# GP-2710 Intelligent Pressure Sensor

## Features

- ✓ Built-in MCU for intelligent processing
- ✓ Flexible configuration
- ✓ Programmable via Modbus over RS485
- ✓ Switchable interfaces: RS485, I2C, and SPI
- ✓ High accuracy up to ±0.03% FS (TEB)
- ✓ Integrated high-precision temperature sensor (±0.1°C)
- ✓ Supports low-power operation for embedded systems

## Introduction

The GP-2710 Intelligent Pressure Sensor represents a new generation of pressure sensing technology, combining a traditional OEM sensor form factor with advanced digital processing capabilities. Unlike conventional digital pressure sensors based on ASIC architecture, the GP-2710 integrates a powerful onboard MCU, enabling enhanced flexibility, configurability, and performance.

Through Modbus communication, users can easily configure sensor parameters, switch communication interfaces, and adapt the sensor to different system requirements. The device supports RS485, I2C, and SPI interfaces, allowing seamless integration into industrial networks as well as embedded platforms.

With proprietary compensation algorithms, the GP-2710 performs real-time temperature compensation and nonlinearity correction, achieving accuracy levels comparable to high-end pressure transmitters. At the same time, the integrated temperature sensor enables precise ambient monitoring.

Designed for modern applications, the GP-2710 supports low-voltage operation down to 3.2V and includes low-power modes, making it suitable for battery-powered and embedded systems. Its ability to combine sensing, processing, and communication in a single compact unit makes it an ideal solution for smart instrumentation, industrial automation, and next-generation IoT systems.

## Specification

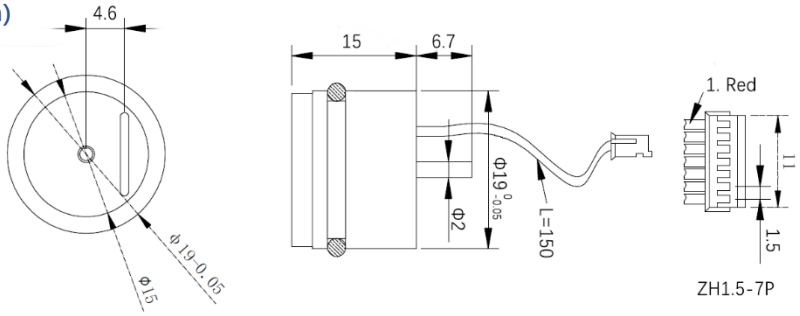
| Items | Parameters |
|---|---|
| Range | -100kPa···0 ~ 20kPa···70MPa |
| Pressure Type | Gauge, Absolute, Sealed gauge |
| Overload | x2 FS or 110MPa (Take minimum value) |
| TEB[1] (Total Error Band) | ±0.1%FS, ±0.05%FS, ±0.03%FS |
| Compensation Temp. | -10℃ ~ 70℃ |
| Operating temp. | -30℃ ~ 80℃ |

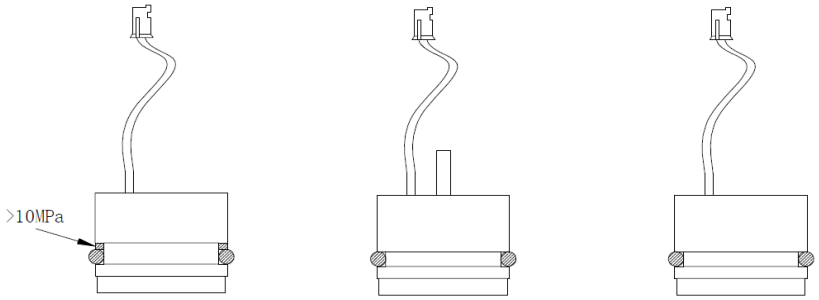
| | |
|---|---|
| Storage Temp.[2] | -40℃ ~ 105℃ |
| Power Supply | 1.8VDC~ 36VDC |
| Communication | RS485 (Modbus RTU, 1200, 2400, 4800, 9600(Default), 19200, 38400 |
| Output Mode | RS485 Slave & I²C Slave & SPI Slave / SPI Streaming |
| Working Current | Typ. 1.7mA standard power mode<br>Typ. 6.5uA Standby low power mode |
| Temp. Accuracy[3] | < ±0.1℃ ( 0℃ ~ 50℃) , < ±0.2℃ (-10℃ ~ 60℃), < ±0.5℃ (-25℃ ~ 75℃) |
| Response Time[4] | 50ms, standard power mode<br><300ms, low power mode |
| Sample Frequency[4] | 20Hz (Default), can be set at 5Hz, 50Hz, 100Hz |
| Insulation resistance[5] | 100MΩ, 100VDC |
| Vibration | No change, 10gRMS, (20~2000)Hz |
| Impact | 100g, 11ms |
| Position Effect | deviate 90° from any orientation, zero change ≤0.05%FS |
| Lifetime | ≥10 million cycles |
| Long-term stability | ±0.1% FS/year |
| Media compatibility | The gas or liquid which is compatible with stainless steel and FKM |
| Protection | IP65, Circuit board black epoxy potting. (Not including connector) |
| Materials | Housing & Diaphragm: 316L Stainless steel<br>O-ring: Viton<br>Filled oil: Silicon oil |
| Cable | 2651 26AWC Flat cable, 105℃, 155mm |
| Connector | ZH1.5-7P connector |
| Net weight | ≈23g |

Note:

1. Including calibration error and temperature drift effects.
2. It can be up to 125℃ without flat cable.
3. Onboard ambient/reference temperature.
4. Increasing the sampling rate may lose some precision and affect response time.
5. Insulation resistance was tested under RH≤65% and Temp.≤35 ℃

## Dimensions (mm)






Sealed Gauge　　Gauge　　Absolute

## Electrical connection

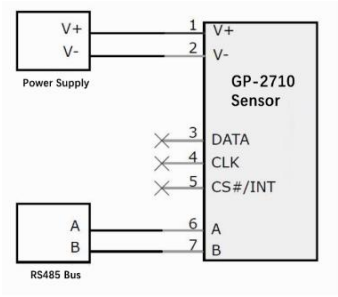
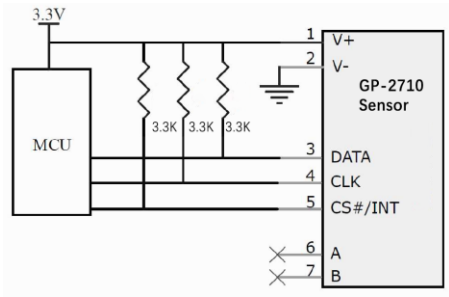
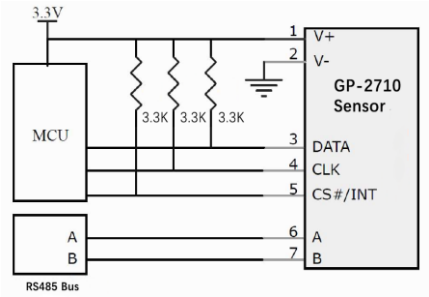
### Working Mode

| Mode | Standard | | | | | Low Power | | | |
|---|---|---|---|---|---|---|---|---|---|
| Code | 0x00 | 0x01 | 0x02 | 0x03 | 0x04 | 0x10 | 0x11 | 0x12 | 0x13 |
| Interface 1 | RS485 | | | | | Low Power RS485 | | | |
| Interface 2 | / | I2C Slave | SPI Slave | SPI Streaming | Reserve | / | I2C Slave | SPI Slave | SPI Streaming |

### Wires Definition

| No. | Definition | Description |
|---|---|---|
| 1 (Red) | V+ | VCC |
| 2 | V- | GND |
| 3 | DATA | · In 0x00 and 0x10 modes, this pin may be left floating. In other modes, an external pull-up resistor must be used to pull the pin up to 3.3V; a resistance of 10kΩ is recommended.<br>· In $I^2C$ slave mode, this pin functions as the SDA pin.<br>· In SPI slave mode, after the data-ready signal has been output as a low-level pulse, it functions as the MISO pin.<br>· In SPI Streaming mode, it functions as the MOSI pin. |
| 4 | CLK | · In 0x00 and 0x10 modes, this pin may be left floating. In other modes, an external pull-up resistor must be used to pull the pin up to 3.3V; a resistance of 10kΩ is recommended.<br>· In $I^2C$ slave mode, this pin functions as the SCL pin.<br>· In SPI mode, this pin functions as the SCLK pin. |
| 5 | CS/INT | · In 0x00 and 0x10 modes, this pin may be left floating. In other modes, an external pull-up resistor to 3.3V is required; a resistance of 10 kΩ is recommended.<br>· $I^2C$ interrupt pin. In $I^2C$ mode, a low-level pulse is generated when data is ready.<br>· SPI chips select pin. |
| 6 | RS485-A | Leave it floating if not in use, it is recommended to reserve a connection point. |
| 7 | RS485-B | |

## Application 1: Sensor imbedded in device

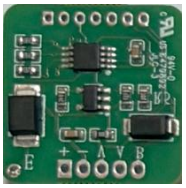
| RS485 | I²C/SPI | I²C/SPI+RS485 |
|---|---|---|
|  |  |  |



1: Simultaneously acquiring pressure values via I²C/SPI and RS485 may affect response speed; however, it is still recommended to retain the RS485 connection points to allow for full-function calibration of the digital sensor in the future.

2: If the digital sensor is used as a standalone product in an RS485 application, it is necessary to include EMC protection components in the circuit to enhance resistance to electromagnetic interference such as static electricity, burst noise, and surges.

## Application 2: Sensor to transmitter (Output Interface Board)

| Type | Optional PCB | Function | Note |
|---|---|---|---|
| **GP-2710-P**<br><br>RS485Digital Transmitter |  | Output: RS485 Modbus<br>Power supply: 3.3~28VDC<br>Size: 20x20mm<br>EMC | / |
| **GP-2710-Io**<br><br>4~20mA Transmitter |  | Output: two wires 4~20mA<br>Power supply: 8~28VDC<br>Calibration: temp. by software<br>Size: 20x20mm<br>EMC | The sensor must be set to Hp_Iout mode |
| **GP-2710-Vo**<br>0.5~2.5/4.5V<br>0~5/10V<br>1~5/10V<br>Transmitter |  | Zero Output: 0~2VDC<br>Full Output: 2.5~10VDC<br>Calibration: temp. by software<br>Size: 20x20mm<br>EMC | The sensor must be set to Hp_Vout mode; Voltage switching requires the removal or replacement of some onboard resistors; There will be an output of a few mV at zero. |

## Configuration Software


Dedicated configuration software that reads sensor data, configures basic parameters, adjusts measurement ranges, switches units, and performs many other functions via the RS485 interface. There's a hide function for current and voltage calibration when sensor connects to transfer PCB.

1. Operating Instructions: Connect the USB-to-RS485 module to your computer, connect the product to the RS485 module, and power on the product.
Open the software, select the corresponding serial port and baud rate (the default for this product is 9600), click the "Open" button, then click the "Search" button until the number of devices online is 1. Click the "Stop" button. The software will automatically read the digital sensor parameters.
2. In the "Parameter Modification" section, you can change the address and baud rate. The address can be set between 1 and 240, and the baud rate can be set to 1200, 2400, 4800, 9600, 19200, or 38400. After entering the address or selecting the baud rate, press the "Modify" button for the changes to take effect immediately. Rescan for the product to confirm the information is correct and click "Save Changes" to ensure the modifications are retained after power is lost.
3. You can modify the operating mode. After selecting the desired mode, click the "Modify" button, then click "Save Changes," and wait for the sensor to automatically save the settings before they take effect (approximately 3–5 seconds).
4. The sensor has 36 built-in pressure units that can be switched via the software.
5. Ambient temperature refers to the temperature data output by the high-precision temperature sensor.
6. The offset setting allows you to shift the sensor's output; the output value = measured value – offset.
7. When executing commands, the software performs a built-in CRC check, eliminating the need for the user to calculate or enter it.

## Order Guide

| GP-2710 | Programmable Pressure Sensor | | | | | |
|---|---|---|---|---|---|---|
| | **Code1** | Pressure Range | | | | |
| | 0B | 0~20kPa | G | 12 | 0~2MPa | G/A |
| | 0A | 0~35kPa | G | 13 | 0~3.5MPa | G/A |
| | 02 | 0~70kPa | G/A | 14 | 0~7MPa | A/S |
| | 03 | 0~100kPa | G/A | 15 | 0~10MPa | A/S |
| | 07 | 0~200kPa | G/A | 17 | 0~20MPa | A/S |
| | 08 | 0~350kPa | G/A | 18 | 0~35MPa | A/S |
| | 09 | 0~700kPa | G/A | 19 | 0~70MPa | A/S |
| | 10 | 0~1MPa | G/A | | | |
| | [X-Y]kPa/MPa | | Only using for customized range | | | |
| | | **Code2** | Pressure Type | | | |
| | | G | Gauge | | | |
| | | A | Absolute | | | |
| | | S | Sealed Gauge | | | |
| | | | **Code3** | Accuracy | | |
| | | | S | ±0.03%FS ( >100kPa G or ⩾200kPa A) | | |
| | | | A | ±0.05%FS | | |
| | | | B | ±0.1%FS | | |
| | | | | **Code4** | Cable Length | |
| | | | | L150 | 150mm (Default) | |
| | | | | LXXX | Customized Length | |
| | | | | | | |
| **GP-2710** | 03 | G | B | L150 | | |

*Completed Sample:* ***GP-2710-03-G-B-L150***

## Optional Output Interface Board

An optional output interface board is available, providing current, voltage, or RS485 outputs with full EMC protection. There're 3 options for choice.

- ☐ GP-2710-P RS485 Modbus
- ☐ GP-2710-Io 4~20mA
- ☐ GP-2710-Vo 0~5VDC (Default)

**Tips:**

The sensor diaphragm is extremely fragile. Any direct contact with hard objects may cause diaphragm deformation and damage to the sensor.

# Communication reference

## 1 RS485 Guide

### 1.1 Overview

The RS485 interface can be used with the digital sensor in any mode. When using only the RS485 interface and the CLK, DATA, and CS/INT pins do not have pull-up resistors, ensure that the digital sensor is in 0x00 or 0x10 mode.
Communication uses the standard MODBUS RTU protocol; this device acts as a slave and supports networking with up to 240 units.
Character format: 1 start bit, 8 data bits, 1 stop bit, no parity bit.
Supported commands: Read (03), Write Single Register (06), Write Multiple Registers (10).
CRC check: 16-bit CRC check, with the low byte first and the high byte last.
Address 00 is the universal address.

### 1.2 Examples of Commands

Pressure Read Command: Addr + 0x03 + 0x00 + 0x02 + 0x00 + 0x02 + CRCL + CRCH
Response: Addr + 0x03 + 0x04 + 0xCC + 0xDD + 0xAA + 0xBB + CRCL + CRCH
Pressure value: The retrieved pressure data is of type float or long, with byte order CDAB

Example 1 Send: 01 03 00 02 00 02 65 CB Response: 01 03 04 42 31 3F A8 AF CA
The pressure value is 0x3FA84231 (float type), which equals 1.314520 in decimal
Example 2: Modify address 02
Command: 01 06 00 01 00 02 59 CB Response: 02 06 00 01 00 02 59 F8

### 1.3 Special Commands

Save data to user area: Addr + 0x10 + 0xF1 + 0xAA + 0x11 + 0x33 + 0x55 + CRCL + CRCH
Restore factory settings: Addr + 0x10 + 0xF3 + 0xAA + 0x11 + 0x33 + 0x55 + CRCL + CRCH
Reset: Addr + 0x10 + 0xB0 + 0xAA + 0x11 + 0x33 + 0x55 + CRCL + CRCH

### 1.4 Register Definition

| Register Start | Bytes | Definition | Date Type | Note |
|---|---|---|---|---|
| 001 / 0x01 | 1 / 2 | Sensor Address | Short | Default 0x01，0x01~0xEF |
| 002 / 0x02 | 2 / 4 | Pressure | Float | Realtime Pressure |
| 006 / 0x06 | 2 / 4 | Temperature ℃ | Float | Temperature sensor output |
| 010 / 0x0A | 2 / 4 | Voltage V | Float | Power supply |
| 028 / 0x1C | 2 / 4 | Pressure | Long | Realtime Pressure |
| 030 / 0x1E | 1 / 2 | Decimal | Short | Decimal digits |

| | | | | |
|---|---|---|---|---|
| 032 / 0x20 | 1 / 2 | Unit | Short | Unit, refer to “Unit Definition” |
| 034 / 0x22 | 1 / 2 | Sampling Rate | Short | 1 (20 times/sec, default)<br>0 (5 times/sec), 2 (50 times/sec) , 3 (100 times/sec) |
| 035 / 0x23 | 1 / 2 | Mode | Short | Refer to “Electrical Connection-Working Mode” |
| 042 / 0x2A | 1 / 2 | Temp. Switch | Short | Temperature switch, 0-Off / 1-On |
| 060 / 0x3C | 2 / 4 | Interval | u-Long | Sample time, unit second |
| 064 / 0x40 | 1 / 2 | Baud rate | Short | Refer to “Baud Rate” |
| 080 / 0x50 | 4 / 8 | Serial No. | Char | |
| 100 / 0x64 | 2 / 4 | Zero Value | Float | Transmitter zero pressure |
| 102 / 0x66 | 2 / 4 | Full value | Float | Transmitter full pressure |
| 104 / 0x68 | 2 / 4 | Analog Zero | Float | Set transmitter’s zero |
| 106 / 0x6A | 2 / 4 | Analog Full | Float | Set transmitter’s full |
| 108 / 0x6C | 2 / 4 | Offset | Float | Measure value-drift value |

### 1.5 Unit Definition

| Unit Register | Unit | Psi-Unit (Psi*N) | Unit-Psi (Unit*N) | Formal Unit |
|---|---|---|---|---|
| 00 / 0x00 | Pa | 6894.757 | 0.000145038 | pascals |
| 01 / 0x01 | hPa | 68.94757 | 0.01450377 | hectopascals |
| 02 / 0x02 | kPa | 6.894757 | 0.1450377 | kilopascals |
| 03 / 0x03 | MPa | 0.006894757 | 145.0377 | megapascals |
| 04 / 0x04 | mBar | 68.94757 | 0.01450377 | millibars |
| 05 / 0x05 | Bar | 0.06894757 | 14.50377 | bars |
| 06 / 0x06 | mTorr | 51715.08 | 1.93367E-05 | millitorr |
| 07 / 0x07 | Torr | 51.71508 | 0.01933672 | torr |
| 08 / 0x08 | PSI | 1 | 1 | pounds per square inch |
| 09 / 0x09 | Dy/cm2 | 68947.57 | 1.45038E-05 | dyne per square centimeter |
| 10 / 0x0A | gm/cm2 | 70.30697 | 0.01422334 | grams per square centimeter |
| 11 / 0x0B | kg/cm2 | 0.07030697 | 14.22334 | kilograms per square centimeter |
| 12 / 0x0C | PSF | 144 | 0.006944444 | pounds per square foot |
| 13 / 0x0D | OSI | 16 | 0.0625 | ounce per square inch |
| 14 / 0x0E | TSF | 0.072 | 13.88889 | tons per square foot |
| 15 / 0x0F | TSI | 0.0005 | 2000 | ton per square inch |
| 16 / 0x10 | mmH2O 4°C | 703.089 | 0.001422295 | millimeters of water @ 4°C |
| 17 / 0x11 | cmH2O 4°C | 70.3089 | 0.01422295 | centimeters of water @ 4°C |
| 18 / 0x12 | inH2O 4°C | 27.68067 | 0.03612629 | inches of water @ 4°C |
| 19 / 0x13 | ftH2O 4°C | 2.306726 | 0.4335149 | feet of water @ 4°C |
| 20 / 0x14 | MH2O 4°C | 0.703089 | 1.422295 | meters of water @ 4°C |

| | | | | |
|---|---|---|---|---|
| 21 / 0x15 | mmH2O 20°C | 704.336 | 0.001419777 | millimeters of water @ 20°C |
| 22 / 0x16 | cmH2O 20°C | 70.4336 | 0.01419777 | centimeter of water @ 20°C |
| 23 / 0x17 | inH2O 20°C | 27.72977 | 0.03606233 | inches of water @ 20°C |
| 24 / 0x18 | ftH2O 20°C | 2.310814 | 0.432748 | feet of water @ 20°C |
| 25 / 0x19 | MH2O 20°C | 0.704336 | 1.419777 | meters of water @ 20°C |
| 26 / 0x1A | inH2O 60°F | 27.70759 | 0.03609119 | inches of water @ 60°F |
| 27 / 0x1B | ftH2O 60°F | 2.308966 | 0.4330943 | feet of water @ 60°F |
| 28 / 0x1C | inSW 0°C 3.5% salinity | 26.92334 | 0.0371425 | inches seawater @ 0°C 3.5% salinity |
| 29 / 0x1D | ftSW 0°C 3.5% salinity | 2.243611 | 0.44571 | feet seawater @ 0°C 3.5% salinity |
| 30 / 0x1E | MSW 0°C 3.5% salinity | 0.6838528 | 1.462303 | meters seawater @ 0°C 3.5% salinity |
| 31 / 0x1F | µHg 0°C | 51715.08 | 1.93367E-05 | micron HG @ 0°C |
| 32 / 0x20 | mmHg 0°C | 51.71508 | 0.01933672 | millimeters of mercury @ 0°C |
| 33 / 0x21 | cmHg 0°C | 5.171508 | 0.1933672 | centimeters of mercury @ 0°C |
| 34 / 0x22 | inHg 0°C | 2.03602 | 0.4911544 | inches of mercury @ 0°C |
| 35 / 0x23 | inHg 60°F | 2.041772 | 0.4897707 | inches of mercury @ 60°F |
| 36 / 0x24 | ATM | 0.06804596 | 14.69595 | atmospheres |
| 37 / 0x25 | User | | | customized, default 1user=10psi |

### 1.6 Baud Rate

| Addr | Baud |
|---|---|
| 0 / 0x00 | 1200 |
| 1 / 0x01 | 2400 |
| 2 / 0x02 | 4800 |
| 3 / 0x03 | 9600 |
| 4 / 0x04 | 19200 |
| 5 / 0x05 | 38400 |

### 1.7 CRC Verify Code Program Reference

```
uint16_t crc_val[2]={0};
void crc16(uint8_t *ptr,uint8_t len2)
{
    uint16_t temp,wcrc=0xFFFF, i=0, j=0;
    for(i=0;i<len2;i++)
    {
        temp=*ptr&0x00FF;
        ptr++;
        wcrc^=temp;
```

```
            for(j=0;j<8;j++)
            {
                if(wcrc&0x0001)
                {
                    wcrc>>=1;
                    wcrc^=0xA001;
                }
                else
                {
                    wcrc>>=1;
                }
            }
        }
        crc_val[0]=(wcrc&0x00ff);
        crc_val[1]=wcrc>>8;
}
```

## 2 I²C Guide

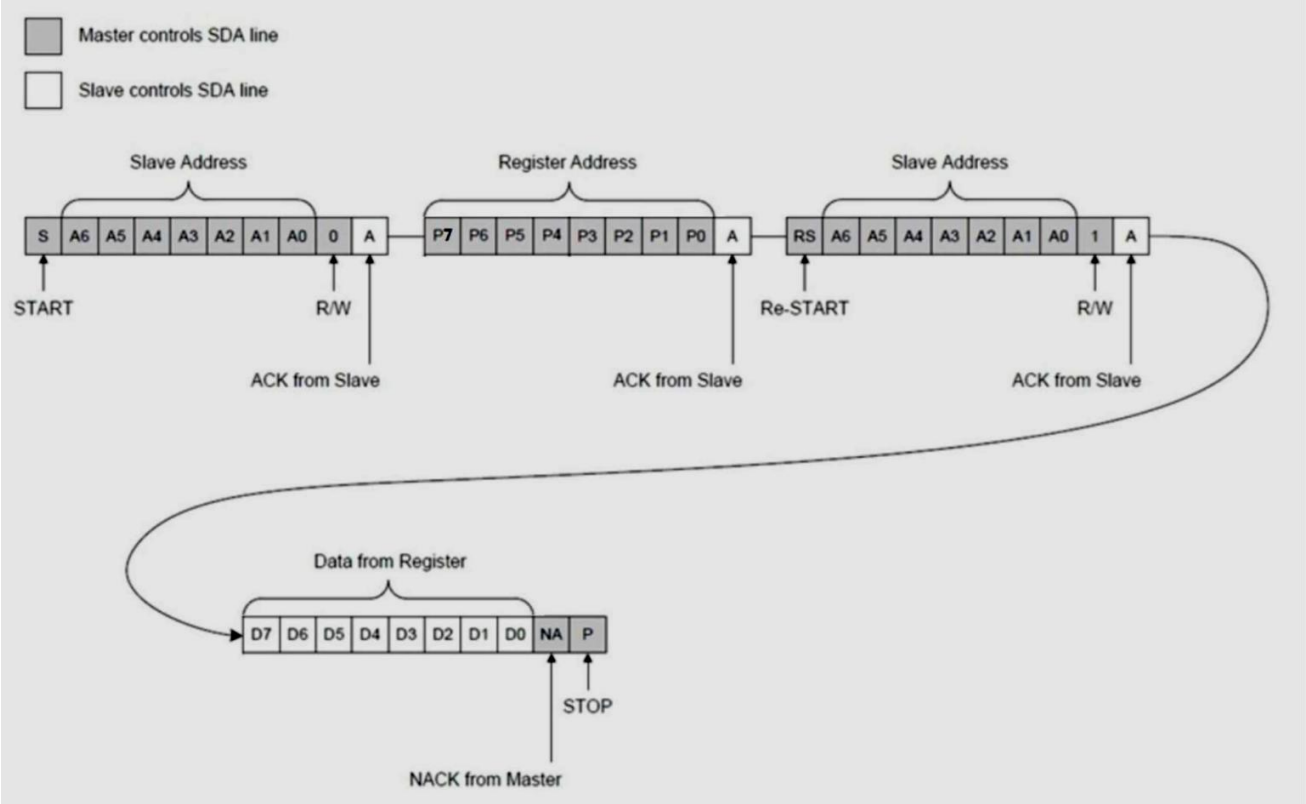
### 2.1 Overview

Digital sensors use the SDA and SCL pins to implement the I²C bus communication protocol. Each transmission sequence begins with a START condition (S) and ends with a STOP condition (P), as described in the I²C bus specification. The slave I²C address is 7 bits long, with a default value of 0x40.

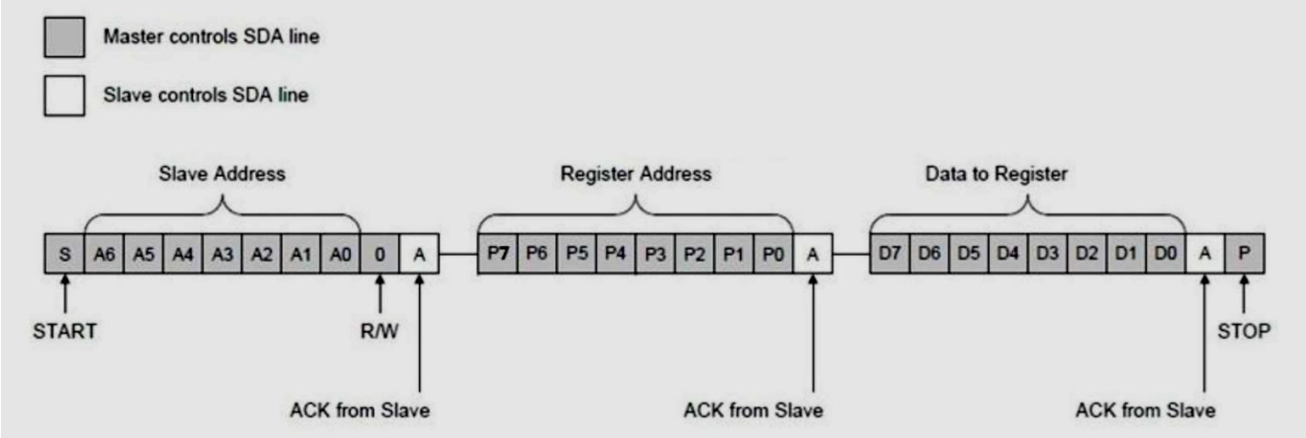
### 2.2 Read and Write Rules

In I²C bus communication, the master first transmits the slave address and the write-flag (Slave Address + W), followed immediately by the register logical address (Register Address). For reading operations, the master then transmits the slave address and the read-flag (Slave Address + R) again, after which the slave sends data to the master (Data from Register). Multiple slave register data can be read sequentially in this manner; For the write sequence, the master then sends data directly to the slave (Data to Register). Note that the slave address is 7 bits wide. Write-flag W=0, read-flag R=1. The specific read and write timing diagrams are as follows:

Read Operation



Write Operation




When reading from a register, the host can read the data from the parity register to verify the accuracy of the transmitted data and check specific fields.

### 2.3 Interface Operations

2.3.1 Under normal power consumption conditions, the digital sensor continuously collects data. After each data collection cycle is complete and the register data is updated, the INT pin outputs a low level. Users can read the data at any time as needed, or read it based on the INT interrupt signal. When reading collected data, it is recommended to read the 13 bytes of data from registers 00 to 12 consecutively and read the checksum to verify that the data was transmitted correctly.

2.3.2 In low-power mode, the digital sensor is in a sleep state. The user can wake up the sensor for data acquisition by writing 0x11 to register 0x1D. Once data acquisition is complete and the register data has been updated, the INT pin will output a low level. The user can read the data after a delay of more than 300 ms, or read it based on the INT low-level signal.

2.3.3 Software reset. The user can write 0x55 to register 0x1D, followed immediately by writing 0xB0 to register 0x1D, to perform a software reset on the digital sensor.

2.3.4. Saving User Settings. The user can write 0x55 to register 0x1D, followed by writing 0xF1 to register 0x1D, to save modified user data and prevent loss due to power failure. This is commonly used for modifying pressure units and offsets.

2.3.5 External Reset. In $I^2C$ slave, SPI slave, and SPI Streaming modes, pull the CS and DATA pins low while keeping the CLK pin high. After approximately 500 ms, the digital sensor will restart.

### 2.4 Register Definition

| Register Start | Bytes | Definition | Date Type | Note |
|---|---|---|---|---|
| 00 / 0x00 | 4 | Pressure | Float | Real-time digital pressure output |
| 04 / 0x04 | 4 | Temperature ℃ | Float | Temperature sensor output |
| 08 / 0x08 | 4 | Voltage V | Float | Power supply voltage |
| 12 / 0x0C | 1 | Verify Code | uChar | 00-12 Verify code of register |
| 16 / 0x10 | 4 | Zero Pressure | Float | Sensor zero pressure value |
| 20 / 0x14 | 4 | Full Pressure | Float | Sensor full pressure value |
| 24 / 0x18 | 4 | Offset | Float | Measure value-drift value |
| 28 / 0x1C | 1 | Unit | uChar | Pressure Units |
| 29 / 0x1D | 1 | Mode | uChar | |
| 30 / 0x1E | 1 | Verify Code | uChar | 16-28 verify code of register |
| Note: In the table, the byte order for Float data is low-byte first, high-byte last. | | | | |

### 2.5 Verify Code Demonstration

```
uint8_t CRC8MY(uint8_t *serial, uint8_t length)
{
    uint8_t result = 0x00;
    uint8_t pDataBuf;
    uint8_t i;
    while(length--)
    {
        pDataBuf = *serial++;
        for(i=0; i<8; i++)
        {

            if((result^(pDataBuf))&0x01)
            {
                result ^= 0x18;
                result >>= 1;
                result |= 0x80;
            }
            else
            {
                result >>= 1;
            }
            pDataBuf >>= 1;
        }
    }
    return result; //返回校验和
}
```

# 3 SPI Slave Mode Guide

## 3.1 Overview

In this mode, the digital sensor acts as a three-wire SPI slave and performs only data output; SPI operates in Mode 3 (CPOL=1, CPHA=1).

## 3.2 Reading Rules

After the DATA pin outputs a low-level pulse to indicate data readiness, it functions as the SPI data output (MISO pin); the SCK pin serves as the SPI clock input (SCLK pin); and the CS/INT pin acts as the SPI chip select input (CS pin), while also being used for wake-up in low-power mode.

## 3.3 Interface Operation

3.3.1. Under normal power consumption conditions, the digital sensor continuously performs data acquisition. After each data acquisition is completed and the register data is updated, the DATA pin outputs a low-level pulse. Users may choose whether to read the data as needed. It is recommended to continuously read the 13 bytes of data from register 00 to 12 and read the checksum to verify that the data has been transmitted correctly.

3.3.2 In low-power mode, the digital sensor is in sleep mode. The user can send a low-level pulse to the CS/INT pin to wake up the sensor for data acquisition. Once data acquisition is complete and the register data is updated, the DATA pin will output a low level. After the user reads the data or approximately 200 ms after data conversion is complete, the digital sensor automatically enters sleep mode. The user may read the data after a period of more than 300 ms following wake-up or based on the low-level signal on the DATA pin.

3.3.3 External Reset. In $I^2C$ slave, SPI slave mode, and SPI Streaming modes, pulling the CS and DATA pins low while keeping CLK high for approximately 500 ms will cause the digital sensor to reboot.

3.3.4 Register Definition

| Register Start | Bytes | Definition | Date Type | Note |
|---|---|---|---|---|
| 00 / 0x00 | 4 | Pressure | Float | Real-time digital pressure output |
| 04 / 0x04 | 4 | Temperature ℃ | Float | Temperature sensor output |
| 08 / 0x08 | 4 | Voltage V | Float | Power supply voltage |
| 12 / 0x0C | 1 | Verify Code | uChar | 00-12 Verify code of register |
| Note: In the table, the byte order for Float data is low-byte first, high-byte last. | | | | |

## 3.4 Verify Code Programming （Refer to 2.5）

# 4 SPI Streaming Guide

## 4.1 Overview

In this mode, the digital sensor acts as a three-wire SPI streaming and performs only data output; SPI operates in Mode 3 (CPOL=1, CPHA=1).

## 4.2 Reading Rules

The DATA pin serves as the SPI data output (MOSI), the SCK pin serves as the SPI clock input (SCLK), and the CS/INT pin serves as the SPI chip select (CS) output.

## 4.3 Interface Operation

4.3.1 Under normal power consumption, the digital sensor continuously collects data. After each data collection cycle is complete and the register data is updated, it actively transmits 13 bytes of data from registers 00 to 12 to the SPI bus. The user receives the data and compares it with the checksum to verify that the data was transmitted correctly.

4.3.2 In low-power mode, the digital sensor is in a sleep state. At set time intervals, the sensor is periodically awakened to perform data acquisition. Once data acquisition is complete and the register data is updated, it actively transmits 13 bytes of data from registers 00 to 12 to SPI bus. The digital sensor then automatically returns to sleep mode. The user receives the data and compares it with the checksum to verify that the data was transmitted correctly.

4.3.3 External Reset. In $I^2C$ slave, SPI slave, and SPI streaming modes, pull the CS and DATA pins low while keeping the CLK pin high. After approximately 500 ms, the digital sensor will perform a restart operation.

4.3.4 Register Definition

| Register Start | Bytes | Definition | Date Type | Note |
|---|---|---|---|---|
| 00 / 0x00 | 4 | Pressure | Float | Real-time digital pressure output |
| 04 / 0x04 | 4 | Temperature ℃ | Float | Temperature sensor output |
| 08 / 0x08 | 4 | Voltage V | Float | Power supply voltage |
| 12 / 0x0C | 1 | Verify Code | uChar | 00-12 Verify code of register |
| Note: In the table, the byte order for Float data is low-byte first, high-byte last. | | | | |

## 4.4 Verify Code Programming (Refer to 2.5)