Barrett Technology Inc.
pyHand 1.0

Last Revised: August 13, 2013

Table of Contents

1 Installation

Figure 1: Barrett Technology pyHand Installer.

Note: If your Windows root directory is not “/C:”, please use the Manual Installer.

Note on 64 versus 32 bit machines:

pyHand is a 32-bit program. However, since Peak’s CAN install is 32/64-bit specific, our installer has to be 32/64-bit specific.

Automatic Installation:

To install pyHand, open the “install” folder. Inside the “install” folder, there will be three items: a 32-bit installer, a 64-bit installer, and a folder named “Setup_Files”. To install pyHand on a 32-bit Windows operating system, use the 32-bit installer. Likewise, to install pyHand on a 64-bit system, use the 64-bit installer.

Note that the 32-bit installer will not work on a 64-bit O.S., and the 64-bit installer will not work on a 32-bit O.S. Both installers will require the “Setup_Files” folder to work properly. To install pyHand, run the appropriate installer. To begin the installation, double click on the installer executable file and follow the on screen instructions. The installer will take you through three steps. The first step will install the Peak-System drivers. The second step will add the Peak-System DLL files to your system folders. The third step will install the pyHand program. A working copy of pyHand is now installed. A desktop shortcut has been created for you.

Manual Installation:

Manual installation requires 3 steps: Step 1, Install the Peak-System drivers. Step 2, Add the Peak-System DLL files to system folders. Step 3, Install pyHand.

Step 1
To install the Peak-System drivers, locate the "install" folder. Inside the install folder, there is another folder "Setup_Files". Inside the "Setup_ Files" folder, there is an installer file "PeakOemDrv.exe". Run it by double clicking on it. This installer will work on both 32 and 64 bit systems. Follow the on screen instructions.

Step 2
Add the Peak-System DLL files to system folders. In order for the CAN bus to work, these DLL files need to be located in the correct place. The steps will varry slightly depending on 32-bit versus 64-bit architecture. The DLL files can be found in the "Setup_Files" folder. The 32-bit DLL file can be found inside the folder "32", and the 64-bit DLL file can be found in the folder "64". If the host machine is running 32-bit Windows, the 32-bit DLL file needs to be copied into the "System32" folder, usually located "C:\Windows\System32". The 32-bit DLL files have now been moved. If the host machine is running 64-bit Windows, The 32-bit DLL needs to be copied into the "SysWOW64" folder, usually located at the path "C:\Windows\SysWOW64", and the 64-bit DLL needs to be copied into the "System32" folder, usually located at the path "C:\Windows\System32". Once you have copied the DLL files to the system folders, you are ready to install pyHand.

Step 3
To install pyHand, locate the executable file in "Setup_Files" named "pyHand-installer.exe". Run this file by double clicking it, and follow the on screen instructions. After completing the installer, you will have a working copy of pyHand installed on your computer. A shortcut has been placed on your desktop.

2 Getting Started

Make sure all physical connections to the hand are properly secured, the hand is plugged in, and turned on before double clicking the desktop shortcut to launch pyHand.

Figure 2: pyHand Loading screen.

As pyHand starts, the loading screen shown in Figure 2 is displayed. Your hand will move and make high pitched noises during the loading process. When loading is complete, pyHand will open.

3 Position Control Tab

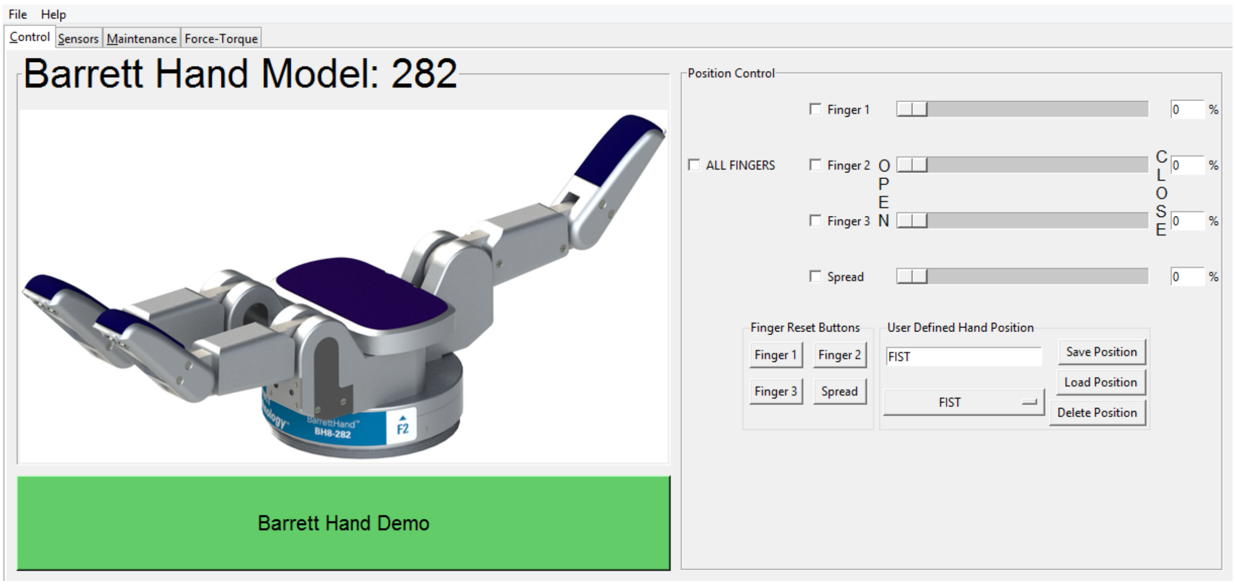


Figure 3: Position Control Tab.

This tab's primary purpose is manipulation of the Barrett Hand.

3.1 Demo

This green button starts and stops a demonstration routine highlighting the hand's dexterity.

3.2 Position Control

3.2.1 Checkboxes

These checkboxes allow the user to select joints; If multiple joints are selected, those joints will move simultaneously. The checkbox marked "ALL FINGERS" toggles the selection of all three fingers, when depressed the checkboxes will return to the configuration in use before the "ALL FINGERS" box was checked.

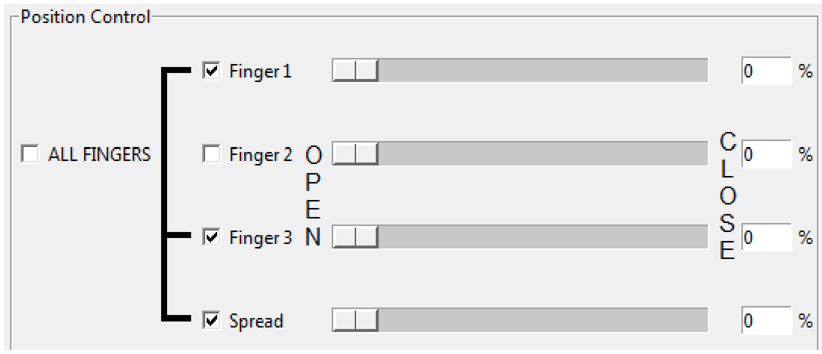


Figure 4: Position Control.

3.2.2 Linking Visual

The bars next to the checkboxes provide a visual representation of selected and linked fingers.

3.2.3 Sliders

The sliders allow the user to manipulate the hand's joints.

Note: The kinematic model of the Barrett Hand is not taken into account in pyHand. As a result, joint collisions are possible. Please use caution when manipulating joints. If a collision occurs, a joint may become unresponsive. It may be fixed by resetting the joint.

3.2.4 Entryboxes

Entryboxes allow the user to input positions. Once the user inputs a position and hits enter, both the slider and the joint will move to the input.

3.3 Finger Reset

These buttons reset the individual joints.

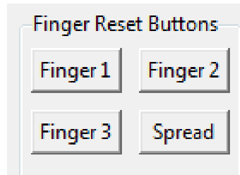


Figure 5: Reset individual joints.

3.4 Load/Save Positions

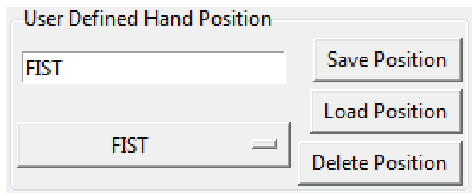


Figure 6: Load/Save Position

To save a position, type a name in the entry box under “User Defined Hand Position” and press ”Save Position”.

To load a position, select the desired position via the drop down menu. Click the ”Load Position” button. Once pressed, the joints will move to the selected position.

To delete a position, select the position via the drop down menu and click the ”Delete Position” button.

Note, some positions are built-in and cannot be deleted.

4 Sensor Tab

This tab displays sensor data from the Barrett Hand. Some of the sensors described are optional equipment.

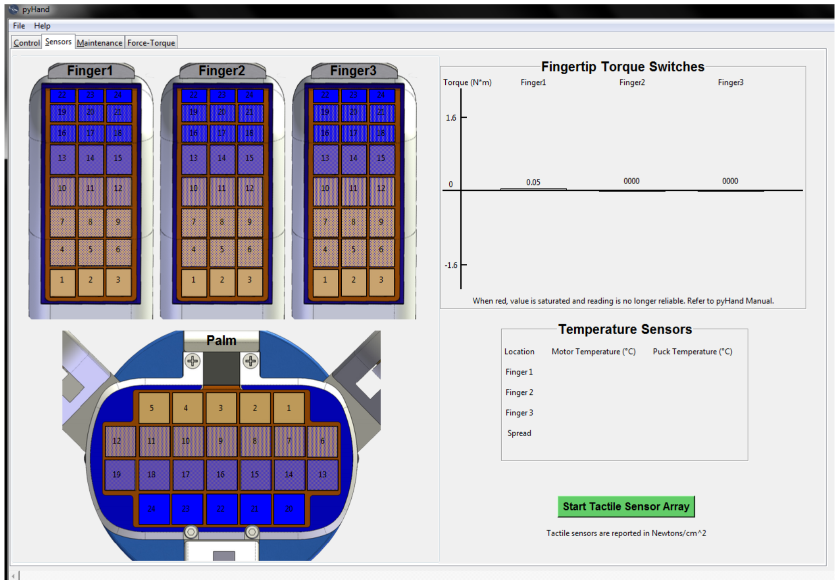


Figure 7: Sensor Tab.

4.1 Temperature Sensors

The temperature sensor data is always displayed. A warning will be displayed when a temperature is outside normal operating range (above 75 °C)

4.2 Fingertip Torque Switches

The Fingertip Torque Switch values, if installed, are shown on a bar graph.

4.2.1 Fingertip Torque Switch Data

The Fingertip Torque Switch data is converted to Newton-meters using the following polynomial fit, derived from experimental data:

$$f(x) = p_1x^3 + p_2x^2 + p_3x + p_4 \tag{1}$$

Where f(x) is torque in Newton-meters and x is motor controller value.

$p_1 = 2.754 * 10^{-10}$, $p_2 = -1.708 * 10^{-6}$, $p_3 = 37.64 * 10^{4}$, $p_4 = -2.85$.

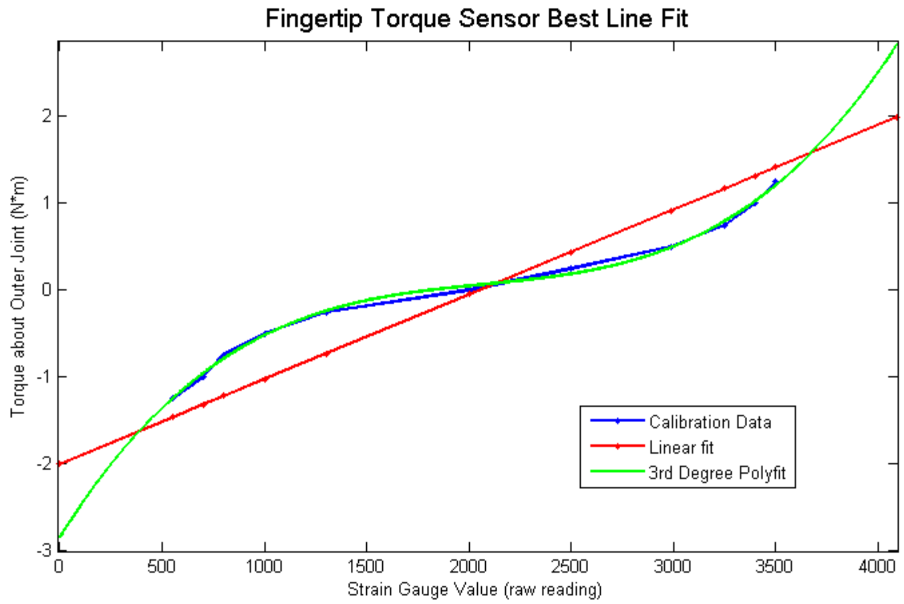


Figure 8: This figure shows the data of the Fingertip Torque Switch sample plotted against a linear and the polynomial fit. We obtain our saturation points for the Fingertip Torque Switches at the intersection points of the linear and polynomial fit.

4.3 Tactile Sensing Arrays

To display tactile sensor data, press the "Start Tactile Sensor" button on the bottom right of the Sensor Tab. When pressed, the Fingertip Torque Switch update rates are reduced. As the pressure on a cell increases, the background color of the cell darkens.

5 Maintenance Tab

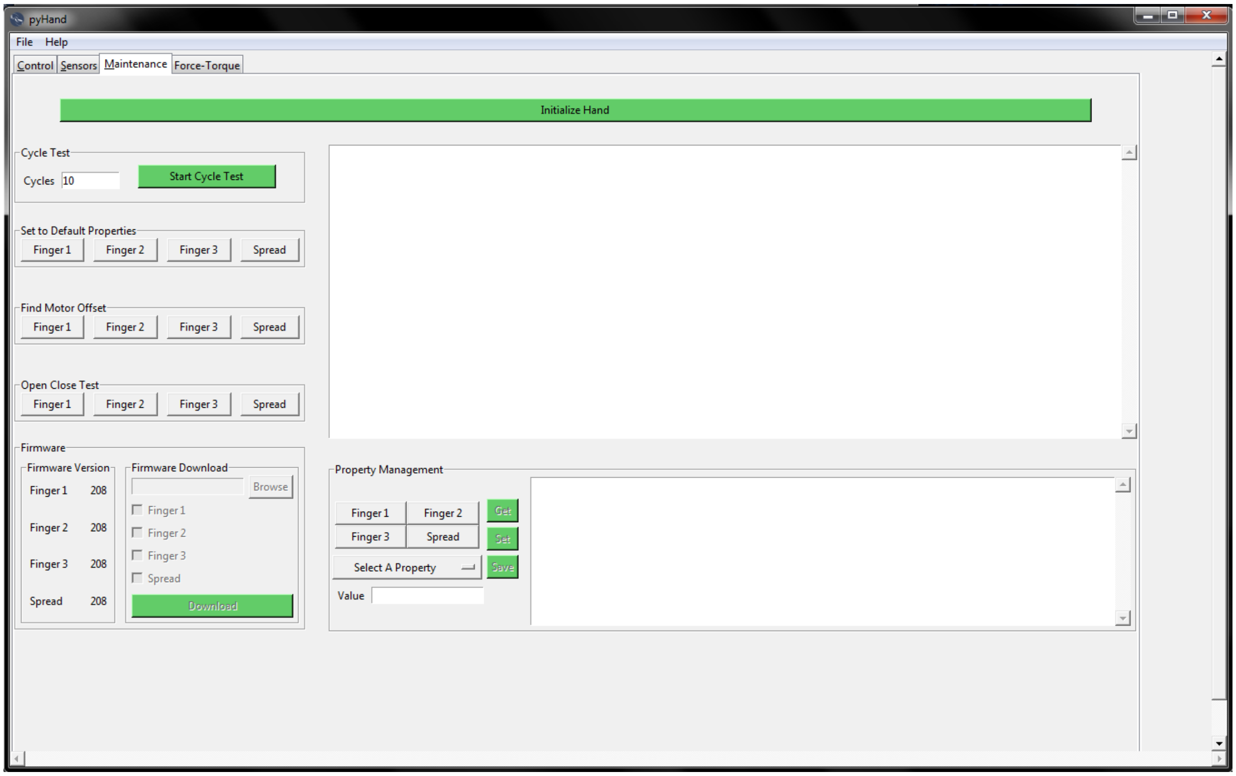


Figure 9: The Maintenance tab.

This tab allows users to perform hand maintenance.

5.1 Initialization

This button re-initializes the CAN connection, then initializes the Barrett Hand.

5.2 Cycle Test

Using trapezoidal control, the cycle test opens and closes the hand.

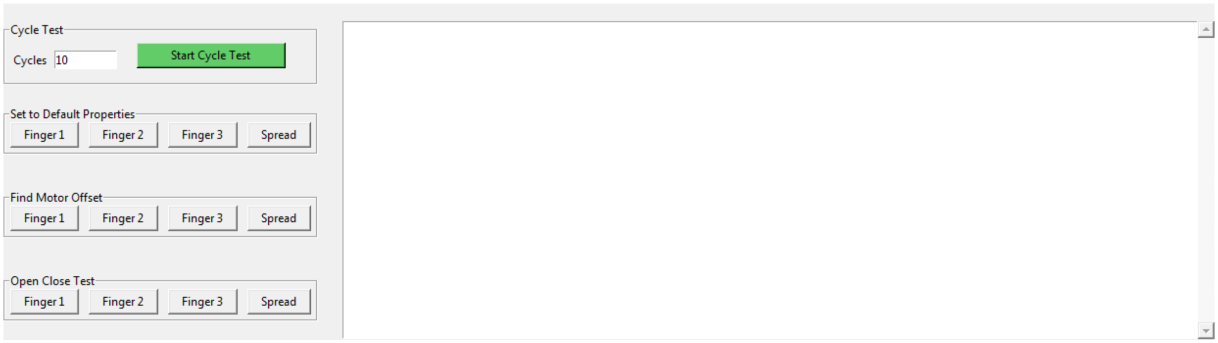


Figure 10: The cycle test, Set to Default Properties, Find Motor Offset, Open Close Test, and output text field.

5.3 Default Properties

These buttons set and save a list of default properties, returning the selected joint to factory configuration.

5.4 Motor Offsets

This test calibrates the offset for the motor current levels and position encoder readings. Warning: The motor offset test can cause permanent damage to the hand. Only use the motor offset test when instructed by Barrett personnel.

5.5 Open\Close Test

This test opens and closes the desired joint three times.

5.6 Firmware

Firmware Version The firmware version is displayed in the bottom left-hand corner of your screen.

Firmware Download This feature is currently unavailable.

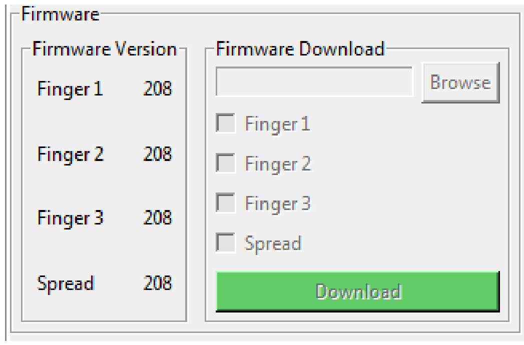


Figure 11: Firmware

5.7 Property Management

The property management can individually get, set, or save properties. Multiple joints can be selected, allowing simultaneous get, set, and save of properties.

Figure 12: Property management and output text field.

6 Force Torque Sensor Tab

This tab displays data from the Barrett Technology Force Torque Sensor. The forces and torques in each of the three Cartesian axes are shown. The Force Torque Sensor may saturate, causing invalid data. It is recommended to tare the sensor any time sturation occurs.

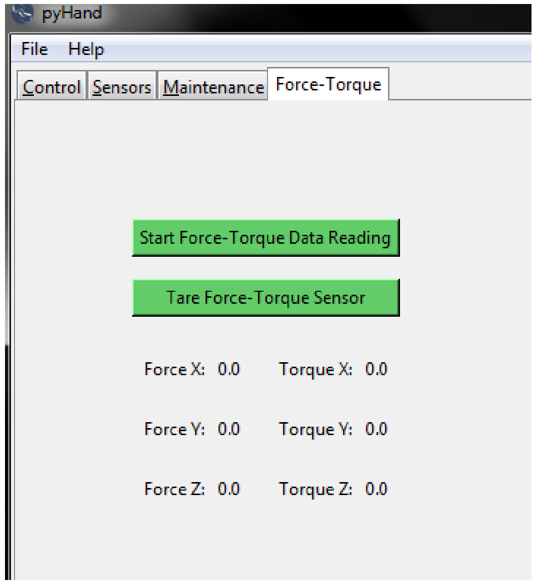


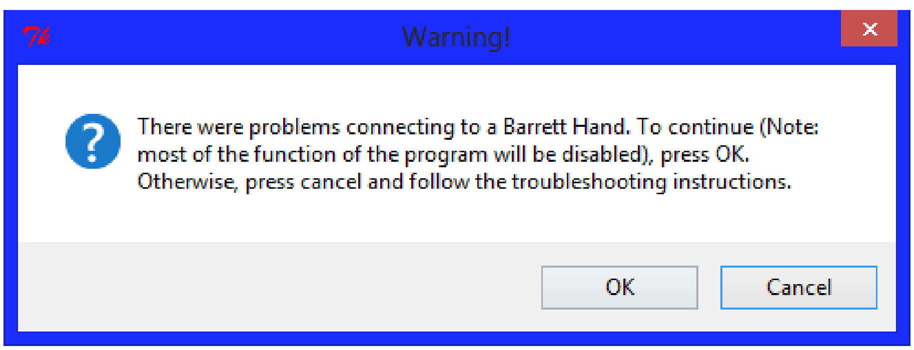
Figure 13: The Force Torque Tab.

7 Troubleshooting

Known Issues:

1. **The following issues can be resolved in the same way:**

    1. pyHand shows the following error message on startup:



    2. pyHand is partially or completely unresponsive

3. pyHand is connected to an unknown hand, even though a model 280 or 282 hand is attached
4. The Barrett Hand closes into a fist at startup

Causes:
These issues may be caused by a poor connection between the hand and the computer, or a residual CAN message.

Solution:
Close pyHand and turn off the hand. Pysically disconnect and reconnect all of the connections between the hand and the computer, ensuring that they are all secure. Turn on the hand, wait 30 seconds, reopen pyHand.

---

2. When the pyHand icon is double clicked, pyHand does not start

Cause:
The Peak-System Drivers are not correctly installed, or the DLL files are not in the proper location

Solution:
Ensure that the Peak-System Drivers are correctly installed, and that the DLL files are in the correct place. Read the Manual Installation instructions to re-install the Peak-System drivers and to confirm that the DLL files are in the correct location.

---

3. pyHand becomes unresponsive after clicking a button, or pyhand's window says "Not Responding"

Cause:
Some button clicks may send many messages to the hand, and the communication between the computer and the hand can sometimes take a noticeable amount of time. During the communication time, pyHand is unresponsive.

Solution:
Wait for pyHand to become responsive again.

8 Contact Us

Contact Barrett Technology Support

support@barrett.com
Barrett Technology Support Wiki
http://support.barrett.com/wiki/Hand