# QIA128/IDC150/IEM100

## UART Communication Guide

Sensor Solutions Source
Load · Torque · Pressure · Multi-Axis · Calibration · Instruments · Software

www.futek.com

DSPM Industria
sensori & trasduttori
Via Paolo Uccello 4 - 20148 Milano
Tel +39 02 48 009 757 Fax +39 02 48 002 070
info@dspmindustria.it www.dspmindustria.it

**Drawing Number: EM1066-G**

## General Description

The QIA128/IDC150/IEM100 is a single-channel, ultra-low-power digital controller featuring UART and SPI interfaces.

## Pin Configurations and Function Descriptions

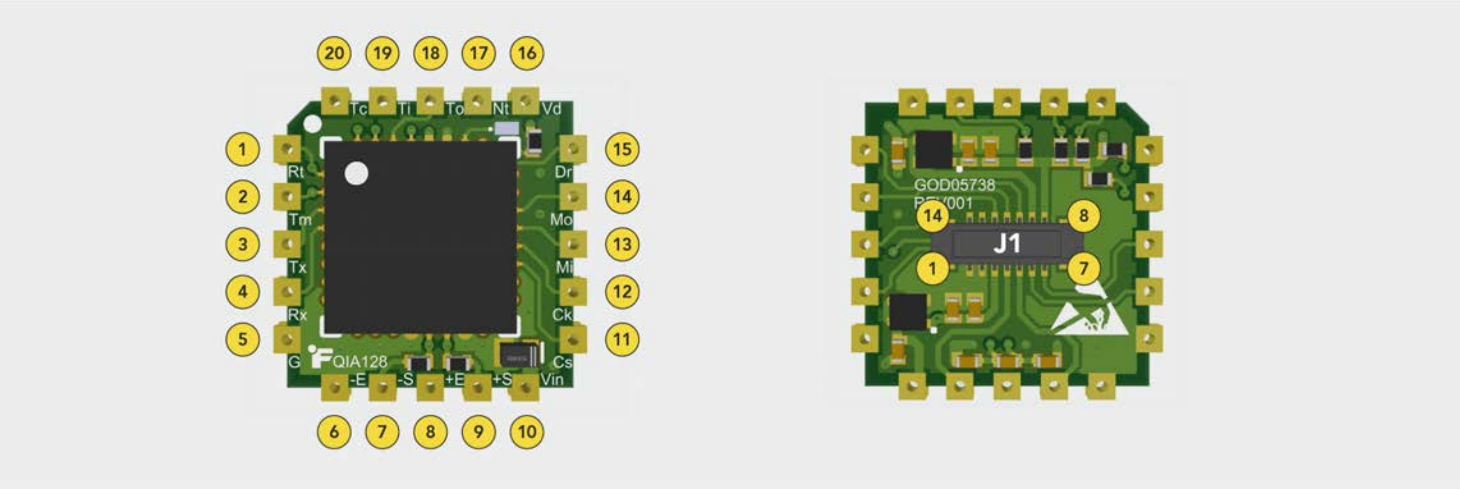


Figure 1.

| TABLE 1 | | | |
|---|---|---|---|
| **#** | **PIN** | **DESCRIPTION** | **J1#** |
| 1 | $\overline{RESET}$ | Active low reset input. | - |
| 2 | TMS | JTAG Test Mode Select input for debug and firmware download. | - |
| 3 | RX | UART receive data input. | 7 |
| 4 | TX | UART transmit data output. | 6 |
| 5 | GND | Ground. All ground pins are internally connected. | 1 |
| 6 | -Excitation | Sensor excitation return, internally connected to GND. | 2 |
| 7 | -Signal | Sensor negative input. | 5 |
| 8 | +Excitation | Sensor excitation output. | 3 |
| 9 | +Signal | Sensor positive input. | 4 |
| 10 | VIN | Supply input, 3 to 5 VDC. | 9 |
| 11 | $\overline{CS}$ | Active low chip select input. Do not assert $\overline{CS}$ low until the device has completed boot. Do not assert $\overline{CS}$ low unless $\overline{DRDY}$ is low. | 14 |
| 12 | SCLK | SPI serial clock input from host. | 13 |
| 13 | HICO | SPI Host In Client Out data line. | 12 |

| TABLE 1 | | | |
|---|---|---|---|
| **#** | **PIN** | **DESCRIPTION** | **J1#** |
| 14 | HOCI | SPI Host Out Client In data line. | 11 |
| 15 | $\overline{\text{DRDY}}$ | Active low data ready output. Indicates new sampled data is available and keeps host communication synchronized. $\overline{\text{DRDY}}$goes low when data is ready to be clocked out and may be used as a host interrupt. $\overline{\text{DRDY}}$ is high during conversion and returns low when new data is available. **Note:** $\overline{\text{DRDY}}$does not return high after data is read. It returns high only when the system enters a conversion state. | 10 |
| 16 | VDD | Digital supply rail, 2.5 V. | - |
| 17 | NTRST | JTAG NTRST and boot mode control input for debug and firmware download only. | - |
| 18 | TDO | JTAG Test Data Out. | - |
| 19 | TDI | JTAG Test Data In. | - |
| 20 | TCK | JTAG test clock input for debug and firmware download. | - |

## QIA128/IDC150/IEM100 UART Configuration

| TABLE 2 | |
|---|---|
| **Data Length** | 8-Bit |
| **Baud Rate** | 320,000bps |
| **Parity** | None |
| **Stop bits** | 1-Bit |
| **Flow Control** | None |

## $\overline{DRDY}$ Pin Functionality

When the $\overline{DRDY}$ pin is high, the device is performing data conversion and post-processing. $\overline{DRDY}$ transitions low immediately once the conversion and associated post-processing are complete. **Note:** UART is inherently asynchronous. The $\overline{\text{DRDY}}$ signal can be used at the system level to synchronize data transfer and timing between the host and the device, when required.

## $\overline{DRDY}$ Period:

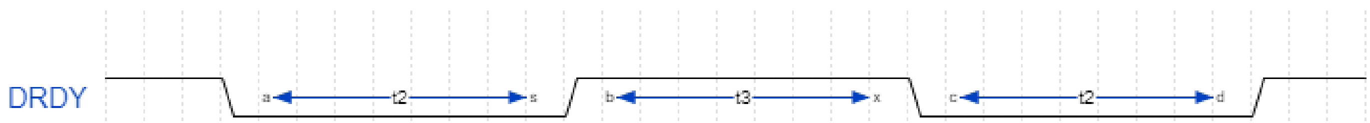


Figure 4.

The following table shows the high and low timing periods of the $\overline{DRDY}$ signal for all supported sampling rates.

| TABLE 3 | | |
|---|---|---|
| $t_2$ (ms) | $t_3$ (µs) | DESCRIPTION |
| 240 | 125 | 4 SPS |
| 55 | | 20 SPS |
| 19 | | 50 SPS |
| 9 | | 100 SPS |
| 4.5 | | 200 SPS |
| 1.5 | | 500 SPS |
| 1.1 | | 850 SPS |
| 0.6 | | 1300 SPS |

## Stream Mode

Stream Mode is enabled by sending the Set System Stream State (SSSS) command with a payload value of 1. Once activated, the device continuously streams data until streaming is disabled.

Streaming stops immediately when the Set System Stream State command is sent with a payload value of 0, or when any other valid command is issued to the QIA128/IDC150/IEM100.

**Note:** If an invalid or unsupported command is transmitted, the QIA128/IDC150/IEM100 may not generate a response.

## UART Packet Structure

The UART packet format and length vary depending on the command type (GET or SET) and its specific functionality. Refer to the Command Set Table for detailed packet definitions and requirements.

## System Behavior

### Start-up and Self-Calibration Mode

At power-up, the device reads configuration data from internal flash memory and then enters internal self-calibration. **Note:** The first $\overline{DRDY}$ pulse can be used as an indication that the device has completed initialization and is ready for system-level communication.

### Sampling Rate Change

When a sampling rate change is requested, the updated $\overline{DRDY}$ period will be reflected within 0.5 seconds, depending on the selected sampling rate.

## Sampling Rates

**TABLE 4**

| Maximum Approximate data rate change timing (ms) | SR Code | Sampling Rate |
|---|---|---|
| ≅250 | 0x00 | 4 SPS |
| | 0x01 | 20 SPS |
| | 0x02 | 50 SPS |
| | 0x03 | 100 SPS |
| | 0x04 | 200 SPS |
| | 0x05 | 500 SPS |
| | 0x06 | 850 SPS |
| | 0x07 | 1300 SPS |

## Command-Set List

| TABLE 6 | | | | |
|---|---|---|---|---|
| TYPE | NAME | DESCRIPTION | TX PACKET STRUCTURE | RX PACKET STRUCTURE |
| Get | GSAI | Get client activity inquiry (Used to test communication) | 00 05 00 01 0E | 00 05 00 01 0E |
| *Get | GCCR | Get channel current reading | 00 06 00 05 00 20 | See Payload Example |
| Set | SSSS | Set system stream state OFF | 00 06 00 0C 00 3C | 00 05 00 0C 3A |
| *Set | SSSS | Set system stream state ON | 00 06 00 0C 01 41 | 00 05 00 0C 3A ... [Stream Bytes] |
| *Get | GDSN | Get device serial number | 00 05 01 00 0D | See Payload Example |
| *Get | GDMN | Get device model number | 00 05 01 01 11 | See Payload Example |
| *Get | GDIN | Get device item number | 00 05 01 02 15 | See Payload Example |
| *Get | GDHV | Get device hardware version | 00 05 01 03 19 | See Payload Example |
| *Get | GDFV | Get device firmware version | 00 05 01 04 1D | See Payload Example |
| *Get | GDFD | Get device firmware date | 00 05 01 05 21 | See Payload Example |
| *Get | GPSSN | Get profile sensor serial number | 00 06 03 00 00 15 | See Payload Example |
| *Get | GPSPR | Get profile sampling rate | 00 06 03 1E 00 8D | See Payload Example |
| Set | SPSPR | Set profile sampling rate 4SPS | 00 07 04 1E 00 00 92 | 00 05 04 1E 8E |
| Set | SPSPR | Set profile sampling rate 20SPS | 00 07 04 1E 00 01 98 | 00 05 04 1E 8E |
| Set | SPSPR | Set profile sampling rate 50SPS | 00 07 04 1E 00 02 9E | 00 05 04 1E 8E |
| Set | SPSPR | Set profile sampling rate 100SPS | 00 07 04 1E 00 03 A4 | 00 05 04 1E 8E |
| Set | SPSPR | Set profile sampling rate 200SPS | 00 07 04 1E 00 04 AA | 00 05 04 1E 8E |
| Set | SPSPR | Set profile sampling rate 500SPS | 00 07 04 1E 00 05 B0 | 00 05 04 1E 8E |
| Set | SPSPR | Set profile sampling rate 850SPS | 00 07 04 1E 00 06 B6 | 00 05 04 1E 8E |
| Set | SPSPR | Set profile sampling rate 1300SPS | 00 07 04 1E 00 07 BC | 00 05 04 1E 8E |
| *Get | GPLP | Get profile loading point value 0 | 00 07 03 18 00 00 77 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 1 | 00 07 03 18 00 01 7D | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 2 | 00 07 03 18 00 02 83 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 3 | 00 07 03 18 00 03 89 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 4 | 00 07 03 18 00 04 8F | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 5 | 00 07 03 18 00 05 95 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 6 | 00 07 03 18 00 06 9B | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 7 | 00 07 03 18 00 07 A1 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 8 | 00 07 03 18 00 08 A7 | See Calibration Point Payload Example |

| TABLE 6 | | | | |
|---|---|---|---|---|
| TYPE | NAME | DESCRIPTION | TX PACKET STRUCTURE | RX PACKET STRUCTURE |
| *Get | GPLP | Get profile loading point value 9 | 00 07 03 18 00 09 AD | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 10 | 00 07 03 18 00 0A B3 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 11 | 00 07 03 18 00 0B B9 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 12 | 00 07 03 18 00 0C BF | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 13 | 00 07 03 18 00 0D C5 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 14 | 00 07 03 18 00 0E CB | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 15 | 00 07 03 18 00 0F D1 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 16 | 00 07 03 18 00 10 D7 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 17 | 00 07 03 18 00 11 DD | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 18 | 00 07 03 18 00 12 E3 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 19 | 00 07 03 18 00 13 E9 | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 20 | 00 07 03 18 00 14 EF | See Calibration Point Payload Example |
| *Get | GPLP | Get profile loading point value 21 | 00 07 03 18 00 15 F5 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 0 | 00 07 03 19 00 00 7B | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 1 | 00 07 03 19 00 01 81 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 2 | 00 07 03 19 00 02 87 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 3 | 00 07 03 19 00 03 8D | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 4 | 00 07 03 19 00 04 93 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 5 | 00 07 03 19 00 05 99 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 6 | 00 07 03 19 00 06 9F | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 7 | 00 07 03 19 00 07 A5 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 8 | 00 07 03 19 00 08 AB | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 9 | 00 07 03 19 00 09 B1 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 10 | 00 07 03 19 00 0A B7 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 11 | 00 07 03 19 00 0B BD | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 12 | 00 07 03 19 00 0C C3 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 13 | 00 07 03 19 00 0D C9 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 14 | 00 07 03 19 00 0E CF | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 15 | 00 07 03 19 00 0F D5 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 16 | 00 07 03 19 00 10 DB | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 17 | 00 07 03 19 00 11 E1 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 18 | 00 07 03 19 00 12 E7 | See Calibration Point Payload Example |

| TABLE 6 | | | | |
|---|---|---|---|---|
| **TYPE** | **NAME** | **DESCRIPTION** | **TX PACKET STRUCTURE** | **RX PACKET STRUCTURE** |
| *Get | GPADP | Get profile analog-to-digital calibration value 19 | 00 07 03 19 00 13 ED | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 20 | 00 07 03 19 00 14 F3 | See Calibration Point Payload Example |
| *Get | GPADP | Get profile analog-to-digital calibration value 21 | 00 07 03 19 00 15 F9 | See Calibration Point Payload Example |
| *Get | GBTR | Get board temperature reading | 00 05 00 07 26 | See Payload Example |

*** Note:** The Payload bytes are located directly before the last byte of the packet which is the Checksum.

## Payload Example

The following transaction shows the response to the GDSN command (Get Device Serial Number). This command returns a 4-byte payload containing the device serial number.

TX: 00 05 01 00 0D
RX: 00 09 01 00 **00 01 E2 40** 49

Hex to decimal: 0x0001E240 → 123456

## Calibration Point Payload Example

The received packet payload follows the format defined in the Payload Example section above. Up to 11 calibration points are supported per direction. Point indexing is defined as follows:

- Point 0 is the positive direction offset.
- If the number of calibration points per direction is P
    - then the positive full-scale point is Point N, where N = P - 1.
    - The negative direction offset is Point N + 1.
    - The negative full-scale point is Point M, where M = (2 x P) - 1.

Example for a 2-point calibration:

- Point 0 = positive offset
- Point 1 = positive full-scale
- Point 2 = negative offset
- Point 3 = negative full-scale
- For this example, P = 2, therefore N = 1 and M = 3.

Example for a 5-point calibration:

- Point 0 = positive offset
- Point 4 = positive full-scale

- Point 5 = negative offset
- Point 9 = negative full-scale
- For this example, P = 5, therefore N = 4 and M = 9.

Each loading point value is stored as a 32-bit unsigned integer (4 bytes). The 4 bytes must be reassembled and interpreted as an IEEE 754 single-precision floating-point value. Refer to the example code in the FUTEK public GitHub repository.

### ADC Data Conversion

The following formula could be used to convert the raw ADC data:

$$CalculatedReading = \frac{[ADCValue - OffsetValue]}{[FullScaleValue - OffsetValue]} \times FullScaleLoad$$

Here are the variables:

- **ADCValue** = the most recent analog-to-digital conversion value.
- **OffsetValue** = the analog-to-digital conversion value stored during calibration that corresponds to the offset (zero physical load).
- **FullScaleValue** = the analog-to-digital conversion value stored during calibration that corresponds to the full scale (maximum physical load).
- **FullScaleLoad** = the numeric value stored during calibration for the maximum physical load.

### ADC Data Conversion Example (Direction 1, 2-point Calibration)

#### Calibration Data

Get profile analog-to-digital calibration value 0 (Direction 1) *[GPADP]:*

Hex to decimal: `0x81B320` → `8,500,000`

Get profile analog-to-digital calibration value 1 (Direction 1) *[GPADP]:*

Hex to decimal: `0xB71B00` → `12,000,000`

Get channel current reading *(GCCR):*

Hex to decimal: `0x989680` → `10,000,000`

Calculation

$OffsetValue$ = 8,500,000

$FullScaleValue$ = 12,000,000

$FullScaleLoad$ = 20g *(Accessible via the calibration certificate or directly from onboard flash memory)*

$$CalculatedReading = \frac{[10000000 \; - \; 8500000]}{[12000000 \; - 8500000]} \times 20g = \; 8.5714g$$

**Temperature Conversion**

$$Output\ (mV) \; = \; 1200 - [\frac{16777215 \; - ADCValue}{6990.506666666667}]$$

$$Temperature\ (°C) = \; -40 + \left[\frac{Output - 80}{0.28}\right]$$

**Temperature Conversion Example**

$ADCValue$*: Get Board Temperature (GBT):* 9095859 (0x8ACAB3)

$$1200 - \left[\frac{16777215 \; - 9095859}{6990.506666666667}\right] = \; 101.1733\ (mV)$$

$$-40 + \left[\frac{101.1733 \; - 80}{0.28}\right] = \; 35.6\ (°C)$$

## Checksum Calculation

Last byte <CHS> carries information from the first three bytes

`<HSB> <MSB> <LSB> <ChS>`

Let's assume:

`HSB = 0x0A`

`MSB = 0x0B`

`LSB = 0x0C`

Then `<ChS>` byte could be calculated as:

`ChS = (0x0A * 1) + (0x0B * 2) + (0x0C * 3) = 0x44`

The received packet will then be:

`<0x0A> <0x0B> <0x0C> <0x44>`

If the `<ChS>` calculation exceeds a byte then the `<LSB>` byte is the only valid byte, for example:

`0x2FD` which includes two bytes, then the `<ChS>` will be calculated as `0xFD`

## Checksum Conversion Example

The subsequent illustration demonstrates the checksum calculation method for the **SPSPR** (**Set Profile Sampling Rate 4SPS**) command, applicable to both **RX** and **TX** buffers. Similar logic may be applied for computing the checksum of other commands.

TX: `00 07 04 1E 00 00` **`92`**
RX: `00 05 04 1E` **`8E`**

Checksum calculation for TX buffer:
(0*1) + (07*2) + (04*3) + (30*4) + (0*5) + (0*6) = 146 (`0x92`)

Checksum calculation for RX buffer:
(0*1) + (05*2) + (04*3) + (30*4) = 142 (`0x8E`)

## Firmware Revision

| QIA128 | |
|---|---|
| **Revision** | 7.0.0 |
| **Release Date** | 09/19/2023 |
| **Hardware Compatibility** | REV002 |
| **Notes** | **New Features**<br>• Added support for hardware revision HW002<br>**Changes**<br>• N/A<br>**Fixes**<br>• • N/A |

| IDC150/IEM100 | |
|---|---|
| **Revision** | 1.0.0 |
| **Release Date** | 04/10/2026 |
| **Hardware Compatibility** | REV003 |
| **Notes** | **New Features**<br>• N/A<br>**Changes**<br>• The model number has been changed from IDC150 to IEM100. The firmware and hardware item numbers remain unchanged<br>**Fixes**<br>• N/A |