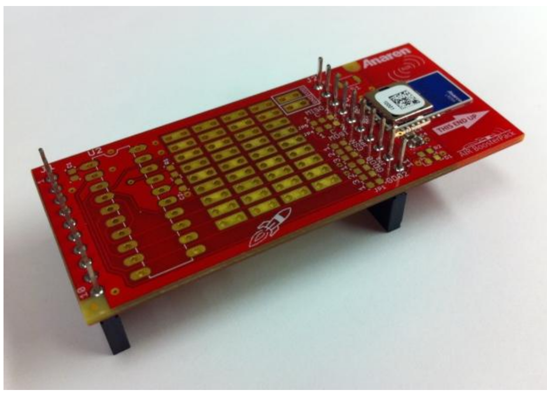
Anaren®

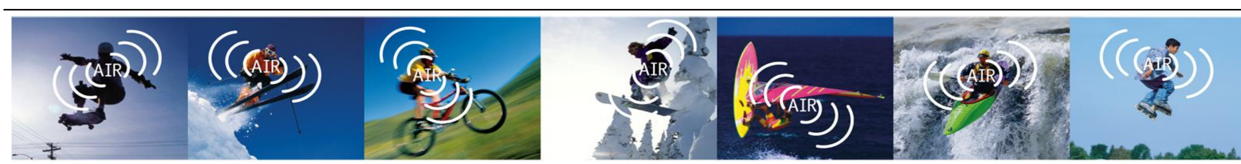
What'll we think of next?®



Anaren Integrated Radio

AIR BoosterPack Users Manual

Release Date 1/18/12



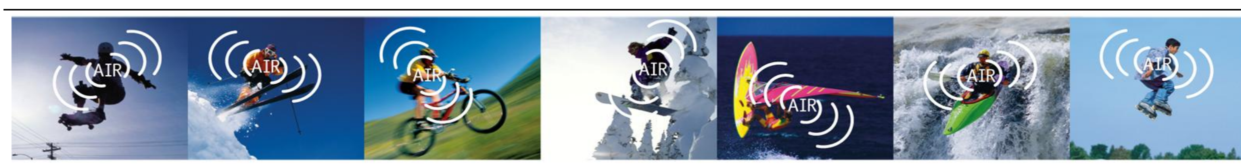
Anaren Integrated Radio

THIS PAGE LEFT INTENTIONALLY BLANK

USERS MANUAL

AIR BoosterPack

Contents



THIS PAGE LEFT INTENTIONALLY BLANK

1. AIR BoosterPack Overview

1.1. Overview

The AIR BoosterPack is a low-power wireless transceiver extension module for use with the Texas Instruments MSP-EXP430G2 LaunchPad development kit. Based on the CC110L device, the on-board A110LR09A radio module with integrated antenna operates in the European 868-870MHz and US 902-928MHz ISM bands. The included AIR BoosterPack software application, called AIR BoosterStack, demonstrates an example sensor network as well as network status reporting.

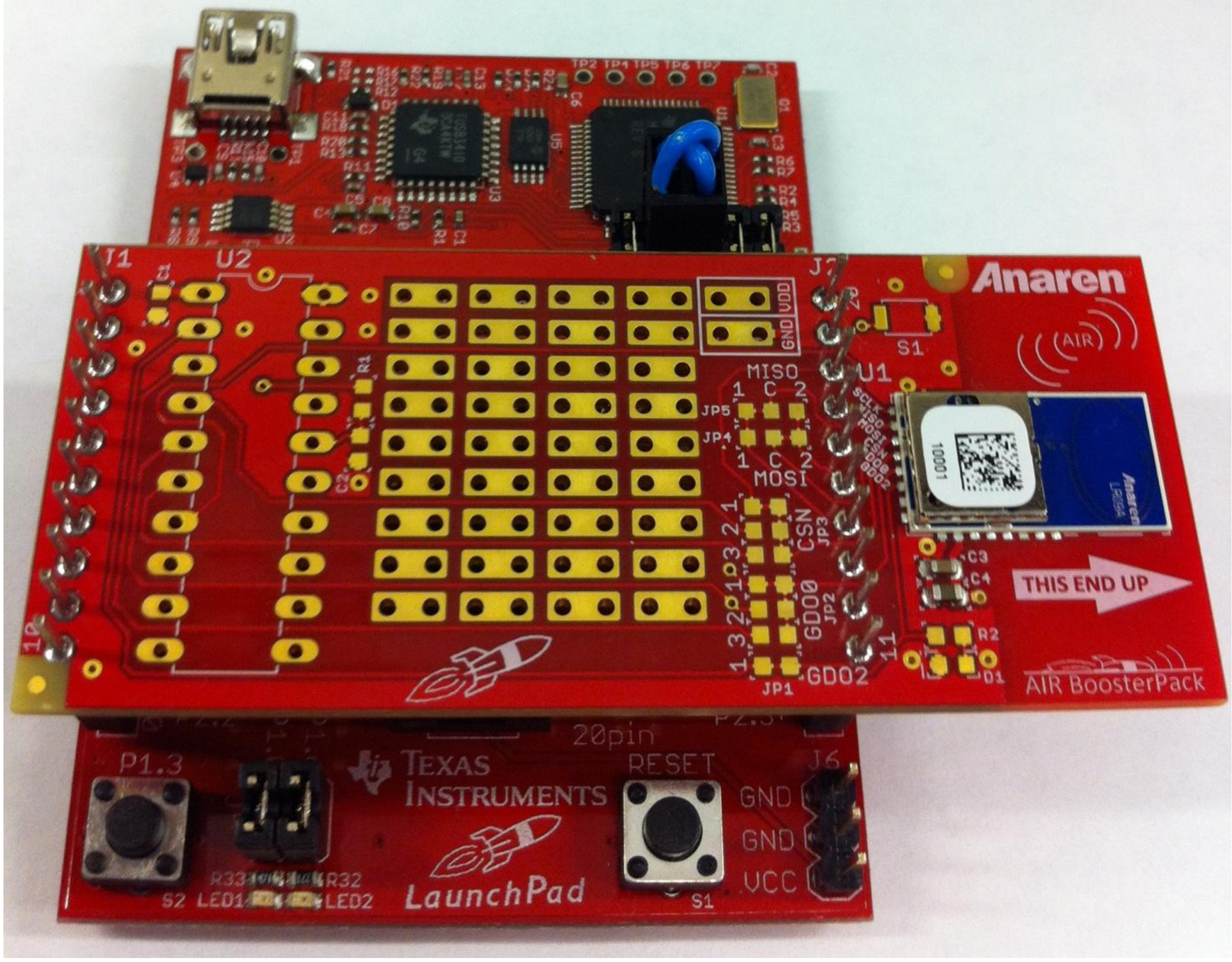


Figure 1 - AIR BoosterPack with LaunchPad

AIR BoosterPack Hardware Features:

- 1.8 to 3.6 V operation
- Low Power Consumption
- SPI Interface
- Prototyping area
- Footprints for adding a microcontroller, pushbutton switch, and LED for standalone operation
- ROHS Compliant
- See A110LR09x Users Manual for radio specific features

AIR BoosterStack Software Features:

- Low-power temperature sensor application
- Star network topology with one hub node and up to four sensor nodes
- Graphical User Interface provides network control and displays key radio parameters
- Ability to change radio settings locally and/or remotely
- Remote control feature to turn on/off LaunchPad green LED
- Node ID, operating state, and radio settings restored at power-up

1.2. Kit Contents

The AIR BoosterPack kit includes the following:

- Two AIR BoosterPack modules
- Two MSP430G2553 devices (preloaded with a sample program)
- Quick Start Guide
- CD containing AIR BoosterStack demo application software, USB UART device driver, and supporting documentation

2. Getting Started with the AIR BoosterPack

The following sections describe the necessary steps to get the AIR BoosterPack hardware and BoosterStack software up and running with the LaunchPad.

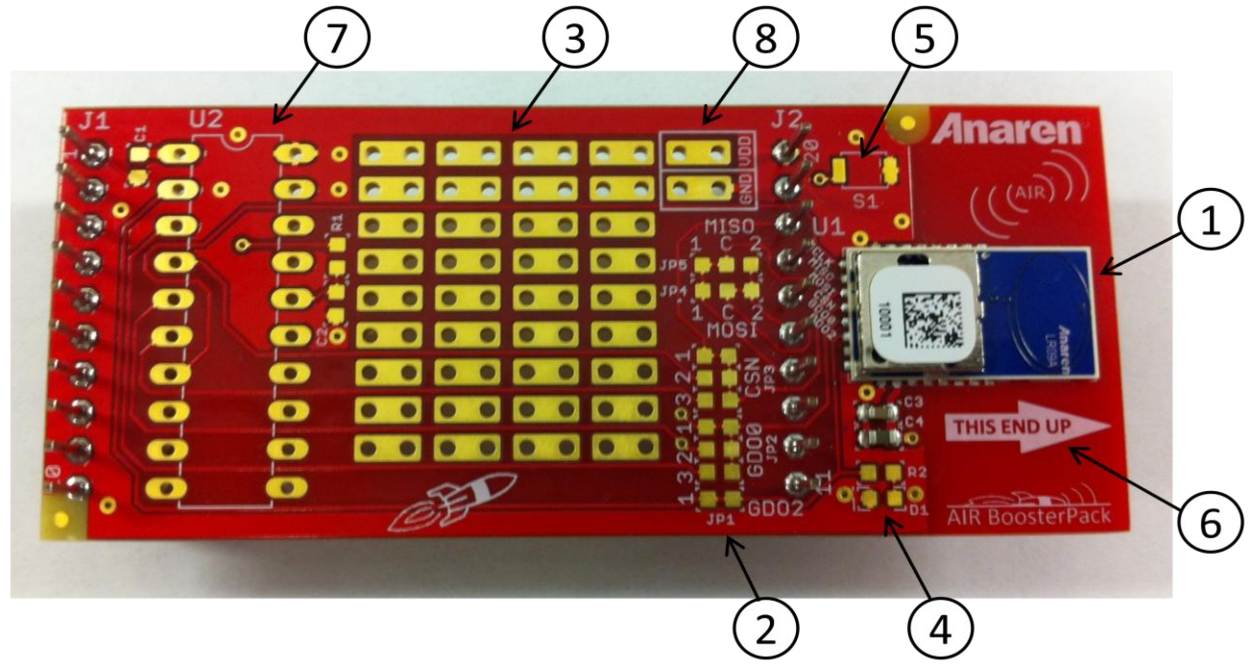


Figure 2 - AIR BoosterPack Overview

① **AIR Radio Module**.

② **Signal Path Jumpers**. See section 4.3 for details on changing jumper settings.

③ **Prototyping Area**. Add your own circuit to this area. These pads have no electrical connection to the AIR radio module, LaunchPad interface connectors, or power/ground.

④ **Pads for LED Expansion**.

⑤ **Pads for Pushbutton Switch Expansion**.

⑥ **Range Test Orientation Indicator**. Optimum RF performance is achieved when the BoosterPack module is oriented vertically with the arrow pointing upward.

⑦ **Pads for MCU expansion**. Add a microcontroller for standalone operation without the LaunchPad board. See section 2.1.2 for details.

⑧ **Pads for External Power Source**. Supply power from a battery for standalone operation without the LaunchPad board. See section 2.1.2 for details.

2.1. Hardware Installation

2.1.1. AIR BoosterPack with LaunchPad

The following steps need to be performed on each LaunchPad to be used with a BoosterPack:

1) If not already populated, solder both 10-pin male headers provided in the LaunchPad kit onto the LaunchPad's J1 and J2 breakout connections.
2) For LaunchPad Rev1.4 and earlier, remove jumpers TXD and RXD from J3 and install crossover jumpers (not included) onto the same group of pins to make the following connections:
    a. Connect J3.3 to J3.6
    b. Connect J3.5 to J3.4
3) Ensure the VCC jumper is populated on J3. Jumpers RST and TEST also need to be installed when programming the microcontroller or debugging software using IAR Embedded Workbench.
4) Replace the existing MSP430 device on the LaunchPad with a preprogrammed MSP430G2553 device provided in the AIR BoosterPack kit. Be sure the microcontroller is oriented properly by ensuring the pin 1 indicator on the device matches the indicators on the socket and LaunchPad silkscreen.
5) Install the AIR BoosterPack onto the LaunchPad board. Ensure the BoosterPack is oriented correctly. The text on both boards should be in the same direction and the rocket logo on the BoosterPack should be directly above the rocket on the LaunchPad board.

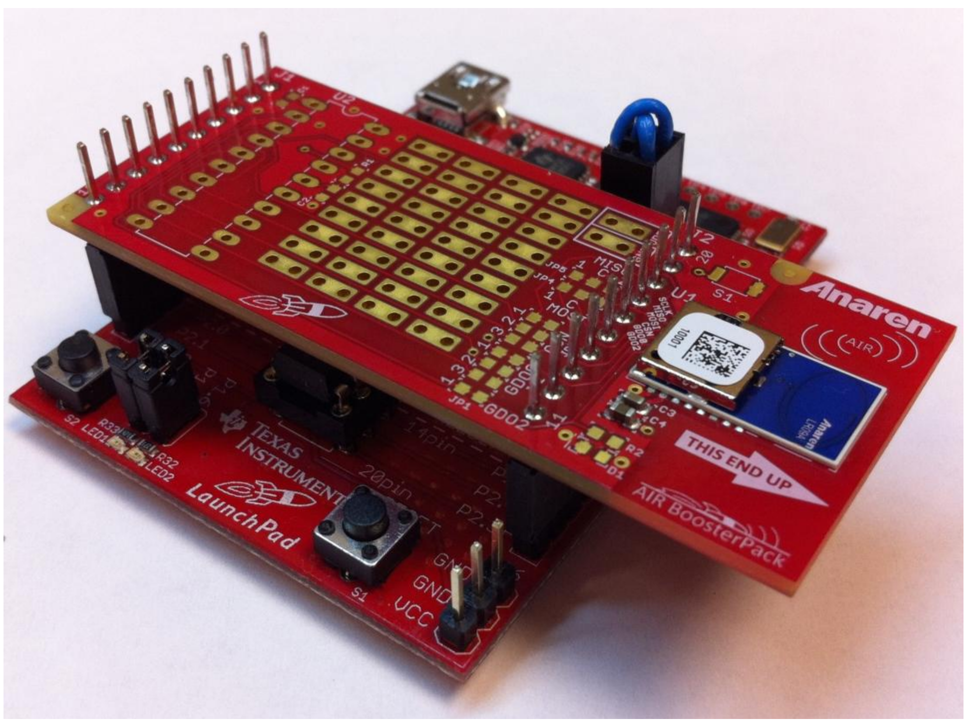


Figure 3 – Installed BoosterPack

2.1.2. AIR BoosterPack Standalone Operation

The BoosterPack comes with expansion pads for upgrading to a standalone configuration that allows operation without the need for the LaunchPad board. Follow the directions below to upgrade the BoosterPack for standalone operation. See section 4.6 for a complete component listing.

1) Install pushbutton tactile switch S1 (not provided). The button is needed to pair the radio with another radio.
2) Install LED D1 and resistor R2 (all not provided). This step may be skipped if a visual indicator is not required.
3) Install a 20-pin DIP socket at U2 (not provided). Ensure the pin 1 indicator notch on the socket aligns with the BoosterPack silkscreen. This step may be skipped if a microcontroller is to be soldered directly to the board.
4) Install capacitors C1, C2 and resistor R1 (all not provided). These components are required for power supply decoupling and generating reset during power-up.
5) Insert a preprogrammed MSP430G2553 device (provided in the AIR BoosterPack kit) into the socket installed in step 3 above. Be sure the microcontroller is oriented properly by ensuring the pin 1 indicator on the device matches the indicators on the socket and BoosterPack silkscreen. If a socket is not desired, solder the MSP430 device to the board at U2.
6) Solder power supply wires or a power connector (all not provided) to the VDD and GND connections on the BoosterPack board. The BoosterPack does not provide any protection against reverse polarity or overvoltage, so it is up to the user to ensure an external supply is properly connected and stays within the operating range specified in section 4.1 Electrical Characteristics.

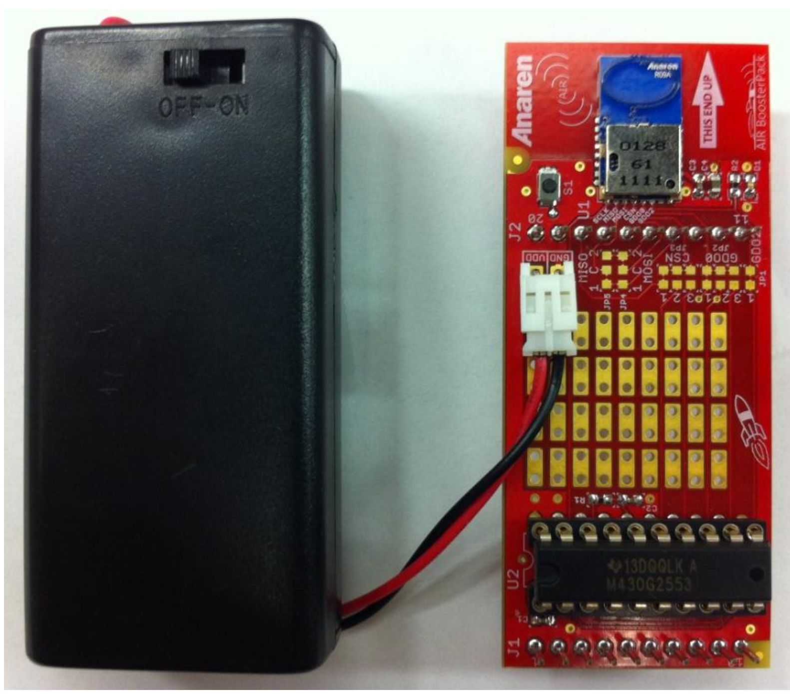


Figure 4 - Standalone BoosterPack

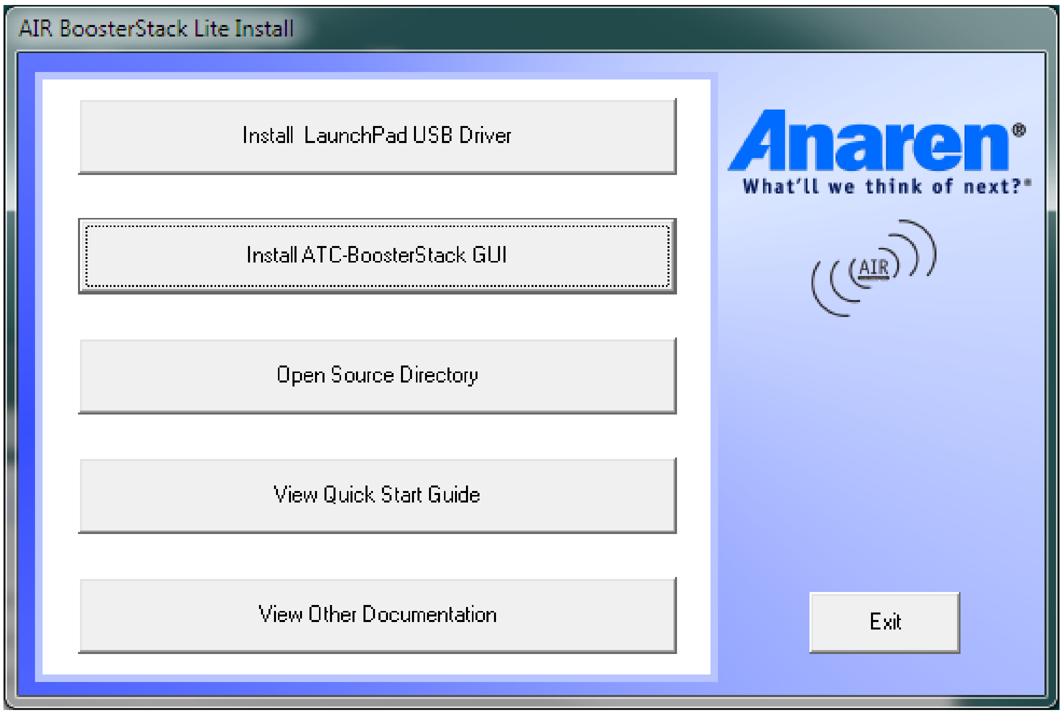
2.2. Software Installation

System Requirements

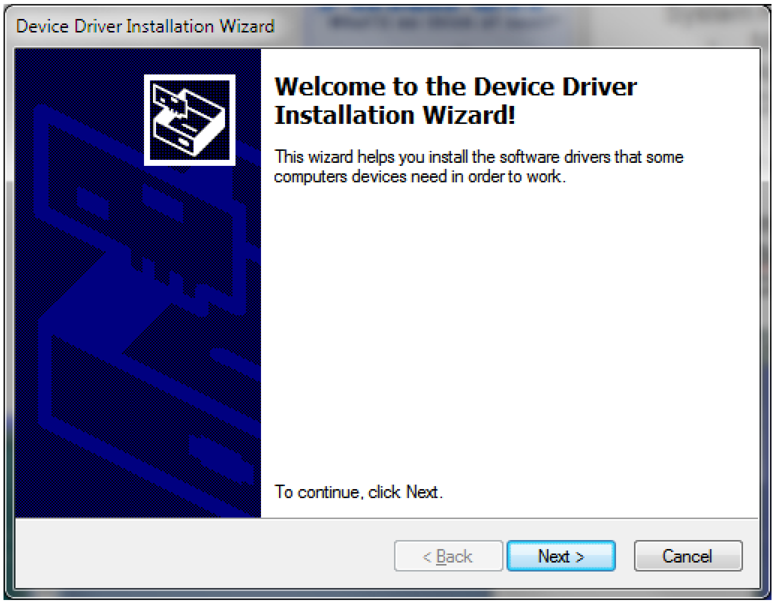
- Microsoft Windows XP SP3 or later operating system (32-bit and 64-bit supported)
- At least 100MB HDD space available
- CD drive

Follow the instructions below to install the BoosterStack software.

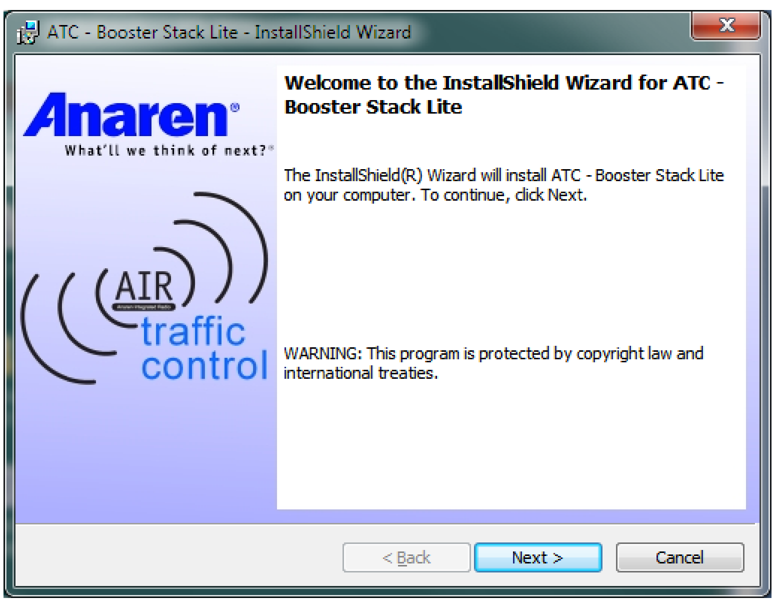
1) Insert the BoosterStack CD provided in the Anaren BoosterPack kit. The AIR BoosterStack Lite Install utility should automatically open. If it does not, browse to the CD drive and double-click on AutoRun.exe to run the installer.



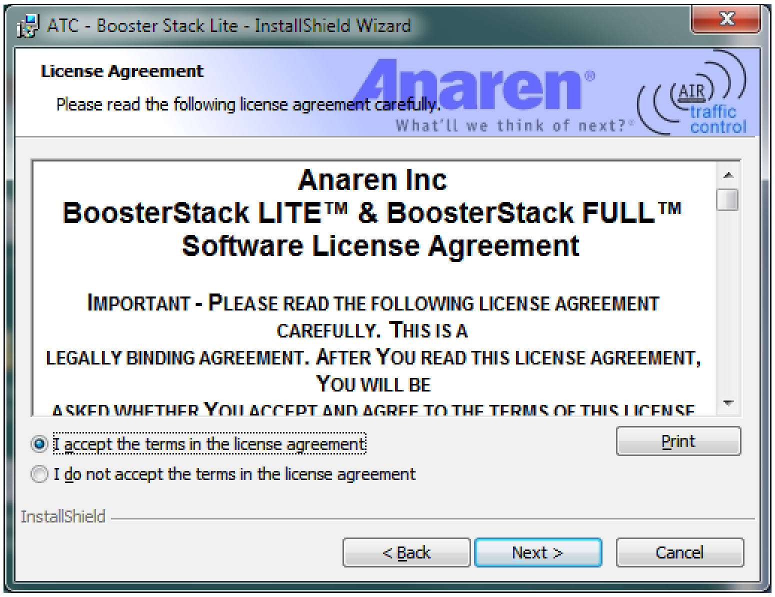
2) Click on ‘Install LaunchPad USB Driver’ to install the MSP430 Application UART driver needed to communicate with the LaunchPad board via serial COM port (USB virtual COM port). Click ‘Next’ in the Device Driver Installation Wizard to install the driver. The installer will automatically detect whether the operating system is 32-bit or 64-bit and install the appropriate device driver. Click ‘Finish’ when the installation has completed.



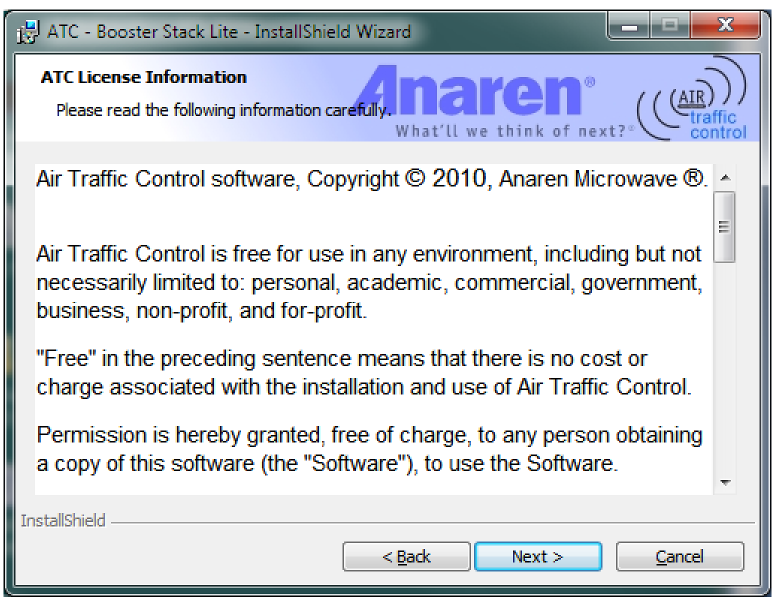
3) Click on ‘Install ATC-BoosterStack GUI’ to install the application software. Click ‘Next’ in the ATC - BoosterStack Lite InstallShield Wizard.



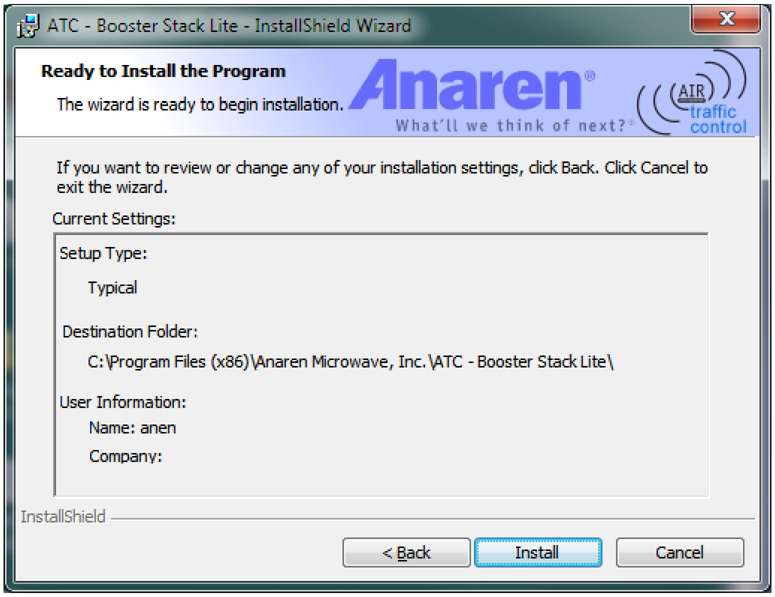
Read the Anaren BoosterStack software license agreement, click the ‘I accept the terms in the license agreement’ button to acknowledge you have read and agree to the license, and then click ‘Next’.



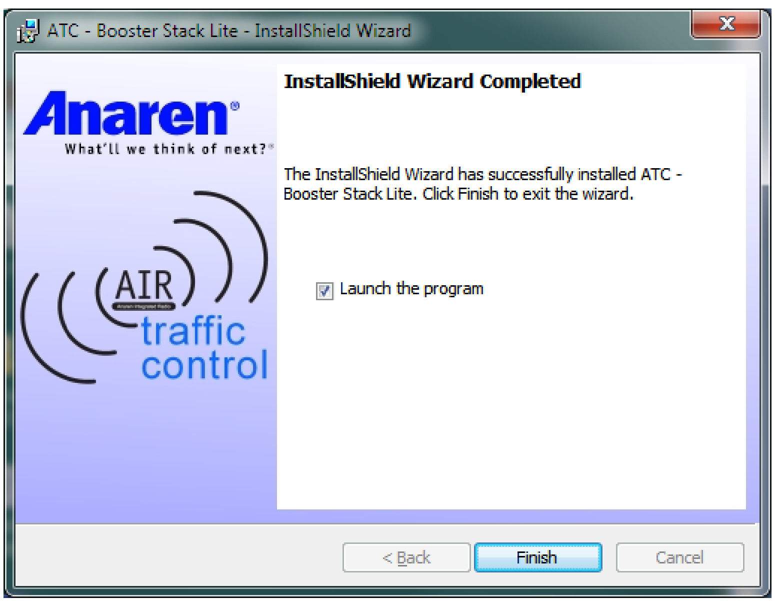
Click ‘Next’ to accept the Third-Party license agreement.



Click 'Install' to start the software installation.



Click 'Finish' when the installation has completed.



Installation of the BoosterStack software is now complete. Please refer to section 3.3 for details on using the GUI.

2.3. IMPORTANT: Required Radio Setup

The AIR BoosterPack must be operated in accordance with local regulations. The firmware preloaded in the MSP430G2553 devices provided with the AIR BoosterPack kit offers two radio operation options; ETSI (default) and FCC/IC. If your location is not covered by either ETSI or FCC/IC then you must check local regulatory codes for how to obtain permission to operate the modules prior to using them.

If covered under FCC/IC regulations you must first connect each module to the PC/laptop and use the GUI to change the "Logical Radio" to "1, A110LR09A, FCC" and click the "Apply Configuration Changes" button. Be sure the “Apply Changes To” pull-down menu is set to option 0 to apply the changes only to the local radio and not option 1 which is for “Remote + Local”. The new settings are stored in non-volatile (i.e. Flash) memory and the devices will continue to power-up with FCC/IC compliant settings. See section 3.3 for more information on using the GUI.

2.4. Using the Demo Application

Follow these directions to start using the demo application.

1) Connect the LaunchPad/BoosterPack assemblies to USB ports.
2) Launch the ATC – BoosterStack Lite software. The GUI screen will appear and the GUI will attempt to connect to the COM port. Once the connection is established, the GUI will populate all the tabs and will be ready to use. A trace for the node temperature will appear for the node itself (see section 3.3 for further details).
3) To identify a particular node, click on ‘Green LED’ check box, which will send a command to that module to illuminate the Green LED on its Launchpad.
4) Change one of the module’s ‘Application State’ to Sensor (by default, both the modules will be Hub).
5) Pair the Sensor and Hub modules. This will display the information of the paired module into the ‘Available Node List’ of the corresponding module (see section 3.1.3 for further details).
6) The traces for Local RSSI, Remote RSSI and Chip Temperature will appear for the paired node.
7) Perform necessary changes to the configuration settings using the populated tabs (see section 3.3 for further details).

3. AIR BoosterStack Software Application

The software application provided with the AIR BoosterPack kit, called AIR BoosterStack, demonstrates an example sensor network as well as network status reporting. The sensor network uses a star topology with one hub node and up to four sensor nodes. Temperature information from each sensor is periodically transmitted to the hub which forwards the data to a Graphical User Interface (GUI) running on a PC/laptop. The GUI displays the temperature readings from each sensor as well as Received Signal Strength Indicator (RSSI) values for each radio link.

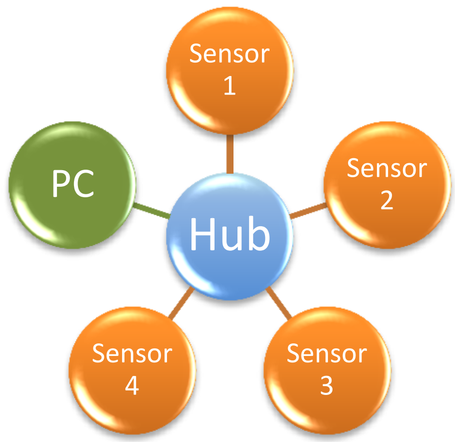
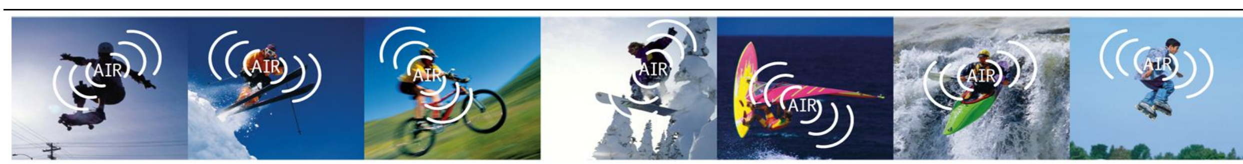


Figure 5 - Network Topology



3.1. AIR BoosterStack Firmware

3.1.1. Low-Power Sensor Application

The sensor is intended to be a low-power device capable of long-term operation while powered from a battery. To achieve this goal, the radio and microcontroller are placed into sleep mode whenever possible. The AIR BoosterStack demo application is configured to periodically (approximately once every second) wake up the processor, perform a temperature measurement, wake up the radio, transmit the data, wait for an acknowledgement, and then finally go back to sleep. For maximum battery life, the debug circuitry on the LaunchPad board needs to be powered down. This can be accomplished by removing the VCC jumper from LaunchPad J3 and applying power to LaunchPad J6. As an alternative, the AIR BoosterPack module can be operated in a standalone mode (see section 2.1.2 for details).

Since it spends the majority of its time in sleep mode, the sensor is responsible for initiating all data transfers to/from the hub. After transmitting its data, the sensor switches to receive mode to listen for an acknowledgement from the hub. The hub may also send any pending messages to the sensor at this time. If data or a command is received from the hub, the sensor will send an acknowledgement and then go back to sleep. If the sensor receives an acknowledgement only (i.e. no payload data) or doesn't receive anything at all within a predefined timeout period it will terminate the communication cycle and go back to sleep. See Figure 6 and Figure 7 for flow diagrams depicting the sensor's behavior.

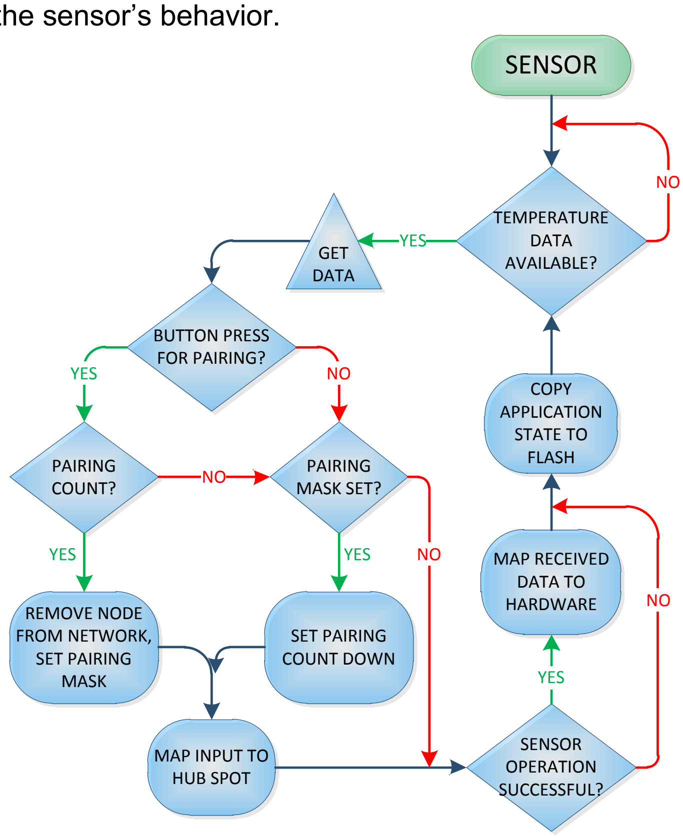


Figure 6 - Sensor Application Flow Diagram

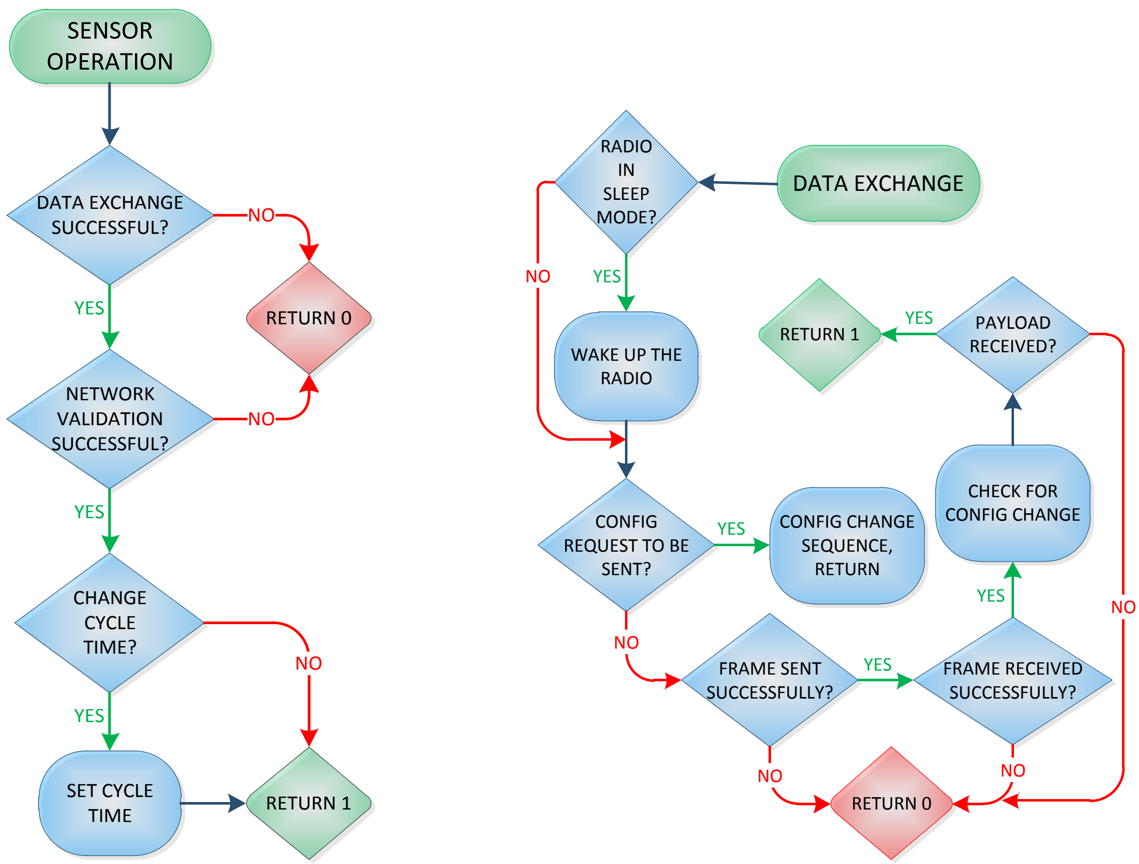


Figure 7 - Sensor Operation and Data Exchange Function Flow Diagrams

Sensor Operation: The sensor initiates a data transfer and if the data exchange is successful, it proceeds with network validation to see if pairing is needed or whether the replying hub node is already in its network. Upon a successful network validation, the sensor checks if the hub is requesting a change to the sensor's data acquisition rate and if so adjusts its timer to the new value and returns true.

Data Exchange: If necessary, the sensor wakes its radio from sleep mode. If a configuration change request is to be sent, it continues with the configuration change sequence and returns. Otherwise it sends the data frame to the hub and waits for a response. If a configuration change request is received from the hub, the sensor follows the configuration change sequence and if payload data is received, it returns true.

3.1.2. Hub Application

The primary function of the hub node is to provide an interface between the GUI running on a PC/laptop and all sensor nodes within the network. The hub has basic network routing functionality built-in which allows pairing with up to four sensors (limited only by available microcontroller resources). Unlike the sensors, it is assumed that the hub is always powered by a USB cable plugged into the LaunchPad board. Data transfers from each sensor are asynchronous events, so the hub spends most of its time with its radio in receive mode listening for packets from the sensor nodes. In general the hub will only transmit a packet after it has received a message from one of the sensors. The only exception to this is when it is in the middle of a configuration change sequence and it sends ping messages to the sensor in order to determine if the configuration change was successful. Figure 8 and Figure 9 show the hub application software flow diagrams.

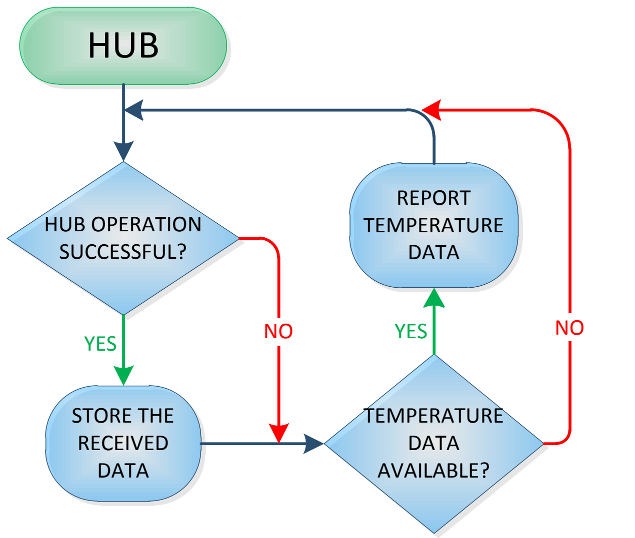
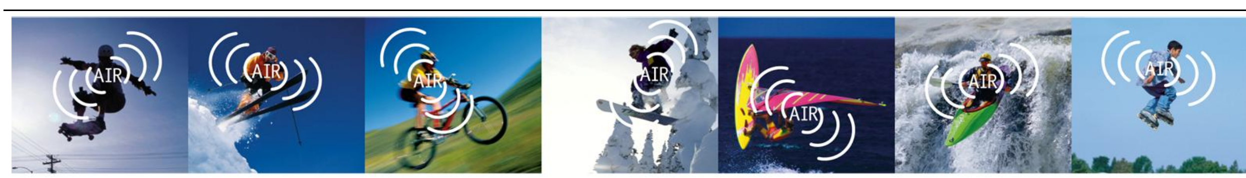


Figure 8 - Hub Application Flow Diagram

Hub Operation: If the hub successfully completes the Listen operation, it proceeds to network validation (see section 3.1.3 for details) to see if pairing is needed or whether the node is already in its network. If network validation is successful, it returns true.

Listen Function: When the hub receives a valid frame from a paired sensor it checks to see if the frame contained a payload. If so, it creates a frame with its own payload data (if any) and an acknowledgement to the payload received from the sensor. If there is a request for a configuration change, either from the sensor or the GUI, it enters the configuration change sequence (see section 3.1.4). Otherwise it sends its frame, enters into receive mode to listen for an acknowledgement from the sensor, then returns to the application.

If it receives just an acknowledgement from the sensor (i.e. no payload), then it does not create a frame in response. Instead, it enters into receive mode to listen for the next frame from the sensor and returns to the application.



If neither payload nor acknowledgement is received within a specified timeout period, the radio is checked to see if it is still in receive mode. If not in receive, the radio is commanded to go to receive mode. In either case the hub returns to the application to listen for the next frame from the sensor.

Along with the temperature data, the hub receives the following information from the sensor and sends the same information about itself back to the sensor. Thus for status updates, the hub provides both the local and remote information to the GUI. It also reports its own temperature data to the GUI at regular intervals.

1. Received Signal Strength Indicator (RSSI)
2. Link Quality Index (LQI)
3. Frequency Offset
4. Power
5. Last Used Channel
6. Frame Counter

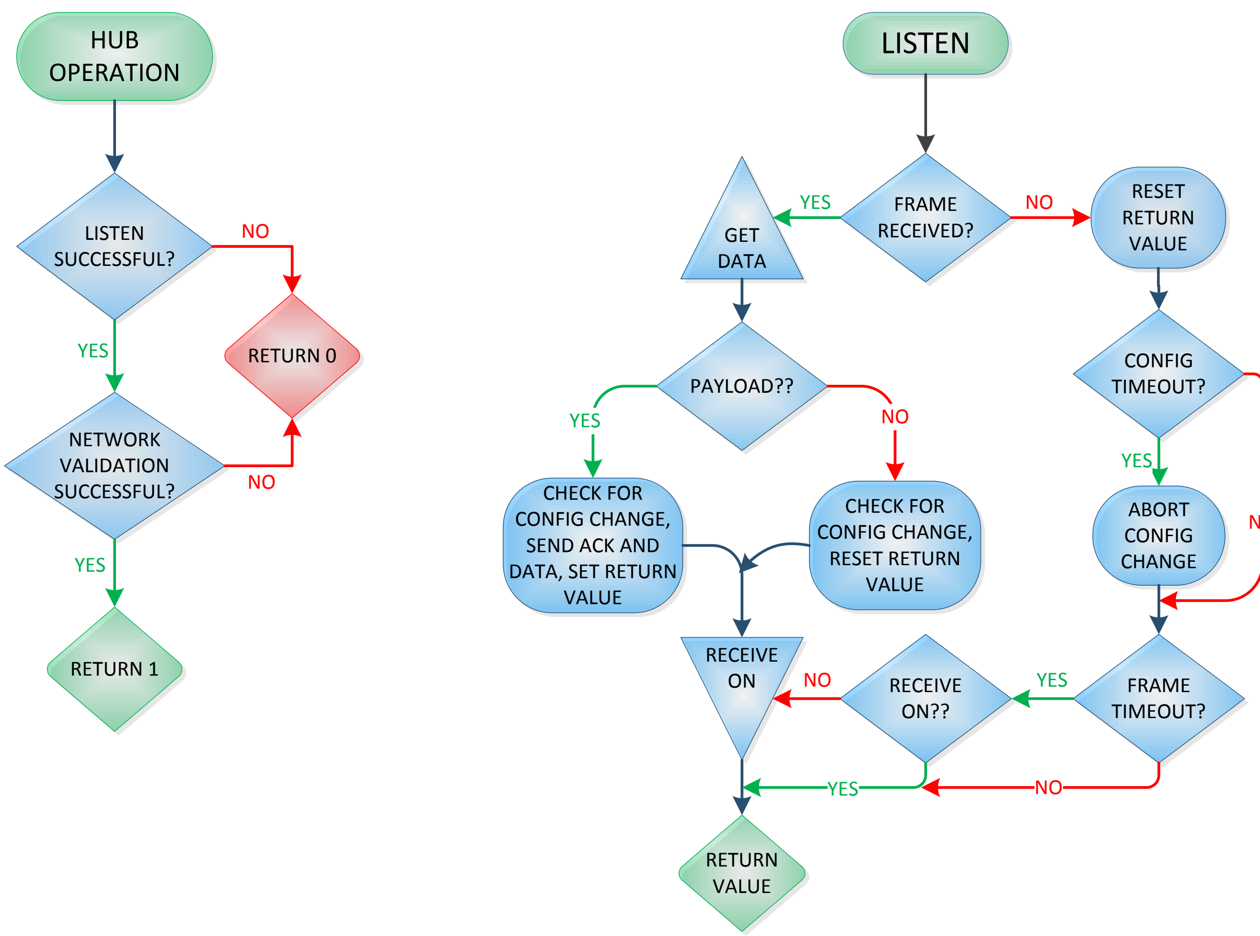


Figure 9 - Hub Operation and Listen Function Flow Diagrams

3.1.3. Device Pairing

The protocol is started by pairing the hub with the sensor node(s). Although the hub stays in receive mode to listen for frames from sensors, it will not respond to a frame that is sent from a non-paired sensor. The only exception to this is during the pairing mode when it receives a frame from a sensor requesting to join the network.

The following steps are used to pair a sensor with the hub:

1. For the hub, ensure the App State is set to Hub and then press the 'Discover Nodes' button in the GUI. The hub will search for sensor nodes requesting to join the network for approximately 10 seconds. During this time, most of the GUI controls are disabled (i.e. grayed out) preventing other user commands from being issued to the BoosterPack. The controls are re-enabled at the end of the node discovery sequence.
2. The sensor has two methods for pairing with the hub. While the hub is scanning for sensor nodes:
    a. Press and hold the pushbutton switch on the sensor (S2 on LaunchPad) for at least 2 seconds as described in section3.1.6. If the pairing was successful, the sensor will be added to the hub's Available Node List.

    -OR-

    b. Open a second session of the ATC BoosterStack Lite GUI for the sensor node and ensure the App State is set to Sensor. Press the 'Discover Nodes' button in the GUI. As with the hub, the sensor will request to join a network for approximately 10 seconds and the controls will be disabled. If the pairing was successful, the hub will be added to the sensor's Available Node List.
3. Once the modules are paired, the RSSI and temperature from each radio will be displayed in their respected graphs in the GUI.

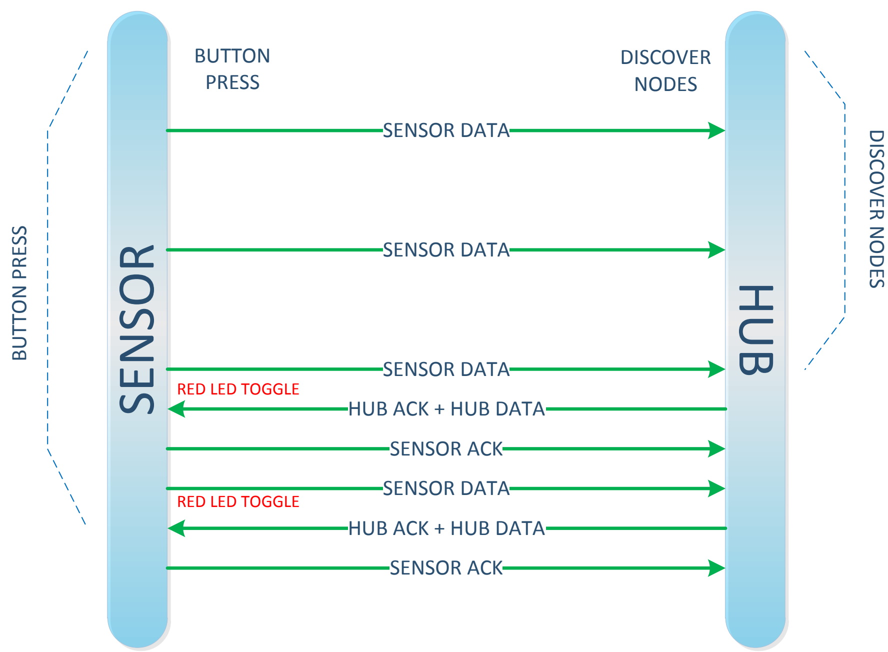


Figure 10 - Pairing Diagram

Sensor Pairing Mechanism: The sensor is placed into pairing mode by pressing the button switch on the module. The pairing mask is set to zero by default; the button press assigns it a pairing mask of 0xFFFFFFFF. The user can manually set the pairing mask if desired. If the pairing mask is set, a pairing-bit is set in the frame indicating that this node requesting to join the network and the data exchange is initiated by the sensor. If the hub is discovering nodes at the same time, it will respond to the frame sent by the sensor. The sensor validates the received frame and proceeds with "Network Validation".

Hub Pairing Mechanism: The hub is placed into pairing mode by pressing the "Discover nodes" button on the GUI. The pairing mask is set to zero by default; the button press assigns it a pairing mask of 0xFFFFFFFF. The user can manually set the pairing mask if desired. When the hub receives a frame from the sensor, it checks if the pairing-bit is set in the received frame. If so, it marks that frame as valid, sets the pairing-bit in its acknowledgement frame and sends it back to the sensor. The hub then proceeds with "Network Validation".

Network Validation: If the pairing-bit is set in the received frame, it indicates that the pairing is requested. Each device then checks if the other node is currently in its own network. If not, the node ID of the node to be paired with is validated. The node ID cannot have content outside the mask, but it must have content within the mask. If the condition is satisfied, it (Sensor/Hub) finds a spot for the node to be added in the network, and adds it in. The module pairing is now complete. The sensor is allowed to pair with only one hub, whereas the hub can pair with more than one sensor (up to four).

The sensor sets its mask back to the default value (zero) to prevent pairing with other nearby hubs, and stores it's Application State (Sensor) into flash to keep it in sensor mode as the pairing has been successful. The hub reports the change in the network to the GUI.

3.1.4. Configuration Change

The AIR BoosterStack software is capable of changing radio configuration settings on the local node connected to the GUI as well as any paired remote nodes. The remote node does not need to be connected to a computer to make the change. A configuration change can be initiated from either the hub or a sensor. In either case, the change is applied to only a single RF link. That is, if a hub has multiple sensors paired with it, only the selected sensor will make the change. Note that this will effectively disconnect all other sensors paired with the hub. There are two options for making the changes on the remaining sensors. The first method requires that each sensor node be connected to a GUI. At that point each node can be manually changed locally to the new settings. The second option, which can be done wirelessly, requires the hub to be locally changed back to the same settings as the sensor to be changed. Then both hub and sensor can be updated with new settings that match those of the first sensor changed. At this point all three nodes should have the same settings. These steps can be repeated for any remaining sensors paired with the hub until all nodes in the network have been updated.

Figure 11 shows the handshaking involved in the configuration change sequence. The sequence is explained below in two parts based on the Initiator operation (requests a change) and Receiver operation (receives change request).

Initiator Operation: It starts the configuration change sequence by sending out a ‘Config Change Request’ with the information of the intended new configuration. It waits for ‘Config Change Request-ACK’. This is shown in the diagram with the solid green arrows (labeled). The green color represents the case where the request/ACK is received successfully, while the red arrows represent the failure. If the acknowledgement is not received within the specified timeout period, it concludes that the Receiver never received the ‘Config Change Request’, so it aborts the configuration change sequence and returns ‘failure’ (red dotted arrow). If ‘Config Change Request-ACK’ is received, it switches to the new configuration itself and sends out a ‘Config Change Ping’. It awaits the response to the ping, a ‘Config Change Ping-ACK’. If the acknowledgement is received within specified timeout, it concludes that the configuration change went through as intended and returns a ‘success’ (green dotted arrow). This ‘Ping-ACK’ handshake is again shown in solid green arrows, which represent that both are received successfully by Initiator/Receiver. Red arrows under either of them represent the case of failure.

If ‘Config Change Ping-ACK’ is not received within specified timeout, the Initiator starts a ‘Config Ping Sequence’. It keeps sending the ‘Config Change Ping’ until the specified configuration timeout occurs or it receives the ‘Config Change Ping-ACK’, whichever is the earliest. If the ‘Config Change Ping-ACK’ is received, it concludes that either the first ‘Config Change Ping’ or first ‘Config Change Ping-ACK’ did not go through as expected, but did work during the ‘Config Ping Sequence’, and returns a ‘success’ (green dotted arrow). The ‘Config Ping Sequence’ is represented by green dotted lines (success/failure cannot be guaranteed at given time) in the diagram.

If ‘Config Change Ping-ACK’ is not received and the specified configuration timeout occurred, it assumes that the Receiver never switched to the new configuration in the first place, so it switches to the old(previous) configuration itself to try to establish the communication again. It again starts a ‘Config Ping Sequence’, now on the old (previous) configuration. This ‘Ping-ACK’ handshake is again shown in solid green arrows, which represent that both are received successfully by Initiator/Receiver. Red arrows under either of them represent the case of failure. It keeps sending the ‘Config Change Ping’ until the specified configuration timeout occurs or it receives the ‘Config Change Ping-ACK’, whichever is the earliest. If the ‘Config Change Ping-ACK’ is received, it concludes that the receiver never even switched to the new configuration for some reason. So it stays on the old (previous) configuration to continue the communication, and returns a ‘failure’ (red dotted arrow). The ‘Config Ping Sequence’ is represented by green dotted lines (success/failure cannot be guaranteed at given time) in the diagram.

If ‘Config Change Ping-ACK’ is not received and specified configuration timeout occurred even at this time, it assumes that the Receiver did switch to the new configuration but the first ‘Config Ping Sequence’ did not go through for some reason. So in order to try to establish the connection again, it switches to the new configuration itself, and concludes that the intended configuration change went through and returns ‘success’ (green dotted arrow).

Receiver Operation: At the receiving end, the configuration change sequence begins when a ‘Config Change Request’ is received from the Initiator. It sends ‘Config Change Request-ACK’ as a response, and changes to the new configuration based on the information received from the Initiator. It awaits the ‘Config Change Ping’, or the specified configuration timeout to occur, whichever is the earliest. If it receives ‘Config Change Ping’, it sends out a ‘Config Change Ping-ACK’ and returns to application with ‘success’ (red dotted arrow).

If 'Config Change Ping' is not received within specified configuration timeout, it concludes that the Initiator never received the 'Config Change Request-ACK', so it must not have switched to the old(previous) configuration. The Receiver then reverts back to the old (previous) configuration and waits for the 'Config Change Ping' or the specified configuration timeout to occur, whichever is the earliest. In either case, it returns to the application with 'failure' (red dotted arrow).

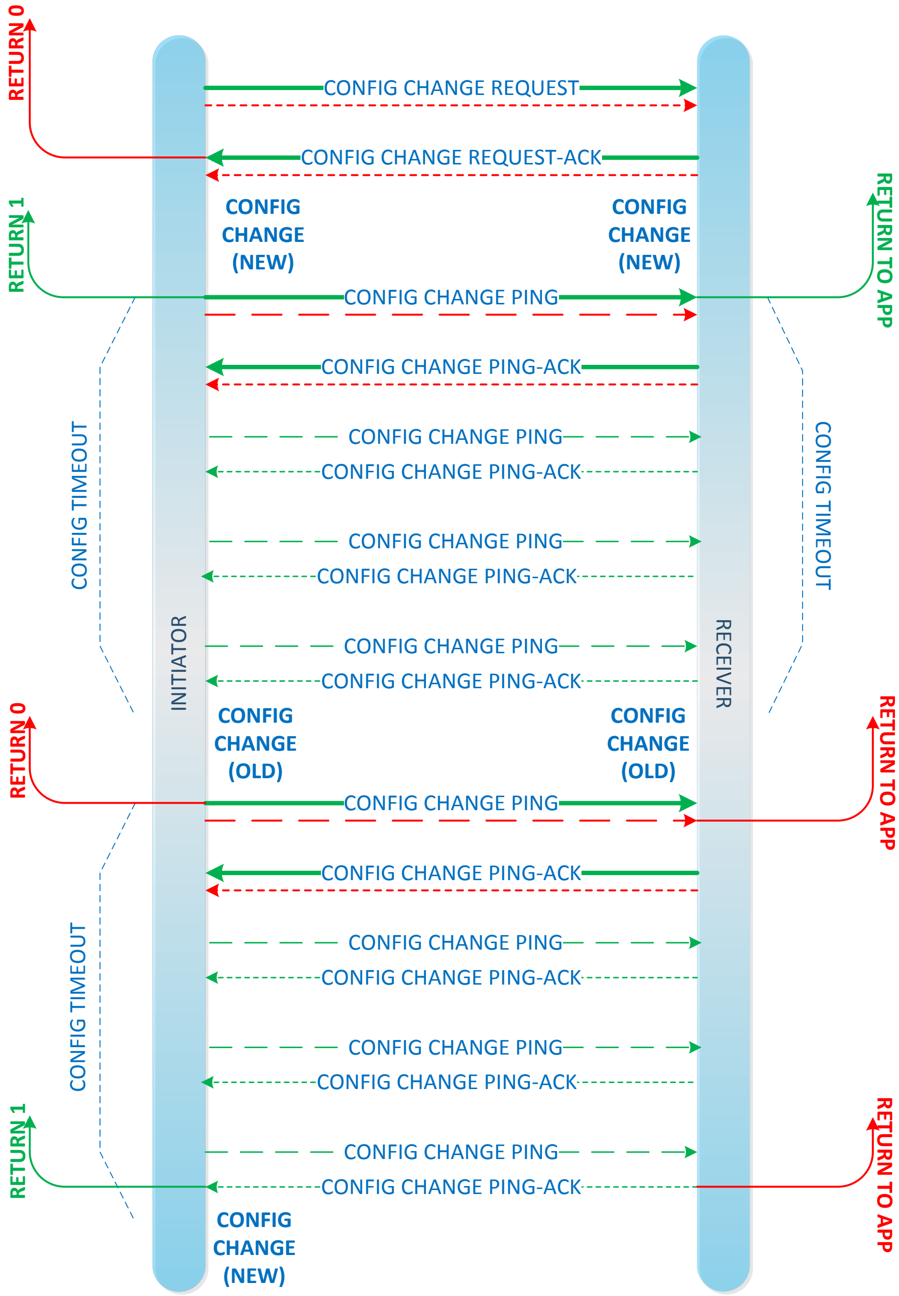


Figure 11 - Configuration Change Diagram

3.1.5. Remote Control Applications

Green LED: A basic remote control function is provided which allows turning on/off the green LED on the LaunchPad board for any node in the network. From the GUI for the hub, clicking the “GreenLED” checkbox for a node in the “Available Node List” will turn the LED on (box checked) or off (box not checked) at the selected node.

Similarly, this feature can also be used from the GUI attached to a sensor. The only limitation is that a sensor cannot turn on the LED of another sensor in the network. It can only control its own LED and that of the hub it is paired with.

Since multiple nodes can be connected to a single PC/laptop with as many GUIs running simultaneously, this feature can be used as an identifier for determining which node is connected to a specific GUI.

Red LED: Momentarily pressing the pushbutton switch (S2) on the LaunchPad board will cause the red LED on that node to toggle. It will also change the status of the “RedLED” checkbox for the chosen node in the “Available Node List”.

3.1.6. Additional Features

Pushbutton Switch: The pushbutton switch (S2) on the LaunchPad board supports multi-click and press-hold capabilities.

1) **Single Click:** This toggles the red LED as stated in section 3.1.5. The duration of the button press must be less than approximately 500 milliseconds.
2) **Double Click:** Rapidly pressing the switch twice will indicate the current application state of the node by blinking the red LED twice for the hub and three times for the sensor. The duration of each button press and the time between them must be less than approximately 500 milliseconds.
3) **Triple Click:** Rapidly pressing the switch three times will toggle between hub and sensor modes. The red LED will blink after the transition to indicate the new application state. The duration of each button press and the time between them must be less than approximately 500 milliseconds.
4) **Hold 2 Seconds:** Press and hold the switch for two seconds to have the node enter pairing mode. The red LED will blink at approximately 2Hz rate and 10% duty cycle to indicate the button was held long enough. At this point the button may be released. The node will continue to scan for other nodes for approximately 10 seconds.
5) **Hold 5 Seconds:** Press and hold the switch for five seconds to have the node change to sensor mode followed by pairing mode. The red LED will blink at approximately 10Hz rate and 50% duty cycle to indicate the button was held long enough. At this point the button may be released. The node will continue to scan for other nodes for approximately 10 seconds.
6) **Hold 15 Seconds:** Press and hold the switch for 15 seconds to have the node clear all of its stored settings. The red LED will blink at approximately 1Hz rate and 50% duty cycle to indicate the button was held long enough and may be released. This sets all values to factory default, including application state and virtual radio selection (i.e. frequency band). ***The required radio setup as described in section 2.3 must be performed again after clearing the node’s settings.***

3.2. Transceiver Configurations

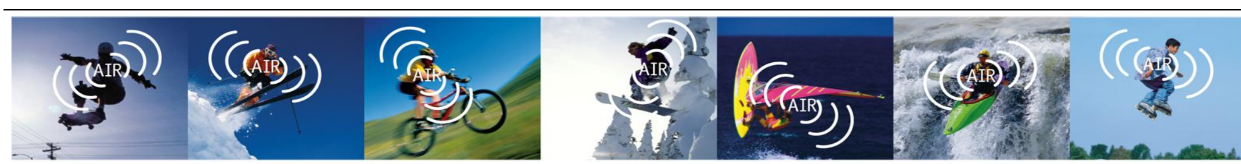
The Boosterpack uses the Anaren A110LR09A module, which is based on the TI CC110L chip. The Anaren module has been certified for use under FCC/IC and ETSI using specific register settings within the transceiver chip. The registers are shown in Figure 12. Please note that BoosterStack does not allow any modifications of these, but instead only allows a choice of predefined values. The register settings used in the predefined sets are provided for information only in Table 1. In addition to these settings, predefined limits for RF output power settings (PATABLE) and maximum duty cycle settings are used for the ETSI configurations to be compliant with the duty cycle requirements of ETSI. Note that BoosterStack does not use listen before talk (LBT).

Legend:

Certification is valid for any value choosen	Use the provided certified settings as other may degrade performance	Use sets of values provided, others are a violation of certification	Any modification of this value is a violation of the certification and the customer is responsible for optaining their own certification	Unused/undocumented function. The provided default value must be written. No assumption should be made on the value read from this field	Read only register (burst mode read only, command strobes otherwise).

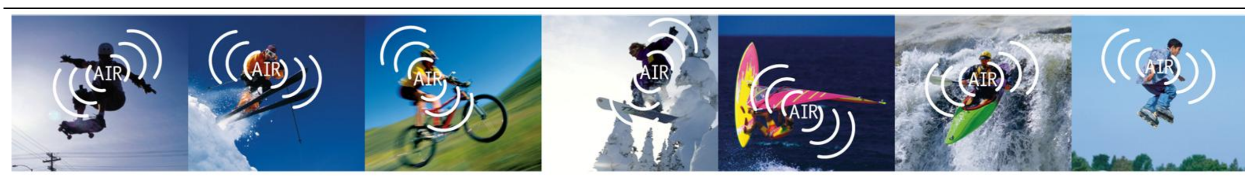
Register Name	Register Address (Hex)	Retained during sleep	7	6	5	4	3	2	1	0
			Bit Fields Within Register							
IOCFG2	00	1	0	GDO2_INV	GDO2_CFG					
IOCFG1	01	1	GDO_DS	GDO1_INV	GDO1_CFG					
IOCFG0	02	1	0	GDO0_INV	GDO0_CFG					
FIFOTHR	03	1	Reserved	ADC_RETENTION	CLOSE_IN_RX		FIFO_THR			
SYNC1	04	1	SYNC_MSB							
SYNC0	05	1	SYNC_LSB							
PKTLEN	06	1	PACKET_LENGTH							
PKTCTRL1	07	1	PQT			0	CRC_AUTOFLUSH	APPEND_STATUS	ADR_CHK	
PKTCTRL0	08	1	0	WHITE_DATA	PKT_FORMAT		0	CRC_EN	LENGTH_CONFIG	
ADDR	09	1	DEVICE_ADDR							
CHANNR	0A	1	CHANNR							
FSCTRL1	0B	1	0		0	FREQ_IF				
FSCTRL0	0C	1	FREQOFF							
FREQ2	0D	1	FREQ[23:22]=0		FREQ[21:16]					
FREQ1	0E	1	FREQ[15:8]							
FREQ0	0F	1	FREQ[7:0]							
MDMCFG4	10	1	CHANBW_E		CHANBW_M		DRATE_E			
MDMCFG3	11	1	DRATE_M							
MDMCFG2	12	1	DEM_DCFILT_OFF	MOD_FORMAT			MANCHESTER_EN	SYNC_MODE		
MDMCFG1	13	1	0	NUM_PREAMBLE			0			
Not Used	14	1								
DEVIATN	15	1	0	DEVIATION_E			0	DEVIATION_M		
MCSM2	16	1	0			RX_TIME_RSSI	RX_TIME_QUAL	7		
MCSM1	17	1	0		CCA_MODE		RXOFF_MODE		TXOFF_MODE	
MCSM0	18	1	0		FS_AUTOCAL		PO_TIMEOUT		0	XOSC_FORCE_ON
FOCCFG	19	1	0		FOC_BS_CS_GATE	FOC_PRE_K		FOC_POST_K	FOC_LIMIT	
BSCFG	1A	1	BS_PRE_K		BS_PRE_KP		BS_POST_K	BS_POST_KP	BS_LIMIT	
AGCCTRL2	1B	1	MAX_DVGA_GAIN		MAX_LNA_GAIN			MAGN_TARGET		
AGCCTRL1	1C	1	0	AGC_LNA_PRIORITY	CARRIER_SENSE_REL_THR		CARRIER_SENSE_ABS_THR			
AGCCTRL0	1D	1	HYST_LEVEL		WAIT_TIME		AGC_FREEZE		FILTER_LENGTH	
Not Used	1E	1								
Not Used	1F	1								
Reserved	20	1	31 (11111)					0	WOR_RES	
FREND1	21	1	LNA_CURRENT		LNA2MIX_CURRENT		LODIV_BUF_CURRENT		MIX_CURRENT	
FREND0	22	1	0		LODIV_BUF_CURRENT_TX		0	PA_POWER		
FSCAL3	23	1	FSCAL3[7:6]		CHP_CURR_CAL_EN		FSCAL3[3:0]			
FSCAL2	24	1	0		VCO_CORE_H_EN	FSCAL2				
FSCAL1	25	1	0		FSCAL1					
FSCAL0	26	1	0	FSCAL0						
RCCTRL1	27	1	0	RCCTRL1						
RCCTRL0	28	1	0	RCCTRL0						
Reserved	29	0	FSTEST							
Reserved	2A	0	PTEST							
Reserved	2B	0	AGCTEST							
TEST2	2C	0	TEST2							
TEST1	2D	0	TEST1							
TEST0	2E	0	TEST0[7:2]						VCO_SEL_CAL_EN	TEST0[0]
Not Used	2F									
PARTNUM	30	1	PARTNUM							
VERSION	31	1	VERSION							
FREQOFF_EST	32	0	FREQOFF_EST							
CRC_REG	33	0	CRC_OK	Reserved						
RSSI	34	0	RSSI							
MARC_STATE	35	0				MARC_STATE				
Reserved	36	0								
Reserved	37	0								
PKTSTATUS	38	0	CRC_OK	CS		CCA	SFD	GDO2		GDO0
Reserved	39	0								
TXBYTES	3A	0	TXFIFO_UNDERFLOW	NUM_TXBYTES						
RXBYTES	3B	0	RXFIFO_OVERFLOW	NUM_RXBYTES						
Reserved	3C	0								
Reserved	3D	0								
PATABLE	3E	1	PATABLE							
FIFO	3F	0	TXFIFO/RXFIFO							

Figure 12 - A110LR09A Transceiver Registers



			ETSI		FCC/IC				
Item Number	Adr	Register Name	M7_GFSK_38kB_18kHzDev_70kHzRxBW	M12_GFSK_4.8kB_13kHzDev_60kHzRxBW	ML4_2FSK_1_2kB_237kHzDev_675kHzRxBW	ML5_2FSK_38kB_237kHzDev_675kHzRxBW	ML6_2FSK_100kB_237kHzDev_675kHzRxBW	ML7_2FSK_250kB_237kHzDev_844kHzRxBW	Allow Mask
0	00	IOCFG2	29	29	29	29	29	29	FF
1	01	IOCFG1	2E	2E	2E	2E	2E	2E	FF
2	02	IOCFG0	06	06	06	06	06	06	FF
3	03	FIFOTHR	07	07	07	07	07	07	FF
4	04	SYNC1	D3	D3	D3	D3	D3	D3	FF
5	05	SYNC0	91	91	91	91	91	91	FF
6	06	PKTLEN	FF	3D	3D	FF	FF	3D	FF
7	07	PKTCTRL1	04	04	04	04	04	04	FF
8	08	PKTCTRL0	05	05	05	05	05	05	8F
9	09	ADDR	00	00	00	00	00	00	FF
10	0A	CHANNR	00	00	00	00	00	00	FF
11	0B	FSCTRL1	06	05	0C	0C	0D	0D	FF
12	0C	FSCTRL0	00	00	00	00	00	00	FF
13	0D	FREQ2	20	20	21	21	21	21	00
14	0E	FREQ1	25	25	6B	6B	6B	6B	00
15	0F	FREQ0	ED	ED	24	24	24	24	00
16	10	MDMCFG4	EA	F7	15	1A	1B	0D	F0
17	11	MDMCFG3	71	75	75	71	E5	2F	00
18	12	MDMCFG2	13	13	03	03	03	03	07
19	13	MDMCFG1	20	20	21	21	21	21	FC
20	14	MDMCFG0	F8	F8	E5	E5	E5	E5	00
21	15	DEVIATN	33	30	71	71	71	71	88
22	16	MCSM2	07	07	07	07	07	07	FF
23	17	MCSM1	30	30	30	30	30	30	FF
24	18	MCSM0	18	18	18	18	18	18	FF
25	19	FOCCFG	16	16	1D	1D	1D	1D	FF
26	1A	BSCFG	6C	6C	1C	1C	1C	1C	FF
27	1B	AGCCTRL2	43	04	47	47	47	C7	FF
28	1C	AGCCTRL1	4F	4F	40	40	40	40	FF
29	1D	AGCCTRL0	91	91	B0	B0	B0	B0	FF
30	1E	WOREVT1	00	00	00	00	00	00	FF
31	1F	WOREVT0	00	00	00	00	00	00	FF
32	20	WORCTRL	F8	FB	F8	F8	F8	F8	FF
33	21	FREND1	56	56	B7	B7	B7	B7	FF
34	22	FREND0	10	10	10	10	10	10	FF
35	23	FSCAL3	E9	E9	E9	E9	EA	EA	FF
36	24	FSCAL2	2A	2A	2A	2A	2A	2A	FF
37	25	FSCAL1	00	00	00	00	00	00	FF
38	26	FSCAL0	1F	1F	1F	1F	1F	1F	FF
39	27	RCCTRL1	00	00	00	00	00	00	FF
40	28	RCCTRL0	00	00	00	00	00	00	FF
41	29	FSTEST	59	59	59	59	59	59	00
42	2A	PTEST	7F	7F	7F	7F	7F	7F	7F
43	2B	AGCTEST	3F	3F	3E	3F	3F	3C	0
44	2C	TEST2	81	81	88	88	88	88	FF
45	2D	TEST1	35	35	31	31	31	31	FF
46	2E	TEST0	09	09	09	09	09	09	FF
47	2F	0	00	00	00	00	00	00	0

Table 1 -Transceiver configuration settings used in BoosterStack



3.3. AIR BoosterStack GUI

At Startup, the GUI scans for the available COM Ports and establishes a connection with one. It starts sending commands to the module through the COM Port for the following:

Information:

1. Application Information.
2. Configuration Information.
3. Current Configuration Settings Indices.
4. Node IDs of all the nodes in the network.
5. Definitions for each data channel.

Settings:

1. Default Application State.
2. Default Pairing Mask.
3. Response Flags to get serial output from the module

The GUI displays the responses from the module on the Serial Data tab. Once the connection with the module is validated, the COM Port number shows up in the tab ‘Current COM Port’ (3). The ‘Reset Serial Communications’ (4) tab breaks the connection with the current COM port and scans again for the available COM Ports and goes through the startup procedure again with the module corresponding to that COM port.

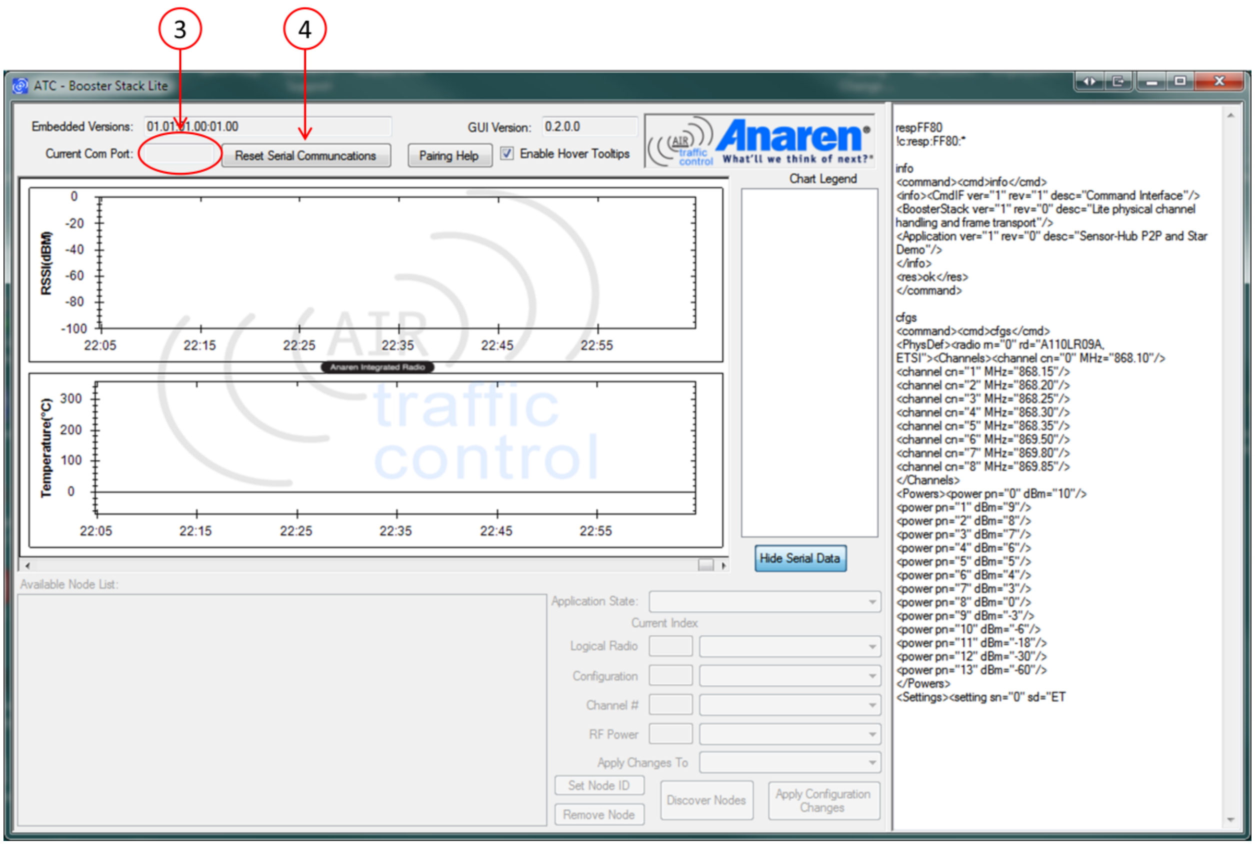


Figure 13 - GUI at Startup

Note: Occassionally the LaunchPad will lose communication with the PC (due to an issue with COM port assignments), resulting in loss of communication between the BoosterStack GUI and BoosterPack module. If this occurs, simply reboot the PC and retry communication.

Until the connection is established between the GUI and the module, none of the GUI tabs are populated. Once the connection is established, all the tabs come up and are ready to use.

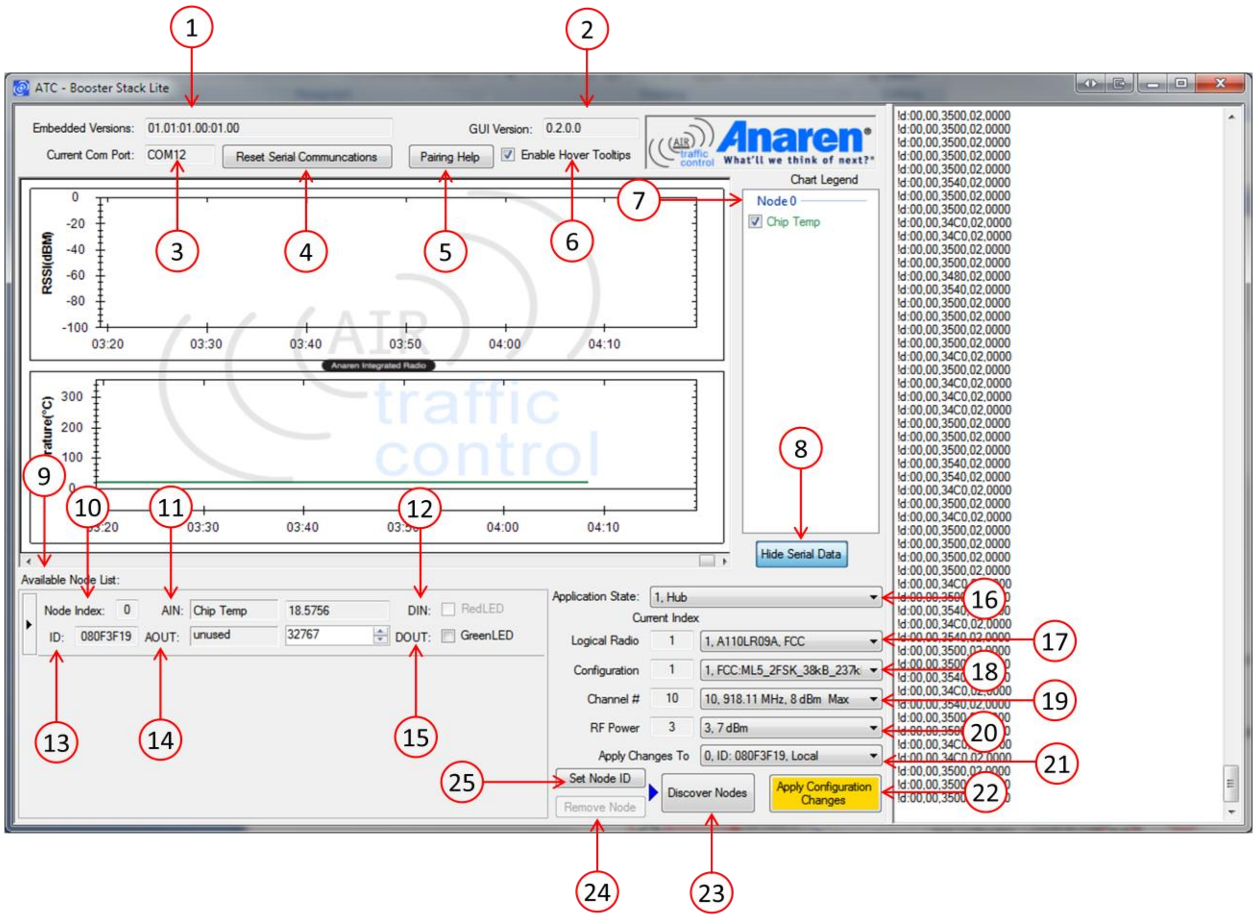


Figure 14 - GUI Populated with all the tabs

No.	Tab Title	Tab Description
1	Embedded Versions	Current Versions of Embedded code in the sequence: Application Level, Protocol Level, Hardware Level.
2	GUI Version	Current Version of GUI.
3	Current Com Port	Current COM Port to which the GUI is connected.
4	Reset Serial Communication	Reset the current COM Port Connection.
5	Pairing Help	Opens a popup window to display the text for Pairing Help.
6	Enable hover Options	Enables the Hover options for the tabs.
7	Chart Legend	Choose which information to show on the chart. Check boxes appear under each Node in the current network.
8	Hide Serial Data	Hide/Unhide the serial data window.
9	Available Node List	List of all the nodes in the network.
10	Node Index	Node Index (Index 0 is the node itself, Index > 0 is for the nodes connected to it in the network.)
11	AIN	Analog Data In (Temperature Data).
12	DIN	Digital Data In (Red LED).
13	ID	Node ID (Generated randomly at each node).
14	AOUT	Analog Data Out (Unused).
15	DOUT	Digital Data Out (Green LED).
16	App State	Application State Drop Down Menu (Displays current Application State by default).
17	Logical Radio	Logical Radio List Drop Down Menu (Displays current Radio by default). Shows current Radio Index in the tab to its left.
18	Configuration	Configuration List Drop Down Menu (Displays current Configuration Information by default). Shows current Configuration Index in the tab to its left.
19	Channel#	Channel List (Specific to each Configuration) Drop Down Menu (Displays current Channel Information by default). Shows current Channel Index in the tab to its left.
20	RF Power	RF Power List (Specific to each Configuration and corresponding Channel) Drop Down Menu (Displays current RF Power Settings Information by default). Shows current RF Power Index in the tab to its left.
21	Apply Changes To	Node ID list Drop Down Menu to decide, which node to apply the Configuration changes to. Shows the Local (its own) and Remote (Node connected to it) node IDs.
22	Apply Configuration Changes	Apply all the Configuration changes chosen from the Drop Down Menus.
23	Discover Nodes	Discover Nodes for Pairing.
24	Remove Node	Select the node from the 'Available Nodes List' and remove it from the Network.
25	Set Node ID	Opens up a pop up box to set the desired Node ID to the local node.

Application State: At initial startup, the default application state of the module is set as ‘Hub’. The tab ‘App State’ opens the drop down menu for application states (Sensor and Hub). The chosen entry from the drop down menu is set as the current application state and is displayed in the tab. Subsequent startups will continue to use the state selected.

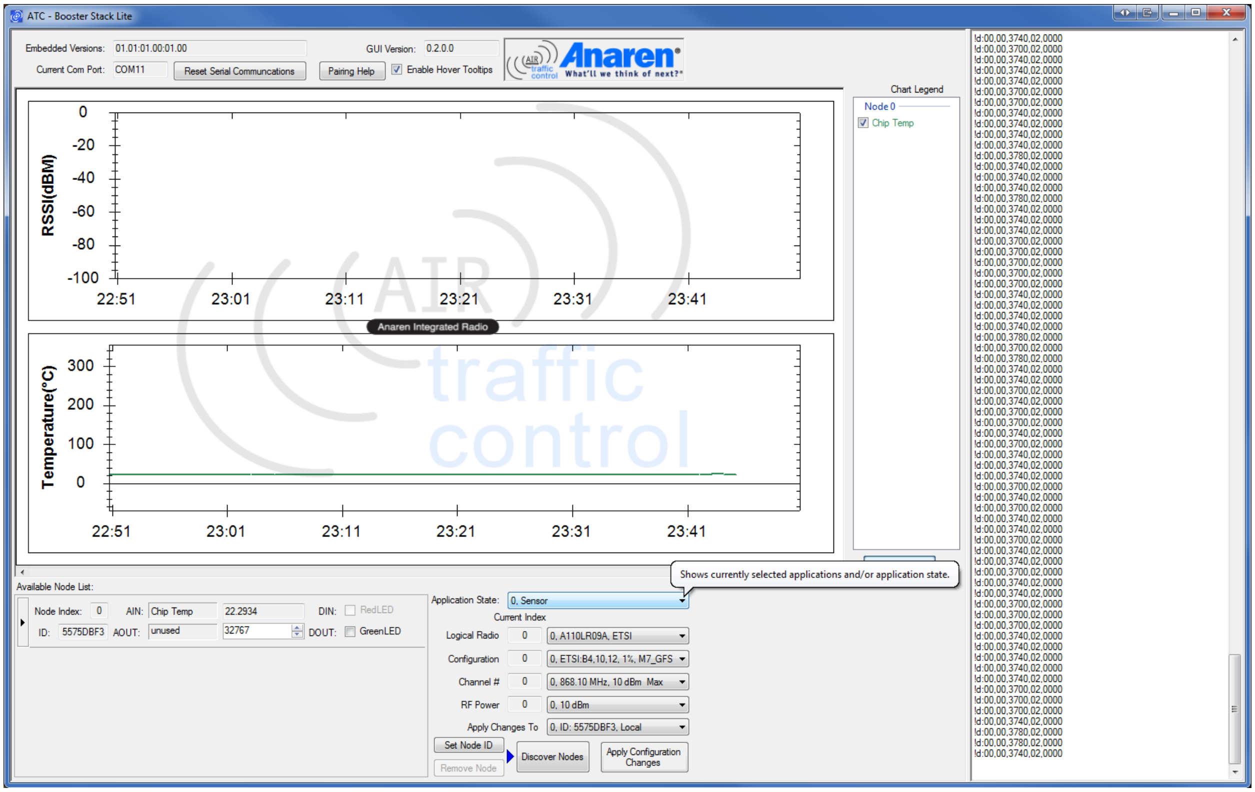


Figure 15 - Application State

Set Node ID: This tab opens a popup window and allows setting the Node ID for the local node. It takes 8 Hex characters as the Node ID.

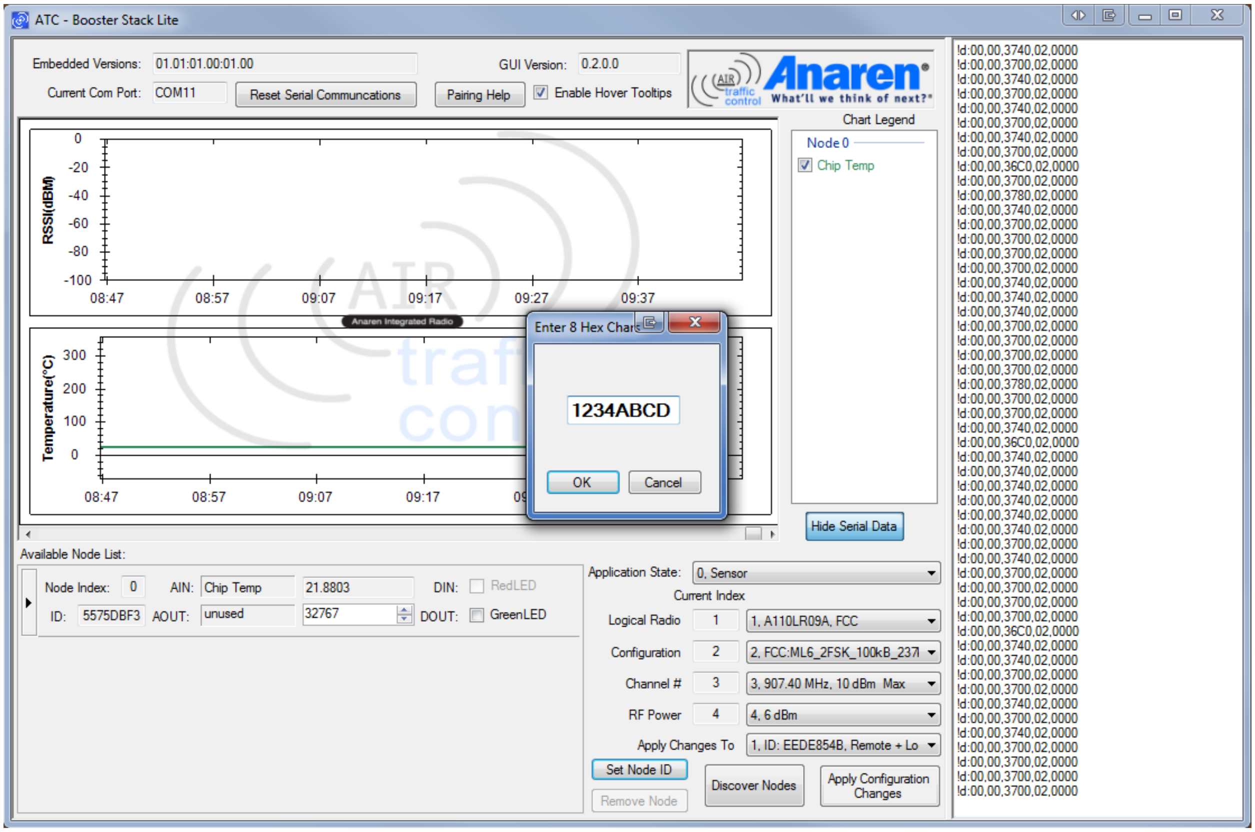


Figure 16 - Set Node ID (Local only)

Pairing: The following section describes the pairing of two modules connected to the GUI.
Hub: The Hub is placed into pairing mode by pressing the tab ‘Discover Nodes’.

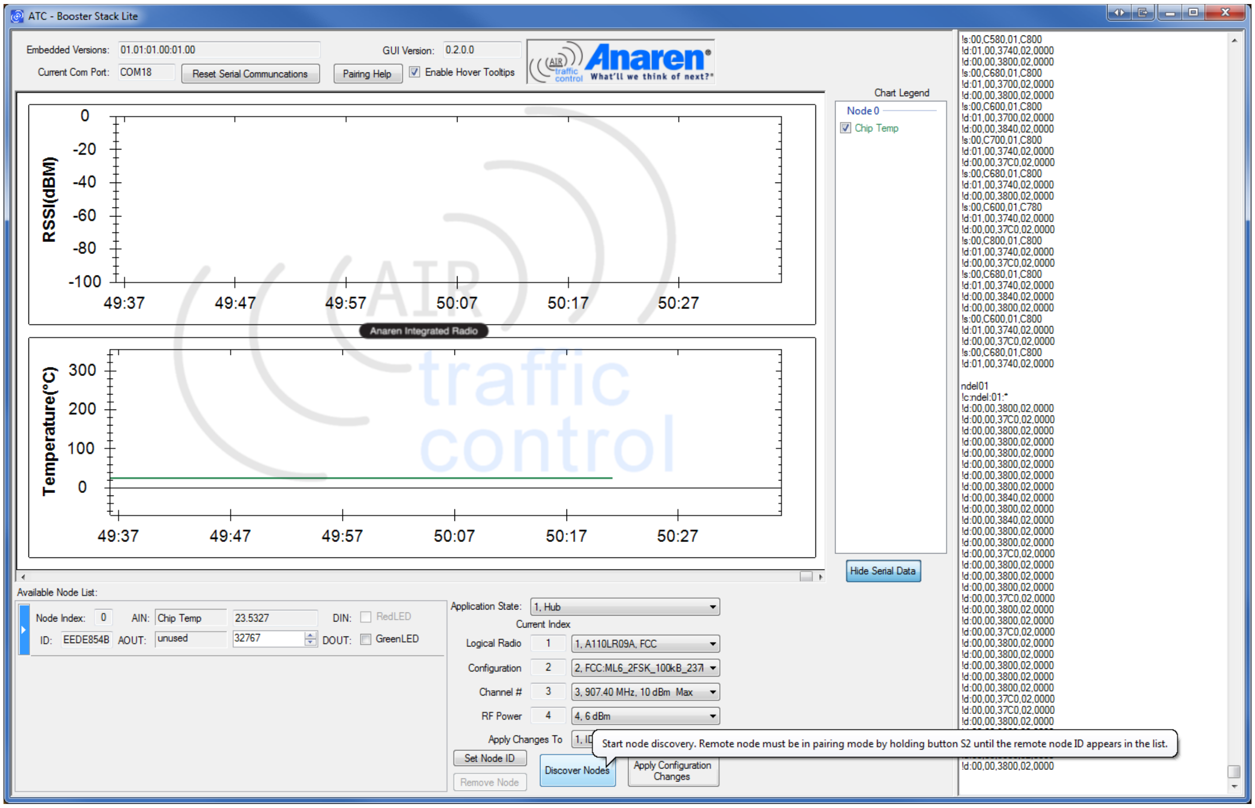


Figure 17 - Discover Nodes

This sets the pairing mask on the Hub and it starts scanning for the available sensors that are in pairing mode themselves. If the Hub establishes a connection with the Sensor, it updates its network (node list as shown by (a)). The paired Sensor (b) shows up in the ‘Available Node List’.

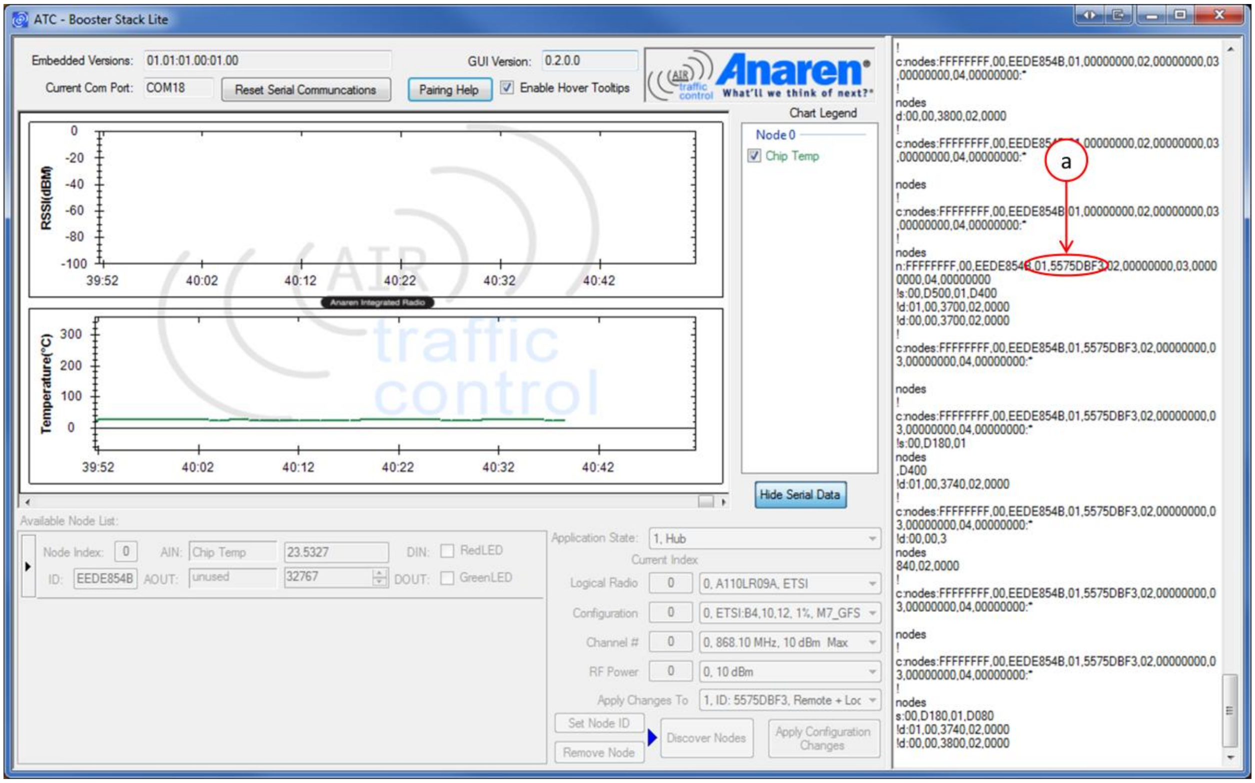
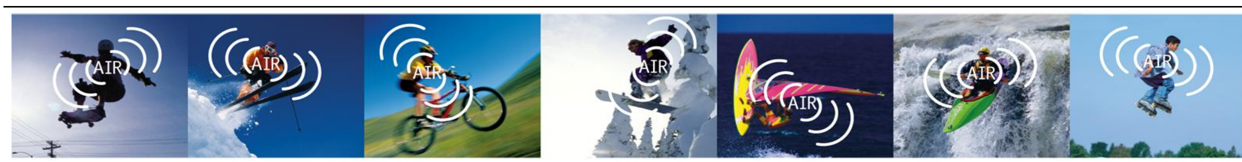


Figure 18 - Hub in Pairing



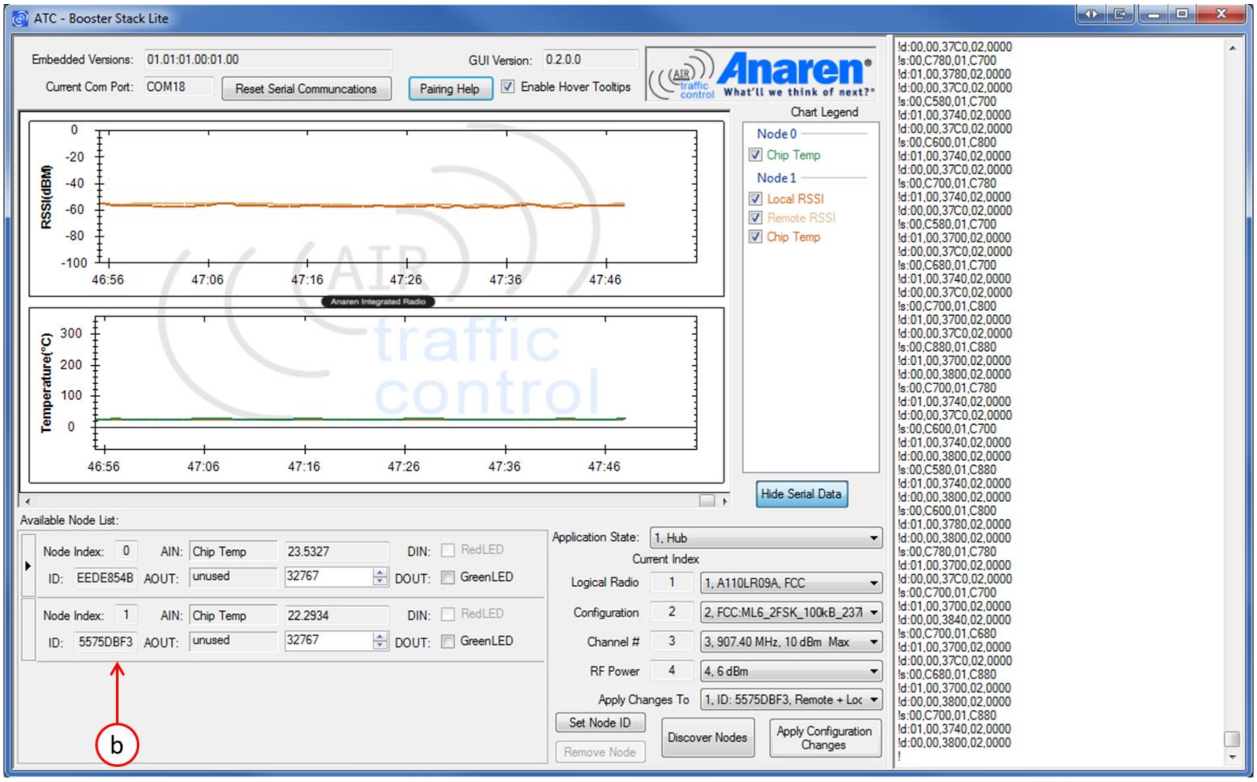


Figure 19 - Hub paired up with the Sensor

Sensor: The Sensor is placed into pairing mode by 2 methods:

1) By pressing the button switch on the module: This sets the pairing mask on the Sensor and if a Hub happens to be in pairing mode at the same time, the Sensor gets paired up with the node. It updates its network (node list as shown by (a)).

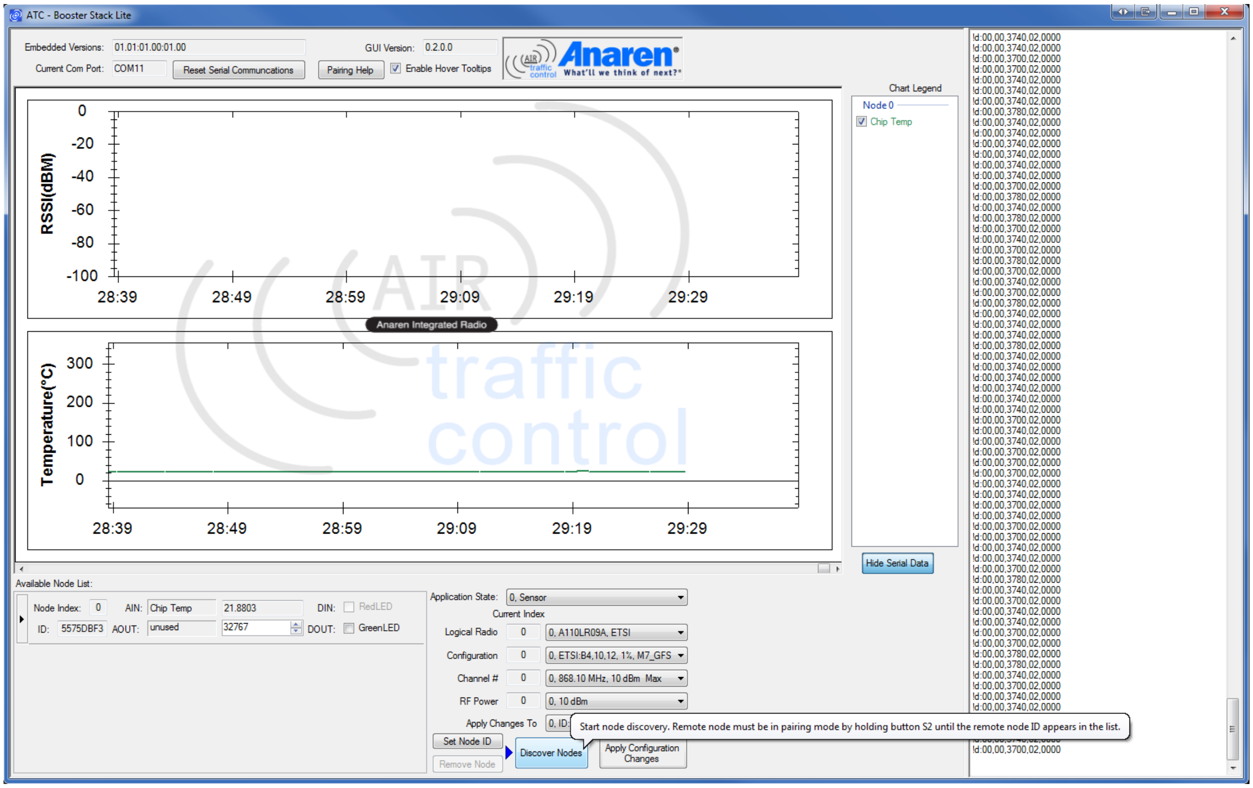


Figure 20 - Sensor in Pairing

Once the Sensor is paired up with the Hub, it displays the information of the paired Hub (b) in the 'Available Node List', by pressing the tab 'Discover Nodes'.

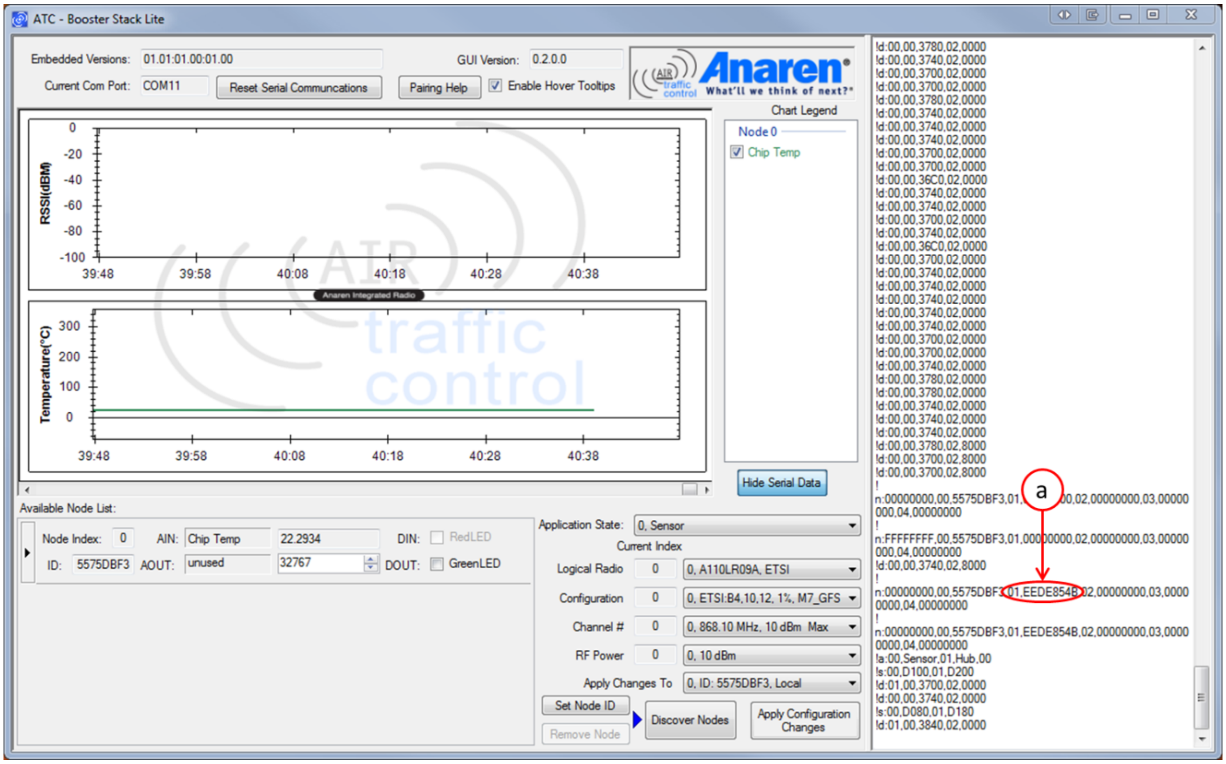


Figure 21 - Sensor paired up with Hub using switch button on the module

2) By pressing the tab ‘Discover Nodes’: This sets the pairing mask on the Sensor and if a Hub happens to be in pairing mode at the same time, the Sensor gets paired up with the node. It updates its network (node list as shown by (a)), and displays the information of the paired Hub (b) in the ‘Available Node List’.

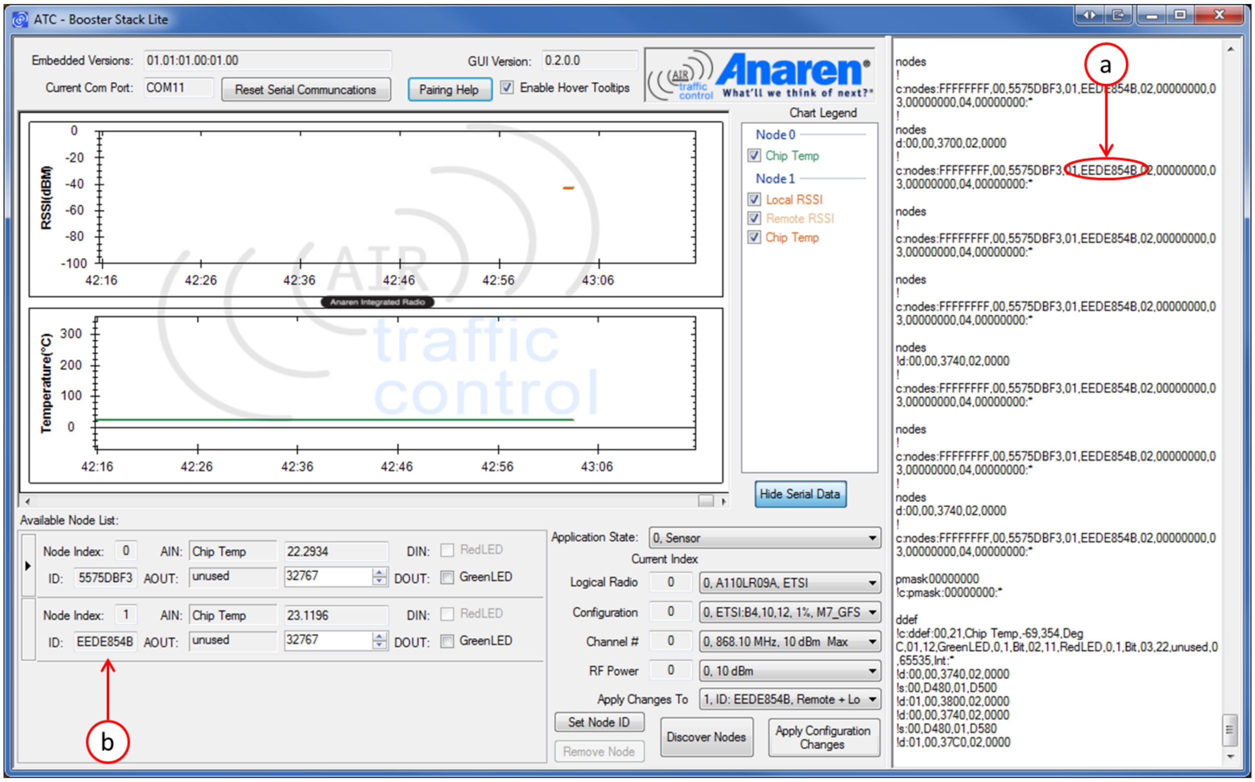
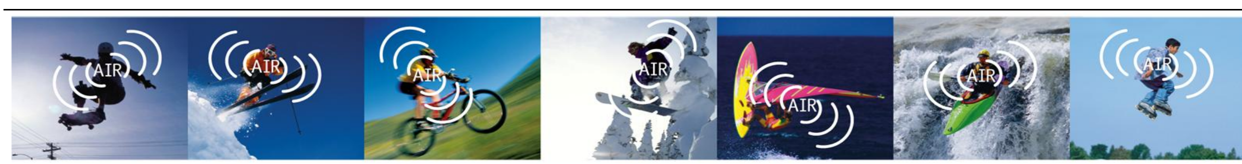


Figure 22 - Sensor paired up with Hub using the ‘Discover Nodes’ tab



Available Node List: This tab (9) shows the information about the node itself and all the nodes it is connected to.

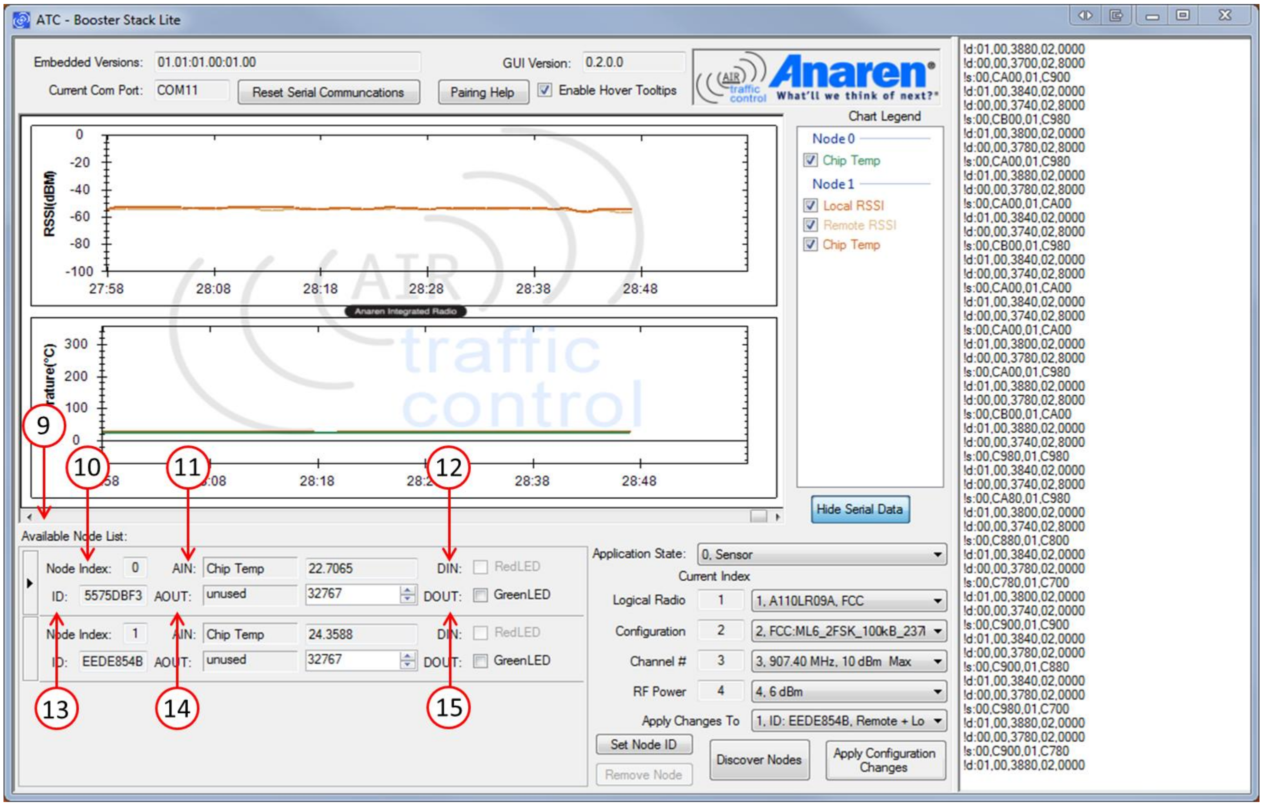


Figure 23 - Available Node List

DOUT: The checkbox allows the GUI to send a command to the module to toggle its Green LED. This is to figure out which module is connected to the GUI. This is to be used only for the local module. The following picture shows the Green LED tab checked, and remote control data (DIN) on Red LED tab.

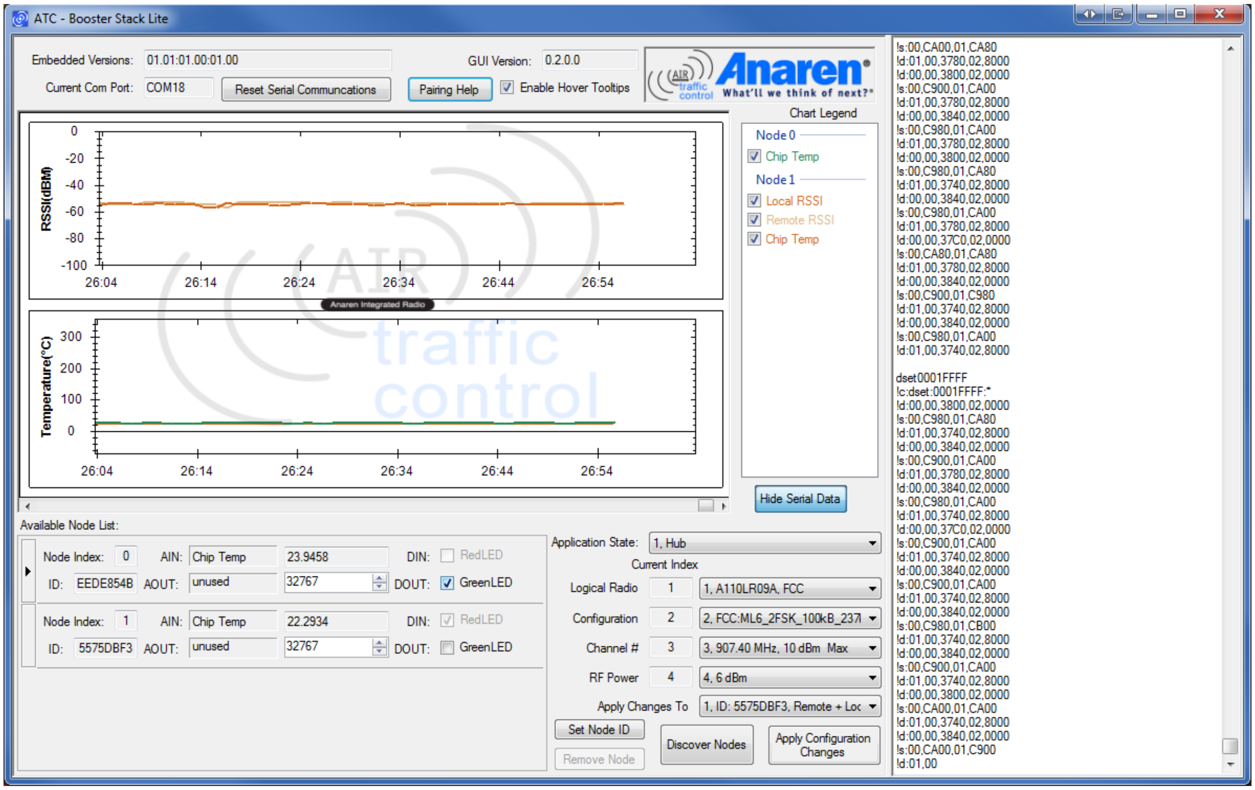


Figure 24 - DIN and DOUT tab Operation

DIN: When the button switch is pressed on the module, it reflects the data on to the GUI and checks the box next to Red LED. This is the remote control application where it also sends the data to the connected remote module and checks the box corresponding to its DIN tab in the node's information.

The following picture shows that the button switch has been pressed on the Sensor so the box is checked against DIN tab. The picture above reflects this on the Sensor's data in the Hub's node list.

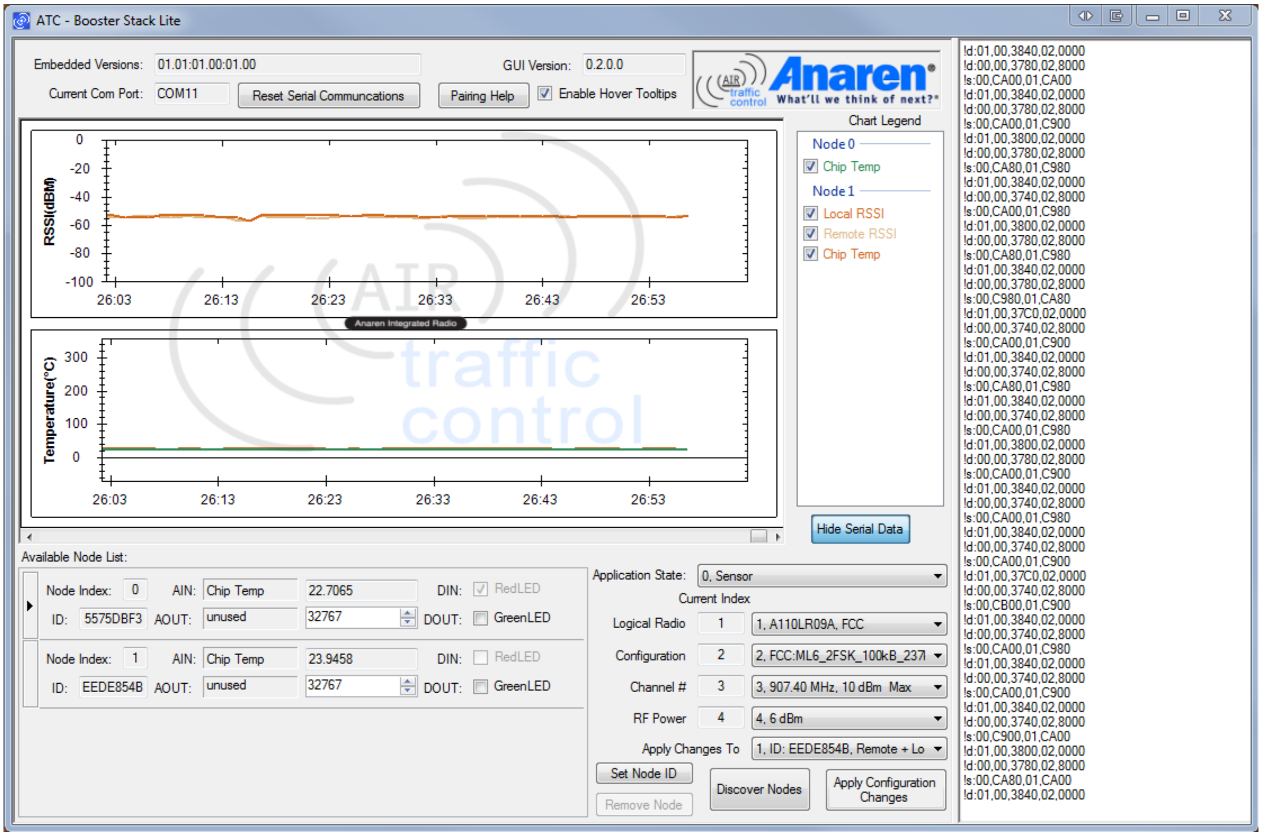


Figure 25 - DIN tab Operation

Remove Node: This tab removes the paired up node from the 'Available Node List'. This necessarily loses the pairing and removes the node from the network. (This has to be done on both the GUIs to take effect.)

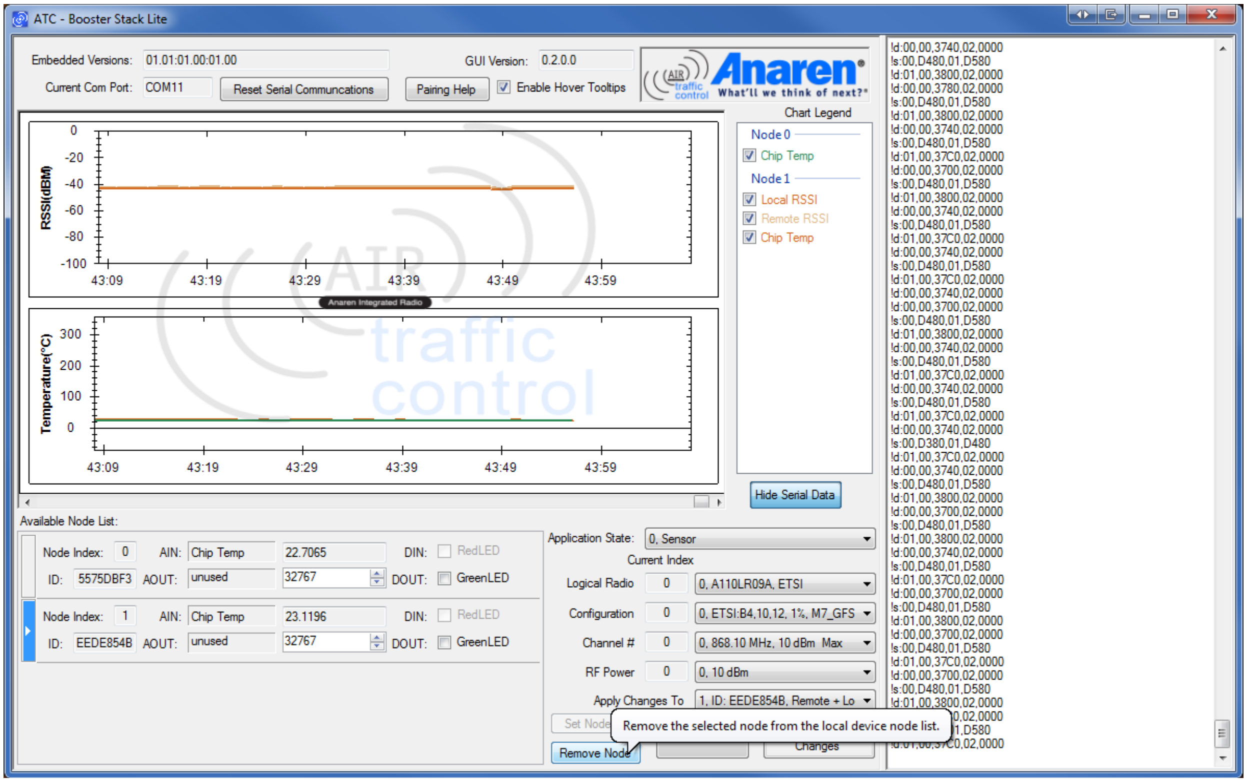


Figure 26 - Remove Node

Configuration Settings:

Logical Radio: This tab displays the information about current Logical Radio. The drop down menu shows all the Logical Radios available. The Current Index tab represents the index of the current radio based on the list of radios available with the module.

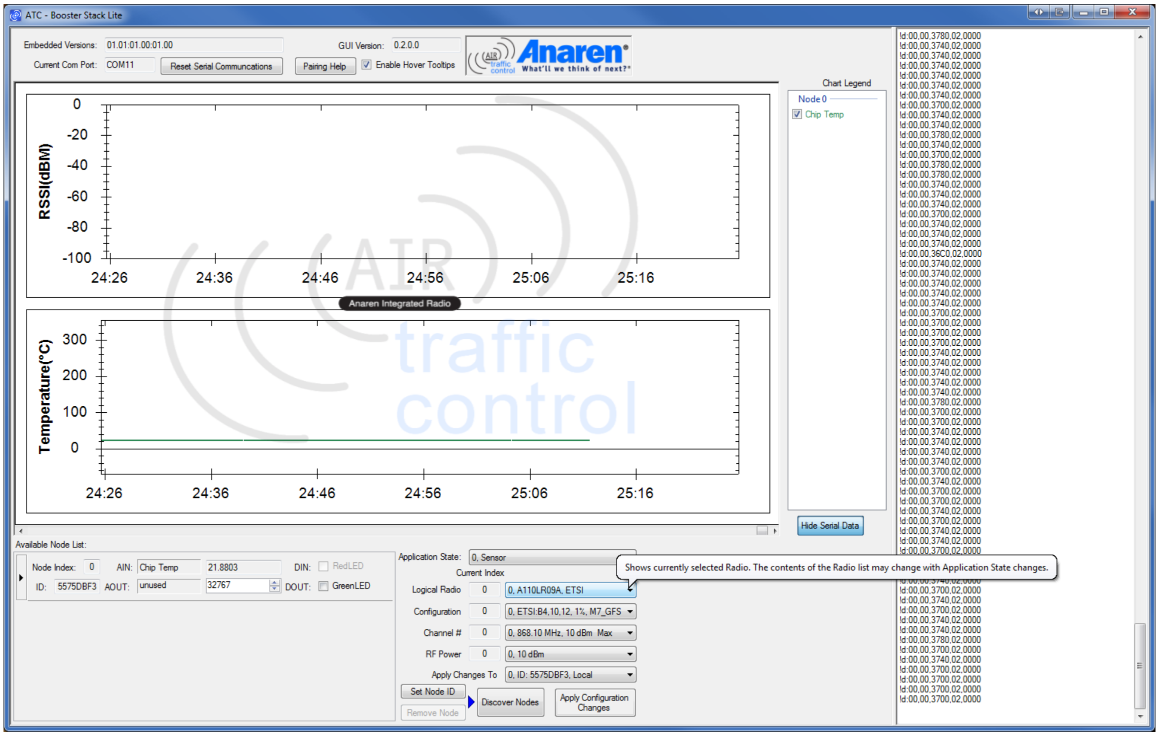


Figure 27 - Logical Radio

Configuration: This tab displays the information about the current Configuration of the current Logical Radio. The drop down menu shows all the configurations available corresponding to the Logical Radio. The Current Index tab represents the index of the current Configuration based on the list of configurations available with the Logical Radio.

Note: When operating as a star network (more than one sensor node), a data rate greater than 1.2kBuad should be used. This is to avoid unreliable performance that can occur from a potential data timing issue.

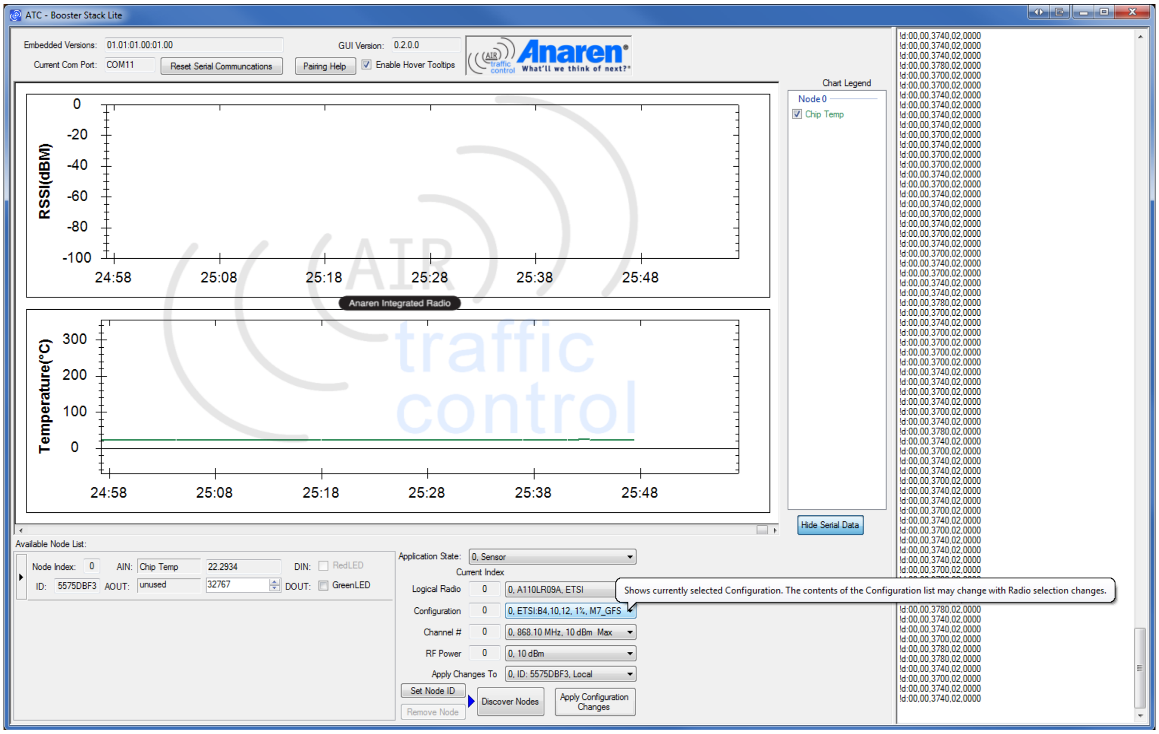


Figure 28 - Configuration

Channel#: This tab shows the information about the current Channel and the maximum power level tolerable with it. The drop down menu displays all the channels available with the corresponding to the current Configuration, corresponding to the current Logical Radio. The Current Index tab shows the index of the current Channel# based on the ascending values of the frequency.

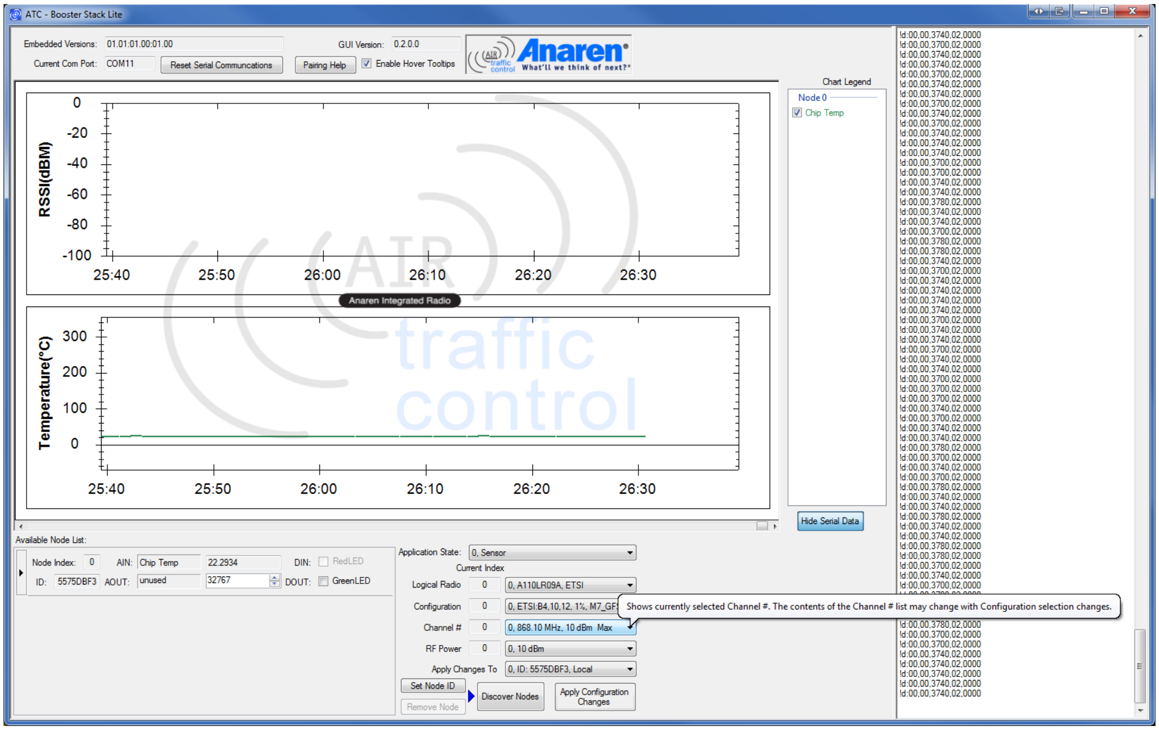


Figure 29 - Channel#

RF Power: This tab shows the information about the current RF Power settings. The drop down menu displays all the power settings tolerable to the current Channel# corresponding to the current Configuration, corresponding to the current Logical Radio.

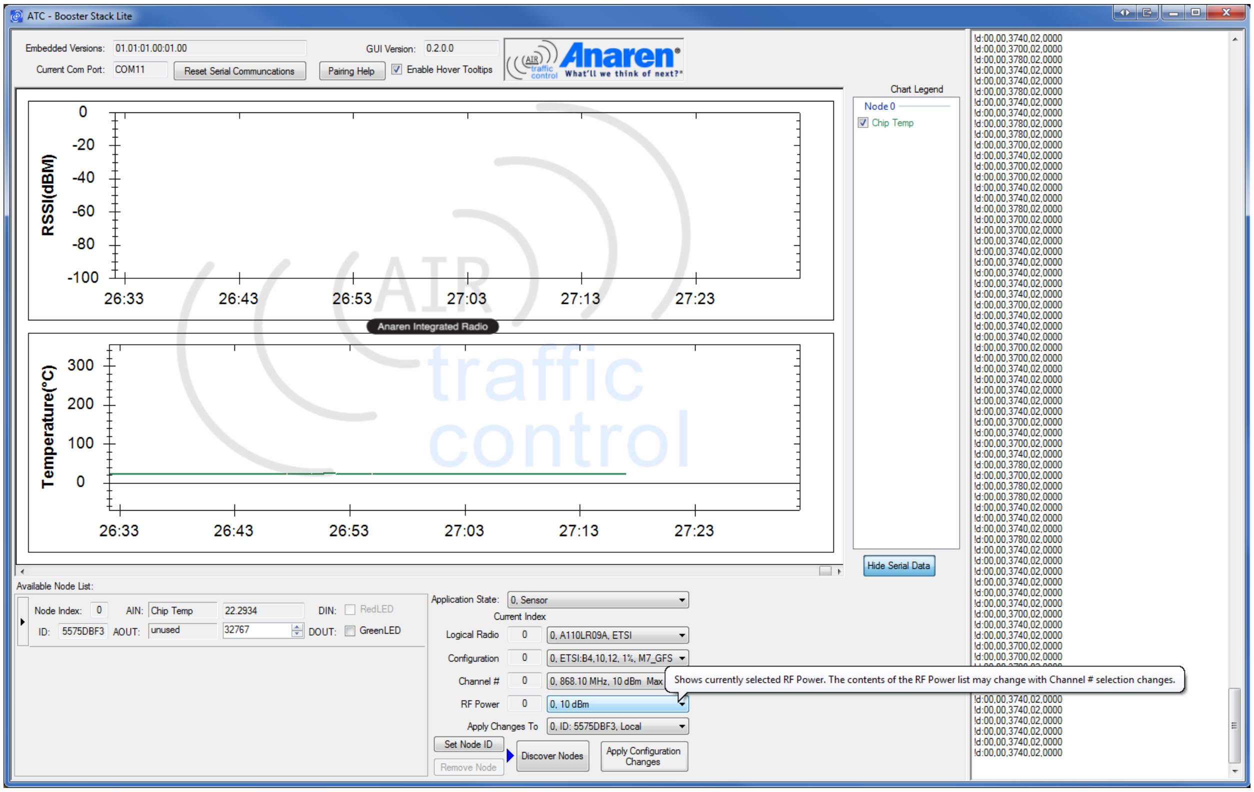


Figure 30 - RF Power

Apply Changes To: This tab shows the chosen Node ID to which to apply the chosen configuration settings of Logical Radio, Configuration, Channel#, RF Power. The drop down menu shows the local Node ID and the combination of local and remote ID, for each connected node in the network.

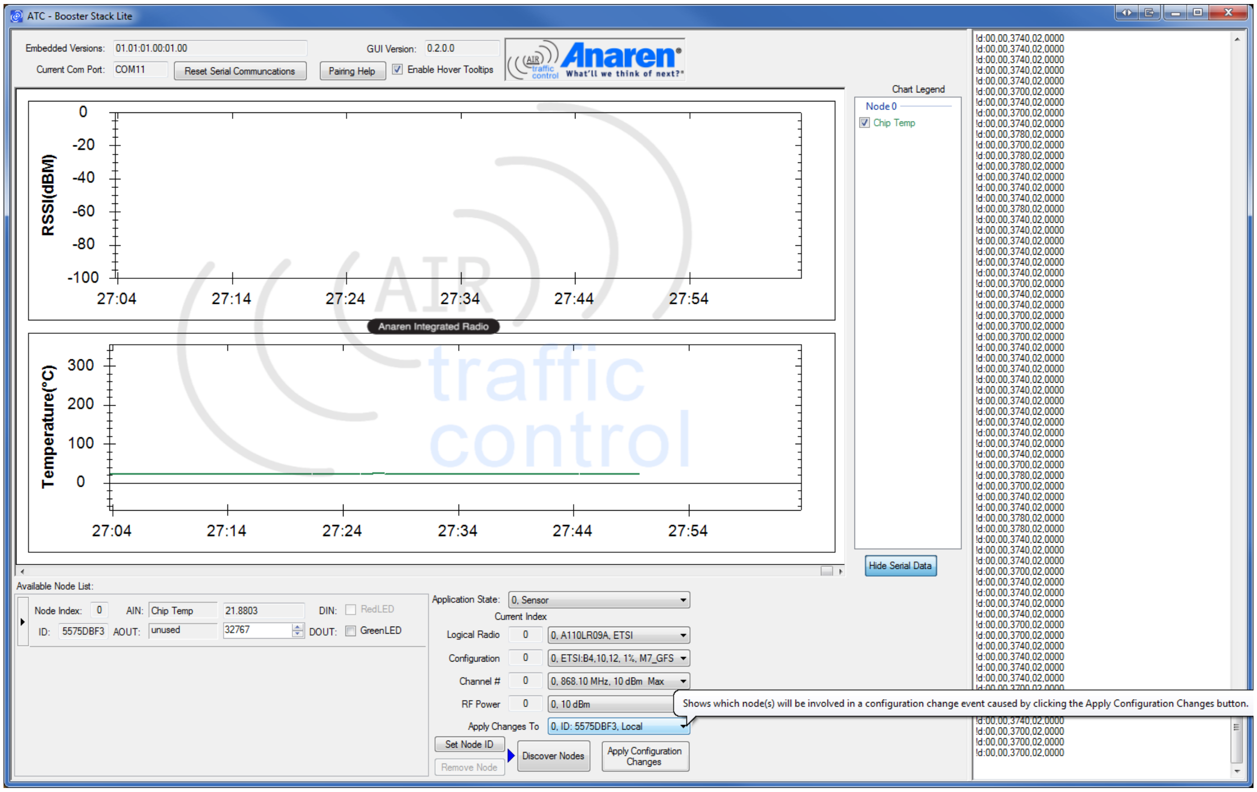


Figure 31 - Apply Changes To

Apply Configuration Changes: This tab is responsible for sending the command to change the configuration settings on the nodes based on the ‘Apply Changes To’ tab. If any of the Logical Radio, Configuration, Channel# or RF Power options are changed with their drop down menus, ‘Apply Configuration Changes’ tab turns yellow to indicate that the command is to be sent by pressing the tab.

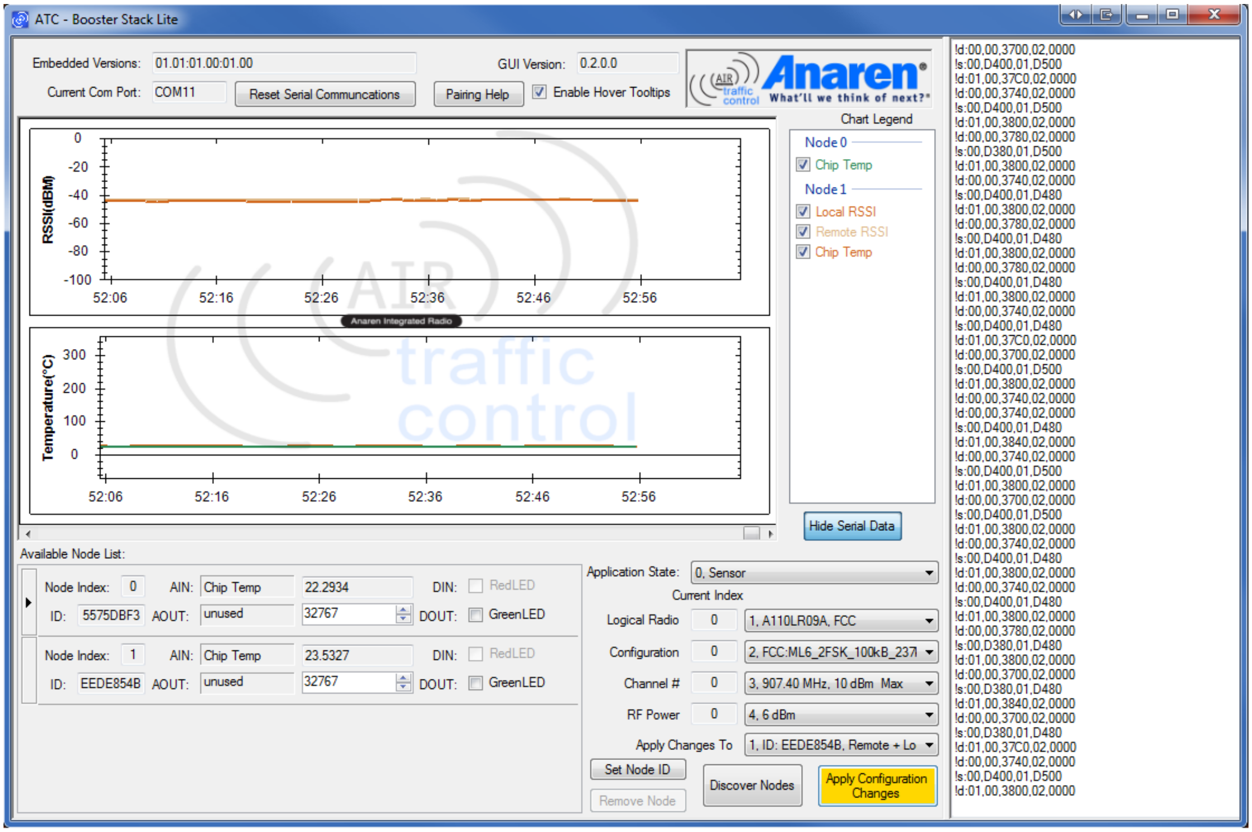


Figure 32 - Changing Configuration Settings

When the tab is pressed, it send the command to the module and based on the module's response, the tab turns Red : if the Configuration Change failed, Green: if the Configuration Change succeded. Pressing the tab again will display the current indices of the configuration settings in the serial data tab.

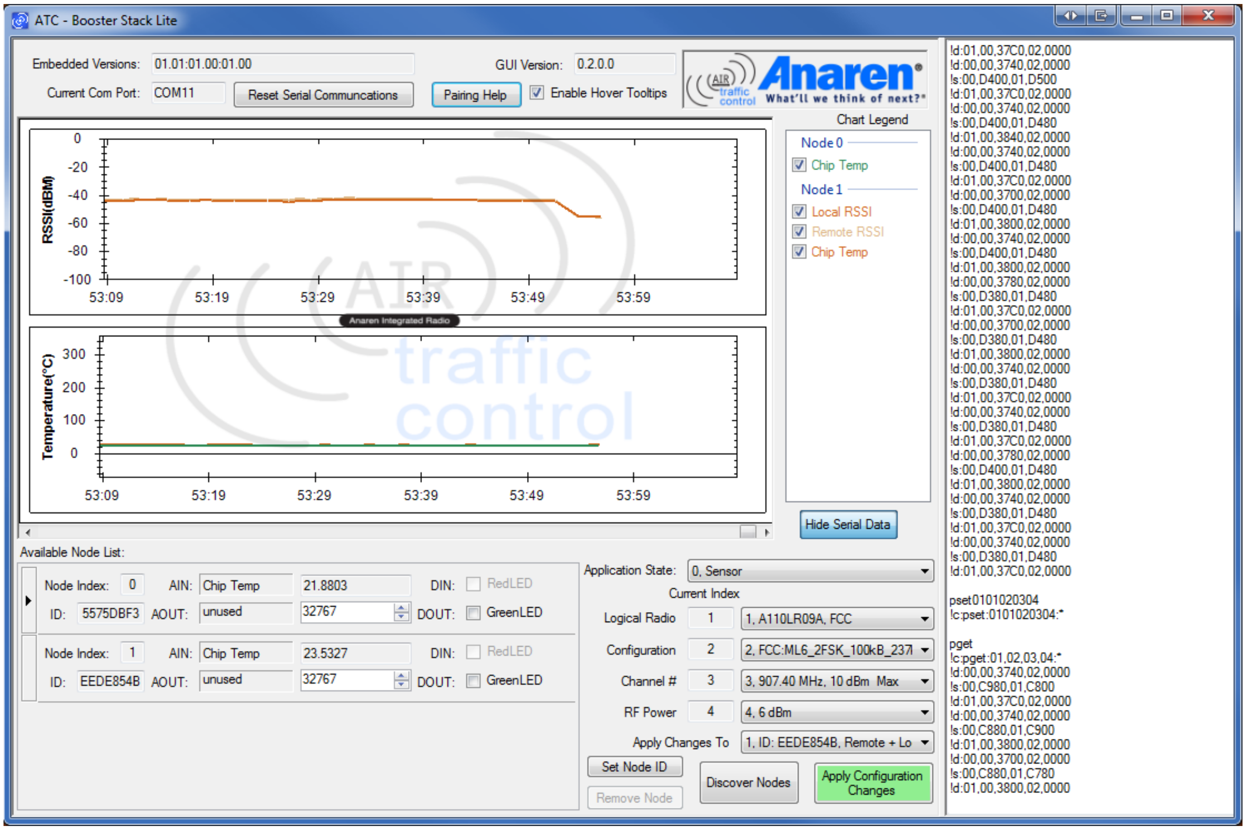


Figure 33 - Configuration Change Successful

Hide Serial Data: This tab hides the serial data window. At that time, the text on the tab changes to ‘Show Serial data’ and pressing that makes the serial data window show up again.

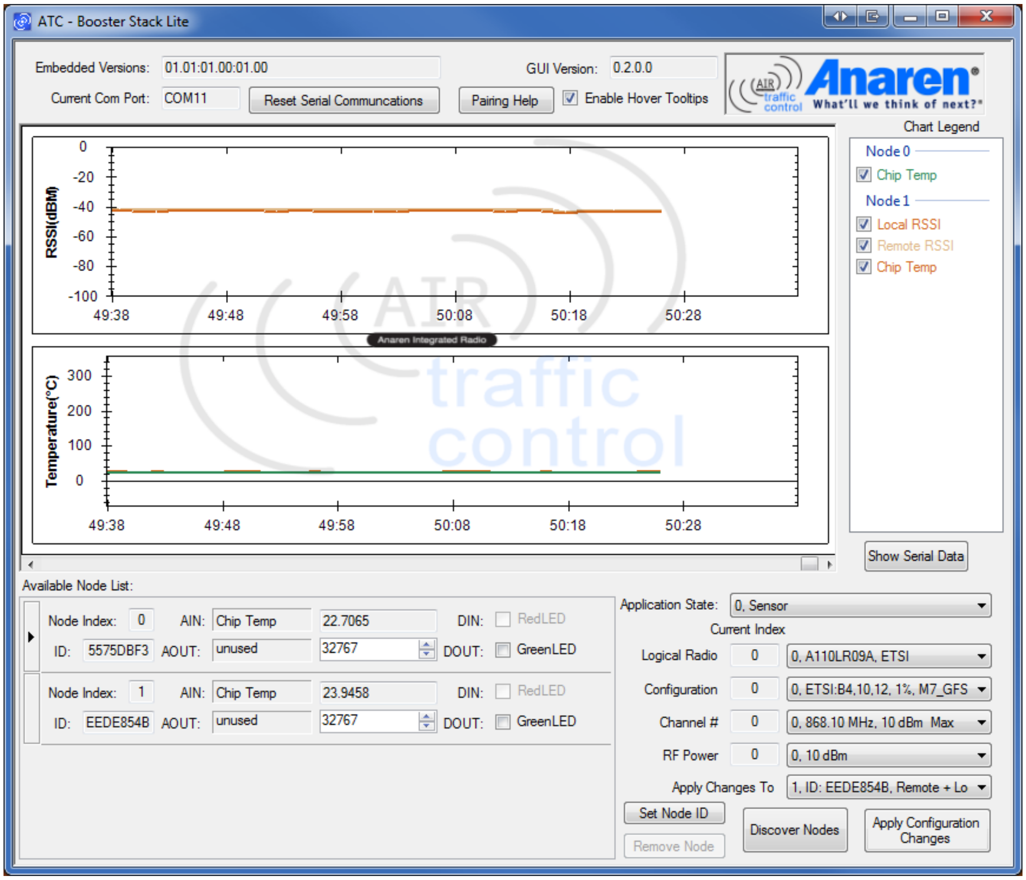


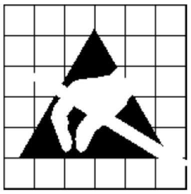
Figure 34 - Show Serial Data

4. AIR BoosterPack Hardware

4.1. Electrical Characteristics

4.1.1. Absolute Maximum Ratings

Under no circumstances shall the absolute maximum ratings given in Table 2 be violated. Stress exceeding one or more of the limiting values may cause permanent damage to the device.



Caution!
ESD sensitive device. Precaution should be used when handling the device in order to prevent permanent damage.

Parameter	Min	Max	Unit	Condition
Supply Voltage	-0.3	3.9	V	
Voltage On Any Digital Pin	-0.3	VDD + 0.3 max 3.9	V	
Input RF Level		+10	dBm	
Storage Temperature Range	-50	150	°C	
ESD		750	V	According to JEDEC STD 22, method A114, Human Body Model (HBM)
		400	V	According to JEDEC STD 22, C101C,Charged Device Model (CDM)

Table 2 - Absolute Maximum Ratings

4.1.2. Recommended Operating Conditions

Parameter	Min	Max	Unit	Condition
Operating Supply Voltage	1.8	3.6	V	
Operating Temperature	-40	85	°C	

Table 3 - Recommended Operating Conditions

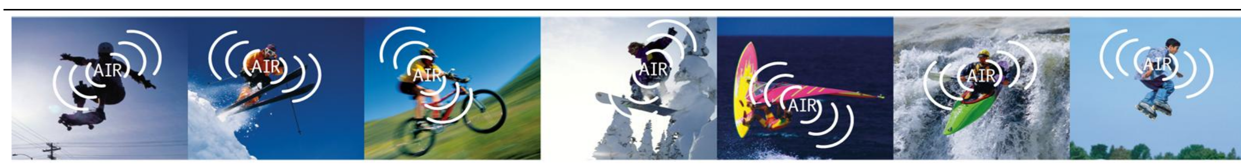
4.2. Connector Pinout

Pin	Name	I/O	Function	Description
J1-1	VDD	-	Power	Supply voltage
J1-2	P1.0	O	GDO2	Radio GDO2 output
J1-3	P1.1	-	Alt GDO0	Alternate connection to GDO0 (see Jumper Settings)
J1-4	P1.2	-	Alt CSN	Alternate connection to CSN (see Jumper Settings)
J1-5	P1.3	-	Alt GDO0	Alternate connection to GDO0 (see Jumper Settings)
J1-6	P1.4	-	Alt CSN	Alternate connection to CSN (see Jumper Settings)
J1-7	P1.5		SCLK	SPI clock
J1-8	P2.0	-	NC	
J1-9	P2.1	-	NC	
J1-10	P2.2	-	NC	

Table 4 – J1 Connector Pinout

Pin	Name	I/O	Function	Description
J2-1	P2.3	-	NC	
J2-2	P2.4	-	NC	
J2-3	P2.5	-	NC	
J2-4	P1.6	O	MISO	SPI data output (master in, slave out)
		I	Alt MOSI	Alternate connection to MOSI (see Jumper Settings)
J2-5	P1.7	I	MOSI	SPI data input (master out, slave in)
		O	Alt MISO	Alternate connection to MISO (see Jumper Settings)
J2-6	RST/ SBWTDIO	I	NC	
J2-7	TEST/ SBWTCK	I	NC	
J2-8	XOUT	I	CSn	SPI chip select (active low)
J2-9	XIN	O	GDO0	Radio GDO0 output
J2-10	GND	-	GND	Ground reference

Table 5 - J2 Connector Pinout



4.3. Jumper Settings

The AIR BoosterPack board provides the ability to remap several of the radio's I/O pins to different MSP430 pins. JP1 thru JP5 consist of one or more 0603 SMT pads which allow a zero Ohm resistor to be soldered in place to select the desired jumper position. The default signal routing is determined by a small trace between one set of pads. To change jumper positions, this trace needs to be cut and a zero Ohm resistor soldered in the new position.

Jumper	Position	Radio Signal Name	LaunchPad Signal Name	Description
JP1	1	GDO2	P1.0	The radio's GDO2 pin is connected to MSP430 P1.0 by default. This pin is shared with the red LED on the LaunchPad board. The Demo Application does not use the GDO2 functionality and instead uses the red LED. During the boot sequence, the radio's GDO2 pin is initialized to a high-impedance state and then the microcontroller pin is set to an output. To use the GDO2 signal as an interrupt/status flag from the radio, the microcontroller pin must be set as an input. The LED will no longer be under processor control but is still connected to the GDO2 pin and therefore will turn on when GDO2 is driven high by the radio. Depending on the selected output function for GDO2, the LED may or may not be visible or may appear to flicker.

Table 6 - JP1 Jumper Settings (GDO2)

Jumper	Position	Radio Signal Name	LaunchPad Signal Name	Description
JP2	1	GDO0	XIN	The radio's GDO0 pin is connected to MSP430 XIN (P2.6) by default. This pin is shared with the crystal on the LaunchPad board. If the crystal is installed, a different jumper setting must be used.
	2		P1.3	Use this setting for GDO0 if the crystal is used and UART functionality is required. This pin is shared with the pushbutton switch, so the switch should not be pressed with this setting unless GDO0 has been configured as a high-impedance output.
	3		P1.1	Use this setting for GDO0 if the crystal is used and the pushbutton switch is needed. This pin is shared with RXD, so the UART will be unavailable with this setting.

Table 7 - JP2 Jumper Settings (GDO0)

Jumper	Position	Radio Signal Name	LaunchPad Signal Name	Description
JP3	1	CSN	XOUT	The radio's CSN pin is connected to MSP430 XOUT (P2.7) by default. This pin is shared with the crystal on the LaunchPad board. If the crystal is installed, a different jumper setting must be used.
	2		P1.2	Use this setting for CSN if the crystal is used and P1.4 is used for another function. This pin is shared with TXD, so the UART will be unavailable with this setting.
	3		P1.4	Use this setting for CSN if the crystal is used and UART functionality is required.

Table 8 - JP3 Jumper Settings (CSN)

Jumper	Position	Radio Signal Name	LaunchPad Signal Name	Description
JP4	1	MOSI	P1.6	Use this setting for MSP430 microcontrollers with a Universal Serial Interface (USI) that output serial data on P1.6 (SDO) when communicating via the SPI bus.
	2		P1.7	The radio's MOSI pin is connected to MSP430 P1.7 by default. Use this setting for MSP430 microcontrollers with a Universal Serial Communication Interface (USCI) that output serial data on P1.7 (MOSI) when communicating via the SPI bus.

Table 9 - JP4 Jumper Settings (MOSI)

Jumper	Position	Radio Signal Name	LaunchPad Signal Name	Description
JP5	1	MISO	P1.6	The radio's MOSI pin is connected to MSP430 P1.6 by default. Use this setting for MSP430 microcontrollers with a Universal Serial Communication Interface (USCI) that input serial data on P1.6 (MISO) when communicating via the SPI bus.
	2		P1.7	Use this setting for MSP430 microcontrollers with a Universal Serial Interface (USI) that input serial data on P1.7 (SDI) when communicating via the SPI bus.

Table 10 - JP5 Jumper Settings (MISO)

4.4. Schematics

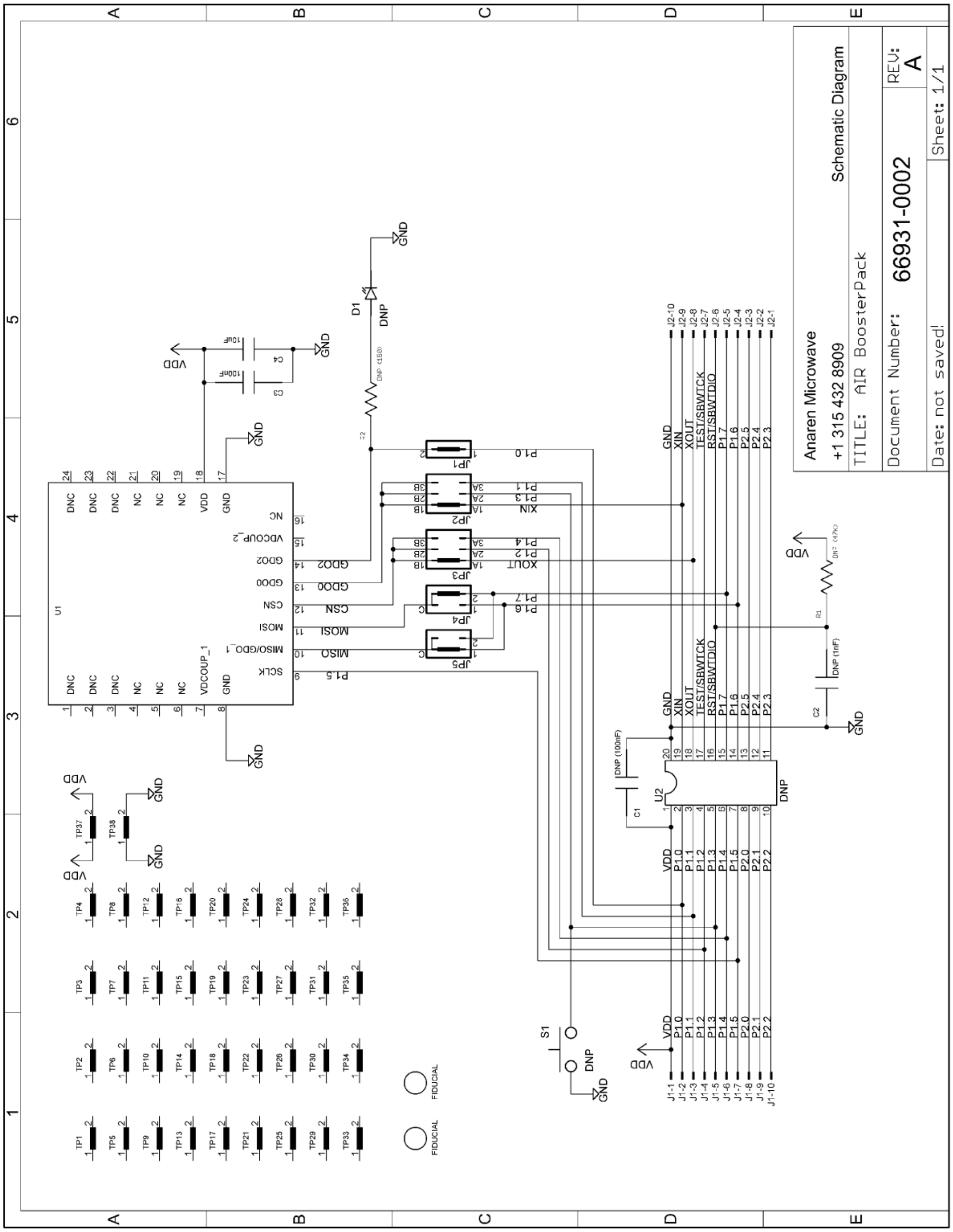
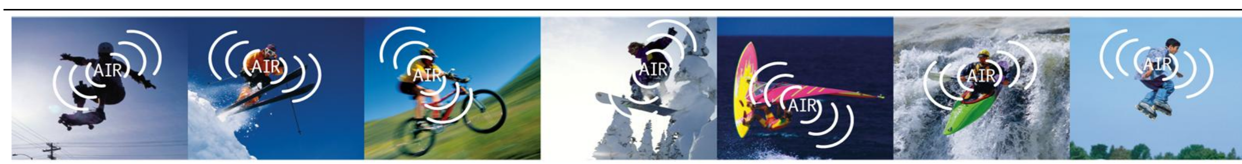


Figure 35 - Schematics



4.5. PCB Layout

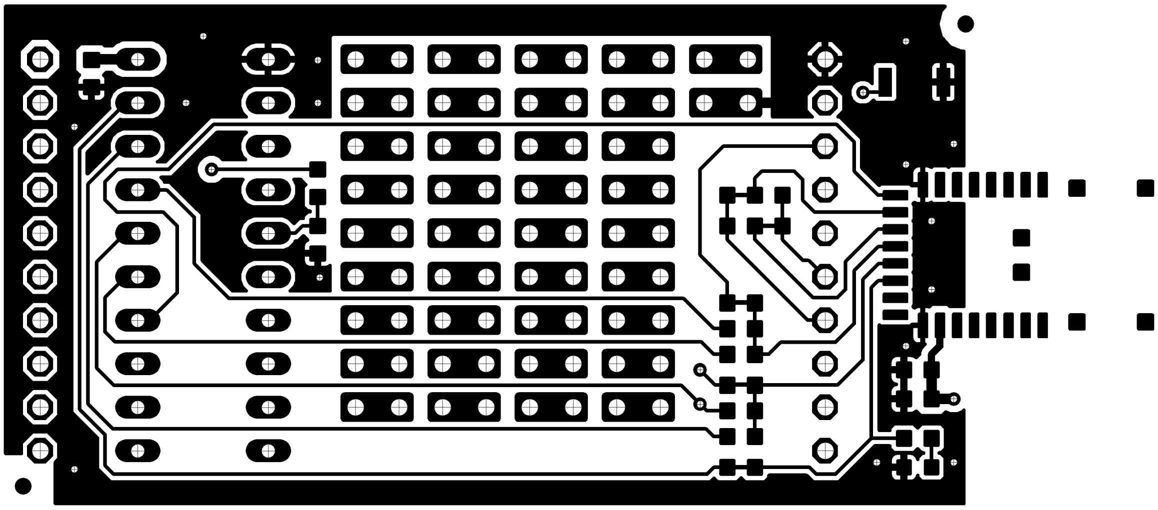


Figure 36 - PCB Layout Top Layer

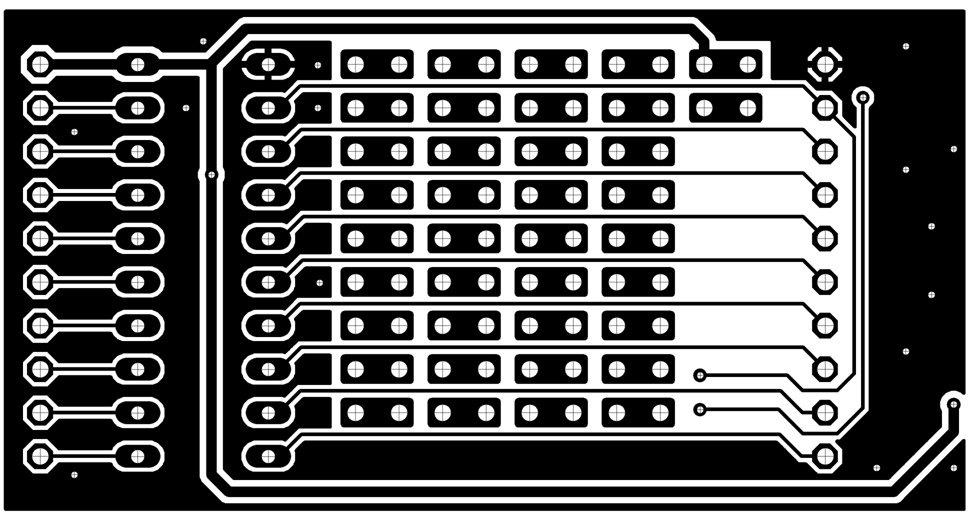


Figure 37 - PCB Layout Bottom Layer

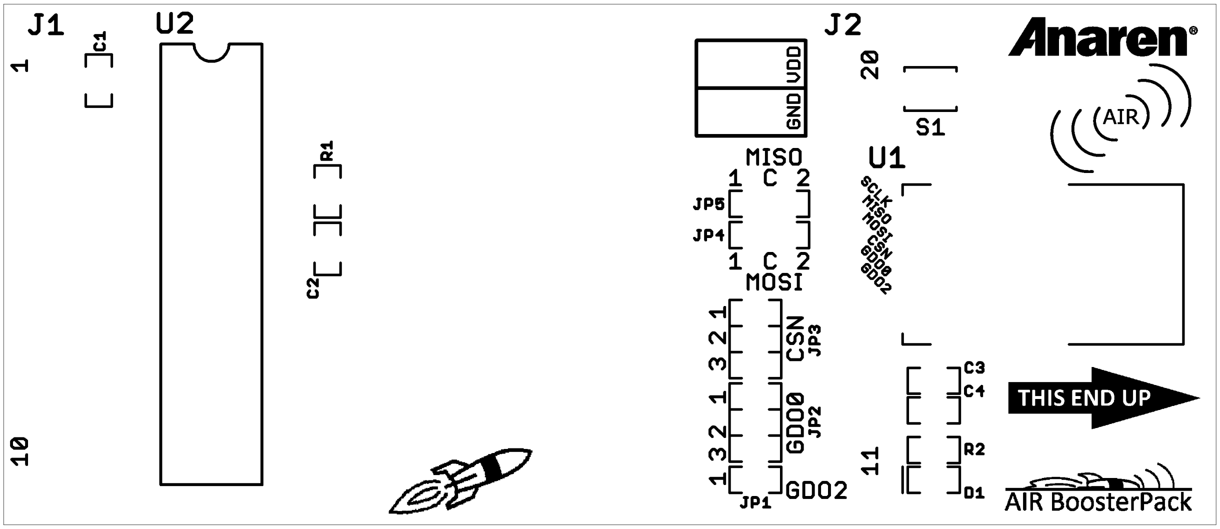


Figure 38 - PCB Layout Top Silkscreen

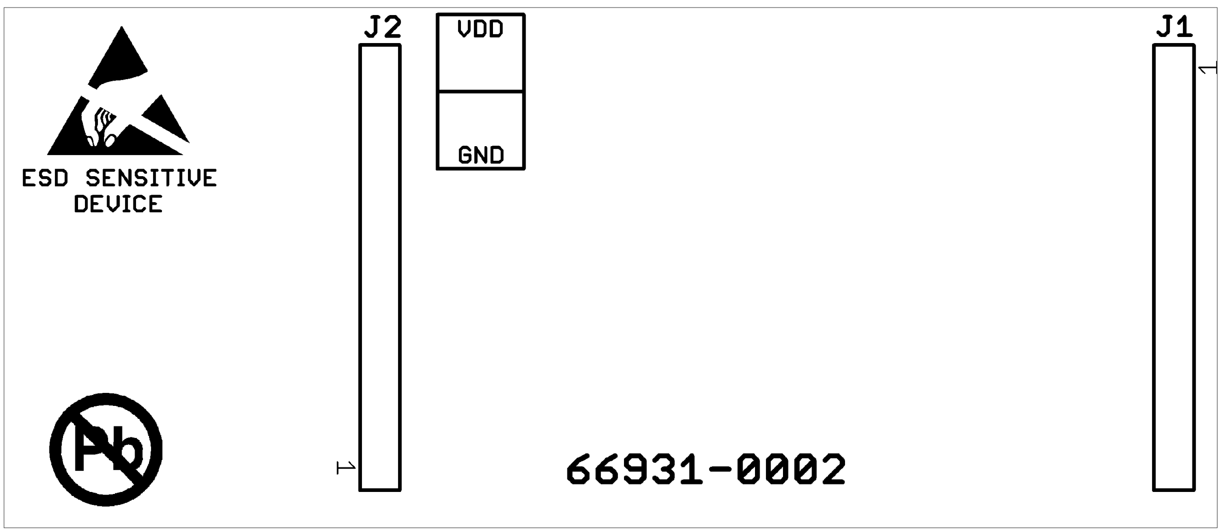


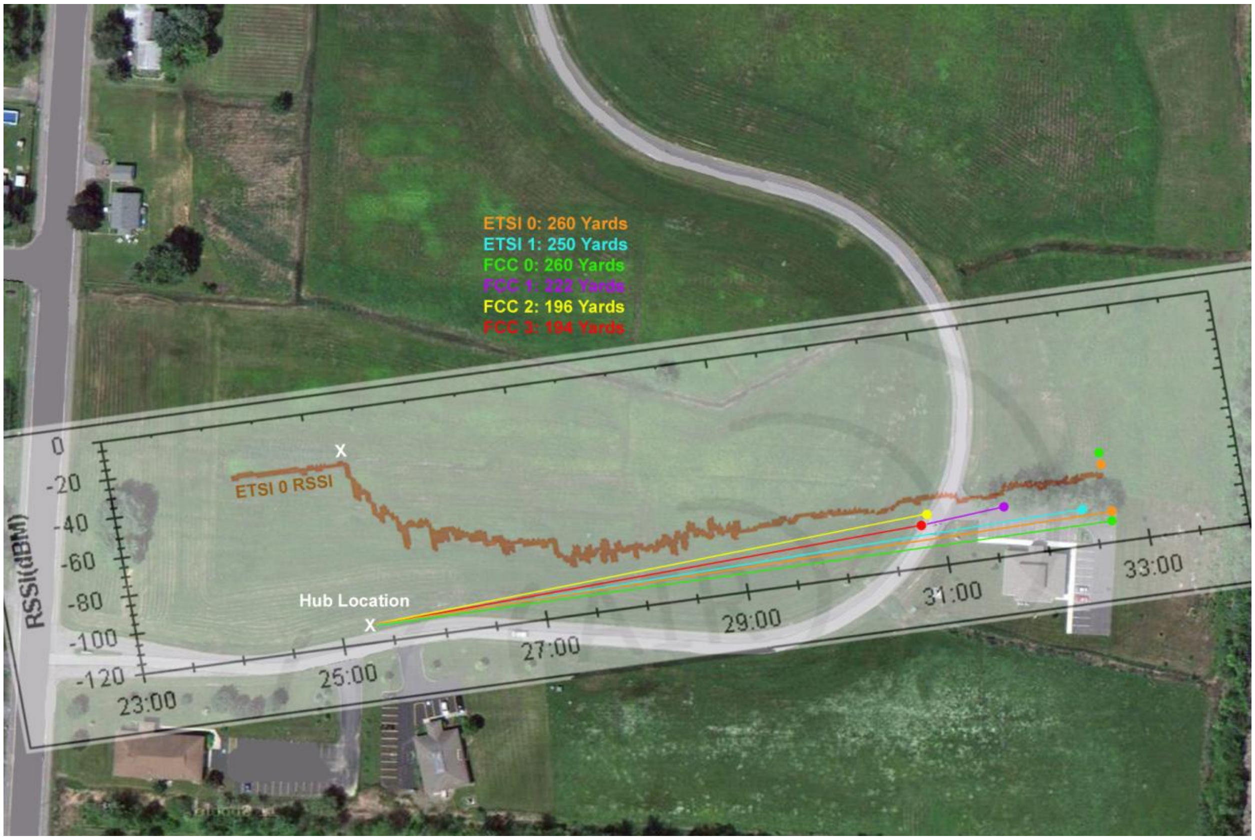
Figure 39 - PCB Layout Bottom Silkscreen

4.6. Bill of Materials (BOM)

Item	Ref Des	Qty	Description	Comment
1	C1,C3	1/1 NP	CAP CER .1UF 16V 10% X7R 0603 SMD	C1 not populated
2	C2	1 NP	CAP CER 1000PF 50V 10% X7R 0603 SMD	Not populated
3	C4	1	CAP CER 10UF 6.3V X5R 0603 SMD	
4	D1	1 NP	LED 0603 RED SMD	Not populated OSRAM LS L29K-G1J2-1-Z (or equivalent)
5	J1,J2	2	CONN SOCKET 10POS 2.54MM SINGLE ROW	
6	JP1	1 NP	RES 0.0 OHM 1/10W 0603 SMD	Not populated
7	JP2,JP3	2 NP	RES 0.0 OHM 1/10W 0603 SMD	Not populated
8	JP4,JP5	2 NP	RES 0.0 OHM 1/10W 0603 SMD	Not populated
9	R1	1 NP	RES 47K OHM 1/10W 5% 0603 SMD	Not populated
10	R2	1 NP	RES 150 OHM 1/10W 5% 0603 SMD	Not populated
11	S1	1 NP	SWITCH TACT SPST NO SMD	Not populated Omron B3U-1000P
12	U1	1	RADIO MMODULE LR09A SMD	
13	U2	1 NP	SOCKET DIP20	Not populated Aries 20-3518-10 (or equivalent)

Table 11 - Bill of Material

5. Range Test


ETSI 0: 260 Yards
ETSI 1: 250 Yards
FCC 0: 260 Yards
FCC 1: 222 Yards
FCC 2: 196 Yards
FCC 3: 194 Yards
ETSI 0 RSSI
X
Hub Location
X
RSSI(dBM)
0
-20
-40
-60
-80
-100
-120
23:00
25:00
27:00
29:00
31:00
33:00

HISTORY

Date	Author	Change Note No./Notes
10/16/11		Initial Draft
12/14/11		Added multi-purpose pushbutton controls. Pairing method changed. Improved green LED remote control function. Added section for radio register settings.
1/18/12		Added notes on radio operation and LaunchPad COM port.

THIS PAGE LEFT INTENTIONALLY BLANK

Thank you for learning more about the
Anaren Integrated Radio (AIR) module line.
If you have additional questions,
need samples, or would like a quote –
please do not hesitate to email the AIR team
at AIR@anaren.com or contact any of these
authorized distributors of the AIR product line.

ARROW ELECTRONICS, INC.

Worldwide

AVNET electronics marketing

North America

MOUSER ELECTRONICS a tti company

Worldwide

Anaren Microwave, Inc.
6635 Kirkville Road
East Syracuse, NY 13057
Tel: +1 315 432 8909
+1 800 411 6596
Fax: +1 315 432 8970

Anaren Microwave (Europe), Inc.
12 Somerset House, Suite 16 & 17
Hussar Court, Waterlooville
Hampshire, England P07-7SG
Tel: +44 2392 232392
Fax: +44 2392 251369

Anaren Communication Suzhou Co. Ltd.
No. 5 Chun Hui Road
Wei Ting, Suzhou Industrial Park
Suzhou 215122, PR China
Tel: +86 512 6287 6400
Fax: +86 512 6274 9283