AN-1340
APPLICATION NOTE

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106, U.S.A. • Tel: 781.329.4700 • Fax: 781.461.3113 • www.analog.com

ADF7023-J IEEE 802.15.4g/UBUSAir/FAN Firmware Module

by Liam O'Hora

INTRODUCTION

This application note describes the IEEE 802.15.4g (15d4g) firmware download module for the ADF7023-J transceiver IC. The firmware module adds the following features to the ADF7023-J:

- IEEE 802.15.4g physical layer (PHY) header formatting
- IEEE 802.15.4g compliant data whitening
- Transmitter (Tx)/receiver (Rx) rolling data buffer
- 4-byte to 1000-byte preamble
- ARIB STD T108 compliant clear channel assessment (CCA)

Figure 1 shows a block diagram of the ADF7023-J low power, 902 MHz to 958 MHz transceiver IC. The transceiver contains a custom microcontroller (MCU) core with a mask ROM, which implements packet handling functions and translates radio commands into internal control sequences. An additional 2 kB of program RAM (PRAM) is available and serves as program code memory. It enables the addition of radio controller commands to provide modified or extended functionality. The IEEE 802.15.4g firmware download module described in this application note is based on program code downloaded into this PRAM.

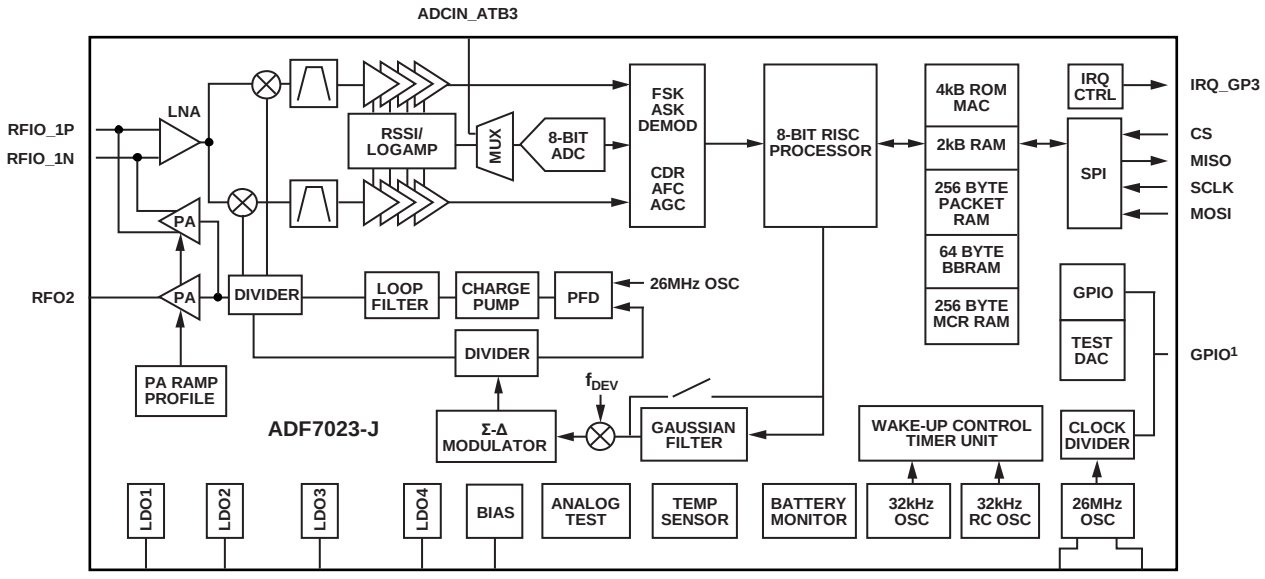
The PRAM block is volatile memory and must be reloaded each time the transceiver wakes up from sleep state. The PRAM locations can be accessed sequentially through the serial peripheral interface (SPI). The details of the code download mechanism are explained in the Code Download Sequence section.

The firmware download binary is **rom_ram_7023_2_2_15d4g_R0p0.dat** and is available from Analog Devices, Inc., at ftp://ftp.analog.com/pub/RFL/FirmwareModules/ADF7023.

The revision (R0p0) may change as new features are added.

The IEEE 802.15.4g firmware download module remaps several of the ADF7023-J registers and commands and extends the data sheet state diagram. See the Register Remapping section for more information.

Note that throughout this application note, multifunction pins (such as the ADCIN_ATB3 pin) are referred to either by the entire pin name or by a single function of the pin (for example, ATB3) when only that function is relevant.



[1]GPIO REFERS TO PIN 17, PIN 18, PIN 19, PIN 20, PIN 25, AND PIN 27.

TABLE OF CONTENTS

REVISION HISTORY

4/15—Revision 0: Initial Version

REGISTER REMAPPING

The firmware extends the ADF7023-J data sheet state diagram to include the 15d4g state, as outlined in Figure 2, and extends the firmware states, as defined in Table 3.

When the firmware module is downloaded and the 15d4g state is entered, some of the battery backup random access memory (BBRAM) registers take on new meaning, and some are rendered redundant. The remapped BBRAM registers are defined in Table 1. The command list is expanded, as defined in Table 2. The INTERRUPT_MASK_0 register and the INTERRUPT_SOURCE_0 register are also remapped while in the 15d4g state, as defined in Table 31 and Table 32.

The 0x20 to 0xFF packet RAM locations are available for packet data in the 15d4g state. Byte 0x00 to Byte 0x1F are allocated for use by the on-chip processor and must not be used for packet data.

When 15d4g state is exited, all normal data sheet functionality is restored. Note, however, that the reused settings must be reprogrammed appropriately according to the data sheet.

Table 1. BBRAM Reuse During 15d4g State

Address (Hex)	Normal Operation	Operation in 15d4g State	Comment
0x100	INTERRUPT_MASK_0	INTERRUPT_MASK_0	Interrupts remapped in 15d4g state
0x101	INTERRUPT_MASK_1	INTERRUPT_MASK_1	
0x102	NUMBER_OF_WAKEUPS_0	NUMBER_OF_WAKEUPS_0	
0x103	NUMBER_OF_WAKEUPS_1	NUMBER_OF_WAKEUPS_1	
0x104	NUMBER_OF_WAKEUPS_IRQ_THRESHOLD_0	NUMBER_OF_WAKEUPS_IRQ_THRESHOLD_0	
0x105	NUMBER_OF_WAKEUPS_IRQ_THRESHOLD_1	NUMBER_OF_WAKEUPS_IRQ_THRESHOLD_1	
0x106	RX_DWELL_TIME	RX_DWELL_TIME	
0x107	PARMTIME_DIVIDER	PARMTIME_DIVIDER	
0x108	SWM_RSSI_THRESH	SWM_RSSI_THRESH	
0x109	CHANNEL_FREQ_0	CHANNEL_FREQ_0	
0x10A	CHANNEL_FREQ_1	CHANNEL_FREQ_1	
0x10B	CHANNEL_FREQ_2	CHANNEL_FREQ_2	
0x10C	RADIO_CFG_0	RADIO_CFG_0	
0x10D	RADIO_CFG_1	RADIO_CFG_1	
0x10E	RADIO_CFG_2	RADIO_CFG_2	
0x10F	RADIO_CFG_3	RADIO_CFG_3	
0x110	RADIO_CFG_4	RADIO_CFG_4	
0x111	RADIO_CFG_5	RADIO_CFG_5	
0x112	RADIO_CFG_6	RADIO_CFG_6	
0x113	RADIO_CFG_7	RADIO_CFG_7	
0x114	RADIO_CFG_8	RADIO_CFG_8	
0x115	RADIO_CFG_9	RADIO_CFG_9	
0x116	RADIO_CFG_10	RADIO_CFG_10	
0x117	RADIO_CFG_11	RADIO_CFG_11	
0x118	IMAGE_REJECT_CAL_PHASE	IMAGE_REJECT_CAL_PHASE	
0x119	IMAGE_REJECT_CAL_AMPLITUDE	IMAGE_REJECT_CAL_AMPLITUDE	
0x11A	MODE_CONTROL	MODE_CONTROL	
0x11B	PREAMBLE_MATCH	PREAMBLE_MATCH	
0x11C	SYMBOL_MODE	SYMBOL_MODE	
0x11D	PREAMBLE_LEN	PREAMBLE_LEN	
0x11E	CRC_POLY_0	CRC_POLY_0	
0x11F	CRC_POLY_1	CRC_POLY_1	
0x120	SYNC_CONTROL	SYNC_CONTROL	
0x121	SYNC_BYTE_0	SYNC_BYTE_0	
0x122	SYNC_BYTE_1	SYNC_BYTE_1	
0x123	SYNC_BYTE_2	SYNC_BYTE_2	
0x124	TX_BASE_ADR	TX_BASE_ADR	
0x125	RX_BASE_ADR	RX_BASE_ADR	

Address (Hex)	Normal Operation	Operation in 15d4g State	Comment
0x126	PACKET_LENGTH_CONTROL	PACKET_LENGTH_CONTROL	
0x127	PACKET_LENGTH_MAX	PACKET_LENGTH_MAX	
0x128	STATIC_REG_FIX	RESERVED	Set to 0 for 15d4g state
0x129	ADDRESS_MATCH_OFFSET	BB_CCA_CFG_0	
0x12A	ADDRESS_LENGTH	BB_CCA_CFG_1	
0x12B	Address matching	BB_CCA_THRESHOLD	
0x12C	Address matching	BB_RX_RSSI	
0x12D	Address matching	BB_ATB_CONTROL	
0x12E	Address matching	BB_NB_PREAMBLE_BYTES_LOW	
0x12F	Address matching	BB_NB_PREAMBLE_BYTES_HIGH	
0x130	Address matching	BB_SFD_LOW	
0x131	Address matching	BB_SFD_HIGH	
0x132	Address matching	BB_PHR_LOW	Controlled by firmware in 15d4g state
0x133	Address matching	BB_PHR_HIGH	Controlled by firmware in 15d4g state
0x134	Address matching	BB_RX_BUFFER_SIGNAL	
0x135	Address matching	BB_RX_BUFFER_SIZE	
0x136	Address matching	BB_TX_BUFFER_SIGNAL	
0x137	Address matching	BB_TX_BUFFER_SIZE	
0x138	RSSI_WAIT_TIME	Reserved	Set to 0x00 for 15d4g state
0x139	TESTMODES	BB_TESTMODES	
0x13A	TRANSITION_CLOCK_DIV	Reserved	Set to 0x01 for 15d4g state
0x13B	Reserved	BB_VCO_BAND_READBACK	
0x13C	Reserved	BB_VCO_AMP_READBACK	
0x13D	Reserved	Reserved	
0x13E	RX_SYNTH_LOCK_TIME	RX_SYNTH_LOCK_TIME	
0x13F	TX_SYNTH_LOCK_TIME	TX_SYNTH_LOCK_TIME	
0x01F	Packet data	PHY_RX_STATUS	Packet RAM used in 15d4g state

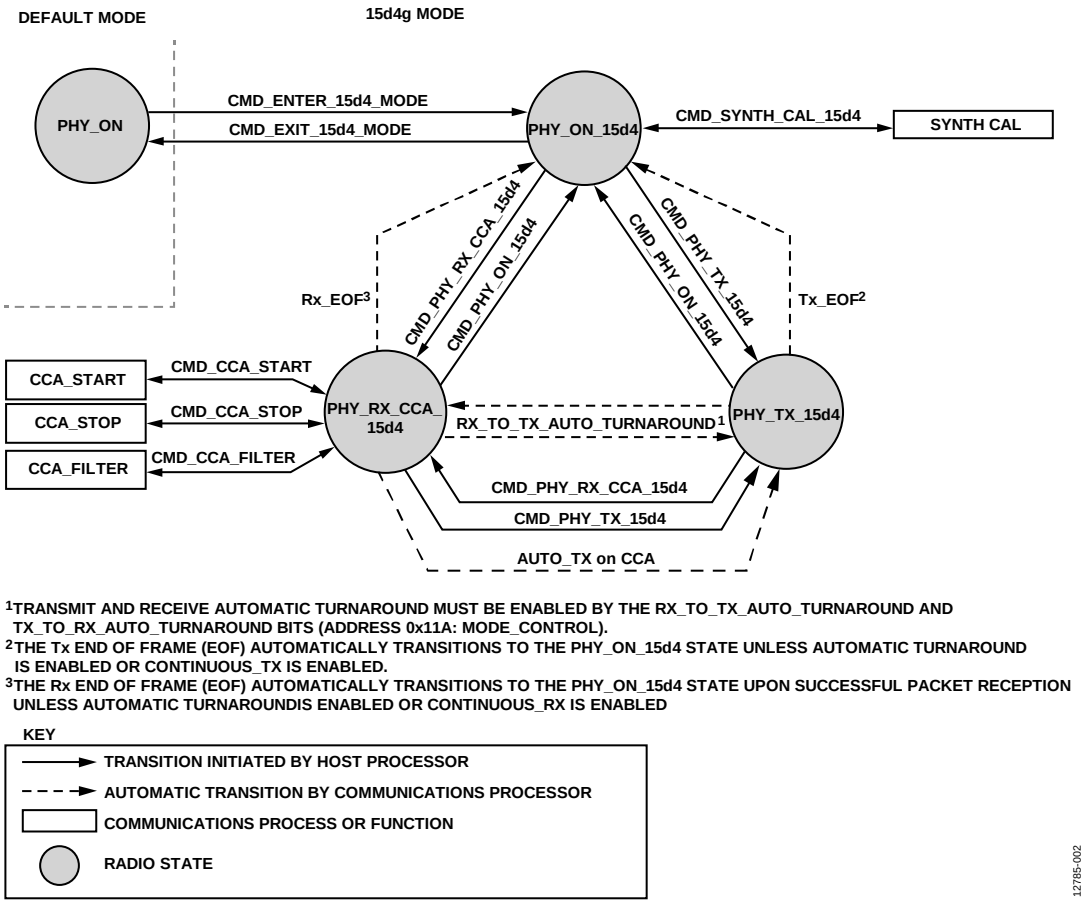


Figure 2. 15d4g State Diagram

Table 2. Commands and States Included in the Firmware Download

Command/Bit	Command Code	Present State	Next State
CMD_ENTER_15d4_MODE	0xC1	PHY_ON	PHY_ON_15d4
CMD_EXIT_15d4_MODE	0xB1	PHY_ON_15d4	PHY_ON
CMD_PHY_RX_CCA_15d4	0xB2	PHY_ON_15d4, PHY_TX_15d4, PHY_RX_CCA_15d4	PHY_RX_CCA_15d4
CMD_PHY_TX_15d4	0xB5	PHY_ON_15d4, PHY_RX_CCA_15d4	PHY_TX_15d4
CMD_PHY_ON_15d4	0xB1	PHY_RX_CCA_15d4, PHY_TX_15d4	PHY_ON_15d4
CMD_SYNTH_CAL_15d4	0xEE	PHY_ON_15d4	PHY_ON_15d4
CMD_CCA_START	0xB7	PHY_RX_CCA_15d4	PHY_RX_CCA_15d4
CMD_CCA_STOP	0xB8	PHY_RX_CCA_15d4	PHY_RX_CCA_15d4
CMD_CONFIG_DEV	0xBB	PHY_ON_15d4	PHY_ON_15d4

Table 3. FW_STATE Description

Value	State	Description
0x18	PHY_ON_15d4	The device is ready to operate in 15d4g mode, and PHY_RX_CCA_15d4 and PHY_TX_15d4 states can be entered.
0x1A	PHY_RX_CCA_15d4	The device is in 15d4g receive mode where valid 15d4g packets can be received and CCA can be enabled. After reception of a valid packet, the device can return to the PHY_ON_15d4 state, can remain in the PHY_RX_CCA_15d4 state, or can transition to the PHY_TX_15d4 state depending on the user settings.
0x1C	PHY_TX_15d4	The device is in 15d4g transmit mode, where it automatically transmits the transmit packet stored in packet RAM. After transmission of the packet, the device can return to the PHY_ON_15d4 state, can remain in the PHY_TX_15d4 state, or can transition to the PHY_RX_CCA_15d4 state depending on the user settings.

REGISTER MAP EXTENSION

The following sections detail the registers for use with the 15d4g firmware download.

CCA CONTROL

Table 4.

Address (Hex)	Register	Description

REGISTER DESCRIPTIONS

Table 10. Address 0x129: BB_CCA_CFG_0

Bit(s)	Name	Access	Description
7	CCA_STATUS	R/W	Indicates if channel is clear (0) or busy (1).
[6:5]	CCA_FILTER_BW	R/W	CCA_FILTER_BW[1:0] sets the IF filter bandwidth for CCA RSSI. 00: 100 kHz 01: 150 kHz 10: 200 kHz 11: 300 kHz
4	CCA_AUTO_TX	R/W	1: remain in PHY_RX_CCA_15d4 state on clear channel assessment (manual Tx). 0: automatic Tx on clear channel assessment.
[3:1]	CCA_TIMER	R/W	CCA_TIMER[2:0] selects one of eight timer options. 000: 160 µs 001: 320 µs 010: 640 µs 011: 1280 µs 100: 1920 µs 101: 2560 µs 110: 5120 µs 111: 9960 µs
0	CCA_BYPASS_UPDATE	R/W	Debug mode only. For normal use, set to 0. 0: CCA data is updated every byte (normal use). 1: CCA data is not updated every byte (debug mode only, causes CCA_STATUS to be frozen).

Table 11. Address 0x12A: BB_CCA_CFG_1

Bit(s)	Name	Access	Description
7	CCA_LIVE_STATUS	R/W	Live indication if channel is clear (0) or busy (1).
6	ENABLE_IFBW_SWITCH_ON_PREAMBLE	R/W	1: changes bandwidth when preamble is detected. 0: keep bandwidth when preamble is detected.
5	ENABLE_IFBW_AUTO_SWITCH	R/W	1: changes bandwidth when CMD_CCA_START is issued. 0: keep default bandwidth when CMD_CCA_START is issued.
4	WAIT_FOR_HOST_EOCCA	R/W	Debug mode only. For normal use, set to 0. 1: wait for host at end of CCA timer after CCA trigger (debug mode only). 0: continue to update CCA registers on CCA timer timeout (normal use).
[3:0]	Reserved	R/W	Set to 0.

Table 12. Address 0x12B: BB_CCA_THRESHOLD

Bit(s)	Name	Access	Description
[7:0]	CCA_THRESHOLD	R/W	This is an 8-bit number representing the clear channel assessment RSSI threshold in dBm. CCA_THRESHOLD = RSSI (dBm) + 107

Table 13. Address 0x12C: BB_RX_RSSI

Bit(s)	Name	Access	Description
[7:0]	RX_RSSI_READBACK	R/W	This register contains the RSSI value measured on the receiver. RSSI (dBm) = RX_RSSI_READBACK − 107

Table 14. Address 0x12D: BB_ATB_CONTROL

Bit(s)	Name	Access	Description
[7:6]	RX_ATB_CONTROL	R/W	Maps the status of ATB pins during Rx Bit 7 = ATB1/ATB3: 0 = low, 1 = high Bit 6 = ATB2/ATB4: 0 = low, 1 = high
5	RX_ATB_LEVEL	R/W	1: select to use 1.8 V drivers (ATB3 and ATB4) during Rx 0: select to use V_{DD} drivers (ATB1 and ATB2) during Rx
4	Reserved	R/W	Set to 0
[3:2]	TX_ATB_CONTROL	R/W	Maps the status of ATB pins during Tx Bit 3 = ATB1/ATB3: 0 = low, 1 = high Bit 2 = ATB2/ATB4: 0 = low, 1 = high
1	TX_ATB_LEVEL	R/W	1: select to use 1.8 V drivers (ATB3 and ATB4) during Tx 0: select to use V_{DD} drivers (ATB1 and ATB2) during Tx
0	Reserved	R/W	Set to 0

Table 15. Address 0x12E: BB_NB_PREAMBLE_BYTES_LOW

Bit(s)	Name	Access	Description
[7:0]	NB_PREAMBLE_BYTES_LOW	R/W	Bits[7:0] of the NB_PREAMBLE_BYTES word

Table 16. Address 0x12F: BB_NB_PREAMBLE_BYTES_HIGH

Bit(s)	Name	Access	Description
[7:0]	NB_PREAMBLE_BYTES_HIGH	R/W	Bits[15:8] of the NB_PREAMBLE_BYTES word. NB_PREAMBLE_BYTES defines the number of preamble bytes transmitted. Note that valid values are within the range of 4 ≤ NB_PREAMBLE_BYTES word ≤ 1000.

Table 17. Address 0x130: BB_SFD_LOW

Bit(s)	Name	Access	Description
[7:0]	BB_SFD_LOW	R/W	Bits[7:0] of the SFD word, see Table 36

Table 18. Address 0x131: BB_SFD_HIGH

Bit(s)	Name	Access	Description
[7:0]	BB_SFD_HIGH	R/W	Bits[15:8] of the SFD word, see Table 36

The values in Address 0x132 (the BB_PHR_LOW register) are read only. In receive mode, the values are copied in by the 15d4g firmware download module from a received packet. In transmit mode, the required PHR value for a transmit must be written to the TX_BASE_ADR register, and the firmware reflects those values back in Address 0x132.

Table 19. Address 0x132: BB_PHR_LOW

Bit(s)	Name	Access	Description
7	MODE_SWITCH	R	Indicates if the packet is a mode switch packet.
[6:5]	Reserved	R	Set to 0.
4	FCS	R	0: 32-bit frame check sequence (FCS). 1: 16-bit FCS. Note that only 16-bit FCS is automatically handled by the ADF7023-J. 32-bit FCS must be calculated by the user. For 32-bit FCS Tx, the user must calculate the FCS and add to the end of the payload. For 32-bit FCS Rx, the user must calculate the CRC and compare with the last 4 bytes received into the packet RAM. In 32-bit FCS Rx, a CRC correct interrupt is not generated, and RxEOF signifies the end of packet reception.
3	WHITENING	R	1: PSDU whitened.
[2:0]	PSDU_LENGTH_HI	R	Bits[10:8] of the PSDU_LENGTH[10:0] word.

The values in Address 0x133 (the BB_PHR_HIGH register) are read only. In receive mode, the values are copied in by the 15d4g firmware download module from a received packet. In transmit mode, the required PHR value for a transmit must be written to the TX_BASE_ADR register, and the firmware reflects those values back in Address 0x133.

Table 20. Address 0x133: BB_PHR_HIGH

Bit(s)	Name	Access	Description
[7:0]	PSDU_LENGTH_LO	R/W	Bits[7:0] of the PSDU_LENGTH[10:0] word. Note that valid values are within the range of 2 < PSDU_LENGTH < 2048.

Table 21. Address 0x134: BB_RX_BUFFER_SIGNAL

Bit(s)	Name	Access	Description
[7:0]	RX_BUFFER_SIGNAL	R/W	The rolling buffer fills from the RX_BASE_ADR with data. When the buffer reaches location RX_BASE_ADR + RX_BUFFER_SIGNAL, the INTERRUPT_BUFFER_ALMOST_FULL interrupt is generated. Note that RX_BUFFER_SIGNAL must have a value greater than 2.

Table 22. Address 0x135: BB_RX_BUFFER_SIZE

Bit(s)	Name	Access	Description
[7:0]	RX_BUFFER_SIZE	R/W	This is the maximum size of the Rx buffer. When the location RX_BASE_ADR + RX_BUFFER_SIZE is filled with data, an INTERRUPT_BUFFER_FULL interrupt is generated, and the buffer loops back to location RX_BASE_ADR and continues to fill from there. Note that RX_BUFFER_SIZE must have a value greater than 2 and that, for a 16-bit FCS case, a 2-byte overflow area is required after the RX_BUFFER_SIZE register.

Table 23. Address 0x136: BB_TX_BUFFER_SIGNAL

Bit(s)	Name	Access	Description
[7:0]	TX_BUFFER_SIGNAL	R/W	The rolling buffer sends from the TX_BASE_ADR register with data. When the buffer reaches and sends from location TX_BASE_ADR + TX_BUFFER_SIGNAL, an INTERRUPT_BUFFER_ALMOST_FULL interrupt is generated. Note that TX_BUFFER_SIGNAL must have a value greater than 2.

Table 24. Address 0x137: BB_TX_BUFFER_SIZE

Bit(s)	Name	Access	Description
[7:0]	TX_BUFFER_SIZE	R/W	This is the maximum size of the Tx buffer. When the data from location TX_BASE_ADR + TX_BUFFER_SIZE is sent, an INTERRUPT_BUFFER_FULL interrupt is generated, and the buffer loops back to location TX_BASE_ADR and continues to send from there. Note that TX_BUFFER_SIZE must have a value greater than 2.

Table 25. Address 0x138: RSSI_WAIT_TIME

Bit(s)	Name	Access	Description
[7:0]	RSSI_WAIT_TIME	R/W	Set to 0 before entering PHY_ON_15d4 from PHY_ON. Set back to 0xA7 on returning to PHY_ON from PHY_ON_15d4, before entering smart wake mode.

Table 26. Address 0x139: BB_TESTMODES

Bit(s)	Name	Access	Description
7	EXT_PA_LNA_ATB_CONFIG	R/W	Set to 0. Do not enable external PA/LNA in Register 0x11A, because this conflicts with the ATB control in 15d4g mode. If an external PA/LNA is to be used in 15d4g mode, control using the logic in BB_ATB_CONTROL.
[6:2]	Reserved	R/W	Set to 0.
1	BIT_CONTINUOUS_TX	R/W	1: device remains in PHY_TX_15d4 after packet transmission. 0: device returns to PHY_ON_15d4 after packet transmission.
0	BIT_CONTINUOUS_RX	R/W	1: device remains in PHY_RX_CCA_15d4 after packet reception. 0: device returns to PHY_ON_15d4 after packet reception.

Table 27. Address 0x13A: TRANSITION_CLK_DIV

Bit(s)	Name	Access	Description
[7:3]	Reserved	R/W	Set to 0.
[2:0]	FAST_TRANSITION	R/W	Set to 1, fast transition times enabled.

Table 28. Address 0x13B: BB_VCO_BAND_READBACK

Bit(s)	Name	Access	Description
[7:0]	VCO_BAND_READBACK	R/W	Stores the VCO band calibration results after synthesizer calibration, for fast Tx/Rx transitions.

Table 29. Address 0x13C: BB_VCO_AMPL_READBACK

Bit(s)	Name	Access	Description
[7:0]	VCO_AMPL_READBACK	R/W	Stores the VCO amplitude calibration results after synthesizer calibration, for fast Tx/Rx transitions.

Table 30. Address 0x01F: PHY_RX_STATUS

Bit(s)	Name	Access	Description
[7:3]	Reserved	R	Reserved.
2	CRC_RX_STATUS	R	1: CRC correct. 0: CRC incorrect or not yet calculated.
1	SFD_RX_STATUS	R	1: SFD detected. 0: SFD not detected.
0	PREAMBLE_RX_STATUS	R	1: preamble detected. 0: preamble not detected.

INTERRUPT MASK AND SOURCE CONFIGURATIONS

Table 31. Address 0x100: INTERRUPT_MASK_0

Bit(s)	Name	Access	Description
7	INTERRUPT_CCA	R/W	Interrupt after CCA_TIMER period has expired. (CCA_STATUS flag indicates if the channel is busy or clear.) 1: interrupt enabled 0: interrupt disabled
6	INTERRUPT_BUFFER_FULL	R/W	Interrupt when the Rx or Tx buffer is full. 1: interrupt enabled 0: interrupt disabled
5	INTERRUPT_BUFFER_ALMOST_FULL	R/W	Interrupt when the Rx or Tx buffer is almost full. 1: interrupt enabled 0: interrupt disabled
4	INTERRUPT_RX_EOF	R/W	Interrupt when a packet has finished receiving. 1: interrupt enabled 0: interrupt disabled
3	INTERRUPT_TX_EOF	R/W	Interrupt when a packet has finished transmitting. 1: interrupt enabled 0: interrupt disabled
2	INTERRUPT_CRC_CORRECT	R/W	Interrupt when a received packet has the correct CRC. 1: interrupt enabled 0: interrupt disabled
1	INTERRUPT_PHR_DETECT	R/W	Interrupt when PHR has been detected in the received packet. Interrupt is generated by SFD detected, but delayed and issued at the end of PHR reception. 1: interrupt enabled 0: interrupt disabled
0	INTERRUPT_PREAMBLE_DETECT	R/W	Interrupt when a qualified preamble has been detected in the received packet. 1: interrupt enabled 0: interrupt disabled

Table 32. Address 0x336: INTERRUPT_SOURCE_0

Bit(s)	Name	Access	Reset	Description
7	INTERRUPT_CCA	R/W	0	Asserted after CCA_TIMER period expires.
6	INTERRUPT_BUFFER_FULL	R/W	0	Asserted when the Rx or Tx buffer is full (Rx or Tx buffer size is reached).
5	INTERRUPT_BUFFER_ALMOST_FULL	R/W	0	Asserted when the Rx or Tx buffer is almost full (Rx or Tx buffer signal is reached).
4	INTERRUPT_RX_EOF	R/W	0	Asserted when a packet finishes receiving.
3	INTERRUPT_TX_EOF	R/W	0	Asserted when a packet finishes transmitting.
2	INTERRUPT_CRC_CORRECT	R/W	0	Asserted when a received packet has the correct CRC.
1	INTERRUPT_PHR_DETECT	R/W	0	Asserted when PHR is detected in the received packet.
0	INTERRUPT_PREAMBLE_DETECT	R/W	0	Asserted when a qualified preamble is detected in the received packet.

FAST Tx/Rx TRANSITIONS

The download module supports fast transitions from PHY_ON_15d4 to PHY_TX_15d4 and PHY_RX_CCA_15d4. The fast transitions are achieved by the following sequence:

1. Issue CMD_SYNTH_CAL (0xEE) while in PHY_ON_15d4. A synthesizer calibration is performed at the programmed channel frequency, and the results are stored in BB_VCO_BAND_READBACK (Address 0x13B) and BB_VCO_AMPL_READBACK (Address 0x13C).
2. Enable synthesizer calibration overwrite by setting VCO_OVRW_EN (Address 0x3CD) = 0x3.

The CMD_PHY_RX_CCA_15d4 and CMD_PHY_TX_15d4 commands complete in a shorter time after this sequence, because a synthesizer calibration is no longer performed.

Note that it is good practice to periodically perform a full synthesizer calibration, which maintains optimum RF performance across fluctuations in operating temperature or battery voltage.

CLEAR CHANNEL ASSESSMENT

For clear channel assessment, the CCA threshold is programmed into the BB_CCA_THRESHOLD register (Address 0x12B). If the measured RSSI is above this threshold, the channel is busy; if the measured RSSI is below this threshold, the channel is clear.

Evaluation of RSSI is automatically activated on entering PHY_RX_CCA_15d4. Multiple times per byte, the RSSI value is evaluated, and the CCA_LIVE_STATUS flag is updated to busy or clear. Therefore, after entering the PHY_RX_CCA_15d4 state, the RSSI value and a live CCA status are always updated, unless updates are turned off with CCA_BYPASS_UPDATE (Bit 0 of Address 0x129).

Note that there is a time lag until the first RSSI value is updated on first entering PHY_RX_CCA_15d4 state. Until the first RSSI value is updated, the previously measured RSSI value is retained. If this is undesirable, the BB_RX_RSSI register may be zeroed before entering PHY_RX_CCA_15d4.

On issuing CMD_CCA_START, the RSSI is evaluated and compared with the CCA threshold to determine if the channel is busy or clear, and the CCA_STATUS flag is updated accordingly.

CCA TIMERS

The CCA evaluation period is programmable in CCA_TIMER in the BB_CCA_CONTROL register. This is a timer in microseconds, as defined in Table 33.

The timer is initiated and the evaluation period is started by the CMD_CCA_START command. When the timer times out, or when the timer is interrupted by an incoming packet, the CCA interrupt is generated (if it is enabled). Note that there is an internal overhead on the timer modes that means the CCA evaluation period is somewhat less than the CCA timer evaluation value. Therefore, it is recommended that the user performs their own characterization tests to ensure that the actual CCA evaluation period matches their requirement, and if not, to select the next available timer, or extend the CCA evaluation time as discussed later in this section.

Table 33. CCA Timer Evaluation Times

Timer Option	Evaluation Period Time (µs)
000	160
001	320
010	640
011	1280
100	1920
101	2560
110	5120
111	9960

The CCA_STATUS is automatically cleared at the beginning of every evaluation period (channel free). If the CCA_LIVE_STATUS bit goes high (that is, RX_RSSI_READBACK exceeds CCA_THRESHOLD) at any stage during the evaluation period, the CCA_STATUS flag is automatically updated to busy.

CCA timer mode is generally used with automatic Tx (the BB_CCA_CFG_0 register, Bit 4 = 0).

If the channel is clear, and automatic Tx is enabled, the device transitions automatically to the PHY_TX_15d4 state (see Figure 3).

If the channel is clear and automatic Tx is not enabled, the CCA interrupt is generated, and the device remains in the PHY_RX_CCA_15d4 state, with CCA_STATUS set to free.

If the channel is busy, and the timer times out, a CCA interrupt is generated, and the device remains in the PHY_RX_CCA_15d4 state, with CCA_STATUS set to busy (see Figure 4).

It is important to note that the RSSI value is continuously updated and evaluated during each byte period once in the PHY_RX_CCA_15d4 state, unless updates are turned off with CCA_BYPASS_UPDATE (Bit 0 of Address 0x129). Therefore, if the channel had evaluated as clear, but automatic Tx was not enabled, the CCA continues to be assessed, and if the channel becomes busy, this is detected and CCA_STATUS is set to busy. This can be used as a default timer extension.

During CCA timer mode, incoming packets can be received. In the event of an SFD being detected, any running CCA timer is terminated, and the CCA interrupt is asserted after SFD detection. CCA_STATUS is set to busy. The packet continues to be received normally (see Figure 5).

If the CCA timer times out during the preamble or SFD of an incoming packet before the device has an opportunity to detect the SFD, that packet is lost, and the CCA interrupt is generated (see Figure 6).

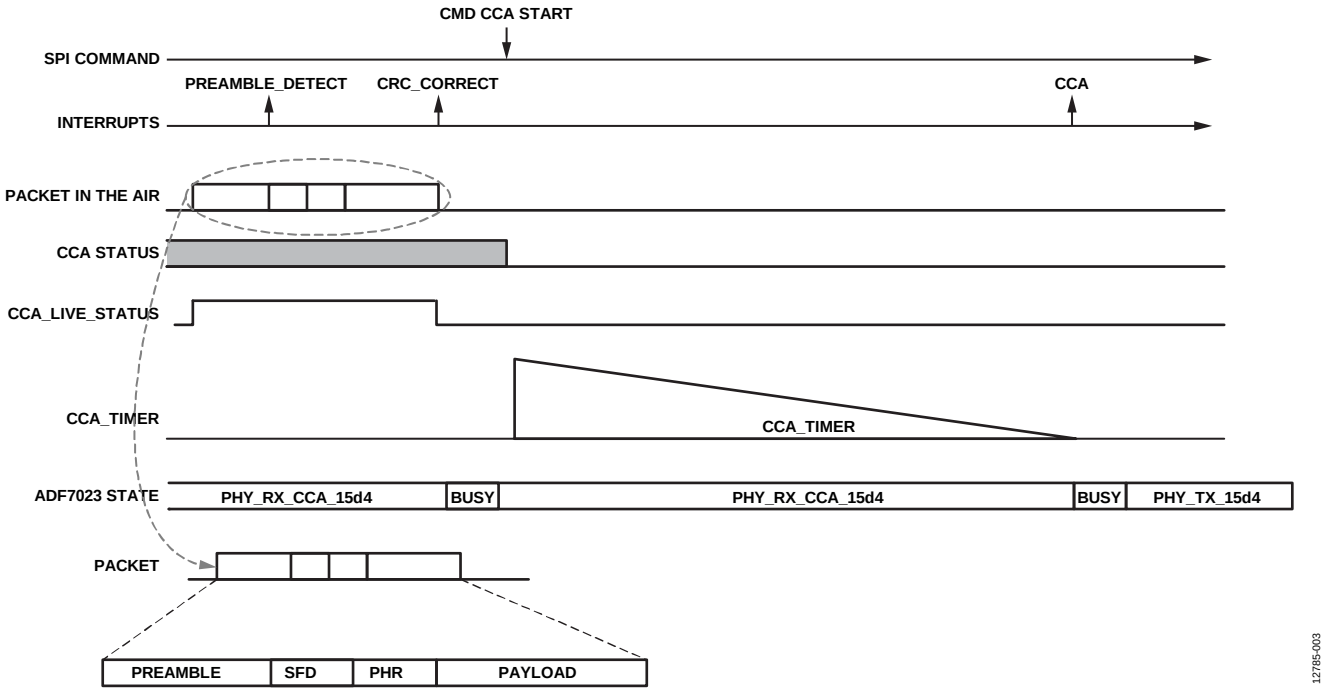


Figure 3. CCA Timer Mode, Channel Clear, Automatic Tx

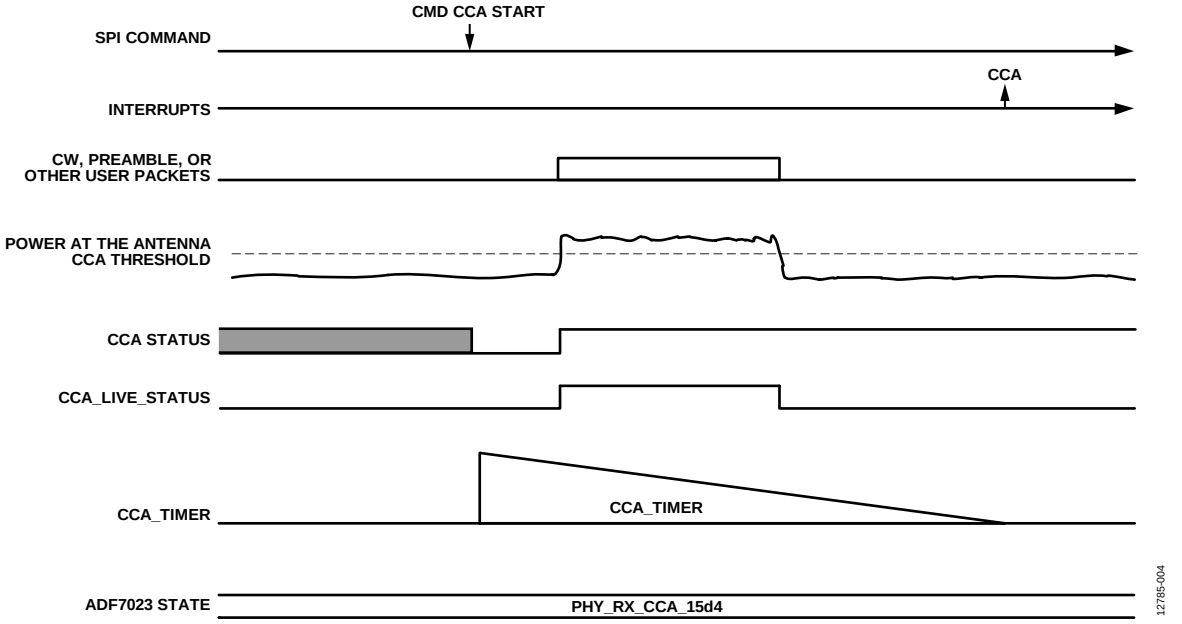


Figure 4. CCA Timer Mode, Channel Busy

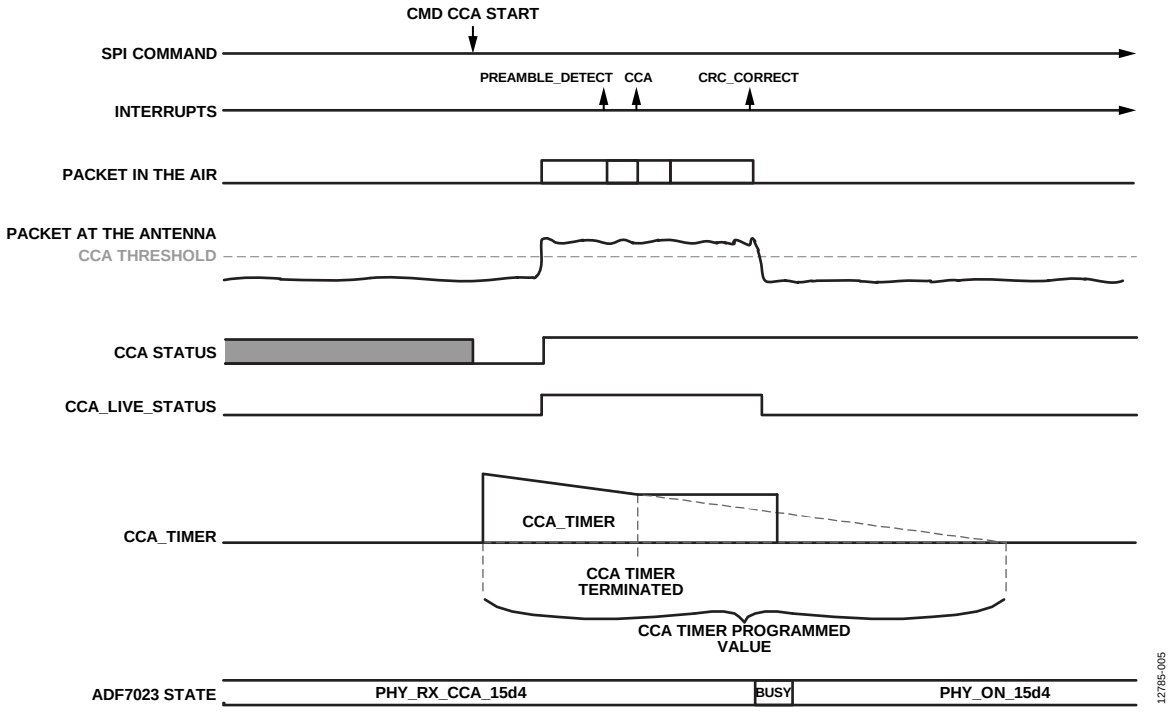


Figure 5. Packet Reception During CCA Timer Mode

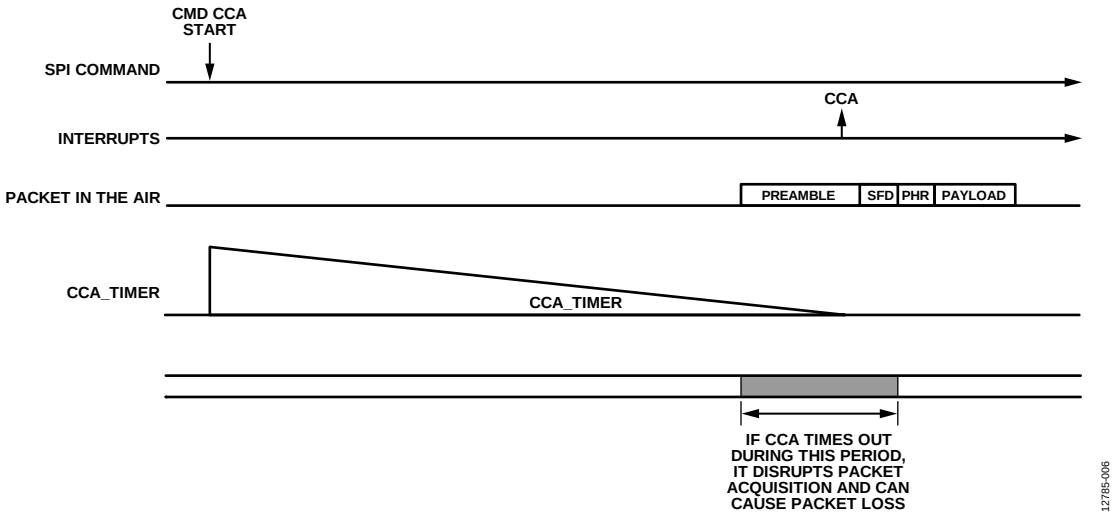


Figure 6. CCA Timer Timeout During Packet Reception

Issuing Commands During Packet Reception

Take care when issuing CMD_CCA_START as there are certain conditions under which issuing the command, coincident with the start of preamble on an incoming packet, cause that packet to be lost. Minimizing this risk is discussed in the Potential Packet Loss when Issuing Commands section.

Dynamic IF Filter Bandwidth Setting During CCA Timer Mode

The functionality of the intermediate frequency (IF) filter bandwidth for Rx during CCA timer mode is discussed in this section.

Upon entering the PHY_RX_CCA_15d4 state, the receiver baseband filter bandwidth is selected from the BBRAM (Bits[7:6] of Address 0x115).

In CCA timer mode, the receiver baseband filter can be automatically changed for CCA RSSI by setting ENABLE_IFBW_AUTO_SWITCH (Bit 5 of CCA_CFG_1, Address 0x12A) to 1. Then, the filter bandwidth automatically switches to the CCA_FILTER_BW setting (Bits[6:5] of Address 0x129) when the CMD_CCA_START command (Address 0xB7) is issued.

Note that the IF of the receiver can toggle between 200 kHz and 300 kHz depending on the selected filter bandwidth.

The IF filter bandwidth of the receiver is switched back to its original BBRAM setting (Bits[7:6] of Address 0x115) when one of the following events occurs:

- If the CCA timer times out
- On issuing CMD_CCA_STOP
- On issuing CMD_PHY_RX_CCA_15d4
- On an incoming packet preamble detection if ENABLE_IFBW_SWITCH_ON_PREAMBLE = 1 (available only if the CCA and normal Rx IFBW are both on the 200 kHz IF frequency; requires 12-byte preamble)

On an incoming packet preamble detection, if ENABLE_IFBW_SWITCH_ON_PREAMBLE = 1, the receiver baseband filter switches back to its original Rx IFBW value for packet reception (see Figure 7). This is only available between filters on the 200 kHz IF (that is, 200 kHz, 150 kHz, and 100 kHz). There is also a minimum preamble requirement of 12 bytes to achieve this dynamic IFBW switching.

On an incoming packet preamble detection, if ENABLE_IFBW_SWITCH_ON_PREAMBLE = 0, the receiver baseband filter bandwidth is retained at its current value for the duration of the packet reception, and switches back at the end of packet reception.

If a 300 kHz IFBW is required for CCA, but the normal receive IFBW is 150 kHz, for example, the dynamic IFBW switching on preamble detect is not possible, because the CCA and RX filters are on different IF frequencies. In this case, ENABLE_IFBW_SWITCH_ON_PREAMBLE must be set to 0, and any packet arriving during the CCA time is received on the 300 kHz IFBW. The 150 kHz Rx IFBW is restored for subsequent receives.

False Preamble Detects During CCA Timer Mode

If a false preamble is detected while in CCA timer mode and ENABLE_IFBW_SWITCH_ON_PREAMBLE = 1, the receiver switches back to its original IFBW value for packet reception, as specified in IFBW (Bits[7:6] of Address 0x115).

If SFD is subsequently not detected, this is a false preamble detect, and the firmware takes the following action:

- If ENABLE_IFBW_AUTO_SWITCH = 1, once SFD is not detected, the ADF7023-J sets the IF filter bandwidth to the CCA_FILTER_BW setting (Bits[6:5] of Address 0x129).
- If ENABLE_IFBW_AUTO_SWITCH = 0, once SFD is not detected, the ADF7023-J sets the IF filter bandwidth to the original IFBW value for packet reception, as specified in IFBW (Bits[7:6] of Address 0x115).

If a false preamble is detected while in CCA timer mode and ENABLE_IFBW_SWITCH_ON_PREAMBLE = 0, the receiver does not switch back during preamble.

If SFD is subsequently not detected, then this is a false preamble detect, and the firmware takes the following action:

- If ENABLE_IFBW_AUTO_SWITCH = 1, once SFD is not detected, the ADF7023-J sets the IF filter bandwidth to the CCA_FILTER_BW setting (Bits[6:5] of Address 0x129).
- If ENABLE_IFBW_AUTO_SWITCH = 0, once SFD is not detected, the ADF7023-J sets the IF filter bandwidth to the original IFBW value for packet reception, as specified in IFBW (Bits[7:6] of Address 0x115).

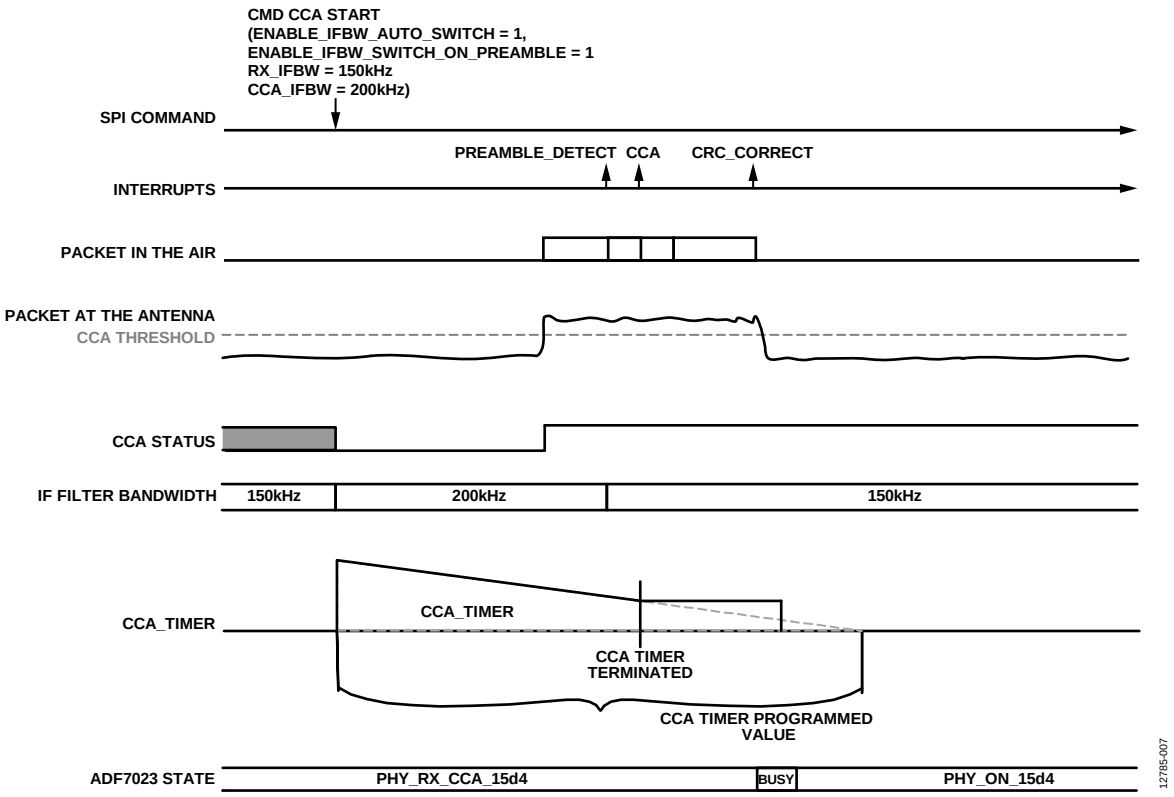


Figure 7. IF Bandwidth Automatic Switching

COMMAND ACCESS DURING PHY_RX_CCA_15d4 STATE

The ADF7023-J is controlled through commands, and command access is described in the ADF7023-J data sheet. The data sheet also describes the use of the status word to ensure that the communications processor is ready to accept a new command.

Potential Packet Loss when Issuing Commands

Take care when issuing commands in the PHY_RX_CCA_15d4 state while waiting for an incoming packet because of the asynchronous timing of the incoming packets.

There are certain conditions under which issuing a command, coincident with the start of preamble on an incoming packet, causes that packet to be lost. Commands involved are the following:

- CMD_PHY_RX_CCA_15d4
- CMD_PHY_TX_15d4
- CMD_PHY_ON_15d4
- CMD_CCA_START

To minimize the risk of losing an incoming packet, a PHY_RX_STATUS (Address 0x01F) byte is offered. Bit 0 of this byte highlights when preamble has been detected; Bit 1 indicates when SFD has been detected; and Bit 2 indicates when CRC is correct.

Upon entering the PHY_RX_CCA_15d4 state, these bits are set to 0. On preamble detection, Bit 0 is set to 1; on SFD detection, Bit 1 is set to 1; on CRC correct, Bit 2 is set to 1.

If a false preamble is detected (that is, a qualified preamble without a subsequent SFD), these bits are set to 0 and the search for preamble continues.

Any command that restarts receive resets these bits to 0.

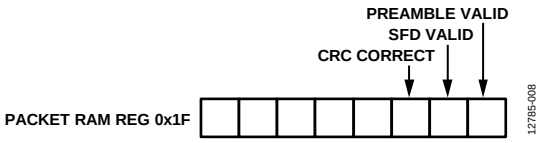


Figure 8. PHY_RX_STATUS Register

Packet loss can be minimized by monitoring the PHY_RX_STATUS register and issuing commands only when PHY_RX_STATUS = 0.

However, while monitoring the PHY_RX_STATUS register method, there is still a condition under which packet loss is possible, where a preamble may have started in the air but is not yet detected by the device.

The two extremes of this condition are outlined in Figure 10 and Figure 11. The worst-case scenario occurs at the latest possible preamble qualification time, which is less than 4 bytes into the 4-byte preamble (see Figure 10). In this scenario, if a command (for example, CMD_CCA_START) occurs during the first 4 bytes of preamble in the air, the preamble acquisition is disrupted and packet loss may occur. The best-case scenario occurs at the earliest possible preamble qualification time, which is 1 byte into the preamble (see Figure 11). In this case, if the CMD_CCA_START occurs during the first byte of preamble, packet loss may occur.

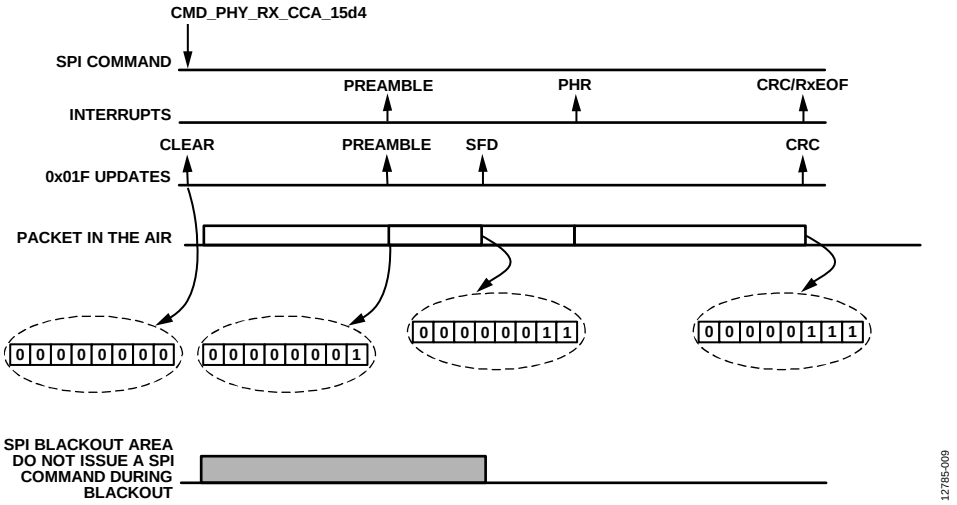


Figure 9. PHY_RX_STATUS Register Update on Packet Reception

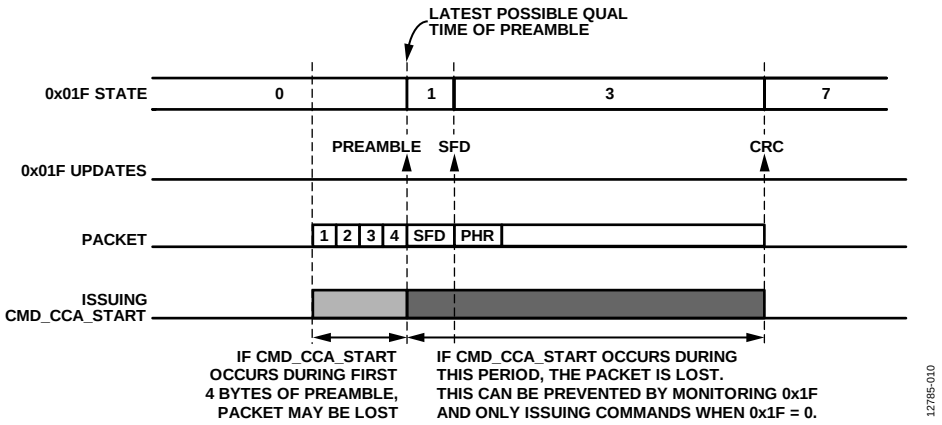


Figure 10. Packet Loss Due to Issuing CMD_CCA_START During Preamble (Worst Case)

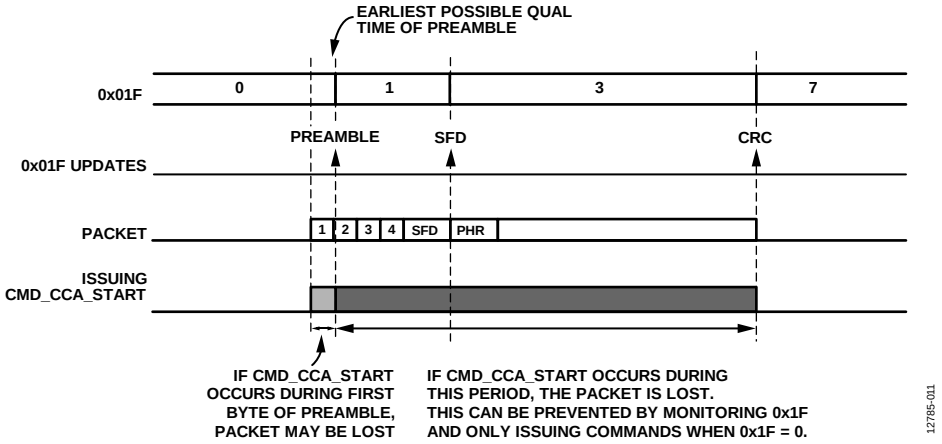


Figure 11. Packet Loss Due to Issuing CMD_CCA_START During Preamble (Best Case)

AUTOMATIC ATB CONTROL

The ATB1, ATB2, ATB3, and ATB4 pins on the ADF7023-J can be used to control external RF antenna switches. In some configurations, the switch control may or may not be the same for Tx and Rx. The firmware download gives complete control of the ATB pins during Tx and Rx to cover for such situations. This section provides configuration details of these four pins in receive and transmit modes.

RECEIVE MODE

In the PHY_RX_CCA_15d4 state, the mapping of the control bits to configure the ATB pins is automatically controlled by Bits[7:5] in BB_ATB_CONTROL (Address 0x12D). Bits[7:6] control the ATB logic, and Bit 5 selects the driver, as shown in Table 34.

These control bits appear at the ATB1 and ATB2 pins as V_{DD} control bits, or at ATB3 and ATB4 as 1.8 V control bits. The selection between V_{DD} or 1.8 V controls is made using RX_ATB_LEVEL (Bit 5 of BB_ATB_CONTROL).

Table 34. ATB Control—Rx Configuration

BB_ATB_CONTROL—0x12D						
RX_ATB_ CONTROL		RX_ATB_ LEVEL	ATB States			
Bit 7	Bit 6	Bit 5	ATB1[1]	ATB2[1]	ATB3[2]	ATB4[2]
0	0	0	L	L	Z	Z
0	1	0	L	H	Z	Z
1	0	0	H	L	Z	Z
1	1	0	H	H	Z	Z
0	0	1	Z	Z	L	L
0	1	1	Z	Z	L	H
1	0	1	Z	Z	H	L
1	1	1	Z	Z	H	H

[1] ATB1 and ATB2 are V_{DD} drivers: H = V_{DD}, L = 0 V, Z = tristate.
[2] ATB3 and ATB4 are 1.8 V drivers: H = 1.8V, L = 0 V, Z = tristate.

TRANSMIT MODE

In the PHY_TX_15d4, the mapping of the control bits to configure the ATB pins is automatically controlled by Bits[3:1] in BB_ATB_CONTROL (Address 0x12D). Bits[3:2] control the ATB logic, and Bit 1 selects the driver, as shown in Table 35.

These control bits appear at the ATB1 and ATB2 pins as V_{DD} control bits, or at ATB3 and ATB4 as 1.8 V control bits. The selection between V_{DD} or 1.8 V controls is made using TX_ATB_LEVEL (Bit 1 of BB_ATB_CONTROL).

Table 35. ATB Control—Tx Configuration

BB_ATB_CONTROL—0x12D						
TX_ATB_ CONTROL		TX_ATB_ LEVEL	ATB States			
Bit 3	Bit 2	Bit 1	ATB1[1]	ATB2[1]	ATB3[2]	ATB4[2]
0	0	0	L	L	Z	Z
0	1	0	L	H	Z	Z
1	0	0	H	L	Z	Z
1	1	0	H	H	Z	Z
0	0	1	Z	Z	L	L
0	1	1	Z	Z	L	H
1	0	1	Z	Z	H	L
1	1	1	Z	Z	H	H

[1] ATB1 and ATB2 are V_{DD} drivers: H = V_{DD}, L = 0 V, Z = tristate.
[2] ATB3 and ATB4 are 1.8 V drivers: H = 1.8 V, L = 0 V, Z = tristate.

Do not enable external PA/LNA in MODE_CONTROL (Register 0x11A) because this conflicts with the ATB control in 15d4g mode. If an external PA/LNA is to be used in 15d4g mode, control using the logic in BB_ATB_CONTROL.

PACKET STRUCTURE IN 15D4G MODE

The ADF7023-J firmware download module supports the MR_FSK PPDU header format without mode switching.

The packet structure is displayed in Figure 12.

In transmit mode, the ADF7023-J firmware module adds the requested number of preamble bytes, and inserts the requested SFD and PHR. If whitening is enabled, the PSDU is whitened. If 16-bit FCS is enabled, the 16-bit CRC is automatically calculated and appended to the transmit payload.

In receive mode, the preamble optionally generates a preamble detect interrupt and is not stored to packet RAM. The SFD is used for byte level synchronization, optionally generates an interrupt synchronous with PHR reception, and is not stored to packet RAM. PHR is received next, and depending on its contents, dictates the receive packet length, whether de-whitening is to be applied or not, and whether FCS is 16 bits or 32 bits. The 16-bit FCS case is handled automatically.

PREAMBLE	SFD	PHR					PSDU		
		MODE SWITCH	RESERVED	FCS	WHITEN-ING	LENGTH	MAC HEADER	MAC PAYLOAD	FCS
4 BYTES TO 1000 BYTES OF PREAMBLE	1001 0000 0100 1110	0	00	1	1	00000010000	WHITENED	WHITENED	WHITENED AND 16-BIT

12785-012

Figure 12. PPDU Format Example

PREAMBLE

The preamble is a 1010 sequence added to the start of the packet during transmission, and removed after receiving the packet. The preamble length is defined in bytes, with a programmable range of 4 bytes to 1000 bytes. It is configured by writing to the BB_NB_PREAMBLE_BYTES_HIGH register (Address 0x12F) and the BB_NB_PREAMBLE_BYTES_LOW register (Address 0x12E).

SFD

The SFD is a 2-byte word that is controlled by writing to BB_SFD_HIGH (Address 0x131) and BB_SFD_LOW (Address 0x130). The SFD takes on one of four valid options selected from Table 36. Note that the SFD is transmitted starting from Bit 0.

PHR

The PHR is a 2-byte word as defined in the PHR section in Figure 12. This word is controlled by writing the low byte to TX_BASE_ADR , and the high byte to TX_BASE_ADR + 1, as these form the first 2 bytes of the transmit packet. On issuing CMD_PHY_TX_15d4, the contents of these 2 bytes are copied into BB_PHR_LOW (Address 0x132) and BB_PHR_HIGH (Address 0x133), respectively. Likewise, on successful packet reception, the packet header bytes are copied into BB_PHR_LOW and BB_PHR_HIGH.

The following sections detail the PHR bit definitions: mode switch, reserved, FCS, whitening, and length.

Mode Switch

Bit 7 of the BB_PHR_LOW register is the MODE_SWITCH bit. The MODE_SWITCH bit is set to 0, indicating that there is no data rate or modulation scheme change during the packet transmission.

If the ADF7023-J receives a packet with mode switch, the MODE_SWITCH bit set to 1, it does not generate a PHR interrupt; however, it does generate an INTERRUPT_CRC_CORRECT interrupt and INTERRUPT_RX_EOF interrupt (if they are enabled).

If the ADF7023-J transmits a packet with MODE_SWITCH set to 1, the transmission is terminated after PHR and the PSDU is not transmitted.

Reserved

Bits[6:5] of the BB_PHR_LOW register are reserved. They are set to 0 for ADF7023-J firmware for transmit mode, and are ignored by the ADF7023-J firmware module in receive mode.

FCS

Bit 4 of the BB_PHR_LOW register is the FCS bit.

FCS = 0 indicates a 32-bit FCS. This case is not handled automatically by the ADF7023-J firmware download module. For transmit, the user must calculate the 32-bit CRC and add it to the PSDU. In receive mode, the 32-bit CRC is received as part of the PSDU and stored in the packet RAM, CRC is not validated, and the CRC interrupt is not asserted. The INTERRUPT_RX_EOF interrupt indicates when the packet is received.

FCS = 1 indicates a 16-bit CRC. This case is handled automatically by the firmware. In transmit mode, the 16-bit CRC is calculated and appended to the transmit PSDU. In receive mode, the 16-bit CRC is calculated from the incoming payload and, if the CRC is correct, the CRC interrupt is asserted (if enabled).

Whitening

Bit 3 of the BB_PHR_LOW register is the whitening bit. If the whitening bit = 0, the PSDU is not whitened; if the whitening bit = 1, the PSDU is whitened on transmission. Only the PSDU is whitened, not the SFD or PHR.

In receive mode, if the whitening bit is set to 1 in the received packet header, the ADF7023-J firmware download module automatically de-whitens the received PSDU.

Length

Bits[2:0] of the BB_PHR_LOW register define Bits[10:8] of the PSDU_LENGTH[10:0] word, and Bits[7:0] of the BB_PHR_HIGH register define Bits[7:0] of the PSDU_LENGTH[10:0] word.

The PSDU_LENGTH[10:0] word can be set to values in the range of 3 to 2047.

Table 36. SFD Values

phyMRFSKSFD	SFD Value for Coded PHR + PSDU (b_0 to b_{15})	SFD Value for Uncoded PHR + PSDU (b_0 to b_{15})
0	0110 1111 0100 1110	1001 0000 0100 1110
1	0110 0011 0010 1101	0111 1010 0000 1110

Tx/Rx ROLLING DATA BUFFER

In packet mode, when the ADF7023-J receives a packet, the data is stored in a linear sequence in the packet RAM. Prior to transmission, the data to be transmitted is written to the packet RAM in a linear sequence. This functionality is described in the ADF7023-J data sheet.

If the IEEE 802.15.4g firmware download module is used, it is possible to receive or transmit packets that are longer than the available packet RAM. This is accomplished via a rolling buffer. The registers applicable to the operation of the rolling buffer are given in Table 37.

The 0x20 to 0xFF packet RAM locations are available for packet data. Byte 0x00 to Bye 0x1F are allocated for use by the on-chip processor and must not be used for packet data.

ROLLING BUFFER IN TRANSMIT MODE

In transmit mode, the start of the rolling buffer in packet RAM is set by TX_BASE_ADR. The size of the buffer is set by BB_TX_BUFFER_SIZE. Take care that the buffer size does not exceed the available packet RAM. When the value in BB_TX_BUFFER_SIZE is added to the address in TX_BASE_ADR, 0x0FF must not be exceeded.

The user must set the value in BB_TX_BUFFER_SIGNAL so that an INTERRUPT_BUFFER_ALMOST_FULL interrupt is generated prior to transmission of all the data in the buffer. Typically, BB_TX_BUFFER_SIGNAL is set so that the interrupt is received when half the data in the buffer has been transmitted.

When an INTERRUPT_BUFFER_ALMOST_FULL is asserted, the host microprocessor should write new data to the locations from TX_BASE_ADR to TX_BASE_ADR + BB_TX_BUFFER_SIGNAL.

When the data in the last byte of the Tx buffer has been transmitted, the ADF7023-J continues transmitting starting with the data at the TX_BASE_ADR. If enabled, an INTERRUPT_BUFFER_FULL interrupt is asserted when the last byte in the buffer is transmitted.

When an INTERRUPT_BUFFER_FULL is asserted, the host microprocessor should write new data to the locations from TX_BASE_ADR + BB_TX_BUFFER_SIGNAL + 1 to TX_BASE_ADR + BB_TX_BUFFER_SIZE.

Transmission of data continues until Tx_EOF or the user issues CMD_PHY_ON_15d4.

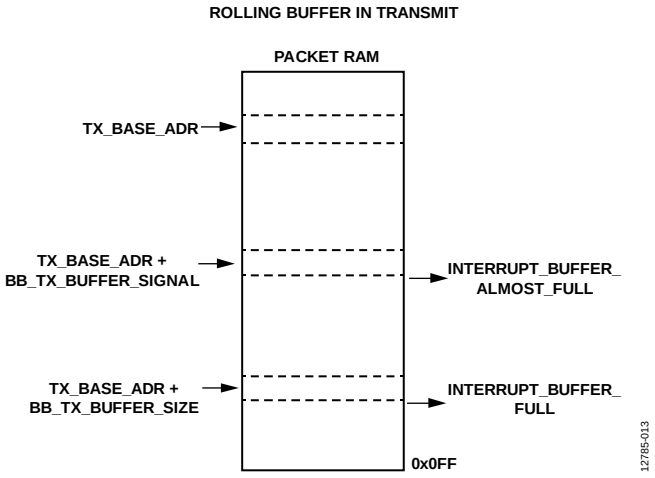


Figure 13. Rolling Buffer in Transmit Mode

Table 37. Registers for Use with Tx/Rx Rolling Data Buffer

Address (Hex)	Register	Description
0x124	TX_BASE_ADR	Start location in packet RAM of Tx buffer
0x125	RX_BASE_ADR	Start location in packet RAM of Rx buffer
0x134	BB_RX_BUFFER_SIGNAL	Rx buffer nearly full
0x135	BB_RX_BUFFER_SIZE	Rx buffer full
0x136	BB_TX_BUFFER_SIGNAL	Tx buffer nearly full
0x137	BB_TX_BUFFER_SIZE	Tx buffer full

ROLLING BUFFER IN RECEIVE MODE

In receive mode, the start of the rolling buffer in packet RAM is set by RX_BASE_ADR. The size of the buffer is set by BB_RX_BUFFER_SIZE. Take care that the buffer size does not exceed the available packet RAM. When the value in BB_RX_BUFFER_SIZE is added to the address in RX_BASE_ADR, 0x0FF must not be exceeded. If 16-bit FCS checking on the ADF7023-J is enabled, 0x0FD must not be exceeded.

The user must set the value in BB_RX_BUFFER_SIGNAL so that an INTERRUPT_BUFFER_ALMOST_FULL interrupt is asserted prior to the buffer being filled by received data. Typically BB_RX_BUFFER_SIGNAL is set so that the interrupt is asserted when half the buffer has been filled with received data.

When an INTERRUPT_BUFFER_ALMOST_FULL is asserted, the host microprocessor should read in the data in the locations from RX_BASE_ADR to RX_BASE_ADR + BB_RX_BUFFER_SIGNAL.

When the last byte of the Rx buffer has been filled, the ADF7023-J continues writing the received data starting at the RX_BASE_ADR. If enabled, an INTERRUPT_BUFFER_FULL interrupt is asserted when the buffer has been filled.

When an INTERRUPT_BUFFER_FULL is asserted, the host microprocessor should read in the data in the locations from RX_BASE_ADR + BB_RX_BUFFER_SIGNAL + 1 to RX_BASE_ADR + BB_RX_BUFFER_SIZE.

Reception of data continues until RX_EOF, a CRC correct interrupt, or the host microprocessor issues CMD_PHY_ON_15d4.

Note that if 16-bit CRC is enabled (16-bit FCS), the FCS bytes are placed into the packet RAM as 2 bytes after the payload, outside of the rolling buffer control. Therefore, if the payload fills right up to RX_BASE_ADR + BB_RX_BUFFER_SIZE, when the 16-bit FCS is calculated and checked versus the incoming FCS, the received FCS bytes are placed in the next two memory locations (that is, beyond the RX_BASE_ADR + BB_RX_BUFFER_SIZE).

If another buffer is defined to start in the memory location immediately following an RX_BASE_ADR + BB_RX_BUFFER_SIZE, the first 2 bytes are overwritten.

To avoid this situation arising for any length and buffer size, an overflow area of 2 bytes must be left vacant immediately after the Rx buffer. If the Rx buffer is assigned the upper part of the packet memory, it must be set so as not to exceed the 0xFD memory location.

If the Rx buffer is placed lower down in packet memory and is immediately followed in packet memory by a Tx buffer, the TX_BASE_ADR must be set to be 3 bytes greater than RX_BASE_ADR + BB_RX_BUFFER_SIZE. This ensures that if the payload length places the packet end at the BB_RX_BUFFER_SIZE boundary, the overflow of the FCS byte(s) does not cause overwriting of the first 2 bytes of the following Tx buffer (the FCS bytes are written into the overflow bytes instead).

This overwriting issue does not arise in 32-bit CRC mode. In 32-bit CRC mode, the ADF7023-J does not calculate the CRC itself; therefore, the incoming 4 bytes of FCS are considered the same as the payload and are loaded into the receive buffer under rolling buffer control.

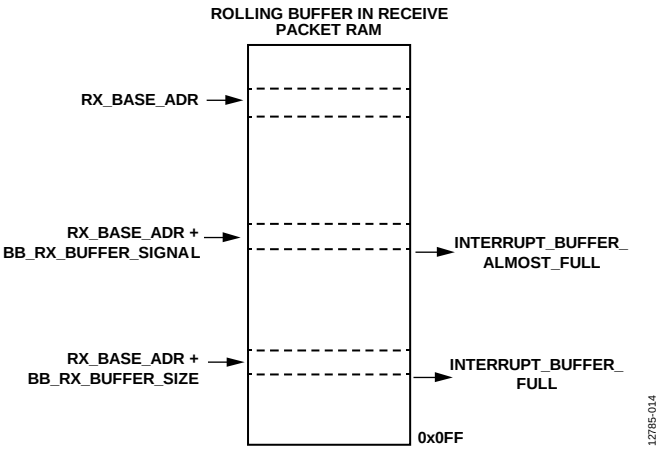


Figure 14. Rolling Buffer in Receive Mode

RECOMMENDED REGISTER SETTINGS

RECOMMENDED AGC SETTINGS

To optimize the receiver for robust packet error rate performance over the full input power range, it is recommended to overwrite the default AGC clock divide setting in the MCR memory. The recommended setting is AGC_CLOCK_DIVIDE (Address 0x32F) = 0x12.

Tx LOOKUP TABLE (LUT) SETTINGS

The optimized PLL bandwidth settings for T108 operation are built into the firmware download. Set Bits[5:4] of the RADIO_CFG_7 register (Address 0x113) to 0 when using the PHY_TX_15d4 state.

REUSED BBRAM SETTINGS

The ADF7023-J 15d4g firmware download reuses BBRAM registers, and therefore, some default functions are no longer available when in 15d4g mode.

Address matching or static register fix are not available with 15d4g download, because the BBRAM Register 0x129 to Register 0x137 are reused by the firmware to control the CCA functionality and 15d4g functionality.

Transmit test modes are not available within PHY_TX_15d4g. To use the transmit test modes, it is necessary to revert to PHY_ON to get to PHY_TX.

To use the optimized PLL settings for T108 operation in PHY_TX, the sequence outlined in the ADF7023-J data sheet must be followed.

Therefore, the full sequence from PHY_ON_15d4 to transmit test mode with optimized T108 PLL settings is as follows:

1. Issue B1 (exit 15d4g mode).
2. Use custom transmit LUT: write 0x2 to Bits[5:4] of Register 0x113 (RADIO_CFG_7).
3. Issue CMD_CONFIG_DEV.
4. Write the custom LUT as defined in Table 38.
5. Select the transmit test mode desired (see Table 42 in the ADF7023-J data sheet).
6. Enter PHY_TX.

Table 38. T108 Custom Transmit Look-Up Table (LUT)

	Data Rates			
Register	50 kbps/ 100 kbps	150 kbps	200 kbps	300 kbps
0x010	0x10	0x18	0x20	0x2E
0x011	0x10	0x18	0x20	0x2E
0x012	0x0F	0x0F	0x0F	0x0B
0x013	0x0F	0x0F	0x0F	0x09
0x014	0x1F	0x1F	0x1F	0x0C
0x015	0x0F	0x0D	0x05	0x03
0x016	0x1F	0x1F	0x1F	0x0C
0x017	0x33	0x22	0x33	0x22
0x018	0x22	0x22	0x22	0x1D

EXAMPLE RECEIVER SETTINGS FOR 100 kbps AND 200 kbps OPERATION, AFC ENABLED

Table 39. Example Rx Settings, AFC Enabled

Parameter	100 kbps, f_{DEV} = 50 kHz	200 kbps, f_{DEV} = 100 kHz
Data Rate	0x3E8	0x7D0
Deviation	0x1F4	0x3E8
IF Filter BW	150 kHz	200 kHz
Discriminator Bandwidth	0x20	0x10
Post Demodulation Bandwidth	0x26	0x4B
AGC Thresholds	Default	Default
AGC Clock Divide	0x12	0x12

SPECIFICATIONS

Table 40. Application to the IEEE 802.15.4g Standard

Parameter	Typical Specification	Test Conditions/Comments
PPDU FORMAT		
Preamble	4 bytes to 1000 bytes[1]	
SFD	2 bytes	
PHR	2 bytes	
PSDU	3 bytes to 2047 bytes	
CRC	16-bit	$x^{16} + x^{12} + x^5 + 1$
DATA WHITENING	PN9	As defined in IEEE 802.15.4g standard
BIT SEQUENCE	LSB first	
TURNAROUND TIME		100 kbps, 50 kHz deviation, GFSK
Rx (CCA) to Tx (Data)	216 µs	CCA_AUTO_TX is automatic, MCR_FAST_SYNTH_CAL_ENABLE enabled
Rx (CCA) to Tx (Data)	270 µs	CCA_AUTO_TX is automatic, MCR_FAST_SYNTH_CAL_ENABLE disabled
Rx (Data) to Tx (ACK)	210 µs	RX_TO_TX_AUTO_TURNAROUND, MCR_FAST_SYNTH_CAL_ENABLE enabled
Rx (Data) to Tx (ACK)	350 µs	RX_TO_TX_AUTO_TURNAROUND, MCR_FAST_SYNTH_CAL_ENABLE disabled
Tx (Data) to Rx	125 µs	TX_TO_RX_AUTO_TURNAROUND, MCR_FAST_SYNTH_CAL_ENABLE enabled
Tx (Data) to Rx	281 µs	TX_TO_RX_AUTO_TURNAROUND, MCR_FAST_SYNTH_CAL_ENABLE disabled

[1] Minimum preamble requirement may vary with data rate. See the Minimum Preamble specification in Table 41.

TEST CONDITIONS

Table 41. Receiver Performance

Parameter	Typical Specification	Test Conditions/Comments
Minimum Preamble	4 bytes[1]	50 kbps, 25 kHz deviation
	4 bytes	100 kbps, 50 kHz deviation
	4 bytes	100 kbps, 25 kHz deviation
	4 bytes	150 kbps, 37.5 kHz deviation
	5 bytes	200 kbps, 100 kHz deviation
	5 bytes	200 kbps, 50 kHz deviation
	5 bytes	300 kbps, 75 kHz deviation
Sensitivity—4-Byte Preamble	−102.0 dBm	50 kbps, 25 kHz deviation, AGC on, AFC on, IFBW = 100 kHz, PSDU length = 12 bytes, PER = 1%
Sensitivity—4-Byte Preamble	−100.0 dBm	100 kbps, 50 kHz deviation, AGC on, AFC on, IFBW = 150 kHz, PSDU length = 12 bytes, PER = 1%
Sensitivity—8-Byte Preamble	−100.0 dBm	100 kbps, 25 kHz deviation, AGC on, AFC on, IFBW = 150 kHz, PSDU length = 12 bytes, PER = 1%
Sensitivity—12-Byte Preamble	−98.0 dBm	150 kbps, 37.5 kHz deviation, AGC on, AFC on, IFBW = 150 kHz, PSDU length = 12 bytes, PER = 1%
Sensitivity—16-Byte Preamble	−96.0 dBm	200 kbps, 100 kHz deviation, AGC on, AFC on, IFBW = 200 kHz, PSDU length = 12 bytes, PER = 1%
Sensitivity—16-Byte Preamble	−96.0 dBm	200 kbps, 50 kHz deviation, AGC on, AFC on, IFBW = 200 kHz, PSDU length = 12 bytes, PER = 1%
Sensitivity—24-Byte Preamble	−94.0 dBm[2]	300 kbps, 75 kHz deviation, AGC on, AFC on, IFBW = 300 kHz, PSDU length = 12 bytes, PER = 1%

[1] Minimum preamble for <10% PER at receiver sensitivity input level $S = (S_0 + 10\log[R/R_0])$ dBm, $S_0 = -91$, $R_0 = 50$ kbps, R = bit rate in kbps. Preamble match (0x11B) set to 0x09.
[2] Preamble match (0x11B) set to 0x0B or 0x0C

CODE DOWNLOAD SEQUENCE

The 15d4g firmware module must be stored in the PRAM starting from Address 0x0000.

The program RAM can be written to only by using the memory block write. SPI_MEM_WR must be set to 0x1E.

The sequence to write a firmware module to program RAM is as follows:

1. Ensure that the ADF7023-J is in PHY_OFF.
2. Issue the CMD_RAM_LOAD_INIT command.
3. Write the module to program RAM using an SPI memory block write.
4. Issue the CMD_RAM_LOAD_DONE command.

The firmware module is now stored in program RAM.

The program RAM is volatile memory and must be reloaded each time the transceiver wakes up from sleep state.