1.2.5 UART-RVC interface

The BNO08X provides a simplified UART interface for use on unmanned ground roving robot such as robot vacuum cleaners (RVC). When configured in this mode the BNO08X simply transmits heading and sensor information at 100Hz over the UART TX pin. A typical connection is shown in Figure 1-23. When using the UART interface either the external clock or crystal must be used. The internal clock is not accurate enough to operate the UART interface reliably.

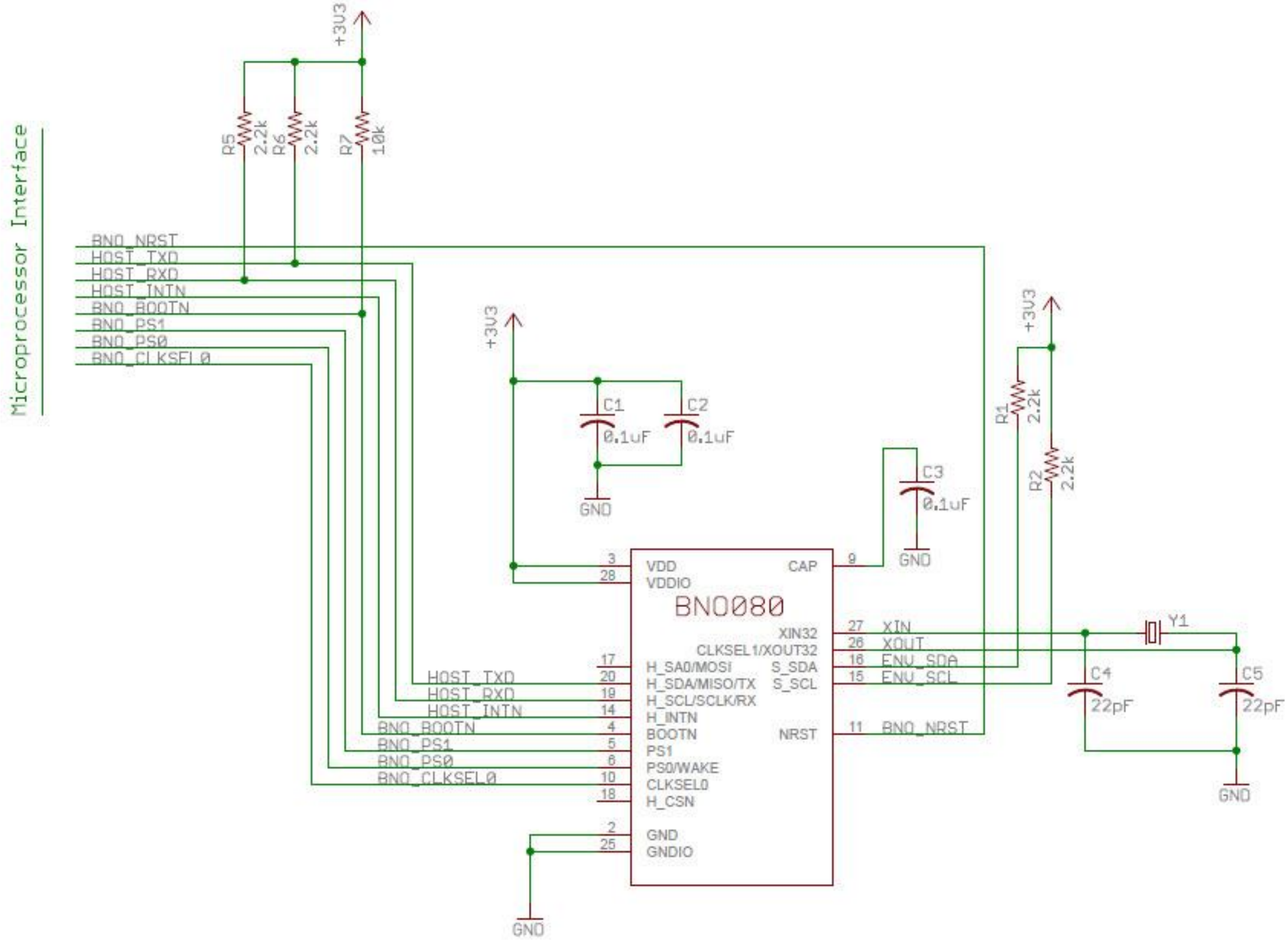

Figure 1-23: BNO08X UART-RVC connection diagram

Figure 1-23 shows how the BNO08X can be connected to an external microcontroller via a UART interface. The following notes are provided as guidelines for connecting the BNO08X in a system design.

1. NRST is the reset line for the BNO08X and can be either driven by the application processor or the board reset.
2. BOOTN is sampled at reset. If low the BNO08X will enter bootloader mode.
3. Pin 4 (BOOTN) should be pulled high through a 10K Ohms resistor. To use the device firmware update (DFU) capability of the BNO08X, it is recommended to connect Pin 4 to a GPIO pin on the external microcontroller.
4. Pin 5 (PS1) and Pin 6 (PS0) are the host interface protocol selection pins. These pins should be tied to ground and VDDIO respectively to select the UART-RVC interface.
5. The BNO08X supports environmental sensors (e.g. pressure sensors, ambient light sensors) on a secondary I^2C interface. This interface should be pulled up via resistors regardless of the presence of the external sensor as the SW polls for sensors at reset.

1.2.5.1 UART-RVC operation

The UART operates at 115200 b/s, 8 data bits, 1 stop bit and no parity. The UART protocol relies on an idle line being ‘high’. A transmission is started with the assertion of a start bit (pulling the line low), followed by the data, LSB first. After the data segment is sent (in this case 8-bits), the line is pulled high (the stop signal) for a minimum number of bits (1 for the BNO08X) to indicate end of that segment.

Figure 1-24: UART signaling

1.2.5.2 UART-RVC protocol

The BNO08X transmits the following data at a rate of 100Hz.

Header	Index	Yaw		Pitch		Roll		X-axis accel		Y-axis accel		Z-axis accel		MI	MR	Rsvd	Csum
0xAAAA		LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB				

Figure 1-25: BNO08X UART-RVC packet format

The 19-byte message has the following fields:

Header: Each report is prefixed with a 0xAAAA header
Index: A monotonically increasing 8-bit count is provided (0-255) per report
Yaw: The yaw is a measure of the rotation around the Z-axis since reset. The yaw has a range of +/- 180˚ and is provided in 0.01˚ increments, i.e. a report of 8734 is equivalent to 87.34˚.
Pitch: The pitch is a measure of the rotation around the Y-axis. The pitch has a range of +/- 90˚ and is provided in 0.01˚ increments, i.e. a report of 1072 is equivalent to 10.72˚.
Roll: The roll is a measure of the rotation around the X-axis. The roll has a range of +/- 180˚ and is provided in 0.01˚ increments, i.e. a report of 1072 is equivalent to 10.72˚.
X-axis acceleration: The acceleration along the X-axis, presented in mg
Y-axis acceleration: The acceleration along the Y-axis, presented in mg
Z-axis acceleration: The acceleration along the Z-axis, presented in mg
MI – Motion Intent – BNO086 only. Otherwise, reserved
MR – Motion Request – BNO086 only. Otherwise, reserved.
Reserved: The message is terminated with one (BNO086) or three (otherwise) reserved bytes, currently set to zero
Checksum (Csum): The Index, yaw, pitch, roll, acceleration and reserved data bytes are added to produce the checksum.

To determine the actual orientation of the module, the rotations should be applied in the order yaw, pitch then roll.

An example complete message and checksum calculation is as follows:

Message: 0xAA AA DE 01 00 92 FF 25 08 8D FE EC FF D1 03 00 00 00 E7

Where:

Index = 0xDE = 222
Yaw = 00.01˚ (1 = 0x0001)
Pitch = -1.10˚ (-110 = 0xFF92)
Roll = 20.85˚ (2085 = 0x0825)
X-acceleration = -371 mg = -3.638 m/s^2 (-371 = 0xFE8D)
Y-acceleration = -20 mg = -0.196 m/s^2 (-20 = 0xFFEC)

Z-acceleration = 977 mg = 9.581 m/s^2 (977 = 0x03D1)
Checksum = 0xE7

1.3 Host Communication

1.3.1 SHTP

The BNO08X uses CEVA's SHTP (Sensor Hub Transport Protocol) to communicate for all interface styles except UART-RVC. SHTP provides a means of passing data between the BNO08X and a host with support for multiple channels. The BNO08X does not currently support the inclusion of 3rd party applications, but the SHTP protocol allows for separation of traffic via these channels such that applications on a host can communicate over this channel.

All data is prefixed with a 4-byte header:

Byte	Field
0	Length LSB
1	Length MSB
2	Channel
3	SeqNum

Figure 1-26: SHTP Header

Length	Bit 15 of the length field is used to indicate if a transfer is a continuation of a previous transfer. Bits 14:0 are used to indicate the total number of bytes in the cargo plus header, which may be spread over multiple messages. The bytes in the header field are counted as part of the length. A length of 65535 (0xFFFF) is reserved because a failed peripheral can too easily produce 0xFFFF. Therefore, the largest cargo that can be transported is 32766 minus the header bytes. The BNO08X does not support receiving fragmented messages but it does support sending them.
Channel	The channel number of the cargo. Channel 0 is the command channel and is used by the SHTP.
SeqNum	The sequence number of the cargo. The sequence number is a monotonically incrementing number that increments once for each cargo sent or cargo continuation sent. Each channel and each direction has its own sequence number. The sequence number is used to detect duplicate or missing cargoes and to associate segmented cargoes with each other.

The length field allows a host to schedule the correct number of clocks to generate for reads over I^{2}C and SPI. The BNO08X supports partial reads if the host cannot accept all the bytes in one read, the length will be updated on a subsequent read. So, for instance a host could read the first 4 bytes to determine the number of clocks to generate and then repeat the read with the required number of clocks. The protocol is fully described in [2].

The BNO08X supports 6 channels:

- Channel 0: the SHTP command channel
- Channel 1: executable
- Channel 2: sensor hub control channel
- Channel 3: input sensor reports (non-wake, not gyroRV)
- Channel 4: wake input sensor reports (for sensors configured as wake up sensors)
- Channel 5: gyro rotation vector

The SHTP control channel provides information about the applications built into the BNO08X firmware image. The BNO08X uses advertisements to publish the channel maps and the names of the built-in applications.