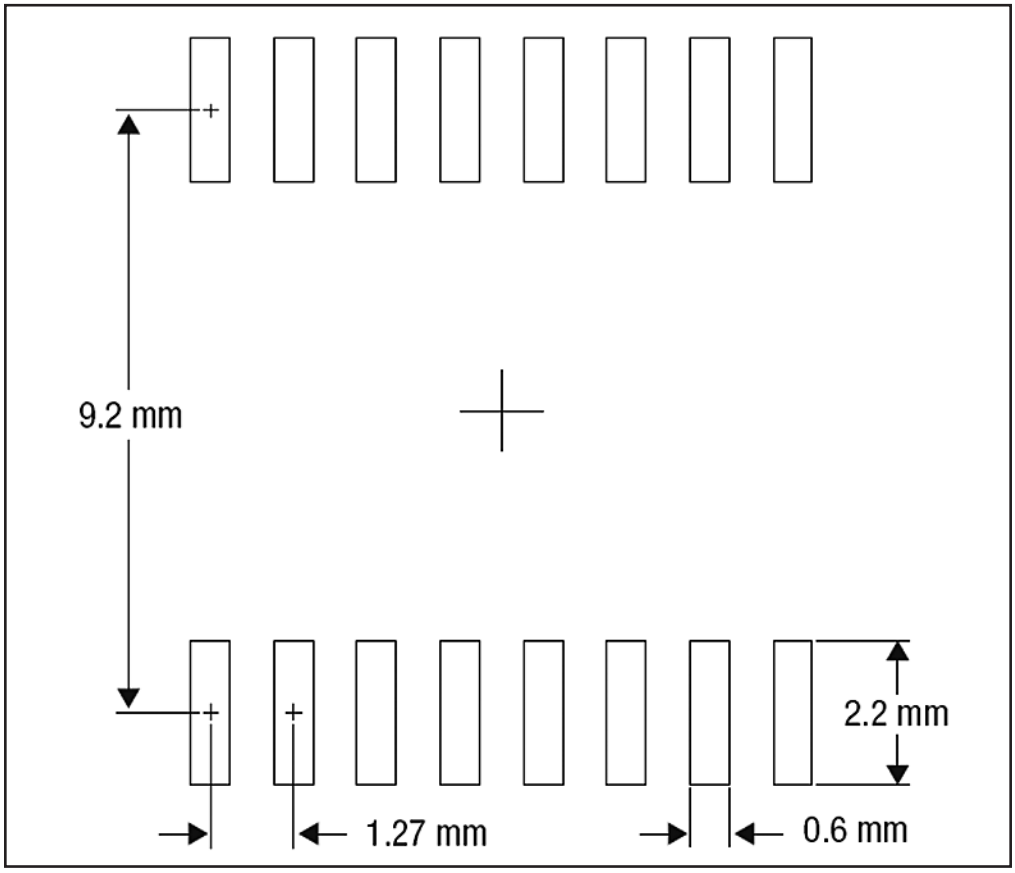



**Figure 24: Suggested PCB Pad Layout (SOIC14 Wide Package)**

# 10. Soldering

## 10.1 Standard Reflow Soldering

The NPA sensor can be soldered using standard reflow ovens (including lead-free soldering) with the following conditions:

- *Maximum temperature:* 250°C for 30 seconds
- *Solder paste:* Use “No-Clean” solder paste only
- *PCB cleaning:* Do not clean or wash circuit boards after soldering

The NPA is classified as *moisture sensitivity level (MSL) 6*, as defined in *Jedec standard J-STD-20*. Product is supplied on carrier tape/reels sealed in moisture-proof bags. Bags are labelled with guidelines on thermal conditioning prior to reflow soldering. Users should follow these instructions in conjunction with *Jedec specification J-STD-033*. Under normal factory conditions parts must be mounted within 8 hours of opening the bag. If parts are not mounted within this time, the taped parts should be removed from the reel and baked at 100°C for 24 hours.

### 10.2 Manual Soldering

- *Contact time:* Limited to 5 seconds at up to 350°C

## 11. Device Branding

Production units are branded with a 2 line code giving product and date information:

| | |
|---|---|
| **PPPPPPPP** | Product code |
| **NPAYYMDDLL** | Manufacturing date |

| | |
|---|---|
| **YY**: | Year |
| **M:** | Month |
| **DD:** | Day |
| **LL:** | Lot # |

# Customer Support Centers


**U.S.A.**
**Sales and Services**
**(Repair/Calibration):**
Amphenol Thermometrics, Inc.
St Marys Center
967 Windfall Road
St Marys, Pennsylvania 15857
U.S.A.
T: 814-834-9140
F: 814-781-7969

**U.S.A.**
**Technical Support:**
Amphenol Thermometrics, Inc.
St Marys Center
967 Windfall Road
St Marys, Pennsylvania 15857
U.S.A.
T: 814-834-9140
F: 814-781-7969

Amphenol Advanced Sensors
1055 Mission Court
Fremont, CA 94359-8203

**Europe, Asia and Middle East**
**Sales and Service:**
Amphenol Thermometrics (UK) Ltd.
Crown Industrial Estate
Priorswood Road
Taunton
Somerset
TA2 8QY
T: +44 (0) 1823 335200
F: +44 (0) 1823 332637

**Brazil**
**Sales and Service**
Amphenol Headquarters - Brazil
Rua Diogo Moreira, 132-20º Andar
Sao Paulo - SP
Brazil
T: +55 11 3815 1003

**China:**
Amphenol (Changzhou)
Connector Systems
305 Room, 5D
Jintong Industrial Park
Wujin, Changzhou, Jiangsu, China
T:+86 519 8831 8080 ext. 50087
F:+86 519 8831 2601


# Amphenol

## Advanced Sensors

**www.amphenol-sensors.com**


©2014 Amphenol Thermometrics, Inc. All rights reserved.
Technical content subject to change without notice.



AAS-910-289 Rev. E July 2014