# Model JI-210
# I2C Bus Monitor

## User Manual

***Jupiter Instruments***

12/14/2005 Edition

TABLE OF CONTENTS

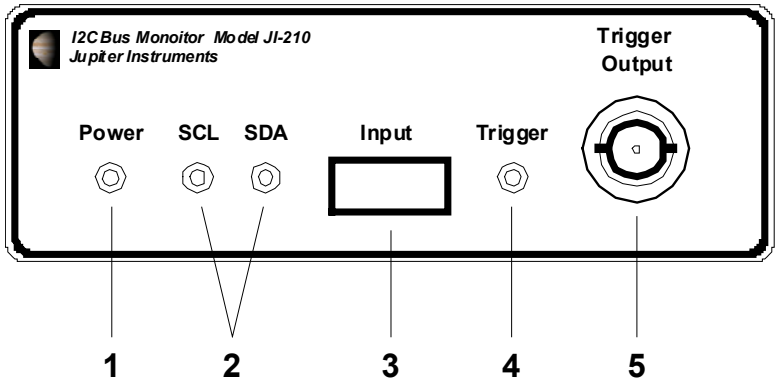
## 1. INTRODUCTION

The Model-210 is a PC hosted I2C bus monitor used to non-invasively monitor and troubleshoot a variety of I2C networks. The desktop unit captures and stores up to 32KBs of bus activity including message data and address values, stop/start, read/write, ACK/NACK, and frame error events. A menu-driven Windows application manages captured I2C messages, trigger setup, and recording settings. RS-232 provides the PC to desktop unit communications link.

**Features**

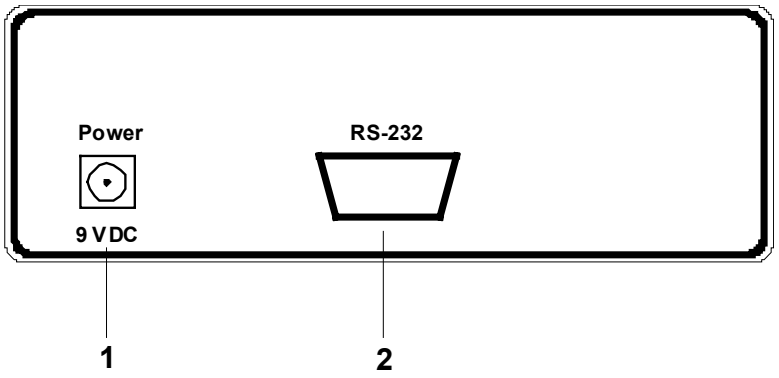
- Compatible with Standard (100kbit/s), Fast (400kbit/s), and bits rates up to 700kbit/s.
- Supports 7-bit and 10-bit addressing formats
- Compliant with 3.3V and 5.0v bus networks
- Combinational Triggering based on Address, Data, and R/W values, and Frame Error events.
- Selectable pre-trigger memory level
- Message period timestamp
- Trigger event output signal via BNC connector

### 1.1 Front Panel Description




1. **Power** – Power on indicator.
2. **SCL, SDA** – I2C bus activity indicators.
3. **Input** – I2C probe cable jack input. 5-pin, polarized, latching header (Molex 70553-0004)
4. **Trigger** – Trigger event / Recording indicator. Blinking LED = Recording in progress, Solid LED = Trigger event has occurred.
5. **Output** – Trigger output signal. 3V, 20us pulse, BNC connector.

## 1.2 Rear Panel Description




1. **Power** – Power input jack, 2.1mm x 5.5mm
2. **RS-232** – Serial Communications, RS-232, Sub-D, 9-pin, Female

## 2. GETTING STARTED

### 2.1 Software Installation

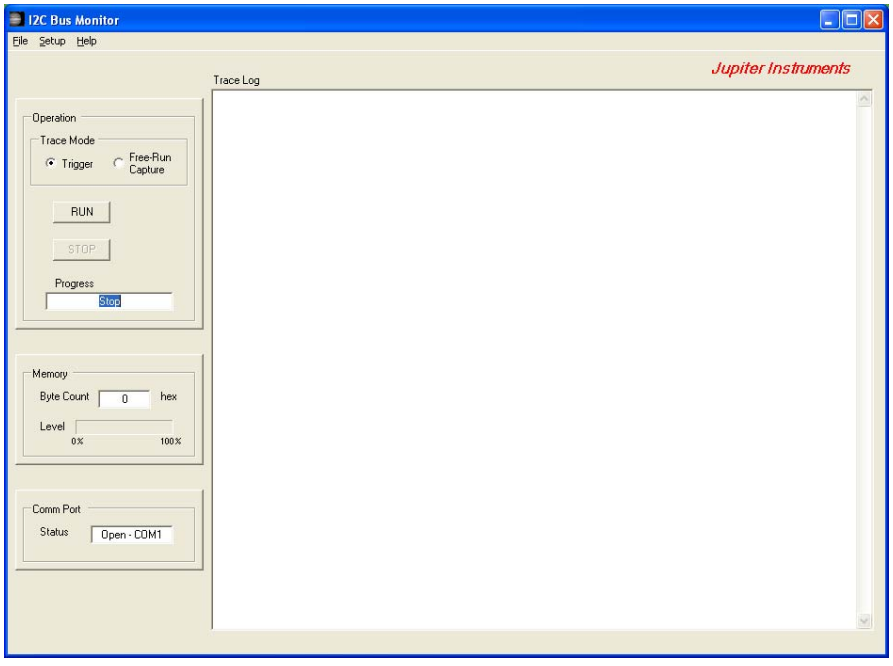
1. Insert the CD-ROM into the CD drive and run **Setup.exe**.
2. Follow the on screen instructions.
3. At the conclusion of the setup, a default folder will be created (**C:\Program file\Jupiter Instruments**) containing the main executable (**I2C_Bus_Mon.exe**.).

### 2.2 Hardware Setup

1. Connect the I2C bus Monitor unit to the host PC using the 9-pin serial cable.
2. Connect the AC adapter to the power-jack on the rear panel of the I2C Bus Monitor unit, and then plug the adapter into the wall.
3. Ensure that the power-on LED on the I2C Bus Monitor unit is illuminated.

### 2.3 Communications Check

1. Go to the folder **C:\Program file\Jupiter Instruments**
2. Launch the I2C Bus Monitor application by clicking **I2C_Bus_Mon.exe**.
3. Verify that the main I2C Bus Monitor window appears:

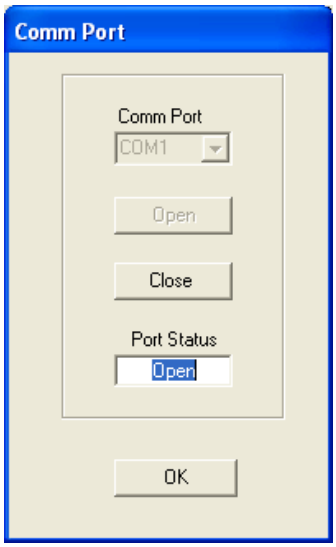



If an error occurs and the window does not appear, begin by verifying that the .NET Framework 1.1 is installed. To do this, click **Start** on your windows desktop, select **Control Panel**, and then double-click the **Add or Remove Programs** icon. When the window appears, scroll through the list of applications. If you see the .NET Framework 1.1 listed, the latest version is installed. If not

listed, go to *msdn.microsoft.com/net**framework**/ downloads/howtoget.aspx* for instructions on downloading and installing the latest .NET Framework version.

4. At the menu bar, configure the COM port by selecting "**Setup**" then "**Comm Port**".

   The Comm Port menu opens.




5. Select the desired host COM port number (**COM 1 – COM4**), and then select "**Open**".
6. Ensure that the selected port is available and open by verifying an "Open" Port Status.
7. Click "**OK**" to save.
8. In the Operations group at the main window, click the "**Run**" button. Wait a few seconds, then click "**Stop**"
9. Verify an output similar to that shown below is now displayed in the "Trace Log" pane.

```
I2C Bus Monitor Application  Version 1.0  Date: 10/07/2004
Hardware Version 0X!

11/11/2004 8:48:34 PM
Memory count = 0000!

No Data!
```

## 2.4 I2C Message Capture w/o Trigger

1. Ensure the I2C bus Monitor unit is connected to the host PC.
2. Ensure that the I2C Bus Monitor application is running and a host COM port has been selected.
3. Connect the I2C probe cable to the Input jack on the I2C bus Monitor front panel.
4. Connect the three I2C bus probes (SDA, SCL, and GND) to an active I2C bus.

5. Verify I2C bus activity by observing the SDA and SCL LEDs on the front panel of the I2C Bus Monitor unit.
6. At the main window, select "**Free-Run Capture**" in the Trace Mode group.
7. Click the "**Run**" button in the operations group to begin capturing I2C bus messages.
8. Verify I2C message recording by observing an increasing byte count in the "Record Memory" text box and the message "Recording" in the "Progress" text box.
9. Click the "**Stop**" recording button after filling memory with a few hundred bytes.
10. Captured I2C bus messages are now displayed in the "Trace Log" pane.

## 2.5 I2C Message Capture with Trigger

1. Begin by re-running the steps in section 2.4 "I2C Message Capture w/o Trigger."
2. Scroll through the captured I2C messages in the "Trace Log" pane to locate a message with an infrequent occurrence suitable for use as a trigger. For this test, select a message with a 7-bit address, data argument, and without "Frame Errors." Note the address, direction (R/W), and data values.
3. At the main window menu bar, open the trigger menu by selecting "**Setup**" then "**Trigger**."
4. Set the trigger menu as follows:
   - **Address Format:** 7-bit
   - **Trigger Configuration:** Address and Data
   - **Address Value:** Trigger Address value
   - **Data Value:** Trigger Data value
   - **R/W Value:** Trigger R/W value
   - **Trigger Output:** Disabled
5. Click "**OK**" to save the trigger setup.
6. At the main window, select "**Trigger**" in the Trace Mode group.
7. Verify I2C bus activity by observing the SDA and SCL LEDs on the front panel of the I2C Bus Monitor unit.
8. At the main window, click the "**Run**" button in the operations group to begin capturing I2C bus messages.
9. Verify I2C message recording by observing an increasing byte count in the "Record Memory" text box.
10. Watch the "Progress" text box for the "Triggered" message.
11. Click the "**Stop**" recording button.
12. At the "Trace Log" pane, scroll through the captured messages to locate the trigger message. This message is identified by an asterisk preceding the message number and horizontal bars above and below the message. As an example:

```
Msg 005   <Start>[Add:A1(R)]<ACK>[#12]<ACK>[#23]<ACK>[#56]<ACK>[#78]<ACK>[#9A]
          <ACK>[#BC]<ACK>[#DE]<ACK>[#F0]<ACK>[#11]<NACK><Stop>                t=109869us

Msg 006   <Start>[Add:A6(W)]<ACK>[#00]<ACK><Stop>                             t=1202us
```

---

* Msg 007 <Start>[Add:A7(R)]<ACK>[#22]<ACK>[#23]<ACK>[#24]<ACK>[#25]<ACK>[#26]
<ACK>[#27]<ACK>[#FF]<ACK>[#FF]<ACK>[#FF]<NACK><Stop> t=109361us

---

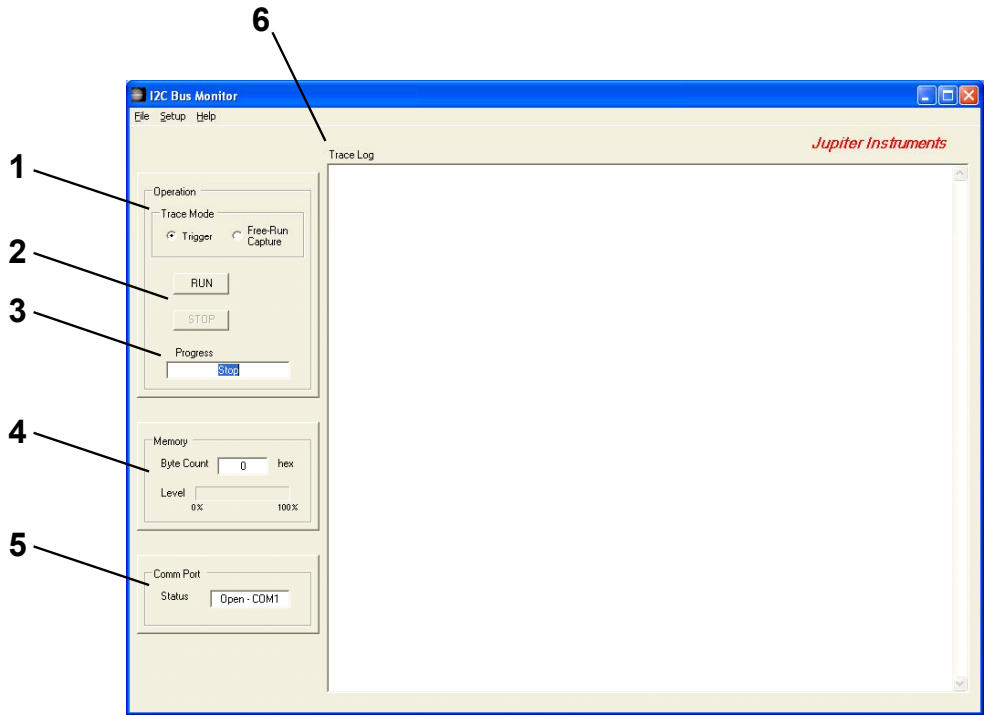
Msg 008 <Start>[Add:A0(W)]<ACK>[#00]<ACK><Stop> t=1202us

Msg 009 <Start>[Add:A1(R)]<ACK>[#12]<ACK>[#23]<ACK>[#56]<ACK>[#78]<NACK><Stop> t=108373us

13. Examine the captured trigger message and verify a match with the trigger menu selections.

## 3. MAIN WINDOW AND MENU DESCRIPTIONS

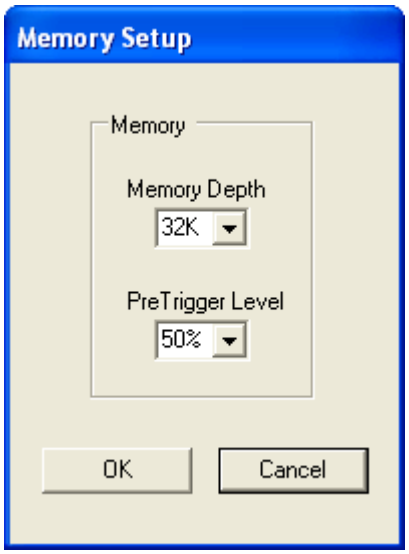
### 3.1 Main Window




1. **Trace Mode** – I2C messages can be captured via two modes: **Trigger** and **Free-Run Capture**. In the Trigger mode, messages are captured and stored in a circular buffer. Until a trigger event, I2C messages are stored in memory up to the pre-fill level, replacing old messages with new using a system of incrementing, memory pointers. When a trigger occurs, the tail pointer stops incrementing, and the head continues to fill the remainder of memory. The result is memory filled with pre and post trigger I2C messages. In the Free Run Capture mode, all triggering is ignored. I2C messages are simply captured and written to memory, beginning at address 0 and ending at the selected memory depth. The depth of the buffer and the pre-fill level is selected by way of the Memory Setup menu.
2. **Run/Stop** – These buttons control the I2C message capture session. The **Run** button initiates a session and the Stop button terminates it. A capture session will automatically stop when the memory is full.
3. **Progress** – Messages reporting the status of the capture session are displayed here.
4. **Memory** – The memory byte count is displayed both numerically and graphically.
5. **Comm Port** – The current status of the selected host COM port is displayed here.
6. **Trace Log** – Captured I2C messages are displayed in this pane.

### 3.2 Memory Setup Menu

This menu selects both the “Memory Depth” and the number of bytes filled before a trigger event occurrence (“PreTrigger Level”).

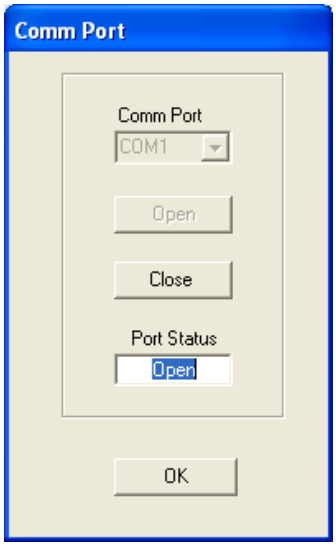



Since the bus monitor will automatically display the I2C messages captured when the memory is full, it is convenient to set the memory depth as small as necessary to take advantage of this feature. Otherwise, the memory should be set to the default value of 32K.

The “PreTrigger Level” is adjustable in three steps from 25% to 75% of the selected memory depth. By judicious selection of the “PreTrigger Level”, viewing a particular I2C message or bus event long before or after a trigger event occurrence is possible.

## 3.3 Comm Port Menu

The selection of a COM port number (**COM1 – COM4**) and status (**Open/Closed**) is provided by this menu.

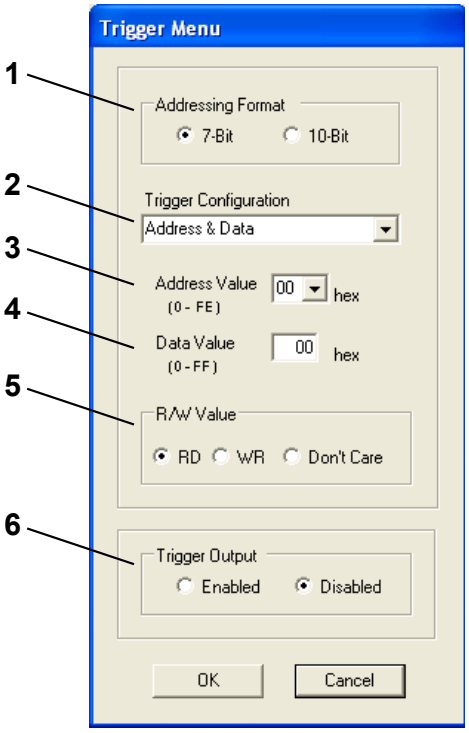



Four COM port numbers are available: COM1 (3F8h), COM2 (3F8h), COM3 (3F8h), or COM4 (3F8h). Click the "**Open**" or "**Close**" button to either open or close the selected port number. Status of the selected port is indicated in the "Port Status" field. A pop-up message warns against a nonexistent or unavailable COM port.

*Note: If the host PC does not have an available COM port, but does have a free USB port, a USB to RS-232 converter can be used to communicate with the desktop unit. Successful operation has been achieved with the IOGear Model GUC232A USB to Serial/PDA converter cable.

## 3.4 Trigger Menu

This menu proves a simplified and convenient means of entering and maintaining a trigger setup. A trigger event is generated if a match occurs between a captured I2C bus message and the trigger setup values. Only a single trigger event is generated during a recording session.




1. **Address Format** – Radio buttons select either a **7-Bit** or **10-Bi**t address. If the type of trigger is not based on an address (such as "Frame Error Only**"**), the buttons are dimmed and not selectable.
2. **Trigger Configuration** – A list box selects 1 of 7 trigger configurations:

| Configuration | Match Variables |
|---|---|
| • **Address and Data** | Address, Data, Address Format, and R/W |
| • **Address only** | Address, Address Format, and R/W |
| • **Data only** | Data and R/W |
| • **(Address & Data) or Frame Error** | (Address, Data, Address Format, and R/W) or Frame Error |
| • **Address or Frame Error** | (Address, Address Format, and R/W) or Frame Error |
| • **Data or Frame Error** | (Data and R/W) or Frame Error |
| • **Frame Error Only** | Frame Error Only |

3. **Address Value** – Text selects values for both 7-bit and 10-bit address formats. When 7-bit addressing is selected, even address values from 0 to FEh are available. This follows the I2C convention using the upper 7-bit of a byte as the address and setting the LSB to 0. For 10-bit addressing, all values from 0 to 3FFh are available.
4. **Data Value** – Value range is from 0 to FFh.
5. **R/W Value** – Radio buttons select read (**RD**), write (**WR**), or a **Don't Care** value for the I2C message direction bit (R/W).
6. **Data Value** – Value range is from 0 to FFh.
7. **Trigger Output** – The "Trigger Output" enables or disables the generation of 10us trigger event signal. This output signal is available at front panel BNC connector. Note that only a single trigger event is generated during a recording session.

## 4. TRACE LOG PANE

Captured I2C messages are assigned a message number and sequentially displayed in the trace log pane. Displayed message details include message data and address values, stop/start, read/write, ACK/NACK, and frame error events. Message execution time (Start to Stop or Start to repeated Start) is also displayed. An asterisk preceding the message number and horizontal bars above and below the message identifies a trigger message.

### 4.1 Message Syntax Definitions

| Syntax | Description |
|---|---|
| <Start> | Start event |
| <Stop> | Stop event |
| <ACK> | Acknowledge event |
| <NACK> | not-Acknowledge event |
| [Add:*hh*(*d*)] | 7-bit Message Address with direction.<br>*hh* Address value displayed in conventional I2C hexadecimal format (i.e. upper 7-bits make up the slave address and the LSB determines message direction)<br>*d* Message direction: R = Read, W = Write. |
| [Add10:*hh*(*d*)]<br>followed by<br>[Add10:*hh*] | 10-bit Message Address with direction.<br>Since a 10-bit address is formed from the first two bytes following a Start condition, the 10-bit address syntax is displayed as two bytes: the upper 2 bits of the 10-bit address make up the first byte value, and the remaining 8 bits of the 10-bit address make up the second byte value. Message direction is displayed in parentheses in the first byte.<br>*hh* Address displayed as hexadecimal value<br>*d* Message direction: R = Read, W = Write. |
| [#*hh*] | Data byte value<br>*hh* Data displayed as hexadecimal value. |

| | |
|---|---|
| <Frame Error!> | An address or data value has been truncated by a Stop condition. |
| <ACK/NACK Error!> | A stop condition has terminated the message before an ACK/NACK is generated. |
| T=*ddddd*uS | Message execution time measured from Start to Stop or Start or repeated Start.<br>*ddddd* Time displayed as a decimal value. |
| *Msg | An asterisk preceding the message number identifies a trigger message. Additionally, horizontal bars above and below the message body are added. A trace log example follows below: |

Msg 005 <Start>[Add:A1(R)]<ACK>[#12]<ACK>[#23]<ACK>[#56]<ACK>[#78]<ACK>[#9A]
<ACK>[#BC]<ACK>[#DE]<ACK>[#F0]<ACK>[#11]<NACK><Stop> t=109869us

Msg 006 <Start>[Add:A6(W)]<ACK>[#00]<ACK><Stop> t=1202us

---

* Msg 007 <Start>[Add:A7(R)]<ACK>[#22]<ACK>[#23]<ACK>[#24]<ACK>[#25]<ACK>[#26]
<ACK>[#27]<ACK>[#FF]<ACK>[#FF]<ACK>[#FF]<NACK><Stop> t=109361us

---

Msg 008 <Start>[Add:A0(W)]<ACK>[#00]<ACK><Stop> t=1202us

Msg 009 <Start>[Add:A1(R)]<ACK>[#12]<ACK>[#23]<ACK>[#56]<ACK>[#78]<NACK><Stop> t=108373us

**APPENDIX A**

1. PC System Requirements

   - Microsoft Windows 98*, 2000*, and XP

     *These OS require the .NET framework. If installation is required, go to *msdn.microsoft.com/net**framework**/ downloads/howtoget.aspx*

   - Serial port, 115.2K BAUD
   - CD ROM drive
   - 5 MB hard disk space

2. Size and Weight

   - 4.2x1.5x6.5 inches
   - 1.0 lbs.

## APPENDIX B

### 1. Specifications

**Power**

| | |
|---|---|
| Input Power: | 6VDC @100mA |
| Receptacle: | 2.1mmx5.5mm coaxial DC jack |

**PC Interface**

| | |
|---|---|
| Communications: | Serial RS-232C |
| BAUD rate: | 115.2K |
| Connector: | 9-pin, sub-D, female |

**I2C Interface**

| | |
|---|---|
| Connector: | 5-pin latching (0.100” spacing, 0.025” pins) |
| Pins: | Pin 1 = Gnd<br>Pin 2 = SCL (Red)<br>Pin 3 = Gnd (Black)<br>Pin 4 = SDA (Green)<br>Pin 5 = Gnd |
| Bus Speed: | 0 to 1.0Mbit/s (w/o bus errors)<br>0 to 700kbit/s (with bus errors) |
| Input Threshold: | $V_{IL}$: 1.0V (typ.)<br>$V_{IH}$: 2.0V (typ.)<br>$V_{H}$: 0.8V (typ.) |
| Input Range: | -0.3V to 5.5V (operational)<br>-5.0 to +10V (max. rating) |
| Input Capacitance: | 35pf – typical (including 34” probe cable) |

**Trigger Output**

| | |
|---|---|
| Connector: | BNC |
| Output Signal: | 10uS, 3.3V pulse |

**LEDs**

| | |
|---|---|
| Power: | Power-On |
| SCL: | Bus clock activity |
| SDA: | Bus data activity |
| Trigger: | Message Recording / Trigger Event |

**Message Recording**

| | |
|---|---|
| Capacity: | 32Kb buffer |
| Triggering: | Address, Data, R/W, Frame Error combination |
| Pre-Trigger: | Selectable 25%, 50%, and 75% pre-trigger recording |
| Message Time Stamp: | 1us to 1.0485s, 1us resolution. Period measured from Message Start to Stop, or Start to repeated Start. |

## APPENDIX C

1. General Information

### 1.1 Warranty

The equipment is warranted for one year from data of purchase against defects in materials or workmanship. Jupiter Instruments reserves the right to repair or replace products at its own and complete discretion. Customer must obtain from Jupiter Instruments a Return Authorization Number (RMA) prior to returning any products to Jupiter Instruments. Products returned under this Warranty must be unmodified and in original packaging. Jupiter Instruments reserves the right to refuse warranty repairs or replacements for any products that are damaged or not in original form.

The customer is responsible for the shipping and insurance cost arising from the return of products to Jupiter Instruments. Jupiter Instruments will return all in-warranty products with shipping cost prepaid.

### 1.2 Thirty-Day Return Policy

Customers may return Jupiter Instruments products for a full refund if Jupiter Instruments is contacted within thirty days of the customer's receipt of the product. Customer may return Jupiter Instruments products for credit, exchange, or a refund. Customer must obtain form Jupiter Instruments a Return Authorization Number (RMA) prior to returning any products to Jupiter Instruments. Products must be returned unmodified and in original packaging. Jupiter Instruments reserves the right to refuse return rights for any products that are damaged or not in original form. Volume orders may be subject to a significant restocking fee.

### 1.3 Limitation of Liability

Jupiter Instruments' liability shall be limited to the repair or replacement of defective products in accordance with the Jupiter Instruments limited warranty.

Jupiter Instruments shall not be liable for any incidental, special or consequential damages for breach of any warranty, expressed or implied, directly or indirectly arising out of Jupiter Instruments' sale of merchandise, including any failure to deliver any merchandise, or arising out of customer's installation or use, whether proper or improper, of the product, separately or in combination with other equipment, or from any other cause. Use the JI-210 at your own risk.

Products sold by Jupiter Instruments are not authorized for use as critical components in life support devices or systems.

### 1.4 Contact Us

- Address: Jupiter Instruments
  25108 Marguerite Pkwy Suite, B #109
  Mission Viejo, CA 92692-2400
- Email: Sales@Jupiteri.com
  Tech@Jupiteri.com
- Phone: Sales and Information: (949)-716-0154
- Website www.Jupiteri.com