# IMU6488 Series High-Precision Inertial Measurement Unit
# User Manual

March 2025

## Version History

| Version | Date | Pages | Description |
|---|---|---|---|
| C | 20250327 | 14 | Original Release |

# 1. Introduction to IMU6488 Series

The IMU6488 series inertial measurement unit has built-in three-axis gyroscopes and three-axis accelerometers for measuring the three-axis angular rate and three-axis acceleration of a carrier. The gyroscope and accelerometer data, which have undergone error compensation (including temperature compensation, installation misalignment angle compensation, non-linear compensation, etc.), are output through a serial port according to the agreed communication protocol, and can be used for precise navigation, control, and dynamic measurement of the carrier. This product adopts high-precision MEMS inertial devices, featuring small size, strong anti-overload capability, high reliability, and high hardness, and has the ability to accurately measure the angular velocity and acceleration information of a moving carrier even in harsh environments.

The IMU6488 series includes four models: IMU6488B, IMU6488C, IMU6488D, and IMU6488E. The appearance and interface performance of the four models are identical. All of them are equipped with built-in three-axis gyroscopes and three-axis accelerometers. Among them, IMU6488B, IMU6488C, and IMU6488D can be optionally equipped with a magnetometer and a barometer.

| Product | Gyroscope (Standard) | Accelerometer (Standard) | Magnetometer (Optional) | Barometer (Optional) |
|---|---|---|---|---|
| IMU6488B | Yes | Yes | Optional | Optional |
| IMU6488C | Yes | Yes | Optional | Optional |
| IMU6488D | Yes | Yes | Optional | Optional |
| IMU6488E | Yes | Yes | No | No |

**Key Features**

- High-precision MEMS configuration with 3-axis accelerometer and 3-axis gyroscope
- Range: Accelerometer ±18g, Gyroscope ±450°/s
- Communication Interface: SPI
- High output frequency: 2000Hz (1000Hz for IMU6488E)
- Wide temperature range: -40°C ~ +80°C
- Small size: 47(L) × 44(W) × 14(H) mm

**Application Fields**

- Anti-vibration and attitude control in the industrial field
- Automated machinery
- Robot control
- Platform stabilization and control
- Instrumentation

# 2. Technical Specifications of IMU6488 Series

The technical specifications of the IMU6488 series are shown in the table below:

| Category | Parameter | B | C | D | E | Unit |
|---|---|---|---|---|---|---|
| Gyroscope | Range: X, Y, Z | ±450 | ±450 | / | ±450 | °/s |
| | Bias Instability | 3 | 1 | / | 0.3 | °/h |
| | Random Walk | 0.08 | 0.08 | / | 0.03 | °/√hr |
| | Bias Stability | 20 | 6 | / | 3 | °/h |
| | Scale Factor Non-linearity | 300 | 300 | / | 300 | ppm |
| | Cross Coupling | 200 | 200 | / | 100 | ppm |
| | Bandwidth | 200 | 200 | / | 268 | Hz |
| Accelerometer | Range: X, Y, Z | ±18 | ±18 | / | ±18 | g |
| | Bias Instability | 80 | 50 | / | 20 | µg |
| | Random Walk | 0.08 | 0.05 | / | 0.02 | m/s/√hr |
| | Bias Stability | 800 | 500 | / | 100 | µg |
| | Scale Factor Non-linearity | 250 | 250 | / | 150 | ppm |
| | Cross Coupling | 200 | 200 | / | 100 | ppm |
| | Bandwidth | 268 | 268 | / | 268 | Hz |
| Magnetometer (Optional) | Range | ±8 | ±8 | ±8 | / | Gauss |
| | Resolution | 0.2 | 0.2 | 0.2 | / | mG |
| | Noise Density | 50 | 50 | 50 | / | µG |
| Barometer (Optional) | Measurement Range | 300~1100 | 300~1100 | 300~1100 | / | mbar |
| | Tolerance | 4.5 | 4.5 | 4.5 | / | mbar |
| Basic Parameters | Dimensions | 47*44*14 | 47*44*14 | 47*44*14 | 47*44*14 | mm |
| | Weight | <40 | <40 | <40 | <40 | g |
| | Operating Voltage | 3.3 | 3.3 | 3.3 | 3.3 | VDC |
| | Power Consumption | <0.8 | <0.8 | <0.8 | <0.8 | W |
| | Interface | SPI | SPI | SPI | SPI | |
| | ODR Output Frequency | 2000 | 2000 | 2000 | 1000 | Hz |
| | Operating | -40 ~ +80 | -40 ~ +80 | -40 ~ +80 | -40 ~ | °C |

| | | | | | | |
|---|---|---|---|---|---|---|
| | Temperature | | | | +80 | |
| | Storage Temperature | -55 ~ +105 | -55 ~ +105 | -55 ~ +105 | -55 ~ +105 | °C |
| | Random Vibration | 6.06g (20-2000Hz) | 6.06g (20-2000Hz) | 6.06g (20-2000Hz) | 6.06g (20-2000Hz) | |
| | Shock | 1500g / 0.5ms | 1500g / 0.5ms | 1500g / 0.5ms | 1500g / 0.5ms | |

# 3. Product Hardware Description

## 3.1 Coordinate System Definition and Outline Dimensions

System dimensions are: 47.0(L) × 44.0(W) × 14.0(H) mm. [See Figure 1 for detailed dimensions]

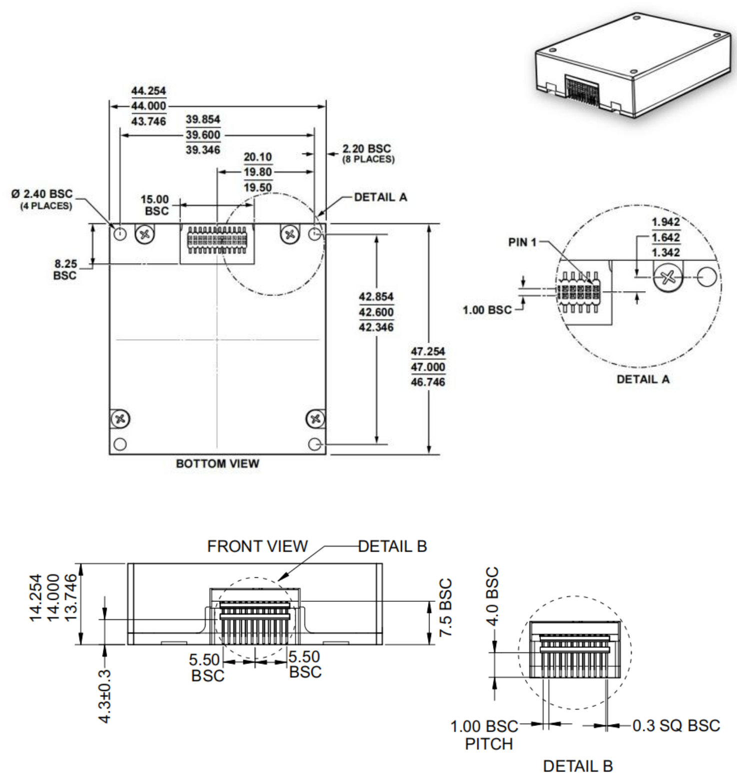
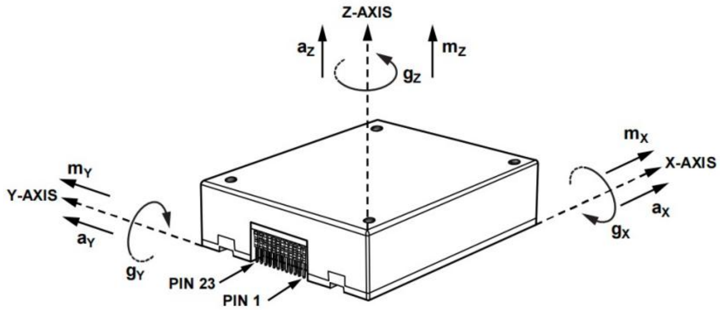



*Figure 1*

Coordinate system definition is shown in [Figure 2].




## 3.2 Electrical Interface

The interface pins are illustrated in [Figure 3].

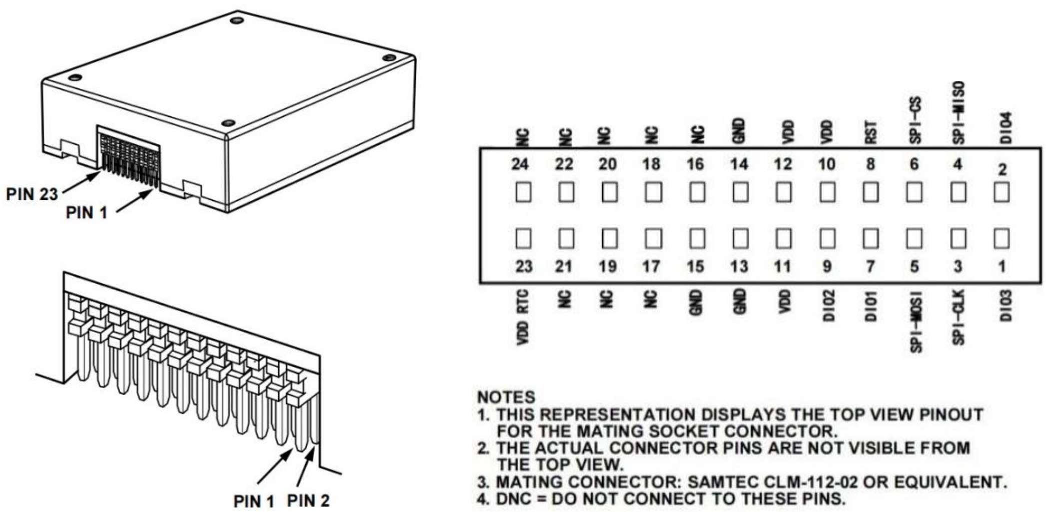



*Figure 3: Interface Pin Diagram*

The electrical interface definitions are shown in Table 2:

| Pin Number | Name | Type | Description |
|---|---|---|---|
| 10, 11, 12 | VDD | Power | 3.3V |
| 13, 14, 15 | GND | Power Ground | |
| 7 | DIO1 | I/O | General IO, configurable |
| 9 | DIO2 | I/O | General IO, configurable |
| 1 | DIO3 | I/O | General IO, configurable |
| 2 | DIO4 | I/O | General IO, configurable |
| 3 | SPI-CLK | Input | SPI master/slave |

| | | | |
|---|---|---|---|
| | | | mode cannot be configured |
| 4 | SPI-MISO | Output | SPI master/slave mode cannot be configured |
| 5 | SPI-MOSI | Input | SPI master/slave mode cannot be configured |
| 6 | SPI-CS | Input | SPI master/slave mode cannot be configured |
| 8 | RST | Input | Reset |
| 23 | VDD RTC | Power | N/A |
| 16, 17, 18, 19, 20, 21, 22, 24 | NC | Reserved | DNC |

# 4. Software Interface Description

## 4.1 SPI Interface Instructions

The IMU6488 series is an autonomous sensor system. It will automatically start when a valid power supply is connected. After completing the initialization process, it begins to sample and process the calibrated sensor data and loads it into the output register, which can be accessed via the SPI port. The SPI port is typically connected to a compatible port on an embedded processor. [See Figure 4 for connection diagram]. Four SPI signals support synchronous serial data transmission. In the factory default configuration, DIO2 provides the data ready signal; the pin goes high when new data is available in the output data register.

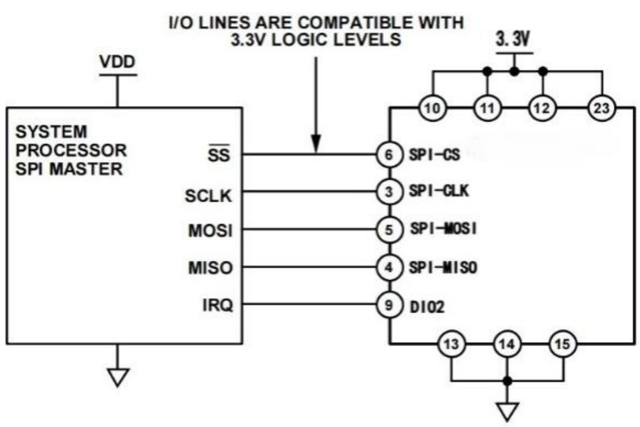



*Figure 4: Electrical Connection Diagram*

Table 3: General Host SPI Settings Table

| Processor Settings | Description |
|---|---|
| Host | IMU6488 series acts as slave |
| SCLK < 10 MHz | Maximum serial clock rate |
| SPI Mode 3 | CPOL = 1 (Polarity), and CPHA = 1 (Phase) |
| MSB First Mode | Bit order |
| 16-bit Mode | Shift register / data length |

If the preceding command is a read request, the SPI port supports full-duplex communication, which allows the external processor to write to DIN while reading DOUT. [See Figure 5] for guidelines on bit coding on DIN and DOUT.

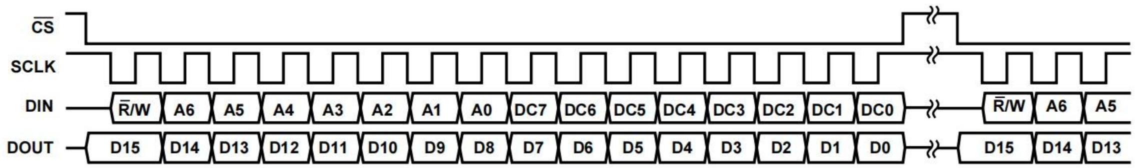



*Figure 5: SPI Communication Bit Sequence*

## 4.2 Reading Sensor Data

The IMU6488 series automatically starts and activates Page 0 for data register access. After accessing any other page, write 0x00 to the PAGE_ID register (DIN=0x8000) to activate Page 0 for data access. A single register read requires two 16-bit SPI cycles.

The first cycle requests the contents of the register using the bit assignment in [Figure 5], and then the register contents follow DOUT in the second sequence. The first bit in the DIN command is zero, followed by the upper or lower address of the register. The last 8 bits are insignificant, but the SPI requires a full 16 SCLKs to receive the request.

[Figure 6] includes two consecutive register reads, starting from DIN=0x1A00, requesting the contents of the Z_GYRO_OUT register, and then starting from 0x1800, requesting the contents of the Z_GYRO_LOW register.

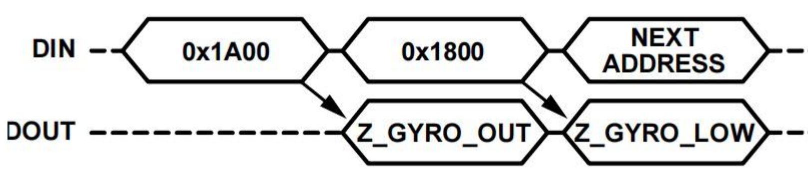



*Figure 6: SPI Read Example*

## 4.3 User Register Memory Map

Table 4: User Register Map

| R/W | PAGE_ID | Address | Default | Register Description |
|---|---|---|---|---|
| R/W | 0x00 | 0x00 | 0x00 | Page Identifier |
| R | 0x00 | 0x0E | N/A | Temperature |
| R | 0x00 | 0x10 | N/A | X-axis Gyroscope Output, Lower Word |
| R | 0x00 | 0x12 | N/A | X-axis Gyroscope Output, Upper Word |
| R | 0x00 | 0x14 | N/A | Y-axis Gyroscope Output, Lower Word |
| R | 0x00 | 0x16 | N/A | Y-axis Gyroscope Output, Upper Word |
| R | 0x00 | 0x18 | N/A | Z-axis Gyroscope Output, Lower Word |
| R | 0x00 | 0x1A | N/A | Z-axis Gyroscope Output, Upper Word |
| R | 0x00 | 0x1C | N/A | X-axis Accelerometer Output, Lower Word |
| R | 0x00 | 0x1E | N/A | X-axis Accelerometer Output, Upper Word |
| R | 0x00 | 0x20 | N/A | Y-axis Accelerometer Output, Lower Word |
| R | 0x00 | 0x22 | N/A | Y-axis Accelerometer Output, Upper |

| | | | | |
|---|---|---|---|---|
| | | | | Word |
| R | 0x00 | 0x24 | N/A | Z-axis Accelerometer Output, Lower Word |
| R | 0x00 | 0x26 | N/A | Z-axis Accelerometer Output, Upper Word |
| R | 0x00 | 0x28 | N/A | X-axis Magnetometer, Upper Word |
| R | 0x00 | 0x2A | N/A | Y-axis Magnetometer, Upper Word |
| R | 0x00 | 0x2C | N/A | Z-axis Magnetometer, Upper Word |
| R | 0x00 | 0x2E | N/A | Barometer Output, Lower Word |
| R | 0x00 | 0x30 | N/A | Barometer Output, Upper Word |
| R/W | 0x03 | 0x00 | 0x00 | Page Identifier |
| R/W | 0x03 | 0x06 | 0x000D | Control, I/O Pins, Function Definition |
| R/W | 0x03 | 0x08 | 0x00X0 | Control, I/O Pins, General |
| R/W | 0x04 | 0x00 | 0x00 | Page Identifier |
| R | 0x04 | 0x20 | N/A | Serial Number |

## 4.4 Inertial Sensor Data Parsing

The gyroscope, accelerometer, and barometer output data registers use a 32-bit two's complement format. Each output uses two registers to support this resolution. [Figure 7] provides an example of how each register supports each inertial measurement. In this case, X_GYRO_OUT is the most significant word (upper 16 bits), and X_GYRO_LOW is the least significant word (lower 16 bits). In many cases, using only the most significant word register provides sufficient resolution for retaining key performance metrics.

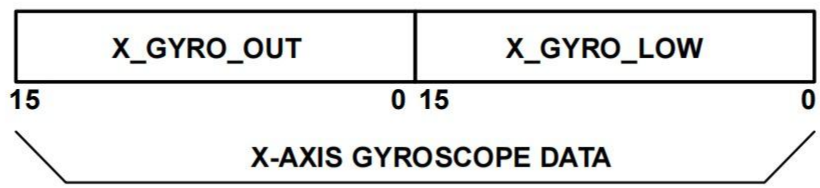



*Figure 7: Gyroscope Output Format Example, DEC_RATE > 0*

Current Temperature = 25 + TEMP_OUT * 0.00565

Angular Rate Value Calculation:

| X-axis Gyroscope | X_GYRO_OUT | X_GYRO_LOW |
|---|---|---|
| X-axis Gyroscope | 1LSB = 0.02 °/s | The weight of MSB is 0.01°/s, and the weight of subsequent data is half of the previous data |
| X-axis Gyroscope | 0.02 * X_GYRO_OUT | 0.01*MSB + 0.005*... |

Acceleration Value Calculation:

| X-axis Accelerometer | X_ACCL_OUT | X_ACCL_LOW |
|---|---|---|
| X-axis Accelerometer | 1LSB = 0.8mg | The weight of MSB is 0.4mg, and the weight of subsequent data is half of the previous data |
| X-axis Accelerometer | 0.8 * X_ACCL_OUT | 0.4 * MSB + 0.2*... |

Magnetic Field Value Calculation:

| X-axis Magnetometer | X_MAGN_OUT | |
|---|---|---|
| X-axis Magnetometer | 1LSB = 0.1mGauss | |
| X-axis Magnetometer | 0.1 * X_MAGN_OUT | |

Pressure Value Calculation:

| Pressure (Bar) | BAROM_OUT | BAROM_LOW |
|---|---|---|
| Pressure (Bar) | 1LSB = 40ubar | The weight of MSB is 20ubar, and the weight of subsequent data is half of the previous data |
| | 40 * BAROM_OUT | 20 * MSB + 10*.... |

## 5. SPI Code Example

```
#define LSB_GYRO        0.02f              //1LSB = 0.02 °/s
#define LSB_ACC        0.8f                //1LSB = 0.8 mg
#define LSB_MAG        0.1f                //1LSB = 0.1 mgauss
#define LSB_BAR        40.0f         //1LSB = 40ubar
#define LSB_TEMP        0.00565f           //1LSB = 0.00565 °C



struct sensor_data
{
     float gyro_x;   // °/s
     float gyro_y;   // °/s
     float gyro_z;   // °/s
     float acc_x;    // g
     float acc_y;    // g
     float acc_z;    // g
     float mag_x;    // gauss
     float mag_y;    // gauss
     float mag_z;    // gauss
     float barom;    // bar
     float temp;          // °C
}sensor_data_6488;

void spi_rw_buf_16bits(uint16_t *tx_buf, uint16_t *rx_buf,
uint16_t len)
{
     spi_cs_enable(&spi_6488);
     spi_transmit_receive(&spi_6488, tx_buf, rx_buf, len);
     spi_cs_disable(&spi_6488);
}
uint16_t read_id(void)
{
     uint16_t tx_buf[3] = {0x8000, 0x7E00, 0x0000};
     uint16_t rx_buf[3] = {0};

     spi_rw_buf_16bits(tx_buf, rx_buf, 3);
```

```
	return rx_buf[2];
}

void set_decimation_rate(uint16_t divisor)
{
	uint16_t tx_buf[4] = {0};
	uint16_t rx_buf[4] = {0};
	uint16_t data_temp[2] = {0};

	divisor = divisor - 1;
	data_temp[0] = divisor & 0x00FF;
	data_temp[1] = divisor >> 8;


	while(1)
	{
		tx_buf[0] = 0x8003;   //page3
		tx_buf[1] = 0x0c00;   //read
		tx_buf[2] = 0x0000;   //dummy
		spi_rw_buf_16bits(tx_buf, rx_buf, 4);

		if(rx_buf[2]==divisor)
		{
			tx_buf[0] = 0x8000;   //page0
			spi_rw_buf_16bits(tx_buf, rx_buf, 1);
			return;
		}

		tx_buf[0] = 0x8c00 | data_temp[0];
		tx_buf[1] = 0x8d00 | data_temp[1];
		spi_rw_buf_16bits(tx_buf, rx_buf, 2);
	}
}

uint32_t sensor_init(void)
{
	uint16_t id = read_id();

	if(id != 0x4068)
	{
		return 0;
	}

	set_decimation_rate(1);          //set ODR

	return 1;
}



void get_6488_data()
{
```

```
	int16_t rx_data[20];

	uint16_t tx_data[20] =
	{
	0x8000,	//page0
	0x0e00,	//temp_out
	0x1000,	//gx_low
	0x1200,	//gx_out
	0x1400,	//gy_low
	0x1600,	//gy_out
	0x1800,	//gz_low
	0x1a00,	//gz_out
	0x1c00,	//ax_low
	0x1e00,	//ax_out
	0x2000,	//ay_low
	0x2200,	//ay_out
	0x2400,	//az_low
	0x2600,	//az_out
	0x2800,	//mx_out
	0x2a00,	//my_out
	0x2c00,	//mz_out
	0x2e00,	//barom_low
	0x3000,	//barom_out
	0x0000	//dummy
	};

	spi_rw_buf_16bits(tx_data, (uint16_t*)rx_data, 20);

	sensor_data_6488.temp    = rx_data[2]*LSB_TEMP + 25.0f;
sensor_data_6488.gyro_x = rx_data[4]*LSB_GYRO +
(uint16_t)rx_data[3]*LSB_GYRO/65536.0f;
sensor_data_6488.gyro_y = rx_data[6]*LSB_GYRO +
(uint16_t)rx_data[5]*LSB_GYRO/65536.0f;
sensor_data_6488.gyro_z = rx_data[8]*LSB_GYRO +
(uint16_t)rx_data[7]*LSB_GYRO/65536.0f;
sensor_data_6488.acc_x  = (rx_data[10]*LSB_ACC +
(uint16_t)rx_data[9]*LSB_ACC/65536.0f) / 1000.0f;
sensor_data_6488.acc_y  = (rx_data[12]*LSB_ACC +
(uint16_t)rx_data[11]*LSB_ACC/65536.0f) / 1000.0f;
sensor_data_6488.acc_z  = (rx_data[14]*LSB_ACC +
(uint16_t)rx_data[13]*LSB_ACC/65536.0f) / 1000.0f;
	sensor_data_6488.mag_x  = (rx_data[15]*LSB_MAG) / 1000.0f;
	sensor_data_6488.mag_y  = (rx_data[16]*LSB_MAG) / 1000.0f;
	sensor_data_6488.mag_z  = (rx_data[17]*LSB_MAG) / 1000.0f;
sensor_data_6488.barom   = (rx_data[19]*LSB_BAR +
(uint16_t)rx_data[18]*LSB_BAR/65536.0f) /1000000.0f;
}
```

# 6. Precautions

### Ø SPI Configuration Mode

- The SPI operates in Mode 3.
- The SPI data word length is 16 bits; 32-bit width is not supported.

### Ø Data Sign Bit Processing (Two's Complement Format)

- Pay attention to the sign bit processing when handling negative numbers. For example, for 16-bit data 0xFD88, the parsing/conversion process is as follows:
- 0xFD88 -> 64904 -> 64904 - 65536 ($2^{16}$) -> -632.

### Ø Power-on Automatic Calibration Function

- The product features a power-on automatic calibration function. Please ensure that the device remains completely stationary within the first 5 seconds after powering up. If you need to disable this power-on automatic calibration function, please contact the manufacturer.

**Haisi Measurement and Control Technology Co., Ltd.**

Mobile: +86-180-6672-6340

Email: sales@haisitek.com

Address: Room A320, Innovation Digital Space, No. 2 Fenglin Road, High-tech Zone, Xi’an