

AndroiDAQ User's Manual



By
Controlled Capture Systems, LLC
Copyright 2013-2014
All rights reserved



AndroiDAQ User's Manual

Table of Contents

Table of Contents	2
Introduction.....	3
Installing the AndroiDAQ.apk.....	4
AndroiDAQ Firmware Source and Tools	5
AndroiDAQ Specifications	6
AndroiDAQ Pin Out	7
AndroiDAQ Bluetooth Daughter board.....	8
Connecting AndroiDAQ to Bluetooth	10
Connecting AndroiDAQ to Wi-Fi	11
AndroiDAQ firmware command structure.....	12
Testing the AndroiDAQ Bare Board option.....	15
Your Notes.....	18



AndroiDAQ User's Manual

Introduction

This manual describes how to setup and use your AndroiDAQ module with the AndroiDAQ DEMO application for Android or LabVIEW, and also presents to you the basic command structure for the numerical string commands that are sent via serial using Serial Port Protocol (SPP) to the AndroiDAQ module for I/O operation.

Note: You must have a micro-SD card inserted into the micro-SD card slot on the AndroiDAQ module before you begin to use the AndroiDAQ module with the AndroiDAQ for Android or LabVIEW DEMO application. The micro-SD card must be formatted in FAT32 for it to be usable with the AndroiDAQ module. The recommended size of the micro-SD card is 256MB or larger. The AndroiDAQ firmware uses routines to write data to the SD card and if the SD card is not present, the firmware will appear to lock up. If you do not desire to have readings recorded to the SD card, you can comment out the SD card write routines in the firmware. After this, you do not need an SD card installed on the AndroiDAQ module.

Please read this manual in its entirety and ask questions if you need to. This will allow you to become completely familiarized with your AndroiDAQ module. Improper connections, mishandling, and voltages greater than 3.3-volts on the ADC and 5-volts on the I/O pins will damage your AndroiDAQ module.

The AndroiDAQ module is very small, fully programmable, and very powerful multi-core single-board micro-controller platform that is used for electronics prototyping and data acquisition. Its eight processors can operate simultaneously, either independently or cooperatively, sharing common resources through a central hub. The developer has full control over how and when each processor is employed; there is no compiler-driven or operating system-driven splitting of tasks among the multiple processors. A shared system clock keeps each processor on the same time reference, allowing for true deterministic timing and synchronization.

AndroiDAQ is very design flexible with a plethora of uses for interfacing software languages to the outside world. AndroiDAQ can be used for robotics, process monitoring and control systems, smart peripherals, and anything else your imagination can dream up. This module is compatible with Android™, LabVIEW™, JAVA™, Python, and most other programming languages. This module was designed to help innovative makers portable device driven and operated products a reality.

Each I/O pin, or channel, can be configured to be a digital input, which reads if the input signal line is high or low. The logic threshold for digital input operation is $\frac{1}{2}$ VDD or 1.65-volts DC at 3.3-volts VDD. The I/O pins can also be configured as pulse counters to measure the input signal's frequency. Further, the I/O pins can be configured as digital outputs, where a digital low output is zero volts and a digital high output is VDD or 3.3-volts DC. Lastly, any of the 16 I/O pins can be configured as a pulsed output channel that can have a set desired frequency and duty cycle for pulse width modulated signals. Here, a desired frequency and desired duty cycle that you want on each channel to have is sent to the desired I/O channels. This input frequency and duty cycle is then synthesized as an output on the desired pin, or channel. For normal period frequency output, the duty cycle default is set to 50%.

AndroiDAQ and the AndroiDAQ for Android or LabVIEW DEMO applications were developed to make it easy for you to interact with the outside world using your Android phone or tablet. This is done with simple electronic circuits that are connected to the AndroiDAQ module which can: turn



AndroiDAQ User's Manual

things on or off, measure levels of things, drive electric motors or other electromechanical devices, make sounds, and much more.

To get started, first you need to install the AndroiDAQ for Android DEMO application onto your Android device. A downloadable version of AndroiDAQ for Android DEMO is available on Google Play at: <https://play.google.com/store/apps/details?id=com.androidaq&hl=en>. As a safety precaution, all Android devices come with the option to install “non-market” apps, disabled by default. To use the source code version of AndroiDAQ for Android DEMO, or to develop your own version of this application, you'll have to follow the steps below to get your device settings configured properly.

Step 1: Click the MENU button on your Android device.

Step 2: Click on the SETTINGS menu item.

Step 3: Scroll down and click on the APPLICATIONS listing in the list.

Step 4: Ensure that the option “UNKNOWN SOURCES” is checked.

Step 5: Read the warning message presented and then acknowledge it by pressing the OK button to save this setting.

Installing the AndroiDAQ.apk

Step 1: Using the Web browser on your Android device, go to <http://androidaq-demo-enhancement-project.googlecode.com/files/AndroiDAQ.apk>. Alternatively as stated above, you can download the AndroiDAQ for Android DEMO from Google Play. If you get AndroiDAQ for Android DEMO from Google Play you can skip steps 2-5.

Step 2: Download AndroiDAQ.apk onto your Android device and save it under in the “download” folder.

Step 3: Open a file browser application. We recommend OI File manager which is available at: <https://play.google.com/store/apps/details?id=org.openintents.filemanager>.

Step 4: Browse to the “download” folder and find the AndroiDAQ.apk file.

Step 5: Click the AndroiDAQ.apk to install the AndroiDAQ for Android DEMO application on your Android device. This is usually the default action when you click on an APK in your file browser.

Step 6: The AndroiDAQ for Android DEMO application is now installed on your Android device. You can find the icon for AndroiDAQ by pressing the Launcher or Applications Icon. If you long press the AndroiDAQ icon after you find it in the list, you can put the AndroiDAQ icon on your Home screen or Desktop so that you can find it more easily.

Upon opening the AndroiDAQ for Android DEMO, the application will first ask you if you want to use Wi-Fi or Bluetooth to connect to the AndroiDAQ module. If you choose Bluetooth, the application will ask you for permission to enable the tablet's BT radio, if your Bluetooth radio is not already turned on, on your Android device. After the radio is enabled and with the AndroiDAQ module powered on, you will then be able to scan for BT devices by pressing the “Connect a device” menu item and pressing “Scan for



AndroiDAQ User's Manual

devices". This process takes care of pairing your device with the AndroiDAQ Bluetooth radio. You will be asked to input a passkey which is the radio's default 1234.

Step 7: To understand how to use the AndroiDAQ for Android DEMO application, go to: <http://www.controlcapture.com/androiddaq-demo>. With this application you can easily read the digital states or measure the input frequencies of the input channels or pins, read the voltage levels from the Analog to Digital Converter (ADC), and/or set the output pins to digital high/low states, or set up a desired channel for the AndroiDAQ to start outputting desired output frequencies and duty cycles.

The Open Source nature of our AndroiDAQ for Android or LabVIEW DEMO allows you to peruse the Android code to help you to quickly understand and learn how to send and receive data to and from the AndroiDAQ module for your own Android application. The AndroiDAQ for Android DEMO application is written in Android™, which is based on JAVA™. There are many books and Internet references to help you learn both languages quickly. The AndroiDAQ module also supports SPP, so that you can use Java, Python, or other programming languages to communicate with AndroiDAQ.

To get the development tools required to modify the AndroiDAQ for Android DEMO application, go to this web page: <http://developer.android.com/sdk/installing/bundle.html> and download the ADT bundle and then follow the instructions available on that page.

AndroiDAQ Firmware Source and Tools

The firmware for the AndroiDAQ module can be found at:

<https://code.google.com/p/androidaq-demo-enhancement-project/>

On that site you will find the zipped firmware files that contain the source code and the Parallax Propeller Tool, which is an integrated development environment or IDE for the Parallax Propeller microcontroller chip that AndroiDAQ uses. We recommend that you study and understand the operation of this source code before you make any modifications to it. You can read more about the Parallax Propeller microcontroller chip that is used on the AndroiDAQ module and the SPIN language that makes up the AndroiDAQ firmware code at:

<http://www.parallax.com/propeller/>

and

<http://www.parallax.com/Portals/0/Downloads/docs/prod/prop/Web-PropellerManual-v1.2.pdf>

After you download the new firmware from <https://code.google.com/p/androidaq-demo-enhancement-project/>, simply unzip the folder to your desktop. In the resultant folder you will find the USB driver that is necessary for your computer to communicate with the AndroiDAQ module. This file is called: CDM_2.04.16.exe. The driver was written by the company FTDI who is the supplier of the USB UART chip that AndroiDAQ uses. Simply click on the CDM_2.04.16.exe file to install the driver. After the USB driver is installed open the Propeller Tool by clicking on the Propeller.exe file. In the Propeller Tool, find the AndroiDAQRev(version number).spin file in that resultant unzipped folder using File, Open..., and click on the AndroiDAQRev(version number).spin file to open it in the Propeller tool. Press F8 in the Propeller Tool to ensure that the firmware compiles, if not please contact us. Connect the AndroiDAQ module to a USB port on your computer, press F7 in the Propeller Tool to ensure that the Propeller Tool finds the Propeller



AndroiDAQ User's Manual

microcontroller chip on the AndroiDAQ module, and lastly press F11 to reprogram the AndroiDAQ module's firmware.

AndroiDAQ Specifications

The AndroiDAQ data acquisition module is a 24-channel, breadboard compatible device. The module has 16-channels of digital input and output functions, all of which are capable of providing pulse width modulation and other complex waveforms, as well as frequency and pulse counting measurements. The remaining 8-channels provide 12-bit analog to digital conversion for sensor output and other voltage measurements. The customizable module can be controlled via USB, Bluetooth, or Wi-Fi.

- 16-channels: each channel is programmable as either an input or an output channel, for use as:
 - Digital inputs
 - Digital outputs high/low signals (sink/source 40mA max. @ 3.3v, 5v compatible)
 - Event counters or frequency measurements (up to 40MHz)
 - User settable frequency and duty cycle for frequency synthesis (1 Hz to as high as 320 KHz, note: the maximum frequencies depend on the amount and combination of frequencies. If you need other frequencies, please contact Controlled Capture Systems, as the firmware has other methods for this using the built in NCO oscillators.)
- Separate 8-channel, 12-bit resolution, Analog to Digital Converter, for voltage measurements (0 to 3.3-volts DC, at 113K Samples/Sec total)
- Eight, 32-bit cores that run independently at 80MHz, for up to 160MIPS, which can also synthesize NTSC and speech without extra support chips.
- Built-in holder for removable micro SD card for data logging which includes a time-stamp for digital, voltage, and frequency input readings (2GB or larger required for data-logging)
- Optional blue-tooth or Wi-Fi connectivity (Class 1 radio) for up to 3 Mbps data rate at distances up to 100 meters*
- Built-in Real Time Clock for time-stamp data logging
- 12-volt or 9-volt DC battery input power for stand-alone field operations or can be powered via AC power supply (sold separately)
- Breadboard prototyping spacing (board dimensions: 3.00" X 1.300", pin spacing: 1.100" X 0.100")
- Internal MCU firmware, AndroiDAQ for Android DEMO application, and AndroiDAQ for LabVIEW DEMO software are easily to learn and customizable by you (or by Controlled Capture Systems) for your projects and applications
- Free downloads of our [AndroiDAQ DEMO for Android](#), [AndroiDAQ for DEMO LabVIEW](#) applications and MCU firmware code.
- Designed and manufactured in the USA.

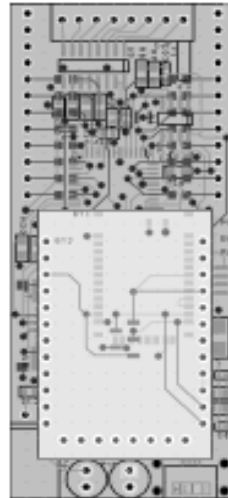


AndroiDAQ User's Manual

AndroiDAQ Pin Out

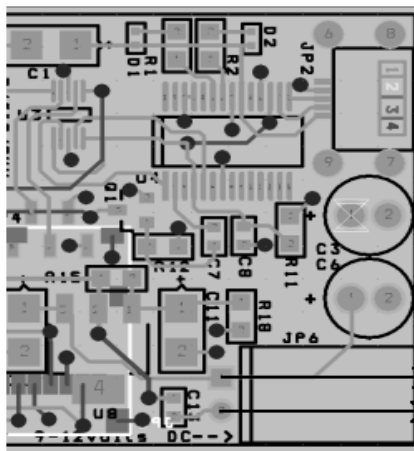
ADC Inputs 7 to 0

Ground
 Ground
 5-volts DC
 3.3-volts DC
 I/O 8
 I/O 9
 I/O 10
 I/O 11
 I/O 12
 I/O 13
 I/O 14
 I/O 15



Ground
 Ground
 5-volts DC
 3.3-volts DC
 I/O 7
 I/O 6
 I/O 5
 I/O 4
 I/O 3
 I/O 2
 I/O 1
 I/O 0

+12 to + 9-Volts DC Input and USB Port



USB Port
 Mates with
 MicroUSB B-Male

Pin 1: 12-volts DC
 Pin 2: Ground

The 3.3-volts can supply ~400mA and the 5-volts can supply ~.75 amps



AndroiDAQ User's Manual

The ADC is input only, 3.3-volts DC max.

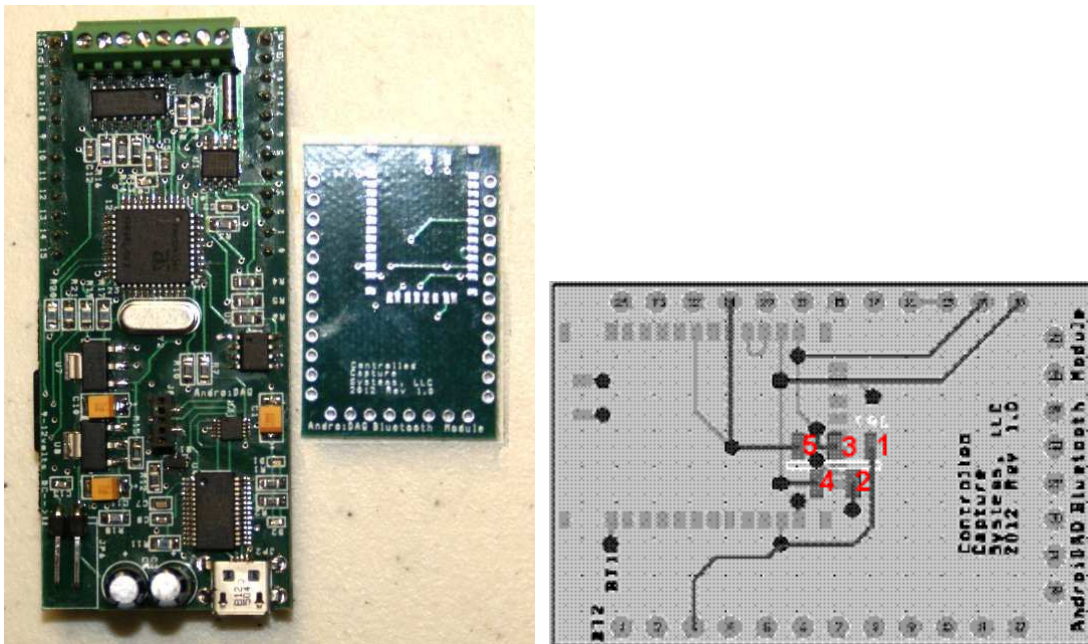
The AndroiDAQ I/O pins: Can source/sink 40mA each at 3.3 VDC. I/O pins are compatible with 5-volt DC input, as they have 1K series resistor on each input.

Note: For the AndroiDAQ power connector mate, we suggest a Molex connector, part number 50-57-9002 which is available at Mouser or Digikey. When you order this connector, you will also need to order the terminals for that connector, which you can solder to the power supply wires, vice crimping the terminals to the wire. You can also order this connector, Molex part number 14-56-2029 which includes the terminals. If you desire to crimp your terminals, vice soldering them, a 601-24-7373P crimper, available at Mouser may be used.

To power your AndroiDAQ module, a 9-volt battery with the above connector or AC wall pluggable 12-volt DC out power supply can be used, such as a Cincon TRG1512 which is available at Mouser under part number: 418-TRG1512-V. This power supply will also need to be modified with the correct AndroiDAQ connector.

AndroiDAQ Bluetooth Daughter board

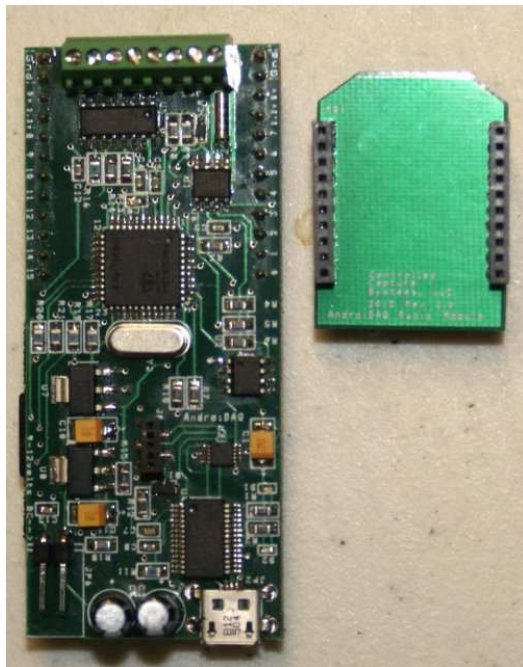
The AndroiDAQ module can be ordered with various Bluetooth and Wi-Fi radio options. If you ordered your AndroiDAQ with a Bluetooth or Wi-Fi radio installed you can skip this section. If you didn't order your AndroiDAQ with a Bluetooth or Wi-Fi radio installed, you received a bare Bluetooth radio daughter board or an xBee foot-printed Bluetooth or Wi-Fi radio board, depending on the version that you ordered.



AndroiDAQ with bare Bluetooth radio board and daughter board pin out numbers.



AndroiDAQ User's Manual



AndroiDAQ with Bare xBee foot-printed radio board and daughter board pin out numbers.

You can order and then install a Bluetooth radio or Wi-Fi radio of your choice along with the connecting 5-pin header on the bottom of the daughter board. The connecting 5-pin header can be ordered from Digikey or Mouser under Harwin part number M52040000P0545. This part is surface mounted, but can be easily soldered to the daughter board by hand using a little patience and pre-planning.

The following Bluetooth radios are pin to pin compatible with the AndroiDAQ Bluetooth Daughter board:

Class 1 Radios: Approximately 100 meter transmit/receive distance.

Roving Networks RN-41:

This radio is a Class 1 Bluetooth module which is perfect for battery powered applications and by default is ready to use in the SPP (Serial Port Profile) configuration. It uses only 250 μ A in sleep mode while still being discoverable and connectable. Multiple low power modes available allow you to dial in the lowest power profile for your application. With its high performance on-chip antenna and support for Bluetooth Enhanced Data Rate (EDR), the RN-41 delivers up to 3 Mbps data rate for distances to 100 meters. The RN-41 is the perfect product for engineers adding Bluetooth capability to their product without spending significant time and money to develop Bluetooth specific hardware and software. **Digikey part#: 740-1007-ND**

Class 2 Radios: Approximately 20 meter transmit/Receive distance.

Roving Networks RN-42:

The RN-42 is form, function, and pin compatible with the RN-41. The primary difference between the RN-42 and RN-41 is that the RN-42 is a Class 2 device, meaning its range is about 50 to 60 feet and correspondingly the power consumption is reduced. The RN-42 is perfect for short



AndroiDAQ User's Manual

range, battery powered applications and by default is ready to use in the SPP (Serial Port Profile) configuration. The RN-42 uses only 26 μ A while in sleep mode. Multiple user-configurable power modes allow the user to dial in the lowest power profile for a given application. **Digikey part#: 740-1038-2-ND**

The following Bluetooth and Wi-Fi xBee foot-printed radios are compatible with the AndroiDAQ xBee radio Daughter board:

Class 1 Radios: Approximately 100 meter transmit/receive distance.

Roving Networks RN41XV:

The RN41XV module provides drop-in, certified Bluetooth connectivity for existing systems using 802.15.4 modules. Based on the popular 2 x 10 (2mm) socket footprint often found in embedded applications, the RN41XV offers a complete wireless solution for customers looking to migrate to a standard protocol without modifying existing hardware. The RN41XV is built upon Microchip's RN41 low power Bluetooth module. The module has an embedded Bluetooth stack and supports multiple interface protocols and profiles including the commonly used SPP and HID profiles. **Digikey part#: RN41XVC-I/RM-ND or RN41XVU-I/RM-ND**

Wi-Fi radios:

Roving Networks RN171XV:

The RN171XV module is a certified Wi-Fi solution especially designed for customers who want to migrate their existing 802.15.4 architecture to a standard TCP/IP based platform without redesigning their existing hardware. The RN171XV module is based upon Microchip's robust RN171 Wi-Fi module and incorporates an 802.11 b/g radio, 32 bit processor, TCP/IP stack, real-time clock, crypto accelerator, power management unit and analog sensor interface. The RN171XV module supports infrastructure networking for worldwide internet access directly by every node and ad-hoc connectivity for fully connected point to point networks, unlike many 802.15.4 implementations that need extensive, custom application profiles and additional bridging products. **Digikey part#: 740-1044-ND or RN171XVS-I/RM-ND**

When installing the AndroiDAQ Bluetooth/Wi-Fi Daughter Module, ensure that the antenna of your Bluetooth or Wi-Fi radio is pointing toward the AndroiDAQ screw terminal which is used for voltage measurements.

Connecting AndroiDAQ to Bluetooth

When you first start the AndroiDAQ for Android DEMO application, you will be asked what kind of connection you desire, Bluetooth or Wi-Fi. To connect your AndroiDAQ to Bluetooth, press the Bluetooth button. If Bluetooth wasn't enabled on your Android device, the DEMO application will ask you for your permission to enable Bluetooth radio functions. Press Yes. After your Android device has turned on your device's Bluetooth radio, you will then need to press your menu button to open the DEMO menu. Select the menu item 'Connect a device'. This opens a dialog box that contains a list of all paired Bluetooth devices on your Android device; your list may be empty due to not pairing any devices to your Android device. Here, you need to press the 'Scan for devices' button under this dialog, which helps the Android system to find the AndroiDAQ Bluetooth module. You will then be prompted to input the Bluetooth module's passkey which is 1234. This will aid in the pairing of the Bluetooth module to your Android device.



AndroiDAQ User's Manual

After the 'Scan for devices' has finished, you then select your AndroiDAQ Bluetooth radio on the list of Bluetooth devices found. The numbers shown on the list, per Bluetooth radio found, are the last four digits of the AndroiDAQ Bluetooth modules serial number. DEMO will then connect to your Bluetooth radio module, if it is powered on via the AndroiDAQ module which also needs to have power applied to it. If your Bluetooth radio is not found, due to the module not being powered on or something else, you will need to rescan your Bluetooth devices, as stated above, to lock your Android device with the radio's availability.

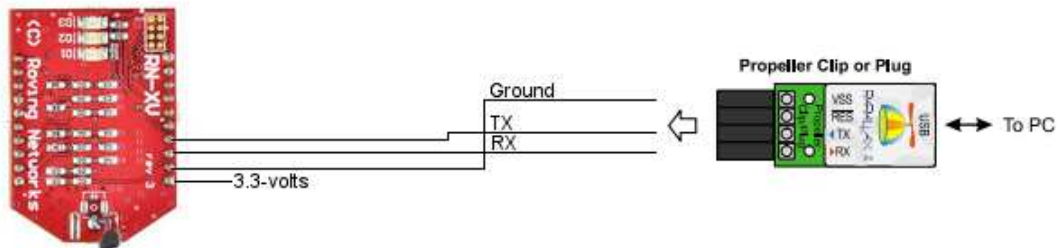
We have found that if you dump the AndroiDAQ for Android DEMO application, by pressing the back button, without first disconnecting the Bluetooth connection under the Menu beforehand, then you will need to 'Scan for devices' again to lock the radio to the Android device. We have also observed that there are times where you will also need to disable and re-enable the Bluetooth radio on your Android device to re-lock on to any Bluetooth radio. We are not sure why the Android OS operates this way.

Upon subsequent restarts of the DEMO application, when you restart the DEMO application under Bluetooth, you will need to connect to your radio via the Menu list item 'Connect a device', it is not automatic.

Connecting AndroiDAQ to Wi-Fi

When you order the AndroiDAQ module with the Wi-Fi radio option, we do the preliminary setup here at the factory to test your radio to ensure proper operation with your AndroiDAQ module, though you will need to reconfigure the Wi-Fi module to log onto your Wi-Fi network using your Wi-Fi network's SSID and its password when you receive your Wi-Fi module.

To do this, please first download the Roving Networks WiFly User's Manual at: <http://ww1.microchip.com/downloads/en/DeviceDoc/50002230A.pdf> and read it to understand your RN-XV module and to learn how to place the Wi-Fi module into Command Mode and to configure the module in either Ad-Hoc mode (by providing 3.3-volts DC to pin 8 before powering up the Wi-Fi module) or via Serial communications using a USB adapter like this one: <http://www.parallax.com/product/32201> , where you can use the Propeller Plug to configure your Wi-Fi module using USB and a serial terminal program such as the Parallax Serial Terminal (available here: <https://code.google.com/p/androidaq-demo-enhancement-project/downloads/detail?name=PST.exe.zip&can=2&q=#makechanges>) and using a circuit like the example below:



or you can order a module like this one to connect via USB to configure your Wi-Fi module: <https://www.sparkfun.com/products/8687> .



AndroiDAQ User's Manual

When the Wi-Fi module is in the Command Mode in either Ad hoc or via USB serial link, please enter the following commands to configure your Wi-Fi module to your Wi-Fi networks SSID and password:

```
set wlan ssid <your Wi-Fi network's name>
set wlan phrase <your Wi-Fi network's log-in password>
set opt jointmr 20000
set uart baud 115200
set wlan join 1
set comm remote 0
set comm match 10
set comm open 0
set comm. Close 0
save
reboot
```

After these steps are complete, your RN-XV radio will reboot and try to connect to your Wi-Fi network (note: if you placed the module into Ad hoc mode, you must remove the wire from pin 8 so that the module reboots normally and not in Ad hoc mode). After you have verified that it is properly connecting to your Wi-Fi network (see WiFly User's Manual) you can now remove the RN-XV from your USB adapter (if applicable) and then install your RN-XV radio onto the AndroiDAQ radio daughter board and connect this assembly to your AndroiDAQ module.

Below are the factory commands that we use to configure RN-XV modules to run with AndroiDAQ. If you run into any problems with the configuration of your RN-XV radio you may need to re-enter these commands to ensure compatibility with the AndroiDAQ module. This can be done via serial link or in Ad hoc mode (see WiFly User's Manual).

```
factory RESET
set wlan ssid "your network's name"
set wlan phrase "your network's password"
set opt jointmr 20000
set uart baud 115200
set comm match 10
set wlan join 1
set comm. Remote 0
save
reboot
```

When you first start the AndroiDAQ for Android DEMO application, you will be asked what kind of connection you desire, Bluetooth or Wi-Fi. To connect your AndroiDAQ to Wi-Fi, press the Wi-Fi button. You will then be asked for the TCP address of the Wi-Fi radio connected to your network (see WiFly User's Manual).

AndroiDAQ firmware command structure

The AndroiDAQ for Android DEMO application is designed to send numerical commands to the AndroiDAQ module via a Bluetooth or a Wi-Fi connection. These commands are in a form of a numerical string and sent via Serial Port Protocol. The channel configuration settings are set according to what type of input or output that you desire for each I/O channel. This command structure allows the AndroiDAQ firmware to be menu driven. For example, if you want to read the Real Time Clock's date and time, one would send a numerical menu command to the AndroiDAQ module of 05. We will explain this further in this section.



AndroiDAQ User's Manual

AndroiDAQ I/O channels (channels 0 through 15) can be set as: a digital input, an input counter, a digital output that is low, a digital output that is high, or lastly an output that is oscillating at a set frequency. The table below will help you envision the numerical commands in accordance to the settings described.

Each I/O channel or pin can be one of these:	Input Digital	Input Counting	Output Digital Low	Output Digital High	Output Pulsed	Output Desired Frequency	Output Desired Duty Cycle
Number sent with 0 prefix:	0	1	2	3	4	Up to 320 KHz	1-99%

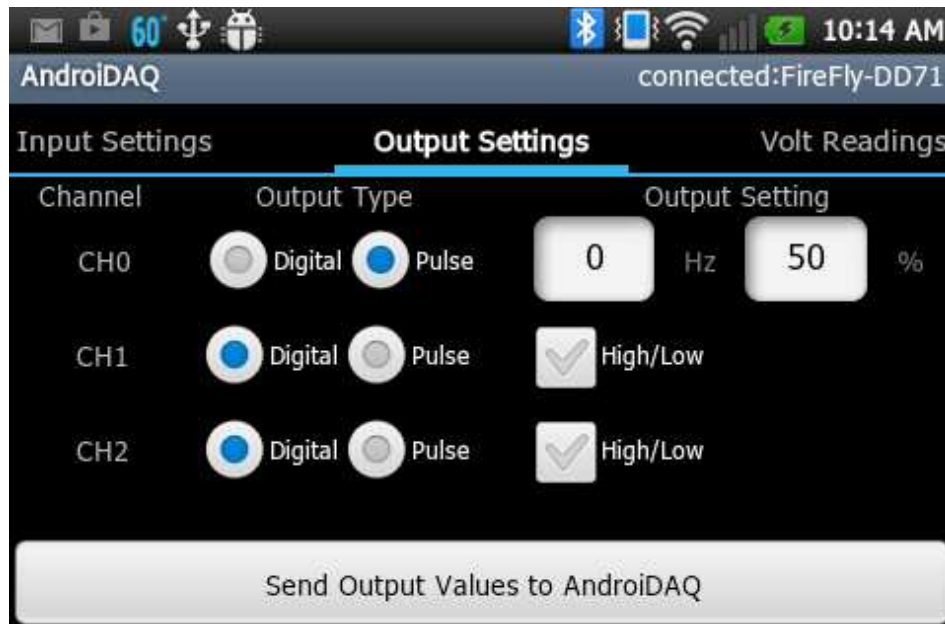
So let's take a look at an example numerical sequence command that could be sent to the AndroiDAQ module, so that you can understand the above table. Let's say that we want to set I/O channels 0-2 to outputs with channel-0 being a pulsed output, channel-1 being a digital low output, and lastly channel-2 being a digital high output, leaving all other channels as inputs, which is the default in the AndroiDAQ application. Here you would open the AndroiDAQ for Android DEMO application and on the Setup I/O screen you would configure channels 0-2 as outputs. This is simply done by clicking each channel to configure them as outputs as shown below:



Then by using your finger to swipe across your Android device's screen right to left, you enter the next AndroiDAQ for Android DEMO configuration screen, which is titled "Input Settings". Since we are only using output channels for this example, we can skip this screen by swiping the screen right to left again, which will land us on the "Output Settings" screen, shown below. Here, we will configure channel 0 to be a Pulse "Output Type" by clicking the "Pulse" radio button. When the "Pulse" radio button is selected, two input boxes with the end label of "Hz" and "%" are displayed. We will use these input boxes to enter the values of the frequency and duty cycle that you wish to be output on the pin or channel. By default Channels 1 and 2 are set to digital "Output Types". For this example will leave channel-1 to output a low or zero volt value, and set channel 2 to output a high or 3.3-volt level, as shown below.



AndroiDAQ User's Manual



Now referencing our table, shown again below, we will build the numerical command that would be sent to the AndroiDAQ module; again channels 0-2 are outputs, with channel-0 being a frequency output, channel-1 being a digital low output, and channel-2 being a digital high output. AndroiDAQ requires that all I/O channels be configured per numerical command sent, meaning channels 0-15 need to be set in one full command, so our numerical command to set the pin configuration settings would look like this (channels 0 through 15):

Each I/O channel or pin can be one of these:	Input Digital	Input Counting	Output Digital Low	Output Digital High	Output Pulsed	Output Desired Frequency	Output Desired Duty Cycle
Number sent with 0 prefix:	0	1	2	3	4	Up to 320 KHz	1-99% (50%-default)

4-0_50, 2, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Referencing the numerical sequence and table above, you will see that channel-0's setting is 4-0, meaning it is setup to be an "Output Pulsed" with a frequency output of 0-Hz with a duty cycle of 50% and channel-1's setting is 2, meaning it is setup to be an "Output Digital Low", and channel-2's setting is 3, meaning it is setup to be an "Output Digital High" with the remaining channels (3 through 15) set as "Input Digital" or ignored if there are no connections to the I/O pins of the AndroiDAQ. Now if we wanted to have channel 0 send an output frequency of 2100-Hz at a duty cycle of 50% our numerical command would be:

4-2100_50, 2, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Of course this is a very basic explanation of the numerical command sequence structure to configure the channels of the AndroiDAQ module, though it is very important that you understand the numerical command structure to be able to modify the AndroiDAQ firmware and DEMO applications, or to create your own application using the AndroiDAQ module. Studying the



AndroiDAQ User's Manual

firmware and AndroiDAQ for Android or LabVIEW DEMO application source code will help you in understanding the structure.

There is more going on under the hood between the AndroiDAQ DEMO applications and the AndroiDAQ firmware. The AndroiDAQ firmware is structured to operate under a numeric menu driven command structure, meaning that simple numerical commands are sent via Bluetooth or USB to command the firmware to run a particular sub-routine. Below is a list of the numerical commands that are currently used for the AndroiDAQ firmware.

Note: all numerical commands are prefixed with a zero when the command is sent. This leading zero is used to trigger the AndroiDAQ module firmware's *MenuLoop* sub-routine.

Number sent:	Firmware sub-routine ran, <i>in italics</i> :
0	Prepare and receive the 16 channel configuration settings. <i>SetPins</i>
1	Set the 16 channels to the received configuration settings. <i>SetConfig</i>
2	Read input channels and send reading to Serial. <i>GetInputs</i>
3	Not used, this is a test loop for your subroutine development using <i>testloop</i>
4	Read ADC voltages and send these readings to Serial. <i>GetVolts</i>
5	Read AndroiDAQ Real Time Clock (RTC) and send values to Serial. <i>GetTime</i>
6	Auto-set RTC with system values. Note: you need to send date/time information to AndroiDAQ firmware in yy_MM_dd_E_kk_mm_ss format. <i>SetTime</i>
7	Read AndroiDAQ log file (log.txt) send strings to Serial. <i>ReadLog</i>
8	Delete AndroiDAQ log file. <i>DeleteLog</i>
9	Stop continuous input reading
10	Start continuous input reading
11	Start continuous voltage reading
12	Stop continuous voltage reading
13	Center Pan and Tilt example
14	Move Servo Motor
15	Move Stepper Motor

Testing the AndroiDAQ Bare Board option

If you ordered the AndroiDAQ Bare Board option you will receive by email the build sheet with part numbers and notes on assembling the module and radio daughter board. After all parts are installed on your AndroiDAQ module and radio daughter board, please follow the below sequence to ensure that your firmware is properly loaded onto your AndroiDAQ module.

1. We use LabVIEW and our AndroiDAQ LabVIEW library for these test steps. If you do not have LabVIEW you can download our stand-alone AndroiDAQ DEMO for LabVIEW application at <https://code.google.com/p/androidaq-demo-enhancement-project/>. After you download the zipped file, unzip it and click on the setup.exe file inside the resulting folder. This will install the AndroiDAQ DEMO for LabVIEW application and the LabVIEW Runtime engine required to run this application.
2. If you haven't already, install Parallax's Propeller/Spin Tool Software onto your laptop or desktop Windows computer. This software is available at: <http://www.parallax.com/downloads/propeller-tool-software> or in the AndroiDAQ firmware download file. After you download the zipped file from the link above, unzip it and click on the setup.exe file inside the resulting folder to install the Propeller/Spin Tool Software.



AndroiDAQ User's Manual

3. If you haven't already, install a micro-SD card into the AndroiDAQ's SD card slot. The recommended size for the SD memory card is 256MB or larger and it must be formatted in FAT32 to operate properly with the onboard AndroiDAQ file system.
4. Apply power to the AndroiDAQ module on JP6. As stated earlier in this manual, the pin nearest the JP6 silkscreen designator is the positive pin, or pin 1, and pin 2 is ground. We recommend using a 9-volt battery or 12-volt DC wall wart style power source. After connecting the power, check that the 5v and 3.3v pins measure 5-volts DC and 3.3-volts DC respectively in reference to the Gnd pins. If not, unplug the AndroiDAQ module and inspect your component orientations to ensure that none of the IC chips are installed in the wrong direction. This is a common mistake which will require you to remove the chip and reinstall a new one in the correct orientation. The component may not be damaged, please check the manufacturer's component data sheet to see if it is reverse voltage protected. When the voltages measure correctly, leave the AndroiDAQ module plugged into your power source.
5. Connect your laptop or desktop computer to the AndroiDAQ module via a USB cable. Your USB cable should have a microUSB-b connector on one end to attach to the AndroiDAQ module. Most Android phones use this style of connector for their USB port so you may have one lying around, or you can order one from Mouser under their part number: 552-IPUSB1MS-R.
6. If your Windows system does not find a driver for the AndroiDAQ module after it is plugged in, unplug it from the USB port and then install the CDM_2.04.16.exe file that is located in the data folder that is located in the directory where you installed the AndroiDAQ DEMO for LabVIEW application. The CDM_2.04.16.exe installs the FTD drivers for the USB portion of the AndroiDAQ module. If your Windows system still doesn't recognize your AndroiDAQ module, remove the USB cable and power source and check the connections on the USB connector JP2, the connections on the FT232R chip (U4), and the connections for the RX/TX switch (U3), and all other connections on your AndroiDAQ module.
7. After your AndroiDAQ module is recognized by the Windows system by plugging in the USB cable with the AndroiDAQ module powered on, start the Propeller/Spin Tool Software program and then ensure that the Propeller/Spin Tool Software recognizes your AndroiDAQ's Propeller chip by pressing the F7 key in the Propeller/Spin Tool Software. The software will tell you if it found AndroiDAQ's Propeller chip and on which COM port it is on. If it is not found, check your connections on U2 and U6 and try again until the Propeller chip is found by the Propeller/Spin Tool Software.
8. If you haven't already, download the latest AndroiDAQ firmware at: <https://code.google.com/p/androidaq-demo-enhancement-project/> and save it to your computer. Unzip this zipped folder and then in the Propeller/Spin Tool Software load the firmware spin code by clicking on File then Open..., and then select the AndroiDAQRev(X).spin file in the folder that you just unzipped. When done properly, the Propeller/Spin Tool Software will load and then display the AndroiDAQ main spin code in the right-hand side of the user's interface.
9. You program your AndroiDAQ module by pressing F11 in the Propeller/Spin Tool Software, after the AndroiDAQRev(X).spin code is loaded into the Propeller/Spin Tool Software. You will be alerted if the programming of the AndroiDAQ module was successful or if it failed.



AndroiDAQ User's Manual

10. Find the AndroiDAQ DEMO for LabVIEW icon under Windows' Start, All Programs, AndroiDAQ and then click on the AndroiDAQ.exe icon. With the AndroiDAQ module connected to your computer via USB as described above, the AndroiDAQ DEMO for LabVIEW application will automatically find your AndroiDAQ module.
11. After the above tests have passed and your AndroiDAQ firmware is installed successfully, you can now try to read AndroiDAQ's Real Time Clock. This is done by pressing the "Read RTC Clock" button on the LabVIEW DEMO screen. If successful you should see a date and time similar to: 01/01/200 00:00:80.
12. To set the AndroiDAQ Real Time Clock to real time, press the "Set RTC Clock" button in LabVIEW DEMO screen. This will sent to the AndroiDAQ module your laptop or desktop's current date and time setting.
13. Apply one of the 3.3-volts DC outputs, found on either side of the AndroiDAQ I/O pins toward the green terminal block ADC connector, to I/O port 0 of the ADC input and ensure that that channel can read the 3.3-volt signal by pressing the "Read Voltages from AndroiDAQ" button. Repeat this step for each ADC input pin and ensure that the voltage is read.
14. Set up an LED test circuit as described below:
Connect any of the four "Gnd" pins on the AndroiDAQ module to one end of a 470 ohm resistor. The other end of the resistor is connected to an LED (cathode) and then the LED (anode) is connected to a length of wire.
15. Attach the loose end of the wire from step 14 above to I/O Port 0 of the AndroiDAQ I/O pins, not the green ADC terminal. This will connect the LED anode to the pin in test. Set I/O channel 0 to an output by clicking on the "Input or Output" button for that channel. This will turn the button on or change it to bright green. Ensure that the Output Type for this channel is set to "digital" and then toggle the "Output Setting" to high (see Legend in DEMO application) and then click on the "Set Outputs of AndroiDAQ" button to ensure that the LED lights up. Toggle the "Output Setting" to low to ensure that the LED turns off. Do this for every I/O pin.
16. If any of the above tests fail, check your solder connections and part orientations on your AndroiDAQ module using a magnifying glass. Please contact us if you need assistance.



AndroiDAQ User's Manual

Your Notes