QFLEX 2 Embedded for Quanser AERO

For additional resources visit www.quanser.com/courseware

Features

Control DC motor voltages
Read encoder feedback from DC motors
Read encoder value from pitch and yaw pivots
Read current sense feedback from PWM amplifier
Read acceleration and velocity information from IMU
Set encoder reference points
Control tri-colour LED output

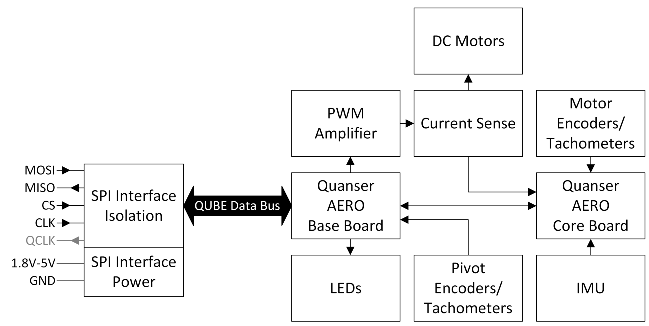

Figure 1: *System Diagram*

System Overview

The QFLEX 2 Embedded provides a standard SPI interface for interacting with the Quanser AERO. The QFLEX 2 Embedded panel is powered via the 1.8V-5V and GND input pins and provides signal isolation between the external connector and the internal QUBE data bus. Data signals in the SPI interface will operate at the voltage provided on the 1.8V-5V pin. The Quanser AERO controller receives the data from the SPI interface and sets the motor PWM duty cycle, LED brightness, and encoder reference. The controller collects current sense and encoder read values and feeds this information back over the data bus, back to the SPI interface. The full system diagram is shown in Figure 1.

Connections

The QFLEX 2 Embedded has a 7-pin connector (shown in Figure 2) which mates with a TE Connectivity 104257-6 rectangular connector. Table 1 outlines the IO connection pins

Table 1: *QFLEX 2 Embedded IO connections*

#	Pin Name	Description
1	1.8V-5V	Controller Vcc
2	MOSI	AERO data IN
3	MISO	AERO data OUT
4	CLK	SPI data clock IN
5	QCLK	QBUS clock OUT
6	CS	AERO slave chip select
7	GND	Controller ground

Communication

The QFLEX 2 Embedded operates as a standard SPI slave with SPI mode 2. Note that the complete data packet is comprised of two sub-packets, the first intended for the AERO base board, the second intended for the core board. When SPI communication is initiated by pulling the CS line low, the first byte of data received by the QFLEX 2 Embedded, referred to as the mode byte, will determine the communication mode to follow.

Mode = 0 Only read ID (no settings changed)

Mode = 1 A full command packet is transmitted

Concurrently, as the mode byte is being received, the Quanser AERO will respond with the upper byte of the base board ID. The next byte sent to the QFLEX 2 Em-

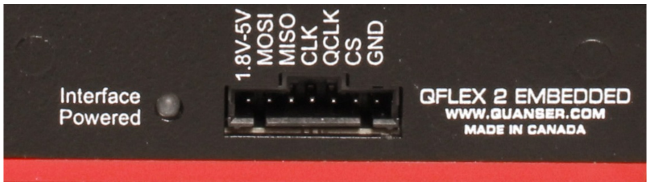

Figure 2: *QFLEX 2 Embedded Panel*

bedded is a padding byte with a value of 0, the data returned during this transmission will be the lower byte of the base board ID. At this point if Mode = 0, the next expected byte will be the mode byte for the core board, or byte 15 in a normal packet. If Mode = 1, the command communication packet continues as outlined in Table 2. Similarly, if mode 0 is selected for the core board, a padding byte will be expected and the lower byte of the core board ID will be returned, after which communication is complete and the CS line can be returned high. Otherwise if Mode = 1 for the core board, communication continues as outlined in Table 2.

Transmit/Receive Bytes

The expected values and description for each of the data bytes are as follows.

Base Mode (Transmit byte 0)

Expected value 0x00-0x01. A value of 0 will result in only the Base ID being returned. A value of 1 will initiate the transmission of a full command packet.

Base Write Mask (Transmit byte 2)

Expected value 0x00-0x7F. This byte controls what values will be overwritten on the AERO base board. The mapping of the bits in the mask to written values is shown in Table 3 To zero the pivot encoders, the write mask must be 0b00011xxx. To set the LED values, the mask must be 0b000xx111.

LED Values (Transmit bytes 3-8)

Expected value 0x0000-0x03E7. The brightness of the LEDs is controlled on a scale from 0 to 999 (decimal) this value is transmitted over two bytes for each LED with the MSB preceding the LSB.

Set Encoder Values (Transmit bytes 9-14,22-27)

These bytes allow for the encoder counts value to be set as desired. The most likely application for these bytes is to send 0x00 for all bytes to zero the encoder counts.

Table 2: *QFLEX 2 Embedded Command Packet Structure*

B	MOSI Data	MISO Data
	Start of Base Packet	
0	Base Mode (0x01)	Base ID MSB
1	Padding byte (0x00)	Base ID LSB
2	Base Write Mask	Encoder 2 (23-16)
3	Red LED MSB	Encoder 2 (15-8)
4	Red LED LSB	Encoder 2 (7-0)
5	Green LED MSB	Encoder 3 (23-16)
6	Green LED LSB	Encoder 3 (15-8)
7	Blue LED MSB	Encoder 3 (7-0)
8	Blue LED LSB	Tachometer 2 (23-16)
9	Set Encoder 2 (23-16)	Tachometer 2 (15-8)
10	Set Encoder 2 (15-8)	Tachometer 2 (7-0)
11	Set Encoder 2 (7-0)	Tachometer 3 (23-16)
12	Set Encoder 3 (23-16)	Tachometer 3 (15-8)
13	Set Encoder 3 (15-8)	Tachometer 3 (7-0)
14	Set Encoder 3 (7-0)	Reserved (0x00)
	Start of Core Packet	
15	Core Mode (0x01)	Core ID MSB
16	Padding byte (0x00)	Core ID LSB
17	Core Write Mask	Current Sense 0 (15-8)
18	Motor 0 Command (15-8)	Current Sense 0 (7-0)
19	Motor 0 Command (7-0)	Current Sense 1 (15-8)
20	Motor 1 Command (15-8)	Current Sense 1 (7-0)
21	Motor 1 Command (7-0)	Tachometer 0 (23-16)
22	Set Encoder 0 (23-16)	Tachometer 0 (15-8)
23	Set Encoder 0 (15-8)	Tachometer 0 (7-0)
24	Set Encoder 0 (7-0)	Tachometer 1 (23-16)
25	Set Encoder 1 (23-16)	Tachometer 1 (15-8)
26	Set Encoder 1 (15-8)	Tachometer 1 (7-0)
27	Set Encoder 1 (7-0)	Status
28	Padding byte (0x00)	Encoder 0 (23-16)
29	Padding byte (0x00)	Encoder 0 (15-8)
30	Padding byte (0x00)	Encoder 0 (7-0)
31	Padding byte (0x00)	Encoder 1 (23-16)
32	Padding byte (0x00)	Encoder 1 (15-8)
33	Padding byte (0x00)	Encoder 1 (7-0)
34	Padding byte (0x00)	Accelerometer X (15-8)
35	Padding byte (0x00)	Accelerometer X (7-0)
36	Padding byte (0x00)	Accelerometer Y (15-8)
37	Padding byte (0x00)	Accelerometer Y (7-0)
38	Padding byte (0x00)	Accelerometer Z (15-8)
39	Padding byte (0x00)	Accelerometer Z (7-0)
40	Padding byte (0x00)	Gyroscope X (15-8)
41	Padding byte (0x00)	Gyroscope X (7-0)
42	Padding byte (0x00)	Gyroscope Y (15-8)
43	Padding byte (0x00)	Gyroscope Y (7-0)
44	Padding byte (0x00)	Gyroscope Z (15-8)
45	Padding byte (0x00)	Gyroscope Z (7-0)
46	Padding byte (0x00)	Reserved (0x00)
47	Padding byte (0x00)	Reserved (0x00)
48	Padding byte (0x00)	Reserved (0x00)
49	Padding byte (0x00)	Reserved (0x00)
50	Padding byte (0x00)	Reserved (0x00)

Table 3: *Base Write Mask Bit Mapping*

b	Action Enabled
7	-
6	-
5	-
4	Set Encoder 3
3	Set Encoder 2
2	Write Blue LED
1	Write Green LED
0	Write Red LED

Core Mode (Transmit byte 15)

Expected value 0x00-0x01. A value of 0 will result in only the Core ID being returned, ending communication after byte 16. A value of 1 will continue the transmission of a full command packet.

Core Write Mask (Transmit byte 2)

Expected value 0x00-0x7F. This byte controls what values will be overwritten on the AERO base board. The mapping of the bits in the mask to written values is shown in Table 3 To set the motor encoders, the write mask must be 0b0011xxxx. To enable the motors and allow a command value to be written the mask must be 0b00xx1111.

Table 4: *Core Write Mask Bit Mapping*

b	Action Enabled
7	-
6	-
5	Set Encoder 1
4	Set Encoder 0
3	Write Motor 1 Voltage
2	Write Motor 1 Enable
1	Write Motor 0 Voltage
0	Write Motor 0 Enable

Motor Command (Transmit bytes 18-21)

Bit 15 of the motor commands control whether the PWM amplifier in the Quanser AERO for that motor is activated. Thus the upper byte of the motor command must be 0b1xxxxxxx in order for the motor to be enabled. The expected value of the bits 14-0 of the motor command is a value between -999 and 999 (decimal) formatted in 2's complement and represents a value equal to 10 times the desired percentage duty cycle of the PWM amplifier. For example, writing the motor command value 0x81F4 would activate the PWM amplifier and apply a 50% duty cycle, equivalent to approximately 12VDC.

Padding Bytes (Transmit bytes 28-50)

The value of these bytes are ignored by the QFLEX 2 Embedded, serving only to provide clock cycles for additional MISO data. By convention these bytes are usually zero.

Base Module ID (Receive bytes 0-1)

Expected Value of these bytes for the Quanser AERO Base Module is always 0x304 or 772 decimal. Any other value indicates a fault in communication between the QFLEX 2 Embedded and the AERO base board.

Read Encoder Values (Receive bytes 2-7,28-33)

These bytes represent the current value of the encoder counts, represented in 2's complement. Each value consists of three bytes and begins with the most significant byte. Encoders 0 and 1 are the encoders indicating the position of the DC motors. Encoder 2 is the encoder connected to the shoulder "pitch" pivot of the AERO body. Encoder 3 is the encoder connected to the base "yaw" pivot of the support yolk. Note that encoders 0-2 are 2048 quadrature counts per rotation, while encoder 3 is 4096 quadrature counts per rotation.

Tachometer Value (Receive bytes 8-13,21-26)

These bytes represent the current value read from the Quanser AERO internal tachometer, consisting of three bytes and beginning with the most significant byte. Tachometer values are calculated for each encoder. The most significant bit represents the direction of rotation with a value of zero indicating counter-clockwise rotation. Bits 23-0 represent the number of clock cycles (at 40MHz) between rising edges of the encoder *A* signal line. The encoder value can be converted to encoder counts per second with the following equation:

$$\text{Counts/Sec} = \frac{4}{Tach \cdot (25 \cdot 10^{-9})} \quad (1)$$

Note that when the motor/pivot velocity is 0 the tachometer value will indicate the maximum value of 0x7FFFFF indicating a reading of approximately 20 counts/s.

Core Module ID (Receive bytes 15-16)

Expected Value of these bytes for the Quanser AERO Core Module is always 0x307 or 775 decimal. Any other value indicates a fault in communication between the QFLEX 2 Embedded and the AERO core board.

Current Sense (Receive bytes 17-20)

These bytes represent the measured current draw of the DC motors, represented in two's complement. The current can be calculated using the equation:

$$\text{Current (mA)} = \frac{I_{sense} - 8190}{9828} \quad (2)$$

Status (Receive byte 27)

The three least significant bits of this byte represent various warning or error states as outlined in Table 5. A status byte value of 0x00 indicates normal operation.

Table 5: *Status Bits*

b	Indicated Status
7	-
6	-
5	Motor 1 Stall Error
4	Motor 1 Stall Detected
3	Motor 0 Stall Error
2	Motor 0 Stall Detected
1	Amplifier 1 Over-Current Fault
0	Amplifier 0 Over-Current Fault

Accelerometer Readings (Receive bytes 34-39)

There are three accelerometer readings each represented by a 16-bit signed integer in two's complement. The full scale reading is $\pm 2g$ so the acceleration can be calculated from the measured value using the following formula:

$$\text{Acceleration} = \frac{2 \cdot Acc_n \cdot 9.81}{32768} \quad (3)$$

Note that only the X axis (extending down the center of the AERO body towards thruster 0) and Z axis (extending vertically when the AERO body is horizontal) will be of interest. Under normal operation the Y-axis acceleration should always be approximately zero.

Gyroscope Readings (Receive bytes 40-45)

There are three accelerometer readings each represented by a 16-bit signed integer in two's complement. The full scale reading is ± 245 degrees/s so the angular velocity can be calculated from the measured value using the following formula:

$$\omega_n = \frac{Gyro_n \cdot 245}{32768} \quad (4)$$

Quanser Inc.
119 Spy Court
Markham, ON
L3R 5H6
Canada
info@quanser.com
Phone: 1-905-940-3575
Fax: 1-905-940-3576

Printed in Markham, Ontario.

For more information on the solutions Quanser Inc. offers, please visit the web site at: http://www.quanser.com