AR5005 EEPROM Device Configuration Guide

PRELIMINARY

Revision March 2005

ATHEROS®
COMMUNICATIONS

Notice

The information in this document has been carefully reviewed and is believed to be accurate. Nonetheless, this document is subject to change without notice, and Atheros Communications, Inc. (Atheros) assumes no responsibility for any inaccuracies that may be contained in this document, and makes no commitment to update or to keep current the contained information, or to notify a person or organization of any updates. Atheros reserves the right to make changes, at any time, in order to improve reliability, function or design and to attempt to supply the best product possible. Atheros does not represent that products described herein are free from patent infringement or from any other third party right.

No part of this document may be reproduced, adapted or transmitted in any form or by any means, electronic or mechanical, for any purpose, except as expressly set forth in a written agreement signed by Atheros. Atheros or its affiliates may have patents or pending patent applications, trademarks, copyrights, maskwork rights or other intellectual property rights that apply to the ideas, material and information expressed herein. No license to such rights is provided except as expressly set forth in a written agreement signed by Atheros.

ATHEROS MAKES NO WARRANTIES OF ANY KIND WITH REGARD TO THE CONTENT OF THIS DOCUMENT. IN NO EVENT SHALL ATHEROS BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL SPECULATORY OR CONSEQUENTIAL DAMAGES ARISING FROM THE USE OR INABILITY TO USE THIS PRODUCT OR DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBLITY OF SUCH DAMAGES. IN PARTICULAR, ATHEROS SHALL NOT HAVE LIABILITY FOR ANY HARDWARE, SOFTWARE, OR DATA TRANSMITTED OR OTHERWISE USED WITH THE PRODUCT, INCLUDING THE COSTS OF REPAIRING, REPLACING, INTEGRATING, INSTALLING OR RECOVERING SUCH HARDWARE, SOFTWARE OR DATA. ATHEROS SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE AS THEY MIGHT OTHERWISE APPLY TO THIS DOCUMENT AND TO THE IDEAS, MATERIAL AND INFORMATION EXPRESSED HEREIN.

Document Number: 984-00019-014

Revision History

Revision	Description of Changes
November 2003	AR5004 initial release. ■ EEPROM revision 4.8. ■ Support for concurrent dual 802.11a operation added. ■ PCI configuration entry updated for wake-on-WLAN support.
June 2004	AR5005 initial release for AR2413 support ■ EEPROM revision 5.0. ■ Support for eep_map = 2 added. ■ Up to four 802.11b/802.11g cal piers for eep_map = 2 added ■ 802.11a and 802.11g turbo mode specific parameters added to EEPROM header: a. switch settling time b. TxRxAtten c. rxtx_margin d. ADC_desired_size e. PGA_desired_size EEPROM revision 5.2 ■ Capabilities bits added (5.1) ■ Added enable heavy clip (5.2) ■ Version 5.2 also supports dynamic EAR ■ Checksum expansion beyond 16K
October 2004	AR5513 support added ■ EEPROM revision 4.9 (based on 4.8 branch). ■ Added phase calibration information for 11a and 11g modes ■ Dual chain calibration format supported : locations 0x400-0x7FF contain calibration data for the 2nd chain. ■ Added tx_disable_chain and rx_disable_chain masks for low cost data falcon designs
March 2005	Support for upcoming Japan regulatory changes added ■ EEPROM revision 5.3 ■ Added following flags : – en_fcc_mid – en_jap_even_u1 – en_jap_u2 – en_jap_mid

Contents

List of Tables

Preface

This document provides information on the use of the EEPROM, as well as on the device configuration information stored in it. The EEPROM is recommended for use with subsystem designs using the AR5005 chip sets. While this version of the EEPROM also supports adapters with AR5004 and AR5002 chipsets, it is recommended to use the previous generation EEPROM versions to support these adapters

About this Document

This document consists of the following chapter and appendices:

Chapter 1 **EEPROM Device Configuration**. Describes the contents stored on the EEPROM.

Appendix A **eep_map 0**—Describes the EEPROM subsystem-specific information for eep_map 0 configuration.

Appendix B **eep_map 1**—Describes the EEPROM subsystem-specific information for eep_map 1 configuration.

Appendix D **EEPROM Contents** — A sample of the file **atheros-eep.txt**.

Appendix E **EAR EEPROM**— Describes scheme for updating hardware register settings from values contained in the EEPROM.

Audience

This document is intended for Atheros customers involved with the definition, design, and implementation of station modules deploying the Atheros AR5005 chip sets.

Additional Resources

Atheros Reference Design hardware, software, and documentation contain proprietary information of Atheros Communications, Inc., and are provided under a license agreement containing restrictions on use and disclosure, and are also protected by copyright law. Reverse engineering of this hardware, software, or documentation is prohibited.

This guide assumes that the reader has studied and is familiar with the *AR5005 Sample Manufacturing Test Flow* and the *AR5005AP Access Point Reference Guide.*

1

EEPROM Device Configuration

This chapter describes the details of the device configuration information stored on the EEPROM. The configuration information stored on the EEPROM is utilized by the NDIS driver for Windows to ensure optimum performance of the wireless network interface card (NIC). The Access Point (AP) reference designs, based on the AR5311, AR5312, or AR2312 store this See the *AR5005AP Access Point Reference Guide* for more details.

Beginning with EEPROM version 4.8, facility has been provided to support an access point reference design with up to two 5 GHz transmit chains. In these cases, the entire configuration information repeats for the second chain. Only the configuration information that differs for the second chain is changed, and the rest of it remains identical to the first chain.

Unless specified, EEPROM contents for stations (STAs) is analogous to the Flash for access points (APs).

The NDIS driver loads three types of information from the EEPROM:

- EEPROM Generic Information — Generic and possibly vendor-specific information, although design and unit independent information is stored in this section. See "EEPROM Generic Information (0x00–0xBE)" on page 8 for details.
- EEPROM Subsystem Specific Information — information specific to a particular subsystem design (a new reference design or a revision). See "EEPROM Subsystem Specific Information (0xBF–0xFF)" on page 10 for details.
- EEPROM NIC Specific Information — Calibration information unique to each individual unit of a design type. See Appendix A, "eep_map 0" and Appendix B, "eep_map 1" for details on the format of this information.

  To learn more about how this calibration information is obtained for each unit, see the *AR5005 Sample Manufacturing Test Flow.*

Device Configuration Algorithm

An EEPROM layout version field is stored at location 0xC1 (see Table 1-5). The NDIS driver or the AP software uses this version number to determine how to load and interpret the device configuration information stored on the EEPROM. In general, the Atheros-provided NDIS driver employs this algorithm to configure a device such as a network interface card (NIC) with appropriate settings:

1. The NDIS device driver first looks for an **athnic.ini** file in a folder appropriate for the current operating system (most commonly **\WINNT\system32\drivers**).

   If an **athnic.ini** file is found, the settings in that file are used to configure the device. The EEPROM contents are ignored in this case.

2. If the file **athnic.ini** is not found, the NDIS device driver looks at the device EEPROM to determine whether valid configuration data exists, then:

   a. It reads the EEPROM layout version number from the location 0xC1.

   b. It checks whether the magic word (location 3D=5AA5) is present:

      - If the magic word is not successfully read, the EEPROM contents are not loaded and default driver settings are used to configure the device instead.
      - If the magic word is successfully read, it reads the EEPROM data and computes the checksum based on the version number.

   c. It reads the checksum from location 0xC0 (see Table 1-5):

      - If the two checksums do not match, the EEPROM contents are not used and the default driver settings are used.
      - If the computed checksum matches the checksum read from location 0xC0, the EEPROM data is deemed valid. It loads the EEPROM contents and interprets them in accordance with the appropriate EEPROM layout for that version number.

3. Default settings stored within the NDIS driver do not provide valid configuration data.

EEPROM Layout Changes from AR5004

The layout of information stored on the EEPROM has been augmented for the AR5005. The major changes in the new EEPROM layout include:

- The major version is assigned to 5 for this AR5005 release. It was assigned to 4 for the AR5004 release.
- Three types of power calibration data formats are supported in EEPROM 5.0. The AR2413 has a further improved power control scheme that provides increased accuracy. The new calibration data format is described in Appendix C. EEPROM 5.0 continues to support the legacy calibration data format for the products using the AR5111/AR2111 and AR5112/AR2112. The eep_layout_map parameter identifies the format of cal data present.

Determining Concepts

This section discusses concepts used to determine what information is stored on the EEPROM, and how to use that information. These concepts include:

- "Piecewise Linear Abstraction"
- "Frequency Piers"
- "Target Power"
- "Target Power Frequencies"
- "Band Edges"
- "Conformance Testing Limits"
- "Country or Domain Code"
- "Support of Multiple Regulatory Domains"
- "Operating Power Algorithm"

Piecewise Linear Abstraction

The piecewise linear abstraction (PLA) design captures general dependence accurately if it is sampled at appropriate turning points (TPs) and linearly interpolated between the TPs. Figure 1-1 demonstrates how the PLA scheme maintains general dependence accuracy if appropriate TPs are selected.

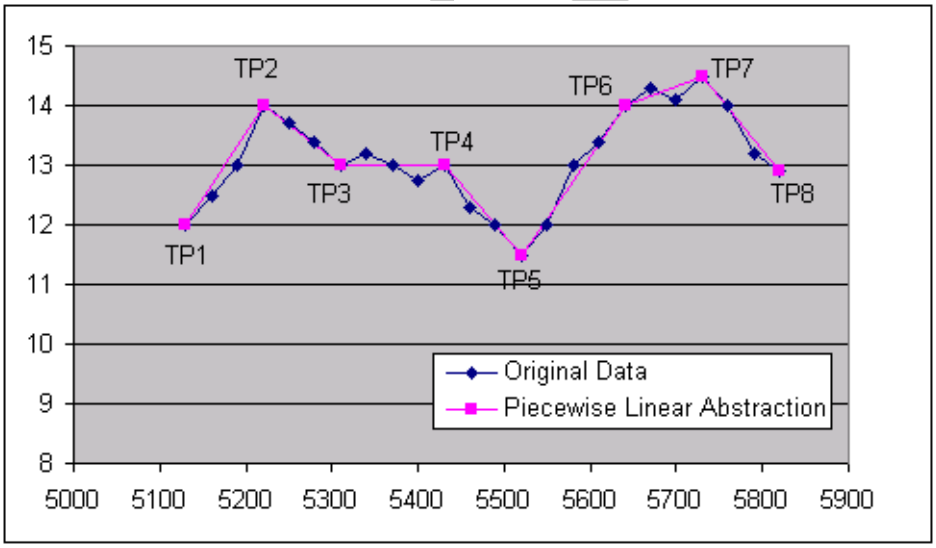

Figure 1-1. **PLA Scheme Applied to a General Dependence**

NOTE: Because these dependences arise from statistical variations of parameters and their interplay, the exact nature of dependences can vary from card to card. As a result, the TPs may turn out to be at different locations for each card. Therefore, fixing the locations of sampling points will, in general, not preserve the data with high accuracy for all cards.

A high degree of accuracy for each card can be preserved if the locations of the TPs are also stored with the sampled values for a sufficient number of TPs. This is the central theme in the approach adopted by Atheros to store any NIC-specific or subsystem-specific calibration information on the EEPROM. This enables card manufacturers to deliver the highest level of performance accuracy tailored for each individual card.

Frequency Piers

The PLA concept applies to any kind of dependence. Figure 1-2 demonstrates how the PLA scheme can extend to a set of curves to accurately reproduce an original dataset by sampling a few TPs, if the sampling points are chosen well.

When the PLA scheme is applied to the dataset obtained by measuring the output power over a range of frequencies for several values of PCDACs, the TPs for this family of curves are referred to as the frequency piers.

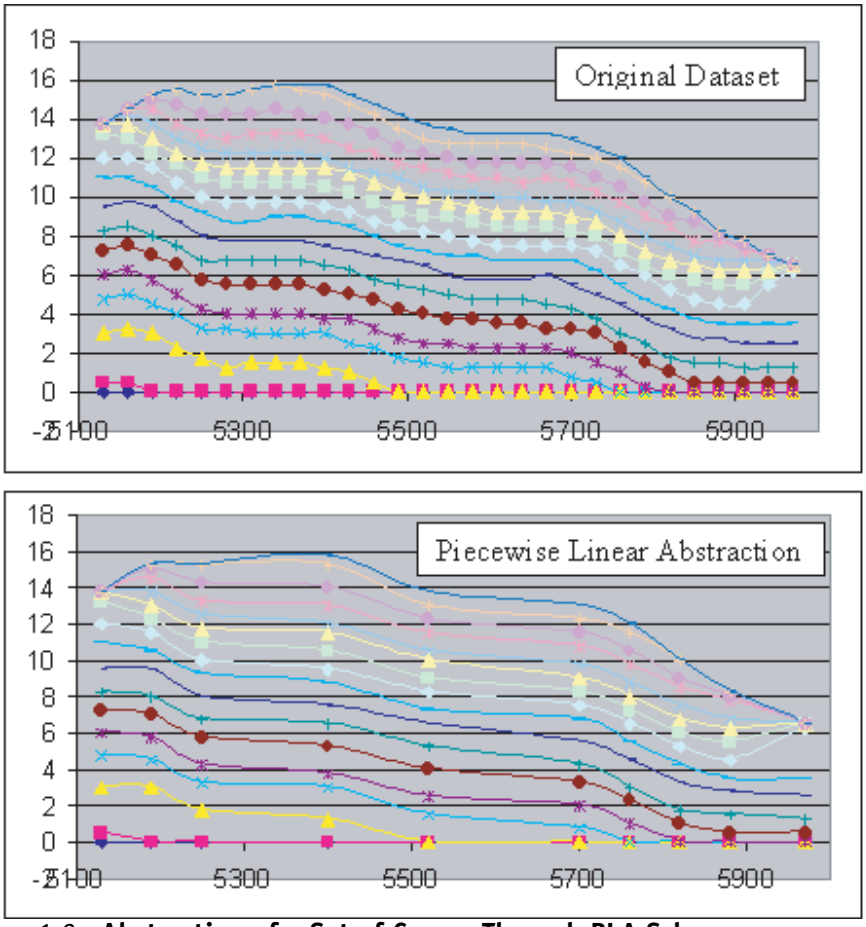

Figure 1-2. **Abstraction of a Set of Curves Through PLA Scheme**

For the AR5005 chipset, the level of output power is controlled by the power control for the digital to analog converter (PCDAC) values. The nature of radio frequency (RF) circuits is, in general, quite sensitive to the impedance match between various stages. The response of external components, such as the power amplifier (PA), and of passive elements, typically depend on the frequency. Thus output power dependence versus PCDAC over the entire range of channels must be conveyed accurately to the driver for each unit.

As part of the manufacturing calibration, dependence of output power over frequency is measured over 4.9 to 5.85 GHz (for 802.11a mode, or the range the customer specified using FORCE_PIERS_LIST) for several values of PCDACs, and the ten most important TPs (frequency piers) are automatically computed for each card. The calibration mode is the most accurate, as it computes custom frequency piers for each network identification card (NIC).

However, if the manufacturing process and the subsystem design are robust enough, then card after card, the power measurements (recorded in the file **cal_AR5211_power.log** after each calibration run) turn out to be nearly identical (within 0.5 dB at all channels for all PCDACs).

An alternative FORCE_PIERS mode exists, which speeds up the calibration process. In the FORCE_PIERS mode, the 10 frequency piers are determined from the pilot runs, and the list of piers is specified as FORCE_PIERS_LIST in the **calSetup_XXXX.txt** files. The power measurements are performed only for these channels and stored on the EEPROM as the calibration data. Because measurements are not performed at all frequencies from 4.9 GHz to 5.85 GHz in steps of 10 MHz, the FORCE_PIERS mode runs faster. This speed enhancement comes at the cost of accuracy, therefore, a thorough evaluation should be made before deciding whether to use FORCE_PIERS.

Target Power

For the 802.11a mode (and as needed for other modes of operation), the maximum power that satisfies all IEEE specification requirements (e.g., spectral mask) and performance criteria (that is, < 10% packet error rate (PER)) is determined for a particular subsystem design through a pilot run over a statistical ensemble of NICs. This power is referred to as target power.

This measurement is not performed individually for each card, and the target power does not take into account the regulatory domain's limited power. Target power is an indication of raw capability of a particular card type, regardless of the regulatory domain where the cards are used.

Generally, a unique target power exists for each rate. However, for all Atheros Reference Designs, the rates of 6–24 Mbps have been found to have the same target power (spectral mask limited) and the rates 36, 48, and 54 Mbps have their own target power (PER-limited) over the entire frequency range.

Target Power Frequencies

The target power for all rates or rate groups has a dependence on the frequency. The EEPROM has provisions for the vendors to specify up to eight TPs of this dependence under the PLA scheme. These TPs, used for conveying the target power information, are referred to as target power frequencies, and are determined by the vendor after analyzing data gathered during the pilot run and conveyed to the calibration routine by the file **calTargetPower.txt**. Refer to the *AR5005 Sample Manufacturing Test Flow* for more information.

Band Edges

In a regulatory domain, the frequency bands open for public infrastructure are typically interspersed with the restricted bands for military or government use. The extreme operating channels in the open frequency bands are referred to as band edges. For example, the band edges in the FCC open band 5.15 GHz–5.35 GHz are 5.18 GHz and 5.32 GHz.

Special consideration is needed to determine the transmit power at the band edges to ensure compliance with the regulations in the adjacent restricted frequency band. Channels that fall within the open band (between the band edges) do not require this special consideration.

Conformance Testing Limits

Significant similarities exist in the boundaries of the open bands in several regulatory domains, because of how the frequency bands have been opened for allocation in the 5 GHz and 2.4 GHz range worldwide. Therefore several regulatory domains exist with identical sets of band edges. Conformance testing limits (CTLs) leverage this overlap, delivering a simplified mechanism supporting several regulatory domains in manufacturing. It is essential to convey band edge maximum power information to the driver using the EEPROM for all regulatory domains where the NIC is targeted. This data is subsystem design-specific and gathered during the pilot run. If a regulatory domain-based approach is used to store this information on the EEPROM, considerable redundancy exists for domains with overlapping band edges.

To alleviate this redundancy and maximize the number of regulatory domains that can be supported by a NIC, a CTL is defined to be a unique set of band edges and adjacent restricted band regulations. For example:

- If RD1 and RD2 have a permitted band from 5180–5240 MHz and the same set of restrictions for frequencies below 5180 and frequencies higher than 5240, then RD1 and RD2 can belong to the same CTL.
- If RD3 also has a permitted band from 5180–5240 MHz, but tolerates higher power for frequencies below 5180 (i.e., 2 dB higher tolerance at 5160), then RD3 can not belong to the same CTL as RD1 and RD2.
- If RD4 does not permit any transmission in 5180–5240 MHz, but has a permitted band from 5400–5520 MHz, then RD4 can belong to the same CTL as RD1 and RD2 because the two bands do not overlap. The CTL would now contain band edges 5180, 5240, 5400, and 5520. This is possible because the software contains a list of legal channels in each regulatory domain, so for RD4, it will not even look at 5180 and 5240.

Note that band edges always appear in pairs (lower and upper band edges). Regulatory domains of the band edge pairs that appear in a CTL belong to that CTL. The CTL may contain additional band edge pairs, thus providing data for one CTL enables support for all regulatory domains belonging to that CTL. Up to 32 CTLs are supported in the EEPROM layout.

EEPROM version 3.3 introduced non-edge flags to the CTLs: one for each CTL frequency. These are 1-bit flags, where 0 indicates that the CTL freq is a band edge, and 1 indicates that the CTL frequency is not a band edge, but an in-band frequency. Band edges always appear in pairs and mark the boundaries of permitted bands. In some cases, the regulatory stipulations imposed outside of this band may restrict power output at not only the band edges, but also for some channels within the band. It then becomes necessary to specify limits on these in-band channels as well, for that CTL. The non-edge flags are introduced to handle such cases. The design to use these flags is:

- All frequencies specified in a CTL should be arranged in ascending order.
- In-band channels are permitted only between starting and ending band edges (i.e., the first and last frequency in a CTL must be a band edge).
- Within a band, every odd in-band frequency marks the beginning of the channel range to apply the corresponding CTL limit to. This range goes up to and includes the next even in-band channel. It is permitted to specify an odd number of in-band frequencies between a pair of band edges, in which case the CTL limit for the last in-band frequency applies to channels ranging from the in-band frequency up to, but excluding, the ending band edge.

These examples demonstrate how in-band frequency can be used.

Example 1

A band exists from 5400–5600 MHz. Out of band regulations require the power be limited to 12 dBm at the band edges, 13 dBm for 5420-5460, and 12.5 dBm for 5540-5580 MHz. Table 1-1 on page 7 demonstrates how to convey this information using the non-edge flags.

Table 1-1. **Non-Edge Flag Usage in CTLs (Example 1)**

	Band Edge	In-band Freq.	In-band Freq.	In-band Freq.	In-band Freq.	Band Edge	Next Band
CTL Freq.	5400	5420	5460	5540	5580	5600	...
CTL Limit	12	13	13	12.5	12.5	12	...
Non-edge Flag	0	1	1	1	1	0	...

Example 2

A band exists from 5400–5500 MHz, out of band regulations require the power be limited to 13 dBm at starting band edge (5400), 15 dBm for 5420–5480 MHz, and 12 dBm at the ending band edge (5500). Table 1-2 demonstrates how to convey this information using the non-edge flags.

Table 1-2. **Non-Edge Flag Usage in CTLs (Example 2)**

	Previous Band	Band Edge	In-band Freq.	Band Edge	Next Band
CTL Freq.	...	5400	5420	5500	...
CTL Limit	...	13	15	12	...
Non-edge Flag	...	0	1	0	...

Country or Domain Code

A unique 12-bit code identifies the intended country or domain, of operation/sale. This value is stored in location 0xBF of the EEPROM. The NDIS driver or AP software uses this code with the Country Code Selector (CCS) and worldwide roaming (WWR) flags to determine the current operating region and overlay the appropriate regulatory domain requirements on top of the target power and the band edge maximum power data. See the support bulletin *Worldwide Roaming Design Specification* for details on how this information is used.

Support of Multiple Regulatory Domains

The following information is coded in the driver to allow support of multiple regulatory domains:

- A mapping of each country code to a regulatory domain
- An association of all regulatory domains to the appropriate CTLs
- All allowed channels and the maximum legal power limits in all regulatory domains

It is important to program a comprehensive set of CTLs in the EEPROM at manufacturing calibration. Thus supporting new frequency allocations in various countries (domains), or changes in regulations in existing regulatory domains, becomes possible through a software release of the NDIS driver or AP software update with the NICs already deployed in the field.

Operating Power Algorithm

The NDIS driver or AP software makes use of information stored on the EEPROM to determine the maximum transmit power for a given channel using the algorithm:

1. Read the country code from location 0xBF in EEPROM.
2. Obtain a list of permitted channels for this country from the driver's regulatory domain table. If the current channel does not appear in the list of permitted channels, no transmission is initiated at this channel.
3. Reconstruct the PCDAC-to-dBm table for the current channel from the calibration data sampled at the frequency piers stored in the EEPROM (see Groups 1-4 listed respectively for eep_mode 0 in Table A-1, Table A-2, Table A-4, and Table A-5, and Groups 1-4 listed respectively for eep_mode 1 in Table B-1, Table B-2, Table B-3, and Table B-4) interpolating as appropriate under the PLA scheme. Program this 64-entry table into MAC/Baseband Processor chip.
4. Obtain the target power for each rate at the current channel from the data stored on the EEPROM (see Groups 5-7 listed respectively for eep_mode 0 in Table A-6, Table A-7, and Table A-8, and Groups 5-7 listed respectively for eep_mode 1 in Table B-5, Table B-6, and Table B-7).
5. The driver determines the CTL for this country code and retrieves data for this CTL from EEPROM (see Group 8 for eep_mode 0 listed in Table A-9 and Group 8 for eep_mode 1 in Table B-8). If the current channel is determined to be a band edge in this CTL, obtain band edge maximum power at this channel.
6. The driver determines the current channel local regulatory power limit.
7. Compute the minimum of the target power, band edge max power, and local regulatory power limit at the current channel values for each rate and program the max power for all supported rates into the MAC/Baseband processor chip.

Description of EEPROM Locations

Three types of information are stored on the EEPROM:

- Generic
- Subsystem-specific
- NIC-specific

Groups of various EEPROM locations are described and appropriately categorized in this section.

EEPROM Generic Information (0x00–0xBE)

Generic information, such as vendor-specific card information structure (CIS), tuples information, Vendor ID, Device ID, Subvendor ID, and Subsystem ID, constitutes the bulk of this section (see Table 1-3 and Table 1-4). Information stored in this section is generic for each design and does not need to be measured for each design. A sample file **atheros-eep.txt** is supplied by Atheros that contains the default data to be programmed in these locations. Atheros partners should modify this file for their needs.

Table 1-3. **EEPROM Generic Information (0x00–0xBE)**

Locations	Description	Remark
0x00–0x3F	PCI configuration data	See Table 1-4
0x40–0xAF	Card Information Structure	CIS tuples
0xB0–0xBE	Vendor OEM Information	Reserved for use by Atheros partners

See Appendix D for a sample **atheros-eep.txt** file, which describes the contents of each location. Some key locations in the PCI configuration data are further detailed in Table 1-4.

Table 1-4. **Key PCI Configuration Locations**

Locations	Description	Remark
0x00	Device ID	Identifies the chip set: ■ Device ID 0x0010 = AR5210 ■ Device ID 0x0012 = AR5211 ■ Device ID 0x0011 = AR5311 ■ Device ID 0x0013 = AR5212 Value in **atheros-eep.txt** is overridden by the actual Device ID read from the chip register.
0x01	Vendor ID	Should remain fixed at Atheros assigned value 0x168C, regardless of partner implementation.
0x07	Subsystem ID[1]	Should reflect the particular NIC hardware design and be appropriately modified by Atheros partners to reflect design revisions and new designs. The value in **atheros-eep.txt** is overridden by the SUBSYSTEM_ID value specified in **calSetup.txt**. (Refer to the *AR5005 Sample Manufacturing Test Flow.*)
0x08	Subsystem Vendor ID[1]	PCI-SIG assigned ID to the Atheros partner. The value in **atheros-eep.txt** is overridden by the SUB_VENDOR_ID value specified in **calSetup.txt**
0x0C	PM_CAP	If wake-on-wlan feature is enabled for the client card, a value of 0xF9C2 is programmed in this location, otherwise, a value of 0x01C2
0x0F	Clockrun enable and RFSilent	■ Bit 0 clockrun enable (0 = disable, 1 = enable) ■ Bit 1 RFSILENT_POLARITY (0 = rfsilent on low, 1 = rfsilent on high) ■ Bits 4:2 RFSILENT_GPIO_SEL (select which GPIO (0–5) to use)
0x1B	End of EAR/ Checksum location lower 16 bits	If the EEPROM contents extend beyond 16K, this location specifies the lower 16 bits of the EEPROM location where EEPROM and hence checksum ends (see below)
0x1C	End of EAR/ Checksum location upper 12 bits and EEPROM size	■ [15:5] End of EAR/Checksum location upper 12 bits ■ [4:0] EEPROM size End of EAR/Checksum selects an EEPROM location — valid values are 0x00000C0 to 0x0080000 EEPROM size values are calculated as 2 ^ (EEPROM size + 9) — valid values are 1 to 11 (1MB) A value of 0 in 0x1C defaults EEPROM size to 2kB and End of Ear/ Checksum to 0x0000400
0x1D–0x1F	MAC ID	■ 0x1D = MacID[15:0] ■ 0x1E = MacID[31:16] ■ 0x1F = MacID[47:32]
0x3D	EEPROM Magic Word	0x5AA5 The driver attempts to read this value to infer a valid programmed EEPROM

Table 1-4. **Key PCI Configuration Locations (continued)**

Locations	Description	Remark
0x3E	KEY_TABLE_RD _PROTECT[0]	A single bit [0] at this location controls the access to the key cache registers on AR5211. Access Control: ■ 0 = Read/Write ■ 1 = Write
0x3F	EEPROM Protection[15:0]	■ Bits 1:0 protect EEPROM locations 0x00–0x1F ■ Bits 3:2 protect EEPROM locations 0x20–0x3F ■ Bits 5:4 protect EEPROM locations 0x40–0x7F ■ Bits 7:6 protect EEPROM locations 0x80–0xBF ■ Bits 9:8 protect EEPROM locations 0xC0–0xFF ■ Bits 11:10 protect EEPROM locations 0x100–0x1FF ■ Bits 13:12 protect EEPROM locations 0x200–0x2FF ■ Bits 15:14 protect EEPROM locations 0x300–0x3FF Access Control: ■ 00 = Read/Write ■ 01 = Write ■ 10 = Read ■ 11 = No Access

[1] Both the Subsystem ID and Subsystem Vendor ID must be populated for Microsoft WHQL submission. Visit the following URL to learn more about using PCI IDs with Windows operating system: http://www.microsoft.com/HWDEV/pci/pciidspec.htm.

EEPROM Subsystem Specific Information (0xBF–0xFF)

These hardware settings are subsystem design specific and span locations 0xBF–0xFF of the EEPROM, as summarized in Table 1-5. This information is typically obtained through a pilot run on a statistical ensemble of NICs or APs of this subsystem type. All manufactured NICs or APs of this design are expected to result in an optimum level of performance upon use of these settings by the NDIS driver or AP software. These settings are not individually measured or calibrated for each NIC or AP.

In this revision of the EEPROM layout, space has been allocated to support operation in 802.11a, 802.11b, and 802.11g modes. Care has been taken to allocate sufficient reserved space to accommodate reasonable future enhancements without requiring a significant layout change.

These values are conveyed by the Atheros partners to the manufacturing test flow by the **calSetup.txt** file to be stored onto the EEPROM. (Refer to the *AR5005 Sample Manufacturing Test Flow* document for more information.)

Based on the input value (0 or 1) of eep_map (location 0xC4), hardware settings may vary. For hardware settings based on eep_map set to 0, refer to Appendix A "eep_map 0". For hardware settings based on eep_map set to 1, refer to Appendix B "eep_map 1".

Table 1-5. **EEPROM Subsystem Design Specific Entries**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xBF	CCS	WWR	X	X	Country_or_Domain_Code											
0xC0	Checksum															
0xC1	Version (Major)				Version (Minor)											
0xC2	TD	RFK	DeviceType		5G_Turbo_2WmaxPower								Turbo2G	Gmode	Bmode	Amode
0xC3	Antenna_Gain_5G								Antenna_Gain_2.4G							
0xC4	eep_map = 0[1] eep_map = 1[2]		XR5	XR2	EAR_start_location											
0xC5	0[3]	32KHz_enable	X	X	Target_Powers_Start_Location											
0xC6	EEP_FILE_VERSION								EAR_FILE_VERSION							
0xC7	ART_BUILD_NUM						X	X	EAR_FILE_IDENTIFIER							
0xC8	Cal_Data_Start_Location												MaskForRadio1		MaskForRadio0	
0xC9	key_cache_size				X	X	enable_clip	max_QCU					disable_bursting	disable_FF	disable_AES	disable_comp
0xCA	RESERVED FOR RD FLAGS					dis_jap_odd_u1	en_jap_mid	en_jap_u2	en_jap_even_u1	en_fcc_mid	disable_rx_chain		disable_tx_chain			
0xCB	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0xD3	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0xD4	0	Switch_Settling_Time_11a							TxRxatten_11a						…	
0xD5	AntennaCTL_11a_0				AntennaCTL_11a_1						AntennaCTL_11a_2					
0xD6	AntennaCTL_11a_3						AntennaCTL_11a_4						AntennaCTL_11a_5			
0xD7	…		AntennaCTL_11a_6						AntennaCTL_11a_7						…	
0xD8	AntennaCTL_11a_8				AntennaCTL_11a_9						AntennaCTL_11a_10					
0xD9	ADC_Desired_Size_11a								OB_11a_4			DB_11a_4			OB_11a_3	
0xDA	…	DB_11a_3			OB_11a_2			DB_11a_2			OB_11a_1			DB_11a_1		
0xDB	Tx_end_to_xlna_on_11a								Thresh62_11a							
0xDC	Tx_end_to_xpa_off_11a								Tx_frame_to_xpa_on_11a							
0xDD	PGA_Desired_Size_11a								Noise_Floor_Thresh_11a							
0xDE	X	X	ForceA	XLNA_Gain_11a								XPD_Gain_11a				XPD_11a
0xDF	...			false_detect_backoff_11a							XR_Target_Power_11a					
0xE0	X	X	iq_cal_I_11a						iq_cal_Q_11a					init_gainI_11a		
0xE1	... (lsb)			Switch_Settling_Time_11a_Turbo							rxtx_margin_11a					
0xE2	ADC_Desired_Size_11a_Turbo							rxtx_margin_11a_Turbo						TxRxatten_11a_Turbo		
0xE3	X	X	X	X	X	X	X	PGA_Desired_Size_11a_Turbo								...(msb)

Table 1-5. **EEPROM Subsystem Design Specific Entries (continued)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xE4	phase_cal_2_11a				phase_cal_1_11a						phase_cal_0_11a					
0xE5	X	X	phase_cal_4_11a						phase_cal_3_11a						... (msb)	
0xE6	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0xF1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0xF2	0	Switch_Settling_Time_11b							TxRxatten_11b						…	
0xF3	AntennaCTL_11b_0				AntennaCTL_11b_1						AntennaCTL_11b_2					
0xF4	AntennaCTL_11b_3				AntennaCTL_11b_4								AntennaCTL_11b_5			
0xF5	…		AntennaCTL_11b_6						AntennaCTL_11b_7						…	
0xF6	AntennaCTL_11b_8				AntennaCTL_11b_9						AntennaCTL_11b_10					
0xF7	ADC_Desired_Size_11b								X	OB_11b			X	DB_11b		
0xF8	Tx_end_to_xlna_on_11b								Thresh62_11b							
0xF9	Tx_end_to_xpa_off_11b								Tx_frame_to_xpa_on_11b							
0xFA	PGA_Desired_Size_11b								Noise_Floor_Thresh_11b							
0xFB	X	X	X	XLNA_Gain_11b							XPD_Gain_11b					XPD_11b
0xFC	...			false_detect_backoff_11b						b_DB_11b				b_OB_11b		
0xFD	X	X	X	X	X	X	X	X	X	X	X	X	X	init_gainI_11b		
0xFE	cal_pier_2_11b								cal_pier_1_11b							
0xFF	X	X	rxtx_margin_11b						cal_pier_3_11b							
0x100	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0x10C	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x10D		Switch_Settling_Time_11g							TxRxatten_11g						…	
0x10E	AntennaCTL_11g_0				AntennaCTL_11g_1						AntennaCTL_11g_2					
0x10F	AntennaCTL_11g_3				AntennaCTL_11g_4								AntennaCTL_11g_5			
0x110	…		AntennaCTL_11g_6						AntennaCTL_11g_7						…	
0x111	AntennaCTL_11g_8				AntennaCTL_11g_9						AntennaCTL_11g_10					
0x112	ADC_Desired_Size_11g								X	OB_11g			X	DB_11g		
0x113	Tx_end_to_xlna_on_11g								Thresh62_11g							
0x114	Tx_end_to_xpa_off_11g								Tx_frame_to_xpa_on_11g							
0x115	PGA_Desired_Size_11g								Noise_Floor_Thresh_11g							
0x116	X	X	ForceB	XLNA_Gain_11g							XPD_Gain_11g					XPD_11g
0x117	...			false_detect_backoff_11g						b_DB_11g				b_OB_11g		
0x118	ch14_filter_cck_delta					cck_ofdm_pwr_delta								init_gainI_11g		
0x119	cal_pier_2_11g								cal_pier_1_11g							
0x11A	X	X	X	XR_Target_Power_11g					2p5G_Turbo_2WmaxPower							

Table 1-5. **EEPROM Subsystem Design Specific Entries (continued)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x11B	X	X	rxtx_margin_11g						cal_pier_3_11g							
0x11C	X	X	X	X	X	iq_cal_I_11g						iq_cal_Q_11g				
0x11D	...(lsb)	Switch_Settling_Time_11g_Turbo							ofdm_cck_gain_delta							
0x11E	ADC_Desired_Size_11g_Turbo					rxtx_margin_11g_Turbo						TxRxatten_11g_Turbo				
0x11F	X	X	X	X	X	PGA_Desired_Size_11g_Turbo							...(msb)			
0x120	X	X	X	X	phase_cal_1_11g						phase_cal_0_11g					
0x121	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0x127	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x128	CTL 1								CTL 2							
0x129	CTL 3								CTL 4							
0x12A	CTL 5								CTL 6							
0x12B	CTL 7								CTL 8							
0x12C	CTL 9								CTL 10							
0x12D	CTL 11								CTL 12							
0x12E	CTL 13								CTL 14							
0x12F	CTL 15								CTL 16							
0x130	CTL 17								CTL 18							
0x131	CTL 19								CTL 20							
0x132	CTL 21								CTL 22							
0x133	CTL 23								CTL 24							
0x134	CTL 25								CTL 26							
0x135	CTL 27								CTL 28							
0x136	CTL 29								CTL 30							
0x137	CTL 31								CTL 32							
0x138	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
...	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x14F	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

[1] For complete information on configuring eep_map0, refer to Appendix A, eep_map 0.
[2] For complete information on configuring eep_map1, refer to Appendix B, eep_map 1.
[3] This location needs to be 0 for compatibility with earlier software releases. Used to be the old location for 32KHz_Enable flag.

Table 1-6 describes the fields listed in Table 1-5 in more detail.

Table 1-6. **Subsystem Design Specific EEPROM Entries Descriptions**

Register	Description
CCS	Country Code Selector flag identifying whether the 12-bit code represents a country code or a regulatory domain code. See the support bulletin *Worldwide Roaming Design Specification* for details.
WWR	Worldwide Roaming flag. If set to 1, the scan manager must use passive scan to discover the regulatory domain. See the support bulletin *Worldwide Roaming Design Specification* for details.
Country_or_Domain_Code	A 12-bit code identifying the currently selected country or domain of operation. The NIC driver or the AP software makes use of this value to determine the channels available for operation and the operating power at those channels for all data rates (see the support bulletin *Setup for Country or Regulatory Domain* for more information).
Checksum	A 16-bit value that causes a bit position X-OR run across EEPROM locations 0xC1 to 0x3FF or EEPROM data end (whichever is larger) to yield a result of 0xFFFF. If the checksum fails, the software should default to low power values, because the data in this section of the EEPROM has been corrupted. If the EEPROM data written extends beyond 16K, then EEPROM locations 0x1B and 0x1C specify where the checksum ends.
Version	Allows the software to decipher the EEPROM contents. Any time the EEPROM layout is changed, the major and minor version combination should be used to convey that information to the driver.
TD (Turbo_disable)	When set, prevents the card from using Atheros Turbo Mode™ in 5 GHz operation.
RFK	This bit is only used by a hardware switch. Ignored by hardware if set to 0. If the RFK bit is set to 1, then a pull up resistor must be placed on the AR5211 GPIO_0, providing a hardware interface to an external on/off switch that will allow manual termination of any RF activity ■ 0= Allow RF activity ■ 1= Disallow RF activity
DeviceType	Provided for definition of device type/form factor. Currently, neither hardware nor software utilize these bits. It is recommended that these bits be set for the appropriate implementation as future versions of hardware and/or software may contain device dependent options. Device Type Definition (Recommended): ■ 001 = CardBus ■ 010 = PCI ■ 011 = Mini PCI ■ 100 = Access Point ■ 101–111 = Reserved
5G_Turbo_2WmaxPower	Maximum suggested power if user wishes to consume less than 2 W power (in dBm).
Amode	Specifies this subsystem design supports operation in 802.11a mode.
Bmode	Specifies this subsystem design supports operation in 802.11b mode.
Gmode	Specifies this subsystem design supports 802.11g mode.
Turbo2G	Flag that specifies whether this subsystem design supports 802.11g Turbo mode. ■ 1 = 802.11g Turbo operation disabled ■ 0 = 802.11g Turbo operation enabled
XR2	Reserved
XR5	Reserved

Table 1-6. **Subsystem Design Specific EEPROM Entries Descriptions (continued)**

Register	Description
EAR_start_location	EEPROM location marking the beginning of the EEPROM Added Registers (EAR). The EAR scheme facilitates a version of software release to work with the hardware released at a later date. Refer to Appendix E, "EAR EEPROM", for further details.
Target_Power_Start_Location	EEPROM location marking the beginning of the target powers section.
32KHz_enable	When set, indicates the presence of a 32 kHz crystal for sleep mode.
eep_map	This flag indicates the type of EEPROM layout. There are 2 types of EEPROM layouts supported starting with EEPROM version 4.0 based on the type of power control scheme supported by the chipset: ■ 00 = Legacy EEPROM layout ■ 01 = EEPROM layout for the advanced power control scheme; offers enhanced accuracy of power control even at low output power levels
EAR_FILE_VERSION	Source control version of the .ear file for traceability.
EEP_FILE_VERSION	Source control version of the .eep file for traceability.
ART_BUILD_NUM	Build number of ART used to calibrate the card. ART version is the same as EEPROM version starting with EEPROM version 4.3.
EAR_FILE_IDENTIFIER	Numerical identifier after the base .ear filename. For example, 240_mb32ag_NN.ear would result in an EAR_FILE_IDENTIFIER = 0xNN.
MaskForRadio0	2-bit mask of modes supported by radio interface 0. ■ lsb = 1 indicates 802.11g/802.11b ■ msb = 1 indicates 802.11a support
MaskForRadio1	2-bit mask of modes supported by radio interface 1. ■ lsb = 1 indicates 802.11g/802.11b ■ msb = 1 indicates 802.11a support
disableComp	When set, prevents the adapter from supporting compression
disable_AES	When set, prevents the adapter from supporting AES
disable_FF	When set, prevents the adapter from supporting Fast Frames
disable_bursting	When set, prevents the adapter from supporting bursting
max_qcu	Specifies the maximum number of QCUs supported by the adapter. Valid values are in the range 0–31. Setting 0 defaults to the current limit of 10.
enable_clip	Set to 1 to enable heavy clipping
key_cache_size	Specify the length of the key cache. The size is specified as $2^{[15-1]}$. Setting 0 defaults to the current size of 128.
Antenna_Gain_5G	Antenna gain in the 5 GHz band. This value is added to the 802.11a calibrated power by the driver to compute the final output power. An 8-bit signed quantity in 0.5 dB steps (e.g., +12 Antenna_Gain_5G = +6 dB).
Antenna_Gain_2.4G	Antenna gain in the 2.4 GHz band. This value is added to the 802.11b and 802.11g calibrated power by the driver to compute the final output power. An 8-bit signed quantity in 0.5 dB steps (e.g., +12 Antenna_Gain_2.4G = +6 dB).
Switch_Settling_Time_11a	802.11a Tx/Rx switch settling time; can be set according to the settling time of the external switch. The equation to calculate switch settling time register is (switch settling time register) = (switching settling time / 25 ns) +19.
Switch_Settling_Time_11a_Turbo	802.11a Tx/Rx switch settling time for the Turbo mode.
TxRxatten_11a	Specifies the difference in attenuation between the Tx/Rx switch in Tx mode and in Rx mode, in 1 dB increments for 802.11a operation (at 5 GHz). For example, from the receiver's perspective, when in Rx mode there will be little loss from the antenna to the AR5111/AR5112. In Tx mode, there will be more loss from the antenna to the AR5111/AR5112 due to the isolation of the PIN diode switch. This difference is what should be placed in TxRxatten. Useful only when a strong signal is presented to the receiver.

Table 1-6. **Subsystem Design Specific EEPROM Entries Descriptions (continued)**

Register	Description
TxRxatten_11a_Turbo	802.11a Tx/Rx switch attenuation difference for the Turbo mode.
AntennaCTL_11a_0 … AntennaCTL_11a_10	Antenna Control Settings for operation in the 802.11a mode. A 6-bit setting controlling the output of pins (ATTEN5, ATTEN2, ANTD, ANTC, ANTB, ANTA) is specified for each of the 11 possible states of transmission/reception.
ADC_Desired_Size_11a	Desired amplitude of signal to be presented to the analog to digital converter (ADC) in 802.11a operation mode. This value is used by the automatic gain control stage in AR5111/AR5112 to output the appropriate signal size to make the best use of ADC range. The value specified is in 0.5 dB step (e.g, –32 ADC_Desired_Size_11a = –16 dBm).
ADC_Desired_Size_11a_Turbo	802.11a ADC desired size for the Turbo mode.
OB_11a_1 … OB_11a_4	Used to set the bias current for the output stage of PA in 802.11a operation mode. To enable optimum performance and power consumption over a wide range of operation, up to 4 values can be specified for OB_11a and DB_11a combinations: ■ OB_11a_1 is used for channels in the range of 5.15–5.25 GHz. ■ OB_11a_2 is used for channels in the range of 5.25–5.50 GHz. ■ OB_11a_3 is used for channels in the range of 5.50–5.70 GHz. ■ OB_11a_4 is used for channels in the range of 5.70–5.85 GHz. Each value supports a range of 1-7.
DB_11a_1 … DB_11a_4	Used to set the bias current for the driver stage of the PA in 802.11a operation mode. (See "OB_11a_1 … OB_11a_4" for an explanation of the four values.) Each value supports a range of 1–7.
Tx_end_to_xlna_on_11a	Specifies the time difference from when the baseband is finished sending a frame to when the external low noise amplifier (LNA) switch is activated in 802.11a operation mode. This parameter can be adjusted based on the ramp-up time of the external LNA. For example, if the external LNA ramp-up time is slow, then it would be desirable to turn on the external LNA sooner so that the beginning of the receive frame is not missed.
Thresh62_11a	Adjusts clear channel assessment (CCA) sensitivity to meet IEEE 802.11a specification. Section 17.3.10.5 of the IEEE 802.11a specification specifies CCA sensitivity as "A start of a valid OFDM transmission at receive level equal or greater than minimum 6 Mbps sensitivity (–82 dBm) shall cause CCA to indicate Busy with probability > 90% within 4 µs. If the preamble portion of a frame was missed, the receiver shall hold the carrier sense (CS) signal Busy for any signal 20 dB above minimum 6 Mbps sensitivity (–62 dBm)". A lower threshold can be chosen for better performance in the presence of collisions by changing the setting in this register.
Tx_end_to_xpa_off_11a	Specifies the time difference from when the baseband is finished sending a frame to when the external PA switch is deactivated in 802.11a operation mode. This parameter can be adjusted based on the ramp-down time of the external PA. For example, if the external PA ramp-down time is very fast, then it would be desirable to delay deactivating the external PA to ensure that the end of the frame being sent is not prematurely truncated.
Tx_frame_to_xpa_on_11a	Specifies the time difference from when the medium access control (MAC) sends the frame to when the external PA switch is activated in 802.11a operation mode. This parameter can be adjusted based on the ramp-up time of the external PA. For example, if the external PA ramp-up time is very fast, then it would be desirable to activate the external PA sooner so that the beginning of the frame being sent is not prematurely truncated.

Table 1-6. **Subsystem Design Specific EEPROM Entries Descriptions (continued)**

Register	Description
PGA_Desired_Size_11a	Desired amplitude of the output of the programmable analog gain stage presented as input to the baseband gain stage in 802.11a operation mode. This value is used to ensure optimal input signal for the external PA. The value specified is in 0.5 dB steps. For example, a typical value of –72 corresponds to a –36 dBm signal level.
PGA_Desired_Size_11a_Turbo	802.11a PGA desired size for the Turbo mode.
Noise_Floor_Thresh_11a	Noise floor threshold in 802.11a operation mode. The value specified is in 1 dB steps (that is, a typical value of –85 Noise_Floor_Thresh_11a = –85 dBm).
XPD_11a	Selects between the internal or external detector for power control in 802.11a operation mode. XPD_11a = 1 selects the external detector and XPD_11a = 0 selects the internal.
XPD_Gain_11a	Controls the gain for the external power detector output in 802.11a operation mode. There are only 4 valid settings for these bits for eep_map = 0: ■ XPD_gain_11a = 14 corresponds to 0 dB gain ■ XPD_gain_11a = 13 corresponds to 6 dB gain ■ XPD_gain_11a = 11 corresponds to 12 dB gain ■ XPD_gain_11a = 7 corresponds to 18 dB gain Any other value for XPD_gain_11a will result in unpredictable output power for eep_map = 0. In eep_map = 1, this field is used as an "xpd_gain_mask" for up to two xpd_gains for which the cal data is stored in the EEPROM (see Appendix B for details). The LSB indicates if cal data is stored for 0 dB xpd_gain, and msb indicates if 18 dB was used. For example: ■ XPD_gain_11a = 1001b indicates cal data for 0 dB and 18 dB xpd_gain is stored in EEPROM ■ XPD_gain_11a = 1010b indicates cal data for 6 dB and 18 dB xpd_gain is stored in EEPROM ■ XPD_gain_11a = 0010b indicates cal data for only 6 dB xpd_gain is stored in EEPROM
XLNA_Gain_11a	Total gain provided by the LNA present on the target board. This value is for consumption by the NDIS driver or AP software and is not programmed into any device register for operation.
Force_A	When set, specifies the use of the DB_11a value for the 5 GHz driver bias, or, if Clear, let it be under automatic control. The chip can automatically select a value based upon the output power level.
False_Detect_Backoff_11a	Due to the capability of receiving extremely low levels of signals, sometimes a spur may interfere with the proper operation of the system. There is a mechanism built in the chipset to overcome those spurs by applying this backoff in sensitivity (in dB) only to the channels affected by the spur (the list of affected channels may be stored in the driver). Using good design practices, it is recommended to keep the spur levels low enough that no backoff is needed.
Init_GainI_11a	An initial gain value to start with after a reset_device. Used by the power control logic to get the initial packets closer to target power levels sooner. Range is 1–50 for AR5001 products; 1–35 for AR5002 products.
iq_cal_I_11a	I coefficient obtained from the iq_cal to correct for the iq_mismatch in the 802.11g receive path. This coefficient is used to correct for the iq_mismatch and improve receive sensitivity.
iq_cal_Q_11a	Q coefficient obtained from the iq_cal to correct for the iq_mismatch in the 802.11g receive path. This coefficient is used to correct for the iq_mismatch and improve receive sensitivity.

Table 1-6. **Subsystem Design Specific EEPROM Entries Descriptions (continued)**

Register	Description
rxtx_margin_11a	Margin (in dB) that controls when the final stage of attenuation (stage r1x12, or r2x12, in antenna control switch table) is kicked in. A higher value for rxtx_margin_11a means the final attenuation stage will be kicked-in at a lower input signal level to attenuate the received signal.
rxtx_margin_11a_Turbo	802.11a rxtx_margin for the Turbo mode.
Locations 0xF2–0xFD	Similar hardware settings as described in "Switch_Settling_Time_11a" through "rxtx_margin_11a" described above as applicable to the 802.11b mode of operation. With one exception: "TxRxatten_11b". The five parameters for the Turbo mode are not applicable for 802.11b.
TxRxatten_11b	Specifies the difference in attenuation between the Tx/Rx switch in Tx mode and in Rx mode, in 1 dB increments for 802.11b operation.
Cal_Pier_1_11b	If the force_piers_list_11b is used to calibrate the card in 802.11b mode, this field represents the location of the first 802.11b frequency pier. 0xFF indicates unused field. fbin = freq in MHz - 2300.
Cal_Pier_2_11b	If the force_piers_list_11b is used to calibrate the card in 802.11b mode, this field represents the location of the second 802.11b frequency pier. 0xFF indicates unused field. fbin = freq in MHz - 2300.
Cal_Pier_3_11b	If the force_piers_list_11b is used to calibrate the card in 802.11b mode, this field represents the location of the third 802.11b frequency pier. 0xFF indicates unused field. fbin = freq in MHz – 2300.
Locations 0x10D–0x119	Similar hardware settings as described in "Switch_Settling_Time_11a" through "rxtx_margin_11a" as applicable to the 802.11g mode of operation. With one exception: "TxRxatten_11g". The five parameters for the Turbo mode are also stored for the 802.11g.
TxRxatten_11g	Specifies the attenuation through the 5 GHz/2.4 GHz Rx switch in 2.4 GHz Rx mode. The attenuation is specified in 1 dB increments.
TxRxatten_11g_Turbo	802.11g TxRxAtten for the Turbo mode.
CCK_OFDM_Pwr_Delta	This value represents measured power difference between a CCK rate (that is, 11 Mbps) and an OFDM rate (that is, 6 Mbps) for the same PCDAC value in 802.11g mode. Default value is 1.5 dB. This number needs to have .1 dB resolution, so the value stored is 10 times the actual delta. For example, if the delta is measured, and specified in calsetup.txt, to be 1.3 dB, value stored in this field will be 1.3 * 10 = 13.
Ch14_Filter_CCK_Delta	Special filter bit is used for channel 14 in 802.11g mode in Japan to satisfy the lower power per MHz limit. This change in spectral shape results in slightly different output power for a given pcdac (about 1.5 dB). This value can be obtained by measuring the difference between 1 mbps power out for a given pcdac between 2472 and 2484 MHz. This value is stored with a 0.1 dB resolution as an unsigned 5-bit field (0–3.1dB). Value stored is 10 times the actual value.
OFDM_CCK_GAIN_DELTA	Difference in gainF to output same power for OFDM vs. CCK packets. This stems from a difference in peak-to-average ratio for OFDM and CCK packets. All things being equal, the Peak-to-Average Ratio (PAR) difference is about 7.5 dB, but different baseband scaling may be used for OFDM and CCK packets.
Cal_Pier_1_11g	If the force_piers_list_11g is used to calibrate the card in 802.11g mode, this field represents the location of the first 802.11g frequency pier. 0xFF indicates unused field. fbin = freq in MHz – 2300.
Cal_Pier_2_11g	If the force_piers_list_11g is used to calibrate the card in 802.11g mode, this field represents the location of the second 802.11g frequency pier. 0xFF indicates unused field. fbin = freq in MHz – 2300.
Cal_Pier_3_11g	If the force_piers_list_11g is used to calibrate the card in 802.11g mode, this field represents the location of the third 802.11g frequency pier. 0xFF indicates unused field. fbin = freq in MHz – 2300.

Table 1-6. **Subsystem Design Specific EEPROM Entries Descriptions (continued)**

Register	Description
2p5G_Turbo_2WmaxPower	Maximum suggested power (in dBm) in 802.11g Turbo mode if user wishes to consume less than 2 W power.
XR_Target_Power_11a	Reserved
XR_Target_Power_11g	Reserved
CTL 1 … CTL 32	Hex codes for the CTLs that this card is calibrated for. The CTL that the driver determines the current country code index belongs to is matched against these hex codes to figure out the location of correct data to be retrieved from the EEPROM. The lower 3 bits (bits [2:0]) of the hex code identify which operating mode (802.11a/802.11b/802.11g) the CTL pertains to: ■ 000 = 802.11a mode ■ 001 = 802.11b mode ■ 010 = 802.11g ■ 011 = Atheros Turbo Mode at 5 GHz ■ 100 = Atheros Turbo Mode at 2.4 GHz
b_OB_11b	Used to set the bias current for the output stage of PA in the AR2112/AR5112 chip in 802.11b mode.
b_DB_11b	Used to set the bias current for the driver stage of PA in the AR2112/AR5112 chip in 802.11b mode.
b_OB_11g	Used to set the bias current for the output stage of PA in the AR2112/AR5112 chip in 802.11g mode.
b_DB_11g	Used to set the bias current for the driver stage of PA in the AR2112/AR5112 chip in 802.11g mode.
Force_B	When set, indicates whether to use the b_DB_11b (or b_DB_11g) value specified for the 2.4 GHz driver bias, or if Clear, let it be under automatic control. The chip can automatically select a value based upon the output power level.
iq_cal_I_11g	I coefficient obtained from the iq_cal to correct for the iq_mismatch in the 802.11g receive path. This coefficient is used to correct for the iq_mismatch and improve receive sensitivity.
iq_cal_Q_11g	Q coefficient obtained from the iq_cal to correct for the iq_mismatch in the 802.11g receive path. This coefficient is used to correct for the iq_mismatch and improve receive sensitivity.
phase_cal_0_11a ... phase_cal_4_11a	Phase calibration information between the two chains for the MIMO chip AR5513 based reference designs in 5 GHz band. The 6-bit value stored is the phase delta in degrees divided by 10. For example, a phase delta of 270 will be stored as 27. Phase calibration is stored for 5 channel frequencies in the 5GHz band: 5, 5.2, 5.4, 5.6 and 5.8 GHz. For all other channels, it needs to be linearly interpolated between the two nearest phase cal channels.
phase_cal_0_11g ... phase_cal_1_11g	Phase calibration information between the two chains for the MIMO chip AR5513 based reference designs in 2.5 GHz. The 6-bit value stored is the phase delta in degrees divided by 10. For example, a phase delta of 270 will be stored as 27. Phase calibration is stored for 2 channel frequencies in the 2.5 GHz band : 2.412 and 2.472 GHz. For all other channels, it needs to be linearly interpolated between the two nearest phase cal channels.
disable_tx_chain	A 3-bit mask indicating which MIMO chains are disabled for transmission on the design. ■ disable_tx_chain[0] → disable chain 0 for tx ■ disable_tx_chain[1] → disable chain 1for tx ■ disable_tx_chain[2] → disable chain 2 for tx (not used for AR5513)

Table 1-6. **Subsystem Design Specific EEPROM Entries Descriptions (continued)**

Register	Description
disable_rx_chain	A 3-bit mask indicating which MIMO chains are disabled for reception on the design. ■ disable_rx_chain[0] → disable chain 0 for rx ■ disable_rx_chain[1] → disable chain 1for rx ■ disable_rx_chain[2] → disable chain 2 for rx (not used for AR5513)
en_fcc_mid	Flag indicating whether operation in FCC band from 5.47-5.7 GHz is supported or not : ■ 0 = Operation is prohibited ■ 1 = Operation is supported
en_jap_even_u1	Flag indicating whether operation in Japan UNII1 band from 5.15-5.25 GHz on 20 MHz even channels (5180, 5200, 5220, 5240) is supported or not : ■ 0 = Operation is prohibited ■ 1 = Operation is supported
en_jap_u2	Flag indicating whether operation in Japan UNII2 band from 5.25-5.35 GHz is supported or not : ■ 0 = Operation is prohibited ■ 1 = Operation is supported
en_jap_mid	Flag indicating whether operation in Japan band from 5.47-5.7 GHz is supported or not : ■ 0 = Operation is prohibited ■ 1 = Operation is supported
dis_jap_odd_u1	Flag to disable whether operation in Japan UNII1 band from 5.15-5.25 GHz on odd channels (5170, 5190, 5210, 5230) is supported or not : ■ 0 = Operation is supported (NOT disabled) ■ 1 = Operation is prohibited (disabled)

Dual 802.11a Configuration Support

Atheros chipsets support the concurrent dual 802.11a AP designs. This support requires ability to store configuration/calibration information for two 802.11a radios. EEPROM layouts prior to version 4.8 could support configuration information for one 802.11a, 802.11b, and 802.11g radio. Primary focus in adding support for dual 802.11a radios was ease of implementation and leverage code re-use.

Beginning with EEPROM version 4.8, only for the access points that have two 802.11a radios, the EEPROM configuration information is duplicated immediately following the first, in the flash for the second chain. Configuration data for the modes supported by the second chain is updated in this block and rest of the information is left unchanged.

A

eep_map 0

This appendix provides EEPROM subsystem-specific information for eep_map value set to 0 (refer to Table 1-5 on page 1-11.)

EEPROM NIC or AP Specific Information (0x150–0x2BE)

Each wireless NIC or AP must comply with local emission regulations and 802.11 spectral limitations. Due to the sensitive nature of RF circuit design, for each manufactured NIC or AP to provide an optimal level of throughput performance and output power, it is essential to calibrate each device and store raw performance capability information in the EEPROM.

For the transmit power control at any channel, the AR5211/AR5311 must be programmed with a 64-entry PCDAC lookup table indexed on the desired power value in 0.5 dB steps. For example, entry 18 should store the PCDAC value (18*0.5 = 9 dB) power output. A snapshot of the NIC or AP raw power capability over the frequency range is stored in the EEPROM at ten frequency piers. For all intermediate channels, the driver reconstructs the 64-entry table from this snapshot through interpolation using the PLA scheme.

At each frequency pier, power versus PCDAC dependence often looks like Figure A-1: negligible output below PCDAC_MIN and monotonic increase in output power up to PCDAC_MAX, beyond which the PA saturates with no increase in output power. To capture maximum details of this data, the EEPROM stores only the region between PCDAC_MIN and PCDAC_MAX.

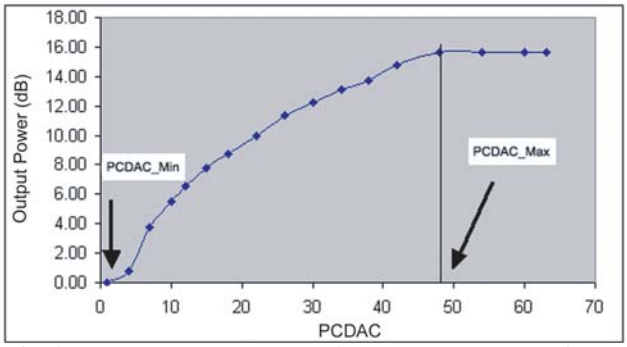

Figure A-1. **Typical Output Power Dependence on PCDAC Values for a Channel**

The format for data stored at a frequency pier is summarized in Figure A-2. The values stored for each frequency pier are: PCDAC_MIN, PCDAC_MAX, and the interpolated output power at fixed percentage intercepts between PCDAC_MIN and PCDAC_MAX (the default set of intercept percentages employed is 0%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% and 100%). The PCDAC values are stored in a 7-bit format. Finer gridsize is employed at the range beginning and end to capture accurate transition region details.

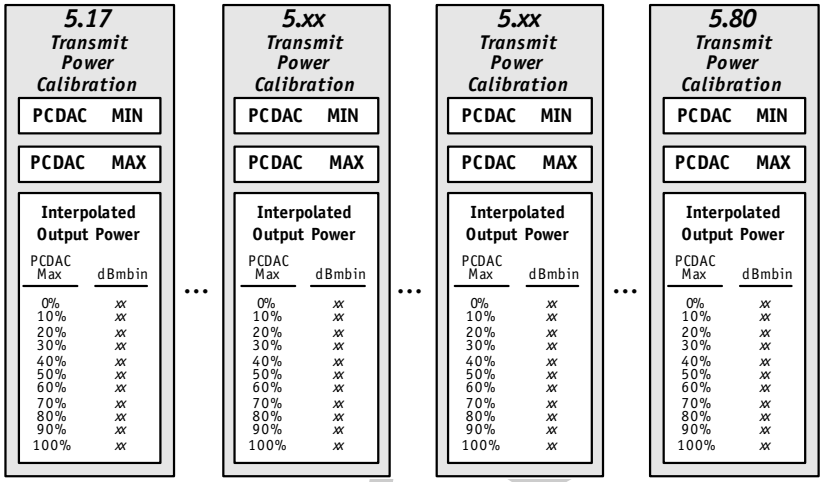

Figure A-2. **Calibration Data Format on the EEPROM for a Frequency Pier**

802.11a Mode Calibration Information (0x150–0x186)

The block of EEPROM locations (0x150–0x186) is divided into Group 1 (0x150–0x154) and Group 2 (0x155–0x186). Group 1 in Table A-1 shows the pier locations for ten frequency piers in 8-bit frequency representation.

This formula relates the frequency in 5 GHz range (freq) to the 8-bit value stored on the EEPROM (fbin):

$$fbin = (freq - 4800) / 5 \text{ if } 4800 \leq freq < 6080 \text{ (freq in MHz)}$$

Table A-1. **Group 1. 802.11a Frequency Piers (0x150–0x154)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x150	Fbin0								Fbin1							
0x151	Fbin2								Fbin3							
0x152	Fbin4								Fbin5							
0x153	Fbin6								Fbin7							
0x154	Fbin8								Fbin9							

Table A-2 shows 4.9–5.85 GHz raw data for one of ten frequency piers for the eep_map = 00 layout: PCDAC_MAX, PCDAC_MIN, and raw power at 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 percent. The PCDAC values are stored in 6-bit format and the power expressed in 6 bits (0–32 dBm in 0.5 dB steps).

Table A-2. **Group2. 802.11a Raw Power Calibration Data at a Frequency Pier (0x155–0x186)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	PCDAC_MAX						PCDAC_MIN						dBmbin0			
1	…		dBmbin10						dBmbin20						…	
2	dBmbin30				dBmbin40						dBmbin50					
3	dBmbin60						dBmbin70						dBmbin80			
4	…		dBmbin90						dBmbin100						X	X

Group 2 in Table A-3 shows 4.9–5.85 GHz raw data for one of ten frequency piers. The PCDAC values are stored in a 6-bit format and the power is expressed in 7 bits (0–32 dBm in 0.25 dB steps).

Table A-3. **Group2. 802.11a Raw Power Calibration Data at a Frequency Pier (0x155–0x186)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	PCDAC_MAX						PCDAC_MIN						dBmbin0			
1	…		dBmbin10						dBmbin20						…	
2	dBmbin30				dBmbin40						dBmbin50					
3	dBmbin60						dBmbin70						dBmbin80			
4	…		dBmbin90						dBmbin100						X	X

802.11b Mode Calibration Information (0x187–0x195)

Group 3 represents the EEPROM locations 0x187–0x195. Table A-4 shows 2.4–2.5 GHz raw data at frequency piers 2.412, 2.447, and 2.484 GHz.

Table A-4. **Group 3. 802.11b Raw Power Calibration Data (0x187–0x195)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	PCDAC_MAX						PCDAC_MIN						dBmbin0			
1	…		dBmbin10						dBmbin20						…	
2	dBmbin30				dBmbin40						dBmbin50					
3	dBmbin60						dBmbin70						dBmbin80			
4	…		dBmbin90						dBmbin100						X	X

802.11g Calibration Information (0x196–0x1A4)

Group 4 represents the EEPROM locations 0x196–0x1A4. Table A-5 shows 2.4–2.5 GHz raw data at 3 frequency piers with OFDM modulation: 2.312 GHz, 2.412 GHz, and 2.484 GHz.

Table A-5. **Group 4. 802.11g Calibration Data (0x196–0x1A4)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	PCDAC_MAX						PCDAC_MIN						dBmbin0			
1	…		dBmbin10						dBmbin20						…	
2	dBmbin30				dBmbin40						dBmbin50					
3	dBmbin60						dBmbin70						dBmbin80			
4	…		dBmbin90						dBmbin100						X	X

Target Power Calibration for 802.11a Mode (0x1A5–0x1B4)

Group 5 represents the EEPROM locations 0x1A5–0x1B4. Table A-6 shows target power for 802.11a mode of operation for 6–24, 36, 48, and 54 Mbps, specified at up to eight test frequencies. For i=0–7, Test_channel_i represents frequency in eight bits, followed by the target power in dBm at various rates in six bits (unsigned value). Not all eight test frequencies must be specified; if fewer than eight are specified, the remaining unused bits are 0.

Table A-6. **Group 5. Target Power Calibration for 802.11a Mode (0x1A5–0x1B4)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Test_channel_i								dB_6-24_i						...	
1	dB_36_i				dB_48_i						dB_54_i					

Target Power Calibration for 802.11b Mode (0x1B5–0x1B8)

Group 6 represents the EEPROM locations 0x1B5–0x1B8. Table A-7 shows target power for 802.11b mode of operation for rates 1, 2, 5.5, and 11 Mbps which are specified at exactly two test frequencies. For i=0–1, Test_channel_i represents frequency in eight bits, followed by the target power in dBm at various rates expressed in six bits (unsigned value).

Exactly two test frequencies must be specified. To cover the entire operating range, it is recommended to specify:

- Test_channel_0 = 2.412 GHz
- Test_channel_1 = 2.484 GHz

This formula is used to relate the frequency in 2.4 GHz range (freq) to the 8-bit value stored on the EEPROM as Test_channel_i (fbin):

fbin = (freq – 2300) (freq in MHz)

Table A-7. **Group 6. Target Power Calibration for 802.11b Mode (0x1B5–0x1B8)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1B5	Test_channel_0								dB_6-24_i						...	
0x1B6	dB_36_i				dB_48_i						dB_54_i					
0x1B7	Test_channel_1								dB_6-24_i						...	
0x1B8	dB_36_i				dB_48_i						dB_54_i					

802.11g Target Power Calibration Information (0x1B9–0x1BE)

Group 7 represents the EEPROM locations 0x1B9–0x1BE. Table A-8 shows target power for 802.11g mode for rates 6-24, 36, 48, and 54 Mbps, specified at up to three test frequencies. For all CCK rates in 802.11g mode, the target powers from 802.11b mode are used. For i=0–2, Test_channel_i represents frequency in seven bits, followed by the target power in dBm at various rates expressed in six bits (unsigned value).

Not all three test frequencies need to be specified. If less than three test frequencies are specified, the remaining unused bits should be 0. To cover the entire operating range, it is recommended to specify:

- Test_channel_0 = 2.412 GHz
- Test_channel_2 = 2.484 GHz

Table A-8. **Group 7. Target Power Calibration for 802.11g Mode (0x1B9–0x1BE)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1B9	Test_channel_0								dB_6-24_i						...	
0x1BA	dB_36_i				dB_48_i						dB_54_i					
0x1BB	Test_channel_1								dB_6-24_i						...	
0x1BC	dB_36_i				dB_48_i						dB_54_i					
0x1BD	Test_channel_2								dB_6-24_i						...	
0x1BE	dB_36_i				dB_48_i						dB_54_i					

CTL Information (0x1BF–0x2BE)

Group 8 represents the EEPROM locations 0x1BF–0x2BE. Table A-9 shows CTL data for one of the 32 CTLs: 8 user defined band edges expressed in 8 bits, the maximum band edge power (6 bits—unsigned value) at each band edge and the non-edge flags for each CTL band edge. This maximum power (dBm) at band edge is used to limit power only at that channel if the non-edge flag is set to 0. If the non-edge flag is set to 1, that indicates the CTL frequency to not be a band edge and the corresponding dBmax_edge<ii> specified will limit power at a range of channels (See "Conformance Testing Limits" on page 1-6 for details). If less than 8 unique band edges are provided, the rest of the band edge locations will be filled with 0.

This format is repeated for all defined CTLs, up to a maximum number of 32.

It is important to determine the mode (802.11a/802.11b/802.11g/TURBO) to which a CTL pertains, to correctly read back the band edge frequencies. The lower 3 bits of the CTL hex code are used to convey this information as described in Table 1-6.

Table A-9. **Group 8. CTL Information (0x1BF–0x2BE)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BandEdge1								BandEdge2							
1	BandEdge3								BandEdge4							
2	BandEdge5								BandEdge6							
3	BandEdge7								BandEdge8							
4	X	Flg1	dBmax_edge1						X	Flg2	dBmax_edge2					
5	X	Flg3	dBmax_edge3						X	Flg4	dBmax_edge4					
6	X	Flg5	dBmax_edge5						X	Flg6	dBmax_edge6					
7	X	Flg7	dBmax_edge7						X	Flg8	dBmax_edge8					

B

eep_map 1

This appendix provides EEPROM subsystem-specific information for eep_map value set to 1 (refer to Table 1-5 on page 11.)

EEPROM NIC or AP Specific Information (0x150–0x2BE)

Each wireless NIC or AP is required to comply with local emission regulations and 802.11 spectral limitations. Due to the sensitive nature of RF circuit design, for each manufactured NIC or AP to provide an optimal level of throughput performance and output power, it is essential to calibrate each device and store the raw performance capability information in the EEPROM.

For the transmit power control at any channel, the AR5212/AR5312 must be programmed with a 64-entry PCDAC lookup table indexed on the desired power value, referred to as the Pmin, in 0.5 dB steps. For example, entry 18 stores the PCDAC value that results from the Pmin + 18*0.5 = Pmin + 9 dBm power output. For the format eep_map = 1, negative power levels in dBm are also supported. Pmin is the minimum power achievable on a given channel (typically for xpd_gain = 18 dB, PCDAC = 1). If the minimum power is positive, Pmin = 0. If Pmin is negative, e.g. –5 dBm, entry 18 in the 64-entry PCDAC table corresponds to –5 + 18*0.5 = +4 dBm. A snapshot of the raw power capability of the NIC or AP over the entire frequency range is stored in the EEPROM at 10 frequency piers. For all intermediate channels, the 64-entry table is reconstructed by the driver from this snapshot at 10 frequency piers through interpolation using the PLA scheme.

At each frequency pier, the idealized power versus PCDAC dependence for AR5112-based designs is shown in Figure B-1. The dependence is expected to be fairly linear with PCDACs. To cover the entire range of output power, calibration information for up to two xpd_gain values is stored on the EEPROM for eep_map = 1. Typically xpd_gains of 0 dB and 18 dB are sufficient to accurately cover the entire range.

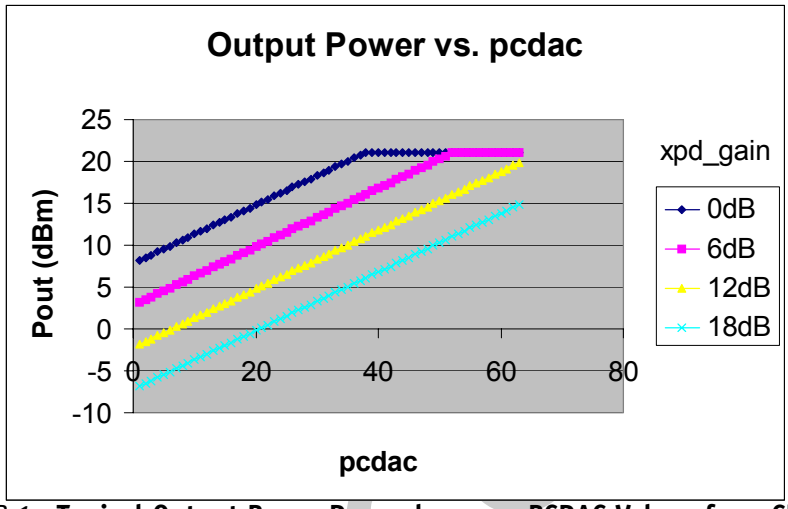

Figure B-1. **Typical Output Power Dependence on PCDAC Values for a Channel**

The format for data stored at a frequency pier is summarized in this section. The values stored for each frequency pier are:

- Power_max: max power at that channel (starting with EEPROM version 4.3, Power_max = Pwr4_xg1)
- For the lower xpd_gain (higher output power curve), four PCDACs are selected: pcd1 = 1 or 25, pcd4 = lowest PCDAC to output Psat, pcd3 = maximum linear output power, pcd2 is at 70% intercept of pcd1 and pcd3
- Pwr1_xg1, Pwr2_xg1, Pwr3_xg1, Pwr4_xg1 for the lower xpd_gain; power levels for four PCDACs are stored for the lower xpd_gain.
- Pcd2_delta, Pcd3_delta, Pcd4_delta for the lower xpd_gain; the first PCDAC (Pcd1) is stored on the EEPROM. Pcd2 = Pcd1 + Pcd2_delta, Pcd3 = Pcd2 + Pcd3_delta, Pcd4 = Pcd3 + Pcd4_delta.
- Pwr1_xg2, Pwr2_xg2, Pwr3_xg2 for the higher xpd_gain; power levels for three PCDACs are stored for the higher xpd_gain. The PCDACs are fixed at 20, 35 and 63, respectively.

The power values are stored in a signed 8-bit format in 0.25 dB steps. The Pcdac_deltas are stored in a 5-bit format. Power levels for intermediate PCDACs are linearly interpolated from these sampling points.

802.11a Mode Calibration Information

One of the major differences between eep_map = 1 and eep_map = 0 EEPROM formats is that this section is conditionally stored on EEPROM for eep_map = 1, and is always present for eep_map = 0. In eep_map = 1, this section is present only if AMode is set to 1 (i.e, 802.11a mode is supported). If present, this section starts at EEPROM location 0x150.

The first block of five EEPROM locations (Group 1) stores the pier locations for up to ten frequency piers expressed in 8-bit frequency representation. Table B-1 shows the format of these five words.

This formula relates the frequency in 5 GHz range (freq) to the 8-bit value stored on the EEPROM (fbin) is:

fbin = (freq - 4800) / 5 if 4800 ≤ freq < 6080

Table B-1. **Group 1. 802.11a Frequency Piers**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x150	Fbin0								Fbin1							
0x151	Fbin2								Fbin3							
0x152	Fbin4								Fbin5							
0x153	Fbin6								Fbin7							
0x154	Fbin8								Fbin9							

Group 2, which follows Group 1, stores the calibration data for up to ten frequency piers. Only as many piers are stored on the EEPROM as are actually measured, i.e, as have a corresponding non-zero entry in the Group 1. Table B-2 shows the format of calibration data at each frequency pier.

Table B-2. **Group 4. 802.11a Raw Power Calibration Data at a Given Frequency Pier**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Pwr2_xg1								Pwr1_xg1							
1	Pwr4_xg1								Pwr3_xg1							
2	X	Pcd4_delta					Pcd3_delta					Pcd2_delta				
3	Pwr2_xg2								Pwr1_xg2							
4	X	X	Pcd1						Pwr3_xg2							

802.11b Mode Calibration Information

Group 3 stores the calibration data for up to three frequency piers in 802.11b. Only as many piers are stored on the EEPROM as are actually measured. The three frequency piers are specified in the EEPROM header locations 0xFE–0xFF. Table B-3 shows the format of calibration data at each frequency pier.

In eep_map = 1, this section is present *only* if BMode is set to 1 (i.e, 802.11b mode is supported). If present, this section does not have a fixed starting location. Instead, it starts wherever Group 2 ends. If Group 1 and Group 2 are not present, then this section starts at EEPROM location 0x150.

Table B-3. **Group 3. 802.11b Raw Power Calibration Data (0x187–0x195)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Pwr2_xg1								Pwr1_xg1							
1	Pwr4_xg1								Pwr3_xg1							
2	X	Pcd4_delta					Pcd3_delta					Pcd2_delta				
3	Pwr2_xg2								Pwr1_xg2							
4	X	X	Pcd1						Pwr3_xg2							

802.11g Calibration Information

Group 4 stores calibration data for up to three frequency piers in 802.11g. Only as many piers are stored on the EEPROM as are actually measured. The three frequency piers are specified in the EEPROM header locations 0x119 and 0x11B. Table B-4 shows the format of calibration data at each frequency pier.

In eep_map = 1, this section is present only if GMode is set to 1 (i.e, 802.11g mode is supported). If present, this section does not have a fixed starting location. Instead, it starts wherever Group 3 ends. If Group 1, Group 2, and Group 3 are not present, then this section starts at EEPROM location 0x150.

Table B-4. **Group 4. 802.11g Calibration Data**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Pwr2_xg1								Pwr1_xg1							
1	Pwr4_xg1								Pwr3_xg1							
2	X	Pcd4_delta					Pcd3_delta					Pcd2_delta				
3	Pwr2_xg2								Pwr1_xg2							
4	X	X	Pcd1						Pwr3_xg2							

Target Power Calibration for 802.11a Mode

In eep_map = 1, the starting location of groups 5, 6, and 7 depends on the sizes of groups preceding it, and is specified in EEPROM location 0xC5. This section is always present and its size and format the same as for eep_map = 0.

Group 5 represents 802.11a target powers. Table B-5 shows 802.11a mode target power for 6–24, 36, 48, and 54 Mbps, at up to eight test frequencies. For i=0–7, Test_channel_i represents frequency in eight bits, followed by the target power in dBm at rates in six bits (unsigned value). Not all eight test frequencies must be specified; if fewer are specified, the remaining unused bits are 0. Target powers are linearly interpolated for intermediate channels.

Table B-5. **Group 5. Target Power Calibration for 802.11a Mode**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Test_channel_i								dB_6-24_i						...	
1	dB_36_i				dB_48_i						dB_54_i					

Target Power Calibration for 802.11b Mode

Group 6 represents the target powers for 802.11b mode. Table B-6 shows target power for 802.11b mode of operation for rates 1, 2, 5.5, and 11 Mbps, which are specified at exactly two test frequencies. For i=0–1, Test_channel_i represents frequency in eight bits, followed by the target power in dBm at various rates expressed in six bits (unsigned value).

Exactly two test frequencies must be specified. To cover the entire operating range, it is recommended to specify:

- Test_channel_0 = 2.412 GHz
- Test_channel_1 = 2.484 GHz

This formula relates the frequency in 2.4 GHz range (freq) to the 8-bit value stored on the EEPROM as Test_channel_i (fbin) is:

fbin = (freq - 2300) (freq in MHz)

Table B-6. **Group 6. Target Power Calibration for 802.11b Mode**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Test_channel_0								dB_6-24_i						...	
1	dB_36_i				dB_48_i						dB_54_i					
2	Test_channel_1								dB_6-24_i						...	
3	dB_36_i				dB_48_i						dB_54_i					

802.11g Target Power Calibration Information

Group 7 represents the target powers for 802.11g mode. Table B-7 shows the target power for 802.11g mode for rates 6–24, 36, 48, and 54 Mbps, which are specified at up to three test frequencies. All CCK rates in 802.11g mode use the target powers from 802.11b mode. For i=0–2, Test_channel_i represents frequency in eight bits, followed by the target power in dBm at various rates expressed in six bits (unsigned value).

Not all three test frequencies need to be specified. If less than three test frequencies are specified, the remaining unused bits should be 0. To cover the entire operating range, it is recommended to specify:

- Test_channel_0 = 2.412 GHz
- Test_channel_2 = 2.484 GHz

Table B-7. **Group 7. Target Power Calibration for 802.11g Mode**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Test_channel_0								dB_6-24_i						...	
1	dB_36_i				dB_48_i						dB_54_i					
2	Test_channel_1								dB_6-24_i						...	
3	dB_36_i				dB_48_i						dB_54_i					
4	Test_channel_2								dB_6-24_i						...	
5	dB_36_i				dB_48_i						dB_54_i					

CTL Information

Group 8 represents the data for up to 32 CTLs. Table B-8 shows the format of EEPROM data for one of the 32 CTLs: eight user defined band edges expressed in 8 bits, the maximum band edge power (6 bits—unsigned value) at each band edge and the non-edge flags for each CTL band edge. This maximum power (dBm) at band edge is used to limit power only at that channel if the non-edge flag is set to 0. If the non-edge flag is set to 1, that indicates the CTL frequency to not be a band edge and the corresponding dBmax_edge<ii> specified will limit power at a range of channels (See "Conformance Testing Limits" on page 6 for details). If less than eight unique band edges are provided, the rest of the band edge locations are filled with 0.

This format is repeated for all defined CTLs, up to a maximum number of 32.

The starting location for this section is immediately after Group 7. In eep_map = 1, only as many CTLs are stored on the EEPROM as are actually defined. Unlike the case for eep_map = 0, the unused CTL locations are not filled with 0s. That space can be utilized for storing EAR instead.

It is important to determine the mode (802.11a/802.11b/802.11g/TURBO) to which a CTL pertains, to correctly read back the band edge frequencies. The lower 3 bits of the CTL hex code are used to convey this information as described in Table 1-6.

Table B-8. **Group 8. CTL Information (0x1BF–0x2BE)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BandEdge1								BandEdge2							
1	BandEdge3								BandEdge4							
2	BandEdge5								BandEdge6							
3	BandEdge7								BandEdge8							
4	X	Flg1	dBmax_edge1						X	Flg2	dBmax_edge2					
5	X	Flg3	dBmax_edge3						X	Flg4	dBmax_edge4					
6	X	Flg5	dBmax_edge5						X	Flg6	dBmax_edge6					
7	X	Flg7	dBmax_edge7						X	Flg8	dBmax_edge8					

C

eep_map 2

This appendix provides EEPROM subsystem-specific information for eep_map value set to 2 (refer to Table 1-5 on page 11.)

EEPROM NIC or AP Specific Information (0x150–0x2BE)

Each wireless NIC or AP is required to comply with local emission regulations and 802.11 spectral limitations. Due to the sensitive nature of RF circuit design, for each manufactured NIC or AP to provide an optimal level of throughput performance and output power, it is essential to calibrate each device and store the raw performance capability information in the EEPROM.

For the transmit power control at any channel, the AR2413 must be programmed with a 128-entry power detector ADC (PDADC) table indexed in 0.5 dB steps. The PDADC table essentially conveys the calibration information on the power detector feedback voltage as measured by the built-in ADC. The ADC can be used with any combination of up to 4 gain values (1/2x, 1x, 2x, 4x) to cover a wide dynamic range. PDADC values for all pd_gain values used are spliced at appropriately transition levels to create the 128 entry PDADC table. PDADC values are stored for every 0.5 dB step in output power. At the transition levels, certain amount of overlap is maintained for both the pd_gains above and below. The transition levels and the overlap amount is programmed into the chip by the driver via the config file.

Snapshot of the raw power capability of the NIC or AP over the entire frequency range is stored in the EEPROM at up to 10 frequency piers for 802.11a mode if 802.11a mode is supported. For all intermediate channels, the 128-entry PDADC table is reconstructed by the driver from this snapshot at 10 frequency piers through interpolation using the PLA scheme.

At each frequency pier, an idealized power versus PDADC dependence for AR2413 based designs is shown in Figure C-1. A piecewise linear approximation of the dependence is stored on the EEPROM. To accurately cover the entire range of output power, calibration information for up to two pd_gain values is stored on the EEPROM for eep_map = 2. Typically pd_gains of 1x and 4x are sufficient to accurately cover the entire range.

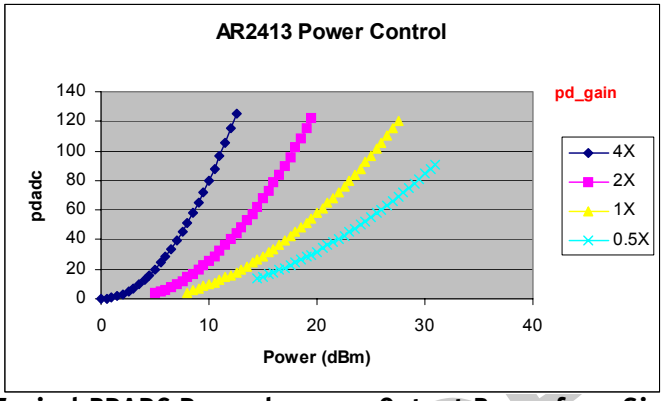

Figure C-1. **Typical PDADC Dependence on Output Power for a Given Channel**

The format for data stored at a frequency pier is summarized below. A piecewise linear approximation of PDADC vs. output power dependence is stored between appropriate transition levels for all the pd_gains used. Data at 4 intercepts is stored for all pd_gains except for the smallest one. The smallest pd_gain is used for the highest power levels and 5 intercepts are stored for this pd_gain for better accuracy.

As an example, for AR2413 based designs, only 2 pd_gains are used: 1x and 4x. Pd_gain = 1x is used for 11dBm and above power levels and 5 intercepts are stored for this pd_gain. Pd_gain = 4x is used for power levels smaller than 11 dBm and 4 intercepts are used for this pd_gain. An overlap of 5dB is used for both pd_gains around the transition level of 11 dBm.

The values stored for each frequency pier are:

- pwr_I_kk: lowest power level for all the pd_gains used. Typically, only two pd_gain values are used, so pwr_I_0 and pwr_I_1 are stored. This value is stored as a 5-bit value in 1 dB stepsize.
- Vpd_I_kk: PDADC value corresponding to the lowest power level for all the pd_gains used. Typically, two pd_gain values are sufficient to accurately cover the entire power range, so only Vpd_I_0 and Vpd_I_1 are stored. Vpd_I values are stored in a 7-bit unsigned integer format and represent the output of the power detector ADC.
- pwr_delta<ll>_kk: power step increment from the previous intercept for kk-th pd_gain. ll = 0...N where N = 3 for the smallest pd_gain (corresponding to highest power output) and N = 2 for all other pd_gains. pwr_delta values are stored as 4-bit values in 0.5 dB steps.
- Vpd_delta<ll>_kk: PDADC step increment from the previous intercept for kk-th pd_gain. ll = 0...N where N = 3 for the smallest pd_gain (corresponding to highest power output) and N = 2 for all other pd_gains. Vpd_delta values are stored as 6-bit unsigned integer values.

PDADC values for intermediate power levels are linearly interpolated from these sampling points.

802.11a Mode Calibration Information

One of the major differences between eep_map = 2 and eep_map = 0 EEPROM formats is that this section is conditionally stored on EEPROM for eep_map = 2, and is always present for eep_map = 0. In eep_map = 2, this section will be present only if AMode is set to 1 (i.e, 802.11a mode is supported). Starting location of this section is not fixed and must be read from "Cal_Data_Start_Location" stored in EEPROM location 0xC8 bits [15:4] (see Table 1-5).

First block of 5 EEPROM locations (Group 1) stores the pier locations for up to 10 frequency piers expressed in 8-bit frequency representation. Format of these 5 words is shown in Table C-1

Shown below is the formula used to relate the frequency in 5 GHz range (freq) to the 8-bit value stored on the EEPROM (fbin):

fbin = (freq - 4800) / 5 if 4800 ≤ freq < 6080

Table C-1. **Group 1. 802.11a Frequency Piers**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Fbin0								Fbin1							
1	Fbin2								Fbin3							
2	Fbin4								Fbin5							
3	Fbin6								Fbin7							
4	Fbin8								Fbin9							

Group 2 that follows Group 1 stores the calibration data for up to 10 frequency piers. Only as many piers are stored on the EEPROM that are actually measured, i.e, have a corresponding non-zero entry in the Group 1. Format of calibration data at each frequency pier is shown in Table C-2. It is important to note that calibration data for each pier occupies 6 EEPROM locations in this format for the case when 2 pd_gains are employed to cover the entire range of power levels. If any other number of pd_gains are employed, then the format will be modified according to the rules described above. It will be very similar: Pwr_I, Vpd_I for each pd_gain and 3 Pwr_delat and Vpd_delta for all pd_gains except the last one, which has 4. As a sanity check, 1 pd_gain will take 4 EEPROM locations, 2 pd_gains will take 6, 3 pd_gains will take 9 and 4 pd_gains will take 12 EEPROM locations to store calibration data for each pier.

All AR2413 based designs employ 2 pd_gains and store the calibration data shown in the format shown in Table C-2 for the supported modes (802.11g and 802.11b).

Table C-2. **Group 4. 802.11a Raw Power Calibration Data at a Frequency Pier For 2 pd_gains with eep_map = 2**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Pwr_delta0_0				Vpd_I_0							Pwr_I_0				
1	Vpd_delta1_0						Pwr_delta1_0				Vpd_delta0_0					
2	...	Pwr_I_1					Vpd_delta2_0						Pwr_delta2_0			
3	Vpd_delta0_1						Pwr_delta0_1				Vpd_I_1					
4	...		Pwr_delta2_1				Vpd_delta1_1						Pwr_delta1_1			
5	15	14	Vpd_delta3_1						Pwr_delta3_1				Vpd_delta2_1			

802.11b Mode Calibration Information

In eep_map = 2, group 4 stores the calibration data for up to four frequency piers in 802.11b mode. Only as many piers are stored on the EEPROM as are actually measured. For forward compatibility over EAR mechanism reasons, the three frequency piers specified in the EEPROM header locations 0xFE–0xFF are not used for eep_map = 2. Instead, a block of two EEPROM locations (Group 3) stores pier locations for up to four frequency piers expressed in 8-bit frequency representation. Table C-3 shows the two words' format.

This formula relates the frequency in 2.5 GHz range (freq) to the 8-bit value stored on the EEPROM (fbin):

fbin = (freq - 2300) for 2300 < freq ≤ 2555

A value of 0 for fbin indicates an unused pier. Format of calibration data at each frequency pier is shown in Table C-4.

In eep_map = 2, this section (Groups 3 and 4) will be present only if BMode is set to 1 (i.e, 802.11b mode is supported). If present, this section does not have a fixed starting location, instead, it starts wherever Group 2 ends. If Group 1 and Group 2 are not present, then the starting location of this section must be read from "Cal_Data_Start_Location" stored in EEPROM location 0xC8 bits [15:4] (see Table 1-5).

In eep_map = 2, there are some reserved locations after the end of the EEPROM header section that need to be maintained for forward compatibility over the EAR mechanism. Typically, there are 10 EEPROM locations reserved for this purpose, but this number may change in subsequent EEPROM formats after version 5.0. Cal_Data_Start_Location marks the end of these reserved locations and the beginning of the valid calibration data. In eep_map = 2, the start of calibration data should always be obtained from Cal_Data_Start_Location.

Table C-3. **Group 1. 802.11b Frequency Piers**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Fbin0								Fbin1							
1	Fbin2								Fbin3							

Table C-4. **Group 4. 802.11b Raw Power Calibration Data For 2 pd_gains for eep_map = 2**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Pwr_delta0_0				Vpd_I_0							Pwr_I_0				
1	Vpd_delta1_0						Pwr_delta1_0				Vpd_delta0_0					
2	...	Pwr_I_1					Vpd_delta2_0						Pwr_delta2_0			
3	Vpd_delta0_1						Pwr_delta0_1				Vpd_I_1					
4	...		Pwr_delta2_1				Vpd_delta1_1						Pwr_delta1_1			
5	15	14	Vpd_delta3_1						Pwr_delta3_1				Vpd_delta2_1			

802.11g Calibration Information

In eep_map = 2, group 6 stores the calibration data for up to four frequency piers in 802.11g mode. Only as many piers are stored on the EEPROM as are actually measured. For forward compatibility over EAR mechanism reasons, frequency piers specified in the EEPROM header locations 0x119 and 0x11B are not used for eep_map = 2. Instead, a block of two EEPROM locations (Group 5) stores pier locations for up to four frequency piers expressed in 8-bit frequency representation. Table C-3 shows these two words' format.

This formula relates the frequency in 2.5 GHz range (freq) to the 8-bit value stored on the EEPROM (fbin):

fbin = (freq - 2300) for 2300 < freq ≤ 2555

A value of 0 for fbin indicates an unused pier. Format of calibration data at each frequency pier is shown in Table C-4.

In eep_map = 2, this section (Groups 3 and 4) will be present only if GMode is set to 1 (i.e, 802.11g mode is supported). If present, this section does not have a fixed starting location, instead, it starts wherever Group 4 ends. If Group 1, Group 2, Group 3 and Group 4 are not present, then the starting location of this section must be read from "Cal_Data_Start_Location" stored in EEPROM location 0xC8 bits [15:4] (see Table 1-5).

In eep_map = 2, there are some reserved locations after the end of the EEPROM header section that need to be maintained for forward compatibility via the EAR mechanism. Typically, there are ten EEPROM locations reserved for this purpose, but this number may change in subsequent EEPROM formats after version 5.0. Cal_Data_Start_Location marks the end of these reserved locations and the beginning of the valid calibration data. In eep_map = 2, start of calibration data should always be obtained from the Cal_Data_Start_Location.

Table C-5. **Group 1. 802.11g Frequency Piers**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Fbin0								Fbin1							
1	Fbin2								Fbin3							

Table C-6. **Group 4. 802.11g Raw Power Calibration Data For 2 pd_gains for eep_map = 2**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Pwr_delta0_0				Vpd_I_0							Pwr_I_0				
1	Vpd_delta1_0						Pwr_delta1_0				Vpd_delta0_0					
2	...	Pwr_I_1					Vpd_delta2_0						Pwr_delta2_0			
3	Vpd_delta0_1						Pwr_delta0_1				Vpd_I_1					
4	...		Pwr_delta2_1				Vpd_delta1_1						Pwr_delta1_1			
5	15	14	Vpd_delta3_1						Pwr_delta3_1				Vpd_delta2_1			

Target Power Calibration for 802.11a Mode

In eep_map = 2, the starting location of this section (Groups 7, 8, and 9) depends upon the sizes of groups preceding it and is specified in the EEPROM location 0xC5. This section is always present and the size and format is the same as that for eep_map = 0.

Group 7 represents the target powers for 802.11a mode. Table C-7 shows target power for 802.11a mode of operation for 6–24, 36, 48, and 54 Mbps, specified at up to eight test frequencies. For i=0–7, Test_channel_i represents frequency in eight bits, followed by the target power in dBm at various rates in 6 bits (unsigned value). Not all eight test frequencies must be specified; if fewer than eight are specified, the remaining unused bits are 0. Target powers are linearly interpolated for intermediate channels.

Table C-7. **Group 7. Target Power Calibration for 802.11a Mode**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Test_channel_i								dB_6-24_i						...	
1	dB_36_i				dB_48_i						dB_54_i					

Target Power Calibration for 802.11b Mode

Group 8 represents the target powers for 802.11b mode. Table C-8 shows target power for 802.11b mode of operation for rates 1, 2, 5.5, and 11 Mbps, specified at exactly two test frequencies. For i=0–1, Test_channel_i represents frequency in eight bits, followed by the target power in dBm at various rates expressed in six bits (unsigned value).

Exactly two test frequencies must be specified. To cover the entire operating range, it is recommended to specify:

- Test_channel_0 = 2.412 GHz
- Test_channel_1 = 2.484 GHz

This formula relates the frequency in 2.4 GHz range (freq) to the 8-bit value stored on the EEPROM as Test_channel_i (fbin):

fbin = (freq - 2300) (freq in MHz)

Table C-8. **Group 8. Target Power Calibration for 802.11b Mode**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Test_channel_0								dB_6-24_i						...	
1	dB_36_i				dB_48_i						dB_54_i					
2	Test_channel_1								dB_6-24_i						...	
3	dB_36_i				dB_48_i						dB_54_i					

802.11g Target Power Calibration Information

Group 9 represents the target powers for 802.11g mode. Table C-9 shows target power for 802.11g mode for rates 6-24, 36, 48, and 54 Mbps, specified at up to three test frequencies. For all CCK rates in 802.11g mode, the target powers from 802.11b mode are used. For i=0–2, Test_channel_i represents frequency in 8 bits, followed by the target power in dBm at various rates expressed in 6 bits (unsigned value).

Not all three test frequencies need to be specified. If less than three test frequencies are specified, the remaining unused bits should be 0. To cover the entire operating range, it is recommended to specify:

- Test_channel_0 = 2.412 GHz
- Test_channel_2 = 2.484 GHz

Table C-9. **Group 9. Target Power Calibration for 802.11g Mode**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Test_channel_0								dB_6-24_i						...	
1	dB_36_i				dB_48_i						dB_54_i					
2	Test_channel_1								dB_6-24_i						...	
3	dB_36_i				dB_48_i						dB_54_i					
4	Test_channel_2								dB_6-24_i						...	
5	dB_36_i				dB_48_i						dB_54_i					

CTL Information

Group 10 represents the data for up to 32 CTLs. Table C-10 shows the format of EEPROM data for one of the 32 CTLs: eight user defined band edges expressed in 8 bits, the maximum band edge power (6 bits—unsigned value) at each band edge and the non-edge flags for each CTL band edge. This maximum power (dBm) at band edge is used to limit power only at that channel if the non-edge flag is set to 0. If the non-edge flag is set to 1, that indicates the CTL frequency to not be a band edge and the corresponding dBmax_edge<ii> specified will limit power at a range of channels (See "Conformance Testing Limits" on page 6 for details). If less than 8 unique band edges are provided, the rest of the band edge locations are filled with 0.

This format is repeated for all defined CTLs, up to a maximum number of 32.

This section starts immediately after Group 7. In eep_map = 1, only as many CTLs are stored on the EEPROM as are actually defined. Unlike eep_map = 0, unused CTL locations are not filled with 0s, but can be used to store EAR.

It is important to determine the mode (802.11a/802.11b/802.11g/11a TURBO/11g TURBO) a CTL pertains to, to correctly read back band edge frequencies. The lower three bits of CTL hex code convey this information (see Table 1-6).

Table C-10. **Group 10. CTL Information (0x1BF–0x2BE)**

Location	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BandEdge1								BandEdge2							
1	BandEdge3								BandEdge4							
2	BandEdge5								BandEdge6							
3	BandEdge7								BandEdge8							
4	X	Flg1	dBmax_edge1						X	Flg2	dBmax_edge2					
5	X	Flg3	dBmax_edge3						X	Flg4	dBmax_edge4					
6	X	Flg5	dBmax_edge5						X	Flg6	dBmax_edge6					
7	X	Flg7	dBmax_edge7						X	Flg8	dBmax_edge8					

D

EEPROM Contents

This appendix presents a sample **atheros-eep.txt** file, which describes the contents of each location.

Description of Format

```
;Atheros Reference Card EEPROM contents
;Version 2.1 7/29/2002
;--------------------------------------------------------------------------
;REVISION HISTORY
;--------------------------------------------------------------------------
;Ver Revision History                                                    Date
;
;1.00 Initial Relase of Tuples                                          10/12/00
;1.01 Small fixes to tuples. Inusre 1MB for F2 Proto's using 1MB.       10/16/00
;1.02 Changed name of file, updated CISTPL_BAR to reflect 64KB for D2+. 12/12/00
;1.03 Changed CISTPL_VERS1 from Card Access information to Atheos info. 12/22/00
;1.04 Updated CISTPL_VERS1 with Atheros PN info: AR5BCB-01-01           01/15/01
;1.05 Reformatted the VENDOR SPECIFIC comment banner to work            01/29/01
;       with eepnumfix.pl. Fixed PN to AR5BCB-01-01.
;1.06 Fix the Link Tuple on CISTPL_VERS1 to take into account the added 02/05/01
;       -01-01.
;1.07 Above edit only fixed the comment. Fixed the LINK Tuple data.    02/07/01
;1.08 Changed the PMData_D0 and PMData_D3 to C606 = D0=1.98W, D3=.06W.03/09/01
;1.09 Added write protect at 0x3f.                                      03/22/01
;1.10 Fixed write protect value: 0x0005 = Write Only /No Read,          04/04/01
;       0x000A = Read Only / No Write.
;1.11 Added 'EERPOM_MAGIC' value to 0x3D, Added comments for            04/11/01
;       MAC words 0-2.
;1.12 Added Static and Avg current to CISTPL_CFTABLE_CB.                04/16/01
;1.13 Added reserved area for Atheros initialization info E0-FF         05/08/01
;1.14 Zeroed protect bits and added MIN_GNT/MAX_LAT                     05/30/01
;1.15 Added offset 0x0F Clkrunen (1 = Enable, 0=disabled) in Sleep Mode 06/01/01
;1.16 Updated Atheros PN and PCMCIA Revision to 7.1.                    07/09/01
;       Added Subsys ID's 168c and 0007
;1.17 Documented last 128B with the new EEPROM setup information        07/30/01
```

```
;1.18 Set location 0x3E to 0001 to protect key cache from SW read       08/10/01
;1.19 Added Vendor Data area in EEPROM in locations B0-BE               08/16/01
;1.20 Changed comments to reflect data sheet names                      08/21/01
;1.21 Rechanged version back to 01-01 for PCMCIA compatibility          08/28/01
;2.0  Added description to 0xf for RF_SILIENT                          07/18/02
;2.1  DeviceID = 0x0012, SSID=0x0000, fixed the 6th tuple, DeviceID
;     reflected correctly in 2nd tuple.                                 07/29/02
;
;--------------------------------------------------------------------------
;DESCRIPTION OF FORMAT:
;--------------------------------------------------------------------------
;EEPROM has 256 locations with 16 bits per location
;
;for CONFIGURATION...
;double byte per line for now
;bytes are in 'readable' order
;the parser will swap the bytes for programming into the EEPROM (little-endian)
;
;for TUPLES ...
;single byte per line - single byte easier to read, & mk notes
;bytes are in the same order as one would expect to see tuples in
;the parser swaps the bytes for burning into the EEPROM
;the 'above' byte goes into bits 7-0 and the 'below' byte goes into bits 15-8
;so a
;13
;03
;would go into the EEPROM as 0313
;ascii-text version of the hex numbers w/o 0x before or h after
;the 0 or F fill lines after tuples (if tpl don't use space) are dbl byte/line
;
;for VENDOR SPECIFIC DATA
;double byte per line for now
;bytes are in 'readable' order
;the parser will swap the bytes for programming into the EEPROM (little-endian)
;
;PCI configuration data first in 00-3F locations (64 WORDS)
;CIS information next in 40-BE locations (112 WORDS)
;VENDOR OEM information next in B0-BE locations (15 WORDS)
;Device specific information next in BF-FF locations (65 WORDS)
;ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
;PCI CONFIGURATION LOCATIONS 00-3F
;ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
0012    ;00;    Device_ID
168C    ;01;    Vendor_ID
0200    ;02;    Class Code (24 b) 23:8 of code -> 15:0 of word
0001    ;03;    Class Code         7:0 of code -> 15:8 of word, Revision ID -> 7:0
0000    ;04;
5001    ;05;    CIS_PTR (total of 32 bits) [15:0] (the lower 16 bits here)
0000    ;06;    CIS_PTR                    [31:16](the upper 16 bits here)
0000    ;07;    SSYS_ID (Subsystem ID)
168C    ;08;    SSYS_VEND_ID (Subsystem Vendor ID)
1C0A    ;09;    MAX_LAT [15:8], MIN_GNT [7:0]
0100    ;0a;    INT_PIN [15:8], 0'S [7:0]
0000    ;0b;    Reserved
0002    ;0c;    PM_CAP (Power Management Capabilities)
0002    ;0d;    0'S [15:2], PM_DATA_SCALE [1:0]
C606    ;0e;    PM_DATA_D0 [15:8], PM_DATA_D3 [7:0]:C606 = 1.98W in D0, .06W in D3
0001    ;0f;    CLKRUN_ENABLE[0], (Enable = 1, Disable = 0), RFSILENT_POLARITY[1],
RFSILENT_GPIO_SEL[4:2]
0000    ;10;    RESERVED
0000    ;11;    RESERVED
```

```
0000    ;12;    RESERVED
0000    ;13;    RESERVED
0000    ;14;    RESERVED
0000    ;15;    RESERVED
0000    ;16;    RESERVED
0000    ;17;    RESERVED
0000    ;18;    RESERVED
0000    ;19;    RESERVED
0000    ;1a;    RESERVED
0000    ;1b;    RESERVED
0000    ;1c;    RESERVED
0000    ;1d;    MAC Address LSW (15:0)
0000    ;1e;    MAC Address (31:16)
0000    ;1f;    MAC Address MSW (47:32)
0000    ;20;    RESERVED
0000    ;21;    RESERVED
0000    ;22;    RESERVED
0000    ;23;    RESERVED
0000    ;24;    RESERVED
0000    ;25;    RESERVED
0000    ;26;    RESERVED
0000    ;27;    RESERVED
0000    ;28;    RESERVED
0000    ;29;    RESERVED
0000    ;2a;    RESERVED
0000    ;2b;    RESERVED
0000    ;2c;    RESERVED
0000    ;2d;    RESERVED
0000    ;2e;    RESERVED
0000    ;2f;    RESERVED
0000    ;30;    RESERVED
0000    ;31;    RESERVED
0000    ;32;    RESERVED
0000    ;33;    RESERVED
0000    ;34;    RESERVED
0000    ;35;    RESERVED
0000    ;36;    RESERVED
0000    ;37;    RESERVED
0000    ;38;    RESERVED
0000    ;39;    RESERVED
0000    ;3a;    RESERVED
0000    ;3b;    RESERVED
0000    ;3c;    RESERVED
5AA5    ;3d;    EEPROM_MAGIC Word
0001    ;3e;    KEY_TABLE_RD_PROTECT [0]
0000    ;3f;    EEPROM Protection [15:0]. 0x0A write protects location 0 - 0x3F.

;ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
;CIS LOCATIONS 40-AF
;ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
;32-bit memory with even word in lower 16 bits and odd word in upper 16 bits
;this space does not correspond to the PCI configuration space
;
;1st tuple
13      ;40;    13=LINKTARGET [3.1.4]
03      ;40;    03=Link to next tuple chain
43      ;41;    "C"
49      ;41;    "I"
53      ;42;    "S"
```

```
;2nd tuple
20      ;42;    20=MANFID, Manufacturer ID tuple [3.2.9]
04      ;43;    04=Link to next tuple chain
71      ;43;
02      ;44;    0271=Atheros manufacturing code
12      ;44;
00      ;45;    0007=Atheros Device ID for AR5001

;3rd tuple
04      ;45;    04=CONFIG_CB, configuration for a cardbus card [3.3.5]
06      ;46;    06=Link to next tuple chain
03      ;46;    size of fields - must be 3 to represent 4 bytes per field
01      ;47;    indx # of last entry in card config table -> see 4th tuple below
00      ;47;
00      ;48;
00      ;48;
00      ;49;    address of 0000 0000 because we don't have any status registers

;4th tuple
05      ;49;    05=CFTABLE_ENTRY_CB, config table entry for cardbus card [3.3.3]
0E      ;4a;    0E=Link to next tuple chain
41      ;4a;    bit 6 set = the default config; 01=last entry in config table
B1      ;4b;    b1:0 = 01 = VCC power description only
        ;    b3 = 0 for no I/O space description
        ;    b4 = 1 for interrupt structure description
        ;    b5 = 1 for memory space description
        ;    b7 = 1 for a miscellaneous field structure
39      ;4b;    b0 = 1 for nominal operating voltage
        ;      b3 = 1 for static current
        ;      b4 = 1 for average current
        ;      b5 = 1 for peak current
B5      ;4c;    b6:3=mant=0110=6->3 b2:0=exp=101=5->1V range -> 3V for VCC power
1E      ;4c;    1E hex = 30 decimal, goes to rt of above value, so total VCC = 3.3
2D     ;4d;    b6:3=mant=0101=5->2.5 b2:0=exp=101=4=10ma Static Current = 25ma for 3.3V
4E      ;4d;    b6:3=mant=1001=9->4.5 b2:0=exp=110=6=100mA Avg Current = 450ma.
56      ;4e;    b6:3=mant=1010=A->5 b2:0=exp=110=6=100mA Avg Current = 500ma.
30      ;4e;    b7=0 for NO INTERRUPT SHARING among several cards
        ;     b6=0 for no pulse interrupts
        ;     b5=1 for level interrupts
        ;     b4=1 for linking the int to any int shown by the following 2 bytes
FF      ;4f;    interrupts 7-0
FF      ;4f;    interrupts 15-8
02      ;50;    b1=1 for first BAR to use for memory
E9      ;50;    b7=1 for there is one more byte -> required by METAFORMAT.
        ;    b6=1 for fast back-to-back supported
        ;    b5=1 for SERR# supported
        ;    b4=0 for no wait cycle control supported
        ;    b3=1 for Parity error response
        ;    b2=0 for no VGA palette snoop
        ;    b1=0 for no memory write & invalidate
        ;    b0=1 for Bus Master
00      ;51;    no wake up events
```

```
;5th tuple
07      ;51;    07=CISTPL_BAR Base address register [3.3.1]
06      ;52;    06=Link to next tuple
01      ;52;    01 = BAR1, mem space, no prefetch, no cache, no mapping restriction
00      ;53;    RESERVED - must be 0
00      ;53;    Base Address Register Size - 4 bytes altogether
00      ;54;    Size in Bytes of the memory space the BAR is mapped to
01      ;54;    Card has 64KB memory space -> 2^16 = 64KB
00      ;55;    0001 0000 = 65,536 of memory

;6th tuple
15      ;55;    15=CISTPL_VERS_1 Level 1 version and product informatoin [3.2.10]
52      ;56;    50=Link to next tuple
07      ;56;    07=major version 7
01      ;57;    01=minor version 1 -> both bytes together mean version 7.1
41   ;57;   A
74   ;58;   t
68   ;58;   h
65   ;59;   e
72   ;59;   r
6f   ;5a;   o
73   ;5a;   s
20   ;5b;   <space>
43   ;5b;   C
6f   ;5c;   o
6d   ;5c;   m
6d   ;5d;   m
75   ;5d;   u
6e   ;5e;   n
69   ;5e;   i
63   ;5f;   c
61   ;5f;   a
74   ;60;   t
69   ;60;   i
6f   ;61;   o
6e   ;61;   n
73   ;62;   s
2c   ;62;   ,
20   ;63;   <space>
49   ;63;   I
6e   ;64;   n
63   ;64;   c
2e   ;65;   .
00   ;65;   <null>
41   ;66;   A
52   ;66;   R
35   ;67;   5
30   ;67;   0
30   ;68;   0
31   ;68;   1
2d   ;69;   -
30   ;69;   0
30   ;6a;   0
30   ;6a;   0
30   ;6b;   0
2d   ;6b;   -
30   ;6c;   0
30   ;6c;   0
30   ;6d;   0
30   ;6d;   0
```

```
00   ;6e;   <null>
57   ;6e;   W
69   ;6f;   i
72   ;6f;   r
65   ;70;   e
6c   ;70;   l
65   ;71;   e
73   ;71;   s
73   ;72;   s
20   ;72;   <space>
4c   ;73;   L
41   ;73;   A
4e   ;74;   N
20   ;74;   <space>
52   ;75;   R
65   ;75;   e
66   ;76;   f
65   ;76;   e
72   ;77;   r
65   ;77;   e
6e   ;78;   n
63   ;78;   c
65   ;79;   e
20   ;79;   <space>
43   ;7a;   C
61   ;7a;   a
72   ;7b;   r
64   ;7b;   d
00   ;7c;   <null>
30   ;7c;   0
30   ;7d;   0
00   ;7d;   <null>
FF      ;7e;    end of tuple
00  ;7e;    allow for version extension
00      ;7f;    Place MAC Address on even boudnary

;7th tuple
21      ;7f;    21=CISTPL_FUNCID - describes the cards function [3.2.7]
02      ;80;    02=Link to next Tuple
06      ;80;    06=function Network Adapter
01      ;81;    01=POST routines may attempt to config card during system init

;8th tuple
22      ;81;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;82;    05=Link to next Tuple
02      ;82;    Lan Speed information follows - raw bit rate
80      ;83;    32 bit integer value for the raw speed
8D      ;83;    format is 808D5B00 is read as 00 5B 8D 80
5B      ;84;
00      ;84;    005B 8D80 = 6,000,000 bits per second

;9th tuple
22      ;85;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;85;    05=Link to next Tuple
02      ;86;    Lan Speed information follows - raw bit rate
40      ;86;
54      ;87;
89      ;87;
00      ;88;    0089 5440 = 9,000,000 bits per second
```

```
;10th tuple
22      ;88;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;89;    05=Link to next Tuple
02      ;89;    Lan Speed information follows - raw bit rate
00      ;8a;
1B      ;8a;
B7      ;8b;
00      ;8b;    00B7 1B00 = 12,000,000 bits per second

;11th tuple
22      ;8c;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;8c;    05=Link to next Tuple
02      ;8d;    Lan Speed information follows - raw bit rate
80      ;8d;
A8      ;8e;
12      ;8e;
01      ;8f;    0112 A880 = 18,000,000 bits per second

;12th tuple
22      ;8f;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;90;    05=Link to next Tuple
02      ;90;    Lan Speed information follows - raw bit rate
00      ;91;
36      ;91;
6E      ;92;
01      ;92;    016E 3600 = 24,000,000 bits per second

;13th tuple
22      ;93;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;93;    05=Link to next Tuple
02      ;94;    Lan Speed information follows - raw bit rate
00      ;94;
51      ;95;
25      ;95;
02      ;96;    0225 5100 = 36,000,000 bits per second

;14th tuple
22      ;96;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;97;    05=Link to next Tuple
02      ;97;    Lan Speed information follows - raw bit rate
00      ;98;
6C      ;98;
DC      ;99;
02      ;99;    02DC 6C00 = 48,000,000 bits per second

;15th tuple
22      ;9a;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;9a;    05=Link to next Tuple
02      ;9b;    Lan Speed information follows - raw bit rate
80      ;9b;
F9      ;9c;
37      ;9c;
03      ;9d;    0337 F980 = 54,000,000 bits per second
```

```
;16th tuple
22      ;9d;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
05      ;9e;   ; 05=Link to next Tuple
02      ;9e;    Lan Speed information follows - raw bit rate
00      ;9f;
A2      ;9f;
4A      ;a0;
04      ;a0;    044A A200 = 72,000,000 bits per second

;17th tuple
22      ;a1;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
02      ;a1;    02=Link to next Tuple
03      ;a2;    LAN technology tuple
08      ;a2;    Spread spectrum radio 5.4 GHz
;we are actually OFDM, not spread spectrum

;18th tuple
22      ;a3;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
08      ;a3;    08=Link to next Tuple
04      ;a4;    LAN network ID
06      ;a4;    06=6 bytes in the network ID
00      ;a5;
03      ;a5;
7F      ;a6;    00 03 7F XX XX XX
00      ;a6;
00      ;a7;
00      ;a7;

;19th tuple
22      ;a8;    22=CISTPL_FUNCE for function extension tuple [3.2.6]
02      ;a8;    02=Link to next Tuple
05      ;a9;    LAN connection type
01      ;a9;    Closed connector standard

;20th tuple
FF      ;aa;    End of tuples

;Tuple extra area - blank - double word zeros.
00      ;aa;   Extra byte to start on even word boundary.
0000    ;ab;
0000    ;ac;
0000    ;ad;
0000    ;ae;
0000    ;af;
;ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
;VENDOR DEFINED LOCATIONS B0-BE
;ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
0000    ;b0;
0000    ;b1;
0000    ;b2;
0000    ;b3;
0000    ;b4;
0000    ;b5;
0000    ;b6;
0000    ;b7;
0000    ;b8;
0000    ;b9;
0000    ;ba;
0000    ;bb;
0000    ;bc;
```

```
0000    ;bd;
0000    ;be;
; THE FOLLOWING VALUES ARE COMMENTED OUT AS THEY SHOULD BE WRITTEN
; BY THE CALIBRATION SCRIPT
;ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
;ATHEROS DEFINED LOCATIONS BF-FF
;ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
; 0000  ;bf;  Regulatory Domain Code Information
; 0000  ;c0;  Checksum for locations C0-FF
; 0000  ;c1;  Version Field
; 0000  ;c2;  Antenna Settings
; 0000  ;c3;  Transmit Power Bias Currents
; 0000  ;c4;  Threshold 62 and external LNA Timing Settings
; 0000  ;c5;  External Amplifier Timing Settings
; 0000  ;c6;  Reserved
; 0000  ;c7;  Regulatory Domain 1
; 0000  ;c8;  Regulatory Domain 2
; Channel 5.17 to 5.19 Transmit Power Calibration
; 0000  ;c9;  Transmit Power Calibration
; 0000  ;ca;  Transmit Power Calibration
; 0000  ;cb;  Transmit Power Calibration
; 0000  ;cc;  Transmit Power Calibration
; 0000  ;cd;  Transmit Power Calibration
; 0000  ;ce;  Transmit Power Calibration
; 0000  ;cf;  Transmit Power Calibration
; 0000  ;d0;  Transmit Power Calibration
; 0000  ;d1;  Regulatory Domain Code Limit
; 0000  ;d2;  Regulatory Domain Code Limit
; 0000  ;d3;  Regulatory Domain Code Limit
; Channel 5.20 to 5.22 Transmit Power Calibration
; 0000  ;d4;  Transmit Power Calibration
; 0000  ;d5;  Transmit Power Calibration
; 0000  ;d6;  Transmit Power Calibration
; 0000  ;d7;  Transmit Power Calibration
; 0000  ;d8;  Transmit Power Calibration
; 0000  ;d9;  Transmit Power Calibration
; 0000  ;da;  Transmit Power Calibration
; 0000  ;db;  Transmit Power Calibration
; 0000  ;dc;  Regulatory Domain Code Limit
; 0000  ;dd;  Regulatory Domain Code Limit
; 0000  ;de;  Regulatory Domain Code Limit
; Channel 5.23 to 5.25 Transmit Power Calibration
; 0000  ;df;  Transmit Power Calibration
; 0000  ;e0;  Transmit Power Calibration
; 0000  ;e1;  Transmit Power Calibration
; 0000  ;e2;  Transmit Power Calibration
; 0000  ;e3;  Transmit Power Calibration
; 0000  ;e4;  Transmit Power Calibration
; 0000  ;e5;  Transmit Power Calibration
; 0000  ;e6;  Transmit Power Calibration
; 0000  ;e7;  Regulatory Domain Code Limit
; 0000  ;e8;  Regulatory Domain Code Limit
; 0000  ;e9;  Regulatory Domain Code Limit
; Channel 5.26 to 5.29 Transmit Power Calibration
; 0000  ;ea;  Transmit Power Calibration
; 0000  ;eb;  Transmit Power Calibration
; 0000  ;ec;  Transmit Power Calibration
; 0000  ;ed;  Transmit Power Calibration
; 0000  ;ee;  Transmit Power Calibration
; 0000  ;ef;  Transmit Power Calibration
; 0000  ;f0;  Transmit Power Calibration
```

```
; 0000  ;f1;  Transmit Power Calibration
; 0000  ;f2;  Regulatory Domain Code Limit
; 0000  ;f3;  Regulatory Domain Code Limit
; 0000  ;f4;  Regulatory Domain Code Limit
; Channel 5.30 to 5.32 Transmit Power Calibration
; 0000  ;f5;  Transmit Power Calibration
; 0000  ;f6;  Transmit Power Calibration
; 0000  ;f7;  Transmit Power Calibration
; 0000  ;f8;  Transmit Power Calibration
; 0000  ;f9;  Transmit Power Calibration
; 0000  ;fa;  Transmit Power Calibration
; 0000  ;fb;  Transmit Power Calibration
; 0000  ;fc;  Transmit Power Calibration
; 0000  ;fd;  Regulatory Domain Code Limit
; 0000  ;fe;  Regulatory Domain Code Limit
; 0000  ;ff;  Regulatory Domain Code Limit
```

E

EAR EEPROM

This appendix contains a scheme for updating hardware register settings from values contained in the EEPROM.

Hardware Design Guidelines

Here are a few basic guidelines with respect to future chip hardware bug fixes as well as feature implementation. Software release 2.4 is regard as a hard limit for any dynamic changes for the hardware as the EAR can only overwrite or add new register writes.

- Do not shift portions (or all) of the register space. If you need a new register, add it to an unused register address. Do not move 30 registers to keep similar functioning registers near each other.
- Do not grow the large analog banks - instead make use of previously undefined bits in that bank. All large analog writes should be considered only a modification of the currently existing register.
- When changing a parameter, try to keep the current definition compatible with the previous software values (1 new write is better than two).
- Avoid adding multi-parameter registers that read to give one parameter, but write to perform a different function on that parameter as these may cause problems if a read-modify-write is performed for other parameters in this register.

- Do not change timing sensitive registers such as: PLL, reset register, calibration registers.
- Do not change registers that get dynamically changed by software, (e.g, noise immunity registers, transmit power registers, gainI, gainF parameters).

Design

The key concepts for the EEPROM Added Registers (EAR) include:

- Provide EEPROM/software synchronized versioning so that software can ignore outdated EAR.
- Provide a few locations in the reset and calibration code to insert registers from the EEPROM.
- Allow EEPROM modifications to be modal in that they can apply to any of our planned operating modes: 802.11a, 802.11b, 802.11g, 5 GHz Turbo, 2 GHz Turbo, with an XR flag or applied only to specific channels.
- Provide a version mask so that changes that are common to more than one software release can be shared.
- New to version 5.2 of the EEPROM, is the concept of a dynamic EAR. The dynamic EAR contents follow the same rules as the regular EAR, however it is automacilly generated by the manufacturing software during calibration. It contains the EEP_MAP type 2 calibration information and is needed for drivers that don't know how to support this EEP_MAP. The dynamic EAR is added to the end of the EAR file information supplied during calibration.

Placement into the Current EEPROM

The current EEPROM calibration uses beyond the first kilobyte of the EEPROM. The EAR will begin at the end of the CTL information of the current EEPROM. Most EEPROM calibration expansion is already built between the various calibration sections. Location 0x0C4 will include an EEPROM location offset for the first entry of the EAR, EARSTART. EARSTART indicates whether the EAR exists or not. If the first bit in EARSTART is 0, EAR does not exist; a non-zero value indicates the existence of an EAR. The EAR section is already included in the Checksum computed by the calibration information. Table E-1 summarizes the EEPROM layout with EAR.

Table E-1. **EEPROM Layout with EAR**

EEPROM Location	Description
0x000-0x03F	PCI configuration data.
0x040-0x0AF	Card Information Structure tuples.
0x0B0-0x0BE	Vendor OEM information.
0x2C0-EARSTART -1	Calibration information.
EARSTART-earend	EEPROM added registers.

EAR Header

The Header for the EAR contains the version identifier and then the first version mask/register header sections.

Table E-2. **EAR Header**

EEPROM Location	Bit 15	Bits 14-0
EARSTART	Version Identifier	
EARSTART +1	1	Version Mask
EARSTART +2	0	Register Header

Version Identifier

Identifies the version corresponding to bit 0 of the version mask scheme. This attribute allows the version mask to be a sliding mask.

Each version of software will have a software version identifier. If the software version identifier falls between the EAR version identifier and the EAR version identifier plus 14, then this EAR can apply to this version of software. The matched version between software and the EAR will increment when a software release has different "built in values" than the previous software release (including when new parameters are added).

Following the EAR Header the rest of the EAR is searched sequentially with each location indicating the type of location that follows. After each set of register writes the bit 15 identifies whether a new Version Mask or a Register Header begins. A version mask is always followed by a Register Header.

Table E-3. **Example EAR Framing**

EAR Header (Version ID)	
1	**Version Mask**
0	**Register Header**
Register Writes	
0	**Register Header**
Register Writes	
1	**Version Mask**
0	**Register Header**
Register Writes	
. . .	

Version Mask

A version mask must follow the version identifier to identify the version mask for subsequent register sections. The version mask requires a set of register blocks framed between two version masks to apply to the first version mask (i.e., a given version of software should apply all register blocks that start with a version identifier whose mask bit position plus the EAR version identifier equals the software version identifier).

Example

The software version is 18 and the EAR version identifier is 15. The software should apply all register blocks framed by a version mask where bit 3 is set.

Table E-4. **Version Mask**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Meaning	1	Version Mask														

Register Blocks

Each block of register settings has two sections: register block header, and register writes. Because of the limited amount of register-write space, a tight encoding format is needed to keep the register modifications small yet versatile.

Table E-5. **Register Header**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Meaning	0	D	CM	Stage		Type		Register Modality Mask								

The Register Header is fixed at 16 bits. The Register Writes description follows in the next EEPROM location. A zeroed register header signals the end of the EAR section of the EEPROM (the modality mask ensures that at least one bit is set in a valid register header).

Channel Modifier (CM)

A channel modifier mask extends the register header one 16-bit location and actively controls which channel(s) the register block should modify. If bit 15 of the first location is clear, the channel modifier mask should be interpreted as shown in the diagram. If bit 15 of the first location is set, then the following 15 bits should be read as a single channel specified in MHz for applying the following register writes.

Table E-6. **Channel Modifier (Extended Register Header)**

Bit	7	6	5	4	3	2	1	0
Desc	4900-5160	2300-2407	2484	2472	2462-2467	2442-2457	2417-2437	2412
Bit	15	14	13	12	11	10	9	8
Desc	MHz Channel	If 32 MHz Spur	5725-5825	5500-5700	5310-5320	5260-5300	5190-5250	5170-5180

Disabler (D)

The disabler mask follows the channel modifier, if present, or the register header. The disabler is subject to the version, register header, and channel modifier (if present) specification before it. When those conditions match, the disabler will disable analog bank writes or enable/disable calibration settings.

Table E-7. **The Disabler**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Meaning	P	X	C5	C4	C3	C2	C1	C0	B7D	B6D	B5D	B4D	B3D	B2D	B1D	B0D

Table E-8 summarizes the bit description for the Disabler.

Table E-8. **The Disabler Bit Descriptions**

Bit(s)	Description
B0D-B7D	Each bit set disables the write of the corresponding analog bank from Bank 0 to Bank 7. (NOTE: This is not implemented in the 2.4 software release of the EAR.)
C0	Enables/Disables the reset DC offset calibration.
C1	Enables/Disables the reset noise floor calibration.
C2	Enables/Disables the IQ phase mismatch calibration.
C3	Enables/Disables the Tx (gain_f read back) fixed gain periodic calibration.
C4	Enables/Disables the noise floor periodic calibration.

Table E-8. **The Disabler Bit Descriptions**

Bit(s)	Description
C5	Enables/Disables the gain circulation (gain_I preset) mechanism. (NOTE: This is not implemented in the 2.4 software release of the EAR.)
P	Specifies a replacement write for the PLL register follows in the next 16-bit location. Only the first PLL modifier for a given mode/flag set is used, i.e., once a matching PLL modifier is found, the code stops searching.

Stage

The stage selects where the register write(s) described by this header will be inserted in the software code. Table E-9 summarizes the bit description for the Stage.

Table E-9. **Stage Bit Descriptions**

Bit(s)	Description
b00	Register writes inserted during the reset analog register write (except channel).
b01	Register writes inserted after channel setup but before the PHY enable - no type 2 writes allowed for this stage.
b10	Register writes inserted after all calibrations of reset are completed - no type 2 writes allowed for this stage.
b11	Register writes inserted at the end of the calibration task - no type 2 writes allowed for this stage.

Register writes are not checked for writes to the same register address. Writes to the same register address can appear across different stages or even within the same stage. All register writes will be made in the order they are parsed from the EAR (with regard to the stage selected, i.e., there is no duplicate address removal.

Type

The register Type controls the format for the register writes following the register header. Table E-10 summarizes the bit description for the Type.

Table E-10. **Type Bit Descriptions**

Bit(s)	Description
b00	Sets of 16-bit addresses are followed by 32-bit values.
b01	One 16-bit address is contained following a variable number of 32-bit values. The addresses for the values after the first are consecutive register writes (i.e., the register address should be increased by 4 for each value after the first).
b10	The register writes are descriptions of modifications for an analog register.
b11	The register writes are descriptions of register modifications that use a read modify write to place their contents into a register.

Register Modality Mask

Selects all the modes that this register write block should modify. Table E-11 summarizes the modes.

Table E-11. **Registry Modality Modes**

Mode	Description
x001	802.11b
x002	802.11g
x004	802.11g Turbo
x008	Reserved
x100	Reserved
x010	802.11a
X020	Extended Range (XR) Additive Flag
x040	802.11a Turbo
x080	Reserved

Table E-12 summarizes the Type 0, explicitly addressed register write.

Table E-12. **Register Write: Explicitly Addressed (Type 0)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit	First Address														Tag	
Bit	Data (most significant word; if whole word)															
Bit	Data (least significant word)															
Bit	Second Address														Tag	
Bit	Data (most significant word; if whole word)															
Bit	Data (least significant word)															
. . .																
Bit	Last Address														Tag = 3	
Bit	Data (most significant word; if whole word)															
Bit	Data (least significant word)															

Type 0 writes can setup any number of addresses that have the same wireless mode modality and register write location. The register writes should be parsed until a register write has the Tag equal to 3. The next EEPROM location will contain a new register header or version mask.

Tag

A modifier for each explicitly addressed register write:

- 0: A whole-word write
- 1: A half-word read-modify-write replacing the lower 16 bits of the word.
- 2: A half-word read-modify-write replacing the upper 16 bits of the word.
- 3: A whole-word write. The last write in the register block.

Table E-13 summarizes the Type 1, sequential addresses register write.

Table E-13. **Register Write: Sequential Addresses (Type 1)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit	First Address														Num	
Bit	Data (most significant word)															
Bit	Data (least significant word)															
Bit	Data (most significant word)															
Bit	Data (least significant word)															
Bit	Data (most significant word)															
Bit	Data (least significant word)															
Bit	Data (most significant word)															
Bit	Data (least significant word)															

Type 1 writes are used to setup a group of up to 4 consecutive addresses. The register writes should be parsed for Num + 1 register writes. The implementation will add 4 to the address for each consecutive write. The next EEPROM location will contain a new register header or version mask.

Table E-14 summarizes the Type 2 analog register modifications.

Table E-14. **Register Write: Analog Register Modifications (Type 2)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Desc	Analog Bank			L	Column		E	Starting Bit								
Desc	Number of Bits				Data											
or Extended Data Mode																
Desc	Number of Bits															
Desc	Data															
Desc	. . .															
Desc	Data															

The Type 2 writes modify an existing software analog register. The **Analog Bank**, **Number of Bits**, **Starting Bit**, **Column**, and **Data** are the common elements used to describe an analog parameter. The given **Data** will be bit reversed, shifted, and masked on top of the given analog bank. When the **E**xtended bit is cleared - only up to 12 bits can be modified. When the **E**xtended bit is set, the next EEPROM location will contain the Number of Bits. The **Number of Bits** field controls how many **Data** locations follow the **Number of Bits** field. There will be **Number of Bits** /16 (rounded up) **Data** locations.

The **L**ast bit controls when this register block is finished. The next EEPROM location will contain a new register header or version mask.

Table E-15 summarizes the Type 3 register modifications.

Table E-15. **Register Write: Register Modifications (Type 3)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Desc	L	X	X	Op Code			Starting Bit					Number of Bits				
Desc	Address															
Desc	Data															
Desc	Data (least significant word - if indicated)															

The **Starting Bit** indicates a value from 0-31 corresponding to the initial bit position for this parameter. The **Number of Bits** describes the size of the data and indicates how large of a mask should be applied to the previous value. The **Op Code** determines how the new **Data** interacts with the existing parameter that was read. The replaced parameter size must match the size of the **Data** that is given. If the **Number of Bits** is 16 or less, then only one EEPROM location of **Data** is consumed. The **Op Code** contains a few simple operations for modification of dynamic values that may have been set by software or chipset algorithms.

Op Code: The mechanism for apply the **Data** value to the parameter:

- 0: Replace the current parameter with the given value.
- 1: Add the current parameter to the read value of this parameter.
- 2: Subtract the current parameter to the read value of this parameter.
- 3: Multiply the current parameter to the read value of this parameter.
- 4: XOR the current parameter to the read value of this parameter.
- 5: OR the current parameter to the read value of this parameter.
- 6: AND the current parameter to the read value of this parameter.

The **L**ast bit controls when this register block is finished. The next EEPROM location will contain a new register header or version mask.

Index

F

L

M

O

P

R

T

V

W