USER MANUAL
QBall 2 for QUARC

Set Up and Configuration

CAPTIVATE. MOTIVATE. GRADUATE.

Quanser Inc.
119 Spy Court
Markham, Ontario
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

1 Presentation

1.1 Introduction

The Quanser QBall 2 (Figure 2.1) is an innovative rotary wing vehicle platform suitable for a wide variety of UAV research applications. The QBall 2 is a quadrotor helicopter design propelled by four brushless motors fitted with 10-inch propellers [3, 4, 5]. The entire quadrotor is enclosed within a protective carbon fiber cage (Patent Pending). The QBall 2's proprietary design ensures safe operation as well as opens the possibilities for a variety of novel applications. The protective cage is a crucial feature since this unmanned vehicle was designed for use in an indoor laboratory, where there are typically many close-range hazards (including other vehicles). The cage gives the QBall 2 a decisive advantage over other vehicles that would suffer significant damage if contact occurs between the vehicle and an obstacle.

To measure on-board sensors and drive the motors, the QBall 2 utilizes Quanser's on-board avionics data acquisition card (DAQ) and a wireless Gumstix DuoVero embedded computer [1]. The DAQ is a high-resolution inertial measurement unit (IMU) and avionics input/output (I/O) card designed to accommodate a wide variety of research applications. QUARC®, Quanser's real-time control software [2], allows researchers and developers to rapidly develop and test controllers on actual hardware through a Matlab® Simulink® interface. QUARC®'s open-architecture hardware and extensive Simulink® blockset provides users with powerful controls development tools. QUARC® can target the Gumstix embedded computer, automatically generating code and executing controllers on-board the vehicle. During flights, while the controller is executing on the Gumstix, users can tune parameters in real-time and observe sensor measurements from a host ground station computer (PC or laptop).

The interface to the QBall 2 is Matlab® Simulink® with QUARC®. The controllers are developed in Simulink® with QUARC® on the host computer, and these models are downloaded and compiled into executables on the target seamlessly. A diagram of this configuration is shown in Figure 2.2.

Section 2 outlines operator warnings found throughout this manual, Section 3 goes through the prerequisites, and Section 4 lists various documents that are referenced in this manual. The general system description, component nomenclature, specifications, and model parameters are all given in Section 5. Section 6 goes into detail on how to setup the QBall 2, and Section 7 describes the battery charging procedure. Lastly, Section 8 contains a troubleshooting guide.

2 Operator Warnings

Caution

This symbol marks specific safety warnings and operating procedures that are important for the safety of the QBall 2 and users. Read these warnings carefully. The QBall 2 is a powerful and potentially dangerous vehicle if used improperly. Always follow safe operating procedures when using the QBall 2. Quanser is not responsible for damages and injury resulting from improper or unsafe use of the QBall 2. Before connecting batteries or attempting to run the QBall 2, be sure to read this document and become familiar with the safety features and operating procedures of the QBall 2.

Caution

When handling the QBall 2, always make sure there are no models running and the power is turned off. It is recommended that users wear safety goggles to protect the eyes.

Figure 2.1: Quanser QBall 2

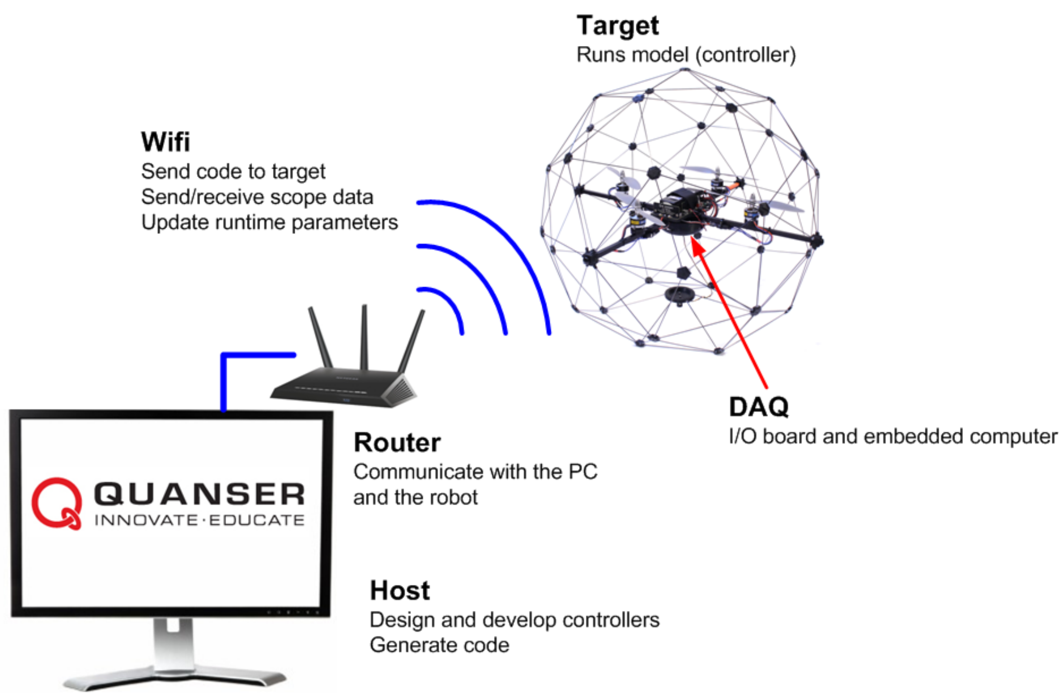


Figure 2.2: System diagram

3 Prerequisites

To successfully operate the QBall 2, the prerequisites are:

i) To be familiar with the wiring and components of the QBall 2.

ii) To have QUARC® version 2.4 installed and properly licensed.

iii) To be familiar with using QUARC® to control and monitor the vehicle in real-time, and in designing a controller through Simulink®. See Reference [2] for more details.

[1] Gumstix: http://gumstix.com/
[2] QUARC® User Manual **(type** `doc quarc` **in Matlab® to access)**
[3] Park 480 Brushless motor - 1020Kv:
http://hobbyhobby.com/store/product/68211%22Park-480-Brushless-Outrunner-Motor%2C-1020Kv%22/
[4] Propellers description and technical information:
http://www.rctoys.com/rc-products/APC-10-047-SF-CR.html
[5] Hobbywing Flyfun-30A electronic speed controller manual:
http://www.hobbywing.com/uploadfiles/sx/file/Manual/HW-01-V4.pdf
[6] STMicroelectronics L3G4200D 3-axis gyroscope:
http://www.st.com/web/catalog/sense_power/FM89/SC1288/PF250373
[7] Freescale MMA8452Q 3-axis accelerometer:
http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MMA8452Q

4 System Hardware

4.1 Main Components

To setup this experiment, the following hardware and software are required:

- **QBall 2:** QBall 2 as shown in Figure 2.1
- **Router** A high-end router pre-configured to enable wireless connectivity to QBall 2
- **Batteries:** Two 3-cell, 2700 mAh Lithium-Polymer batteries
- **Real-Time Control Software:** The QUARC® Simulink® configuration, as detailed in Reference [2]

4.2 Diagram

Figure 4.1 below is a basic diagram of the QBall 2, showing the axes and angle. Note that the axes follow a right-hand rule with the x axis aligned with the front of the vehicle.

Caution

The tail or back of the vehicle is marked with colored tape. When flying the vehicle it is common to orient the vehicle such that the tail is pointing towards the operator with the positive x axis (front) pointing away from the operator.

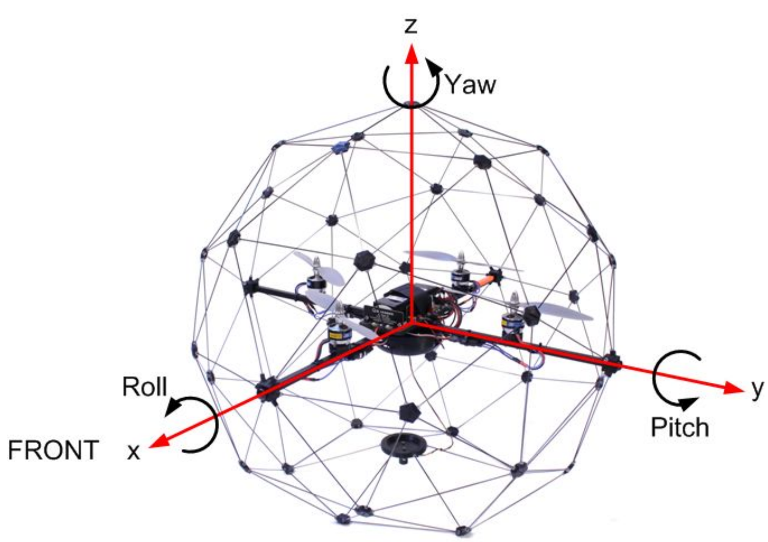
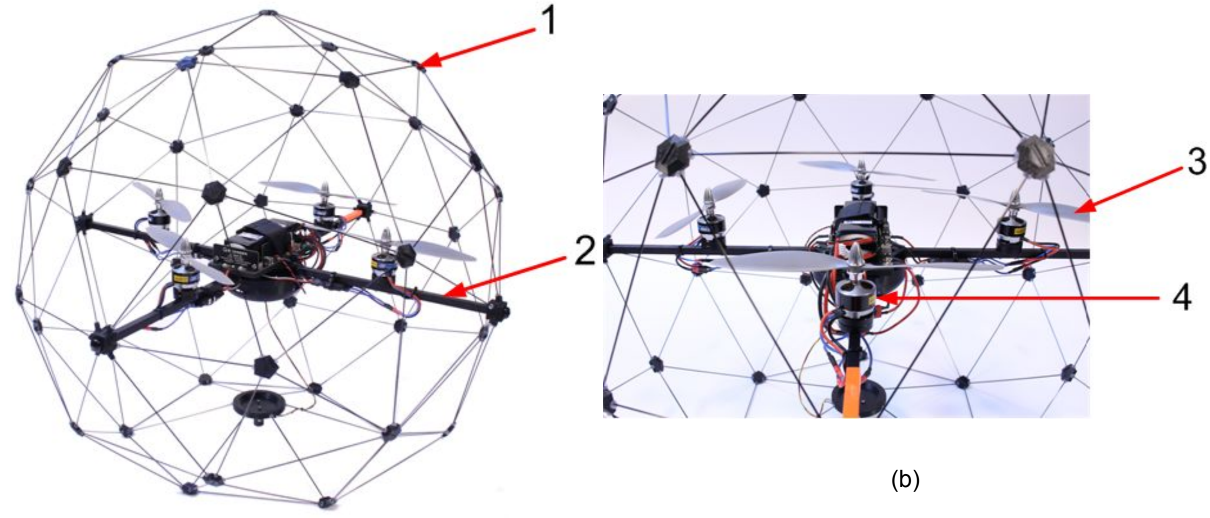


Figure 4.1: Quanser QBall 2axes and sign convention

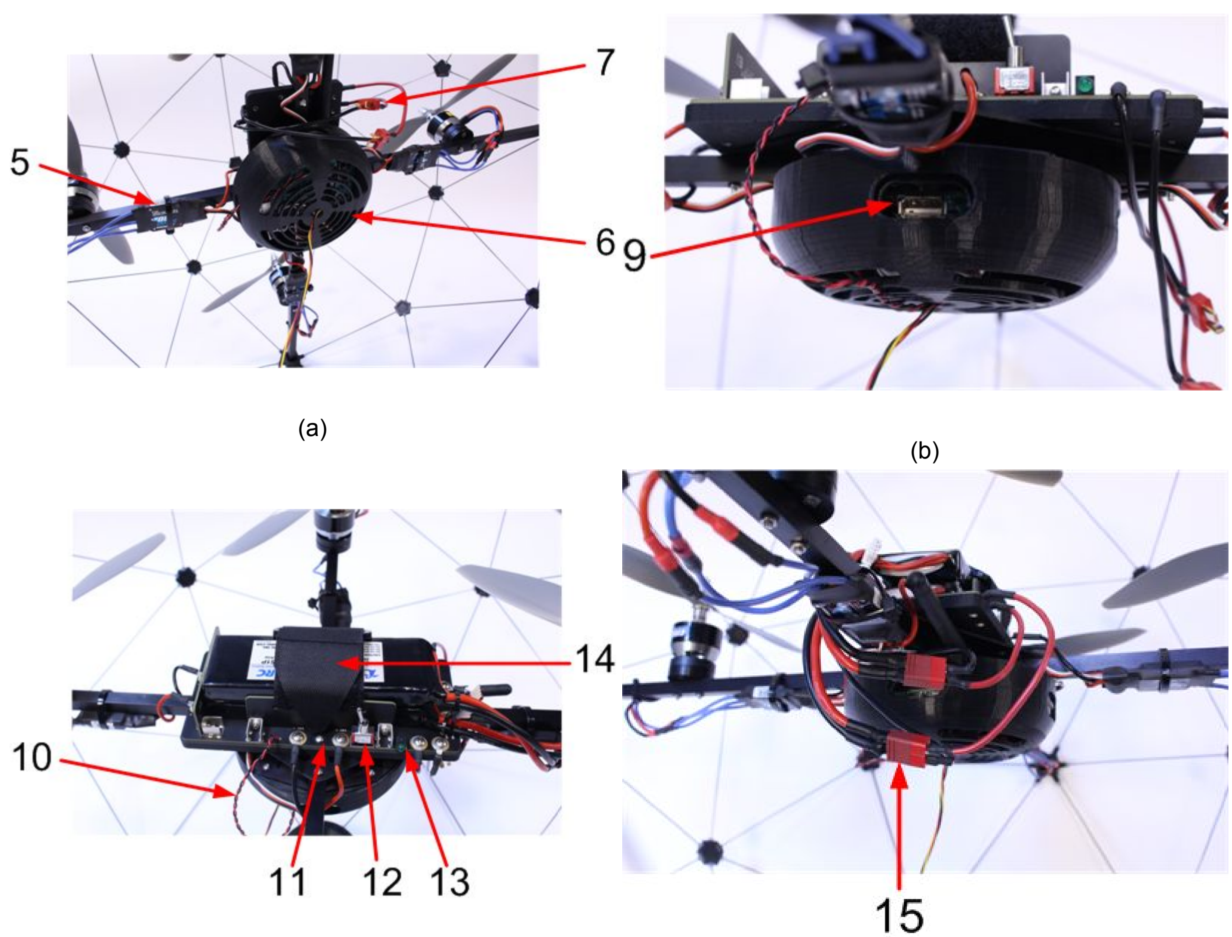
4.3 QBall 2 Components

The components comprising the QBall 2 are labled in Figure 4.2, Figure 4.3 and Figure 4.4 and described in Table 4.1. The QBall 2 joystick is illustrated in Figure 4.5.



(a) (b)

Figure 4.2: QBall 2 components: cage and propellers



(a) (b) (c) (d)

Figure 4.3: QBall 2 components: DAQ and power board

(a)

(b)

Figure 4.4: QBall 2 components: sonar and batteries

Figure 4.5: QBall 2 joystick

ID #	Description	ID #	Description
1	QBall 2 protective cage	9	USB input
2	QBall 2 frame	10	QBall 2 DAQ power cable
3	10x4.7 propeller	11	QBall 2 power distribution board
4	Brushless DC motor	12	QBall 2 power switch
5	ESC	13	QBall 2 power LED
6	QBall 2 DAQ	14	Battery velcro strap
7	Battery connector	15	Connected batteries
8	Sonar	16	QBall 2 LiPo batteries

Table 4.1: QBall 2 components

4.3.1 QBall 2 frame

The QBall 2 frame (#2 in Figure 4.2) is the crossbeam structure to which the QBall 2 components are mounted including the DAQ, power distribution board, motors and speed controllers. The frame rests inside the QBall 2 protective cage (#1 in Figure 4.2). The QBall 2's protective cage is a carbon fiber structure designed to protect the frame, motors, propellers, and embedded control module (DAQ and Gumstix computer) during minor collisions. The cage is not intended to withstand large impacts or drops from heights greater than 2 meters.

Caution

Do not pick up the QBall 2 from the cage as this may stress the cage and cause damage. Instead, when transporting the QBall 2 lift it from the ends of the frame using both hands to lift the frame from both sides.

4.3.2 QBall 2 DAQ

The DAQ is the QBall 2's data acquisition board. Together with the Gumstix embedded computer, the DAQ controls the vehicle by reading on-board sensors and outputting motor commands. The DAQ is located inside a protective enclosure underneath the cross frame of the QBall 2. The enclosure lid is opened by gently rotating the lid clockwise when viewed from the top.

Each motor speed controller (#5 in Figure 4.2) is connected to a PWM motor output on the DAQ (#6 in Figure 4.2). There are four motor output channels available on the DAQ and they are labeled F, B, L, and R to represent motor commands to the front, back, left, and right motors, respectively. Each motor speed controller should be connected to its corresponding PWM output with the ground (black wire) towards the inside of the DAQ board (see Section 6.2 for wiring details). The QBall 2 comes with the motors already connected to the DAQ, so no manual assembly is necessary.

If it is ever required to remove the DAQ for testing or troubleshooting, the cables and wires must first be disconnected from the DAQ PCB and removed from the DAQ enclosure. Remove the DAQ cover by gently rotating it clockwise (when viewed from the top of the QBall 2). Disconnect the power, sonar, motor, and any other I/O cables plugged into the DAQ and carefully extract them through the slots in the DAQ enclosure. Reattach the DAQ cover. Using an allen wrench, remove the four screws shown in Figure 4.6, being careful to support the DAQ enclosure so it does not fall. When reassembling the DAQ, first attach the DAQ enclosure to the frame, making sure to align the arrow on the DAQ enclosure with the front of the vehicle (positive X axis direction). Then, remove the lid and carefully feed the power, sonar, and motor cables through the enclosure slots. Reattach the cables to their corresponding headers and screw on the enclosure lid.

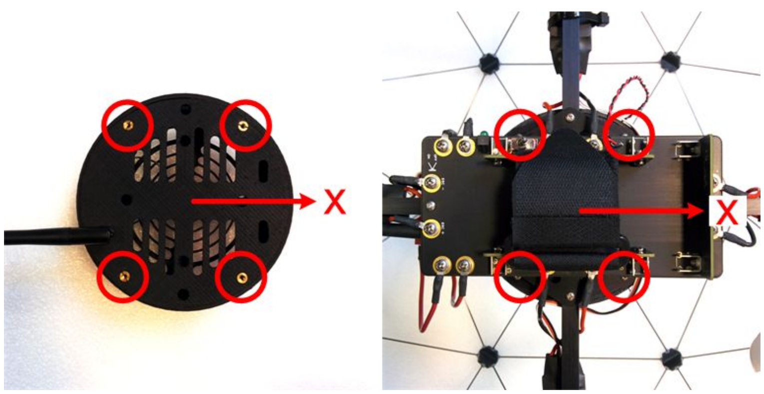


Figure 4.6: QBall 2 DAQ case mounting points

4.3.3 QBall 2 power distribution board

The QBall 2 uses two 3-cell 2700 mAh LiPo batteries (#16 in Figure 4.2) to power the DAQ and motors. These batteries are housed atop the QBall 2 power distribution board (#11 in Figure 4.2) above the QBall 2's cross frame and held in place using the provided velcro strap (#14 in Figure 4.2). The power distribution board connects both LiPo battery packs in parallel and routes power to the four motors as well as the DAQ.

Caution

Make sure the batteries are connected and secured with the velcro strap before attempting to fly the QBall 2.

Secure the batteries to the power board before connecting the batteries to the QBall 2 battery connectors (#7 in Figure 4.2) and always turn off the power using the QBall 2 power switch (#12 in Figure 4.2) before changing batteries.

Caution

LiPo batteries can be dangerous if charged improperly. Review the battery charging procedures (see Section 7) and monitor battery levels frequently during flight. The 3-cell LiPo batteries can become damaged and unusable if discharged below 10 V. It is recommended that the batteries be fully charged once they reach 10 V or less.

4.3.4 QBall 2 motors and propellers

The QBall 2 uses four E-Flite Park 480 (1020 Kv) motors [3] (#4 in Figure 4.2) fitted with paired counter-rotating APC 10×4.7 propellers [4] (#3 in Figure 4.2). The motors are mounted to the QBall 2 frame along the X and Y axes and connected to the four speed controllers [5], which are also mounted on the frame. The motors and propellers are configured so that the front and back motors spin clockwise and the left and right motors spin counter-clockwise (when viewed from the top). The electronic speed controllers (ESCs) receive commands from the controller in the form of PWM outputs from 1 ms (minimum throttle) to 2 ms (maximum throttle). Minimum throttle and maximum throttle are mapped to values between 0 and 1, respectively, using the HIL Write block to outputs the motor commands (see Section 6.2 for details on the HIL blocks). The ESCs used in the QBall 2 are configured with the appropriate throttle range during assembly. It is important that when the controller executes it initializes the ESCs by setting the motor outputs to the minimum throttle 0, otherwise you can enter the program mode and alter the ESC settings. Review the ESC's manual for instructions on changing ESC settings [5].

4.3.5 QBall 2 Joystick

The QBall 2 joystick is a critical component in operating the QBall 2. The joystick allows the operator to fly the QBall 2 using the controls for height (using the sonar to regulate the QBall 2 height), roll (rotating the QBall 2 about the x axis to fly left/right), pitch (rotating the QBall 2 about the y axis to fly forward/backward), and yaw (rotating the QBall 2 about the z axis to change its direction or heading). Even when flying the QBall 2 in autonomous modes with the provided controller the joystick is used to initialize and enable the QBall 2 and acts as a kill switch in the event the QBall 2 controller goes unstable and must be stopped. Section 6.5 describes how to use the joystick to fly the QBall 2 using the provided Simulink models.

5 QBall 2 Model

This section describes the dynamic model of the QBall 2. The nonlinear models are described as well as linearized models for use in controller development. For the following discussion, the axes of the QBall 2 vehicle are denoted (x, y, z) and are defined with respect to the vehicle as shown in Figure 4.1. Roll, pitch, and yaw are defined as the angles of rotation about the x, y, and z axis, respectively. The global workspace axes are denoted (X, Y, Z) and are defined with the same orientation as the QBall 2 sitting upright on the ground.

5.1 Actuator Dynamics

The thrust generated by each propeller is modeled using the following first-order system

$$F = K\frac{\omega}{s+\omega}u \tag{5.1}$$

where u is the PWM input to the actuator, ω is the actuator bandwidth and K is a positive gain. These parameters were calculated and verified through experimental studies and are stated in Table 5.1. A state variable, v, will be used to represent the actuator dynamics, which is defined as follows,

$$\nu = \frac{\omega}{s+\omega}u. \tag{5.2}$$

5.2 Roll/Pitch Model

Assuming that rotations about the x and y axes are decoupled, the motion in roll/pitch axis can be modeled as shown in Figure 5.1. As illustrated in this figure, two propellers contribute to the motion in each axis. The rotation around the center of gravity is produced by the difference in the generated thrust forces. From Equation 5.2, let $u = \tilde{u}$, where $\tilde{u}$ is the control input for the pitch or roll dynamics that causes an increase or decrease in thrust force in the two pitch/roll motors shown in Figure 5.1 and such that the changes in force of each motor are opposite in direction so that the net result is a torque. E.g., the control signal is applied to increase the force in motor 1 and decrease the force in motor 2. This change in motor forces is what causes the resulting torque and roll or pitch dynamics, so we can ignore the net thrust force used to hover the QBall 2. The change in thrust generated by each motor can be calculated from Equation 5.1. The roll/pitch angle, θ , can be formulated using the following dynamics

$$J\ddot{\theta} = \Delta F L, \tag{5.3}$$

where

$$J = J_{roll} = J_{pitch} \tag{5.4}$$

are the rotational inertia of the device in roll and pitch axes and are given in Table 5.1. L is the distance between the propeller and the center of gravity, and

$$\Delta F = \Delta F_1 - \Delta F_2 \tag{5.5}$$

represents the net change in the forces generated by the motors. Note that the difference in the forces is generated by the difference in the inputs to the motors, i.e.

$$\Delta u = 2\tilde{u}. \tag{5.6}$$

By combining the dynamics of motion for the roll/pitch axis and the actuator dynamics for each propeller the following state space equations can be derived

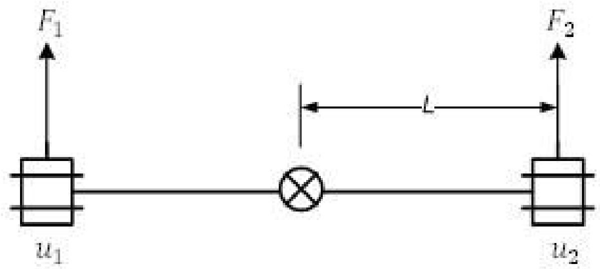


Figure 5.1: A model of the roll/pitch axis

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{\nu} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{2KL}{J} \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \nu \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \tilde{u}. \tag{5.7}$$

To facilitate the use of an integrator in the feedback structure a fourth state can be added to the state vector, which is defined as follows

$$\dot{s} = \theta. \tag{5.8}$$

After augmenting this state into the state vector, the system dynamics can be rewritten as

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{\nu} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{2KL}{J} \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \nu \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \tilde{u}. \tag{5.9}$$

5.3 Height Model

The motion of the QBall 2 in the vertical direction (along the Z axis) is affected by all the four propellers. The dynamic model of the QBall 2 height can be written as

$$M\ddot{Z} = 4Fcos(r)cos(p) - Mg, \tag{5.10}$$

where F is the thrust generated by each propeller, M is the total mass of the device, Z is the height and r and p represent the roll and pitch angles, respectively. The total mass, M, is given in the Table 5.1. As expressed in this equation, if the roll and pitch angles are nonzero the overall thrust vector will not be perpendicular to the ground. Assuming that these angles are close to zero, the dynamics equations can be linearized to the following state space form

$$\begin{bmatrix} \dot{Z} \\ \ddot{Z} \\ \dot{\nu} \\ \dot{s} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{4K}{M} & 0 \\ 0 & 0 & -\omega & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Z \\ \dot{Z} \\ \nu \\ s \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ -g \\ 0 \\ 0 \end{bmatrix}. \tag{5.11}$$

5.4 X-Y Position Model

The motion of the QBall 2 along the X and Y axes is caused by the total thrust and by changing roll/pitch angles. Assuming that the yaw angle is zero the dynamics of motion in X and Y axes can be written as

$$\begin{aligned} M\ddot{X} &= 4Fsin(p), \\ M\ddot{Y} &= -4Fsin(r). \end{aligned}$$

Assuming the roll and pitch angles are close to zero, the following liner state space equations can be derived for X and Y positions.

$$\begin{bmatrix} \dot{X} \\ \ddot{X} \\ \dot{\nu} \\ \dot{s} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{4K}{M}p & 0 \\ 0 & 0 & -\omega & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \\ \nu \\ s \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \\ 0 \end{bmatrix} u,$$

$$\begin{bmatrix} \dot{Y} \\ \ddot{Y} \\ \dot{\nu} \\ \dot{s} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-4K}{M}r & 0 \\ 0 & 0 & -\omega & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Y \\ \dot{Y} \\ \nu \\ s \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \\ 0 \end{bmatrix} u.$$

5.5 Yaw Model

The torque generated by each motor, τ, is assumed to have the following relationship with respect to the PWM input, u:

$$\tau = K_y u,$$

where K_y is a positive gain and its value is given in Table 5.1. The motion in the yaw axis is caused by the difference between the torques exerted by the two clockwise and the two counter-clockwise rotating propellers. The model of the yaw axis is shown in Figure 5.2.

The motion in the yaw axis can be modeled using the following equation

$$J_y\ddot{\theta}_y = \Delta\tau.$$

In this equation, θ_y is the yaw angle and J_y is the rotational inertia about the z axis, which is given in Table 5.1. The resultant torque of the motors, $\Delta\tau$, can be calculated from

$$\Delta\tau = \tau_1 + \tau_2 - \tau_3 - \tau_4$$

$$\begin{bmatrix} \dot{\theta}_y \\ \ddot{\theta}_y \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_y \\ \dot{\theta}_y \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_y}{J_y} \end{bmatrix} \Delta\tau$$

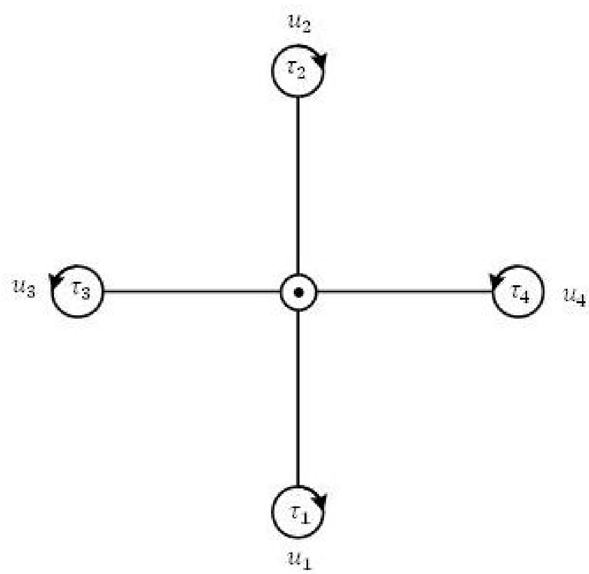


Figure 5.2: A model of the yaw axis with propeller direction of rotation shown.

Parameter	Value
K	120 N
ω	15 rad/sec
J_{roll}	0.03 kg.m^2
J_{pitch}	0.03 kg.m^2
M	1.79 kg
K_y	4 N.m
J_{yaw}	0.04 kg.m^2
L	0.2 m

Table 5.1: System parameters

6 System Setup

Section 6.1 describes setting up the vehicle hardware. Section 6.2 describes the QBall 2 sensors and how they are accessed in QUARC®. Section 6.3 and 6.4 describe the procedures for configuring the wireless connection in order to communicate with the QBall 2. Finally, Section 6.6 and Section 6.7 list the Matlab® Simulink® files provided with the QBall 2 and describe in detail the QBall 2 controller.

6.1 QBall 2 vehicle setup

1. First, make sure that the router is setup and connected to your PC (its network adapter). See Section 6.3 for network and IP settings.
2. Check that all motors are securely fastened to the vehicle frame. Check that the propellers are firmly attached to the motors in the correct order: clockwise propellers (viewed from the top) on the front and back motors, counter-clockwise propellers on the left and right motors. Note that the back motor is indicated by a bright colored marking tape on the QBall 2 frame.

   Caution

   Check that the motors are firmly secured to the frame regularly (after every 2 hours of flight). Over time, vibrations in the frame may loosen the motor mounts. If a motor or mount feels loose, tighten it immediately.

   If a propeller is loose, use an allen key to remove the cap holding the propeller to the motor and ensure the propeller mounting collar is pushed fully down onto the motor shaft. Replace the propeller on the mounting shaft and replace the motor cap and tighten it with an allen key. **Never change propellers or other components of the QBall 2 with batteries connected.**
3. Install the batteries as illustrated in Figure 6.1. With the power switch in the off position, insert the batteries into the battery compartment on top of the power distribution board, making sure the batteries rest against the far wall of the compartment. Secure the batteries in place using the velcro strap. Connect the batteries to the power board battery connectors.
4. Power on the QBall 2 using the power switch located on the power distribution board. After approximately 1 minute the Gumstix wireless module should be active and connected to the WiFi (see Section 6.3 for network setup).

6.2 QBall 2 sensors

This section describes the blocks that are used to read the QBall 2 sensors in Simulink® and write outputs to the motors. The QUARC® Hardware-In-the-Loop (HIL) blockset it used to communicate with Quanser® data acquisition cards. For detailed information on the HIL blockset see the QUARC® HIL user guide in the Matlab® help under QUARC Targets/User's Guide/Accessing Hardware.

The QBall 2 DAQ provides several high-resolution avionics sensors, which are used to measure and control the stability of aerial vehicles. The I/O of the QBall 2 DAQ includes:

- 4 PWM motor outputs
- 2 configurable PWM outputs
- 3-axis gyroscope, $250/500/2000$ degrees-per-second selectable range
- 3-axis accelerometer, $\pm 2g/4g/8g$ selectable range
- Sonar height sensor, $0.2 - 7.65$ m range, 1 cm resolution

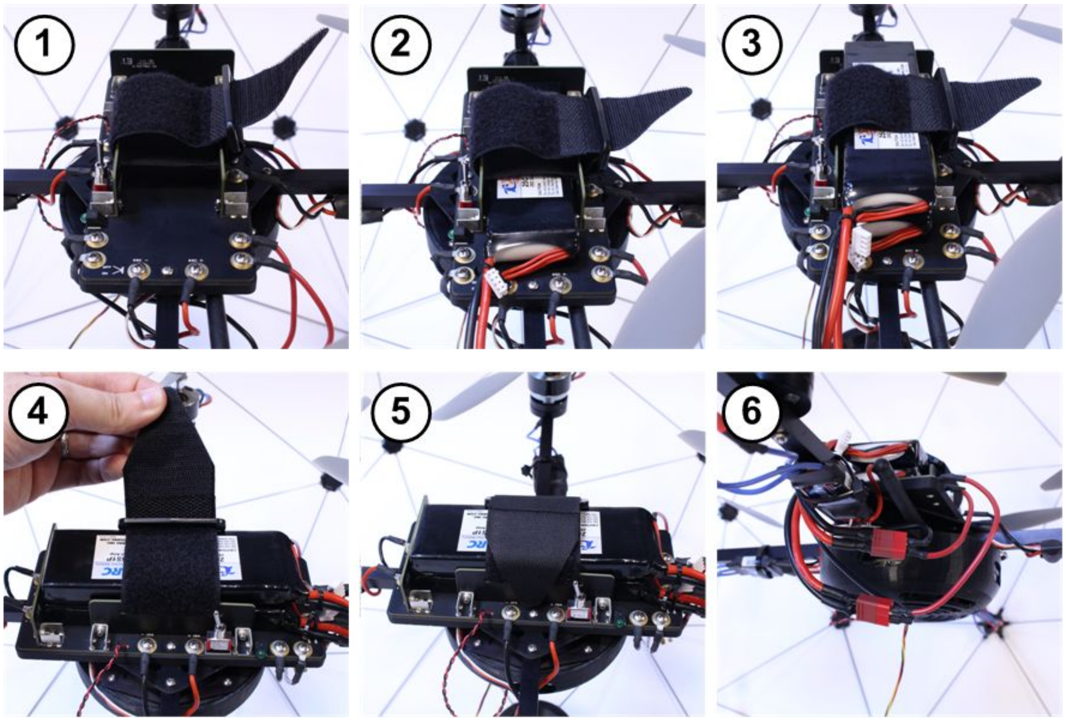


Figure 6.1: Battery connection procedure

- Battery voltage measurement
- 2 analog inputs, 12-bit, $0 - 5$ V
- 1 SPI
- 8 digital I/O
- 1 UART
- 1 I^{2}C

Figure 6.2 shows the location of the I/O listed above on the QBall 2 DAQ. The DAQ I/O listed above is accessed using the QUARC HIL blockset. The UART, SPI, and I2C communication channels are accessed through the QUARC® Stream blockset. For more information on accessing communication stream data see the QUARC® help under QUARC Targets/User's Guide/Communications. Table 6.1 lists the HIL blocks used to communicate with the QBall 2's data acquisition hardware.

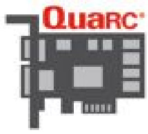
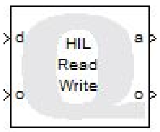
Block	Description
 HIL Initialize QBall 2 (qball2-0)	The HIL Initialize block selects the DAQ board and configures the board parameters. The HIL Initialize block is named via the Board name parameter, and all other HIL blocks reference the corresponding HIL Initialize through its name. The HIL blocks will interface to the DAQ specified in the HIL Initialize Board type parameter (qball2).
 HIL Read Write1 (QBall 2)	The HIL Read Write block is used to read sensor measurements from the DAQ and write motor commands to the four QBall 2 motors. The inputs and outputs are specified with numeric channel numbers given in Table 6.2 and Table 6.3, respectively.
 HIL Watchdog (QBall 2)	The HIL Watchdog block is used to set the timeout limit for the watchdog timer. For the QBall 2 DAQ board, if there is no motor output command received for a consecutive period of time exceeding the watchdog timeout value then the watchdog will trigger, forcing the motor outputs to 0. The default timeout value for the watchdog is 50ms unless specified otherwise with this block. This block can be used to change the timeout value if 50ms is not suitable.

Table 6.1: HIL blocks

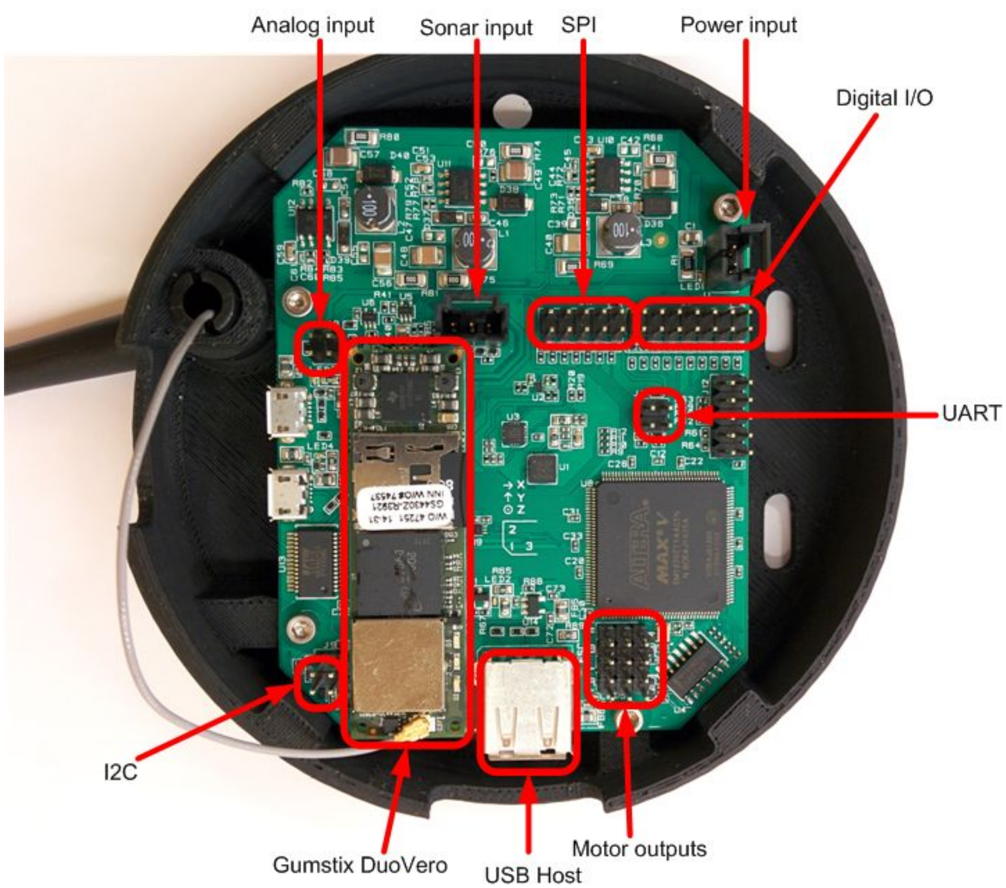


Figure 6.2: QBall 2 DAQ board.

To initialize the QBall 2 DAQ board, a HIL Initialize block must be placed in the model. The HIL Initialize block is used to initialize a data acquisition card and setup the I/O parameters. In the HIL Initialize block, select the board type 'qball2' to configure the QBall 2 DAQ and, if desired, enter a name in the Board name field as shown in Figure 6.3.

Next, to read and write from the QBall 2 DAQ, add a HIL Read Write block to the model (note that the QBall 2 DAQ is optimized for best performance when a single HIL Read Write block is used in a model, adding more HIL I/O blocks may reduce the performance, particularly the maximum sample rate). In the HIL Read Write block, select the board name corresponding to the board name given in the HIL Initialize block. The channels available for reading and writing for the DAQ are listed in Table 6.2 and Table 6.3 below. Enter the channel numbers to be read/written or use the browse buttons to open a channel selection dialog as shown in Figure 6.4.

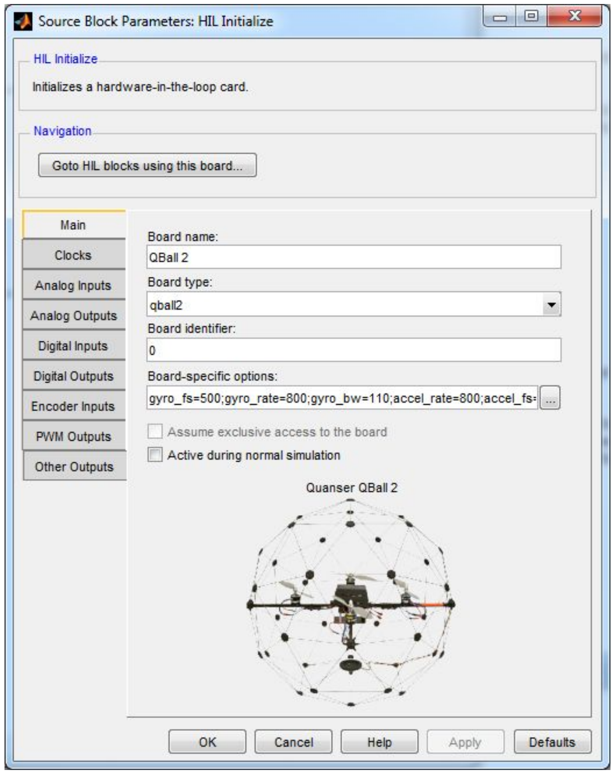


Figure 6.3: HIL Initialize block with the QBall 2 board selected.

Channel type	Read channel numbers	Description	Units
Analog	$0-1$	Analog inputs	V
	2	Supply voltage (battery)	V
Encoder	$none$	-	
Digital	$0-7$	Reconfigurable digital I/O	
Other	0	Sonar height sensor	m
	$3000-3002$	3-axis gyroscope (x, y, z)	$\mathrm{rad/s}$
	$4000-4002$	3-axis accelerometer (x, y, z)	$\mathrm{m/s^2}$
	10000	Temperature sensor	$^\circ\mathrm{C}$

Table 6.2: QBall 2 input channels

Channel type	Write channel numbers	Description	Units
Analog	$none$	-	
PWM	$0-1$	PWM outputs	% duty cycle from 0-1
Digital	$0-7$ 8	Reconfigurable digital I/O LED	
	9	ESC enable	
Other	11000 11001 11002 11003	Left motor Right motor Front motor Back motor	Throttle from 0-1 Throttle from 0-1 Throttle from 0-1 Throttle from 0-1

Table 6.3: QBall 2 output channels

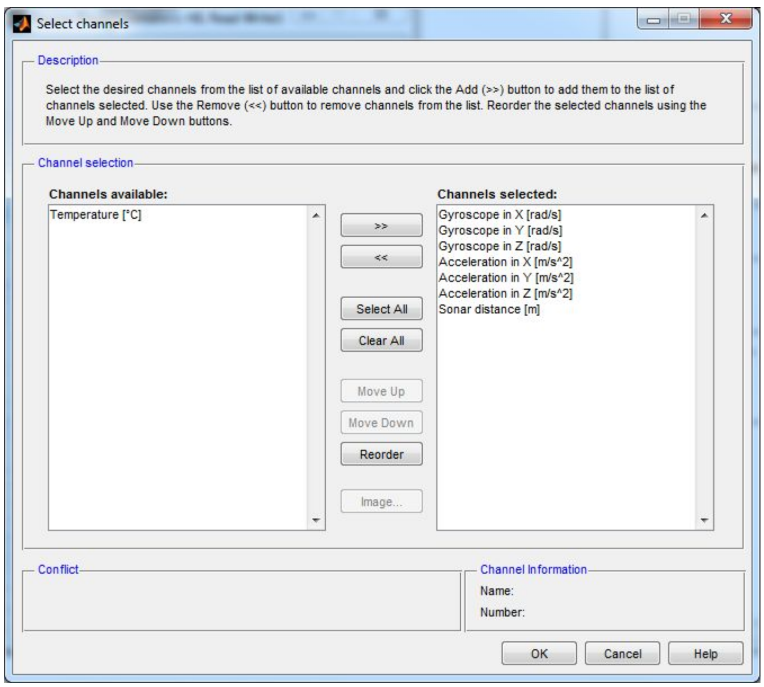


Figure 6.4: Channel selection dialog for the HIL Read Write block

For the QBall 2, the Other output channels $11000 - 11003$ are used to command the front, back, left, and right motors, respectively. The range of the motor output values is 0 to 1 (minimum throttle to maximum throttle), which corresponds to a 1ms to 2 ms PWM pulse, respectively. A command of 0 corresponds to zero throttle, which will cause the motors to stop.

The 3-axis gyroscope and accelerometer measurements are used to measure the QBall 2 dynamics and orientation (roll, pitch and yaw). These IMU inputs are crucial for controlling the flight of the QBall 2. The QBall 2 DAQ utilizes a STMicroelectronics 3-axis gyroscope [6] and a Freescale 3-axis accelerometer [7]. The QBall 2 sonar sensor is the Maxbotix XL-Maxsonar EZ3, which measures distances between 20cm and 765cm with 1cm resolution. Objects between 0-20cm are ranged as 20cm. The sonar sensor is positioned at the bottom of the QBall 2 and is used to measure the QBall 2 height for closed-loop height control.

Caution

Note that the sonar works best over a hard surface which will reflect the ultrasonic signals. The sonar may not work over carpet or other surfaces that will disperse the ultrasonic signals. Always test the sonar first by disabling the QBall 2 motor outputs and lifting the QBall 2 to see if the sonar is functions as expected.

The battery voltage input measures the supply voltage connected to the QBall 2 DAQ. Since the LiPo batteries used to power the QBall 2 should be charged when they reach a voltage of no less than 10 V, the battery capacity should be monitored.

Caution

It is recommended that the QBall 2 batteries are always changed in pairs. Follow the directions of the charging system that is supplied to ensure the batteries are charged properly and safely (see Section 7).

The QBall 2 DAQ provides several I/O channels for interfacing additional sensors. Figure 6.2 shows the QBall 2 DAQ and its interfaces. Figure 6.5 shows the layout of the QBall 2 DAQ header pins and Table 6.4, Table 6.5 and Table 6.6 list the various I/O pins found on the QBall 2 DAQ.

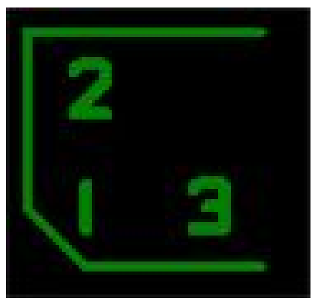


Figure 6.5: Pin mapping for the DAQ headers

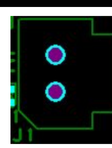
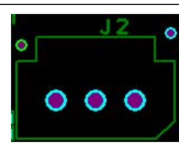
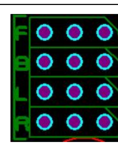
I/O	Header	Pin	Signal
Power input	J1 	1	V_{in}
		2	GND
Sonar input	J2 	1	5V
		2	sonar height measurement
		3	GND
Motor outputs	ESC F/B/L/R 	1	PWM motor output
		2	Not connected
		3	GND

Table 6.4: QBall 2 DAQ pin list: J1, J2, ESC F/B/L/R

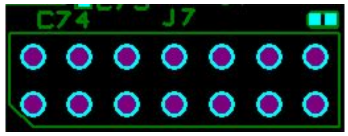
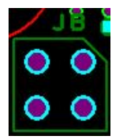
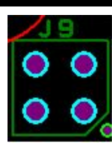
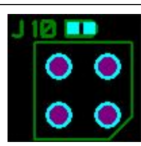
I/O	Header	Pin	Signal
Digital I/O	J7 	1	DIO channel 7
		2	DIO channel 3
		3	DIO channel 6
		4	DIO channel 2
		5	DIO channel 5
		6	DIO channel 1
		7	DIO channel 4
		8	DIO channel 0
		9	PWM output channel 5
		10	PWM output channel 4
		11	GND
		12	3.3V
		13	GND
		14	3.3V
Analog inputs	J8 	1	Analog input channel 6
		2	GND
		3	Analog input channel 2
		4	5V

Table 6.5: QBall 2 DAQ pin list: J7, J8

I/O	Header	Pin	Signal
I2C	J9 	1	GND
		2	SDA
		3	3.3V
		4	SCL
UART	J10 	1	GND
		2	RX
		3	3.3V
		4	TX

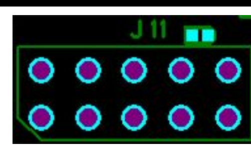
I/O	Header	Pin	Signal
SPI	J11 	1	SOMI
		2	CS3
		3	SIMO
		4	CS2
		5	CLK
		6	CS1
		7	GND
		8	CS0
		9	GND
		10	3.3V

Table 6.6: QBall 2 DAQ pin list: J9, J10, J11

6.3 Establishing Network Connection

The QBall 2 package comes with a pre-configured wireless router and automatically connects to the wifi network Quanser_UVS. It uses TCP/IP connection for communicating with the host computer and/or other Quanser®

unmanned vehicles. The Host PC and each of the vehicles must have unique IP addresses and the range of these addresses are defined below (suggested):

Host PC(s)	192.168.2.10 to 192.168.2.19
Quanser vehicles (Gumstix)	192.168.2.20 to 192.168.2.254

These Steps outlined below for setting up the host computer wireless connection only need to be performed once.

1. Power up and turn on the wireless router.
2. After turning on the router that is provided, wait for about 60 seconds for the wireless network to establish and then turn on the QBall 2 power and wait for it to boot up (approximately 60 seconds).
3. Connect your PC network card to any of the ports of the router (e.g. port number 1 to 4) using the network cable provided (you can also connect to the Quanser_UVS or Quanser_UVS-5G wireless network if your computer has wireless adapters, however, wired connection between your PC and the router is preferred for better performance). If you choose wireless connectivity between your PC and the router, you should use the password UVS_wifi to connect to the wifi network.
4. Using the *Windows Network Connection* utility in the *Windows Task Bar*, open the Network and Sharing Center.

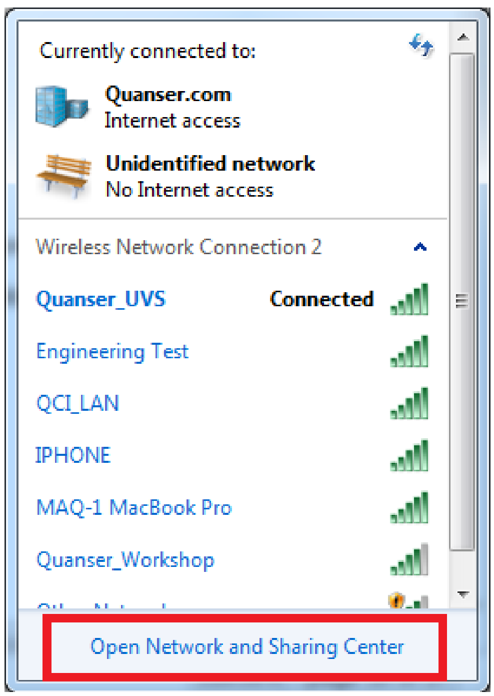


Figure 6.6: Open network settings

5. Click on Change adapter settings.
6. Right-click on the Local Area Connection x, Unidentified network connection and click Properties. If you chose wireless connectivity between your PC and the router, choose the Wireless Network Connection instead, right click on it and click Properties.
7. Under *This connection uses the following items:*, select *Internet Protocol Version 4 (TCP/IPv4)* and click Properties.

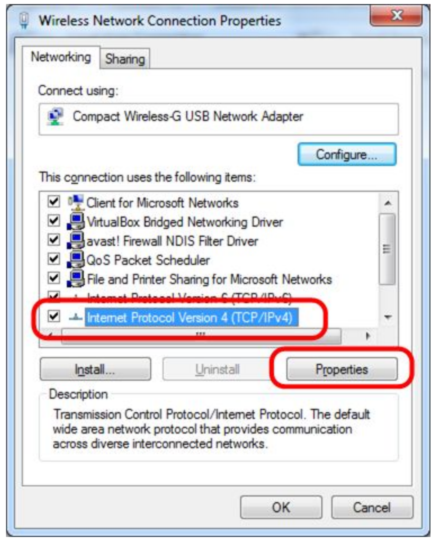


Figure 6.7: Network properties

8. Instead of obtaining an IP address of the computer automatically, select *Use the following IP address* and enter the following:
   IP address: $192.168.2.10$ (For multiple host PCs, use different IP addresses within the valid range)
   Subnet mask: $255.255.255.0$

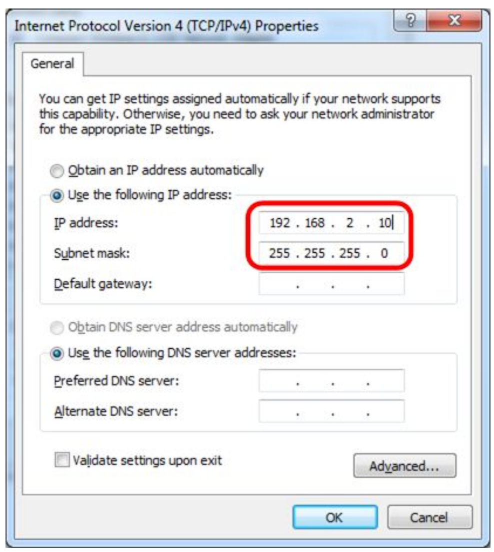


Figure 6.8: IP settings

9. Make sure you can ping the router by typing ping 192.168.2.1 in the Run box in Windows (go to the *Start* menu and search for Run and click Run, Figure 6.9). If the connection to the router is successful you will see the ping replies in the command window. If you cannot ping the router, check network connectivity and your IP address before going to the next steps.

10. If the QBall 2 is powered on, the QBall 2 can be pinged by typing ping {IP of the QBall 2} in the Run box in Windows (go to the *Start* menu and click Run). If the connection is successful you will see the ping replies in the command window.

    Note: You may need ti disable Windows firewall to establish a connection.

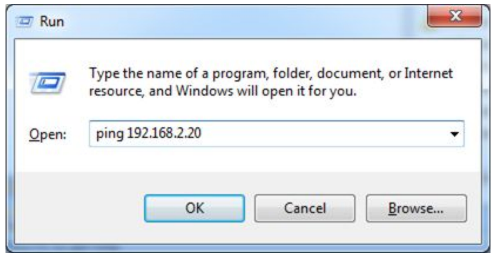


Figure 6.9: Pinging the QBall 2

6.4 Configuring models for the QBall 2 target

Note: this section applies only to files that are run on the Gumstix target (i.e., on the QBall 2) such as `qball_2_control_v1.mdl` (see Table 6.7 in Section 6.6). Simulink® should have a new menu item called QUARC® once QUARC® has been installed. The following steps are required to setup a new QUARC® model for the QBall 2:

1. Create a new Simulink® model, or open an existing model to be run on the Gumstix.
2. Click on the *QUARC* menu, then select Options.
3. The System target file under *Code Generation* should be `quarc_linux_duovero.tlc`. Browse through the system target list to locate the proper file if necessary (Figure 6.10).

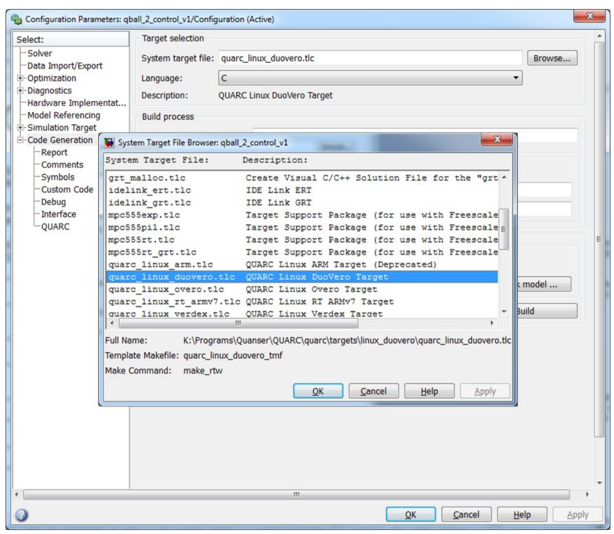


Figure 6.10: QUARC® Option Menu

4. In order to run the QUARC® model on the target vehicle, the target's IP address must be specified. To setup the default target address for **all linux-duovero targets**, go to the *QUARC* menu and select Preferences. The *Target type* parameter should be set to `linux_duovero`. Replace the *Default Model URI* with the IP address of the desired target vehicle, e.g., `tcpip://192.168.2.20:17001`.

   Alternatively, to set the target address for the **current model only** open the model options under the *QUARC | Options* menu and choose *Code Generation* > Interface on the left hand pane. Under the MEX-file arguments, type `'-w -d /tmp -uri %u', 'tcpip://{IP of Gumstix}:17001'`. Include the single quotation marks (Figure 6.11). Replace {IP of Gumstix} with the IP of your Gumstix, e.g. `'tcpip://192.168.2.20:17001'`.

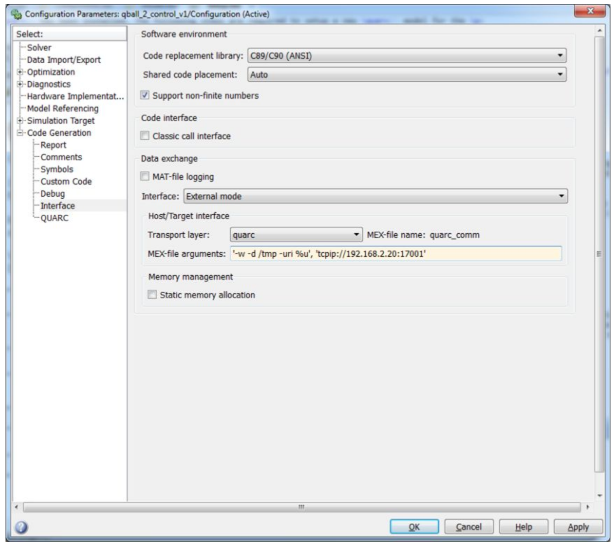


Figure 6.11: Model target IP settings

5. Select External for simulation mode, instead of Normal, which indicates that the model is to be run on the target machine (Gumstix) rather than simulating the model on the host machine.

6. The model is now ready to be compiled and downloaded to the target. If the wireless connection to the vehicle has been established, a QUARC® console can be opened to show additional messages and progress during model compilation by going to the menu item *QUARC* | Console for all. Building the model (*QUARC* | Build) will begin the code generation and compiling steps. Output from the compilation is shown in the QUARC® console. This step may take a few minutes to complete.

6.5 Joystick

The joystick is an integral part of the QBall 2 and must be used in every test flight. Even in closed-loop control modes, the joystick is still used as an enable/stop switch for safety reasons. The joystick has one control stick that operates the pitch, roll, and yaw commands as well as an analog slider to enable takeoff and landing. The mapping of the control stick is shown below in Figure 6.12. The stick controls the pitch and roll (down->up is pitch backwards -> forwards, left-> right is roll left->right) and yaw rate by rotating the stick about the vertical axis. The joystick commands are always given in the reference frame of the QBall 2, therefore it is recommended to control the QBall 2 with the QBall 2 facing away from the operator, so he/she is viewing the QBall 2 tail.

Before running any QBall 2 controllers, calibrate your joystick using the Windows game controller calibration. Build and run the `host_joystick_logitech_extreme3d_pro.mdl` model for your configuration. Verify using the host joystick model that the joystick is responding and covering the correct range for each control stick.

6.6 Simulink Files

To operate the QBall 2 there are several files needed. Table 6.7 lists the various files and their purpose.

The QBall 2 operates using a host-target structure. The host machine (ground station PC) runs one host model to stream joystick (and possibly localization) data to the target QBall 2. The target computer is the QBall 2 Gumstix DuoVero, which executes the QBall 2's controller (`qball_2_control_v1.mdl`). Various scripts are run upon opening the QBall 2 controller model to initialize the controller parameters. If the Matlab® workspace is cleared and these parameters are no longer stored, simply run the `setup_qball_2.m` script to reinstate the configuration parameters.

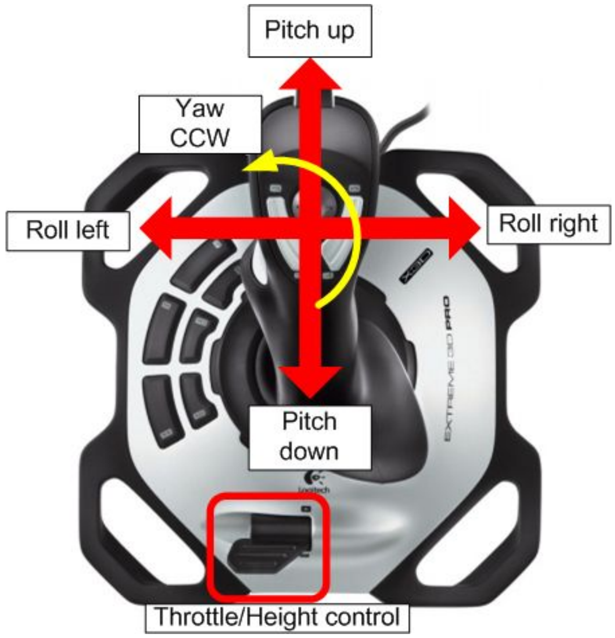


Figure 6.12: Joystick control mapping

File name	Description
setup_qball_2.m	A Matlab® script that is run whenever the QBall 2 controller model is opened. This script runs the controller_design.m script and sets up any other model parameters.
controller_design.m	A script used to compute the LQR controller gains used in stabilizing the QBall 2's orientation and configures other controller gains for controlling the QBall 2 height and vehicle position. This file is run by the setup_qball_2.m script.
host_joystick_logitech.mdl	A Simulink model used on the host PC to stream joystick data to the QBall 2 for joystick control.
host_joystick_logitech_optitrack.mdl	A Simulink model used on the host PC to stream joystick and OptiTrack data to the QBall 2 for joystick control and autonomous position control.
plot_log_data.m	A script used to plot experiment data from a MAT file saved by the qball_2_control_v1.mdl controller.
qball_2_control_v1.mdl	This file contains the main flight controller for the QBall 2. This model is downloaded and executed on the QBall 2 and controls the stability of the QBall 2 orientation and, if localization information is available, the QBall 2 position. Opening this file automatically runs the setup scripts above.

Table 6.7: Simulink® files used to operate the QBall 2

Note that files highlighted in grey in Table 6.7 indicate that the model must be targeted and run on the quarc_linux_duovero (Gumstix) target; all other files are run on the Windows target.

Caution **If you are using an older Gumstix target (`quarc_linux_verdex` or `quarc_linux_arm`), you cannot run the QBall 2 controller files on these targets. Please use the controller that was provided with your target.**

6.7 The QBall 2 Controller

The main QBall 2 controller file contains several operating modes and various subsystems responsible for stabilizing the vehicle. Figure 6.13 illustrates the QBall 2 controller's top level and the various subsystems contained in the model. A detailled descriptions of the functionality of the subsystems used to command the QBall 2 and change its operating modes can be found in the following subsections.

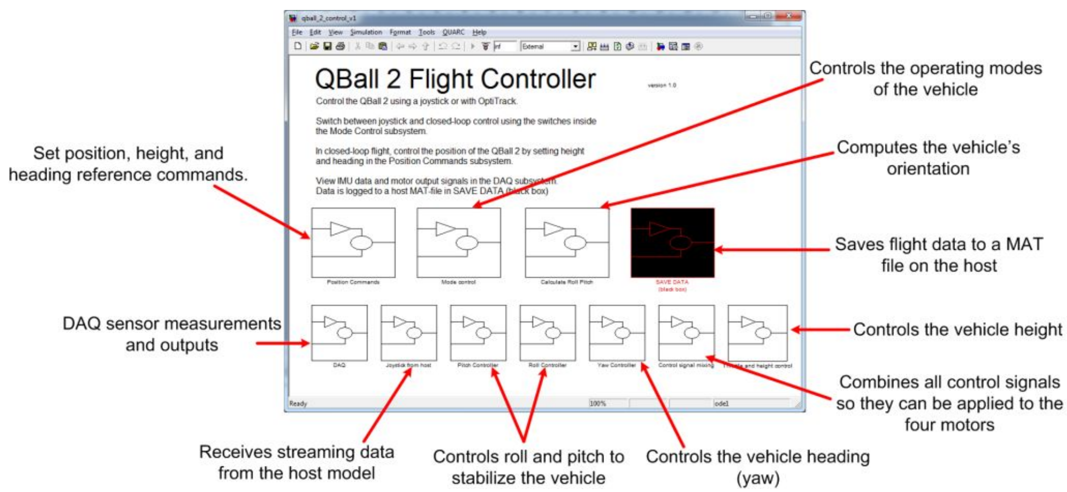
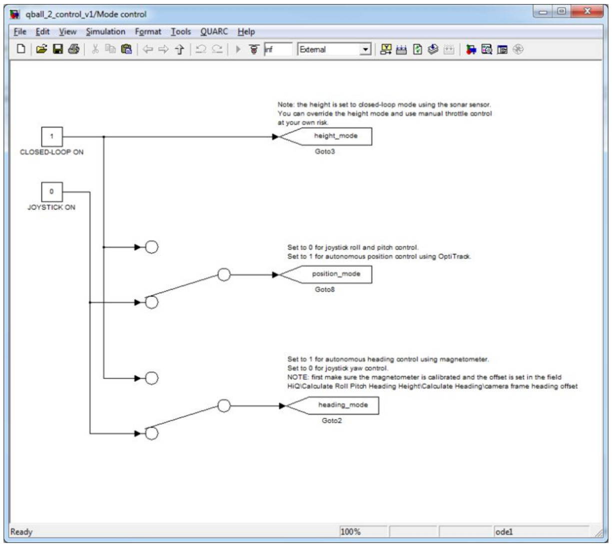


Figure 6.13: The main QBall 2 controller subsystems.

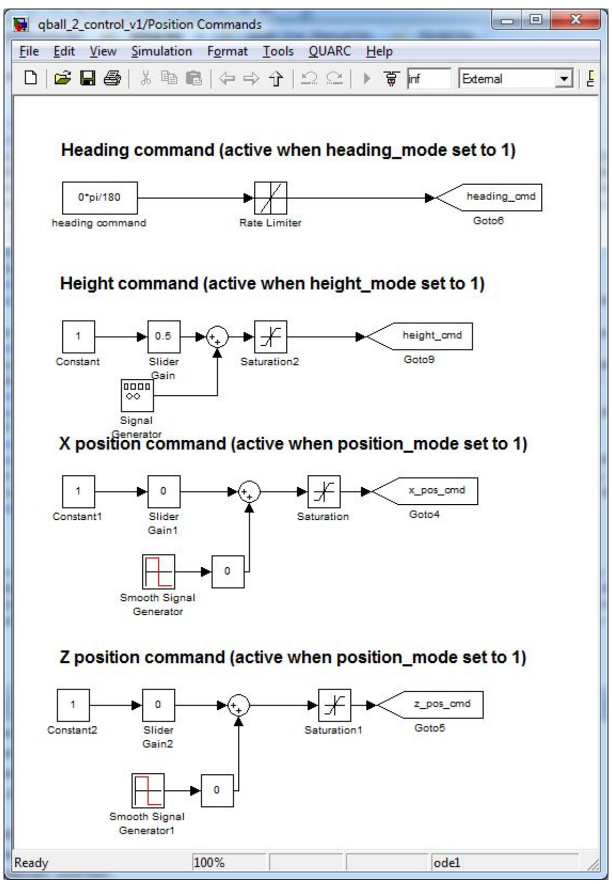
Note that even in closed-loop operating modes, the host model must still be connected and streaming joystick data to the QBall 2 since the joystick acts as a safety switch. In closed-loop mode the joystick throttle disables the QBall 2 flight when it is below 10% and enables takeoff and flight when the throttle is above 10%. This is a safety feature so that users can quickly disable the QBall 2 during an experiment if desired. When disabling the QBall 2 using the joystick, the QBall 2 will attempt to land safely using the sonar height controller; after 4 seconds the motors will be turned off. Stopping the QBall 2 controller directly through Simulink will force the motors to the final motor output values set using the HIL Initialize block in the DAQ subsystem, which in this model is set to the minimum throttle value 0; this disables the motors immediately.

6.7.1 Mode Control



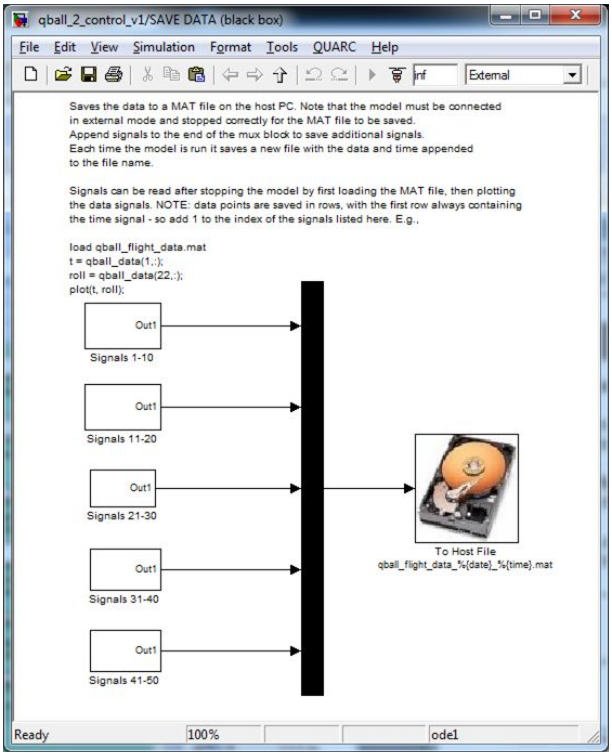
This subsystem is used to change the operating mode of the QBall 2 from open-loop joystick control to closed-loop autonomous control. There are 2 switches used to change modes for position and heading control. Operating modes should only be changed in between experiments when the model is not running or the system may go unstable. Note that the height mode is fixed to closed-loop, which forces the QBall 2 controller to control the height using the on-board sonar sensor. You may choose to override the height mode and use the joystick throttle input to manually control the QBall 2 throttle, but this is a more challenging mode of operation and should only be attempted if you have experience flying quadrotor helicopters and wish to do so at your own risk. The closed-loop position and heading modes require a localization system such as OptiTrack to control the position and heading of the QBall 2. Joystick modes allow control of the roll and pitch (position) and yaw (heading) from the joystick.

6.7.2 Position Commands



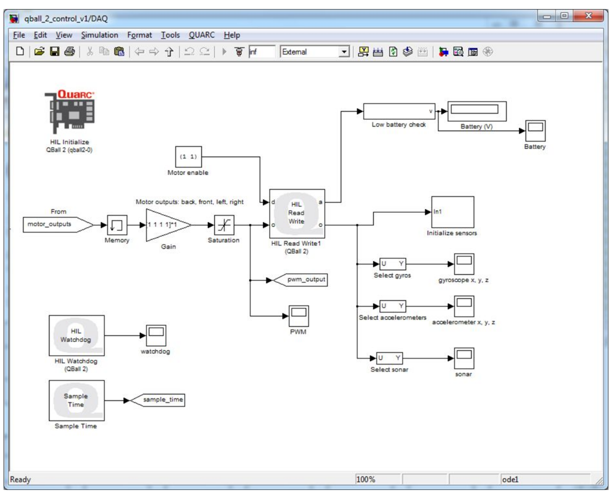
This subsystem contains the reference commands (set points) for the QBall 2 heading, height, and horizontal position. These commands are the reference points used in closed-loop operating modes. If the QBall 2 is operated in Joystick mode then these commands have no effect.

6.7.3 Save Data (black box)



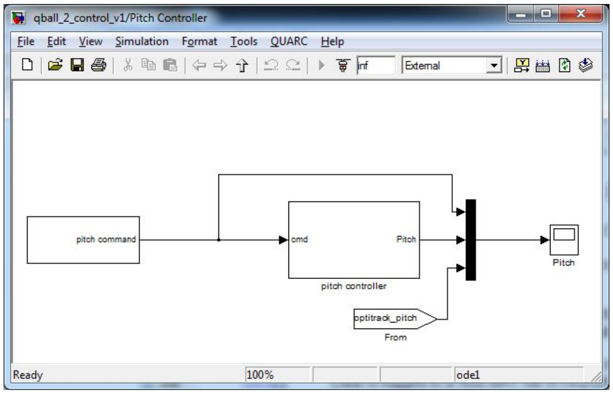
This subsystem collects data from the QBall 2 model to save to a MAT file on the host using the QUARC To Host File block. The `plot_log_data.m` file can be used to extract and plot some of the data saved once the MAT file has been loaded in Matlab®. See the help in QUARC® for the To Host File block for more information. Note that additional data from the QBall 2 model can be added to the signals in order to save additional information as desired.

6.7.4 DAQ



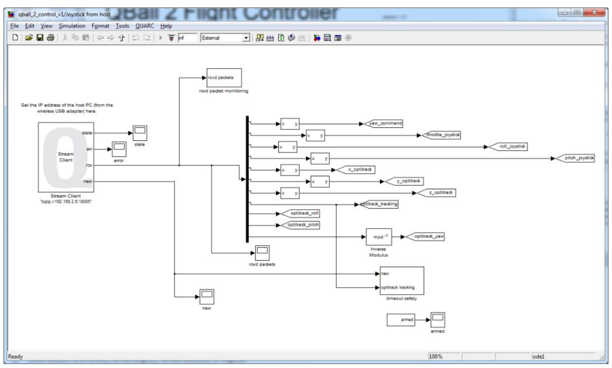
The DAQ subsystem contains the Hardware-In-the-Loop (HIL) blocks used to configure the QBall 2 DAQ and read-/write values. The gain block is used to enable or disable the QBall 2 motors by multiplying the motor signals by 1 or 0, respectively. The digital output signal is also used to enable the ESCs (1) or disable the ESCs (0). When testing the QBall 2 sensors or handling the QBall 2, disable the motors for safety by multiplying the motor signals by 0 using the gain block and disable the ESCs using the digital enable signal.

6.7.5 Pitch/Roll Controller



These subsystems contain the reference commands (either from joystick inputs or from the Position Commands subsystem depending on the operating mode) and controller used to stabilize the roll and pitch of the QBall 2.

6.7.6 Joystick from host



This subsystem receives streaming data from the host model. **Ensure the IP address in the Stream Client URI matches the IP address of the host computer so that the QBall 2 controller can connect to the host model.** The data packet contains joystick data and OptiTrack localization data if available. If the operating mode is set to closed-loop position mode then this subsystem monitors the status of the localization tracking and issues a land if the tracking is invalid for an extended period of time. Also, a land is issued if for any reason there is no communications from the host for 1 consecutive second.

6.8 OptiTrack Localiztion System

When operating the QBall 2 in closed-loop position or heading control mode, a localization system is required to provide pose feedback. In the supplied controllers, the host OptiTrack Simulink® models stream the localization data from an OptiTrack system to the QBall 2. To use this feature, you must have an OptiTrack camera system and have the OptiTrack Motive software installed (see the OptiTrack Quick Start Guide).

Caution

Make sure to check the QUARC OptiTrack block help pages for the most up to date version of Motive that is recommended. Using a different version of Motive than what is recommended may not function or may cause the QBall 2 to become unstable due to different axis conventions across Motive versions. At the time of writing of this user manual, it is recommended to use Motive version 1.7.1 Final.

The axes of the OptiTrack workspace are arranged in a specific orientation. With respect to the operator, the positive X axis points to the right, the positive Z axis points toward the operator and ground station, and the positive Y axis points upward with the origin located on the ground in the center of the workspace as shown in Figure 6.14. This is the workspace frame expected by the QBall 2 controller. Typically, the workspace frame will undergo a transformation in the host model that interfaces with the localization system so that it conforms to this desired orientation.

In order to configure the localization system axes for the OptiTrack localization system, you must set the ground plane after calibrating the cameras. The ground plane setup for OptiTrack is shown in Figure 6.15.

QUARC® supports several localization and tracking systems including: OptiTrack, Vicon, and PhoeniX Technologies. Make sure your chosen localization system is setup with the axes convention described above if you are using the provided closed-loop position controllers. Tracking multiple vehicles or objects requires more advanced configurations and is not covered in the supplied controllers but can be achieved by utilizing the tools

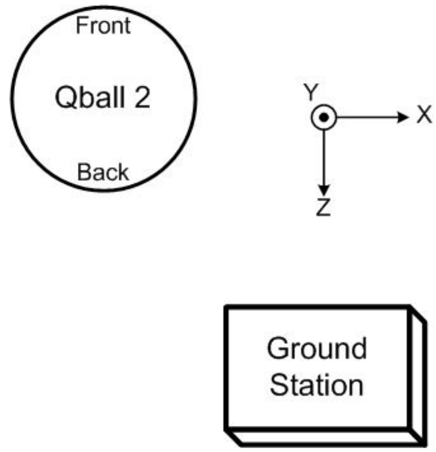


Figure 6.14: The localization system's coordinate frame.

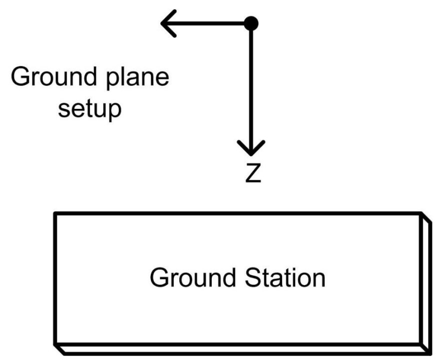


Figure 6.15: Ground plane setup for the OptiTrack localization system.

provided.

The localization systems supported in QUARC® typically track marker positions or rigid body objects (rigid geometric shapes made of 3 or more tracking markers, also refered to as trackables in this document). For operating the QBall 2 in closed-loop modes, the OptiTrack localization system is used to provide measurements of the QBall 2 position and orientation. The closed-loop position controller assumes there is a rigid body object placed on the top of the QBall 2's protective cage and configured such that the pivot point (or center) corresponds to the position of the center of rotation of the QBall 2 where crossed frame is located. The rigid body also provides measurement of the rotation about the vertical Y axis to control the QBall 2 heading. To setup the rigid body on the QBall 2, the OptiTrack system must already be calibrated using the OptiTrack Motive software; if it is not calibrated refer to the OptiTrack Quick Start Guide. With the cameras calibrated, follow these steps to create a QBall 2 rigid body object:

1. Add tracking markers to the top of the QBall 2 protective cage in a fixed, unique pattern, as shown in Figure 6.16. At least 3 markers must be used and additional markers provide better redundancy and robustness to occlusion. This procedure assumes there is a marker on the top-most node of the cage, which allows us to more easily translate the pivot point (center) of the rigid body. Other configurations can be used as long as the pivot point is appropriately translated to the center of the QBall 2.
2. Load the OptiTrack software and open the calibration file (requires that the system has already been calibrated - if not, refer to the OptiTrack Quick Start Guide).
3. Place the QBall 2 in the workspace so that its markers are clearly visible and so that the QBall 2 is oriented correctly with respect to the OptiTrack axes with its tail pointing in the $+Z$ axis direction (see Figure 6.14).
4. Open the Rigid Bodies view by clicking on *View* | Rigid Body Properties.
5. Using the mouse, select the markers corresponding to the QBall 2 as shown in Figure 6.17.

6. Click on the Create From Selection button shown in Figure 6.17. The previously highlighted markers now show up as `Rigid Body 1`, as seen in Figure 6.18.
7. The pivot point of the rigid body must now be moved to the center of the QBall 2. Select the QBall 2 rigid body in the left side *Project Explorer* pane. While the rigid body is selected, hold the CTRL key and select the top-most marker on the QBall 2 and set it as the pivot point by right-clicking it and select Set Rigid Body Pivot Point, as shown in Figure 6.18.
8. Select the Orientation tab in the *Rigid Body* view pane as shown in Figure 6.19. Under *Translation*, set the Y translation to -0.35 m (negative 35 centimeters from the top point is the center of the QBall 2), as shown in Figure 6.19. Click Apply Translation. The pivot point will move from the top of the cage to the center, as shown in Figure 6.20.
9. Once the trackables have been defined, they must be saved to a trackables file. This is accomplished by navigating to *File* | Save Trackables.

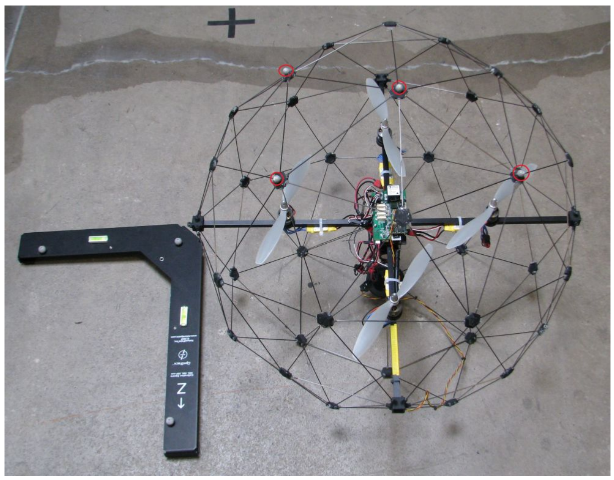


Figure 6.16: Tracking markers placed for the rigid body

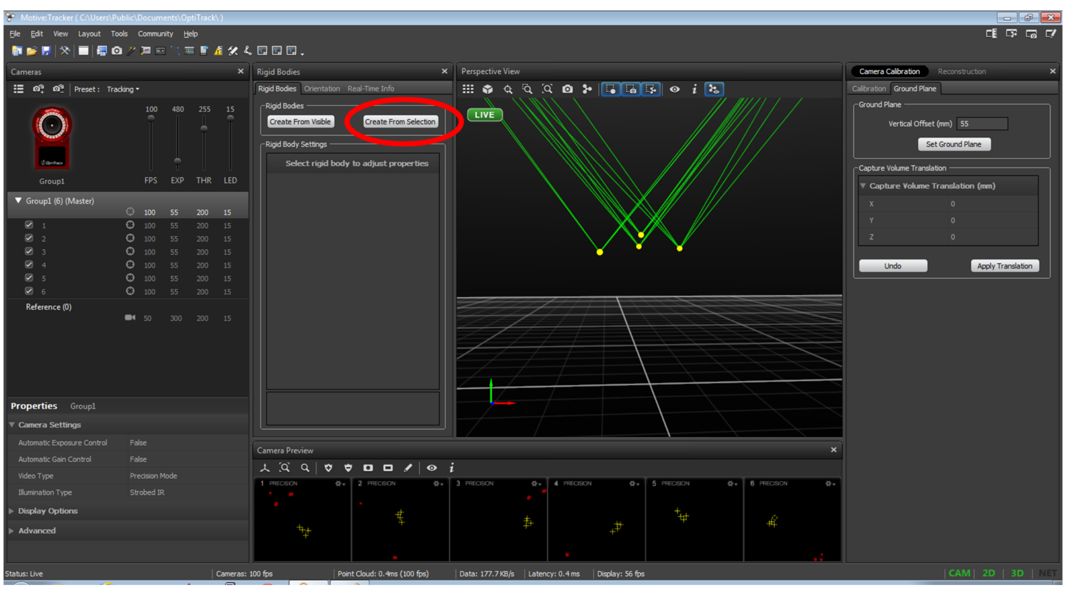


Figure 6.17: Select the markers to create a rigid body object

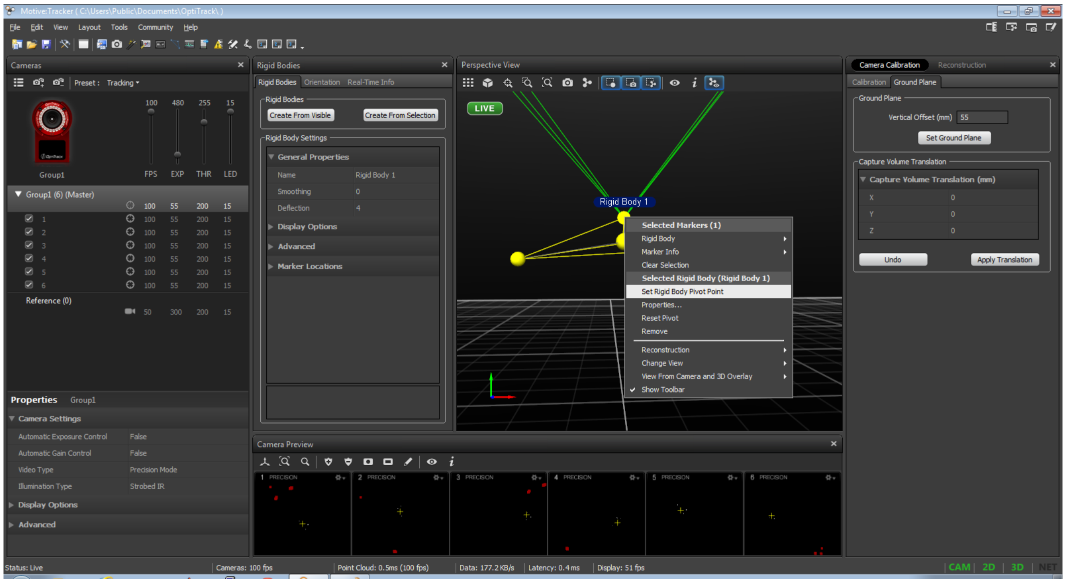


Figure 6.18: Set the new rigid body pivot point

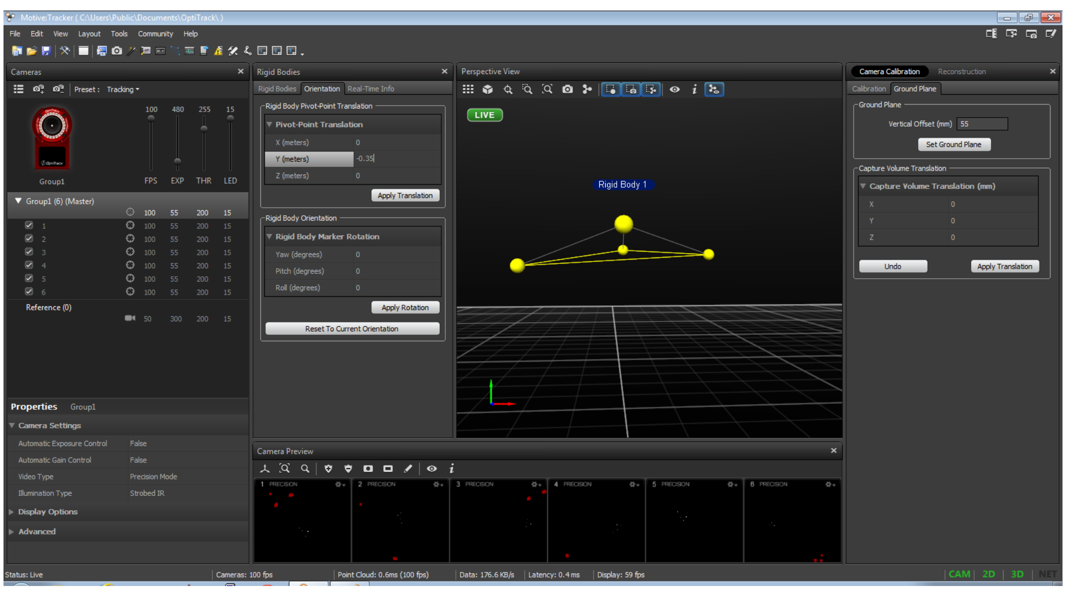


Figure 6.19: Apply a vertical translation to the QBall 2 rigid body pivot point

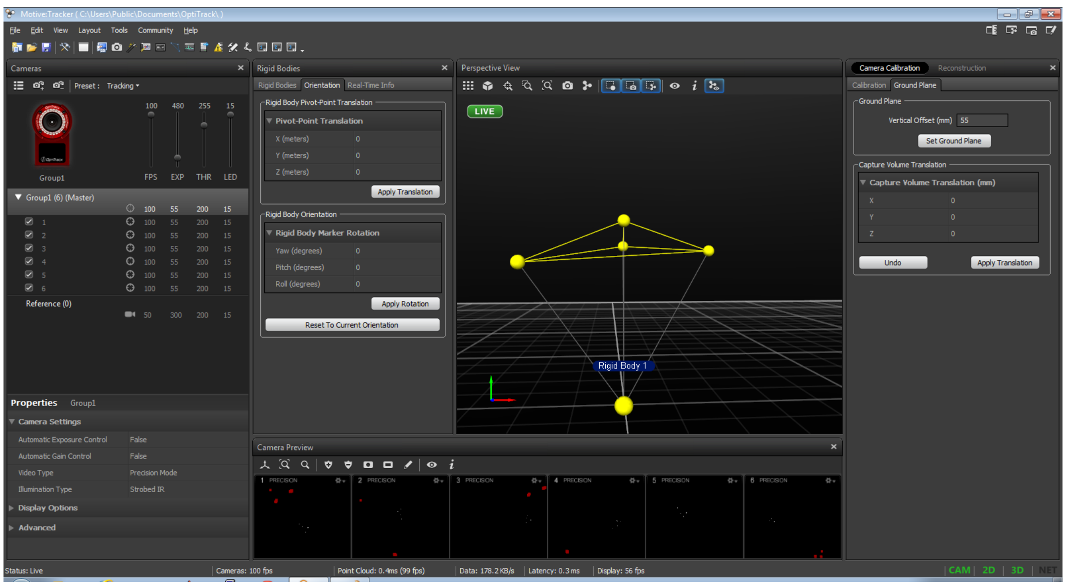


Figure 6.20: QBall 2 rigid body pivot point moved to the center

7 Charging Batteries

Caution

Before using any batteries, chargers/balancers, or power supplies, users must first read the manuals packaged with their equipment. Quanser supplies these guidelines for charging batteries but it is the users' responsibility to ensure they are operating their equipment safely and correctly. Quanser is not responsible for any damages resulting from use of batteries, power supplies, chargers, or balancers.

Before charging or using any batteries observe these safety guidelines:

- Read all instruction manuals for batteries, chargers, balancers, and power supplies.
- Use and store system in a dry environment.
- Do not charge under direct sunlight.
- Do not charge battery when battery feels hot.
- Charge battery away from flammable objects.
- Always be present when charging batteries and do not leave batteries connected to the chargers or the QBall 2 overnight.
- Charge and store LiPo batteries in a location where a battery fire or explosion (including smoke hazard) will not endanger life or property.
- Keep LiPo batteries away from children and animals.
- Consider how you would deal with a LiPo battery fire/explosion as part of your normal fire safety and evacuation planning.
- Never charge a LiPo battery that has ballooned or swelled due to overcharging, undercharging or from a crash.
- Never charge a LiPo battery that has been punctured or damaged in a crash. After a crash, inspect the battery pack for the signs of damage.
- When discarding a LiPo battery, discard it in accordance with your country's recycling laws.
- Never charge the LiPo battery in a moving vehicle.
- Never overcharge the LiPo battery.
- Never leave the LiPo battery unattended during recharging.
- Do not charge LiPo batteries near flammable materials or liquids.
- Ensure that charging leads are connected correctly. Reverse polarity charging can lead to battery damage, fire, or explosion.
- Have a suitable fire extinguisher (for electrical fires) or a large bucket of dry sand near the charging area. Do not try to extinguish electrical battery fires with water.
- Reduce risks from fire/explosion by storing and charging LiPo batteries inside a suitable container: a LiPo sack or metal/ceramic container is advised.
- Monitor recharging LiPo batteries for signs of overheating.
- Protect your LiPo battery from accidental damage during storage and transportation. Do not put battery packs in pockets or bags where they can short circuit or can come into contact with sharp or metallic objects.
- If your LiPo battery is subjected to a shock (such as a crash) you should place it in a metal container and observe for signs of swelling or heating for at least 30 minutes.
- Do not attempt to disassemble or modify or repair the LiPo battery.

7.1 Battery Charging Components

Figure 7.1 illustrates the hardware components supplied with the battery chargers.

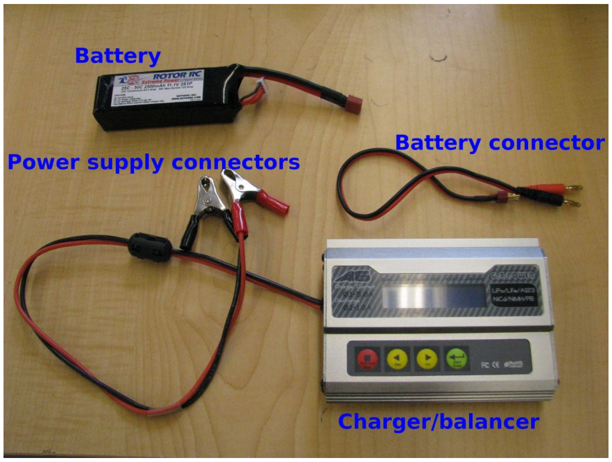


Figure 7.1: Battery charging components

The battery charger/balancer is supplied with either an individual power supply that connects to one charger or a shared power supply that can be connected to multiple chargers. The battery charger supplied with the system can typically charge various types of batteries. Ensure that the charger is set to LiPo type batteries since these are the batteries supplied with the QBall 2. The charger contains a balancer that ensures even charging across cells within the battery. Figure 7.2 shows the output terminals of the charger connected to the battery cable and the balancer ports.

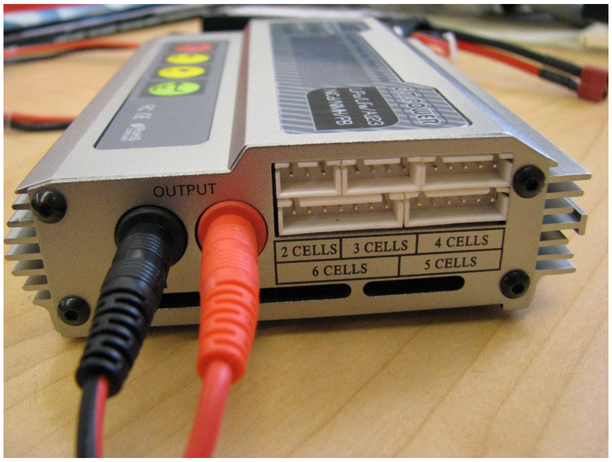


Figure 7.2: Battery charger output ports

7.2 Battery charging procedure

Setup procedure:

1. Ensure all prerequisites are met.
2. Connect the power supply to the charger.
3. Connect the battery connectors to the battery charger output terminals.

Charging procedure:

1. The correct number of cells and charge must be set. Set the charger to 3-cell LiPo battery, $11.1(3S)$, and a charge rate of 2.7 A (since the supplied batteries have a rating of 2700 mAh). Use the navigation keys to change the settings as described in the battery charger's user manual.
2. Connect the battery with the appropriate wires. Also, connect the balancer cable to the balancer. The connections are labeled according to the number of cells in the battery. Refer to your user manual for the battery details.
3. Start the charger according to the procedure described in the charger user manual. Typically this is done by holding the Start/Enter button until a beep is heard and if prompted to confirm press START again to begin charging.
4. Upon completing a full charge the charger should beep and display that the charge is complete.

8 Troubleshooting

For any issue, the first and easiest troubleshooting solution on any electronic device is to reboot the device. Turn off the QBall 2, then turn it back on again. For troubleshooting any problem with the QBall 2, it is always a good idea to open the QUARC® console in case additional information is printed to the console by going to the QUARC® menu and clicking on Console for all.... The console must be opened after the QBall 2 has booted and established a WiFi connection. If the console is opened successfully it establishes a connection to the target and the console window has the title `QUARC Console for * at tcpip://192.168.2.xxx:17000`, where *xxx* corresponds to the IP address of the QBall 2.

If you are still unable to resolve the issue after reading through this section, contact tech@quanser.com for further assistance.

Q1 The QBall 2 has crashed! What should I do?

First, make sure that the model is stopped and the power is turned off. Do not approach the QBall 2 if the model is still running or the propellers are turning. Upon stopping the QBall 2 model, a saved data MAT-file is created on the host PC in the current directory. Make a backup of this saved file for review. If support is needed from Quanser®, they will ask for this file so that the issue may be better diagnosed.

Load the saved file and review the data in Matlab®. The rows in the saved data file always begin with the time in row 1 and subsequent rows contain the data as ordered in the SAVE DATA subsystem in the QBall 2 model (see Section 6.7). The M-file script `plot_log_data.m` can be run to display several figures from the saved data log.

Verify that the joystick is properly connected and calibrated by using the host joystick controllers. First, calibrate your joystick in Windows using the USB game controller calibration procedure in Windows. Run the appropriate host joystick model for your machine. Verify that the joystick follows the correct range of outputs (throttle between 0 and 1, all other channels between -1 and 1).

□□□

Q2 You cannot ping the QBall 2

Make sure the router is on and the WiFi light on the router is on. Check that the network adapter of the host PC is connected to the router (or the wireless network Quanser_UVS or Quanser_UVS-5G) and is configured according to the network configuration procedure outlined in this manual (Section 6.3). Verify that you can successfully ping the QBall 2 by going to the Windows *Start* | Run and typing `ping 192.168.2.xxx`, where *xxx* corresponds to the IP address of your QBall 2. Turn off the QBall 2 power and verify that the Gumstix and antenna are properly connected. Make sure the Gumstix and antenna are securely connected and retry the above steps to establish a wireless connection. Recycle the power or turn on the robot and wait for approximately 60 seconds.

□□□

Q3 The model fails to build/connect or the QUARC® console does not successfully open

Remove the QBall 2 embedded controller cover so that the Gumstix board is visible. Plug in the battery to the battery connector. Turn on the power switch and look at the Gumstix for the green power LED. After approximately 30 seconds, a red LED will flash to indicate the WiFi module is powering on, and is attempting to connect to the wifi network. If the red LED flashes and remains on, then the WiFi module is functioning and is able to find the wireless network. If the red LED flashes and then turn off, the Gumstix is not able to detect the wireless network. Check that the host PC and is configured according to the network configuration procedure outlined in this manual (Section 6.3). Verify that the host PC is connected to the router (or the wireless network Quanser_UVS or Quanser_UVS-5G) and try to successfully ping the Gumstix by going to the Windows *Start* | Run and typing `ping 192.168.2.xxx`, where *xxx* corresponds to the IP address of your vehicle. If the red LED never flashes, the wireless antenna may be disconnected. Turn off the power and verify that the Gumstix wireless antenna is properly connected. Make sure the wireless antenna is secured and retry the above steps to establish a wireless connection. Recycle the power or turn on the robot and wait for approximately 60 seconds.

□□□

Q4 The QBall 2 sensors are not being read correctly or they are stuck at some constant value

Using the HIL Read block, output all possible channels. Check these outputs using scopes and displays, and determine if the problem lies with a particular sensor, or set of sensors, or if the issue is global across all sensors.

□□□

Q5 The Simulink® model appears to run slowly (i.e., the simulation time runs slower than actual time), or the console displays the message "Sampling rate is too fast for base rate".

(a) The maximum sample rate recommended for the Gumstix is 1000 Hz$(0.001$ s$)$. However, if there are complex calculations (such as image processing) performed within the model, then this could potentially limit the sample rate of the model. Try reducing the model sample rate in the menu *QUARC* | *Options* | Solver by increasing the *Fixed-step size (fundamental sample time)* parameter or change sample rates of blocks that take longer to run. If you are using image processing blocks, ensure that signal duration is set to 1 by going to Tools | External Mode Control Panel | Signal & Triggering | Duration menu on the model (The default value is 10000).

(b) The HIL Read Write block should only be used once in a diagram. These blocks perform large data transfers between the data acquisition board and the Gumstix, so placing more than one of these blocks will cause multiple reads to be performed in the same sample instant, which is unnecessary. To achieve the optimal performance, use only one HIL Read Write block for the entire model.

(c) To determine the execution time of blocks or subsystems within the model, use the Computation Time block found in the library under *QUARC Targets* | *Sources* | Time. This block outputs the computation time of a function call subsystem, measured using an independent high-resolution time source. Blocks can be placed inside a function call subsystem and connected to the Computation Time block to determine their execution time during each sample instant. This helps identify the bottlenecks in the model (blocks/subsystems with the highest execution time) and can identify blocks/subsystems whose computation time is greater than the sample time of the model. Try increasing the sample time of those blocks whose computation time is greater than the sample time of the model so that the blocks run in a slower rate thread.

□□□

Q6 Trying to start the QBall 2 model results in the error "Unable to locate the dynamic link library or shared object"

This error indicates that the QBall 2 driver is not found on the target. Make sure that the model target type is set to `quarc_linux_duovero.tlc` by navigating to the menu *QUARC* | *Options* | Code Generation pane and changing the System target file to `quarc_linux_duovero.tlc`. Open a console through the menu *QUARC* | Console for all, and verify that the console window displays the target IP of your vehicle in the window title.

□□□

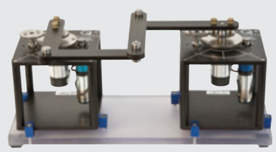
Q7 Building a model fails with the error "Not enough system resources are available to perform the operation"

When several models are compiled, the disk space on the Gumstix may become full, and you will no longer have space to build models. Using the clean option in the QUARC® menu under *QUARC* | Clean all will remove all generated code and compiled code for the current model, but this will only free up the space used by the current model. To view all models currently downloaded on the target select Manage target under the QUARC® menu. The current model's target must be powered on and ready to accept a connection. The target information is displayed including all models that have been downloaded to the target. To clear all downloaded models select all the models in the list and click Remove. Note: this will only remove generated code from the target and will not delete the source models on the host PC.

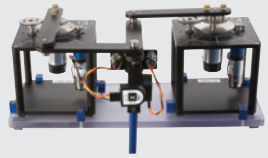
□□□

Solutions for teaching and research in robotics and autonomous systems

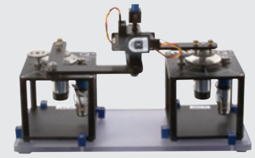
▶ 2 DOF Robot



▶ 2 DOF Gantry

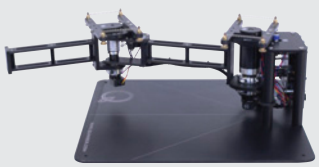


▶ 2 DOF Inverted Pendulum

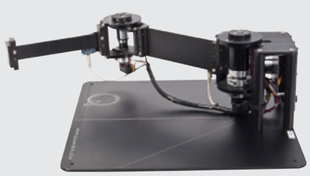


▶ 2 DOF Planar Robot

▶ 2 DOF Serial Flexible Joint

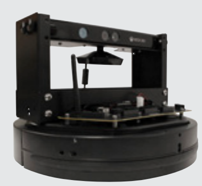


▶ 2 DOF Serial Flexible Link



▶ 6 DOF Denso Open Architecture Robot

▶ QBot 2



▶ QBall 2

▶ Unmanned Vehicle Systems Lab

With Quanser robotic systems, you can introduce control concepts related to stationary and mobile robotics, from vibration analysis, resonance and planar position control to sensors, computer, vision-guided control to unmanned systems control. All of the experiments/platforms are compatible with MATLAB®/Simulink®.

QUANSER
INNOVATE·EDUCATE

INFO@QUANSER.COM +1-905-940-3575 QUANSER.COM

Solutions for teaching and research. Made in Canada.