

User Guide

AC4490

Version 4.6

REVISION HISTORY

Version	Date	Notes	Contributors	Approver
1.0		Initial Release		Chris Downey
1.1	10/2012	Major changes and revisions throughout document		Chris Downey
2.0	04/5/2013	Major changes and revisions; updated format and data	FCC	Chris Downey
3.0	--	--		Chris Downey
4.0	--	--		Chris Downey
4.1	10 Dec 2013	Updated download instructions for Laird Configuration Utility in section How do I configure Sync to Channel?		Chris Downey
4.2	11 Dec 2013	Separated into two separate docs (Hardware Integration Guide and User Guide). Added a Related Documents section.		Sue White
4.3	6 Feb 2013	Fixed return data for EEPROM Byte Write.		Chris Downey
4.4	15 Apr 2015	Updated contact information and links to new website.		Sue White
4.5	18 May 2017	Removed references to EOL part numbers		Jonathan Kaye
4.6	21 Oct 2022	Updates for AC4490-1000M Revision 2.0 hardware:- Added RF TX power Table 4 Converted to newest template.	Dave Drogowski Raj Khatri	Jonathan Kaye

CONTENTS

1	AC4490 RF Transceiver	4
1.1	Overview	4
2	Features	4
2.1	Networking and Security	4
2.2	Easy to Use	4
3	Theory of Operation	5
3.1	RF Architecture	5
3.2	Modes of Operation	5
4	AC4490 Configuration	8
4.1	AT Commands	8
4.2	Command Descriptions	10
4.3	API Control	22
5	EEPROM Parameters	23
6	Appendix I: API	30
7	Appendix II: Sync-to-Channel	34
7.1	Sync to Channel - What is it and do I need to use it?	35
7.2	How do I configure Sync to Channel?	36
7.3	I've configured my radios, what's next?	42
8	Related Documents and Files	45
9	Additional Assistance	45

1 AC4490 RF TRANSCEIVER

The compact AC4490 900 MHz transceiver can replace miles of cable in harsh industrial environments. Using field-proven frequency hopping spread spectrum (FHSS) technology which needs no additional FCC licensing in the Americas, OEMs can easily make existing systems wireless with little or no RF expertise.

1.1 Overview

The AC4490 is a cost effective, high performance, frequency hopping spread spectrum transceiver designed for integration into OEM systems operating under FCC part 15.247 regulations for the 900 MHz ISM band.

AC4490 transceivers provide an asynchronous TTL level serial interface for OEM Host communications, which include both system and configuration data. The Host supplies system data for transmission to other Host(s). Configuration data is stored in the on-board EEPROM. All frequency hopping, synchronization, and RF system data transmission/reception is performed by the transceiver.

To boost data integrity and security, the AC4490 uses Laird's field-proven FHSS technology featuring optional Data-Encryption Standards (DES). Fully transparent, these transceivers operate seamlessly in serial cable replacement applications.

AC4490 transceivers can operate in Point-to-Point, Point-to-Multipoint, Client-Server, or Peer-to-Peer architecture. One transceiver is configured as a server with one or many client-configured transceivers synchronized to it. To establish synchronization between transceivers, the server emits a beacon; upon detecting a beacon an RF link is established and a GPIO is toggled to signify to the host that the link is active.

This document contains information about the hardware and software interface between a Laird AC4490 transceiver and an OEM Host. Information includes the theory of operation, specifications, interface definition, configuration information and mechanical drawings. The OEM is responsible for ensuring the final product meets all appropriate regulatory agency requirements listed herein before selling any product.

Additionally, this document contains a list of [Related Documents and Files](#).

Note: Unless mentioned by name, the AC4490 module is referred to as the “radio” or “transceiver”. Individual naming is used to differentiate product specific features. The host (PC / Microcontroller / Any device to which the AC4490 module is connected) will be referred to as “OEM Host”.

2 FEATURES

2.1 Networking and Security

- Generic I/O digital lines and integrated DAC/ADC functions
- Retries and Acknowledgements
- API Commands to control packet routing and acknowledgement on a packet-by-packet basis
- Frequency Hopping Spread Spectrum for security and interference rejection
- Customizable RF Channel number and system ID
- Dynamic link analysis, remote radio discovery
- Low latency and high throughput
- Hardware Protocol Status monitoring

2.2 Easy to Use

- Continuous 76.8 Kbps RF data stream
- Software selectable interface baud rates from 1200 bps to 115.2 Kbps
- Low cost, low power and small size ideal for high volume, portable and battery powered applications
- All modules are qualified for Industrial temperatures (-40° C to 85° C)
- Advanced configuration available using AT commands

3 THEORY OF OPERATION

3.1 RF Architecture

The AC4490 utilizes a server-client network architecture where all clients synchronize their hopping to the server. The server transmits a beacon during the first 20 milliseconds of every hop. The client transceivers listen for this beacon and, upon hearing it, assert their In_Range Low and synchronize hops with the server.

Each network consists of only one server. There should never be two servers on the same RF channel number in the same coverage area because the interference between the two servers will severely hinder RF communications. For those applications requiring collocated servers, Laird recommends using the Sync-to-Channel feature, further explained in [Appendix II: Sync-to-Channel](#).

3.2 Modes of Operation

The AC4490 has three different operating modes:

- [Transmit Mode](#)
- [Receive Mode](#)
- [Command Mode](#)

If the transceiver is not communicating with another radio, it is in Receive mode actively listening for a beacon from the server. If the client determines that the beacon is from a server operating on the same RF channel and system ID, it responds by asserting [In_Range](#) Low. A transceiver enters Transmit or Command mode when the OEM host sends data over the serial interface. The state of the [CMD](#) Data pin (Pin 17) or the data contents determine which of the two modes the transceiver enters.

3.2.1 Transmit Mode

All packets sent over the RF are either Addressed or Broadcast packets. Broadcast and Addressed delivery can be controlled dynamically with the API Control feature set, which can be enabled in the EEPROM configuration. To prohibit transceivers from receiving broadcast packets, Unicast Only can be enabled.

Addressed Packets When sending an addressed packet, the RF packet is sent only to the receiver specified in the destination address. To increase the odds of successful delivery, the packet uses transmit retries. Transparent to the OEM host, the sending radio sends the RF packet to the intended receiver. If the receiver receives the packet error-free it returns an RF Acknowledgement in the same 20 ms hop. If a Receive Acknowledgement is not received, the radio uses a transmit retry to resend the packet. This continues until either an acknowledgement is received or all transmit retries are used. The received packet is only sent to the OEM Host if and when it is received free of errors.

Broadcast Packets When sending a broadcast packet, the RF packet is sent to every eligible transceiver on the network. To increase the odds of successful delivery, it uses broadcast attempts. Transparent to the OEM host, the sending radio sends the RF packet to the intended receiver(s). Unlike Transmit Retries, all broadcast attempts are used, regardless of when the RF packet is actually received. RF acknowledgments are not sent or received when a broadcast packet is transmitted. If the packet is received on the first attempt, the receiver ignores the remaining broadcast attempts. The received packet is only sent to the OEM host if and when it is received free of errors.

3.2.2 Receive Mode

When a transceiver is not in Transmit or Command mode, it is in Receive mode listening for data. While in Receive mode, subsequent data of up to 80 bytes can be received every hop (20 ms).

3.2.3 Command Mode

A radio enters Command mode when data is received over the serial interface from the OEM host and contains the **AT+++** (Enter AT Command mode) command or when the state of the CMD/DATA pin is transitioned low. Once in Command mode, the radio interprets all data received as command data. Command data can be either EEPROM configuration or on-the-fly commands.

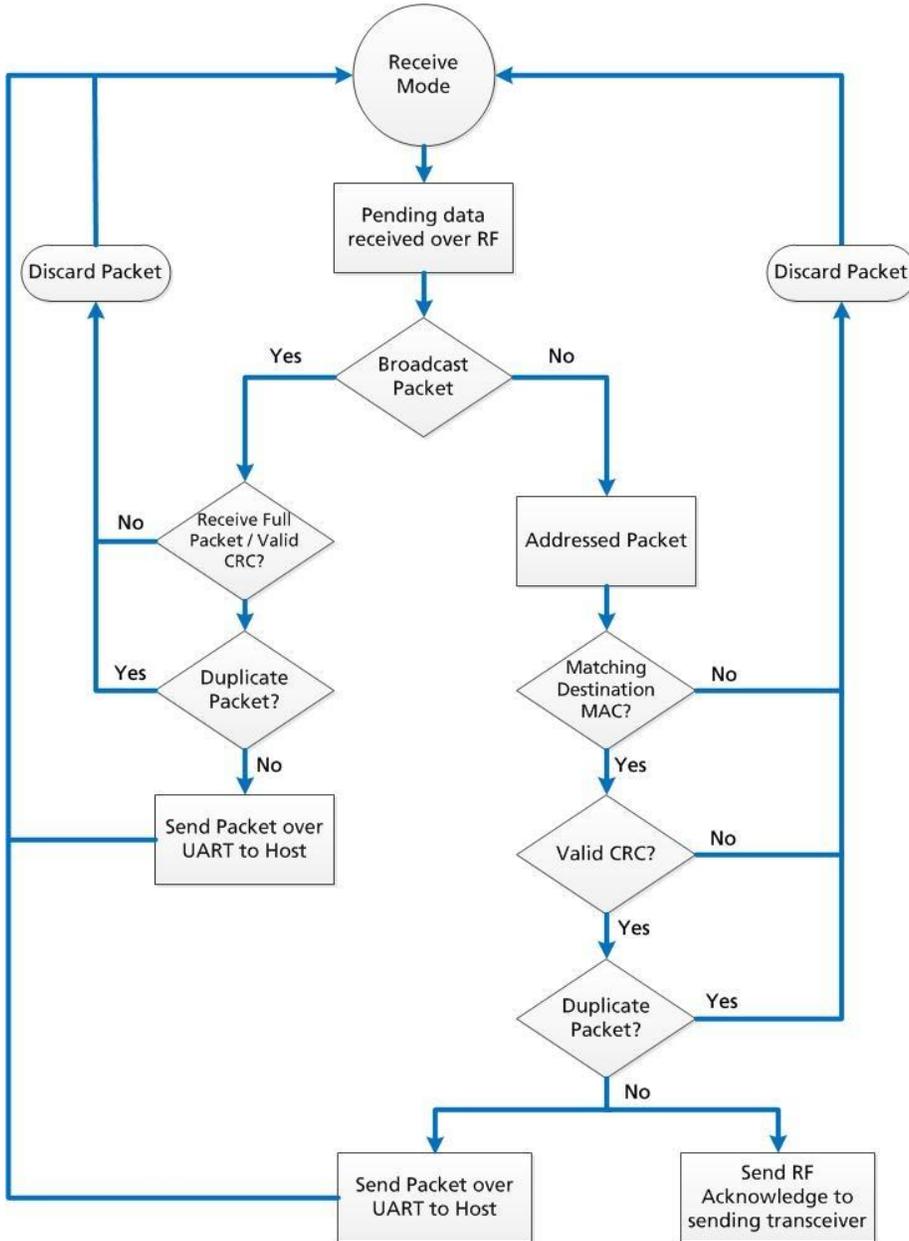


Figure 1: Pending RF Buffer Flow

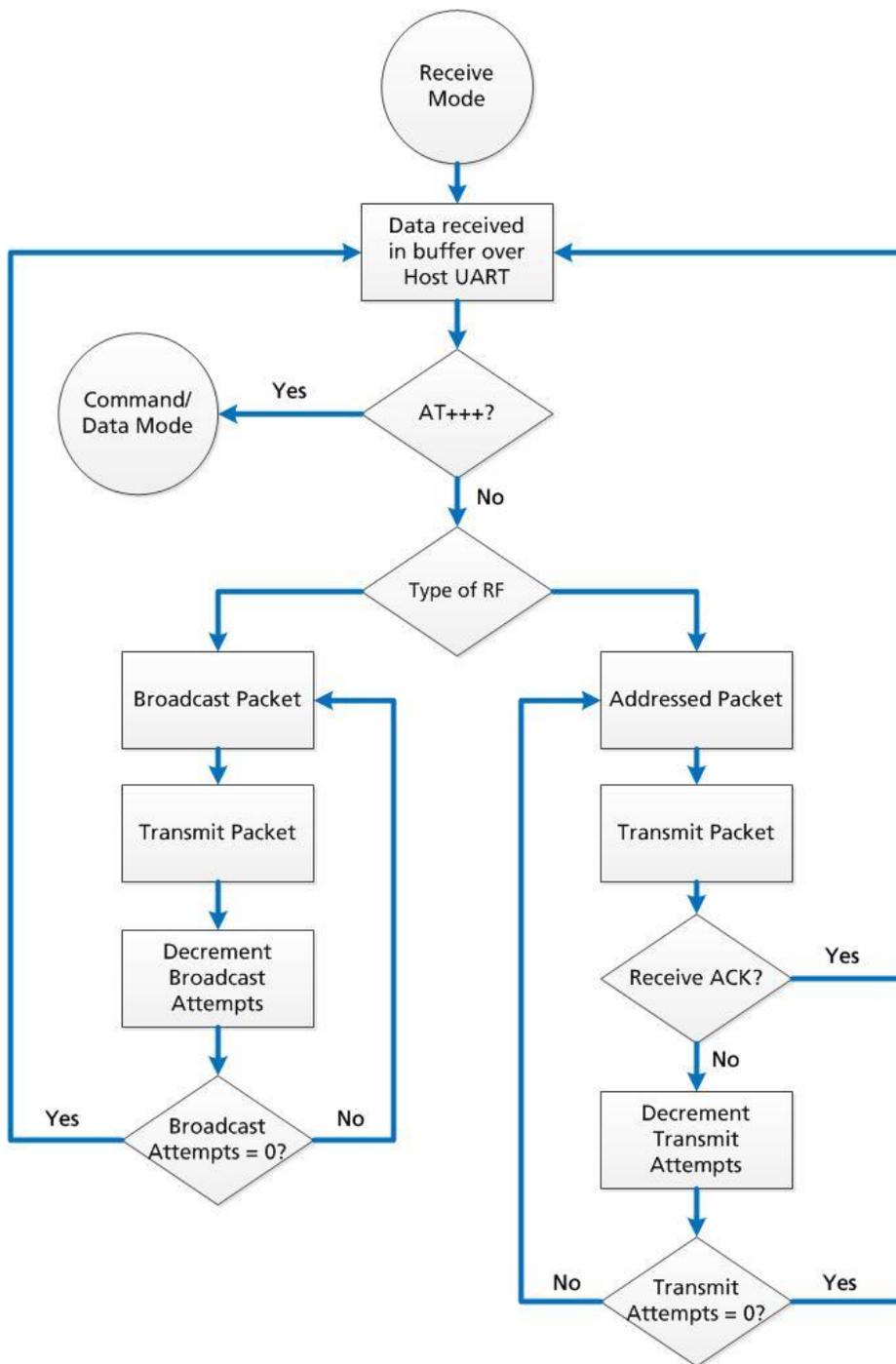


Figure 2: Pending Data in Buffer Flow

4 AC4490 CONFIGURATION

4.1 AT Commands

The AT Command mode implemented in the AC4490 creates a virtual version of the Command/Data pin. The “Enter AT Command Mode” Command asserts this virtual pin Low (to signify Command Mode) and the “Exit AT Command Mode” Command asserts this virtual pin High (to signify Data). Once this pin has been asserted Low, all On-the-Fly CC Commands documented in the manual are supported.

Note: The Command/Data RX Disable feature can be enabled in firmware versions 8.6+. When enabled in EEPROM, the radio disables the RF receiver while pin 17 (Command/Data) is Low. To ensure that the radio is not in the middle of transmitting data to the OEM Host, the host should be prepared to receive data for up to 20ms after taking pin 17 Low.

On-the-Fly Control Commands The AC4490 transceiver contains static memory that holds many of the parameters that control the transceiver operation. Using the “CC” command set allows many of these parameters to be changed during system operation. Because the memory these commands affect is static, when the transceiver is reset, these parameters will revert back to the settings stored in the EEPROM. While in CC Command mode using pin 17 (Command/Data), the RF interface of the transceiver is still active. Therefore, it can receive packets from remote transceivers while in CC Command mode and forward these to the OEM Host.

AT Command Mode While in CC Command mode using AT Commands, the RF interface of the transceiver is active, but packets sent from other transceivers will not be received. The transceiver uses Interface Timeout/RF Packet Size to determine when a CC Command is complete. Therefore, there should be no delay between each character as it is sent from the OEM Host to the transceiver or the transceiver will not recognize the command.

The link between the OEM host and the transceiver does not need to resync when exiting Command Mode. Acknowledgements will be sent while in Command Mode, but the packet will be dumped on the receiving end. However, if an RF packet is received before the Interface Timeout expires on a CC Command, the transceiver will send the packet to the OEM Host before sending the CC Command response.

When an invalid command is sent, the radio scans the command to see if it has a valid command followed by bytes not associated with the command, in which case the radio discards the invalid bytes and accepts the command. In all other cases, the radio returns the first byte of the invalid command to the user and discards the rest.

Table 1: Command Quick Reference

Command Name	Command (All Bytes in Hex)						Return (All Bytes in Hex)			
Enter Command Mode	0x41	0x54	0x2B	0x2B	0x2B	0x0D	0xCC	0x43	0x4F	0x4D
Exit Command Mode	0xCC	0x41	0x54	0x4F	0x0D	-	0xCC	0x44	0x41	0x54
Status Request	0xCC	0x00	0x00	-	-	-	0xCC	Firmware Version	0x00: Server	0x01: Client in range 0x03: Out of range
Change Channel	0xCC	0x02	New Channel	-	-	-	0xCC	New Channel	-	-
Change Server/Client	0xCC	0x03	0x00: Server 0x03: Client	-	-	-	0xCC	Firm-ware Version	0x00: Server	0x03: Client

Command Name	Command (All Bytes in Hex)						Return (All Bytes in Hex)			
Change Sync Channel	0xCC	0x05	New Sync Channel	-	-	-	0xCC	New Sync Channel	-	-
Sleep Walk Power Down	0xCC	0x06	-	-	-	-	0xCC	Channel	-	-
Sleep Walk Wake Up	0xCC	0x07	-	-	-	-	0xCC	Channel	-	-
Broadcast Packets	0xCC	0x08	0x00: Addressed 0x01: Broadcast	-	-	-	0xCC	0x00 or 0x01	-	-
Write Destination Address	0xCC	0x10	Byte 4 of Dest. MAC	Byte 5	Byte 6	-	0xCC	Byte 4 of Dest. MAC	Byte 5	Byte 6
Read Destination Address	0xCC	0x11	-	-	-	-	0xCC	Byte 4 of Dest. MAC	Byte 5	Byte 6
Forced Calibration	0xCC	0x12	0x00	0x00	-	-	0xCC	Firmware Version	0x00: Server in Normal Operation 0x01: Client in Normal Operation 0x02: Server in Acquisition Sync 0x03: Client in Acquisition Sync	-
Auto Destination	0xCC	0x15	bit-0: Auto Destination bit-1: Auto Channel bit-4: Enable Auto Destination bit-5: Enable Auto Channel	-	-	-	0xCC	bit-0: Auto Destination bit1: Auto Channel bits-2-7: 0	-	-
Read Digital Inputs	0xCC	0x20	-	-	-	-	0xCC	bit-0: GI0 bit-1: GI1	-	-
Read ADC	0xCC	0x21	0x01: AD In 0x02: Temp 0x03: RSSI	-	-	-	0xCC	MSB of 10 bit ADC	LSB of 10 bit ADC	-
Report Last Valid RSSI	0xCC	0x22	-	-	-	-	0xCC	RSSI	-	-
Write Digital Outputs	0xCC	0x23	bit-0: GO0 bit-1: GO1	-	-	-	0xCC	bit-0: GO0 bit-1: GO1	-	-
Write DAC	0xCC	0x24	Update Period	Duty Cycle	-	-	0xCC	Update Period	Duty Cycle	-
Set Max Power	0xCC	0x25	New Max Power	-	-	-	0xCC	Max Power	-	-
Report Last Packet RSSI	0xCC	0x26	-	-	-	-	0xCC	RSSI	-	-
Long Range Mode ¹	0xCC	0x27	0x00: Normal Mode (Disabled) 0x01: Long Range Mode (Enabled)	-	-	-	0xCC	0x00: Normal Mode (Disabled) 0x01: Long Range Mode (Enabled)	-	-
Transmit Buffer Empty	0xCC	0x30	-	-	-	-	0xCC	0x00	-	-
Disable Sync to Channel	0xCC	0x85	-	-	-	-	0xCC	Channel	-	-

Command Name	Command (All Bytes in Hex)						Return (All Bytes in Hex)			
Deep Sleep Mode	0xCC	0x86	-	-	-	-	0xCC	Channel	-	-
Enter Probe	0xCC	0x8E	0x00: Enter Probe 0x01: Exit Probe			-	0xCC	0x00 or 0x01	-	-
Read Temperature	0xCC	0xA4	-	-	-	-	0xCC	Temp (C)	-	-
Read Temperature at last calibration	0xCC	0xA5	-	-	-	-	0xCC	Temp (C)	-	-
EEPROM Byte Read	0xCC	0xC0	Start Address	Length			0xCC	Start Address	Length	Data
EEPROM Byte Write	0xCC	0xC1	Start Address	Length	Data	-	Starting Address	Length	Data written	
Soft Reset	0xCC	0xFF	-	-	-	-	-	-	-	-

1. Available only on AC4490LR-1000 transceivers.

4.2 Command Descriptions

4.2.1 Enter AT Command Mode

Prior to sending this command, the OEM Host must ensure that the transceiver's RF transmit buffer is empty. If the buffer is not empty, the radio will interpret the command as data and it will be sent over the RF. This can be accomplished by waiting up to one second between the last packet and the AT command.

Note: RF Packet Size must be set to a minimum of 6 to use this command.

Command: <0x41> <0x54> <0x2B> <0x2B> <0x2B> <0x0D>

Number of Bytes Returned: 4

Response: <0xCC> <0x43> <0x4F> <0x4D>

4.2.2 Exit AT Command Mode

The OEM Host should send this command to exit AT Command mode and resume normal operation.

Command: <0xCC> <0x41> <0x54> <0x4F> <0x0D>

Number of Bytes Returned: 4

Response: <0xCC> <0x44> <0x41> <0x54>

4.2.3 Status Request

The OEM Host issues this command to request the status of the transceiver.

Command: <0xCC> <0x00> <0x00>

Number of Bytes Returned: 3

Response: <0xCC> <Version> <Status>

Parameter Range:

<Version> = Firmware version of radio

<Status> = 0x00: Server

0x01: Client in Range

0x03: Client out of Range

4.2.4 Change Channel

The OEM Host issues this command to change the channel of the transceiver.

Command: 0xCC 0x02 <Channel>

Number of Bytes Returned: 2

Response: 0xCC <Channel>

Parameter Range: <Channel> = RF Channel in use

4.2.5 Change Server/Client

The OEM Host issues this command to change the mode of the transceiver from server to client and vice versa.

Command: <0xCC> <0x03> <Mode>

Number of Bytes Returned: 3

Response: <0xCC> <Version> <Mode>

Parameter Range:

<Mode> = 0x00: Server

0x03: Client

<Version> = Firmware version of radio

4.2.6 Change Sync Channel

The OEM Host issues this command to change the sync channel byte and enable sync to channel. See [Appendix II: Sync-to-Channel](#) for more information.

Note: Valid only for server transceivers.

Command: <0xCC> <0x05> <Channel>

Number of Bytes Returned: 3

Response: <0xCC> <Channel>

Parameter Range: <Channel> = Sync Channel

4.2.7 Sleep Walk Power-Down

After the host issues this command, the client transceiver will issue its In_Range line logic high after entering power down. A client in Power Down will remain in sync with a server for a minimum of 2 minutes. To maintain synchronization with the server, the client should re-sync at least once every 2 minutes. This is done by sending the Power Down Wake Up command

and waiting for the In_Range line to issue logic low. Once this occurs, the client is in sync with the server and can be put back into power-down mode.

Note: This command is valid only for client transceivers.

Command: <0xCC> <0x06>

Number of Bytes Returned: 2

Response: <0xCC> <Channel>

Parameter Range: <Channel> = RF Channel currently being used

4.2.8 Sleep Walk Power-Down Wake Up

The OEM Host issues this command to bring the client transceiver out of Power Down mode.

Note: This command is valid only for client transceivers.

Command: <0xCC> <0x07>

Number of Bytes Returned: 2

Response: <0xCC> <Channel>

Parameter Range: <Channel> = RF Channel currently being used

4.2.9 Broadcast Packets

The OEM Host issues this command to change the transceiver operation between Addressed Packets and Broadcast Packets. If Addressed Packets are selected, the transceiver will send all packets to the transceiver designated by the Destination Address programmed in the transceiver. If Broadcast Packets are selected, the transceiver will send its packets to all transceivers on that network.

Command: <0xCC> <0x08> <Mode>

Number of Bytes Returned: 2

Response: <0xCC> <Mode>

Parameter Range: <Mode> = 0x00: Addressed
 0x01: Broadcast

4.2.10 Write Destination Address

The OEM Host issues this command to the transceiver to change the Destination Address.

Note: Only the three Least Significant Bytes of the MAC Address are used for packet delivery.

Command: <0xCC> <0x10> <MAC3> <MAC2> <MAC1>

Number of Bytes Returned: 4

Response: <0xCC> <MAC3> <MAC2> <MAC1>

Parameter Range: <MAC> = 0x00 - 0xFF corresponding to 3 LSB's of destination MAC Address

4.2.11 Read Destination Address

The OEM Host issues this command to the transceiver to read the destination address.

Note: Only the three Least Significant Bytes of the MAC Address are used for packet delivery.

Command: <0xCC> <0x11>

Number of Bytes Returned: 4

Response: <0xCC> <MAC3> <MAC2> <MAC1>

Parameter Range: <MAC> = 0x00 - 0xFF corresponding to 3 LSB's of destination MAC Address

4.2.12 Auto Calibration (Forced Recalibration)

When enabled, Auto Calibrate causes the radio to measure the temperature every 30 to 60 seconds. If the temperature changes more than 30°C from the last calibration, the radio will initiate a recalibration.

During the recalibration, the radio will not assert CTS high. Recalibration can take up to 3 seconds and the command response will not be sent to the OEM Host until recalibration is complete.

Note: If Auto Calibration is *disabled*, the CL4490 radio may fail to lock onto frequency. If it does, the radio timeouts after 5 ms and performs a recalibration.

Command: <0xCC> <0x12> <0x00> <0x00>

Number of Bytes Returned: 3

Response: <0xCC> <Version> <Status>

Parameter Range: <Version> = Firmware version of radio
<Status> = 0x00: Server in range
0x01: Client in range
0x02: Server out of range
0x03: Client out of range

4.2.13 Auto Destination/Auto Channel

The Host issues this command to change the Auto Destination & Auto Channel settings. When issuing this command, the Auto Destination/Auto Channel settings is only changed if the corresponding enable bit is set.

Command: <0xCC> <0x15> <Auto Dest>

Number of Bytes Returned: 2

Response: <0xCC> <Auto Dest>

Parameter Range:

<Auto Dest>= bit 7: Ignored
bit 6: Ignored
bit 5: Enable Auto Chan. Modification
bit 4: Enable Auto Dest. Modification
bit 3: Ignored
bit 2: Ignored
bit 1: Auto Channel
bit 0: Auto Destination

4.2.14 Read Digital Inputs

The OEM Host issues this command to read the state of both digital input lines.

Command: <0xCC> <0x20>

Number of Bytes Returned: 2

Response: <0xCC> <Digital In>

Parameter Range: <Digital In> = bit-0: GI0
bit-1: GI1

4.2.15 Read ADC

The OEM Host issues this command to read any of the three onboard 10-bit A/D converters. Because the RF is still active in On-the-Fly Command Mode, the transceiver will not process the command until there is no activity on the network. The Read RSSI command is therefore useful for detecting interfering sources but will not report the RSSI from a remote transceiver on the network. The equations for converting these 10 bits into analog values are as follows:

Analog Voltage = (10 bits / 0x3FF) * 3.3 V

Temperature (°C) = ((Analog Voltage - 0.3) / 0.01) - 30

Instantaneous RSSI value (dBm) = -105 + (0.22 * (0x3FF - 10 bits))

Command: <0xCC> <0x21> <Port>

Number of Bytes Returned: 3

Response: <0xCC> <Hi ADC> <Lo ADC>

Parameter Range:

<Port> = 0x00: AD In
0x01: Temperature
0x02: Instantaneous RSSI
<Hi ADC> = MSB of requested 10-bit ADC value
<Lo ADC> = LSB of requested 10-bit ADC value

4.2.16 Report Last Valid RSSI

Since RSSI values are only valid when the local transceiver is receiving an RF packet from a remote transceiver, instantaneous RSSI can be tricky to use. Therefore, the transceiver stores the most recent valid RSSI value as measured the last time the transceiver received a packet or beacon. The Host issues this command to retrieve that value.

Note: This value will default to 0xFF on a client and 0x00 on a server if no valid RSSI measurement has been made since power-up.

Command: <0xCC> <0x22>

Number of Bytes Returned: 2

Response: <0xCC> <Last Valid RSSI>

Table 2: Received Signal Strength

RSSI (dBm)	Hex Value	RSSI (dBm)	Hex Value
-92	C0	-72	63
-91	BC	-71	5F
-90	BB	-70	5B
-89	B9	-69	58
-88	B8	-68	54
-87	AE	-67	4F
-86	A9	-66	4B
-85	A2	-65	47
-84	92	-64	43
-83	8D	-63	3D
-82	86	-62	2A
-81	82	-60	25
-80	7D	-58	1A
-79	79	-56	16
-78	75	-54	13
-77	72	-52	11
-76	6F	-50	0E
-75	6B	-48	0D
-74	68	-46	0C
-73	66	-44	0B

Note: The receiver is saturated after -45 dBm and cannot accurately measure the RSSI above -45 dBm.

4.2.17 Write Digital Outputs

The OEM Host issues this command to write both digital output lines to particular states.

Note: This command should only be used when Protocol Status (0xC2) is not set to 0xE3.

Command: <0xCC> <0x23> <Digital Out>

Number of Bytes Returned: 2

Response: <0xCC> <Digital Out>

Parameter Range:

<Digital Out>= bit-0: GO0
bit-1: GO1

4.2.18 Write DAC

The OEM Host issues this command to write DA_Out to a particular voltage. The transceiver uses a PWM (Pulse Width Modulator) to generate the analog voltage. The theory behind a PWM is that a binary pulse is generated with a fixed rate (<Data 1>) and duty cycle (<Data 2>). As such, this pin toggles between High & Low. This signal is filtered via an on-board R-C circuit and an analog voltage is generated.

Duty cycle specifies the ratio of time in one cycle that the pulse spends High proportionate to the amount of time it spends Low. So, with a duty cycle of 50% (0x80), the pulse is High 50% of the time and Low 50% of the time; therefore the analog voltage would be half of 3.3V or 1.65V. A broad filter has been implemented on the transceiver and there is no advantage to using a slower update period. Generally, a faster update period is preferred.

Command: <0xCC> <0x24> <Data1> <Data2>

Number of Bytes Returned: 3

Response: <0xCC> <Data1> <Data2>

Parameter Range: <Data1> = Period of PWM, Hex value

<Data2> = Duty Cycle of PWM, Hex value

$\text{Data1} = \frac{(T_{\text{period}} * 14.7456^6)}{255} - 1$; Where T_{period} == Period of the PWM in seconds
Data1 value is then converted from decimal to hex

$\text{Data2} = \text{DC}_{\text{percentage}} * 255$; Where $\text{DC}_{\text{percentage}}$ == duty cycle of the PWM in decimal percent (50% = 0.5)

Note: The duty cycle is represented at this pin as an analog voltage. 50% duty cycle is half of 3.3V or 1.65V.

4.2.19 Set Max Power

The OEM Host issues this command to limit the maximum transmit power emitted by the transceiver. This can be useful to minimize current consumption and satisfy certain regulatory requirements.

Note: The radios are shipped at maximum allowable power.

Command: <0xCC> <0x25> <Max Power>

Number of Bytes Returned: 2

Response: <0xCC> <Max Power>

Parameter Range: <Max Power>= New Max Power setting

4.2.19.1 Output Power

Table 3: AC4490-200

EEPROM Value (Hex)	Current (mA)	dBm	mW
0	61	-22	0.01
1	64	-9	0.13
2	65	-3	0.5
3	66	1	1.26
4	67.5	4	2.51
5	70	7	5.01
6	73	9	7.94
7	77	10.5	11.22
8	83	12	15.85
9	88	13.5	22.39
A	93.5	14.5	28.18
B	99	15.5	35.48
C	105	16.5	44.67
D	110.5	17	50.12
E	114.5	17.5	56.23
F	117.5	18.5	70.79
1E	126	19.5	89.13
60	127.5	20	100

Table 4: AC4490-1000M (revision 1.0 and 2.0 hardware)

Revision 1.0 Hardware				Revision 2.0 Hardware			
EEPROM Value (Hex)	Current (mA)	dBm	mW	EEPROM Value (Hex)	Current (mA)	dBm	mW
00	440	-5	0.31	0	400	-9.7	0.10
01 (Low Power)	450	9.5	8.91	1	410	6.0	3.98
02	460	15.3	33.88	2	420	11.0	12.58
03	470	18.2	66.06	3	430	14.2	26.30
04	480	20.3	107.15	4	440	16.1	40.73
05	510	21.8	151.35	5	470	17.8	60.25
06	530	22	158.48	6	490	19.2	83.17
07 (Quarter Power)	560	24	251.18	7	520	20.4	109.64
08	590	25.1	323.59	8	560	21.5	141.25
09	620	25.8	380.18	9	590	22.3	169.82
0A (Half Power)	660	26.5	446.68	A	630	23.1	204.17
B	700	27.1	512.86	B	660	23.8	239.88
C	730	27.7	588.84	C	690	24.5	281.83
D	760	28.2	660.69	D	720	25.1	323.59
E	800	28.5	707.94	E	750	25.7	371.53
				F	790	26.1	407.38
				50	950	28.3	876.08

Note: Maximum RF TX power of 28 dBm is calibrated and programmed in production per module, this is the same for revision 1.0 and 2.0 hardware. However, Half power (26dBm)=eeprom value 0A, Quarter power (23dBm)=eeprom value 07 and Low power (10dBm)=eeprom value 01 settings gives a lower RF TX power on AC4490-1000M revision 2.0 hardware. Customer must set the appropriate EEPROM hex value on AC4490-1000M revision 2.0 hardware to get half power (26dBm), quarter power (23dBm), and low power (10dBm) modes.

4.2.20 Long Range Mode

The OEM Host issues this command to temporarily enable or disable Long Range Mode in the transceiver.

Note: Only available on AC4490LR-1000 transceivers with firmware v6.7+.

Command: <0xCC> <0x27>

Number of Bytes Returned: 2

Response: <0xCC> <Mode>

Parameter Range:

<Mode> = 0x00: Disable Long Range Mode

0x01: Enable Long Range Mode

4.2.21 Transmit Buffer Empty

The OEM Host issues this command to determine when the RF transmit buffer is empty. The Host will not receive the transceiver response until that time.

Command: <0xCC> <0x30>

Number of Bytes Returned: 2

Response: <0xCC> <0x00>

4.2.22 Disable Sync-to-Channel

The OEM Host issues this command to disable Sync-to-Channel mode. See [Appendix I: API](#) for more information.

Note: This command is valid only for servers.

Command: <0xCC> <0x85>

Number of Bytes Returned: 2

Response: <0xCC> <Channel>

Parameter Range: <Channel> = RF Channel currently being used

4.2.23 Deep Sleep Mode

The OEM Host issues this command to put the transceiver into Deep Sleep mode. Once in Deep Sleep mode, the transceiver disables all RF communications and will not respond to any further commands until being reset or power-cycled.

Note: This command is valid for both servers and clients.

Command: <0xCC> <0x86>

Number of Bytes Returned: 2

Response: <0xCC> <Channel>

Parameter Range: <Channel> = RF Channel currently being used

4.2.24 Read Temperature

The OEM Host issues this command to read the onboard temperature sensor. The transceiver reports the temperature in °C where 0x00 - 0x50 corresponds to 0 - 80 °C and 0xD8 - 0x00 corresponds to - 40 - 0 °C.

Command: <0xCC> <0xA4>

Number of Bytes Returned: 2

Response: <0xCC> <Temp>

Parameter Range: <Temp> = Temperature of module

4.2.25 Read Temperature at Last Calibration

The OEM Host issues this command to read the temperature of the radio at the time of its last calibration. The transceiver reports the temperature in °C where 0x00 - 0x80 corresponds to 0 - 80° C and where 0xD8 - 0x00 is the two's complement representation corresponding to -40 - 0 °C.

Note: 0xD8 is a twos complement representation of -40 – 0.

Command: <0xCC> <0xA5>

Number of Bytes Returned: 2

Response: <0xCC> <Temp>

Parameter Range: <Temp> = Temperature at last calibration

4.2.26 Probe

When the OEM Host issues this command, the transceiver sends out a query every 500 ms. The transceivers randomly choose a query to respond to. After responding to a probe, the transceiver will wait at least 10 seconds before responding to another probe.

Note: This command can only be sent from a server radio.

Command: <0xCC> <0x8E> <Probe>

Number of Bytes Returned: 2

Response: <0xCC> <Probe>

Parameter Range:

<Probe> = 0x00: Disable Probe Mode

0x01: Enable Probe Mode

Transceiver's Response

Upon hearing the remote transceiver's probe acknowledge, the transceiver sends a response to the OEM Host.

Command: N/A

Number of Bytes Returned: 5

Response: <0xCC> <Data> <MAC3> <MAC2> <MAC1>

Parameter Range:

<Data> = bit-7: 0 Client

bit-7: 1 Server

bits 6-0: RF Channel

4.2.27 EEPROM Byte Read

Upon receiving this command, a transceiver will respond with the desired data from the addresses requested by the OEM Host. See [EEPROM Parameters](#).

Command: <0xCC> <0xC0> <Start> <Length>

Number of Bytes Returned: 4+

Response: <0xCC> <Start> <Length> <Data>

Parameter Range:

<Start> = Address to begin reading from

<Length> = Length of bytes to read

<Data> = Requested data

4.2.28 EEPROM Byte Write

Upon receiving this command, a transceiver will write the data byte to the specified address but will not echo it back to the OEM Host until the EEPROM write cycle is complete (up to 10 ms).

Multiple byte writes of up to 128 bytes are allowed. An EEPROM boundary exists between addresses 0x7F and 0x80. No single EEPROM write command shall write to addresses on both sides of that EEPROM boundary. See [EEPROM Parameters](#).

Note: Only the last byte written will be displayed in the command response.

Command: <0xCC> <0xC1> <Start> <Length> <Data>

Number of Bytes Returned: 4+

Response: <Starting Address> <Length> <Data written>

Parameter Range:

<Start> = Address to begin writing from

<Length> = Length of bytes to write

<Data> = Last byte of data written

4.2.29 Reset

The OEM Host issues this command to perform a soft reset of the transceiver. Any transceiver settings modified by CC commands will revert to the default values stored in the EEPROM.

Command: <0xCC> <0xFF>

Number of Bytes Returned: None

Response: None

4.3 API Control

API Control is a powerful feature offered by the AC4490. When enabled, the API Receive Packet, API Transmit Packet, API Send Data Complete and Enhanced API Receive Packet features provide dynamic packet routing and packet accounting ability to the OEM host, thereby eliminating the need for extensive programming on the OEM host side. API operation utilizes specific packet formats; specifying various vital parameters used to control radio settings and packet routing on a packet-by-packet basis. The API features can be used in any combination that suits the OEM's specific needs.

4.3.1 Receive API Packet

Note: Implemented in firmware v.6.3 and later.

By default, the source MAC is not included in the received data string sent to the OEM host. For applications where multiple radios are sending data, it may be necessary to determine the origin of a specific data packet. Receive API Packet can be enabled to determine the sender of a message. This causes the receiving radio to add a header to the received packet detailing the length of the data packet and the sender's MAC address. The format of the Receive API Packet is:

0x83	Payload Data Length	Sender's MAC	Payload Data
------	---------------------	--------------	--------------

Note: If Receive API is enabled, the Enhanced API Receive feature should be disabled by clearing bit-0 of the Enhanced API control byte, EEPROM address 0xC6.

4.3.2 Enhanced API Receive Packet

Note: Implemented in firmware v.6.7 and later.

When Enhanced API Receive Packet is enabled, all packets received by the transceiver include the MAC address of the source radio as well as an RSSI indicator which can be used to determine the link quality between the two. API Receive Packet is enabled when bit-0 of the Enhanced API Control byte is enabled. Upon receiving a packet the radio sends its OEM Host the packet in the following format:

0x81	Payload Data Length (0x01 – 0x50)	Laird Use	RSSI*	Source MAC (2, 1, 0)	Payload Data
------	-----------------------------------	-----------	-------	-------------------------	--------------

Note: When both API Send Data Complete and API Receive Packet are enabled, the Send Data Complete will be received before the transceiver sees the Receive API Packet. This may be reversed when the API Send Data Complete is missed and is being resent after the API Receive Packet is received.

Note: If Enhanced API Receive is enabled, the Receive API feature should be disabled by setting EEPROM byte 0xC1 to 0xFF.

4.3.3 API Transmit Packet

Note: Implemented in firmware v6.7 and later.

API Transmit Packet is a powerful command that allows the OEM host to dynamically send data to a single or multiple (broadcast) transceiver(s) on a packet-by-packet basis. API Transmit Packet is enabled when bit-1 of the Enhanced API Control byte (EEPROM byte 0xC6) is enabled. The OEM host must use the following format to transmit a packet over the RF when using Transmit API packets:

0x81	Payload Data Length (0x01 – 0z50)	Laird Use	Transmit Retries/ Broadcast Attempts	Destination MAC (2, 1, 0)	Payload Data
------	--------------------------------------	-----------	---	------------------------------	--------------

If the OEM Host does not encode the header correctly, the transceiver sends the entire string (up to 80 bytes) and looks for the header in the next data.

Although the seven bytes of overhead are not sent over the RF, they are kept in the buffer until the packet is sent. Keep this in mind so as not to overrun the 256-byte buffer.

Setting the Destination MAC to **0xFF 0xFF 0xFF** broadcasts the packet to all available transceivers.

Note: If the OEM host does not properly encode the header of the Tx API packet, the string (up to 80 bytes) is sent to the MAC address from the header of the last known good Tx API encoded packet.

4.3.4 API Send Data Complete

Note: Implemented in v6.7 of the firmware and later.

API Send Data complete can be used as a software acknowledgement indicator. When a radio sends an addressed packet, it looks for a received acknowledgement (transparent to OEM host). If an acknowledgement is not received, the packet is retransmitted until one is received or all retries are used.

API Send Data Complete is enabled when bit-2 of the Enhanced API Control byte (EEPROM byte 0xC6) is enabled. The transceiver sends the OEM host the following data upon receiving an RF acknowledge or exhausting all attempts:

0x82	Laird Use	RSSI*	0x00: Failure 0x01: Success
------	-----------	-------	--------------------------------

RSSI* is how strong the local transceiver heard the remote transceiver.

Successful RF Acknowledge updates the Success/Failure bit.

A success is always displayed when sending broadcast packets (after all broadcast attempts have been exhausted).

5 EEPROM PARAMETERS

The OEM Host can program various parameters that are stored in EEPROM which become active after a power-on reset. [Table 5](#) gives the locations and descriptions of the parameters that the OEM Host can read or write. Factory default values are also shown.

WARNING: Do not write to any EEPROM addresses other than those listed below. Do not copy one transceiver's EEPROM to another transceiver as doing so may cause the transceiver to malfunction. Only the Configuration Utility should be used to copy one configuration into another transceiver.

Table 5: EEPROM Parameters

Parameter	EEPROM Address	Length (Bytes)	Range	Default	Description
Product ID	0x00	40			Product identifier string. Includes revision information for software and hardware.
Page Refresh	0x3D	1	0x01 - 0xFF	0x18	Specifies the maximum amount of time a transceiver will report In-Range without having heard a server's beacon (equal to hop period * value). Note: Do not set to 0x00.

Parameter	EEPROM Address	Length (Bytes)	Range	Default	Description
Stop Bit Delay	0x3F	1	0x00 - 0xFF	0xFF	For systems employing Parity, the serial stop bit might come too early. Stop Bit Delay controls the width of the last bit before the stop bit occurs. 0xFF = Disable Stop Bit Delay (12 μs) 0x00 = (256 * 1.6 μs) + 12 μs 0x01 - 0xFE = (value * 1.6 μs) + 12 μs
Channel Number	0x40	1	0x00 - 0x37	1x1: 0x00 200: 0x00 1000: 0x10	Set 0 = 0x00 - 0x0F (US/Canada): 1x1/200 Set 1 = 0x10 - 0x2F (US/Canada): 1x1/1000 Set 2 = 0x30 - 0x37 Australia: 1x1/200/1000
Server/Client Mode	0x41	1	0x01 - 0x02	0x02	0x01 = Server 0x02 = Client
Baud Rate Low	0x42	1	0x00 - 0xFF	0xFC	Low byte of the interface baud rate. Default baud rate is 57600.
Baud Rate High	0x43	1	0x00	0x00	High byte of interface baud. Always 0x00.
Control 0	0x45	1	0x00 - 0xFF	0x14	Settings are: bit-7: One Beacon Mode 0 = Disable One Beacon Mode 1 = Enable One Beacon Mode bit-6: DES Enable 0 = Disable Encryption 1 = Enable Encryption bit-5: Sync-to-Channel 0 = Disable Sync-to-Channel 1 = Enable Sync-to-Channel bit-4: Laird Use Only bit-3: Laird Use Only bit-2: Laird Use Only bit-1: RF Delivery 0 = Transmit using Addressed packets 1 = Transmit using Broadcast packets bit-0: Laird Use Only
Frequency Offset	0x46	1	0x00 - 0xFF	0x01	Protocol parameter used in conjunction with Channel Number to satisfy unique regulations.
$\overline{\text{CMD}}$ /Data RX Disable	0x4B	1	0xE3, 0xFF	0xFF	0xE3 = Enable $\overline{\text{CMD}}$ /Data RX Disable 0xFF = Disable $\overline{\text{CMD}}$ /Data RX Disable
Transmit Retries	0x4C	1	0x01 - 0xFF	0x10	Maximum number of times a packet is transmitted when Addressed packets are selected. Note: Do not set to 0.

Parameter	EEPROM Address	Length (Bytes)	Range	Default	Description
Broadcast Attempts	0x4D	1	0x01 - 0xFF	0x04	Number of times each packet is transmitted when Broadcast packets are selected. Note: Do not set to 0.
API Control	0x56	1	0x00 - 0xFF	0x43	Settings are: bit-7: Laird Use Only bit-6: Laird Use Only bit-5: Unicast Only 0 = Disable Unicast Only 1 = Enable Unicast Only bit-4: Auto Destination 0 = Use destination address 1 = Use auto destination bit-3: Client Auto Channel 0 = Disable Auto Channel 1 = Enable Auto Channel bit-2: RTS Enable 0 = Ignore RTS 1 = Transceiver obeys RTS bit-1: Duplex 0 = Half Duplex 1 = Full Duplex bit-0: Auto Config 0 = Use EEPROM values 1 = Auto Configure values
Interface Timeout	0x58	1	0x02 - 0xFF	0x04	Specifies a byte gap timeout, used in conjunction with RF Packet Size to determine when a packet coming over the interface is complete (0.5 ms per increment).
Sync Channel	0x5A	1	0x00 - 0xFF	0x01	Used to synchronize the hopping of collocated systems to minimize interference.
RF Packet Size	0x5B	1	0x01 - 0x80	0x80	Used in conjunction with Interface Timeout; specifies the maximum size of an RF packet. Note: Must be set to a minimum of 6 in order to send the Enter AT command.
$\overline{\text{CTS}}$ On	0x5C	1	0x01 - 0xFF	0xD2	$\overline{\text{CTS}}$ will be deasserted (High) when the transmit buffer contains at least this many characters.
$\overline{\text{CTS}}$ On Hysteresis	0x5D	1	0x00 - 0xFE	0xAC	Once $\overline{\text{CTS}}$ has been deasserted, $\overline{\text{CTS}}$ will be reasserted (Low) when the transmit buffer is contains this many or less characters.
Max Power	0x63	1	0x00 - 0x60	Set in Production & can vary	Used to increase/decrease the output power. Note: The transceivers are shipped at maximum allowable power.

Parameter	EEPROM Address	Length (Bytes)	Range	Default	Description
Modem Mode	0x6E	1	0xE3, 0xFF	0xFF	0xE3 = Enable Modem Mode 0xFF = Disable Modem Mode
Parity	0x6F	1	0xE3, 0xFF	0xFF	0xE3 = Enable Parity 0xFF = Disable Parity Note: Enabling parity cuts throughput and the interface buffer size in half.
Destination ID	0x70	6	0x00 - 0xFF	0xFF	Specifies destination for RF packets
System ID	0x76	1	0x00 - 0xFF	0x01	Similar to network password. Radios must have the same system ID to communicate with each other.
MAC ID	0x80	6	0x00 - 0xFF		Factory programmed unique IEEE MAC address.
Original Max Power	0x8E	1		Set in Production & can vary	Copy of original max power EEPROM setting. This may be referenced but should not be modified.
Product ID	0x90	15			0x90 - 0x93: Product ID 0x94 - 0x95: Prefix (CL, CN, or AC) 0x96 - 0x99: Power (200M, 200A , 1000, 1x1) Note: There will be a period in front of the 1x1 to keep the field at four bytes. 0x9A - 0x9C: Interface (232, 485 , TTL) 0x9D - 0x9E: Setup script (01 is stock) 0x9F: Reserved for future use; always 0xFF
Protocol Status / Receive ACK	0xC0	1	0xE3, 0xFF	0xFF	0xE3 = GO0 outputs the Protocol Status and GO1 outputs the Received Acknowledgement signal 0xFF = Disable Protocol Status / Receive ACK
Receive API	0xC1	1	0xE3, 0xFF	0xFF	0xE3 = Enabled 0xFF = Disabled

Parameter	EEPROM Address	Length (Bytes)	Range	Default	Description
Enhanced API Ctrl.	0xC6	1		0xF8	Settings are: bit-7: Enhanced API Control Enable 0 = Enable Enhanced API Control 1 = Disable Enhanced API Control bit-6: Laird Use Only bit-5: Laird Use Only bit-4: Laird Use Only bit-3: Laird Use Only bit-2: Send Data Complete Enable 0 = Disable 1 = Enable bit-1: API Transmit Packet Enable 0 = Disable 1 = Enable bit-0: Enhanced API Receive Packet Enable 0 = Disable 1 = Enable
Auto Calibrate	0xCC	1	0xE3, 0xFF	0xFF	0xE3 = Enable Auto Calibrate 0xFF = Disable Auto Calibrate
DES Key	0xD0	7	0x00 - 0xFF		56-bit Data Encryption key

5.1.1 Radio Interface

The Radio Interface section manages the following settings:

Interface Timeout	Interface Timeout specifies a maximum byte gap between consecutive bytes. When that byte gap is exceeded, the bytes in the transmit buffer are sent out over the RF as a complete packet. Interface Timeout is adjustable in 0.5 ms increments and has a tolerance of ± 0.5 ms. The Interface Timeout should not be set below 2. The default value for Interface Timeout is 0x04 (2 ms) and should be adjusted accordingly when changing the transceiver baud rate.
RF Packet Size	RF Packet Size is used in conjunction with Interface Timeout to determine when to delineate incoming data as an entire packet based on whichever condition is met first. When the transceiver receives the number of bytes specified by RF Packet Size without experiencing a byte gap equal to Interface Timeout, that block of data is processed as a complete packet. Every packet the transceiver sends over the RF contains extra header bytes not counted in the RF Packet Size. It is much more efficient to send a few large packets than to send many short packets.
CTS On/ CTS On Hysteresis (Flow Control)	See Flow Control in the AC4490 Datasheet for more information.

<p>Max Transmit Retries (For clients and servers in Point-to-Point networks only)</p>	<p>This value represents the maximum number of times a particular data packet will be transmitted unsuccessfully, or without an acknowledgement, before the AC4490 discards the packet. The default value is 16 attempts. If communication is lost and the client's Link LED is on, try increasing this value in small increments until communication is re-established.</p>
	<p>Note: This value is always associated to client radios and server radios in Point to Point Mobile. <u>The valid range of values for this field is 1 to 255.</u></p>
<p>Broadcast Attempts (Point-to-Multipoint only)</p>	<p>This value represents the number of times a data packet will be transmitted by a transceiver in Broadcast Mode. The default value is 4 attempts. If communication is lost and the clients' Link LED is on, try increasing this value in small increments until communication is reestablished. Valid values for this field are 1 - 255.</p>
	<p>Note: All Broadcast Attempts are used whether the packet was received without error by the receiving radios or not.</p>
<p>Range Refresh</p>	<p>Range Refresh specifies the maximum amount of time a client reports in range without hearing a server beacon. Each time the client hears a beacon, it resets its Range Refresh timer. If the timer reaches zero, the client goes out of range, takes In_Range pin High and enters acquisition mode to find the server once again. The range refresh is equal to the hop period (20 ms) x Range refresh value.</p>
	<p>Note: Range Refresh should not be set to 0x00.</p>

5.1.2 Radio RF

The Radio RF section manages the following settings:

<p>Client / Server</p>	<p>Designates AC4490 type. In each network, there must be only one server. All other AC4490 units must be programmed as clients. The number of clients in the network is not limited; however, if performance diminishes, consider additional RF Networks.</p>
<p>RF Channel Number</p>	<p>A number that designates an independent network of AC4490 units. Up to 32 independent networks can be created.</p>
	<p>Note: The valid range of values for this field is 16 to 47.</p>
<p>Sync to Channel</p>	<p>Sync-to-channel can be used to synchronize server frequency hopping and prevent interference between collocated systems. A server transceiver with Sync-to-Channel enabled must have its Sync Channel set to another server's RF Channel number. A server with Sync-to-Channel enabled must set its sync channel to a value less than its RF Channel number. Collocated networks using Sync-to-Channel must use the same channel set and system ID.</p>
	<p>Note: Sync-to-Channel Radio Feature must be enabled.</p>
	<p>If server A (with sync-to-channel enabled) can't sync to server B (on the sync channel), server A will be unable to communicate with its clients. It must wait until it syncs with server B (at which point In_Range is asserted), before establishing communications. Server B will not be affected and can communicate with its clients.</p>

See [Appendix I: API](#) for further details and sample configuration.

Max Power

Max Power provides a means for controlling the RF output power of the AC4490. Output power and current consumption can vary by as much as $\pm 10\%$ per transceiver for a particular Max Power setting. Contact Laird for assistance in adjusting Max Power.

Note: The max power is set during Production and may vary slightly from one transceiver to another. The max power can be set as low as desired but should not be set higher than the original factory setting. A backup of the original power setting is stored in EEPROM address 0x8E.

Output Power

Table 6: AC4490-200

EEPROM Value (Hex)	Current (mA)	dBm	mW
0	61	-22	0.01
1	64	-9	0.13
2	65	-3	0.5
3	66	1	1.26
4	67.5	4	2.51
5	70	7	5.01
6	73	9	7.94
7	77	10.5	11.22
8	83	12	15.85
9	88	13.5	22.39
A	93.5	14.5	28.18
B	99	15.5	35.48
C	105	16.5	44.67
D	110.5	17	50.12
E	114.5	17.5	56.23
F	117.5	18.5	70.79
1E	126	19.5	89.13
60	127.5	20	100

Table 7: AC4490-1000 (revision 1.0 hardware)
(Refer to [Table 4](#) for revision 1.0 hardware vs. revision 2.0 hardware comparison)

EEPROM Value (Hex)	Current (mA)	dBm	mW
00	440	-5	0.31
01 (Low Power)	450	9.5	8.91
02	460	15.3	33.88
03	470	18.2	66.06
04	480	20.3	107.15
05	510	21.8	151.35

06	530	22	158.48
07 (Quarter Power)	560	24	251.18
08	590	25.1	323.59
09	620	25.8	380.18
0A (Half Power)	660	26.5	446.68
B	700	27.1	512.86
C	730	27.7	588.84
D	760	28.2	660.69
E	800	28.5	707.94

System ID A number from 0 to 256 that provides added security to each independent network of AC4490 units. The System ID is used in conjunction with the Channel Number and serves as an RF password to maintain secure transfers of data. The combination of the Channel Number and System ID must be unique to each network of AC4490s to establish communication. Multiple servers in the same coverage area must be programmed with different Channel Numbers to prevent inoperability of the networks. The System ID will not prevent inoperability that occurs from locating multiple servers with the same Channel Number in the same coverage area.

Note: Separate Collocated AC4490 networks must operate on different Channel Networks. All units in a given AC4490 network must have identical Channel Numbers and System IDs.

6 APPENDIX I: API

The API feature set of the AC4490 provides powerful, dynamic packet routing capabilities to the OEM host. The number of API configurations is endless since individual radios can all be configured differently to suit the OEM host's varying needs. Two of the most common implementations are described in the following pages.

6.1.1 Polling Network

Many applications require multiple locations to report back to a single access point. One solution is to enter Command mode, change the transceiver's destination address and then exit Command mode to resume normal operation. When it is time to communicate with another transceiver, the process is repeated. However, this method costs time and reduction in throughput since unnecessary commands are issued. As an alternative, the Transmit API command can be used to control packet routing on a packet-by-packet basis.

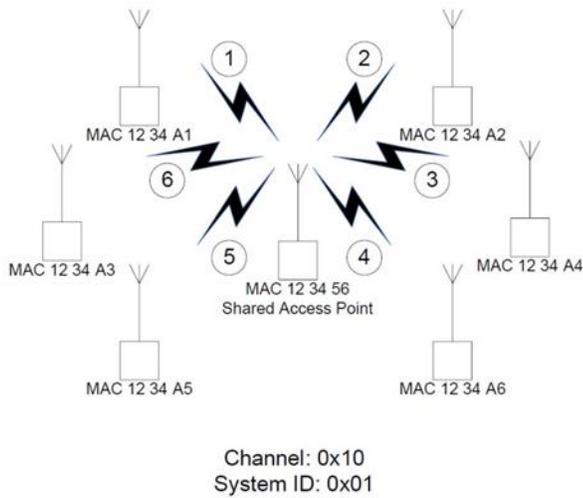
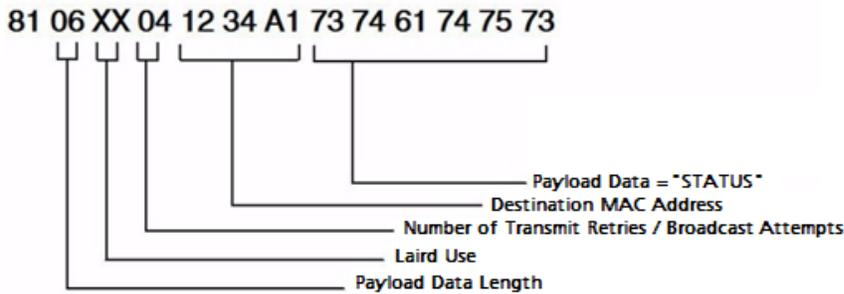


Figure 3: Polling network

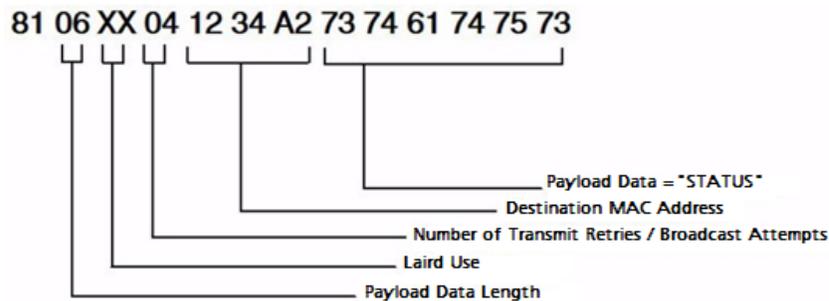
The simplest implementation consists of a smart Shared Access Point (SAP) with a microcontroller or processor that has Transmit API enabled. The SAP controls which transceiver(s) each packet is routed to. Broadcast packets should be used when all remotes are to receive the same message. Addressed packets should be used when communication with a single remote only is desired. An example of each is shown in the following sections.

6.1.1.1 Addressed Transmit API

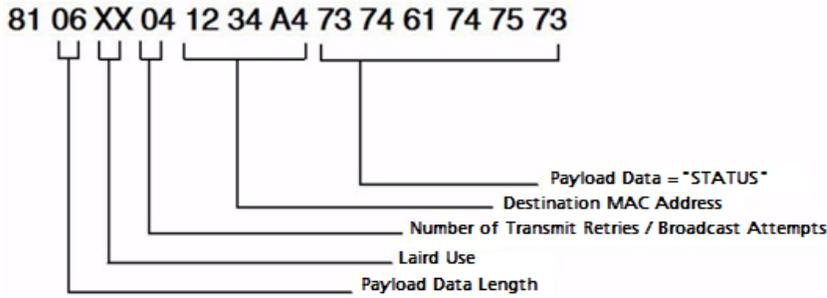
To poll radio 1, the SAP transmits the packet using the following format:



To poll radio 2, the SAP transmits the packet using the following format:



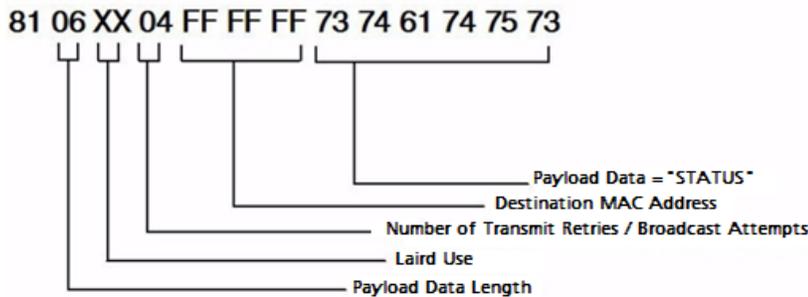
To poll radio 2, the SAP transmits the packet using the following format:



This continues until the SAP successfully polls all radios.

6.1.1.2 Broadcast Transmit API

To send out a universal poll request or data packet, the OEM may wish to utilize the broadcast portion of the Transmit API command. The Broadcast command is similar to the addressed command. However, you must set all three Destination MAC Address values to **0xFF**.



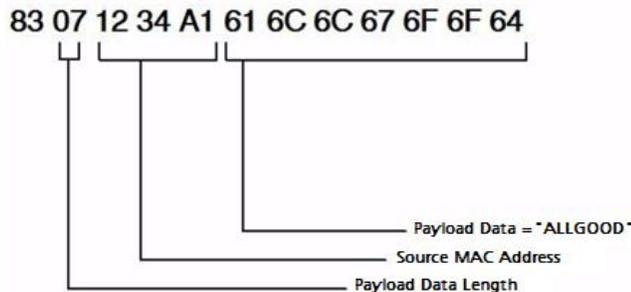
The remote response is dependent on the OEM's specific needs and equipment. In many cases, remote radios are connected to dummy terminals without the intelligence to filter out or append specific portions of a packet that is transmitted or received. Since the seven bytes of overhead in the Transmit API command are not sent over the RF, the remotes receive only the payload data, **STATUS**. If auto-destination is enabled on the remote radio, the transceiver automatically changes its destination address to that of the radio from which it last received a packet. When the remote device sends its response, it is automatically routed back to the SAP.

Depending on the API configuration of the SAP, the packet will be received in one of two formats:

- [Receive API](#)
- [Time Division Multiple Access Network](#)

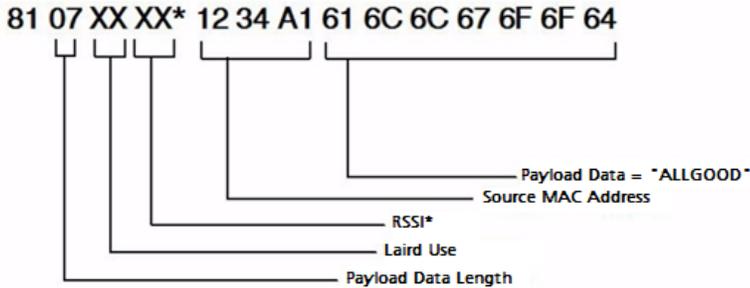
6.1.2 Receive API

When Receive API is enabled, the transceiver receives the reply data and the MAC address of the source radio. When Receive API is enabled, every packet received by the transceiver is sent to the host in the following format:



6.1.2.1 Enhanced Receive API

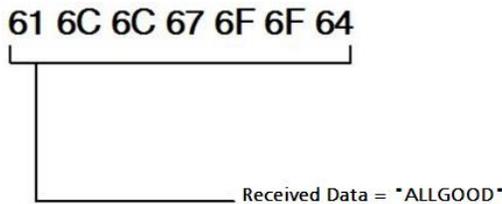
When Enhanced Receive API is enabled, the transceiver receives the reply data plus the MAC address of the source radio and one RSSI value (RSSI* is how strong the local transceiver hears the remote transceiver).



It may be useful to the OEM Host to determine from which radio each packet originated. When Enhanced Receive API is enabled, every packet received by the transceiver is received in the above format.

6.1.2.2 Normal Receive Mode (non-API)

If Receive API is not enabled, the transceiver receives the reply data only (i.e. "ALLGOOD") from each transceiver. With this receive mode, the SAP will not know which radio the data was received from and the messages will have random receipt times depending on which radio was able to control the bandwidth first.



6.1.2.3 Loopback Repeater

The simplest repeater to implement is a loopback repeater. A loopback repeater can be created by connecting the transceiver's RXD and TXD lines together. When the radio receives data, it will retransmit the data to all available transceivers on the network. It is important not to have two loopback repeaters in range of each other as they will continuously transmit data back and forth.

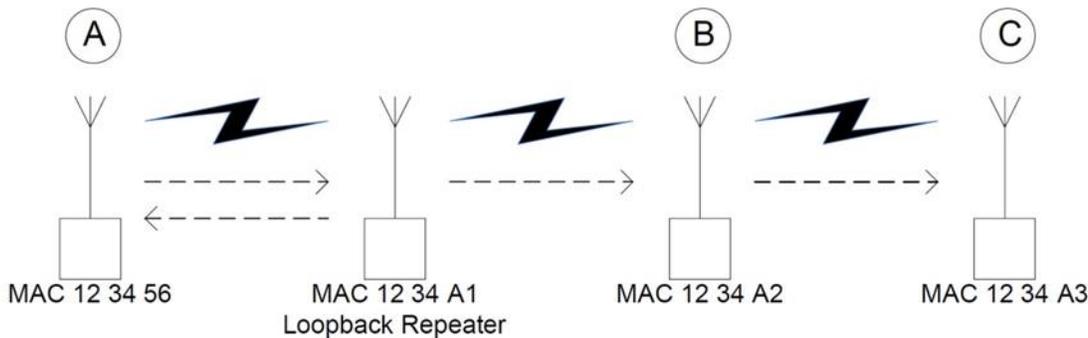


Figure 4: Loopback Repeater

If radios B & C in the above picture are not within range of radio A, they will not be able to receive or respond to communications from radio A. A loopback repeater can be added between the three such that it is in range of both radio A and radios B & C. When the repeater receives a packet from radio A, it will transmit the packet out to radios B & C. If the repeater is set to Broadcast mode, radio A will receive a copy of each packet that it sends. If the repeater has a specific destination address (i.e. 12 34 A2), then radio A will not receive the packet as its MAC address will not match the specified destination address.

6.1.3 Time Division Multiple Access Network

For a more intelligent network, a Time Division Multiple Access Network (TDMA) system can be implemented. In this system various radios transmit data to a SAP during an assigned time interval. The system is synchronous so that only one radio is transmitting at a time and has full access to the SAP's bandwidth. In a TDMA network, each radio must store its data for the amount of time between its transmissions or bursts. A typical format for data passing through a SAP is shown in Figure 5. A frame consists of arriving bursts from remote radios and each frame is then divided into multiple time slots. The bursts can be of varying lengths and can be longer for heavy-traffic stations. To prevent overlaps, guard intervals can be inserted to absorb small timing errors in burst arrivals.

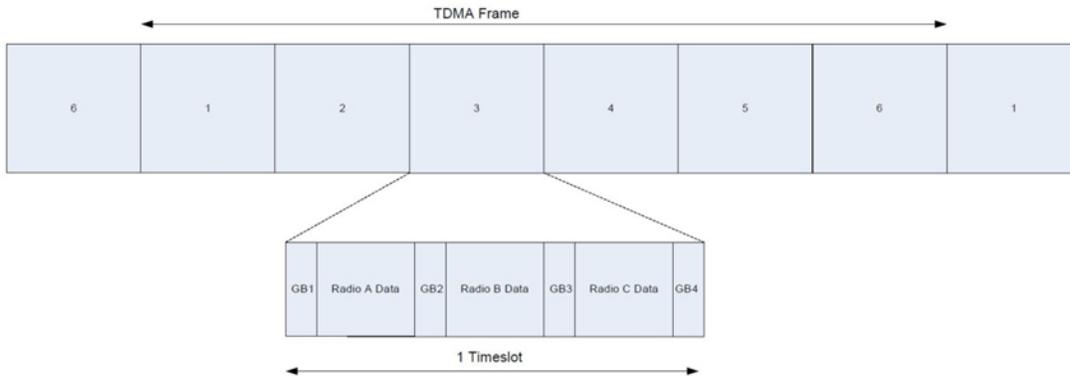


Figure 5: Quad Intervals

Example:

- SAP sends broadcast packet which includes a sync pulse
- Remote radios hear the sync pulse and join the session
- Radio A transmits during time interval $t = 1$
- Radio B transmits during time interval $t = 2$
- Radio N transmits during time interval $t = N - 1$

This type of implementation requires careful planning and should allow enough time for retries if necessary. When full duplex is enabled, the radio which initiated the session (SAP) will transmit during the even numbered hops and the remote radios will transmit only during odd numbered hops.

7 APPENDIX II: SYNC-TO-CHANNEL

Note: Information furnished by Laird in this specification is believed to be accurate. Devices sold by Laird are covered by the warranty and patent indemnification provisions appearing in its Terms of Sale only. Laird makes no warranty, express, statutory, and implied or by description, regarding the information set forth herein. Laird reserves the right to change specifications at any time and without notice. Laird products are intended for use in normal commercial applications. Applications requiring extended temperature range or unusual environmental requirements such as military, medical life-support or life-sustaining equipment are specifically not recommended without additional testing for such application.

Note: For a period of one (1) year from the date of purchase, Laird warrants the transceiver against defects in materials and workmanship. Laird will not honor this warranty (and this warranty will be automatically void) if there has been any: (1) Tampering, signs of tampering, or opening the transceiver's case. (2) Use of AC power adapters and cables other than those originally supplied with the transceivers. (3) Repair or attempt to repair by anyone other than a Laird authorized technician. This warranty does not cover and Laird will not be liable for, any damage or failure caused by misuse, abuse, acts of God, accidents, electrical irregularity, or other causes beyond Laird control, or claim by other than the original purchaser.

7.1 Sync to Channel - What is it and do I need to use it?

Laird uses frequency hopping protocol with a fixed pseudo-random hopping sequence on our transceivers. This protocol yields superior interference rejection and multipath immunity. The server sends timing beacons on a regular interval and clients hear these beacons and synchronize their hopping to the server.

Though servers cannot send packets to each other, they can hear the timing beacons sent out by other servers. Normally, the servers ignore these beacons. However, when Sync to Channel is enabled, and a specific server is designated as the synchronization master, the other servers will listen for the beacons from the master server and then synchronize their hop timing to that server.

Why is this important? If two servers (and their clients) are operating in the same area and their frequency hopping is not synchronized to each other it's possible that they might try to occupy the same frequency at the same time. In severe cases, they could interfere on every frequency, causing very slow communications.

To avoid this kind of interference, collocated servers can use Sync to Channel. Sync to Channel synchronizes frequency hop timing between these servers so that they never occupy the same frequency simultaneously.

To use Sync to Channel, you should select one server (preferably the most centrally located server) to be the "Hop Master." This server should be programmed to a numerically low RF Channel Number and should have Sync-to-Channel **disabled**. All other servers in the area should have Sync to Channel **enabled** and have their Sync-Channel set to the RF Channel Number of the server chosen as the Hop Master. Preferably, if a server is outside of the range of the Hop Master Server it can have its Sync Channel set to the RF Channel Number of another server (with a lower RF Channel Number than its own) that is in range of, and synchronized to, the Hop Master server.

The following rules apply to Sync-to-Channel:

1. One server should perform the function of Hop Master.
2. The Hop Master server should have its RF Channel Number set to a numerically low value and should have Sync to Channel **disabled**.
3. It is preferable that the Hop Master Server be centrally located.
4. All other Servers in the collocated system (servers that are being synchronized to the Hop Master server) should have Sync to Channel **enabled**.
5. All other servers in the collocated system should have their Sync Channel set to a value lower than their RF Channel Number.
6. All servers, including the Hop Master server, should have their RF Channel Numbers separated by a minimum of 4-5 Channels (i.e. Server 1, Hop Master = RF Ch 16, Server 2 = RF Ch 21, Server 3 = RF Ch 26...) to avoid inter-channel interference between the radios as they hop through their pseudo-random hopping sequence.
7. If the servers to be synchronized are in range of the Hop Master server, it is preferable that their Sync Channel be set to the RF Channel Number of the Hop Master server.
8. If some of the servers to be synchronized are outside of the range of the Hop Master server, their Sync Channel should be set to the RF Channel Number of a server (with a lower RF Channel Number than its own) that is in range of, and synchronized to, the Hop Master server.

All collocated servers must be programmed to the same channel set. There are 32 available channels for the CL4490-1000, as shown in [Table 8](#).

Table 8: RF Channels for AC4490

Channel Set	RF Channel Number Range (0x40)	Frequency Details & Regulatory Requirements	Countries
0 (AC4490 - 1x1 AC4490 - 200)	0x00 - 0x0F	902 - 915 MHz (26 hop bins)	US / Canada
1 (AC4490 - 1x1 AC4490-200 AC4490 - 1000)	0x10 - 0x2F	902 - 928 MHz (50 hop bins)	US / Canada

2 (AC4490 - 1x1 AC4490 - 200 AC4490 - 1000)	0x30 - 0x37	915 - 928 MHz (22 hop bins)	Australia (-1x1/-200/-1000)
---	-------------	-----------------------------	-----------------------------

What happens if you don't enable Sync to Channel and you have collocated servers? There are good odds that you will see a decrease in throughput due to the systems trying to occupy the same frequency at the same time. In severe cases, you could lose communications all together depending on how much bandwidth your system requires. Due to crystal differences between the servers, you may see intermittent interference.

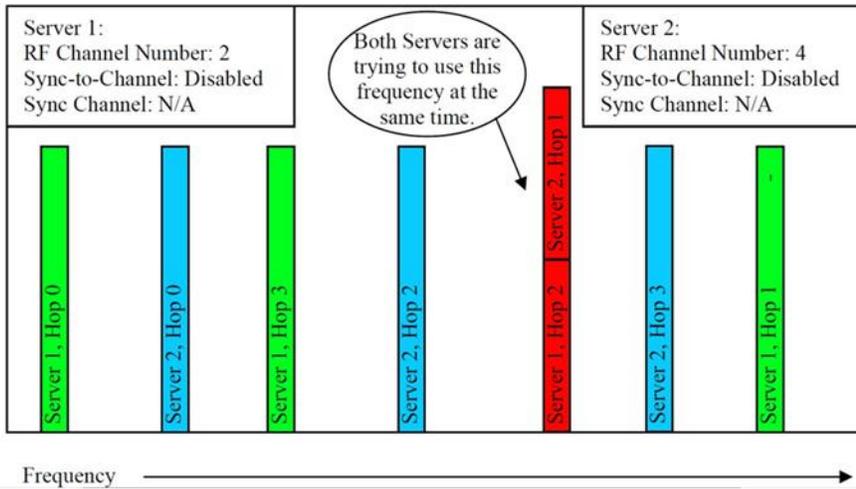


Figure 6: Two servers without Sync to Channel enabled

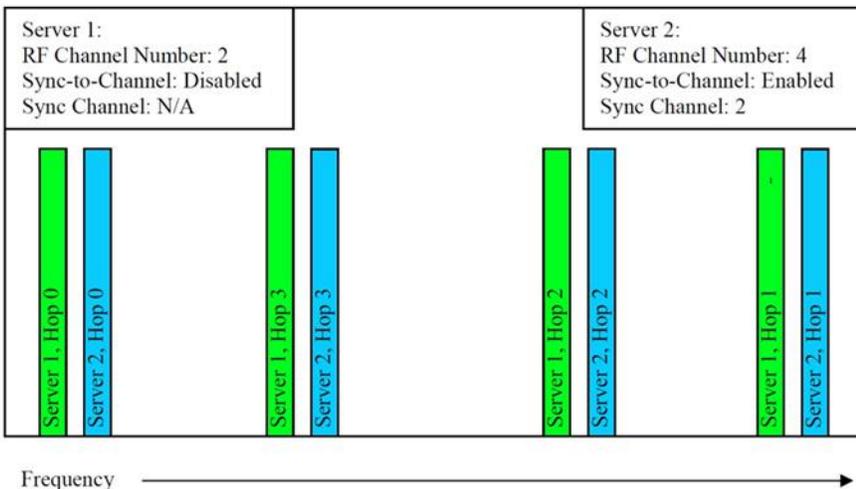


Figure 7: Two servers with Sync to Channel enabled

7.2 How do I configure Sync to Channel?

To configure Sync to Channel, you must use our OEM configuration software. To download the utility, visit [the AC4490 product page](#) and underneath the Software Downloads heading, click the **Laird Configuration Utility**. Download and run the installer file.

You will be prompted to install the software on your PC. Once the install is completed, you can open the software from **Start -> All Programs -> Laird Technologies Wireless -> Laird Technologies Config.exe**.

Note: Items 2-6 in the following list correlate to the numbered items in [Figure 8](#).

1. The software will open on the Configure tab and you will need to change to the PC Settings tab at the top of the window.

Note: RF Options mentioned in this procedure require that the “Show All Options box is selected in the Security Pane on the PC Settings tab of the Configuration Utility. To enable the Security Pane, see the *Enabling the Security Pane in the Laird Configuration Utility* accessible from the AC4490 product page (Documentation tab) on the Laird Connectivity website: <https://www.lairdconnect.com/wireless-modules/ramp-ism-modules/proprietary-900-mhz-modules/ac4490-series>

2. Select the appropriate product from the Product drop-down menu (AC4490).
3. Select the COM Port that your radio is connected to. If you are unsure, press the Find Ports button and the drop down list will be updated with available COM ports.
4. Select the baud rate that matches the baud rate that the radio is programmed to (the default baud rate for the 4490 family is 57600).
5. Verify that the COM Port selected is OPEN and that CTS Port 1 is LOW.

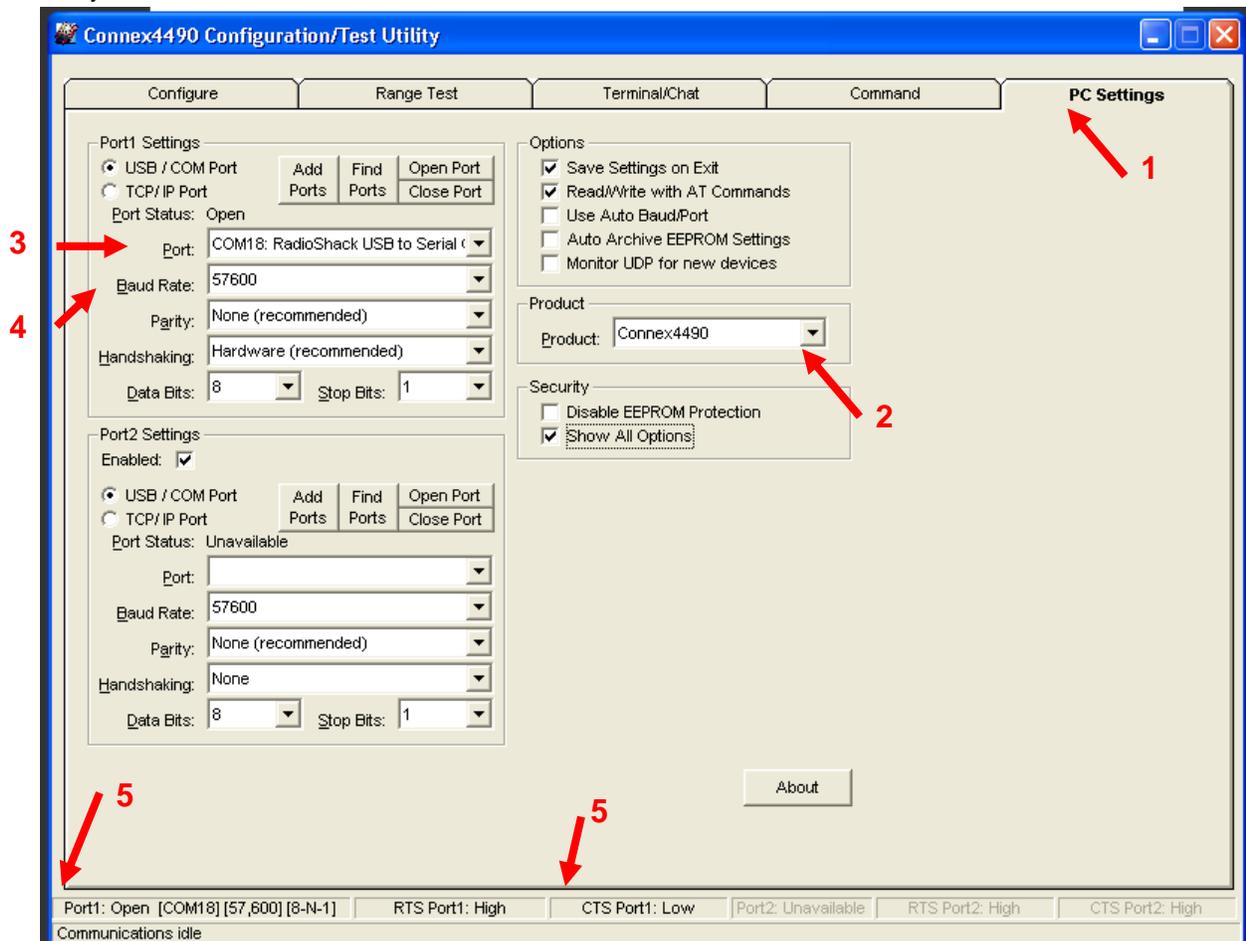


Figure 8: PC Settings tab

- Go to the Configure tab and click the **Read Radio** button at the bottom right of the screen. A message stating “Read Successful” should appear after a successful read (Figure 9).

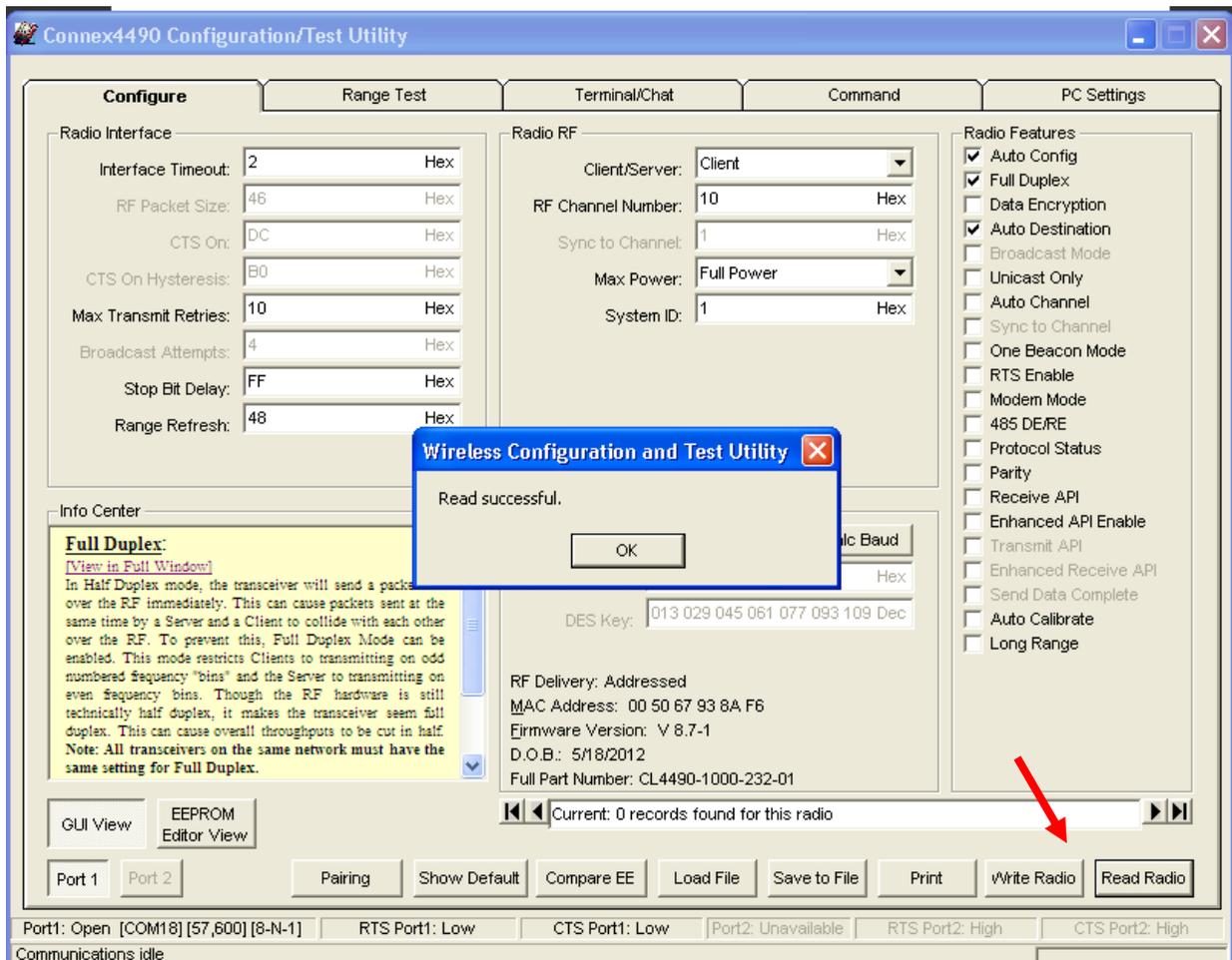


Figure 9: Configure tab - Read Successful

- To configure the Hop Master, change the Mode to **Server** and select **Broadcast Mode**. Make note of the RF Channel Number. Once the appropriate changes have been made, press the **Write Radio** button. A Write Successful prompt will appear after a successful write. Note that the values are shown using hexadecimal representation; this can be changed to decimal notation by double-clicking on the word “Hex” (it will change to “Dec”).

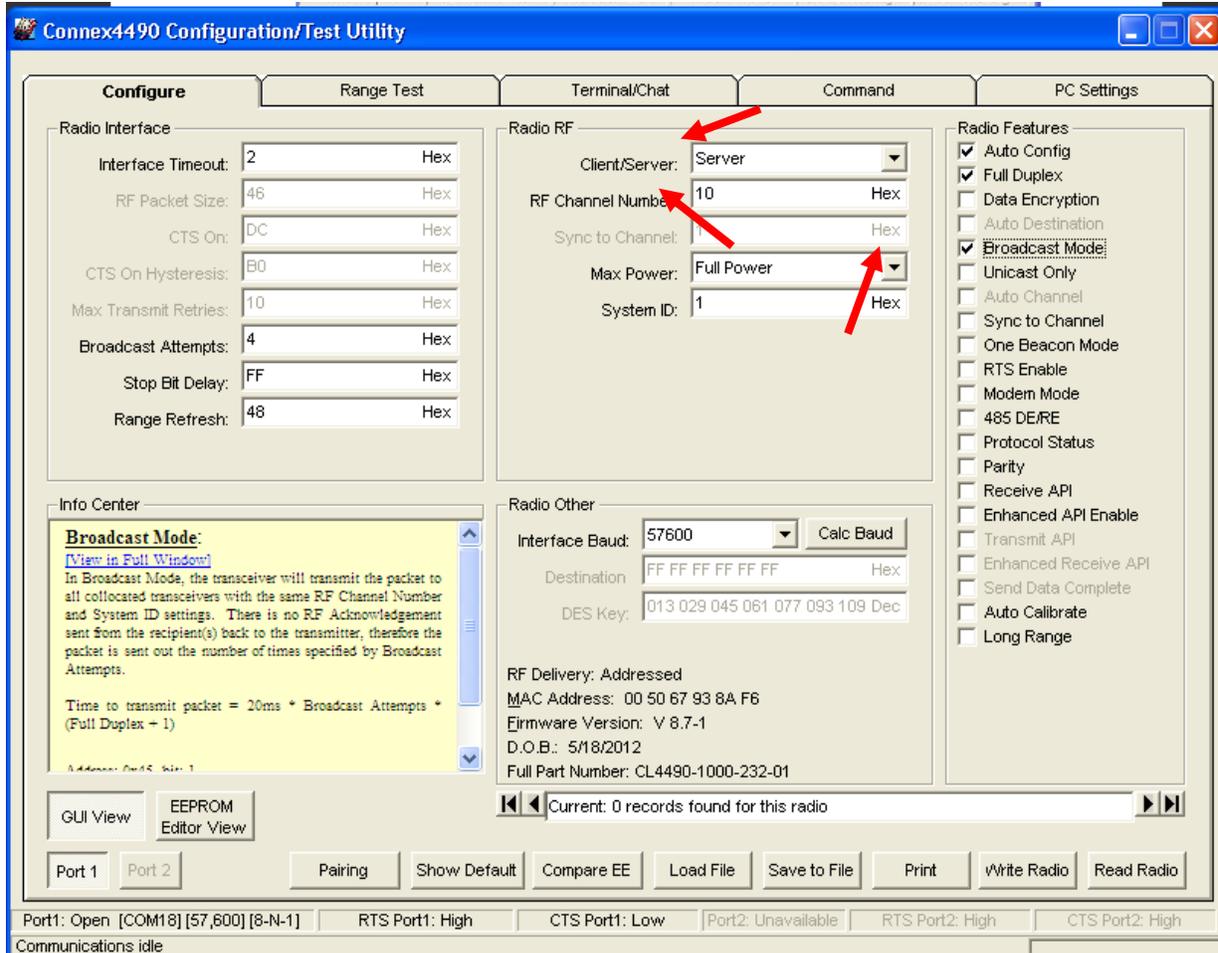


Figure 10: Hop Master settings

- Configure all clients that will communicate with the Hop Master Server as clients in Auto Destination and with the same RF Channel Number as the Hop Master Server (Figure 11), and then press the **Write Radio** button.

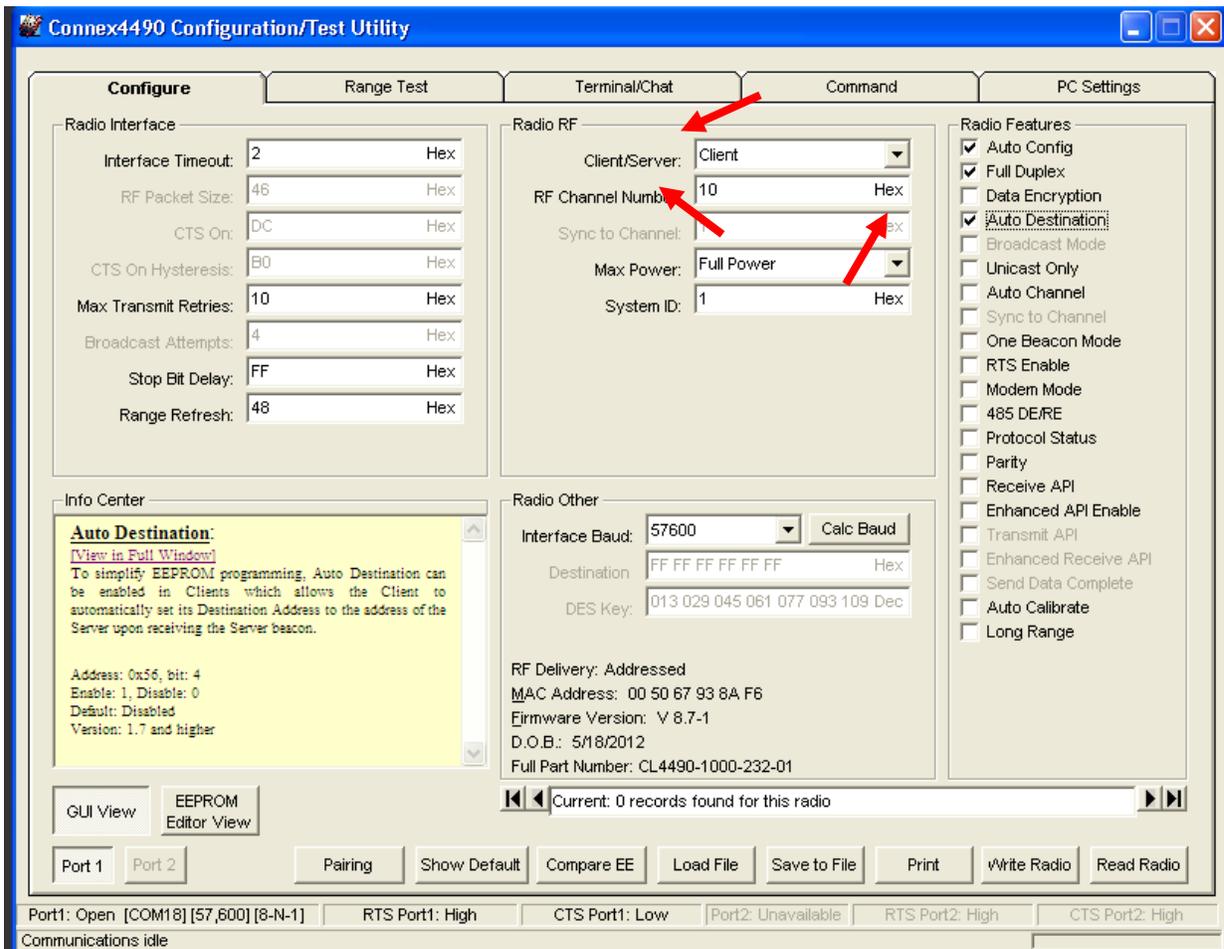


Figure 11: Client settings

- Set Server #2 as a server in **Broadcast Mode** with an RF Channel Number at least 4-5 steps above the RF Channel Number of the Hop Master. Under the Radio Features section, select the **Sync to Channel** box and in the Radio RF section, set the Sync to Channel to the RF channel of the Hop Master (Figure 12). Press the **Write Radio** button to write the changes to the radios EEPROM.

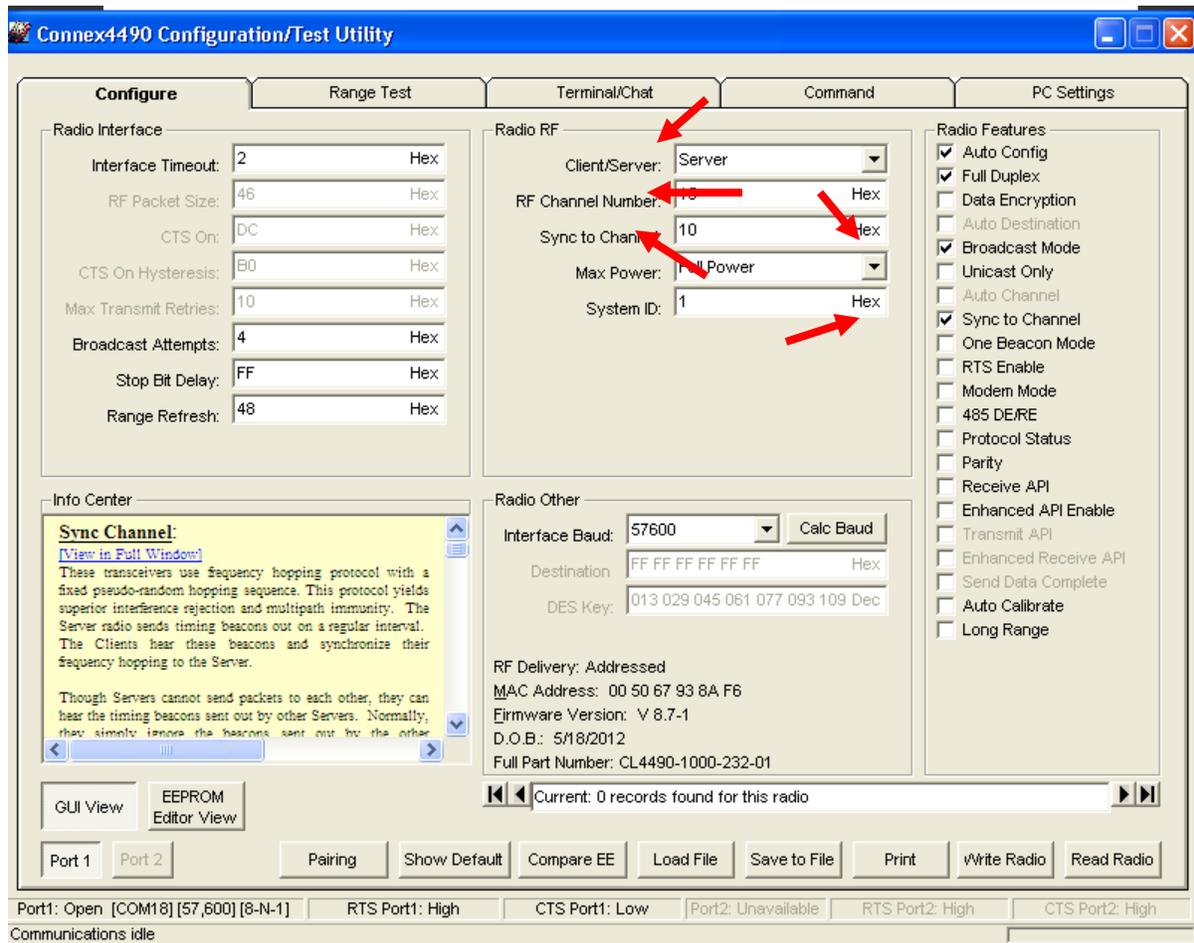


Figure 12: Server #2 settings

- Configure the clients that will communicate with Server #2 as Clients in Auto Destination and with the same RF Channel Number as Server #2 (Figure 13). Press the **Write Radio** to write the changes to the radios EEPROM.

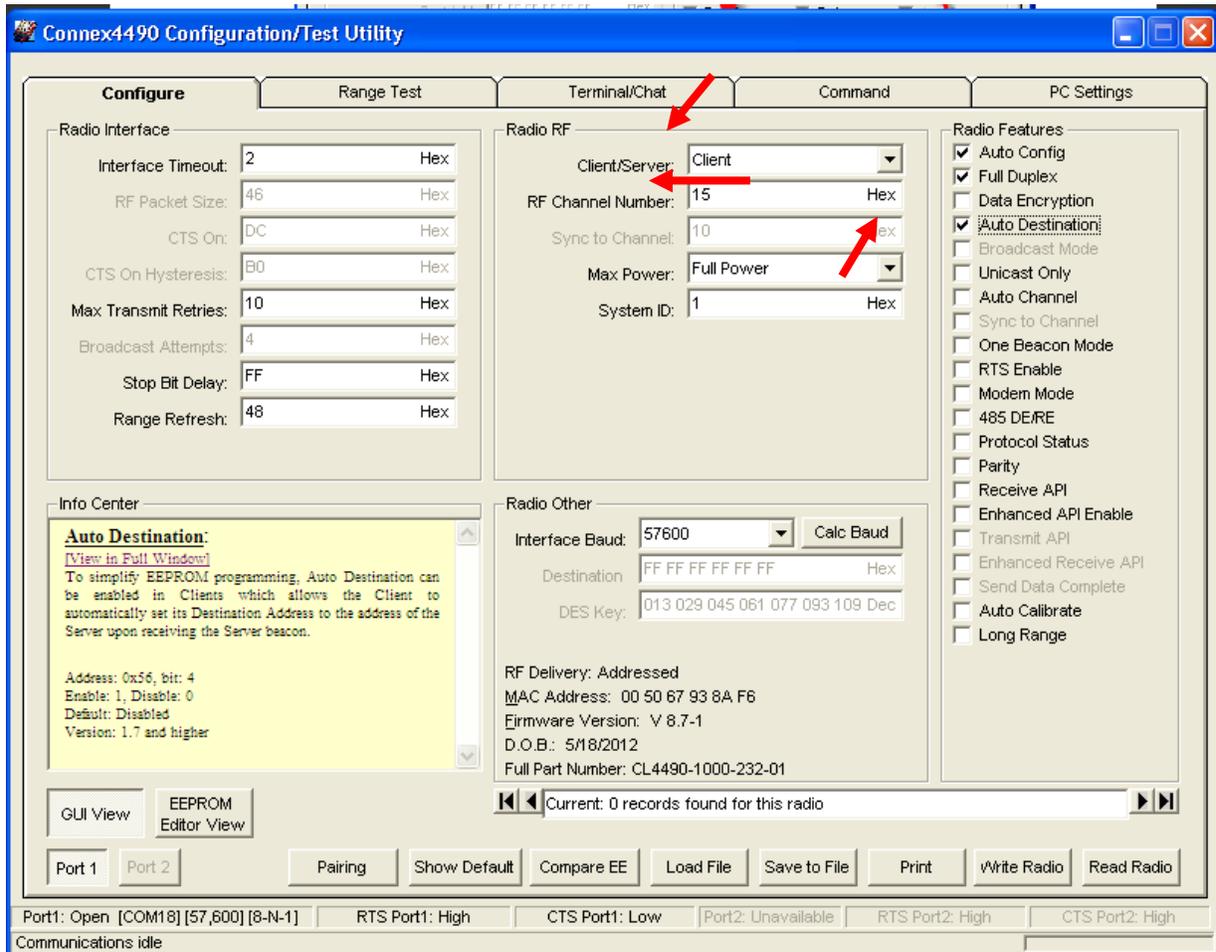


Figure 13: Client settings

- Repeat step 10 for each server that needs to synchronize to the Hop Master; if the server will not be in range of the Hop Master Server, set its Sync to Channel to the RF Channel Number of another synchronized Server that is in range of the Hop Master (make sure the RF Channel Number of the server is higher than the Sync to Channel).
- Repeat step 11 for all clients that will communicate with each server.

7.3 I've configured my radios, what's next?

Once you have configured all radios, your network should be set up similar to the one shown in Figure 14. The main server or hop master must be powered on anytime that the other servers are connected to enable them to synchronize and communicate with their clients. If a centralized network does not work and all servers are not in range of the hop master, a daisy chain network can be used as shown in Figure 15.

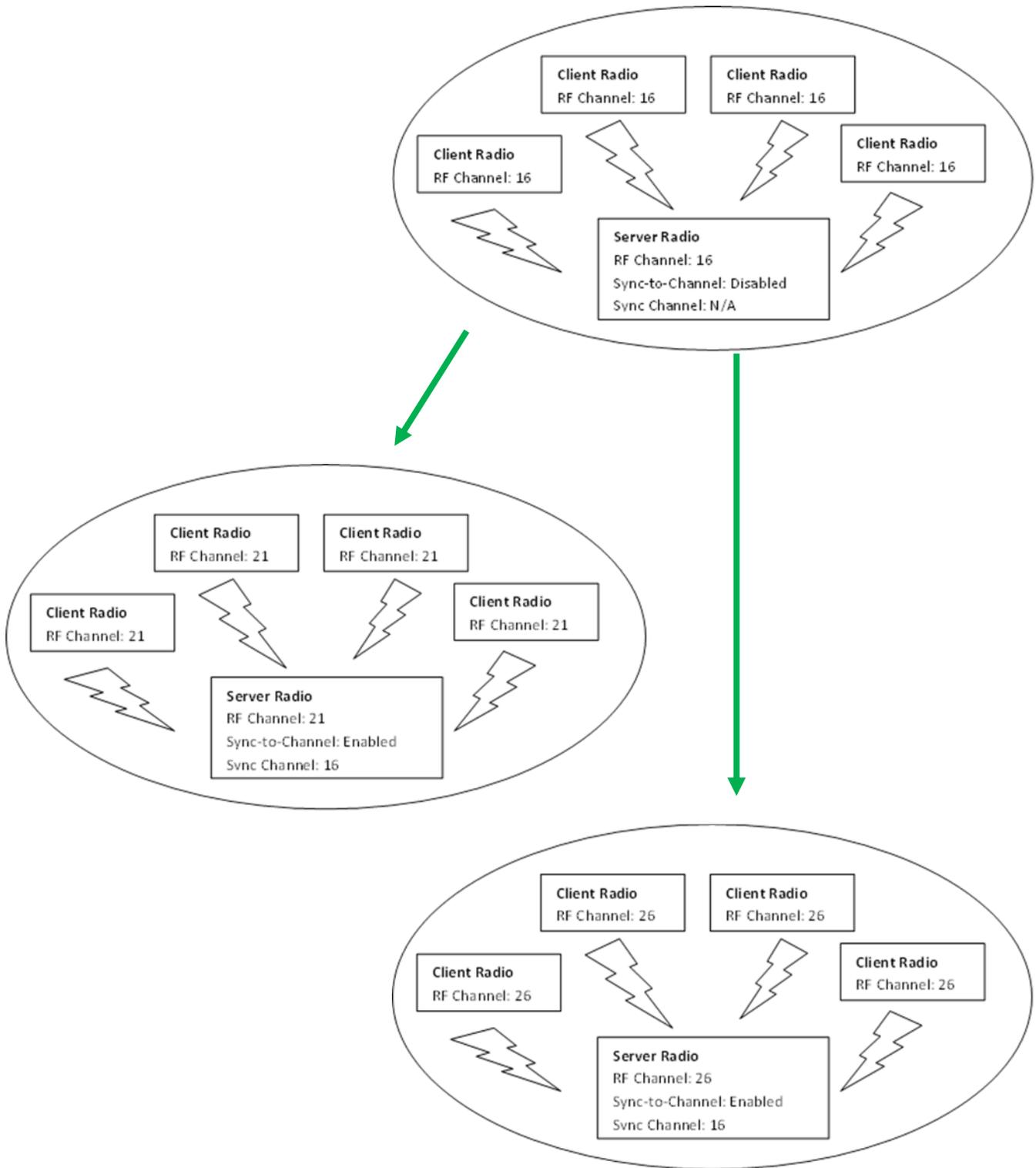


Figure 14: Sample Centralized Sync to Channel Network configuration

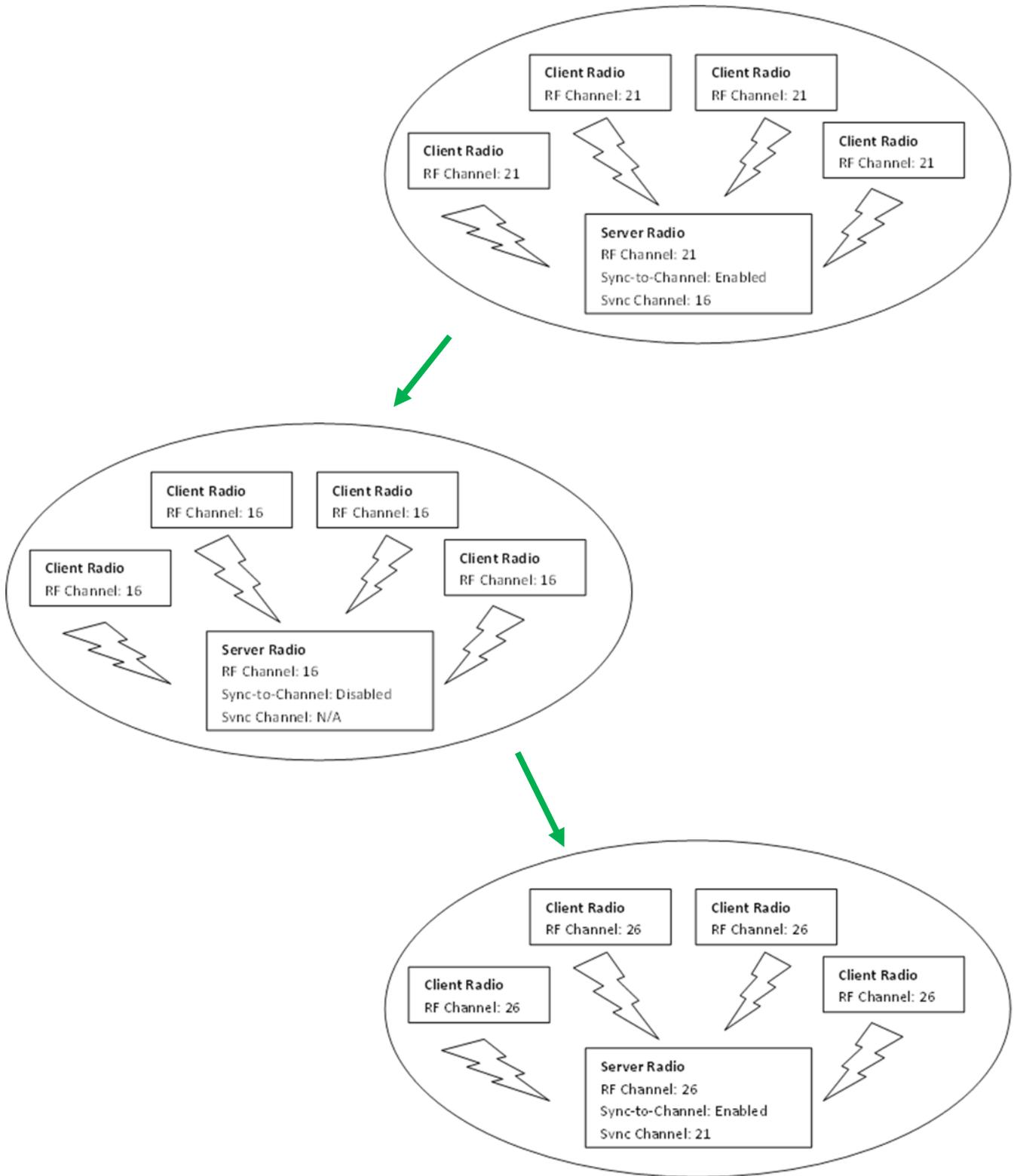


Figure 15: Sample Daisy Chain Sync to Channel Network configuration

8 RELATED DOCUMENTS AND FILES

The following additional AC4490 technical documents are available from the AC4490 product page: (Documentation and Datasheet tabs) located on the Laird website: <https://www.lairdconnect.com/ac4490>

- Product Brief
- AC4490 Datasheet
- Statement of Compliance to EU WEEE Directive and RoHS Directive
- The following downloads are also available from the RAMP Product Information tab:
- Laird Configuration Utility
- USB Drivers
- AC4490 RF Diagnostic Suite

9 ADDITIONAL ASSISTANCE

Please contact your local sales representative or our support team for further assistance:

Laird Connectivity Support Center: <https://www.lairdconnect.com/resources/support>

Email: wireless.support@lairdconnectivity.com

Phone: Americas: +1-800-492-2320

Europe: +44-1628-858-940

Hong Kong: +852-2762-4823

Web: <https://www.lairdconnect.com/products>

Note: Information contained in this document is subject to change.

© Copyright 2022 Laird Connectivity. All Rights Reserved. Any information furnished by Laird and its agents is believed to be accurate and reliable. All specifications are subject to change without notice. Responsibility for the use and application of Laird materials or products rests with the end user since Laird and its agents cannot be aware of all potential uses. Laird makes no warranties as to non-infringement nor as to the fitness, merchantability, or sustainability of any Laird materials or products for any specific or general uses. Laird, Laird Technologies, Inc., or any of its affiliates or agents shall not be liable for incidental or consequential damages of any kind. All Laird products are sold pursuant to the Laird Terms and Conditions of Sale in effect from time to time, a copy of which will be furnished upon request. When used as a trademark herein, *Laird* means Laird PLC or one or more subsidiaries of Laird PLC. Laird™, Laird Technologies™, corresponding logos, and other marks are trademarks or registered trademarks of Laird Connectivity. Other marks may be the property of third parties. Nothing herein provides a license under any Laird or any third party intellectual property right.