3-Space Sensor

3-Space Sensor Embedded Evaluation Kit

Ultra-Miniature Attitude & Heading Reference System

User's Manual

YEI Technology
630 Second Street
Portsmouth, Ohio 45662

www.YeiTechnology.com
www.3SpaceSensor.com

3-Space Sensor Embedded Evaluation Kit

Ultra-Miniature Attitude & Heading Reference System

User's Manual

YEI Technology
630 Second Street
Portsmouth, Ohio 45662

www.YeiTechnology.com
www.3SpaceSensor.com

Toll-Free: 888-395-9029
Phone: 740-355-9029

Table of Contents

1. Usage/Safety Considerations

1.1 Usage Conditions

- Do not use the 3-Space Sensor in any system on which people's lives depend(life support, weapons, etc.)
- Because of its reliance on a compass, the 3-Space Sensor will not work properly near the earth's north or south pole.
- Because of its reliance on a compass and accelerometer, the 3-Space Sensor will not work properly in outer space or on planets with no magnetic field.
- Care should be taken when using the 3-Space Sensor in a car or other moving vehicle, as the disturbances caused by the vehicle's acceleration may cause the sensor to give inaccurate readings.
- Because of its reliance on a compass, care should be taken when using the 3-Space Sensor near ferrous metal structures, magnetic fields, current carrying conductors, and should be kept about 6 inches away from any computer screens or towers.
- The YEI 3-Space Embedded module and Embedded Evaluation Kit contain components that are sensitive to electro- static-discharge. Care should be taken when handling the module or evaluation kit board.
- PCB layout can affect the performance of the 3-Space Embedded module. Placing magnetic components, ferrous metal containing components, high-current conductors, and high-frequency digital signal lines should be avoided during PCB layout.
- The 3-Space Embedded Evaluation Kit contains a special socket for connecting 3-Space Embedded modules. Ensure that power is removed from the board when inserting or removing 3-Space Embedded modules.

1.2 Technical Support and Repairs

YEI provides technical and user support via our toll-free number (888-395-9029) and via email (support@YostEngineering.com). Support is provided for the lifetime of the equipment. Requests for repairs should be made through the Support department. For damage occurring outside of the warranty period or provisions, customers will be provided with cost estimates prior to repairs being performed.

2. YEI 3-Space Embedded Evaluation Kit Overview

2.1 Introduction

The YEI 3-Space Sensor™ Embedded Evaluation Kit (TSS-EEVK) is a special interface board designed to simplify interfacing and experimenting with the 3-Space Sensor Embedded module AHRS systems . The TSS-EEVK provides the following features:

- Special socket accepts 3-Space Sensor Embedded modules without soldering or modification
- USB 2.0 interface
- RS-232 interface.
- Serial SPI interface emulation.
- Break out pads for all 3-Space Sensor Embedded module pins.
- Jumper configurable power options allow USB power, external adapter power, pin-pad power.
- Jumper configurable RS232 communication routing.
- Power switch.
- Mounting holes and threaded standoffs ease mechanical fastening.
- Status LEDs.

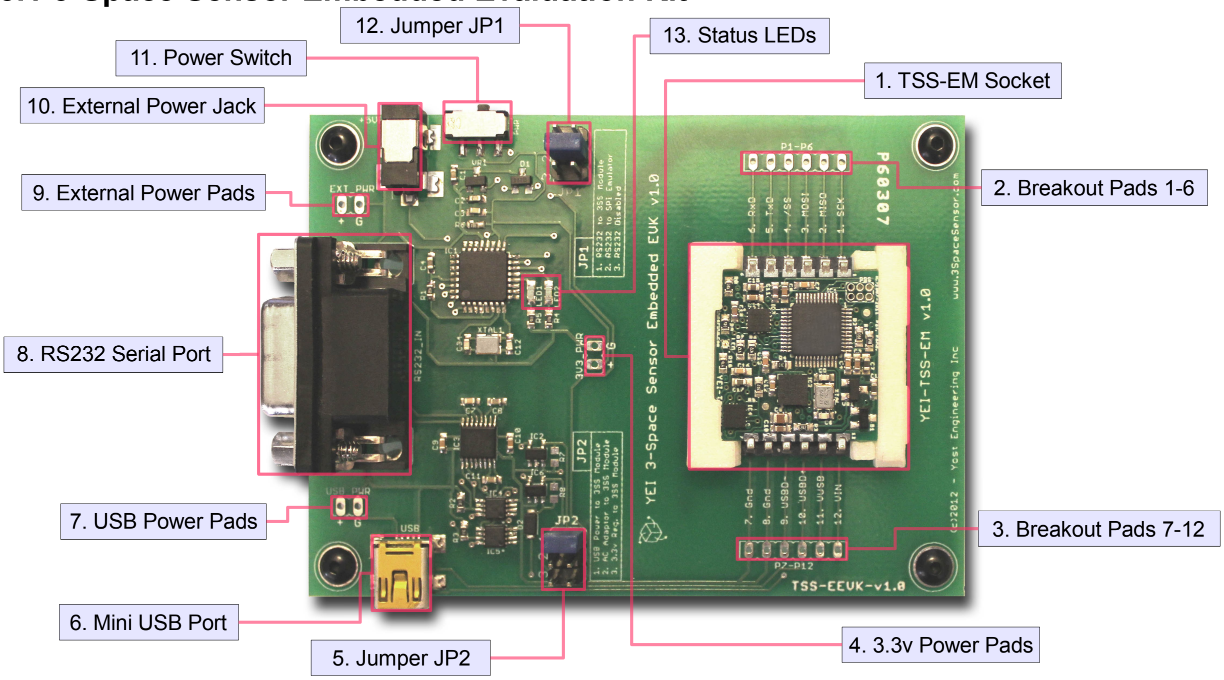
The YEI 3-Space Sensor Embedded Evaluation Kit is a packaged as a 94mmx67mmx152mm circuit board with a special carrier socket for installation of a 3-Space Sensor Embedded module. The board includes mounting holes and threaded standoffs to allow the TSS-EEVK board to be mechanically fastened to a target system during evaluation.

2.2 Applications

- Robotics
- Motion capture
- Positioning and stabilization
- Vibration analysis
- Inertial augmented localization
- Personnel / pedestrian navigation and tracking
- Unmanned air/land/water vehicle navigation
- Education and performing arts
- Healthcare monitoring
- Gaming and motion control
- Accessibility interfaces
- Virtual reality and immersive simulation

2.3 Hardware Overview

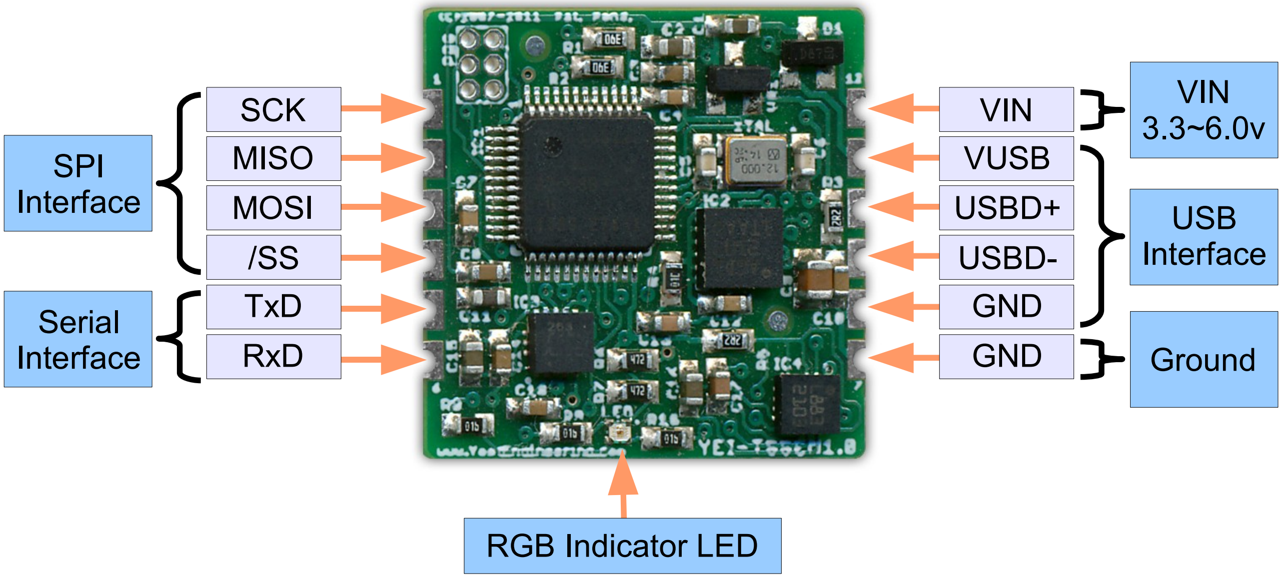
2.3.1 3-Space Sensor Embedded Evaluation Kit



1. **TSS-EM Socket** – This socket securely holds a TSS-EM module without requiring soldering.
2. **Breakout Pin Pads 1-6** – These pads are directly connected to pins 1 through 6 of the TSS-EM installed in the socket. These pins can be soldered to or contacted with a probe for testing and measurement purposes.
3. **Breakout Pin Pads 7-12** – These pads are directly connected to pins 7 through 12 of the TSS-EM installed in the socket. These pins can be soldered to or contacted with a probe for testing and measurement purposes.
4. **3.3v Power Pads** – These pads can be soldered to or probed to access the regulated +3.3v on-board power supply signals.
5. **Jumper JP2** – Jumper JP2 is used to select the power source that is routed to the TSS-EM module.
6. **Mini USB Port** – This is a 5-pin mini USB connector that is used to connect the TSS-EEVK to a computer via USB. The USB connector provides for both power and communication signals.
7. **USB Power Pads** – These pads can be soldered to or probed to access the +5v USB power supply signals.
8. **RS232 Serial Port** – This is a 9-pin DSUB RS-232 serial connector that can be used to connect to a PC serial port via a straight-through 9-pin serial cable. The position of jumper JP1 affects the routing of the RS232 communication signals.
9. **External Power Pads** – These pads can be soldered to or probed to access the +5v external power supply signals.
10. **External Power Jack** – This jack can be used to power the development board via an external +5v AC adapter. The jack is type EIAJ-1 (0.70mm ID, 2.35mm OD) and is commonly available on cell phone chargers.
11. **Power Switch** – This switch can be used to switch the power to both the TSS-EM module and TSS-EEVK board on and off.
12. **Jumper JP1** – Jumper JP1 is used to select the routing of the RS-232 serial communication signals.
13. **Status LEDs** – These LEDs are used to indicate information. LED1 is blue and indicates power status of the EEVK board. LED2 is orange and indicates serial communication mode of the TSS-EEVK.

2.3.2 3-Space Sensor Embedded Module

The YEI 3-Space Sensor Embedded module is a packaged as a 23mmx23mmx2.2mm castellated edge SMT module. Alternatively, the module can be through-hole mounted by adding standard 0.1" header strips to the castellated edge pads.



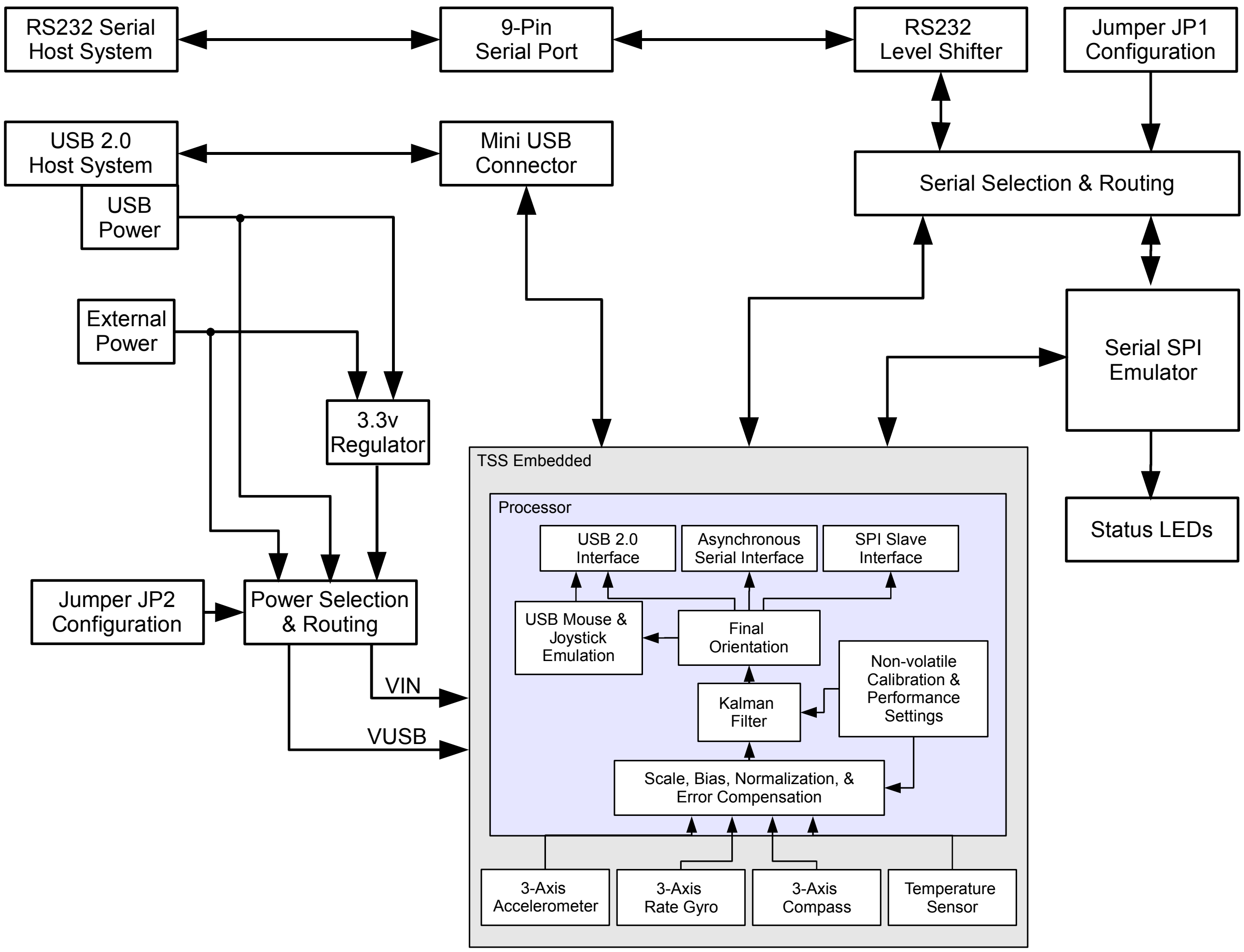
3-Space Sensor Embedded Pin Functions

Pad Number	Signal Name	Description
1	SCK	SPI Serial Clock. Input to Module.
2	MISO / INT	SPI Master In Slave Out. Output from Module. Can be configured to act as filter update Interrupt.
3	MOSI	SPI Master Out Slave In. Input to Module.
4	/SS	SPI Slave Select. Active Low Input to Module.
5	TxD / INT	UART Asynchronous Transmit Data. Output from Module. Can be configured to act as filter update Interrupt.
6	RxD	UART Asynchronous Receive Data. Input to Module.
7	GND	Ground. Only one ground pad must be connected.
8	GND	Ground. Only one ground pad must be connected. Commonly connected to USB supply ground.
9	USBD-	USB Data Minus. Only requires connection during USB mode use.
10	USBD+	USB Data Plus. Only requires connection during USB mode use.
11	VUSB	+5v USB Power Supply Input . Only requires connection during USB mode use.
12	VIN	Voltage Input +3.3v ~ +6.0v. Only required when USB power is not being used.

For convenience of interconnection or probing all TSS-EM module pin signals are available on the TSS-EEVK breakout pads.

For full documentation and complete specifications pertaining to the TSS-EM module, refer to the 3-Space Sensor Embedded User's Manual and other documentation which can be found on-line at: http://www.3SpaceSensor.com

2.4 Block Diagram of Evaluation Kit Operation



2.5 Electrical Characteristics

2.5.1 Absolute Maximum Ratings*

Operating Temperature …... -40C ~ 85C (-40F ~ 185F)
Storage Temperature …... -60C ~ 150C (-76F ~ 302F)
Supply Voltage on TSS-EM VIN Pin with respect to Ground …................... -0.3v ~ 6.5v
Supply Voltage on TSS-EM VUSB Pin with respect to Ground …............… -0.3v ~ 6.5v
Supply Voltage on TSS-EEVK USB Port with respect to Ground …............ -0.3v ~ 6.5v
Supply Voltage on TSS-EEVK EXT-PWR Jack with respect to Ground ...... -0.3v ~ 6.5v
Voltage on I/O Pins with respect to Ground …... -0.3v ~ 5.5v
Current Sink/Source from I/O pins …... -4mA ~ +4mA

* NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may adversely affect device reliability.

2.5.2 DC Characteristics

The following characteristics are applicable to the operating temperature range: TA = -40°C to 85°C

Symbol	Parameter	Min.	Typ.	Max.	Units
V_{IN}	Operating Supply Voltage on TSS-EM VIN pin	3.2	3.3	6.0	V
V_{USB}	Operating Supply Voltage on TSS-EM VUSB pin	3.8	5.0	6.0	V
V_{EXTP}	Operating Supply Voltage on TSS-EEVK EXT_PWR Port	3.2	3.3	6.0	V
V_{USBP}	Operating Supply Voltage on TSS-EEVK USB Port	3.8	5.0	6.0	V
V_{IL}	Input Low-level Voltage	-0.3		+0.8	V
V_{IH}	Input High-level Voltage	2.0		5.5	V
V_{OL}	Output Low-level Voltage			0.4	V
V_{OH}	Output High-level Voltage	2.6			V
I_{OL}	Output Low-level Current			-4	mA
I_{OH}	Output High-level Current			4	mA
C_{IN}	Input Capacitance			7	pF
I_{ACT}	Active Current Consumption		45	60	mA

2.5.3 USB Characteristics

The on-chip USB interface complies with the Universal Serial Bus (USB) v2.0 standard. All parameters related to the USB interface can be found within the USB 2.0 electrical specifications.

2.5.4 RS232 Serial Characteristics

The on-chip RS232 Serial interface is compatible with any RS232 compliant serial interface. The interface utilizes a minimum-wire configuration of three wires: a TxD serial output and an RxD serial input and a ground wire.

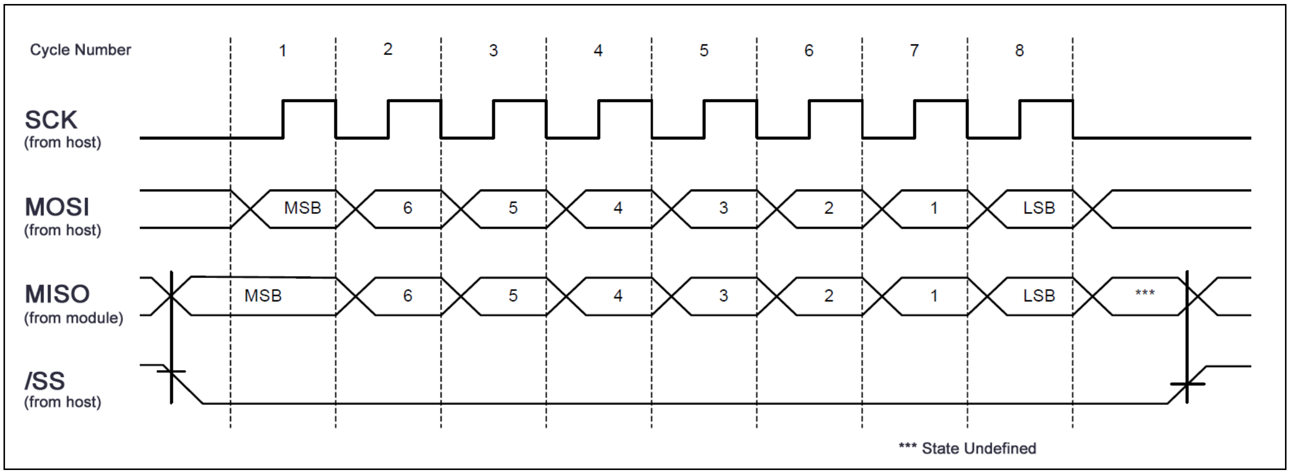
The TSS-EEVK, when used in Serial SPI Emulation mode, supports communication settings of 8N1 (8 data bits, no parity, 1 stop bit) format and a baud rate of 115200.

The TSS-EM module uses asynchronous serial communication settings of 8N1 (8 data bits, no parity, 1 stop bit) format and supports the following standard baud rates: 1200, 2400, 4800, 9600, 19200, 28800, 38400, 57600, 115200, 230400, 460800, 921600. The factory default baud rate is 115200.

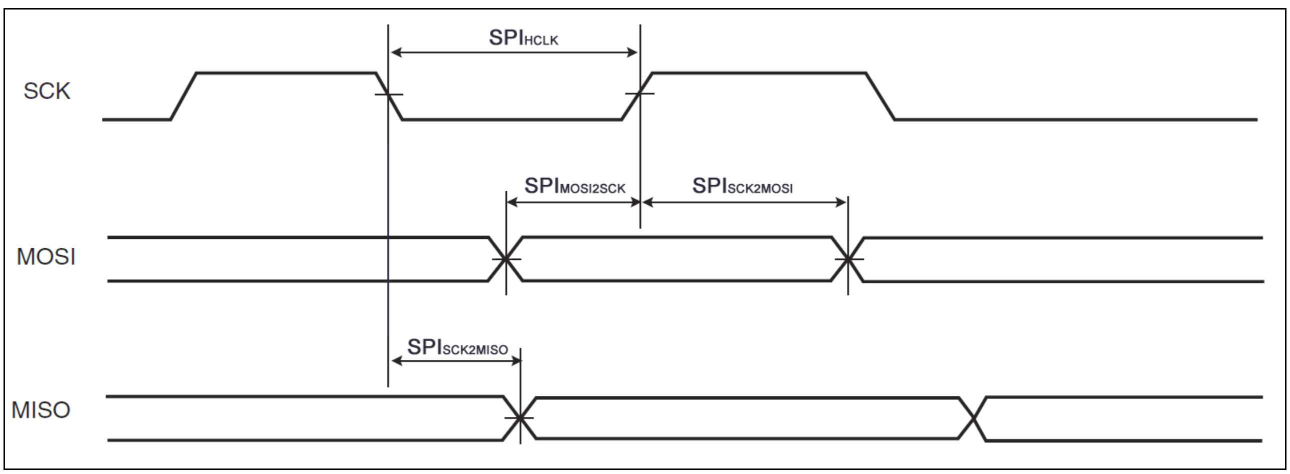
2.5.5 SPI Characteristics

The Serial Peripheral Interface or SPI is a full-duplex synchronous serial communication standard that is commonly supported on many micro-controllers and embedded systems.

The SPI interface is implemented as an SPI mode 0 slave device. This means that the SPI clock polarity is 0 (CPOL=0) and the SPI clock phase is 0 (CPHA=0). Bytes are transferred one bit at a time with the MSB being transferred first. The on-board SPI interface has been tested at speeds up to 6MHz. The diagram below illustrates a single complete SPI byte transfer.



The diagram and parameter table below illustrates additional timing requirements and limits of the SPI interface:



Symbol	Parameter	Min.	Max.	Units
SPI_{HCLK}	SPI Clock Cycle Period / 2	80		ns
$SPI_{SCK2MISO}$	SPI SCK falling to MISO Delay		26.5	ns
$SPI_{MOSI2SCK}$	SPI MOSI Setup time before SPI SCK rises	0		ns
$SPI_{SCK2MOSI}$	SPI MOSI Hold time after SPI SCK rises	1.5		ns

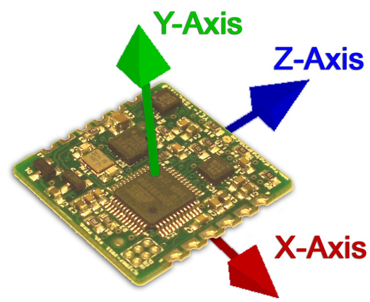
2.8 Axis Assignment

All YEI 3-Space Sensor product family members have re-mappable axis assignments and axis directions. This flexibility allows axis assignment and axis direction to match the desired end-use requirements.

The natural axes of the 3-Space Sensor Embedded module are as follows:

- The positive X-axis points out of the side of the sensor with pins 1 through 6.
- The positive Y-axis points out of the top of the sensor (the component side of the board).
- The positive Z-axis points out of the back of the sensor (the side with the LED, towards pins 6 and 7).

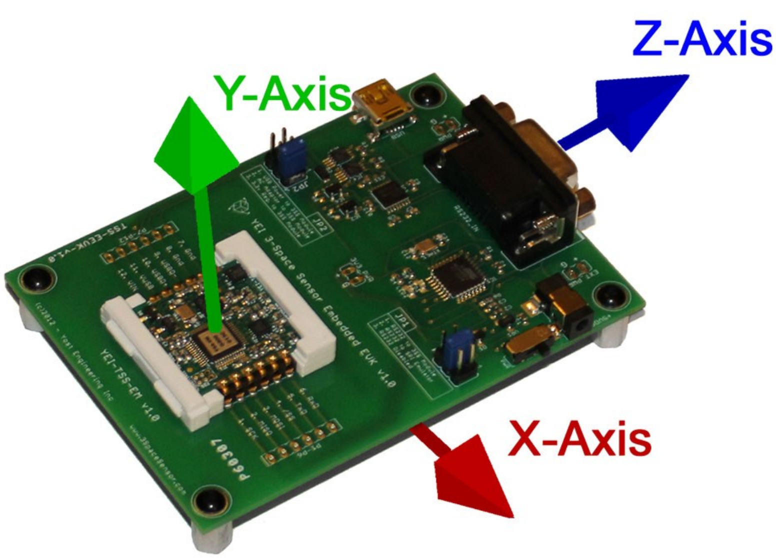
The natural axes of the TSS-EM module are illustrated in the diagram below:



The natural axes of the 3-Space Sensor Embedded Evaluation Kit are as follows:

- The positive X-axis points out of the side of the TSS-EEVK board with breakout pads 1 through 6.
- The positive Y-axis points out of the top of the TSS-EEVK board (the component side of the board).
- The positive Z-axis points out of the back of the TSS-EEVK board (the side with the RS232 connector).

The natural axes of the TSS-EEVK board are illustrated in the diagram below:



Bear in mind the difference between natural axes and the axes that are used in protocol data. While they are by default the same, they can be remapped so that, for example, data axis Y could contain data from natural axis X. This allows users to work with data in a reference frame they are familiar with.

3. Using the TSS-EEVK

3.1 Initial Hardware Set-up

3.1.1 Inserting TSS-EM Modules into the TSS-EEVK Socket

The TSS-EEVK employs a unique socket that allows for connection to the castellated edge SMT pads of the TSS-EM module without soldering.

To insert a TSS-EM module into the TSS-EEVK, perform the following steps:

1. Ensure that the TSS-EEVK is powered off and you are working in a static/ESD safe work area.
2. Ensure that the TSS-EEVK socket retention bar is either removed or in the open position.
3. Insert the TSS-EM module into the socket at a 15 degree to 25 degree angle. Be sure to observe correct orientation and pin alignment and be sure that the edge of the module is fully contacting the edge of the socket.
4. Gently push the module down into the socket such that the spring contacts are equally compressed.
5. Slide the socket retention bar toward the module until it is fully seated. You may need to alternate pressing gently on each side of the compression bar to ensure that it is fully engaged.

3.1.2 Removing TSS-EM Modules from the TSS-EEVK Socket

To remove a TSS-EM module from the TSS-EEVK, perform the following steps:

1. Ensure that the TSS-EEVK is powered off and you are working in a static/ESD safe work area..
2. Gently put pressure on the center of the TSS-EM module in the socket.
3. Gently slide the TSS-EEVK socket retention bar away from the module until it is fully in the open position. You may need to alternate applying pressure on each side in a slight wiggling or rocking action to open the socket retention bar.
4. Remove pressure from the TSS-EM module and allow the spring contacts to push it upward at an angle.
5. Gently grab the TSS-EM module by the sides and gently increase the angle to about 15 to 25 degrees.
6. Remove the TSS-EM module by pulling up and away from the socket.

3.1.3 Communication Selection & Routing Jumper JP1

Jumper JP1 is located near the power switch and is used to select and route RS232 communication signals. The table below summarizes the settings of Jumper JP1.

JP1 Position	Function
1	A jumper placed in position 1 routes the serials signals from the DB9 serial port to the TSS-EM module.
2	A jumper placed in position 2 routes the serial signals from the DB9 serial port to the serial SPI emulator.
3	A jumper placed in position 3 disables the routing of all serial signals from the DB9 serial port. This is useful when communicating directly with the TSS-EM module installed in the socket via the breakout pin-pads.

3.1.4 Power Selection & Routing Jumper JP2

Jumper JP2 is located near the USB connector and is used to select and route power sources. The table below summarizes the settings of Jumper JP2.

JP2 Position	Function
1	A jumper placed in position 1 routes the +5v USB bus signal to the VUSB pin of the TSS-EM module. The jumper must be placed in this position if enumeration and communication via USB is desired.
2	A jumper placed in position 2 routes the +5v AC adapter power from the external connector to the VIN pin of the TSS-EM module. This option may be selected if either serial or SPI communication is desired.
3	A jumper placed in position 3 routed the +3.3v regulated power to the VIN pin of the TSS-EM module. This regulator is powered from either USB or the AC adapter. The main use of this is to power the TSS-EM module with a typical 3.3v power supply without USB enumeration.

Note, the TSS-EM module can be powered from an external power source via the external break-out pin pads by removing the jumper from all positions of jumper JP2.

3.2 Configuring the TSS-EEVK for use with USB

When USB use of the TSS-EEVK is desired, perform the following steps:

1. Ensure that the power switch is in the off position.
2. Ensure that the TSS-EM module is correctly installed in the socket.
3. Ensure that jumper JP2 is set to position 1.
4. Connect the mini USB connector to an available USB port on the host system using an appropriate cable.
5. Power the system on. The TSS-EM module will enumerate and create a virtual COM port to facilitate communication. Refer to the TSS-EM User's Manual for detailed information describing the TSS-EM module communication protocol.

3.3 Using the TSS-EEVK with RS232 Serial

When RS232 serial use of the TSS-EEVK is desired, perform the following steps:

1. Ensure that the power switch is in the off position.
2. Ensure that the TSS-EM module is correctly installed in the socket.
3. Ensure that jumper JP2 is set to select the appropriate power source and that that power source is connected.
4. Ensure that jumper JP1 is set to position 1. This routes the serial communication signals to the TSS-EM module.
5. Connect the DB9 serial connector to the RS232 serial port on the host system using an appropriate cable.
6. Power the system on. The TSS-EM module will now be available for communication via the COM port to which the system is attached. Refer to the TSS-EM User's Manual for detailed information describing the TSS-EM module communication protocol.

3.4 Using the TSS-EEVK with RS232 SPI Emulation

When RS232 SPI Emulation use of the TSS-EEVK is desired, perform the following steps:

1. Ensure that the power switch is in the off position.
2. Ensure that the TSS-EM module is correctly installed in the socket.
3. Ensure that jumper JP2 is set to select the appropriate power source and that that power source is connected.
4. Ensure that jumper JP1 is set to position 2. This routes the serial communication signals to the TSS-EEVK SPI emulation processor on the TSS-EEVK board.
5. Connect the DB9 serial connector to the RS232 serial port on the host system using an appropriate cable.
6. Power the system on. The TSS-EM module will now be available for communication via the COM port to which the system is attached. Refer to the section 4 of this manual for detailed information describing the TSS-EEVK SPI emulation command protocol.

3.5 Using the TSS-EEVK with Logic-Level Serial

When logic-level serial communication use of the TSS-EEVK is desired, perform the following steps:

1. Ensure that the power switch is in the off position.
2. Ensure that the TSS-EM module is correctly installed in the socket.
3. Ensure that jumper JP2 is set to select the appropriate power source and that that power source is connected.
4. Ensure that jumper JP1 is set to position 3. This disables all routing of serial communication signals to the TSS-EM and allows the serial communications signals pins to be connected to an alternate source.
5. Connect the RxD and TxD break-out pan pins of the TSS-EM to the logic-level TxD and RxD signals, respectively, of the desired host system. Note that theses pins are designed with a native logic-level of 3.3v, but are 5v tolerant. Also, note that the two systems must also have a common ground, thus the grounds of the TSS-EEVK and the host system may need to be connected.
6. Power the system on. The TSS-EM module will now be available for communication via logic-level serial. Refer to the TSS-EM User's Manual for detailed information describing the TSS-EM module communication protocol.

3.6 Using the TSS-EEVK with Logic-Level SPI

When logic-level SPI use of the TSS-EEVK is desired, perform the following steps:

1. Ensure that the power switch is in the off position.
2. Ensure that the TSS-EM module is correctly installed in the socket.
3. Ensure that jumper JP2 is set to select the appropriate power source and that that power source is connected.
4. Ensure that jumper JP1 is set to position 3. This disables all routing of serial communication and SPI signals to the TSS-EM and allows the SPI communications signals pins to be connected to an alternate source.
5. Connect the MISO, MOSI, SCK, and /SS break-out pan pins of the TSS-EM to the corresponding logic-level signals of the desired host system. Note that theses pins are designed with a native logic-level of 3.3v, but are 5v tolerant. Also, note that the two systems must also have a common ground, thus the grounds of the TSS-EEVK and the host system may need to be connected.
6. Power the system on. The TSS-EM module will now be available for communication an an SPI slave device on the attached SPI master system to which it is attached. Refer to the TSS-EM User's Manual for detailed information describing the TSS-EM module communication protocol.

4. TSS-EEVK Serial SPI Emulation Protocol

4.1 Serial SPI Emulation Protocol Overview

The TSS-EEVK has the ability to test SPI commands using a unique serial SPI emulation mode. In this mode, a serial communication message is sent to an on-board emulation processor that translates the message into an action being taken on the SPI bus. This allows users to experiment with and troubleshoot the SPI functionality of the TSS-EM module without the need for making wire connections to a hardware SPI host.

Specific details of the TSS-EEVK serial SPI emulation protocol and its control commands are discussed in the following pages.

4.2 Serial SPI Emulation Protocol Commands

4.2.1 Binary SPI Emulation Protocol

The binary packet size can be one or more bytes long, depending upon the nature of the command being sent to the TSS-EEVK. Each packet consists of an initial **"command value"** specifier byte, followed by zero or more "**command data"** bytes.

Each binary packet is formatted as illustrated in figure 1 below.

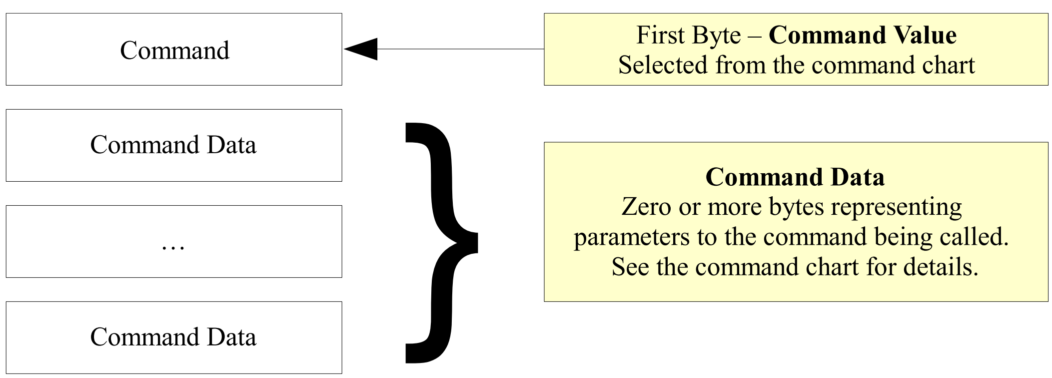


Figure 1 - Typical Binary Command Packet Format

Binary Return Values:

When a TSS-EEVK command is called in binary mode, any data it returns will also be returned as raw bytes in binary format.

4.2.2 ASCII Text Packet Format

ASCII text command packets are similar to binary command packets, but are received as a single formatted line of text. Each text line consists of the following: an ASCII colon character followed by an integral command id in decimal, followed by a list of ASCII encoded integer parameter values, followed by a terminating newline character. The command id and parameter values are given as positive decimal integers. The ASCII encoded command values must be separated by an ASCII comma character or an ASCII space character. Thus, legal command characters are: the colon, the comma, the digits 0 through 9, the new-line, the carriage return, the space, and the backspace. When a command calls for an integer or byte sized parameter, the floating point number given for that parameter will be interpreted as being the appropriate data type. For simplicity, the ASCII encoded commands follow the same format as the binary encoded commands, but ASCII text encodings of values are used rather than raw binary encodings.

Each ASCII packet is formatted as shown in figure 2.

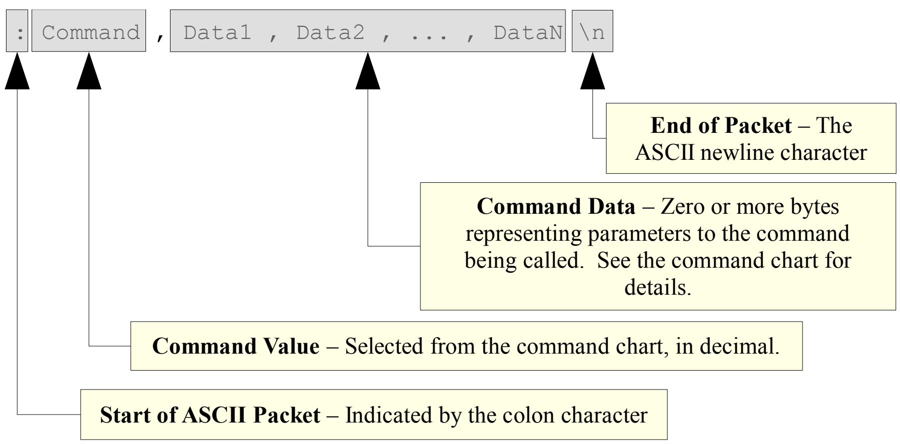


Figure 2 - Typical ASCII Command Packet Format

Thus the ASCII packet consists of the the following characters:

- **:** – the ASCII colon character signifies the start of an ASCII text packet.
- **,** – the ASCII comma character acts as a value delimiter when multiple values are specified.
- **0~9** – the ASCII digits are used to in integer and floating point values.
- **\n** – the ASCII newline character is used to signify the end of an ASCII command packet.
- **\r** – the ASCII return character can also be used to signify the end of an ASCII command packet.
- **\b** – the ASCII backspace character can be used to backup through the partially completed line to correct errors.

If a command is given in ASCII mode but does not have the right number of parameters, the entire command will be ignored and an error message will result.

Sample ASCII commands:

`:240\n`	Get Product String
`:242\n`	Set SPI Select Line Low
`:244,255\n`	Send Byte Value 255 to SPI Interface

ASCII Return Values:

All values are returned in ASCII text format when an ASCII-format command is issued. Each command response returns the command response data, followed by an additional carriage return and line feed. To read the return data, simply read data characters sequentially until a Windows newline(a carriage return and a line feed) is encountered. ASCII commands that return multiple data items return comma separated lists of integers.

4.2.2 ASCII Text Help Screen

When connected, a single ASCII question mark character will result the TSS-EEVK returning an ASCII table of commands supported by the TSS-EEVK SPI Emulator.

4.3 Serial SPI Emulation Protocol Command Details

4.3.1 TSS-EEVK Serial SPI Emulator Protocol Command Chart

The table below summarizes the commands supported by the TSS-EEVK when used in Serial SPI Emulator mode.

Byte	Function	Parameter Bytes	Response Bytes
F0 (240)	Get Product String	None	[byte] x 16 (product string)
F1 (241)	Get Version String	None	[byte] x 8 (version string)
F2 (242)	SPI Select Low	None	None
F3 (243)	SPI Select High	None	None
F4 (244)	SPI Send Byte	[byte]	None
F5 (245)	SPI Send Byte / Receive Byte	[byte]	[byte]
F6 (246)	SPI Send String / Receive String	[count] (count*[byte])	(count*[byte])
F7 (247)	SPI Send String / Receive String with Select	[count] (count*[byte])	(count*[byte])
F8 (248)	Set SPI rate	[rate enum]	None
F9 (249)	Get SPI rate	None	[rate enum]
FA (250)	Set SPI Mode	[SPI Mode enum]	None
FB (251)	Get SPI Mode	None	[SPI Mode enum]
FC (252)	Command Test (TSS-EM Command 230)	None	[byte] x 32 (command response)
FD (253)	Command Test	[command] [numBytes]	[byte] x numBytes

4.3.1 TSS-EEVK Serial SPI Emulator Protocol Command Chart

In the tables below you'll find a description of each of the TSS-EEVK Serial SPI Emulator commands and a brief explanation of how and where each command would be used.

Function:	Get Product String
Command Value:	240 (0xF0)
Return Data Bytes:	16
Return Data Format:	Product String (byte x 16)
Description:	Returns the 16 byte product string for the TSS-EEVK. For the TSS-EEVK, the product string should be: TSS_EEVK_SPI_EMU

Function:	Read Product Version
Command Value:	241 (0xF1)
Return Data Bytes:	8
Return Data Format:	Product Version (byte x 8)
Description:	Returns the 8 byte version string for the TSS-EEVK. Unused bytes will be padded with spaces. This string will be something like: 1.0.0

Function:	Set Select Line Low
Command Value:	242 (0xF2)
Description:	This command sets the slave select line (/SS) of the SPI interface to a low state. Since the TSS-EM module uses an active-low slave select, this will select the TSS-EM module to enable further SPI communication.

Function:	Set Select Line High
Command Value:	243 (0xF3)
Description:	This command sets the slave select line (/SS) of the SPI interface to a high state. Since the TSS-EM module uses an active-low slave select, this will deselect the TSS-EM module and disable further SPI communication.

Function:	SPI Send Byte
Command Value:	244 (0xF4)
Data Bytes:	1
Data Format:	[byte] - Byte to be transmitted via SPI
Description:	This command sends the byte specified by the first command parameter over the SPI interface. The /SS line state is unchanged and the nothing is returned.

Function:	SPI Send Byte / Receive Byte
Command Value:	245 (0xF5)
Data Bytes:	1
Data Format:	[byte] - Byte to be transmitted via SPI
Return Data Bytes:	1
Return Data Format:	Byte returned from the SPI transaction
Description:	This command sends the byte specified by the first command parameter over the SPI interface. The /SS line state is unchanged and the result of the SPI transaction is returned as a single byte.

Function:	SPI Send String / Receive String
Command Value:	246 (0xF6)
Data Bytes:	Varies (1~256 bytes)
Data Format:	[count] (count*[byte])
Return Data Bytes:	Varies (0~255 bytes)
Return Data Format:	[byte] *count – String of requested bytes.
Description:	This command sends a number of bytes specified by the first command parameter over the SPI interface and subsequently returns the number of bytes as received from the SPI interface. The /SS line state is unchanged and the number of bytes returned will always match the number specified by the first command parameter.

Function:	SPI Send String / Receive String with Select
Command Value:	247 (0xF7)
Data Bytes:	Varies (1~256 bytes)
Data Format:	[count] (count*[byte])
Return Data Bytes:	Varies (0~255 bytes)
Return Data Format:	[byte] *count – String of requested bytes.
Description:	This command selects the SPI slave, sends a number of bytes specified by the first command parameter over the SPI interface and subsequently returns the number of bytes as received from the SPI interface, and deselects the SPI slave. The /SS line state is set low during transfer and the number of bytes returned will always match the number specified by the first command parameter.

Function:	Set SPI Rate
Command Value:	248 (0xF8)
Data Bytes:	1
Data Format:	[byte] – Enum byte specifying the SPI rate to be used.
Description:	This command sets the SPI transfer rate according to rate specified by the enum in parameter byte 0. Possible rate selection bytes are as follows:

Rate enum value	SPI Rate
0 (reset value)	6 Mhz
1	3 Mhz
2	1.5 Mhz
3	750 Khz
4	375 Khz
5	187 Khz
6	93.75 Khz

Upon reset the SPI rate will revert to rate setting 0 (6 Mhz).

Function:	Get SPI Rate
Command Value:	249 (0xF9)
Return Data Bytes:	1
Return Data Format:	[byte] – Enum byte specifying the SPI rate currently selected.
Description:	This command reads the SPI transfer rate setting. For possible rate values see the description of command 248 above. Upon reset the SPI rate will revert to rate setting 0 (6 Mhz).

Function:	Set SPI Mode
Command Value:	250 (0xFA)
Data Bytes:	1
Data Format:	[byte] – Enum byte specifying the SPI mode to be used.
Description:	This command sets the SPI transfer rate according to rate specified by the enum in parameter byte 0. Possible rate selection bytes are as follows:

Mode enum	SPI Mode	Data Order	Details
0 (reset value)	0	MSB First	CPOL=0, CPHA=0, DORD=0
1	1	MSB First	CPOL=0, CPHA=1, DORD=0
2	2	MSB First	CPOL=1, CPHA=0, DORD=0
3	3	MSB First	CPOL=1, CPHA=1, DORD=0
4	0	LSB First	CPOL=0, CPHA=0, DORD=1
5	1	LSB First	CPOL=0, CPHA=1, DORD=1
6	2	LSB First	CPOL=1, CPHA=0, DORD=1
7	3	LSB First	CPOL=1, CPHA=1, DORD=1

Upon reset the SPI rate will revert to rate setting 0 (SPI Mode 0, MSB first).

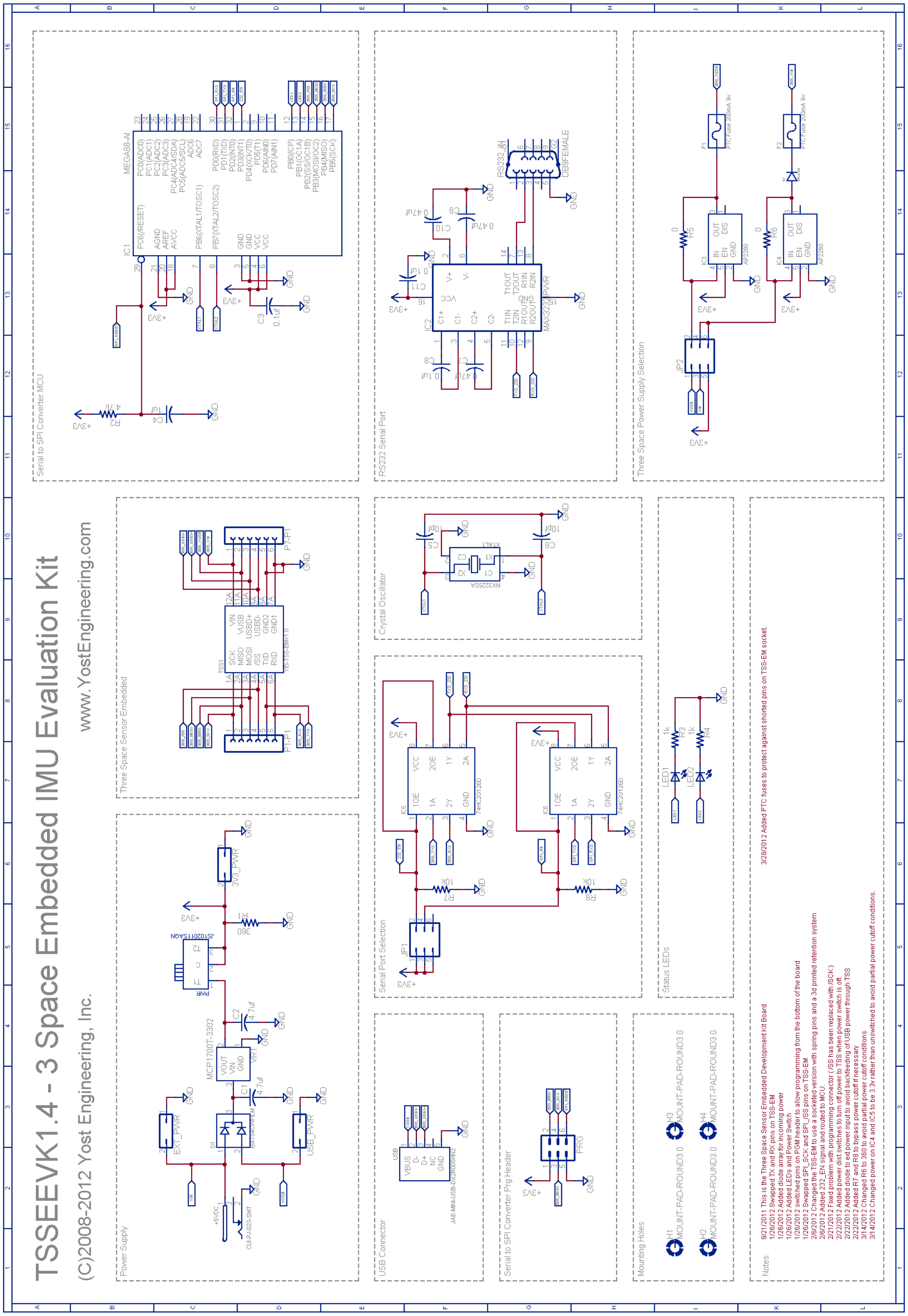
Function:	Get SPI Mode
Command Value:	251 (0xFB)
Return Data Bytes:	1
Return Data Format:	[byte] – Enum byte specifying the SPI mode currently selected.
Description:	This command reads the SPI mode setting. For possible rate values see the description of command 250 above. Upon reset the SPI rate will revert to mode setting 0 (SPI Mode 0, MSB first).

Function:	Command Test (TSS-EM Command 230)
Command Value:	252 (0xFC)
Return Data Bytes:	32
Return Data Format:	Data as returned from TSS-EM command 230 (byte x 32)
Description:	This command issues an entire command sequence for command 230 of the TSS-EM module. This command is useful for easily generating a known-working timing-correct SPI command sequence for the TSS-EM module for test purposes or oscilloscope-capture purposes. For more complex commands, use the general SPI commands.

Function:	Command Test
Command Value:	253 (0xFD)
Data Bytes:	2
Data Format:	[command] [numBytes] - command byte to be issued to the TSS-EM followed by the number of bytes to be returned from that command.
Return Data Bytes:	Varies (1~255)
Return Data Format:	[byte] *count – String of requested bytes as returned from TSS-EM command specified by the [command] byte in parameter 0.
Description:	This command issues an entire command sequence for the TSS-EM module. The command is specified by parameter byte 0 and the number of expected return bytes is specified by parameter 1. This command is useful for easily generating a known-working timing-correct SPI command sequence for the TSS-EM module for test purposes or oscilloscope-capture purposes. For more complex commands, use the general SPI commands.

Appendix

Circuit Schematic



Hex / Decimal Conversion Chart

	Second Hexadecimal digit															
First Hexadecimal Digit	***0***	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***	***A***	***B***	***C***	***D***	***E***	***F***
0	000	001	002	003	004	005	006	007	008	009	010	011	012	013	014	015
1	016	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031
2	032	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047
3	048	049	050	051	052	053	054	055	056	057	058	059	060	061	062	063
4	064	065	066	067	068	069	070	071	072	073	074	075	076	077	078	079
5	080	081	082	083	084	085	086	087	088	089	090	091	092	093	094	095
6	096	097	098	099	100	101	102	103	104	105	106	107	108	109	110	111
7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
9	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
A	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
B	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
C	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
D	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
E	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
F	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Notes:

YEI Technology
630 Second Street
Portsmouth, Ohio 45662

Toll-Free: 888-395-9029
Phone: 740-355-9029

www.YeiTechnology.com
www.3SpaceSensor.com

Patents Pending

Printed in USA