Si446x API Descriptions

1. Introduction

This document provides API descriptions for the commands and properties used to control and configure the Si446x family.

2. API Summary

2.1. Command Summary

Table 1. Command Summary

Boot Commands		
Number	**Name**	**Summary**
0x02	POWER_UP	Power-up device and mode selection. Modes include operational function.
Common Commands		
Number	**Name**	**Summary**
0x00	NOP	No operation command
0x01	PART_INFO	Reports basic information about the device.
0x10	FUNC_INFO	Returns the Function revision information of the device.
0x11	SET_PROPERTY	Sets the value of a property.
0x12	GET_PROPERTY	Retrieve a property's value.
0x13	GPIO_PIN_CFG	Configures the GPIO pins.
0x14	GET_ADC_READING	Retrieve the results of possible ADC conversions.
0x15	FIFO_INFO	Provides access to transmit and receive fifo counts and reset.
0x16	PACKET_INFO	Returns information about the last packet received and optionally overrides field length.
0x17	IRCAL	Calibrate Image Rejection.
0x18	PROTOCOL_CFG	Sets the chip up for specified protocol.
0x20	GET_INT_STATUS	Returns the interrupt status byte.
0x21	GET_PH_STATUS	Returns the packet handler status.
0x22	GET_MODEM_STATUS	Returns the modem status byte.
0x23	GET_CHIP_STATUS	Returns the chip status.
0x31	START_TX	Switches to TX state and starts packet transmission.
0x32	START_RX	Switches to RX state.
0x33	REQUEST_DEVICE_STATE	Request current device state.
0x34	CHANGE_STATE	Update state machine entries.
0x44	READ_CMD_BUFF	Used to read CTS and the command response.
0x50	FRR_A_READ	Reads the fast response registers starting with A.

Table 1. Command Summary (Continued)

0x51	FRR_B_READ	Reads the fast response registers starting with B.
0x53	FRR_C_READ	Reads the fast response registers starting with C.
0x57	FRR_D_READ	Reads the fast response registers starting with D.
0x66	WRITE_TX_FIFO	Writes the TX FIFO.
0x77	READ_RX_FIFO	Reads the RX FIFO.
0x36	RX_HOP	Fast RX to RX transitions for use in frequency hopping systems

2.2. Property Summary

Common Properties			
Number	**Name**	**Default**	**Summary**
0x0000	GLOBAL_XO_TUNE	0x40	Configure crystal oscillator frequency tuning bank
0x0001	GLOBAL_CLK_CFG	0	Clock configuration options
0x0002	GLOBAL_LOW_BATT_THRESH	0x18	Low battery threshold
0x0003	GLOBAL_CONFIG	0	Global configuration settings
0x0004	GLOBAL_WUT_CONFIG	0x00	GLOBAL WUT configuation
0x0005	GLOBAL_WUT_M_15_8	0x00	Configure WUT_M_15_8
0x0006	GLOBAL_WUT_M_7_0	0x01	Configure WUT_M_7_0
0x0007	GLOBAL_WUT_R	0x00	Configure WUT_R
0x0008	GLOBAL_WUT_LDC	0x00	Configure WUT_LDC
0x0100	INT_CTL_ENABLE	0x04	Interrupt enable property
0x0101	INT_CTL_PH_ENABLE	0x00	Packet handler interrupt enable property
0x0102	INT_CTL_MODEM_ENABLE	0x00	Modem interrupt enable property
0x0103	INT_CTL_CHIP_ENABLE	0x04	Chip interrupt enable property
0x0200	FRR_CTL_A_MODE	0x01	Fast Response Register A Configuration
0x0201	FRR_CTL_B_MODE	0x02	Fast Response Register B Configuration
0x0202	FRR_CTL_C_MODE	0x09	Fast Response Register C Configuration
0x0203	FRR_CTL_D_MODE	0x00	Fast Response Register D Configuration
0x1000	PREAMBLE_TX_LENGTH	0x08	Preamble length
0x1001	PREAMBLE_CONFIG_STD_1	0x14	Standard preamble configuration
0x1002	PREAMBLE_CONFIG_NSTD	0x00	Non-standard preamble configuation
0x1003	PREAMBLE_CONFIG_STD_2	0x0F	Standard preamble configuration
0x1004	PREAMBLE_CONFIG	0x21	Preamble configuration bits
0x1005	PREAMBLE_PATTERN_31_24	0	Preamble pattern
0x1006	PREAMBLE_PATTERN_23_16	0	Preamble pattern
0x1007	PREAMBLE_PATTERN_15_8	0	Preamble pattern
0x1008	PREAMBLE_PATTERN_7_0	0	Preamble pattern
0x1100	SYNC_CONFIG	0x01	Sync configuration bits
0x1101	SYNC_BITS_31_24	0x2D	Byte 3 of sync word
0x1102	SYNC_BITS_23_16	0xD4	Byte 2 of sync word
0x1103	SYNC_BITS_15_8	0x2D	Byte 1 of sync word

Common Properties			
Number	**Name**	**Default**	**Summary**
0x1104	SYNC_BITS_7_0	0xD4	Byte 0 of sync word
0x120A	PKT_LEN_ADJUST	0	Adjust length field by this amount to derive the byte count of the variable length field.
0x120B	PKT_TX_THRESHOLD	0x30	TX almost empty threshold.
0x120C	PKT_RX_THRESHOLD	0x30	RX almost full threshold.
0x120D	PKT_FIELD_1_LENGTH_12_8	0x00	Byte 1 of field length
0x120E	PKT_FIELD_1_LENGTH_7_0	0x00	Byte 0 of field length
0x120F	PKT_FIELD_1_CONFIG	0x00	Field 1 configuration bits.
0x1200	PKT_CRC_CONFIG	0	Select a CRC polynomial and seed
0x1206	PKT_CONFIG1	0	General packet configuration bits
0x1208	PKT_LEN	0x00	Provides information regarding how to use the length from the received packet.
0x1209	PKT_LEN_FIELD_SOURCE	0	Field number containing the embedded length field.
0x121A	PKT_FIELD_4_LENGTH_7_0	0x00	Byte 0 of field length
0x121B	PKT_FIELD_4_CONFIG	0x00	Field 4 configuration bits.
0x121C	PKT_FIELD_4_CRC_CONFIG	0x00	Field 4 CRC configuration bits.
0x121D	PKT_FIELD_5_LENGTH_12_8	0x00	Byte 1 of field length
0x121E	PKT_FIELD_5_LENGTH_7_0	0x00	Byte 0 of field length
0x121F	PKT_FIELD_5_CONFIG	0x00	Field 5 configuration bits.
0x1210	PKT_FIELD_1_CRC_CONFIG	0x00	Field 1 CRC configuration bits.
0x1211	PKT_FIELD_2_LENGTH_12_8	0x00	Byte 1 of field length
0x1212	PKT_FIELD_2_LENGTH_7_0	0x00	Byte 0 of field length
0x1213	PKT_FIELD_2_CONFIG	0x00	Field 2 configuration bits.
0x1214	PKT_FIELD_2_CRC_CONFIG	0x00	Field 2 CRC configuration bits.
0x1215	PKT_FIELD_3_LENGTH_12_8	0x00	Byte 1 of field length
0x1216	PKT_FIELD_3_LENGTH_7_0	0x00	Byte 0 of field length
0x1217	PKT_FIELD_3_CONFIG	0x00	Field 3 configuration bits.
0x1218	PKT_FIELD_3_CRC_CONFIG	0x00	Field 3 CRC configuration bits.
0x1219	PKT_FIELD_4_LENGTH_12_8	0x00	Byte 1 of field length
0x122A	PKT_RX_FIELD_3_LENGTH_7_0	0x00	Byte 0 of field length for RX
0x122B	PKT_RX_FIELD_3_CONFIG	0x00	Field 3 configuration bits for RX.

Common Properties			
Number	**Name**	**Default**	**Summary**
0x122C	PKT_RX_FIELD_3_CRC_CONFIG	0x00	Field 3 CRC configuration bits for RX.
0x122D	PKT_RX_FIELD_4_LENGTH_12_8	0x00	Byte 1 of field length for RX
0x122E	PKT_RX_FIELD_4_LENGTH_7_0	0x00	Byte 0 of field length for RX
0x122F	PKT_RX_FIELD_4_CONFIG	0x00	Field 4 configuration bits for RX.
0x1220	PKT_FIELD_5_CRC_CONFIG	0x00	Field 5 CRC configuration bits.
0x1221	PKT_RX_FIELD_1_LENGTH_12_8	0x00	Byte 1 of field length for RX
0x1222	PKT_RX_FIELD_1_LENGTH_7_0	0x00	Byte 0 of field length for RX
0x1223	PKT_RX_FIELD_1_CONFIG	0x00	Field 1 configuration bits for RX.
0x1224	PKT_RX_FIELD_1_CRC_CONFIG	0x00	Field 1 CRC configuration bits for RX.
0x1225	PKT_RX_FIELD_2_LENGTH_12_8	0x00	Byte 1 of field length for RX
0x1226	PKT_RX_FIELD_2_LENGTH_7_0	0x00	Byte 0 of field length for RX
0x1227	PKT_RX_FIELD_2_CONFIG	0x00	Field 2 configuration bits for RX.
0x1228	PKT_RX_FIELD_2_CRC_CONFIG	0x00	Field 2 CRC configuration bits for RX.
0x1229	PKT_RX_FIELD_3_LENGTH_12_8	0x00	Byte 1 of field length for RX
0x1230	PKT_RX_FIELD_4_CRC_CONFIG	0x00	Field 4 CRC configuration bits for RX.
0x1231	PKT_RX_FIELD_5_LENGTH_12_8	0x00	Byte 1 of field length for RX
0x1232	PKT_RX_FIELD_5_LENGTH_7_0	0x00	Byte 0 of field length for RX
0x1233	PKT_RX_FIELD_5_CONFIG	0x00	Field 5 configuration bits for RX.
0x1234	PKT_RX_FIELD_5_CRC_CONFIG	0x00	Field 5 CRC configuration bits for RX.
0x200A	MODEM_FREQ_DEV_2	0x00	Byte 2 of TX frequency deviation (a 17-bit unsigned number). This only programs the MSB of TX frequency deviation.
0x200B	MODEM_FREQ_DEV_1	0x06	Byte 1 of frequency deviation.
0x200C	MODEM_FREQ_DEV_0	0xD3	Byte 0 of frequency deviation.
0x2000	MODEM_MOD_TYPE	0x02	Modulation Type
0x2001	MODEM_MAP_CONTROL	0x80	Controls bit mapping.
0x2003	MODEM_DATA_RATE_2	0x0F	Byte 2 of TX data rate in bps (bits per second).
0x2004	MODEM_DATA_RATE_1	0x42	Byte 1 of TX data rate in bps (bits per second).
0x2005	MODEM_DATA_RATE_0	0x40	Byte 0 of TX data rate in bps (bits per second).
0x204A	MODEM_RSSI_THRESH	0xFF	RSSI threshold control
0x204B	MODEM_RSSI_JUMP_THRESH	0x0C	RSSI jumping detection threshold.

Common Properties			
Number	**Name**	**Default**	**Summary**
0x204C	MODEM_RSSI_CONTROL	0x01	RSSI control
0x204D	MODEM_RSSI_CONTROL2	0x00	RSSI control
0x204E	MODEM_RSSI_COMP	0x32	RSSI reading offset.
0x2049	MODEM_ANT_DIV_CONTROL	0x80	Specifies antenna diversity controls. Antenna diversity mode is valid for standard packet only.
0x2200	PA_MODE	0x08	PA operating mode and groups.
0x2201	PA_PWR_LVL	0x7F	PA Level Configuration
0x2202	PA_BIAS_CLKDUTY	0x00	PA Bias and TX clock duty cycle configuration
0x2203	PA_TC	0x5D	PA cascode ramping Configuration
0x300A	MATCH_MASK_4	0x00	Match 4 mask.
0x300B	MATCH_CTRL_4	0x00	Match 4 configuration.
0x3000	MATCH_VALUE_1	0x00	Match 1 value.
0x3001	MATCH_MASK_1	0x00	Match 1 mask.
0x3002	MATCH_CTRL_1	0x00	Packet match enable and match 1 configuration.
0x3003	MATCH_VALUE_2	0x00	Match 2 value.
0x3004	MATCH_MASK_2	0x00	Match 2 mask.
0x3005	MATCH_CTRL_2	0x00	Match 2 configuration.
0x3006	MATCH_VALUE_3	0x00	Match 3 value.
0x3007	MATCH_MASK_3	0x00	Match 3 mask.
0x3008	MATCH_CTRL_3	0x00	Match 3 configuration.
0x3009	MATCH_VALUE_4	0x00	Match 4 value.
0x4000	FREQ_CONTROL_INTE	0x3C	Frac-N PLL integer number.
0x4001	FREQ_CONTROL_FRAC_2	0x08	Byte 2 of Frac-N PLL fraction number.
0x4002	FREQ_CONTROL_FRAC_1	0x00	Byte 1 of Frac-N PLL fraction number.
0x4003	FREQ_CONTROL_FRAC_0	0x00	Byte 0 of Frac-N PLL fraction number.
0x4004	FREQ_CONTROL_CHANNEL_STEP_SIZE_1	0x00	Byte 1 of channel step size.
0x4005	FREQ_CONTROL_CHANNEL_STEP_SIZE_0	0x00	Byte 0 of channel step size.
0x4007	FREQ_CONTROL_VCOCNT_RX_ADJ	0xFF	VCO target count adjustment for RX
0x5000	RX_HOP_CONTROL	0x04	RX hop control.
0x5001	RX_HOP_TABLE_SIZE	0x01	Number of entries in the RX hop table.

Common Properties			
Number	**Name**	**Default**	**Summary**
0x5002	RX_HOP_TABLE_ENTRY_0	0	No.1 entry in RX hopping table.
0x500x	RX_HOP_TABLE_ENTRY_xx	1	Entries 2-63 in RX hopping table.
0x5041	RX_HOP_TABLE_ENTRY_63	2	No.64 entry in RX hopping table.

3. Commands

3.1. Boot Commands

3.1.1. POWER_UP

- Summary: *Power-up device and mode selection. Modes include operational function*
- Purpose:
  - Power-up the device with the specified function. Power-up is complete when the CTS bit is set. This command may take longer to set the CTS bit than other commands.
- Command Stream

POWER_UP Command	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CMD	0x02							
BOOT_OPTIONS	PATCH	0	FUNC[5:0]					
XTAL_OPTIONS	0000000							TCXO
XO_FREQ	XO_FREQ[31:24]							
XO_FREQ	XO_FREQ[23:16]							
XO_FREQ	XO_FREQ[15:8]							
XO_FREQ	XO_FREQ[7:0]							

- Reply Stream

POWER_UP Reply	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CTS	CTS[7:0]							

- Parameters:
  - PATCH - Select patch mode.
    0 = Copy selected functional image from OTP and boot device.
    1 = Indicates a patch has been applied. Validate patched image matches function selected and boot.
  - FUNC[5:0] - Selects the boot function of the device
    0 = Stay in bootload
    1 = Boot main application image
  - TCXO - Select if TCXO is in use.
    0 = TCXO not used, XTAL used.
    1 = TCXO used.
  - XO_FREQ[31:0] - Frequency of TCXO or external crystal oscillator in Hz. The default is 30000000 (30MHz).
    Range: 25000000 to 32000000
- Response:
  - None

3.2. Common Commands

3.2.1. NOP

- Summary: *No operation command*
- Purpose:
  - Can be used to ensure communication with the device.
- Command Stream

NOP Command	7	6	5	4	3	2	1	0
CMD	0x00							

- Reply Stream

NOP Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							

- Parameters:
  - None
- Response:
  - None

3.2.2. PART_INFO

- Summary: *Reports basic information about the device.*
- Purpose:
  - Returns Part Number, Part Version, ROM ID, etc
- Command Stream

PART_INFO Command	7	6	5	4	3	2	1	0
CMD	0x01							

- Reply Stream

PART_INFO Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
CHIPREV	CHIPREV[7:0]							
PART	PART[15:8]							
PART	PART[7:0]							
PBUILD	PBUILD[7:0]							
ID	ID[15:8]							
ID	ID[7:0]							
CUSTOMER	CUSTOMER[7:0]							
ROMID	ROMID[7:0]							

- Parameters:
  - None
- Response:
  - CHIPREV[7:0] - Chip Mask Revision
  - PART[15:0] - Part Number. (e.g. Si4461 will return - 0x4461)
  - PBUILD[7:0] - Part Build
  - ID[15:0] - ID
  - CUSTOMER[7:0] - Customer ID
  - ROMID[7:0] - ROM ID

3.2.3. FUNC_INFO

- Summary: *Returns the Function revision information of the device.*
- Purpose:
  - Return Function revision numbers for currently loaded functional mode firmware. Contrast with PART_INFO
- Command Stream

FUNC_INFO Command	7	6	5	4	3	2	1	0
CMD	0x10							

- Reply Stream

FUNC_INFO Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
REVEXT	REVEXT[7:0]							
REVBRANCH	REVBRANCH[7:0]							
REVINT	REVINT[7:0]							
PATCH	PATCH[15:8]							
PATCH	PATCH[7:0]							
FUNC	FUNC[7:0]							

- Parameters:
  - None
- Response:
  - REVEXT[7:0] - External revision number
    Range: 0 to 255
  - REVBRANCH[7:0] - Branch revision number
    Range: 0 to 255
  - REVINT[7:0] - Internal revision number
    Range: 0 to 255
  - PATCH[15:0]
    ID of applied patch. This is also the last 2 bytes in the associated patch file (*.csg).
    0x0000 = No patch applied.
  - FUNC[7:0] - Current functional mode
    0 = Part is currently in boot mode
    1 = Part is currently running main application image

3.2.4. SET_PROPERTY

- Summary: *Sets the value of a property.*
- Purpose:
  - Sets a property common to one or more commands. These are similar to parameters for a api command but are not expected to change frequently and may be controlled by higher layers of the user's software. Setting some properties may not cause the device to take immediate action, however the property will take affect once a command which uses it is issued.
- Command Stream

SET_PROPERTY Command	7	6	5	4	3	2	1	0
CMD	0x11							
GROUP	GROUP[7:0]							
NUM_PROPS	NUM_PROPS[7:0]							
START_PROP	START_PROP[7:0]							
DATA0	DATA0[7:0]							
DATA1	DATA1[7:0]							
DATA2	DATA2[7:0]							
DATA3	DATA3[7:0]							
DATA4	DATA4[7:0]							
DATA5	DATA5[7:0]							
DATA6	DATA6[7:0]							
DATA7	DATA7[7:0]							
DATA8	DATA8[7:0]							
DATA9	DATA9[7:0]							
DATA10	DATA10[7:0]							
DATA11	DATA11[7:0]							

- Reply Stream

SET_PROPERTY Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							

- Parameters:
  - GROUP[7:0] - Selects the group of the property to set.
  - NUM_PROPS[7:0] - Number of properties to write starting at START_PROP.
    Range: 1 to 12
  - START_PROP[7:0] - Selects the property index to set. The available properties are determined by the part number and the POWER_UP :FUNC selection.
  - DATA0[7:0] - Value of the property START_PROP
  - DATA1[7:0] - Value of the property START_PROP + 1 (don't care if NUM_PROPS < 2)
  - DATA2[7:0] - Value of the property START_PROP + 2 (don't care if NUM_PROPS < 3)
  - DATA3[7:0] - Value of the property START_PROP + 3 (don't care if NUM_PROPS < 4)
  - DATA4[7:0] - Value of the property START_PROP + 4 (don't care if NUM_PROPS < 5)

- DATA5[7:0] - Value of the property START_PROP + 5 (don't care if NUM_PROPS < 6)
- DATA6[7:0] - Value of the property START_PROP + 6 (don't care if NUM_PROPS < 7)
- DATA7[7:0] - Value of the property START_PROP + 7 (don't care if NUM_PROPS < 8)
- DATA8[7:0] - Value of the property START_PROP + 8 (don't care if NUM_PROPS < 9)
- DATA9[7:0] - Value of the property START_PROP + 9 (don't care if NUM_PROPS < 10)
- DATA10[7:0] - Value of the property START_PROP + 10 (don't care if NUM_PROPS < 11)
- DATA11[7:0] - Value of the property START_PROP + 11 (don't care if NUM_PROPS < 12)

- Response:
  - None

3.2.5. GET_PROPERTY

- Summary: *Retrieve a property's value.*
- Purpose:
  - Retrieve a property's value; The value will either be the default or the value set with SET_PROPERTY.
- Command Stream

GET_PROPERTY Command	7	6	5	4	3	2	1	0
CMD	0x12							
GROUP	GROUP[7:0]							
NUM_PROPS	NUM_PROPS[7:0]							
START_PROP	START_PROP[7:0]							

- Reply Stream

GET_PROPERTY Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
DATA0	DATA0[7:0]							
DATA1	DATA1[7:0]							
DATA2	DATA2[7:0]							
DATA3	DATA3[7:0]							
DATA4	DATA4[7:0]							
DATA5	DATA5[7:0]							
DATA6	DATA6[7:0]							
DATA7	DATA7[7:0]							
DATA8	DATA8[7:0]							
DATA9	DATA9[7:0]							
DATA10	DATA10[7:0]							
DATA11	DATA11[7:0]							
DATA12	DATA12[7:0]							
DATA13	DATA13[7:0]							
DATA14	DATA14[7:0]							
DATA15	DATA15[7:0]							

- Parameters:
  - GROUP[7:0] - Selects the group of the properties to retrieve.
  - NUM_PROPS[7:0] - Number of properties to retrieve starting at START_PROP.
    Range: 1 to 16
  - START_PROP[7:0] - Selects the first property index to retrieve. The available properties are determined by the part number and the POWER_UP:FUNC selection.

- Response:
  - DATA0[7:0] - Value of the property START_PROP
  - DATA1[7:0] - Value of the property START_PROP + 1 (don't care if NUM_PROPS < 2)
  - DATA2[7:0] - Value of the property START_PROP + 2 (don't care if NUM_PROPS < 3)
  - DATA3[7:0] - Value of the property START_PROP + 3 (don't care if NUM_PROPS < 4)
  - DATA4[7:0] - Value of the property START_PROP + 4 (don't care if NUM_PROPS < 5)
  - DATA5[7:0] - Value of the property START_PROP + 5 (don't care if NUM_PROPS < 6)
  - DATA6[7:0] - Value of the property START_PROP + 6 (don't care if NUM_PROPS < 7)
  - DATA7[7:0] - Value of the property START_PROP + 7 (don't care if NUM_PROPS < 8)
  - DATA8[7:0] - Value of the property START_PROP + 8 (don't care if NUM_PROPS < 9)
  - DATA9[7:0] - Value of the property START_PROP + 9 (don't care if NUM_PROPS < 10)
  - DATA10[7:0] - Value of the property START_PROP + 10 (don't care if NUM_PROPS < 11)
  - DATA11[7:0] - Value of the property START_PROP + 11 (don't care if NUM_PROPS < 12
  - DATA12[7:0] - Value of the property STAT_PROP + 12 (don't care if NUM_PROPS < 13)
  - DATA13[7:0] - Value of the property START_PROP + 13 (don't care if NUM_PROPS < 14)
  - DATA14[7:0] - Value of the property START_PROP + 14 (don't care if NUM_PROPS < 15)
  - DATA15[7:0] - Value of the property START_PROP + 15 (don't care if NUM_PROPS < 16)

3.2.6. GPIO_PIN_CFG

- Summary: *Configures the GPIO pins*
- Command Stream

GPIO_PIN_CFG Command	7	6	5	4	3	2	1	0
CMD	0x13							
GPIO0	0	GPIO0_PULL_CTL	GPIO0_MODE[5:0]					
GPIO1	0	GPIO1_PULL_CTL	GPIO1_MODE[5:0]					
GPIO2	0	GPIO2_PULL_CTL	GPIO2_MODE[5:0]					
GPIO3	0	GPIO3_PULL_CTL	GPIO3_MODE[5:0]					
NIRQ	0	NIRQ_DRV_PULL	NIRQ_MODE[5:0]					
SDO	0	SDO_PULL_CTL	SDO_MODE[5:0]					
GEN_CONFIG	0	DRV_STRENGTH[1:0]		00000				

- Reply Stream

GPIO_PIN_CFG Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
GPIO0	GPIO0_STATE	X	GPIO0r_[5:0]					
GPIO1	GPIO1_STATE	X	GPIO1r_[5:0]					
GPIO2	GPIO2_STATE	X	GPIO2r_[5:0]					
GPIO3	GPIO3_STATE	X	GPIO3r_[5:0]					
NIRQ	NIRQSTATE	X	NIRQr_[5:0]					
SDO	SDOSTATE	X	SDOr_[5:0]					
GEN_CONFIG	X	DRV_STRENGTH[1:0]		XXXXX				

- Parameters:
  - GPIO0_PULL_CTL
    0 = Disable pullup. Recommended setting if pin is driven.
    1 = Enable pullup.
  - GPIO0_MODE[5:0]
    0 = Do not modify the behavior of this pin.
    1 = Input and output drivers disabled.
    2 = CMOS output driven low.
    3 = CMOS output driven high.
    4 = CMOS input. This is used for all GPIO functions that require the pin to be an input (e.g., TX Direct Mode Data In). The actual function of this pin is controlled by other properties.
    5 = 32 kHz clock. Outputs 32 kHz clock selected using GLOBAL_CLK_CFG:CLK_32K_SEL. Output low if the 32 kHz clock is not enabled.
    6 = BOOT_CLK. Outputs boot clock. This will only output when the device is in SPI ACTIVE state because that is the only time the boot clock is active.

7 = Divided MCU clock. Outputs divided clock. Output divided boot clock in SPI ACTIVE state, Output low while in SLEEP state as the divided clock source is not running, and Output divided XTAL in all other states. The divider is configured using GLOBAL_CLK_CFG:DIVIDED_CLK_SEL

8 = CTS. Output High when clear to send a new command, output low otherwise.

9 = INV_CTS. Output low when clear to send a new command, output high otherwise.

10 = Output low unless command overlap occurs. When command overlap occurs output goes high until the rising edge of CTS.

11 = SPI. Serial data out.

12 = Output low until power on reset is complete then output high.

13 = Output low normally. Pulses high when calibration timer expires. To use the calibration timer, the 32 kHz clock must be enabled. Calibration timer period is configured using GLOBAL_WUT_CONFIG:WUT_CAL_PERIOD and enabled by GLOBAL_WUT_CONFIG:CAL_EN.

14 = Output low normally. Pulses high when wakeup timer expires. To use the wakeup timer the 32 kHz clock must be enabled. The wut period is configured using GLOBAL_WUT_M_15_8, GLOBAL_WUT_M_7_0 and GLOBAL_WUT_R and enabled by GLOBAL_WUT_CONFIG:WUT_EN.

15 = Unused0

16 = TX data CLK output to be used in conjuction with TX Data pin.

17 = RX data CLK output to be used in conjuction with RX Data pin.

18 = unused1

19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4..

20 = RX data.

21 = RX raw data.

22 = Antenna 1 Switch used for antenna diversity.

23 = Antenna 2 Switch used for antenna diversity.

24 = High when a valid preamble is detected. Returns to output low after a packet is received or sync word timeout occurs.

25 = High when an invalid preamble is detected. Output low normally. Pulses output high when the preamble is not detected within a period of time (determined by PREAMBLE_CONFIG_STD_2:RX_PREAMBLE_TIMEOUT) after the demodulator is enabled.

26 = High when a sync word is detected. Returns to output low after the packet is received.

27 = High when RSSI above clear channel assesment threshold, low when below threshold. Threshold set by MODEM_RSSI_THRESH.

32 = High while in the transmit state.

33 = High while in the receive state.

34 = High while the rx fifo is almost full.

35 = High while the tx fifo is almost empty.

36 = High while the battery voltage is low.

37 = High when RSSI above clear channel assesment threshold. Goes low on sync detect or exiting rx state.

38 = Toggles when hop occurs.

39 = Toggles when the hop table wraps.

- GPIO1_PULL_CTL

0 = Disable pullup. Recommended setting if pin is driven.

1 = Enable pullup.

- GPIO1_MODE[5:0]

0 = Do not modify the behavior of this pin.

1 = Input and output drivers disabled.

2 = CMOS output driven low.

3 = CMOS output driven high.

4 = CMOS input. This is used for all GPIO functions that require the pin to be an input (e.g. TX Direct Mode Data In). The actual function of this pin is controlled by other properties.

5 = 32 kHz clock. Outputs 32 kHz clock selected using GLOBAL_CLK_CFG:CLK_32K_SEL. Output low if the 32 kHz clock is not enabled.

6 = BOOT_CLK. Outputs boot clock. This will only output when the device is in SPI ACTIVE state because that is the only time the boot clock is active.

7 = Divided MCU clock. Outputs divided clock. Output divided boot clock in SPI ACTIVE state, Output low while in SLEEP state as the divided clock source is not running, and Output divided XTAL in all other states. The divider is configured using GLOBAL_CLK_CFG:DIVIDED_CLK_SEL

8 = CTS. Output High when clear to send a new command, output low otherwise.

9 = INV_CTS. Output low when clear to send a new command, output high otherwise.

10 = Output low unless command overlap occurs. When command overlap occurs output goes high until the rising edge of CTS.

11 = SPI. Serial data out.

12 = Output low until power on reset is complete then output high.

13 = Output low normally. Pulses high when calibration timer expires. To use the calibration timer, the 32 kHz clock must be enabled. Calibration timer period is configured using GLOBAL_WUT_CONFIG:WUT_CAL_PERIOD and enabled by GLOBAL_WUT_CONFIG:CAL_EN.

14 = Output low normally. Pulses high when wakeup timer expires. To use the wakeup timer the 32 kHz clock must be enabled. The wut period is configured using GLOBAL_WUT_M_15_8, GLOBAL_WUT_M_7_0 and GLOBAL_WUT_R and enabled by GLOBAL_WUT_CONFIG:WUT_EN.

15 = unused0

16 = TX data CLK output to be used in conjuction with TX Data pin.

17 = RX data CLK output to be used in conjuction with RX Data pin.

18 = unused1

19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.

20 = RX data.

21 = RX raw data.

22 = Antenna 1 Switch used for antenna diversity.

23 = Antenna 2 Switch used for antenna diversity.

24 = High when a valid preamble is detected. Returns to output low after a packet is received or sync word timeout occurs.

25 = High when an invalid preamble is detected. Output low normally. Pulses output high when the preamble is not detected within a period of time (determined by PREAMBLE_CONFIG_STD_2:RX_PREAMBLE_TIMEOUT) after the demodulator is enabled.

26 = High when a sync word is detected. Returns to output low after the packet is received.

27 = High when RSSI above clear channel assesment threshold, low when below threshold. Threshold set by MODEM_RSSI_THRESH.

32 = High while in the transmit state.

33 = High while in the receive state.

34 = High while the rx fifo is almost full.

35 = High while the tx fifo is almost empty.

36 = High while the battery voltage is low.

37 = High when RSSI above clear channel assesment threshold. Goes low on sync detect or exiting rx state.

38 = Toggles when hop occurs.

39 = Toggles when the hop table wraps.

- GPIO2_PULL_CTL

  0 = Disable pullup. Recommended setting if pin is driven.

  1 = Enable pullup.

- GPIO2_MODE[5:0]

  0 = Do not modify the behavior of this pin.

  1 = Input and output drivers disabled.

  2 = CMOS output driven low.

  3 = CMOS output driven high.

  4 = CMOS input. This is used for all GPIO functions that require the pin to be an input (e.g. TX Direct Mode Data In). The actual function of this pin is controlled by other properties.

5 = 32 kHz clock. Outputs 32 kHz clock selected using GLOBAL_CLK_CFG:CLK_32K_SEL. Output low if the 32 kHz clock is not enabled.

6 = BOOT_CLK. Outputs boot clock. This will only output when the device is in SPI ACTIVE state because that is the only time the boot clock is active.

7 = Divided MCU clock. Outputs divided clock. Output divided boot clock in SPI ACTIVE state, Output low while in SLEEP state as the divided clock source is not running, and Output divided XTAL in all other states. The divider is configured using GLOBAL_CLK_CFG:DIVIDED_CLK_SEL

8 = CTS. Output High when clear to send a new command, output low otherwise.

9 = INV_CTS. Output low when clear to send a new command, output high otherwise.

10 = Output low unless command overlap occurs. When command overlap occurs output goes high until the rising edge of CTS.

11 = SPI. Serial data out.

12 = Output low until power on reset is complete then output high.

13 = Output low normally. Pulses high when calibration timer expires. To use the calibration timer, the 32 kHz clock must be enabled. Calibration timer period is configured using GLOBAL_WUT_CONFIG:WUT_CAL_PERIOD and enabled by GLOBAL_WUT_CONFIG:CAL_EN.

14 = Output low normally. Pulses high when wakeup timer expires. To use the wakeup timer the 32 kHz clock must be enabled. The wut period is configured using GLOBAL_WUT_M_15_8, GLOBAL_WUT_M_7_0 and GLOBAL_WUT_R and enabled by GLOBAL_WUT_CONFIG:WUT_EN.

15 = Unused0

16 = TX data CLK output to be used in conjuction with TX Data pin.

17 = RX data CLK output to be used in conjuction with RX Data pin.

18 = unused1

19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.

20 = RX data.

21 = RX raw data.

22 = Antenna 1 Switch used for antenna diversity.

23 = Antenna 2 Switch used for antenna diversity.

24 = High when a valid preamble is detected. Returns to output low after a packet is received or sync word timeout occurs.

25 = High when an invalid preamble is detected. Output low normally. Pulses output high when the preamble is not detected within a period of time (determined by PREAMBLE_CONFIG_STD_2:RX_PREAMBLE_TIMEOUT) after the demodulator is enabled.

26 = High when a sync word is detected. Returns to output low after the packet is received.

27 = High when RSSI above clear channel assesment threshold, low when below threshold. Threshold set by MODEM_RSSI_THRESH.

32 = High while in the transmit state.

33 = High while in the receive state.

34 = High while the rx fifo is almost full.

35 = High while the tx fifo is almost empty.

36 = High while the battery voltage is low.

37 = High when RSSI above clear channel assesment threshold. Goes low on sync detect or exiting rx state.

38 = Toggles when hop occurs.

39 = Toggles when the hop table wraps.

- GPIO3_PULL_CTL

  0 = Disable pullup. Recommended setting if pin is driven.

  1 = Enable pullup.

- GPIO3_MODE[5:0]

  0 = Do not modify the behavior of this pin.

  1 = Input and output drivers disabled.

  2 = CMOS output driven low.

  3 = CMOS output driven high.

4 = CMOS input. This is used for all GPIO functions that require the pin to be an input (e.g. TX Direct Mode Data In). The actual function of this pin is controlled by other properties.

5 = 32 kHz clock. Outputs 32 kHz clock selected using GLOBAL_CLK_CFG:CLK_32K_SEL. Output low if the 32 kHz clock is not enabled.

6 = BOOT_CLK. Outputs boot clock. This will only output when the device is in SPI ACTIVE state because that is the only time the boot clock is active.

7 = Divided MCU clock. Outputs divided clock. Output divided boot clock in SPI ACTIVE state, Output low while in SLEEP state as the divided clock source is not running, and Output divided XTAL in all other states. The divider is configured using GLOBAL_CLK_CFG:DIVIDED_CLK_SEL

8 = CTS. Output High when clear to send a new command, output low otherwise.

9 = INV_CTS. Output low when clear to send a new command, output high otherwise.

10 = Output low unless command overlap occurs. When command overlap occurs output goes high until the rising edge of CTS.

11 = SPI. Serial data out.

12 = Output low until power on reset is complete then output high.

13 = Output low normally. Pulses high when calibration timer expires. To use the calibration timer, the 32 kHz clock must be enabled. Calibration timer period is configured using GLOBAL_WUT_CONFIG:WUT_CAL_PERIOD and enabled by GLOBAL_WUT_CONFIG:CAL_EN.

14 = Output low normally. Pulses high when wakeup timer expires. To use the wakeup timer the 32 kHz clock must be enabled. The wut period is configured using GLOBAL_WUT_M_15_8, GLOBAL_WUT_M_7_0 and GLOBAL_WUT_R and enabled by GLOBAL_WUT_CONFIG:WUT_EN.

15 = unused0

16 = TX data CLK output to be used in conjuction with TX Data pin.

17 = RX data CLK output to be used in conjuction with RX Data pin.

18 = unused1

19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.

20 = RX data.

21 = RX raw data.

22 = Antenna 1 Switch used for antenna diversity.

23 = Antenna 2 Switch used for antenna diversity.

24 = High when a valid preamble is detected. Returns to output low after a packet is received or sync word timeout occurs.

25 = High when an invalid preamble is detected. Output low normally. Pulses output high when the preamble is not detected within a period of time (determined by PREAMBLE_CONFIG_STD_2:RX_PREAMBLE_TIMEOUT) after the demodulator is enabled.

26 = High when a sync word is detected. Returns to output low after the packet is received.

27 = High when RSSI above clear channel assesment threshold, low when below threshold. Threshold set by MODEM_RSSI_THRESH.

32 = High while in the transmit state.

33 = High while in the receive state.

34 = High while the rx fifo is almost full.

35 = High while the tx fifo is almost empty.

36 = High while the battery voltage is low.

37 = High when RSSI above clear channel assesment threshold. Goes low on sync detect or exiting rx state.

38 = Toggles when hop occurs.

39 = Toggles when the hop table wraps.

- NIRQ_DRV_PULL
- 0 = Disable pullup. Recommended setting if pin is driven.
- 1 = Enable pullup.
- NIRQ_MODE[5:0]

0 = Do not modify the behavior of this pin.

1 = Input and output drivers disabled.

2 = CMOS output driven low.

3 = CMOS output driven high.

4 = CMOS input. This is used for all GPIO functions that require the pin to be an input (e.g. TX Direct Mode Data In). The actual function of this pin is controlled by other properties.

5 = 32 kHz clock. Outputs 32 kHz clock selected using GLOBAL_CLK_CFG:CLK_32K_SEL. Output low if the 32 kHz clock is not enabled.

6 = BOOT_CLK. Outputs boot clock. This will only output when the device is in SPI ACTIVE state because that is the only time the boot clock is active.

7 = Divided MCU clock. Outputs divided clock. Output divided boot clock in SPI ACTIVE state, Output low while in SLEEP state as the divided clock source is not running, and Output divided XTAL in all other states. The divider is configured using GLOBAL_CLK_CFG:DIVIDED_CLK_SEL

8 = CTS. Output High when clear to send a new command, output low otherwise.

9 = INV_CTS. Output low when clear to send a new command, output high otherwise.

10 = Output low unless command overlap occurs. When command overlap occurs output goes high until the rising edge of CTS.

11 = SPI. Serial data out.

12 = Output low until power on reset is complete then output high.

13 = Output low normally. Pulses high when calibration timer expires. To use the calibration timer, the 32 kHz clock must be enabled. Calibration timer period is configured using GLOBAL_WUT_CONFIG:WUT_CAL_PERIOD and enabled by GLOBAL_WUT_CONFIG:CAL_EN.

14 = Output low normally. Pulses high when wakeup timer expires. To use the wakeup timer the 32 kHz clock must be enabled. The wut period is configured using GLOBAL_WUT_M_15_8, GLOBAL_WUT_M_7_0 and GLOBAL_WUT_R and enabled by GLOBAL_WUT_CONFIG:WUT_EN.

15 = Unused0

16 = TX data CLK output to be used in conjuction with TX Data pin.

17 = RX data CLK output to be used in conjuction with RX Data pin.

18 = unused1

19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.

20 = RX data.

21 = RX raw data.

22 = Antenna 1 Switch used for antenna diversity.

23 = Antenna 2 Switch used for antenna diversity.

24 = High when a valid preamble is detected. Returns to output low after a packet is received or sync word timeout occurs.

25 = High when an invalid preamble is detected. Output low normally. Pulses output high when the preamble is not detected within a period of time (determined by PREAMBLE_CONFIG_STD_2:RX_PREAMBLE_TIMEOUT) after the demodulator is enabled.

26 = High when a sync word is detected. Returns to output low after the packet is received.

27 = High when RSSI above clear channel assesment threshold, low when below threshold. Threshold set by MODEM_RSSI_THRESH.

39 = Toggles when the hop table wraps.

- SDO_PULL_CTL

0 = Disable pullup. Recommended setting if pin is driven.

1 = Enable pullup.

- SDO_MODE[5:0]

0 = Do not modify the behavior of this pin.

1 = Input and output drivers disabled.

2 = CMOS output driven low.

3 = CMOS output driven high.

4 = CMOS input. This is used for all GPIO functions that require the pin to be an input (e.g. TX Direct Mode Data In). The actual function of this pin is controlled by other properties.

5 = 32 kHz clock. Outputs 32 kHz clock selected using GLOBAL_CLK_CFG:CLK_32K_SEL. Output low if the

32 kHz clock is not enabled.

6 = BOOT_CLK. Outputs boot clock. This will only output when the device is in SPI ACTIVE state because that is the only time the boot clock is active.

7 = Divided MCU clock. Outputs divided clock. Output divided boot clock in SPI ACTIVE state, Output low while in SLEEP state as the divided clock source is not running, and Output divided XTAL in all other states. The divider is configured using GLOBAL_CLK_CFG:DIVIDED_CLK_SEL

8 = CTS. Output High when clear to send a new command, output low otherwise.

9 = INV_CTS. Output low when clear to send a new command, output high otherwise.

10 = Output low unless command overlap occurs. When command overlap occurs output goes high until the rising edge of CTS.

11 = SPI. Serial data out.

12 = Output low until power on reset is complete then output high.

13 = Output low normally. Pulses high when calibration timer expires. To use the calibration timer, the 32 kHz clock must be enabled. Calibration timer period is configured using GLOBAL_WUT_CONFIG:WUT_CAL_PERIOD and enabled by GLOBAL_WUT_CONFIG:CAL_EN.

14 = Output low normally. Pulses high when wakeup timer expires. To use the wakeup timer the 32 kHz clock must be enabled. The wut period is configured using GLOBAL_WUT_M_15_8, GLOBAL_WUT_M_7_0 and GLOBAL_WUT_R and enabled by GLOBAL_WUT_CONFIG:WUT_EN.

15 = Unused0

16 = TX data CLK output to be used in conjuction with TX Data pin.

17 = RX data CLK output to be used in conjuction with RX Data pin.

18 = unused1

19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.

20 = RX data.

21 = RX raw data.

22 = Antenna 1 Switch used for antenna diversity.

23 = Antenna 2 Switch used for antenna diversity.

24 = High when a valid preamble is detected. Returns to output low after a packet is received or sync word timeout occurs.

25 = High when an invalid preamble is detected. Output low normally. Pulses output high when the preamble is not detected within a period of time (determined by PREAMBLE_CONFIG_STD_2:RX_PREAMBLE_TIMEOUT) after the demodulator is enabled.

26 = High when a sync word is detected. Returns to output low after the packet is received.

27 = High when RSSI above clear channel assesment threshold, low when below threshold. Threshold set by MODEM_RSSI_THRESH.

- DRV_STRENGTH[6:5]

  0 = GPIOs configured as outputs will have highest drive strength.

  1 = GPIOs configured as outputs will have medium drive strength.

  2 = GPIOs configured as outputs will have medium drive strength.

  3 = GPIOs configured as outputs will have lowest drive strength.

- Response:
  - GPIO0_STATE

    0 = Pin was read back as a 0

    1 = Pin was read back as a 1
  - GPIO0r[5:0]

    0 = Do not modify the behavior of this pin.

    1 = Input and output drivers disabled.

    2 = CMOS output driven low.

    3 = CMOS output driven high.

    4 = CMOS input.

    5 = 32 kHz clock.

    6 = 30 MHz clock.

7 = Divided MCU clock.
8 = High when command complete, low otherwise.
9 = Low when command complete, high otherwise.
10 = High when command overlap occurs. TODO: What clears this.
11 = Serial data out.
12 = Pulses high on power on reset.
13 = Pulses high when calibration timer expires.
14 = Pulses high when wakeup timer expires.
15 = unused0
16 = TX data CLK output to be used in conjuction with TX Data pin.
17 = RX data CLK output to be used in conjuction with RX Data pin.
18 = unused1
19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.
20 = RX data.
21 = RX raw data.
22 = Antenna 1 Switch used for antenna diversity.
23 = Antenna 2 Switch used for antenna diversity.
24 = High when a valid preamble is detected. Cleared when sync is received.
25 = High when an invalid preamble is detected. TODO: What clears this
26 = High when a sync word is detected. TODO: What clears this
27 = High when RSSI above clear channel assesment threshold, low when below threshold.
32 = High while in the transmit state.
33 = High while in the receive state.
34 = High while the rx fifo is almost full.
35 = High while the tx fifo is almost empty.
36 = High while the battery voltage is low.
37 = High when RSSI above clear channel assesment threshold, goes low on sync detect or exiting rx state.
38 = Toggles when hop occurs.
39 = Toggles when the hop table wraps.

- GPIO1_STATE

  0 = Pin was read back as a 0
  1 = Pin was read back as a 1

- GPIO1r[5:0]

  0 = Do not modify the behavior of this pin.
  1 = Input and output drivers disabled.
  2 = CMOS output driven low.
  3 = CMOS output driven high.
  4 = CMOS input.
  5 = 32 kHz clock.
  6 = 30 MHz clock.
  7 = Divided MCU clock.
  8 = High when command complete, low otherwise.
  9 = Low when command complete, high otherwise.
  10 = Output low unless command overlap occurs. When command overlap occurs output goes high until the rising edge of CTS.
  11 = Serial data out.
  12 = Pulses high on power on reset.
  13 = Pulses high when calibration timer expires.
  14 = Pulses high when wakeup timer expires.
  15 = unused0

16 = TX data CLK output to be used in conjuction with TX Data pin.
17 = RX data CLK output to be used in conjuction with RX Data pin.
18 = unused1
19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.
20 = RX data.
21 = RX raw data.
22 = Antenna 1 Switch used for antenna diversity.
23 = Antenna 2 Switch used for antenna diversity.
24 = High when a valid preamble is detected. Cleared when sync is received.
25 = High when an invalid preamble is detected. TODO: What clears this
26 = High when a sync word is detected. TODO: What clears this
27 = High when RSSI above clear channel assesment threshold, low when below threshold.
32 = High while in the transmit state.
33 = High while in the receive state.
34 = High while the rx fifo is almost full.
35 = High while the tx fifo is almost empty.
36 = High while the battery voltage is low.
37 = High when RSSI above clear channel assesment threshold, goes low on sync detect or exiting rx state.
38 = Toggles when hop occurs.
39 = Toggles when the hop table wraps.

- GPIO2_STATE
  0 = Pin was read back as a 0
  1 = Pin was read back as a 1
- GPIO2r[5:0]
  0 = Do not modify the behavior of this pin.
  1 = Input and output drivers disabled.
  2 = CMOS output driven low.
  3 = CMOS output driven high.
  4 = CMOS input.
  5 = 32 kHz clock.
  6 = 30 MHz clock.
  7 = Divided MCU clock.
  8 = High when command complete, low otherwise.
  9 = Low when command complete, high otherwise.
  10 = High when command overlap occurs. TODO: What clears this.
  11 = Serial data out.
  12 = Pulses high on power on reset.
  13 = Pulses high when calibration timer expires.
  14 = Pulses high when wakeup timer expires.
  15 = unused0
  16 = TX data CLK output to be used in conjuction with TX Data pin.
  17 = RX data CLK output to be used in conjuction with RX Data pin.
  18 = unused1
  19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.
  20 = RX data.
  21 = RX raw data.
  22 = Antenna 1 Switch used for antenna diversity.
  23 = Antenna 2 Switch used for antenna diversity.
  24 = High when a valid preamble is detected. Cleared when sync is received.
  25 = High when an invalid preamble is detected. TODO: What clears this

26 = High when a sync word is detected. TODO: What clears this
27 = High when RSSI above clear channel assesment threshold, low when below threshold.
32 = High while in the transmit state.
33 = High while in the receive state.
34 = High while the rx fifo is almost full.
35 = High while the tx fifo is almost empty.
36 = High while the battery voltage is low.
37 = High when RSSI above clear channel assesment threshold, goes low on sync detect or exiting rx state.
38 = Toggles when hop occurs.
39 = Toggles when the hop table wraps.

- GPIO3STATE

0 = Pin was read back as a 0
1 = Pin was read back as a 1

- GPIO3r[5:0]

0 = Do not modify the behavior of this pin.
1 = Input and output drivers disabled.
2 = CMOS output driven low.
3 = CMOS output driven high.
4 = CMOS input.
5 = 32 kHz clock.
6 = 30 MHz clock.
7 = Divided MCU clock.
8 = High when command complete, low otherwise.
9 = Low when command complete, high otherwise.
10 = High when command overlap occurs. TODO: What clears this.
11 = Serial data out.
12 = Pulses high on power on reset.
13 = Pulses high when calibration timer expires.
14 = Pulses high when wakeup timer expires.
15 = unused0
16 = TX data CLK output to be used in conjuction with TX Data pin.
17 = RX data CLK output to be used in conjuction with RX Data pin.
18 = unused1
19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.
20 = RX data.
21 = RX raw data.
22 = Antenna 1 Switch used for antenna diversity.
23 = Antenna 2 Switch used for antenna diversity.
24 = High when a valid preamble is detected. Cleared when sync is received.
25 = High when an invalid preamble is detected. TODO: What clears this
26 = High when a sync word is detected. TODO: What clears this
27 = High when RSSI above clear channel assesment threshold, low when below threshold.
32 = High while in the transmit state.
33 = High while in the receive state.
34 = High while the rx fifo is almost full.
35 = High while the tx fifo is almost empty.
36 = High while the battery voltage is low.
37 = High when RSSI above clear channel assesment threshold, goes low on sync detect or exiting rx state.
38 = Toggles when hop occurs.
39 = Toggles when the hop table wraps.

- NIRQSTATE

0 = Pin was read back as a 0
1 = Pin was read back as a 1

- NIRQr[5:0]

0 = Do not modify the behavior of this pin.
1 = Input and output drivers disabled.
2 = CMOS output driven low.
3 = CMOS output driven high.
4 = CMOS input.
7 = Divided MCU clock.
8 = High when command complete, low otherwise.
11 = Serial data out.
12 = Pulses high on power on reset.
15 = unused0
16 = TX data CLK output to be used in conjuction with TX Data pin.
17 = RX data CLK output to be used in conjuction with RX Data pin.
18 = unused1
19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.
20 = RX data.
21 = RX raw data.
22 = Antenna 1 Switch used for antenna diversity.
23 = Antenna 2 Switch used for antenna diversity.
24 = High when a valid preamble is detected. Cleared when sync is received.
25 = High when an invalid preamble is detected. TODO: What clears this
26 = High when a sync word is detected. TODO: What clears this
27 = High when RSSI above clear channel assesment threshold, low when below threshold.
39 = Active low interrupt signal

- SDOSTATE

0 = Pin was read back as a 0
1 = Pin was read back as a 1

- SDOr[5:0]

0 = Do not modify the behavior of this pin.
1 = Input and output drivers disabled.
2 = CMOS output driven low.
3 = CMOS output driven high.
4 = CMOS input.
5 = 32 kHz clock.
7 = Divided MCU clock.
8 = High when command complete, low otherwise.
11 = Serial data out.
12 = Pulses high on power on reset.
14 = Pulses high when wakeup timer expires.
15 = unused0
16 = TX data CLK output to be used in conjuction with TX Data pin.
17 = RX data CLK output to be used in conjuction with RX Data pin.
18 = unused1
19 = TX data. This option is not to be used for Tx direct mode input. For Tx direct mode input, use option 4.
20 = RX data.
21 = RX raw data.
22 = Antenna 1 Switch used for antenna diversity.
23 = Antenna 2 Switch used for antenna diversity.

24 = High when a valid preamble is detected. Cleared when sync is received.
25 = High when an invalid preamble is detected. TODO: What clears this
26 = High when a sync word is detected. TODO: What clears this
27 = High when RSSI above clear channel assesment threshold, low when below threshold.

- DRV_STRENGTH[6:5]
  0 = GPIOs configured as outputs will have highest drive strength.
  1 = GPIOs configured as outputs will have medium drive strength.
  2 = GPIOs configured as outputs will have medium drive strength.
  3 = GPIOs configured as outputs will have lowest drive strength.

3.2.7. GET_ADC_READING

- Summary: Performs and returns results of selected ADC conversion.
- Purpose:
  - Retrieve the result of the last ADC conversion.
- Command Stream

GET_ADC_READING Command	7	6	5	4	3	2	1	0
CMD	0x14							
ADC_EN	000			TEMPERATURE_EN	BATTERY_VOLTAGE_EN	ADC_GPIO_EN	ADC_GPIO_PIN[1:0]	

- Reply Stream

GET_ADC_READING Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
GPIO_ADC	GPIO_ADC[15:8]							
GPIO_ADC	GPIO_ADC[7:0]							
BATTERY_ADC	BATTERY_ADC[15:8]							
BATTERY_ADC	BATTERY_ADC[7:0]							
TEMP_ADC	TEMP_ADC[15:8]							
TEMP_ADC	TEMP_ADC[7:0]							
TEMP_SLOPE	TEMP_SLOPE[7:0]							
TEMP_INTERCEPT	TEMP_INTERCEPT[7:0]							

- Parameters:
  - TEMPERATURE_EN

    0 = Don't do ADC conversion of temperature, will read 0 value in reply TEMPERATURE

    1 = Do ADC conversion of temperature, result in TEMP_ADC.
    Temp in Celsius = ((800 + TEMP_SLOPE) / 4096) x TEMP_ADC – (INTERCEPT / 2 + 256)
  - BATTERY_VOLTAGE_EN

    0 = Don't do ADC conversion of battery voltage, will read 0 value in reply BATTERY_ADC

    1 = Do ADC conversion of battery voltage, results in BATTERY_ADC. Vbatt = 3*BATTERY_ADC/1280
  - ADC_GPIO_EN

    0 = Don't do ADC conversion on GPIO, will read 0 value in reply

    1 = Do ADC conversion of GPIO, results in GPIO_ADC. Vgpio = 3*GPIO_ADC/1280
  - ADC_GPIO_PIN[1:0] - Select GPIOx pin. The pin must be set as input

    0 = Measure votage of GPIO0

    1 = Measure votage of GPIO1

    2 = Measure votage of GPIO2

    3 = Measure votage of GPIO3
- Response:
  - GPIO_ADC[15:0] - ADC value of voltage on GPIO
  - BATTERY_ADC[15:0] - ADC value of battery voltage
  - TEMP_ADC[15:0] - ADC value of temperature sensor voltage of the chip in degrees kelvin
  - TEMP_SLOPE[7:0] - Slope in the formula of Vtempadc -- Temperature
  - TEMP_INTERCEPT[7:0] - Intercept in the fromula of Vtempadc -- Temperature

3.2.8. FIFO_INFO

- Summary: Provides access to transmit and receive fifo counts and reset.
- Purpose:

  This command is normally used for error recovery, fifo hardware does not need to be reset prior to use.

- Command Stream:

FIFO_INFO Command	7	6	5	4	3	2	1	0
CMD	0x15							
FIFO	000000						RX	TX

- Reply Stream

FIFO_INFO Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
RX_FIFO_COUNT	RX_FIFO_COUNT[7:0]							
TX_FIFO_SPACE	TX_FIFO_SPACE[7:0]							

- Parameters:
  - RX

    1 = Resets receive data fifo.
  - TX

    1 = Resets transmit data fifo.
- Response:
  - RX_FIFO_COUNT[7:0]
  - TX_FIFO_SPACE[7:0] - Amount of space currently available in transmit fifo.

3.2.9. PACKET_INFO

- Summary: *Returns information about the last packet received and optionally overrides field length.*
- Purpose:
  - This command is used to retrieve the length field extracted from the packet when using variable length packets.
  - On 802.15.4g receive mode, the FCS byte length embedded in the FHR is included to inform the host of the number of FCS bytes in the FIFO following the MAC Payload. if FCS is 0, 2 is added to the return length; if FCS is 1, 4 is added to the return length.
  - If arguments follow, it can also be used to override packet length that was originally programmed with non-zero RX_LEN in START_RX or field length originally programmed with PKT field length properties.
- Command Stream

PACKET_INFO Command	7	6	5	4	3	2	1	0
CMD	0x16							
FIELD_NUMBER	000			FIELD_NUM[4:0]				
LEN	LEN[15:8]							
LEN	LEN[7:0]							
LEN_DIFF	LEN_DIFF[15:8]							
LEN_DIFF	LEN_DIFF[7:0]							

- Reply Stream

PACKET_INFO Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
LENGTH_15_8	LENGTH_15_8[7:0]							
LENGTH_7_0	LENGTH_7_0[7:0]							

- Parameters:
  - FIELD_NUM[4:0]

    0x00 = Nothing to override.

    0x01 = Override FIELD 1 length or RX_LEN in START_RX.

    0x02 = Override FIELD 2 length.

    0x04 = Override FIELD 3 length.

    0x08 = Override FIELD 4 length.

    0x10 = Override FIELD 5 length.
  - LEN[15:0] - Length in bytes.
  - LEN_DIFF[15:0] - Difference in bytes based on the original field length or RX_LEN. Signed.
- Response:
  - LENGTH_15_8[7:0] - Most significant byte of the extracted length
  - LENGTH_7_0[7:0] - Least significant byte of the extracted length

3.2.10. IRCAL

- Summary: Calibrate receiver image rejection.
- Purpose:
  - Automatically calibrates the receiver image rejection with no requirement for an external signal source.
- Command Stream

IRCAL Command	7	6	5	4	3	2	1	0
CMD	0x17							
SEARCHING_ STEP_SIZE	0	INITIAL_ PH_AMP	FINE_STEP_ SIZE[1:0]		COARSE_STEP_SIZE[3:0]			
SEARCHING_ RSSI_AVG	00		RSSI_FINE_ AVG[1:0]		00		RSSI_COARSE_ AVG[1:0]	
RX_CHAIN_ SETTING1	EN_HRMNIC_ GEN	IRCLKDIV	RF_SOURCE_ PWR[1:0]		CLOSE_ SHUNT_ SWITCH	PGA_GAIN[2:0]		
RX_CHAIN_ SETTING2	0000000							ADC_ HIGH_ GAIN

- Reply Stream

IRCAL Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							

- Parameter:
  - INITIAL_PH_AMP - Initial ph and amp value when start IR calibration.
    0 = Use previous calibration values as starting values (default).
    1 = Use zero for phase and amplitude values as starting values.
  - FINE_STEP_SIZE[5:4] - Step size of fine stepping. Range:0~3.
    0 = Value 0 is used to skip fine stepping.
  - COARSE_STEP_SIZE[3:0] - Coarse Step Size of course stepping. Range:0~15.
    0 = Value 0 is usedn to skip course stepping
  - RSSI_FINE_AVG[5:4] - How many measurements(2^avg) per RSSI measurement while fine stepping.
    0 = 1 measurements
    1 = 2 measurements
    2 = 4 measurements
    3 = 8 measurements
  - RSSI_COARSE_AVG[1:0] - How many measurements(2^avg) per RSSI measurement while coarse stepping.
    0 = 1 measurements
    1 = 2 measurements
    2 = 4 measurements
    3 = 8 measurements
  - EN_HRMNIC_GEN - Enable harmonic generator.
    0 = Not enable
    1 = Enable
  - IRCLKDIV - Set irclkdiv
    0 = Set to nominal gain
    1 = Harmonics at N x 30 MHz

- RF_SOURCE_PWR[5:4] - Power of internal generator(Default 3).
  - 0 = smallest
  - 1 = small
  - 2 = big
  - 3 = biggest
- CLOSE_SHUNT_SWITCH - Close shunt switch.
  - 0 = Open LNA input shunt switch
  - 1 = Close Open LNA input shunt switch switch
- PGA_GAIN[2:0] - Set PGA Gain, see PGA datasheet.
  - 0 = 6dB
  - 1 = 9dB
  - 2 = 12dB
  - 3 = others are all 6dB
  - 6 = 0dB
  - 7 = 3dB
- ADC_HIGH_GAIN - Set ADC to high gain.
  - 0 = Set to nominal gain
  - 1 = Set to high gain

■ Response:

- None

3.2.11. PROTOCOL_CFG

■ Summary: *Sets the chip up for specified protocol.*

■ Purpose:

-

■ Command Stream

PROTOCOL_CFG Command	7	6	5	4	3	2	1	0
CMD	0x18							
PROTOCOL	PROTOCOL[7:0]							

■ Reply Stream

PROTOCOL_CFG Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							

■ Parameters:

- PROTOCOL[7:0] - Selects the protocol for which to configure the chip.
  - 0 = Packet format is generic, no dynamic reprogramming of packet handler properties.
  - 1 = Packet format is IEEE802.15.4g compliance. The following properties are overridden: PKT_CRC_CONFIG, CRC_ENDIAN/BIT_ORDER in PKT_CONFG1 for TX and RX, PKT_FIELD_1_CRC_CONFIG for RX. Other applicable properties in the packet handler group still need to be programmed. Field 1 should have the length of 16 bits to contain the Packet Header with PKT_LEN_FIELD_SOURCE set to 1 for RX. PSDU field shall use Field 2 with variable length. Field 2 length should be set to the maximum allowed including the anticipated FCS length. It is anticipated that the FCS will be calculated by the host and transmitted over the air. Si446x will receive Packet Header and put FCS in the FIFO for the host to retrieve and check. Therefore, CRC shouldn't be enabled on Si446x.

■Response:

- None

3.2.12. GET_INT_STATUS

- Summary: Returns the interrupt status byte.
- Purpose:
  - Returns the current interrupt status byte.
- Command Stream

GET_INT_STATUS Command	7	6	5	4	3	2	1	0
CMD	0x20							
PH_CLR_PEND	FILTER_MATCH_PEND_CLR	FILTER_MISS_PEND_CLR	PACKET_SENT_PEND_CLR	PACKET_RX_PEND_CLR	CRC_ERROR_PEND_CLR	0	TX_FIFO_ALMOST_EMPTY_PEND_CLR	RX_FIFO_ALMOST_FULL_PEND_CLR
MODEM_CLR_PEND	00		INVALID_SYNC_PEND_CLR	RSSI_JUMP_PEND_CLR	RSSI_PEND_CLR	INVALID_PREAMBLE_PEND_CLR	PREAMBLE_DETECT_PEND_CLR	SYNC_DETECT_PEND_CLR
CHIP_CLR_PEND	CAL_PEND_CLR		FIFO_UNDERFLOW_OVERFLOW_ERROR_PEND_CLR	STATE_CHANGE_PEND_CLR	CMD_ERROR_PEND_CLR	CHIP_READY_PEND_CLR	LOW_BATT_PEND_CLR	WUT_PEND_CLR

- Reply Stream

GET_INT_STATUS Reply	7	6	5	4	3	2	1	0
CMD_COMPLETE	CTS[7:0]							
INT_PEND	XXXXX					CHIP_INT_STATUS_PEND	MODEM_INT_STATUS_PEND	PH_INT_STATUS_PEND
INT_STATUS	XXXXX					CHIP_INT_STATUS	MODEM_INT_STATUS	PH_INT_STATUS
PH_PEND	FILTER_MATCH_PEND	FILTER_MISS_PEND	PACKET_SENT_PEND	PACKET_RX_PEND	CRC_ERROR_PEND	X	TX_FIFO_ALMOST_EMPTY_PEND	RX_FIFO_ALMOST_FULL_PEND
PH_STATUS	FILTER_MATCH	FILTER_MISS	PACKET_SENT	PACKET_RX	CRC_ERROR	X	TX_FIFO_ALMOST_EMPTY	RX_FIFO_ALMOST_FULL
MODEM_PEND	XX		INVALID_SYNC_PEND	RSSI_JUMP_PEND	RSSI_PEND	INVALID_PREAMBLE_PEND	PREAMBLE_DETECT_PEND	SYNC_DETECT_PEND
MODEM_STATUS	XX		INVALID_SYNC	RSSI_JUMP	RSSI	INVALID_PREAMBLE	PREAMBLE_DETECT	SYNC_DETECT

CHIP_PEND	CAL_PEND	FIFO_ UNDERFLOW_ OVERFLOW_ ERROR_PEND	STATE_ CHANGE_ PEND	CMD_ ERROR_ PEND	CHIP_ READY_ PEND	LOW_BATT_ PEND	WUT_ PEND
CHIP_ STATUS	CAL	FIFO_ UNDERFLOW_ OVERFLOW_ ERROR	STATE_ CHANGE	CMD_ ERROR	CHIP_ READY	LOW_BATT	WUT

- Parameters
  - FILTER_MATCH_PEND_CLR - If clear, Clear pending FILTER_MATCH interrupt. If set, leave interrupt pending
  - FILTER_MISS_PEND_CLR - If clear, Clear pending FILTER_MISS interrupt. If set, leave interrupt pending
  - PACKET_SENT_PEND_CLR - If clear, Clear pending PACKET_SENT interrupt. If set, leave interrupt pending
  - PACKET_RX_PEND_CLR - If clear, Clear pending PACKET_RX interrupt. If set, leave interrupt pending
  - CRC_ERROR_PEND_CLR - If clear, Clear pending CRC_ERROR interrupt. If set, leave interrupt pending
  - TX_FIFO_ALMOST_EMPTY_PEND_CLR - If clear, Clear pending TX_FIFO_ALMOST_EMPTY interrupt. If set, leave interrupt pending
  - RX_FIFO_ALMOST_FULL_PEND_CLR - If clear, Clear pending RX_FIFO_ALMOST_FULL interrupt. If set, leave interrupt pending
  - INVALID_SYNC_PEND_CLR - If clear, Clear pending INVALID_SYNC interrupt. If set, leave interrupt pending
  - RSSI_JUMP_PEND_CLR - If clear, Clear pending RSSI_JUMP interrupt. If set, leave interrupt pending
  - RSSI_PEND_CLR - If clear, Clear pending RSSI interrupt. If set, leave interrupt pending
  - INVALID_PREAMBLE_PEND_CLR - If clear, Clear pending INVALID_PREAMBLE interrupt. If set, leave interrupt pending
  - PREAMBLE_DETECT_PEND_CLR - If clear, Clear pending PREAMBLE_DETECT interrupt. If set, leave interrupt pending
  - SYNC_DETECT_PEND_CLR - If clear, Clear pending SYNC_DETECT interrupt. If set, leave interrupt pending.
  - CAL_PEND_CLR - If clear, clear pending CAL interrupt. If set, leave interrupt pending.
  - FIFO_UNDERFLOW_OVERFLOW_ERROR_PEND_CLR - If clear, Clear pending FIFO_UNDERFLOW_OVERFLOW_ERROR interrupt. If set, leave interrupt pending.
  - STATE_CHANGE_PEND_CLR - If clear, Clear pending STATE_CHANGE interrupt. If set, leave interrupt pending.
  - CMD_ERROR_PEND_CLR - If clear, Clear pending CMD_ERROR interrupt. If set, leave interrupt pending.
  - CHIP_READY_PEND_CLR - If clear, Clear pending CHIP_READY interrupt. If set, leave interrupt pending.
  - LOW_BATT_PEND_CLR - If clear, Clear pending LOW_BATT interrupt. If set, leave interrupt pending.
  - WUT_PEND_CLR - If clear, Clear pending WUT interrupt. If set, leave interrupt pending.
- Response
  - CHIP_INT_STATUS_PEND - If set, CHIP_INT_STATUS interrupt is pending.
  - MODEM_INT_STATUS_PEND - If set, MODEM_INT_STATUS interrupt is pending.
  - PH_INT_STATUS_PEND - If set, PH_INT_STATUS interrupt is pending.
  - CHIP_INT_STATUS - If set, chip status has interrupt pending.
  - MODEM_INT_STATUS - If set, modem status has interrupt pending.
  - PH_INT_STATUS - If set, packet handler status has interrupt pending.
  - FILTER_MATCH_PEND - If set, FILTER_MATCH interrupt is pending.
  - FILTER_MISS_PEND - If set, FILTER_MISS interrupt is pending.
  - PACKET_SENT_PEND - If set, PACKET_SENT interrupt is pending.
  - PACKET_RX_PEND - If set, PACKET_RX interrupt is pending.
  - CRC_ERROR_PEND - If set, CRC_ERROR interrupt is pending.
  - TX_FIFO_ALMOST_EMPTY_PEND - If set, TX_FIFO_ALMOST_EMPTY interrupt is pending.
  - RX_FIFO_ALMOST_FULL_PEND - If set, RX_FIFO_ALMOST_FULL interrupt is pending.
  - FILTER_MATCH - If set, incoming packet matched filter.
  - FILTER_MISS - If set, incoming packet was discarded because filter did not match.
  - PACKET_SENT - If set, Packet Sent.
  - PACKET_RX - If set, Packet Received.

- CRC_ERROR - If set, CRC-32 error
- TX_FIFO_ALMOST_EMPTY - If set, TX fifo is below watermark
- RX_FIFO_ALMOST_FULL - If set, RX fifo is above watermark
- INVALID_SYNC_PEND - If set, INVALID_SYNC interrupt is pending.
- RSSI_JUMP_PEND - If set, RSSI_JUMP interrupt is pending.
- RSSI_PEND - If set, RSSI interrupt is pending.
- INVALID_PREAMBLE_PEND - If set, INVALID_PREAMBLE interrupt is pending.
- PREAMBLE_DETECT_PEND - If set, PREAMBLE_DETECT interrupt is pending.
- SYNC_DETECT_PEND - If set, SYNC_DETECT interrupt is pending.
- INVALID_SYNC - If set, invalid sync has been detected
- RSSI_JUMP - If set, RSSI jump above MODEM_RSSI_JUMP_THRESH has occured
- RSSI - If set, RSSI is above MODEM_RSSI_THRESH
- INVALID_PREAMBLE - If set, invalid preamble has been detected
- PREAMBLE_DETECT - If set, preamble has been detected
- SYNC_DETECT - If set, sync has been detected.
- CAL_PEND - If set, CAL interrupt is pending.
- FIFO_UNDERFLOW_OVERFLOW_ERROR_PEND - If set, FIFO_UNDERFLOW_OVERFLOW_ERROR interrupt is pending.
- CAL - If set, calibration timer has expired.
- STATE_CHANGE_PEND - If set, STATE_CHANGE interrupt is pending.
- CMD_ERROR_PEND - If set, CMD_ERROR interrupt is pending.
- CHIP_READY_PEND - If set, CHIP_READY interrupt is pending.
- LOW_BATT_PEND - If set, LOW_BATT interrupt is pending.
- WUT_PEND - If set, WUT interrupt is pending.
- FIFO_UNDERFLOW_OVERFLOW_ERROR - If set, fifo underflow or overflow occured
- STATE_CHANGE - If set, a state change has occured
- CMD_ERROR - If set, command error has occured
- CHIP_READY - If set, chip is ready to accept commands
- LOW_BATT - If set, low battery has been detected
- WUT - If set, wakeup timer has expired

3.2.13. GET_PH_STATUS

- Summary: *Returns the packet handler status.*
- Purpose:
  - Returns current packet handler status bytes and possibly clears pending packet handler interrupts.
- Command Stream

GET_PH_STATUS Command	7	6	5	4	3	2	1	0
CMD	0x21							

- Reply Stream

GET_PH_ STATUS Reply	7	6	5	4	3	2	1	0
CMD_ COMPLETE	CTS[7:0]							
PH_PEND	FILTER_ MATCH_ PEND	FILTER_ MISS_ PEND	PACKET_ SENT_ PEND	PACKET_ RX_ PEND	CRC_ ERROR_ PEND	X	TX_FIFO_ ALMOST_ EMPTY_PEND	RX_FIFO_ALMOST_ FULL_PEND
PH_STATUS	FILTER_ MATCH	FILTER_ MISS	PACKET_ SENT	PACKET_ RX	CRC_ ERROR	X	TX_FIFO_ ALMOST_EMPTY	RX_FIFO_ALMOST_ FULL

- Parameters:
  - None
- Response:
  - FILTER_MATCH_PEND - If set, FILTER_MATCH interrupt is pending.
  - FILTER_MISS_PEND - If set, FILTER_MISS interrupt is pending.
  - PACKET_SENT_PEND - If set, PACKET_SENT interrupt is pending.
  - PACKET_RX_PEND - If set, PACKET_RX interrupt is pending.
  - CRC_ERROR_PEND - If set, CRC_ERROR interrupt is pending.
  - TX_FIFO_ALMOST_EMPTY_PEND - If set, TX_FIFO_ALMOST_EMPTY interrupt is pending.
  - RX_FIFO_ALMOST_FULL_PEND - If set, RX_FIFO_ALMOST_FULL interrupt is pending.
  - FILTER_MATCH - If set, incoming packet matched filter.
  - FILTER_MISS - If set, incoming packet was discarded because filter did not match
  - PACKET_SENT - If set, Packet Sent
  - PACKET_RX - If set, Packet Received
  - CRC_ERROR - If set, CRC-32 error
  - TX_FIFO_ALMOST_EMPTY - If set, TX fifo is below watermark
  - RX_FIFO_ALMOST_FULL - If set, RX fifo is above watermark

3.2.14. GET_MODEM_STATUS

- Summary: *Returns the modem status byte.*
- Purpose:
  - Returns and possibly clears the current modem status byte.
- Command Stream

GET_MODEM_STATUS Command	7	6	5	4	3	2	1	0
CMD	0x22							

- Reply Stream

GET_MODEM_STATUS Reply	7	6	5	4	3	2	1	0
CMD_ COMPLETE	CTS[7:0]							
MODEM_PEND	XX		INVALID_ SYNC_ PEND	RSSI_ JUMP_ PEND	RSSI_ PEND	INVALID_ PREAMBLE_ PEND	PREAMBLE_ DETECT_PEND	SYNC_ DETECT_ PEND
MODEM_ STATUS	XX		INVALID_ SYNC	RSSI_ JUMP	RSSI	INVALID_ PREAMBLE	PREAMBLE_ DETECT	SYNC_ DETECT
CURR_RSSI	CURR_RSSI[7:0]							
LATCH_RSSI	LATCH_RSSI[7:0]							
ANT1_RSSI	ANT1_RSSI[7:0]							
ANT2_RSSI	ANT2_RSSI[7:0]							
AFC_FREQ_ OFFSET	AFC_FREQ_OFFSET[15:8]							
AFC_FREQ_ OFFSET	AFC_FREQ_OFFSET[7:0]							

- Parameters:
  - None
- Response:
  - INVALID_SYNC_PEND - If set, INVALID_SYNC interrupt is pending.
  - RSSI_JUMP_PEND - If set, RSSI_JUMP interrupt is pending.
  - RSSI_PEND - If set, RSSI interrupt is pending.
  - INVALID_PREAMBLE_PEND - If set, INVALID_PREAMBLE interrupt is pending.
  - PREAMBLE_DETECT_PEND - If set, PREAMBLE_DETECT interrupt is pending.
  - SYNC_DETECT_PEND - If set, SYNC_DETECT interrupt is pending.
  - INVALID_SYNC - If set, invalid sync has been detected
  - RSSI_JUMP - If set, RSSI jump above MODEM_RSSI_JUMP_THRESH has occured
  - RSSI - If set, RSSI is above MODEM_RSSI_THRESH
  - INVALID_PREAMBLE - If set, invalid preamble has been detected
  - PREAMBLE_DETECT - If set, preamble has been detected
  - SYNC_DETECT - If set, sync has been detected
  - CURR_RSSI[7:0] - Current RSSI reading from the modem.
  - LATCH_RSSI[7:0] - Latched RSSI reading from the modem as configured by MODEM_RSSI_CONTROL. Reset to 0 at the start of every RX.
  - ANT1_RSSI[7:0] - RSSI of ANT1 while antenna diversity. Latched during preamble evaluation and avaliable for reading after sync detection.
  - ANT2_RSSI[7:0] - RSSI of ANT2 while antenna diversity. Latched during preamble evaluation and avaliable for reading after sync detection.
  - AFC_FREQ_OFFSET[15:0] - The AFC value that is generated by the AFC loop during receive mode.

3.2.15. GET_CHIP_STATUS

- Summary: *Returns the chip status.*
- Purpose:
  - Returns current chip status bytes and possibly clears pending chip status interrupts.
- Command Stream

GET_CHIP_STATUS Command	7	6	5	4	3	2	1	0
CMD	0x23							

- Reply Stream

GET_CHIP_ STATUS Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
CHIP_PEND	CAL_ PEND		FIFO_UNDERFLOW_ OVERFLOW_ERROR_ PEND	STATE_ CHANGE_ PEND	CMD_ ERROR_ PEND	CHIP_ READY_ PEND	LOW_ BATT_ PEND	WUT_PEND
CHIP_ STATUS	CAL		FIFO_UNDERFLOW_ OVERFLOW_ERROR	STATE_ CHANGE	CMD_ ERROR	CHIP_ READY	LOW_ BATT	WUT
CMD_ERR_ STATUS	CMD_ERR_STATUS[7:0]							

- Parameters:
  - None
- Response:
  - FIFO_UNDERFLOW_OVERFLOW_ERROR_PEND - If set, FIFO_UNDERFLOW_OVERFLOW_ERROR interrupt is pending.
  - CAL_PEND - If set, CAL interrupt is pending.
  - STATE_CHANGE_PEND - If set, STATE_CHANGE interrupt is pending.
  - CMD_ERROR_PEND - If set, CMD_ERROR interrupt is pending.
  - CHIP_READY_PEND - If set, CHIP_READY interrupt is pending.
  - LOW_BATT_PEND - If set, LOW_BATT interrupt is pending.
  - WUT_PEND - If set, WUT interrupt is pending.
  - FIFO_UNDERFLOW_OVERFLOW_ERROR - If set, fifo underflow or overflow occured.
  - CAL - If set, calibration timer has expired.
  - STATE_CHANGE - If set, a state change has occured
  - CMD_ERROR - If set, command error has occured
  - CHIP_READY - If set, chip is ready to accept commands
  - LOW_BATT - If set, low battery has been detected
  - WUT - If set, wakeup timer has expired
  - CMD_ERR_STATUS[7:0] - Last command error cause. Only valid if CMD_ERROR status bit is set.
  - 0x00 = No error.
  - 0x10 = Bad command issued.
  - 0x11 = Argment(s) in issued command were invalid.
  - 0x12 = Command was issued before previous command was completed.
  - 0x20 =
  - 0x30 =
  - 0x31 =
  - 0x40 = Bad Property ID was provided.

3.2.16. START_TX

- Summary: *Switches to TX state and starts packet transmission.*
- Purpose:
  - Switches to TX state when condition is met. Command arguments are retained though sleep state, so they only need to be written when they change. CTS will not return high until in TX state.
- Command Stream

START_TX Command	7	6	5	4	3	2	1	0
CMD	0x31							
CHANNEL	CHANNEL[7:0]							
CONDITION	TXCOMPLETE_STATE[3:0]				0	RETRANSMIT	START[1:0]	
TX_LEN	TX_LEN[15:8]							
TX_LEN	TX_LEN[7:0]							

- Reply Stream

START_TX Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							

- Parameters:
  - CHANNEL[7:0] - Channel number to transmit the packet on. Frequency is determined using integer, fractional, and step size properties in the FREQ_CONTROL property group. This value will be overwritten with START_RX:CHANNEL
  - TXCOMPLETE_STATE[7:4] - State to go to when current packet transmission completes. If this parameter is not sent with the command, the last value specified is used. Defaults to 3.

    0 = No change.

    1 = Sleep state.

    2 = Spi Active state.

    3 = Ready state.

    4 = Another enumeration for Ready state.

    5 = Tune state for TX.

    6 = Tune state for RX.

    7 = TX state.

    8 = RX state.
  - RETRANSMIT

    0 = Send data that has been written to fifo. If fifo is empty a fifo underflow interrupt will occur.

    1 = Send last packet again. If this option is used, ensure that no new data is written to the fifo.
  - START[1:0]

    0 = Start TX immediately.

    1 = Start TX when wake up timer expires.
  - TX_LEN[15:0] - If this field is nonzero, the packet will be transmitted using only field 1. If this field is zero, the configuration of the packet handler fields is used. If RETRANSMIT is set, this field is ignored.
- Response
  - None

3.2.17. START_RX

- Summary: *Switches to RX state.*
- Purpose:
  - Switches to RX state when condition is met and switch to specified state when RX packet completes. CTS will not return until in RX mode
- Command Stream

START_RX Command	7	6	5	4	3	2	1	0
CMD	0x32							
CHANNEL	CHANNEL[7:0]							
CONDITION	0000000							START
RX_LEN	RX_LEN[15:8]							
RX_LEN	RX_LEN[7:0]							
NEXT_STATE1	0000				RXTIMEOUT_STATE[3:0]			
NEXT_STATE2	0000				RXVALID_STATE[3:0]			
NEXT_STATE3	0000				RXINVALID_STATE[3:0]			

- Reply Stream

START_RX Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							

- Parameters
  - CHANNEL[7:0] - Channel number to transmit the packet on. Frequency is determined using integer, fractional, and step size properties in the FREQ_CONTROL property group. This value will be overwritten with START_TX:CHANNEL
  - START

    0 = Start RX immediately.

    1 = Start RX when wake up timer expires.
  - RX_LEN[15:0] - If this field is nonzero, the packet will be received using only field 1 with no packet handler features (eg. crc, whitening). If this field is zero, the configuration of the packet handler fields is used.
  - RXTIMEOUT_STATE[3:0] - If preamble detection times out, RX will transition to RXTIMEOUT_STATE. Defaults to 0. See PREAMBLE_CONFIG_STD_2 for details on configuring preamble timeout. Defaults to 0.

    0 = No change.

    1 = Sleep state.

    2 = Spi Active state.

    3 = Ready state.

    4 = Another enumeration for Ready state.

    5 = Tune state for TX.

    6 = Tune state for RX.

    7 = TX state.

    8 = RX state.

- RXVALID_STATE[3:0] - RXVALID_STATE[3:0] - RX transitions to RXVALID_STATE if CRC check passes when CRC is enabled via PKT group properties. If CRC is not enabled, upon receiving packet received interrupt, RX transitions to RXVALID_STATE. Defaults to 3.

  0 = No change

  1 = Sleep state.

  2 = Spi Active state.

  3 = Ready state.

  4 = Another enumeration for Ready state.

  5 = Tune state for TX.

  6 = Tune state for RX.

  7 = TX state.

  8 = RX state.
- RXINVALID_STATE[3:0] - If CRC checking is enabled, in case of CRC error, RX will transition to RXINVALID_STATE. Defaults to 3.

  0 = No change

  1 = Sleep state.

  2 = Spi Active state.

  3 = Ready state.

  4 = Another enumeration for Ready state.

  5 = Tune state for TX.

  6 = Tune state for RX.

  7 = TX state.

  8 = RX state.
- Response

  None

3.2.18. REQUEST_DEVICE_STATE

- Summary: *Request current device state.*
- Purpose:
  - Requests the current state of the device and lists pending TX and RX requests.
- Command Stream

REQUEST_DEVICE_STATE Command	7	6	5	4	3	2	1	0
CMD	0x33							

- Reply Stream

REQUEST_DEVICE_STATE Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
CURR_STATE	XXXX				MAIN_STATE[3:0]			
CURRENT_CHANNEL	CURRENT_CHANNEL[7:0]							

- Parameters:
  - None
- Response:
  - MAIN_STATE[3:0] - Current State.
    0 = No change
    1 = Sleep state.
    2 = Spi Active state.
    3 = Ready state.
    4 = Another enumeration for Ready state.
    5 = Tune state for TX.
    6 = Tune state for RX.
    7 = TX state.
    8 = RX state.
  - CURRENT_CHANNEL[7:0]

3.2.19. CHANGE_STATE

- Summary: *Update state machine entries.*
- Purpose:
  - This command is used to manually switch to a specified state, or to cancel pending state transitions.
- Command Stream

CHANGE_STATE Command	7	6	5	4	3	2	1	0
CMD	0x34							

- Reply Stream

CHANGE_STATE Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							

- Parameters:
  - NEW_STATE[3:0] - State to go to immediately.

    0 = No change
    1 = Sleep state.
    2 = Spi Active state.
    3 = Ready state.
    4 = Another enumeration for Ready state.
    5 = Tune state for TX.
    6 = Tune state for RX.
    7 = TX state.
    8 = RX state.
- Response:
  - None

3.2.20. READ_CMD_BUFF

- Summary: Used to read CTS and the command response.
- Purpose:
  - This command does not cause CTS to go low and can be sent and the reply read while CTS is low. This command is used to read values from the hardware. The command ID should be clocked in on SDI then the reply should be clocked out on SDO without deasserting NSEL.
- Command Stream

READ_CMD_BUFF Command	7	6	5	4	3	2	1	0
CMD	0x44							

- Reply Stream

READ_CMD_BUFF Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							
CMD_BUFF0	CMD_BUFF0[7:0]							
CMD_BUFF1	CMD_BUFF1[7:0]							
CMD_BUFF2	CMD_BUFF2[7:0]							
CMD_BUFF3	CMD_BUFF3[7:0]							
CMD_BUFF4	CMD_BUFF4[7:0]							
CMD_BUFF5	CMD_BUFF5[7:0]							
CMD_BUFF6	CMD_BUFF6[7:0]							
CMD_BUFF7	CMD_BUFF7[7:0]							
CMD_BUFF8	CMD_BUFF8[7:0]							
CMD_BUFF9	CMD_BUFF9[7:0]							
CMD_BUFF10	CMD_BUFF10[7:0]							
CMD_BUFF11	CMD_BUFF11[7:0]							
CMD_BUFF12	CMD_BUFF12[7:0]							
CMD_BUFF13	CMD_BUFF13[7:0]							
CMD_BUFF14	CMD_BUFF14[7:0]							
CMD_BUFF15	CMD_BUFF15[7:0]							

- Parameters:
  - None
- Response:
  - CMD_BUFF0[7:0] - Byte 0 of the command response buffer. Only valid if CTS is 0xFF
  - CMD_BUFF1[7:0] - Byte 1 of the command response buffer. Only valid if CTS is 0xFF
  - CMD_BUFF2[7:0] - Byte 2 of the command response buffer. Only valid if CTS is 0xFF
  - CMD_BUFF3[7:0] - Byte 3 of the command response buffer. Only valid if CTS is 0xFF
  - CMD_BUFF4[7:0] - Byte 4 of the command response buffer. Only valid if CTS is 0xFF
  - CMD_BUFF5[7:0] - Byte 5 of the command response buffer. Only valid if CTS is 0xFF
  - CMD_BUFF6[7:0] - Byte 6 of the command response buffer. Only valid if CTS is 0xFF

- CMD_BUFF7[7:0] - Byte 7 of the command response buffer. Only valid if CTS is 0xFF
- CMD_BUFF8[7:0] - Byte 8 of the command response buffer. Only valid if CTS is 0xFF
- CMD_BUFF9[7:0] - Byte 9 of the command response buffer. Only valid if CTS is 0xFF
- CMD_BUFF10[7:0] - Byte 10 of the command response buffer. Only valid if CTS is 0xFF
- CMD_BUFF11[7:0] - Byte 11 of the command response buffer. Only valid if CTS is 0xFF
- CMD_BUFF12[7:0] - Byte 12 of the command response buffer. Only valid if CTS is 0xFF
- CMD_BUFF13[7:0] - Byte 13 of the command response buffer. Only valid if CTS is 0xFF
- CMD_BUFF14[7:0] - Byte 14 of the command response buffer. Only valid if CTS is 0xFF
- CMD_BUFF15[7:0] - Byte 15 of the command response buffer. Only valid if CTS is 0xFF

3.2.21. FRR_A_READ

- Summary: Reads the fast response registers starting with A.
- Purpose:
  - This command does not cause CTS to go low, and can be sent and the reply read while CTS is low. This command is used to read values from the hardware. The command ID should be clocked in on SDI then the reply should be clocked out on SDO without deasserting NSEL..
- Command Stream

FRR_A_READ Command	7	6	5	4	3	2	1	0
CMD	0x50							

- Reply Stream

FRR_A_READ Reply	7	6	5	4	3	2	1	0
FRR_A_VALUE	FRR_A_VALUE[7:0]							
FRR_B_VALUE	FRR_B_VALUE[7:0]							
FRR_C_VALUE	FRR_C_VALUE[7:0]							
FRR_D_VALUE	FRR_D_VALUE[7:0]							

- Parameters:
  - None
- Response:
  - FRR_A_VALUE[7:0] - Value in Fast Response Register A.
  - FRR_B_VALUE[7:0] - Value in Fast Response Register B.
  - FRR_C_VALUE[7:0] - Value in Fast Response Register C.
  - FRR_D_VALUE[7:0] - Value in Fast Response Register D.

3.2.22. FRR_B_READ

- Summary: Reads the fast response registers starting with B.
- Purpose:
  - This command does not cause CTS to go low, and can be sent and the reply read while CTS is low. This command is used to read values from the hardware. The command ID should be clocked in on SDI then the reply should be clocked out on SDO without deasserting NSEL..
- Command Stream

FRR_B_READ Command	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CMD	0x51							

- Reply Stream

FRR_B_READ Reply	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FRR_B_VALUE	FRR_B_VALUE[7:0]							
FRR_C_VALUE	FRR_C_VALUE[7:0]							
FRR_D_VALUE	FRR_D_VALUE[7:0]							
FRR_A_VALUE	FRR_A_VALUE[7:0]							

- Parameters:
  - None
- Response:
  - FRR_B_VALUE[7:0] - Value in Fast Response Register B.
  - FRR_C_VALUE[7:0] - Value in Fast Response Register C.
  - FRR_D_VALUE[7:0] - Value in Fast Response Register D.
  - FRR_A_VALUE[7:0] - Value in Fast Response Register A.

3.2.23. FRR_C_READ

- Summary: Reads the fast response registers starting with C.
- Purpose:
  - This command does not cause CTS to go low, and can be sent and the reply read while CTS is low. This command is used to read values from the hardware. The command ID should be clocked in on SDI then the reply should be clocked out on SDO without deasserting NSEL..
- Command Stream

FRR_C_READ Command	7	6	5	4	3	2	1	0
CMD	0x53							

- Reply Stream

FRR_C_READ Reply	7	6	5	4	3	2	1	0
FRR_C_VALUE	FRR_C_VALUE[7:0]							
FRR_D_VALUE	FRR_D_VALUE[7:0]							
FRR_A_VALUE	FRR_A_VALUE[7:0]							
FRR_B_VALUE	FRR_B_VALUE[7:0]							

- Parameters:
  - None
- Response:
  - FRR_C_VALUE[7:0] - Value in Fast Response Register C.
  - FRR_D_VALUE[7:0] - Value in Fast Response Register D.
  - FRR_A_VALUE[7:0] - Value in Fast Response Register A.
  - FRR_B_VALUE[7:0] - Value in Fast Response Register B.

3.2.24. FRR_D_READ

- Summary: Reads the fast response registers starting with D.
- Purpose:
  - This command does not cause CTS to go low, and can be sent and the reply read while CTS is low. This command is used to read values from the hardware. The command ID should be clocked in on SDI then the reply should be clocked out on SDO without deasserting NSEL..
- Command Stream

FRR_D_READ Command	7	6	5	4	3	2	1	0
CMD	0x57							

- Reply Stream

FRR_D_READ Reply	7	6	5	4	3	2	1	0
FRR_D_VALUE	FRR_D_VALUE[7:0]							
FRR_A_VALUE	FRR_A_VALUE[7:0]							
FRR_B_VALUE	FRR_B_VALUE[7:0]							
FRR_C_VALUE	FRR_C_VALUE[7:0]							

- Parameters:
  - None
- Response:
  - FRR_D_VALUE[7:0] - Value in Fast Response Register D.
  - FRR_A_VALUE[7:0] - Value in Fast Response Register A.
  - FRR_B_VALUE[7:0] - Value in Fast Response Register B.
  - FRR_C_VALUE[7:0] - Value in Fast Response Register C.

3.2.25. WRITE_TX_FIFO

- Summary: Writes the TX FIFO.
- Purpose:
  - This command does not cause CTS to go low, and can be sent and the reply read while CTS is low. This command has no response to be read. If you write more data than the TX FIFO can hold, it will trigger a FIFO Overflow interrupt.
- Command Stream

WRITE_TX_FIFO Command	7	6	5	4	3	2	1	0
CMD	0x66							
FIRST_BYTE	FIRST_BYTE[7:0]							

- Reply Stream

WRITE_TX_FIFO Reply	7	6	5	4	3	2	1	0

- Parameters:
  - FIRST_BYTE[7:0] - First byte to write to the TX FIFO.
- Response:
  - None.

3.2.26. READ_RX_FIFO

- Summary: Reads the RX FIFO.
- Purpose:
  - This command does not cause CTS to go low, and can be sent and the reply read while CTS is low. This command is used to read values from the hardware. The command ID should be clocked in on SDI; then, the reply should be clocked out on SDO without deasserting NSEL. If you read more data than the RX FIFO contains, it will trigger a FIFO Underflow interrupt.
- Command Stream

READ_RX_FIFO Command	7	6	5	4	3	2	1	0
CMD	0x77							

- Reply Stream

READ_RX_FIFO Reply	7	6	5	4	3	2	1	0
Read FIFO contents	RX FIFO BYTES							

- Parameters:
  - None.
- Response:
  - None.

3.2.27. RX_HOP

- Summary: Fast RX hopping
- Purpose:RX Hop is designed to provide the fastest RX to RX switching time for frequency hopping systems. The RX to RX time is 75usec using this command. The VCO_CNT must be calculated offline and stored in the host.
  -
- Command Stream

RX_HOP Command	7	6	5	4	3	2	1	0
CMD	0x36							
INTE	INTE[7:0]							
FRAC2	FRAC2[7:0]							
FRAC1	FRAC1[7:0]							
FRAC0	FRAC0[7:0]							
VCO_CNT1	VCO_CNT1[7:0]							
VCO_CNT0	VCO_CNT0[7:0]							

- Reply Stream

RX_HOP Reply	7	6	5	4	3	2	1	0
CTS	CTS[7:0]							

- Parameters:
  - INTE[7:0] - INTE register value. Range 0–127.
  - FRAC2[7:0] - FRAC2 register value. Range 0–15.
  - FRAC1[7:0] - FRAC1 register value. Range 0–255.
  - FRAC0[7:0] - FRAC0 register value. Range 0–255.
  - VCO_CNT0[7:0] - VCO_CNT0 register value. Range 0–255.
  - VCO_CNT1[7:0] - VCO_CNT1 register value. Range 0–255.
- Response:
  - None

3.3. Properties

3.4. Common Properties

3.4.1. INT_CTL_ENABLE

- Summary: *Interrupt enable property*
- Purpose:
  - Enables top-level interrupt sources
- Property: 0x0100
- Default: 0x04
- Fields:
  - CHIP_INT_STATUS_EN - default:1 If set, Enables CHIP_INT_STATUS interrupt.
  - MODEM_INT_STATUS_EN - default:0 If set, Enables MODEM_INT_STATUS interrupt.
  - PH_INT_STATUS_EN - default:0 If set, Enables PH_INT_STATUS interrupt.
- Register View

INT_CTL_ENABLE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x00					CHIP_INT_STATUS_EN	MODEM_INT_STATUS_EN	PH_INT_STATUS_EN
0x00					1	0	0

3.4.2. INT_CTL_PH_ENABLE

- Summary: Packet handler interrupt enable property
- Purpose:
  - Enables packet handler interrupt sources
- Property: 0x0101
- Default: 0x00
- Fields:
  - FILTER_MATCH_EN - default:0 If set, Enables FILTER_MATCH interrupt.
  - FILTER_MISS_EN - default:0 If set, Enables FILTER_MISS interrupt.
  - PACKET_SENT_EN - default:0 If set, Enables PACKET_SENT interrupt.
  - PACKET_RX_EN - default:0 If set, Enables PACKET_RX interrupt.
  - CRC_ERROR_EN - default:0 If set, Enables CRC_ERROR interrupt.
  - TX_FIFO_ALMOST_EMPTY_EN - default:0 If set, Enables TX_FIFO_ALMOST_EMPTY interrupt.
  - RX_FIFO_ALMOST_FULL_EN - default:0 If set, Enables RX_FIFO_ALMOST_FULL interrupt.
- Register View

INT_CTL_PH_ENABLE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FILTER_MATCH_EN	FILTER_MISS_EN	PACKET_SENT_EN	PACKET_RX_EN	CRC_ERROR_EN	0	TX_FIFO_ALMOST_EMPTY_EN	RX_FIFO_ALMOST_FULL_EN
0	0	0	0	0	0	0	0

3.4.3. INT_CTL_MODEM_ENABLE

- Summary: Modem interrupt enable property
- Purpose:
  - Enables modem interrupt sources
- Property: 0x0102
- Default: 0x00
- Fields:
  - INVALID_SYNC_EN - default:0 If set, Enables INVALID_SYNC interrupt.
  - RSSI_JUMP_EN - default:0 If set, Enables RSSI_JUMP interrupt.
  - RSSI_EN - default:0 If set, Enables RSSI interrupt.
  - INVALID_PREAMBLE_EN - default:0 If set, Enables INVALID_PREAMBLE interrupt.
  - PREAMBLE_DETECT_EN - default:0 If set, Enables PREAMBLE_DETECT interrupt.
  - SYNC_DETECT_EN - default:0 If set, Enables SYNC_DETECT interrupt.
- Register View

INT_CTL_MODEM_ENABLE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0		INVALID_ SYNC_EN	RSSI_ JUMP_EN	RSSI_ EN	INVALID_ PREAMBLE_EN	PREAMBLE_ DETECT_EN	SYNC_ DETECT_EN
0x0		0	0	0	0	0	0

3.4.4. INT_CTL_CHIP_ENABLE

- Summary: Chip interrupt enable property
- Purpose:
  - Enables chip interrupt sources
- Property: 0x0103
- Default: 0x04
- Fields:
  - FIFO_UNDERFLOW_OVERFLOW_ERROR_EN - default:0 If set, Enables FIFO_UNDERFLOW_OVERFLOW_ERROR interrupt.
  - STATE_CHANGE_EN - default:0 If set, Enables STATE_CHANGE interrupt.
  - CMD_ERROR_EN - default:0 If set, Enables CMD_ERROR interrupt.
  - CHIP_READY_EN - default:1 If set, Enables CHIP_READY interrupt.
  - LOW_BATT_EN - default:0 If set, Enables LOW_BATT interrupt.
  - WUT_EN - default:0 If set, Enables WUT interrupt.
- Register View

INT_CTL_CHIP_ENABLE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0		FIFO_UNDERFLOW_ OVERFLOW_ERROR_EN	STATE_ CHANGE_EN	CMD_ ERROR_EN	CHIP_ READY_EN	LOW_ BATT_EN	WUT_EN
0x0		0	0	0	1	0	0

3.4.5. FRR_CTL_A_MODE

- Summary: Fast Response Register A Configuration
- Purpose:
  - Set the data that is present in fast response register A.
- Property: 0x0200
- Default: 0x01
- Fields:
  - FRR_A_MODE[7:0] - default:0x01

    0 = Disabled. Will always read back 0
    1 = Global status
    2 = Global interrupt pending
    3 = Packet Handler status
    4 = Packet Handler interrupt pending
    5 = Modem status
    6 = Modem interrupt pending
    7 = Chip status
    8 = Chip status interrupt pending
    9 = Current state
    10 = Latched RSSI value as defined in MODEM_RSSI_CONTROL:LATCH

- Register View

FRR_CTL_A_MODE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FRR_A_MODE[7:0]							
0x01							

3.4.6. FRR_CTL_B_MODE

- Summary: Fast Response Register B Configuration
- Purpose:
  - Set the data that is present in fast response register B.
- Property: 0x0201
- Default: 0x02
- Fields:
  - FRR_B_MODE[7:0] - default:0x02

    0 = Disabled. Will always read back 0
    1 = Global status
    2 = Global interrupt pending
    3 = Packet Handler status
    4 = Packet Handler interrupt pending
    5 = Modem status
    6 = Modem interrupt pending
    7 = Chip status
    8 = Chip status interrupt pending
    9 = Current state
    10 = Latched RSSI value as defined in MODEM_RSSI_CONTROL:LATCH

- Register View

FRR_CTL_B_MODE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FRR_B_MODE[7:0]							
0x02							

3.4.7. FRR_CTL_C_MODE

- Summary: Fast Response Register C Configuration
- Purpose:
  - Set the data that is present in fast response register C.
- Property: 0x0202
- Default: 0x09
- Fields:
  - FRR_C_MODE[7:0] - default:0x09

    0 = Disabled. Will always read back 0
    1 = Global status
    2 = Global interrupt pending
    3 = Packet Handler status
    4 = Packet Handler interrupt pending
    5 = Modem status
    6 = Modem interrupt pending
    7 = Chip status
    8 = Chip status interrupt pending
    9 = Current state
    10 = Latched RSSI value as defined in MODEM_RSSI_CONTROL:LATCH

- Register View

FRR_CTL_C_MODE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FRR_C_MODE[7:0]							
0x09							

3.4.8. FRR_CTL_D_MODE

- Summary: Fast Response Register D Configuration
- Purpose:
  - Set the data that is present in fast response register D.
- Property: 0x0203
- Default: 0x00
- Fields:
  - FRR_A_MODE[7:0] - default:0x00

    0 = Disabled. Will always read back 0
    1 = Global status
    2 = Global interrupt pending
    3 = Packet Handler status
    4 = Packet Handler interrupt pending
    5 = Modem status
    6 = Modem interrupt pending
    7 = Chip status
    8 = Chip status interrupt pending
    9 = Current state
    10 = Latched RSSI value as defined in MODEM_RSSI_CONTROL:LATCH

- Register View

<table>
<tr><th colspan="8">FRR_CTL_D_MODE</th></tr>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">FRR_A_MODE[7:0]</td></tr>
<tr><td colspan="8">0x00</td></tr>
</table>

3.4.9. SYNC_BITS_31_24

- Summary: Byte 3 of sync word
- Purpose:
  - Sync bytes are always sent bit 0 first.
- Property: 0x1101
- Default: 0x2D
- Fields:
  - BITS_31_24[7:0] - default:0x2D Sync bytes are always sent bit 0 first.
    Range: 0-0xff
- Register View

SYNC_BITS_31_24							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BITS_31_24[7:0]							
0x2D							

3.4.10. SYNC_BITS_23_16

- Summary: Byte 2 of sync word
- Purpose:
  - Sync bytes are always sent bit 0 first.
- Property: 0x1102
- Default: 0xD4
- Fields:
  - BITS_23_16[7:0] - default:0xD4 Sync bytes are always sent bit 0 first.
    Range: 0-0xff
- Register View

SYNC_BITS_23_16							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BITS_23_16[7:0]							
0xD4							

3.4.11. FREQ_CONTROL_INTE

- Summary: Frac-N PLL integer number.
- Purpose:
  - Fractional-N PLL integer number defined by the modem calculator. See datasheet for frequency equation for manual calculation.
- Property: 0x4000
- Default: 0x3C
- Fields:
  - inte[6:0] - default:0x3C
    Range: 0–127
- Register View

FREQ_CONTROL_INTE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	INTE[6:0]						
0	0x3C						

3.4.12. FREQ_CONTROL_FRAC_2

- Summary: Byte 2 of Frac-N PLL fraction number.
- Purpose:
  - Fractional-N PLL fraction number defined by the modem calculator. See data sheet for frequency equation for manual calculation.
- Property: 0x4001
- Default: 0x08
- Fields:
  - frac_2[2:0] - default:0x0
    Range: 0–7
- Register View

FREQ_CONTROL_FRAC_2							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x01					frac_2[2:0]		
0x01					0x0		

3.4.13. FREQ_CONTROL_FRAC_1

- Summary: Byte 1 of Frac-N PLL fraction number.
- Purpose:
  - Fractional-N PLL fraction number defined by the modem calculator. See datasheet for frequency equation for manual calculation.
- Property: 0x4002
- Default: 0x00
- Fields:
  - frac_1[7:0] - default:0x00
    Range: 0–255
- Register View

FREQ_CONTROL_FRAC_1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FRAC_1[7:0]							
0x00							

3.4.14. FREQ_CONTROL_FRAC_0

- Summary: Byte 0 of Frac-N PLL fraction number.
- Purpose:
  - Fractional-N PLL fraction number defined by the modem calculator. See datasheet for frequency equation for manual calculation.
- Property: 0x4003
- Default: 0x00
- Fields:
  - frac_0[7:0] - default:0x00
    Range: 0-255
- Register View

FREQ_CONTROL_FRAC_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FRAC_0[7:0]							
0x00							

3.4.15. FREQ_CONTROL_CHANNEL_STEP_SIZE_1

- Summary: Byte 1 of channel step size.
- Purpose:
  - Channel frequency step size used when using EZ frequency programming. EZ frequency programming is defined by base frequency (inte + frac) + channel number x step size.
- Property: 0x4004
- Default: 0x00
- Fields:
  - channel_step_size_1[7:0] - default:0x00
    Range: 0-255
- Register View

FREQ_CONTROL_CHANNEL_STEP_SIZE_1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CHANNEL_STEP_SIZE_1[7:0]							
0x00							

3.4.16. FREQ_CONTROL_CHANNEL_STEP_SIZE_0

- Summary: Byte 0 of channel step size.
- Purpose:
  - Channel frequency step size used when using EZ frequency programming. EZ frequency programming is defined by base frequency (inte + frac) + channel number x step size.
- Property: 0x4005
- Default: 0x00
- Fields:
  - channel_step_size_0[7:0] - default:0x00
    Range: 0–255
- Register View

FREQ_CONTROL_CHANNEL_STEP_SIZE_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CHANNEL_STEP_SIZE_0[7:0]							
0x00							

3.4.17. GLOBAL_XO_TUNE

- Summary: Configure crystal oscillator frequency tuning bank
- Purpose:
  - Crystal oscillator frequency tuning value. 0x00 is maximum frequency value and 0x7F is lowest frequency value. Each LSB code corresponds to a 70fF capacitance change. The total adjustment range assuming a 30MHz XTAL is +/-100ppm.
- Property: 0x0000
- Default: 0x40
- Fields:
  - TUNE_VALUE[6:0] - default:0x40

    0 = Maximum frequency

    127 = Lowest frequency
- Register View

GLOBAL_XO_TUNE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	TUNE_VALUE[6:0]						
0	0x40						

3.4.18. GLOBAL_CLK_CFG

- Summary: Clock configuration options
- Purpose:
  -
- Property: 0x0001
- Default: 0
- Fields:
  - DIVIDED_CLK_EN - default:0

    0 = Divided clock output is disabled.

    1 = Divided clock output is enabled.
  - DIVIDED_CLK_SEL[2:0] - default:0x0

    0 = Clock output is system clock divided by 1.

    1 = Clock output is system clock divided by 2.

    2 = Clock output is system clock divided by 3.

    3 = Clock output is system clock divided by 7.5.

    4 = Clock output is system clock divided by 10.

    5 = Clock output is system clock divided by 15.

    6 = Clock output is system clock divided by 30.
  - CLK_32K_SEL[2:0] - default:0x0

    0 = 32 kHz clock is disabled

    1 = 32 kHz clock is driven by internal RC oscillator

    2 = 32 kHz clock is driven by External crystal
- Register View

GLOBAL_CLK_CFG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	DIVIDED_CLK_EN	DIVIDED_CLK_SEL[2:0]			CLK_32K_SEL[2:0]		
0	0	0x0			0x0		

3.4.19. GLOBAL_LOW_BATT_THRESH

- Summary: Low battery threshold
- Purpose:
  - Sets the low battery threshold
- Property: 0x0002
- Default: 0x18
- Fields:
  - THRESHOLD[4:0] - default:0x18

    Range: 0–31

    0 = 1.52 V

    31 = 3.13 V

    Threshold is a linear 5 mV/step
- Register View

GLOBAL_LOW_BATT_THRESH							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			THRESHOLD[4:0]				
0x0			0x18				

3.4.20. GLOBAL_CONFIG

- Summary: Global configuration settings
- Purpose:
  - Various settings that affect entire chip. If PROTOCOL is specified, the chip is placed into protocol aware state.
- Property: 0x0003
- Default: 0
- Fields:
  - FIFO_MODE - default:0

    0- TX and RX FIFO are independent, 64-byte size each.

    1- TX/RX FIFO are sharing with 128-byte size buffer.
  - PROTOCOL[2:0] - default:0x0

    0x0 = Packet format is generic, no dynamic reprogramming of packet handler properties.

    0x1 = Packet format is IEEE802.15.4g compliant.
  - POWER_MODE - default:0

    0 = High performance mode for RX and TX. RX current = 13 mA.

    1 = Low power mode for RX and TX. RX current = 10 mA.
- Register View

GLOBAL_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			FIFO_MODE	PROTOCOL[2:0]			POWER_MODE
0x0				0x0			0

3.4.21. GLOBAL_WUT_CONFIG

- Summary: GLOBAL WUT configuation
- Purpose:
  - Program WUT and enable events, Low Battery Detector support , Low Duty Cycle operation.
- Property: 0x0004
- Default: 0x00
- Fields:
  - WUT_LDC_EN[1:0] - default:0x0

    0 = Disable LDC operation

    1 = treated as wake-up - START_RX. - START_RX end state is used .
    Could allow hopping; need to determine how to support LDC in this case .

    2 = treated as wake-up - START_TX. - START_TX end state is used .
  - WUT_CAL_PERIOD[2:0] - default:0x0. If enabled by CAL_EN, selects how often to power up the chip and perform 32 kHz RC calibration.

    0 = If the CAL function is enabled, the chip will be powered on every 1 s.

    1 = If the CAL function is enabled, the chip will be powered on every 2 s.

    2 = If the CAL function is enabled, the chip will be powered on every 4 s.

    3 = If the CAL function is enabled, the chip will be powered on every 8 s.

    4 = If the CAL function is enabled, the chip will be powered on every 16 s.

    5 = If the CAL function is enabled, the chip will be powered on every 32 s.

    6 = If the CAL function is enabled, the chip will be powered on every 64 s.

    7 = If the CAL function is enabled, the chip will be powered on every 128 s.
  - WUT_LBD_EN - default:0

    0 = Disable low battery detect

    1 = Enable low battery detect on WUT inteval
  - WUT_EN - default:0

    0 = Disable wake up timer

    1 = Enable wake up timer
  - CAL_EN - default:0

    0 = Disable calibration timer

    1 = Enable calibration timer
- Register View

<table>
<tr><th colspan="8">GLOBAL_WUT_CONFIG</th></tr>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="2">WUT_LDC_EN[1:0]</td><td colspan="3">WUT_CAL_PERIOD[2:0]</td><td>WUT_LBD_EN</td><td>WUT_EN</td><td>CAL_EN</td></tr>
<tr><td colspan="2">0x0</td><td colspan="3">0x0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

3.4.22. GLOBAL_WUT_M_15_8

- Summary: Configure WUT_M_15_8
- Purpose:
  - Sets HW WUT_M higher byte
- Property: 0x0005
- Default: 0x00
- Fields:
  - WUT_M_15_8[7:0] - default:0x00
    Range: 0–255
- Register View

GLOBAL_WUT_M_15_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
WUT_M_15_8[7:0]							
0x00							

3.4.23. GLOBAL_WUT_M_7_0

- Summary: Configure WUT_M_7_0
- Purpose:
  - Sets HW WUT_M lower byte
- Property: 0x0006
- Default: 0x01
- Fields:
  - WUT_M_7_0[7:0] - default:0x01
    Range: 1-255
- Register View

GLOBAL_WUT_M_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
WUT_M_7_0[7:0]							
0x01							

3.4.24. GLOBAL_WUT_R

- Summary: Configure WUT_R
- Purpose:
  - Sets HW WUT_R
- Property: 0x0007
- Default: 0x00
- Fields:
  - WUT_SLEEP - default:0
    0 = Go to Ready state after WUT
    1 = Go to Sleep state after WUT
  - WUT_R[4:0] - default:0x00
    Range: 0-20
- Register View

GLOBAL_WUT_R							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0		WUT_SLEEP	WUT_R[4:0]				
0x0		0	0x00				

3.4.25. GLOBAL_WUT_LDC

- Summary: Configure WUT_LDC
- Purpose:
  - Sets firmware internal WUT_LDC
- Property: 0x0008
- Default: 0x00
- Fields
  - WUT_LDC[7:0] - default:0x00
    Range: 0–255
- Register View

GLOBAL_WUT_LDC							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
WUT_LDC[7:0]							
0x00							

3.4.26. PREAMBLE_TX_LENGTH

- Summary: Preamble length
- Purpose:
  - Byte or nibble length of preamble to send, depends on LENGTH_CONFIG field in PREAMBLE_CONFIG property.
- Property: 0x1000
- Default: 0x08
- Fields:
  - TX_LENGTH[7:0] - default:0x08 Byte or nibble length of preamble to send, depends on LENGTH_CONFIG field in PREAMBLE_CONFIG property.

    Range: 0–255
- Register View

PREAMBLE_TX_LENGTH							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
TX_LENGTH[7:0]							
0x08							

3.4.27. PREAMBLE_CONFIG_STD_1

- Summary: Standard preamble configuration
- Purpose:
  - Note: This field only applies to standard preambles.
- Property: 0x1001
- Default: 0x14
- Fields:
  - SKIP_SYNC_TIMEOUT - default:0

    0x1 = In standard packet mode, if set the system will ignore the syncword search timeout reset.
  - RX_THRESH[6:0] - default:0x14

    Number of preamble bits that must be valid to detect a valid preamble.

    Zero is a valid value in this field means that the preamble checking will be skipped.

    Range: 0–127
- Register View:

PREAMBLE_CONFIG_STD_1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SKIP_SYNC_TIMEOUT	RX_THRESH[6:0]						
0	0x14						

3.4.28. PREAMBLE_CONFIG_NSTD

- Summary: Non-standard preamble configuration
- Purpose:
  - Note: This field only applies to non-standard preambles.
- Property: 0x1002
- Default: 0x00
- Fields:
  - RX_ERRORS[2:0] - default:0x0 Number of preamble bit errors that are allowed when detecting a valid preamble.

    Range: 0–7
  - PATTERN_LENGTH[4:0] - default:0x00 This value plus 1 is the number of valid bits of PREAMBLE_PATTERN. If PREAM_TX_LENGTH is longer, this pattern will repeat.

    Range: 0–31
- Register View

PREAMBLE_CONFIG_NSTD							
7	6	5	4	3	2	1	0
RX_ERRORS[2:0]			PATTERN_LENGTH[4:0]				
0x0			0x00				

3.4.29. PREAMBLE_CONFIG_STD_2

- Summary: Standard preamble configuration
- Purpose:
  - Note: This field only applies to standard preambles.
- Property: 0x1003
- Default: 0x0F
- Fields:
  - RX_PREAMBLE_TIMEOUT_EXTEND[3:0] - default:0x0

    This is only used for a long preamble timeout, more than 15 nibbles.

    If this field is non-zero, then PREAMBLE_TIMEOUT is RX_PREAMBLE_TIMEOUT_EXTEND by 15 nibbles, up to 225 nibbles.

    Range: 0–15
  - RX_PREAMBLE_TIMEOUT[3:0] - default:0xF Number of nibbles to search for before determining that a preamble does not exist. This is usually used for hopping.

    Range: 0–15
- Register View

PREAMBLE_CONFIG_STD_2							
7	6	5	4	3	2	1	0
RX_PREAMBLE_TIMEOUT_EXTEND[3:0]				RX_PREAMBLE_TIMEOUT[3:0]			
0x0				0xF			

3.4.30. PREAMBLE_CONFIG

- Summary: Preamble configuration bits
- Purpose:
  - Misc preamble configuration bits.
- Property: 0x1004
- Default: 0x21
- Fields:
  - PREAM_FIRST_1_OR_0 - default:1

    0x0 = First bit is 0, calculated from the calculator.

    0x1 = First bit is 1, calculated from the calculator.
  - LENGTH_CONFIG - default:0

    0x0 = Preamble tx_length register is in nibbles.

    0x1 = Preamble tx_length register is in bytes.
  - MAN_CONST - default:0

    0x0 = When Manchester is enabled, if preamble pattern is 0101, the post-Manchester transmitted bits will be 10011001…. If the preamble pattern is 1010, the post-Manchester transmitted bits will be 01100110...

    0x1 = When Manchester is enabled, if preamble pattern is 0101, the pre-Manchester pattern will be 1111, the post-Manchester transmitted bits will be 01010101... If the preamble pattern is 1010, the pre-Manchester pattern will be 0000, the post-Manchester transmitted bits will be 10101010....
  - MAN_ENABLE - default:0

    0x0 = Preamble is not manchester encoded.

    0x1 = Preamble is manchester encoded.
  - STANDARD_PREAM[1:0] - default:0x1

    0x0 = Use non-standard preamble

    0x1 = Use standard preamble of 1010.

    0x2 = Use standard preamble of 0101.
- Register View

PREAMBLE_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0		PREAM_FIRST_1_OR_0	LENGTH_CONFIG	MAN_CONST	MAN_ENABLE	STANDARD_PREAM[1:0]	
0x0		1	0	0	0	0x1	

3.4.31. PREAMBLE_PATTERN_31_24

- Summary: Preamble pattern
- Purpose:
  - Preambles always sent bits 0-31 timewise.
  - Preamble pattern to be transmitted or expected to be received. Field is expressed in chips, after Manchester encoding or before Manchester decoding.
  - To use this register, PREAM_CONFIG_STANDARD_PREAM should be set to 0, use non-standard preamble.
- Property: 0x1005
- Default: 0
- Fields:
  - PATTERN_31_24[7:0] - default:0x00

    Preambles always sent bits 0–31 timewise.

    Preamble pattern to be transmitted or expected to be received. Field is expressed in chips, after Manchester encoding or before Manchester decoding.

    To use this register, PREAM_CONFIG_STANDARD_PREAM should be set to 0, use non-standard preamble.

    Range: 0–0xff
- Register View

PREAMBLE_PATTERN_31_24							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PATTERN_31_24[7:0]							
0x00							

3.4.32. PREAMBLE_PATTERN_23_16

- Summary: Preamble pattern
- Purpose
  - Preambles always sent bits 0–31 timewise.
  - Preamble pattern to be transmitted or expected to be received. Field is expressed in chips, after Manchester encoding or before Manchester decoding.
  - To use this register, PREAM_CONFIG_STANDARD_PREAM should be set to 0, use non-standard preamble.
- Property: 0x1006
- Default: 0
- Fields
  - PATTERN_23_16[7:0] - default:0x00

    Preambles always sent bits 0-31 timewise.

    Preamble pattern to be transmitted or expected to be received. Field is expressed in chips, after Manchester encoding or before Manchester decoding.

    To use this register, PREAM_CONFIG_STANDARD_PREAM should be set to 0, use non-standard preamble.

    Range: 0–0xff
- Register View

PREAMBLE_PATTERN_23_16							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PATTERN_23_16[7:0]							
0x00							

3.4.33. PREAMBLE_PATTERN_15_8

- Summary: Preamble pattern
- Purpose:
  - Preambles always sent bits 0-31 timewise.
  - Preamble pattern to be transmitted or expected to be received. Field is expressed in chips, after Manchester encoding or before Manchester decoding.
  - To use this register, PREAM_CONFIG_STANDARD_PREAM should be set to 0, use non-standard preamble.
- Property: 0x1007
- Default: 0
- Fields:
  - PATTERN_15_8[7:0] - default:0x00

    Preambles always sent bits 0-31 timewise.

    Preamble pattern to be transmitted or expected to be received. Field is expressed in chips, after Manchester encoding or before Manchester decoding.

    To use this register, PREAM_CONFIG_STANDARD_PREAM should be set to 0, use non-standard preamble.

    Range: 0–0xff
- Register View

<table>
<tr><th colspan="8">PREAMBLE_PATTERN_15_8</th></tr>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">PATTERN_15_8[7:0]</td></tr>
<tr><td colspan="8">0x00</td></tr>
</table>

3.4.34. PREAMBLE_PATTERN_7_0

- Summary: Preamble pattern
- Purpose:
  - Preambles always sent bits 0–31 timewise.
  - Preamble pattern to be transmitted or expected to be received. Field is expressed in chips, after Manchester encoding or before Manchester decoding.
  - To use this register, PREAM_CONFIG_STANDARD_PREAM should be set to 0, use non-standard preamble.
- Property: 0x1008
- Default: 0
- Fields:
  - PATTERN_7_0[7:0] - default:0x00

    Preambles always sent bits 0-31 timewise.

    Preamble pattern to be transmitted or expected to be received. Field is expressed in chips, after Manchester encoding or before Manchester decoding.

    To use this register, PREAM_CONFIG_STANDARD_PREAM should be set to 0, use non-standard preamble.

    Range: 0–0xff
- Register View

<table>
<tr><th colspan="8">PREAMBLE_PATTERN_7_0</th></tr>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">PATTERN_7_0[7:0]</td></tr>
<tr><td colspan="8">0x00</td></tr>
</table>

3.4.35. SYNC_CONFIG

- Summary: Sync configuration bits
- Purpose:
  - Misc sync word configuration bits. Least significant bit of sync word is transmitted/received first.
- Property: 0x1100
- Default: 0x01
- Fields:
  - SKIP_TX - default:0

    0x0 = Sync word is transmitted as defined by LENGTH field.

    0x1 = Sync word is not transmitted.
  - RX_ERRORS[2:0] - default:0x0 Number of sync bit errors that are allowed in the sync field during receive sync detection.
  - 4FSK - default:0

    0x0 = Sync word is not 4FSK modulated.

    0x1 = Sync word is 4FSK modulated.
  - MANCH - default:0

    0x0 = Sync word is not manchester encoded.

    0x1 = Sync word is manchester encoded.
  - LENGTH[1:0] - default:0x1

    0x0 = Sync word is 8 bits, sync byte 3 is used.

    0x1 = Sync word is 16 bits, sync bytes 2 and 3 are used.

    0x2 = Sync word is 24 bits, sync bytes 1, 2, and 3 are used.

    0x3 = Sync word is 32 bits, sync bytes 0, 1, 2, and 3 are used.
- Register View

SYNC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SKIP_TX	RX_ERRORS[2:0]			4FSK	MANCH	LENGTH[1:0]	
0	0x0			0	0	0x1	

3.4.36. SYNC_BITS_15_8

- Summary: Byte 1 of sync word
- Purpose:
  - Sync bytes are always sent bit 0 first.
- Property: 0x1103
- Default: 0x2D
- Fields:
  - BITS_15_8[7:0] - default:0x2D Sync bytes are always sent bit 0 first.

    Range: 0–0xff
- Register View

SYNC_BITS_15_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BITS_15_8[7:0]							
0x2D							

3.4.37. SYNC_BITS_7_0

- Summary: Byte 0 of sync word
- Purpose:
  - Sync bytes are always sent bit 0 first.
- Property: 0x1104
- Default: 0xD4
- Fields:
  - BITS_7_0[7:0] - default:0xD4 Sync bytes are always sent bit 0 first.
    Range: 0–0xff
- Register View

SYNC_BITS_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BITS_7_0[7:0]							
0xD4							

3.4.38. PKT_CRC_CONFIG

- Summary: Select a CRC polynomial and seed
- Purpose:
  - Pick the desired CRC polynomial and CRC seed.
- Property: 0x1200
- Default: 0
- Fields:
  - CRC_SEED - default:0
    0 = Use all 0s for the CRC Seed.
    1 = Use all 1s for the CRC Seed.
  - CRC_POLYNOMIAL[3:0] - default:0x0
    0 = No CRC.
    1 = ITU-T CRC8: X8+X2+X+1
    2 = IEC-16: X16+X14+X12+X11+X9+X8+X7+X4+X+1
    3 = Baicheva-16: X16+X15+X12+X7+X6+X4+X3+1
    4 = CRC-16 (IBM): X16+X15+X2+1
    5 = CCIT-16: X16+X12+X5+1
    6 = Koopman: X32+X30+X29+X28+X26+X20+X19+X17+X16+X15+X11+X10+X7+X6+X4+X2+X+1
    7 = IEEE 802.3: X32+X26+X23+X22+X16+X12+X11+X10+X8+X7+X5+X4+X2+X+1
    8 = Castagnoli: X32+X28+X27+X26+X25+X23+X22+X20+X19+X18+X14+X13+X11+X10+X9+X8+X6+1
- Register View

PKT_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CRC_SEED	0x0			CRC_POLYNOMIAL[3:0]			
0	0x0			0x0			

3.4.39. PKT_CONFIG1

- Summary: General packet configuration bits
- Purpose:
  - General packet configuration bits.
- Property: 0x1206
- Default: 0
- Fields:
  - PH_FIELD_SPLIT - default:0

    0 = Field level properties (property 0x120D to 0x1220) are shared between TX and RX.

    1 = Field level properties are split between TX and RX. TX: from 0x120D ~ 0x1220, RX: from 0x1221 ~ 0x1234
  - PH_RX_DISABLE - default:0

    0 = Packet handler is enabled in RX.

    1 = Packet handler is disabled in RX.
  - 4FSK_EN - default:0

    0 = The modem is not in 4FSK mode.

    1 = The modem is in 4FSK mode.
  - RX_MULTI_PKT - default:0

    0x0 = Turn off receive chain after packet received.

    0x1 = Leave receive chain enabled after packet received.
  - MANCH_POL - default:0

    0x0 = 0 is encoded/decoded to/from 01 Manchester pattern.

    0x1 = 0 is encoded/decoded to/from 10 Manchester pattern.
  - CRC_INVERT - default:0

    0x0 = Leave each CRC bit intact.

    0x1 = Invert each CRC bit before transmit. Invert received CRC before comparison. Data in fifo remains untouched.
  - CRC_ENDIAN - default:0

    0x0 = CRC low bytes are received/transmitted first.

    0x1 = CRC high bytes are received/transmitted first.
  - BIT_ORDER - default:0

    0x0 = Msb first for all fields. Bit 7 transmitted first timewise. Note: Preamble and sync word are always transmitted lsb first.

    0x1 = Lsb first for all fields. Bit 0 transmitted first timewise.
- Register View

PKT_CONFIG1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PH_FIELD_SPLIT	PH_RX_DISABLE	4FSK_EN	RX_MULTI_PKT	MANCH_POL	CRC_INVERT	CRC_ENDIAN	BIT_ORDER
0	0	0	0	0	0	0	0

3.4.40. PKT_LEN

- Summary: Provides information regarding how to use the length from the received packet.
- Purpose:
  - This property is used for variable length packet reception.
- Property: 0x1208
- Default: 0x00
- Fields:
  - ENDIAN - default:0

    0x0 = The length field is least significant byte first.

    0x1 = The length field is most significant byte first.
  - SIZE - default:0

    0x0 = The length field is one byte in length.

    0x1 = The length field is two bytes in length.
  - IN_FIFO - default:0

    0x0 = The data bytes containing the length field are not put in the fifo.

    0x1 = The data bytes containing the length field are put in the fifo.
  - DST_FIELD[2:0] - default:0x0

    Selects field number that will vary in length.

    A value of 0 in this field specifies fixed packet length mode. Field 2 to 5 can be designated as variable length field.
- Register View

PKT_LEN							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0		ENDIAN	SIZE	IN_FIFO	DST_FIELD[2:0]		
0x0		0	0	0	0x0		

3.4.41. PKT_LEN_FIELD_SOURCE

- Summary: Field number containing the embedded length field.
- Purpose:
  - This property is used in variable packet mode defining where the length field is in the packet.
  - The length field must be the last byte in a fixed length field and precede the variable length field
- Property: 0x1209
- Default: 0
- Fields:
  - SRC_FIELD[2:0] - default:0x0

    Selects field number that contains the length field.

    A value of 0 in this field is treated as 1.

    Range: 0–4
- Register View

PKT_LEN_FIELD_SOURCE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x00					SRC_FIELD[2:0]		
0x00					0x0		

3.4.42. PKT_LEN_ADJUST

- Summary: Adjust length field by this amount to derive the byte count of the variable length field.
- Purpose:
  - This property is added to the value extracted from the length field in the packet.
  - The result is used to set the length of the selected destination field that varies in length.
  - This constant is typically defined in a protocol specification or can be derived from the specification.
  - For example, if a protocol defines the first byte as the length field, the length field specifies the number of subsequent payload excluding the CRC bytes, LEN_ADJUST should be set to 0; If the length field specifies the number of subsequent payload including the CRC bytes, LEN_ADJUST should be set to the negative number of the CRC byte count in 2's complement.
  - LEN_ADJUST is a signed char.
- Property: 0x120A
- Default: 0
- Fields:
  - LEN_ADJUST[7:0] - default:0x00

    This property is used to add or subtract a constant to the value extracted from the length field in the packet.

    The result is used to set the length of the selected destination field that varies in length.

    It is assumed the length field embedded in the packet includes the length field itself. LEN_ADJUST can be set to 0xFF for 1byte length field or 0xFE for 2-byte length field if the length field is not inclusive.

    LEN_ADJUST is a signed char.

    Range: –128 to 127
- Register View

PKT_LEN_ADJUST							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LEN_ADJUST[7:0]							
0x00							

3.4.43. PKT_TX_THRESHOLD

- Summary: TX almost empty threshold.
- Purpose:
  - Transmit almost empty interrupt fires when the amount of space in the transmit fifo equal to or greater than TX_THRESHOLD.
- Property: 0x120B
- Default: 0x30
- Fields:
  - TX_THRESHOLD[7:0] - default:0x30 Transmit almost empty interrupt fires when the amount of space in the transmit fifo equal to or greater than TX_THRESHOLD.

    Range: 0–64
- Register View

PKT_TX_THRESHOLD							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
TX_THRESHOLD[7:0]							
0x30							

3.4.44. PKT_RX_THRESHOLD

- Summary: RX almost full threshold.
- Purpose:
  - Receive almost full interrupt fires when there are at least RX_THRESHOLD number of bytes present in the receive fifo.
- Property: 0x120C
- Default: 0x30
- Fields:
  - RX_THRESHOLD[7:0] - default:0x30 Receive almost full interrupt fires when there are at least RX_THRESHOLD number of bytes present in the receive fifo.

    Range: 0–64
- Register View

PKT_RX_THRESHOLD							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RX_THRESHOLD[7:0]							
0x30							

3.4.45. PKT_FIELD_1_LENGTH_12_8

- Summary: Byte 1 of field length
- Purpose:
  - This property specifies the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x120D
- Default: 0x00
- Fields:
  - FIELD_1_LENGTH_12_8[4:0] - default:0x00

    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_FIELD_1_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			FIELD_1_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.46. PKT_FIELD_1_LENGTH_7_0

- Summary: Byte 0 of field length
- Purpose:
  - See byte 1 for details.
- Property: 0x120E
- Default: 0x00
- Fields:
  - FIELD_1_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0–0xff
- Register View

PKT_FIELD_1_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FIELD_1_LENGTH_7_0[7:0]							
0x00							

3.4.47. PKT_FIELD_1_CONFIG

- Summary: Field 1 configuration bits.
- Purpose:
  - Field 1 configuration bits common to TX and RX.
- Property: 0x120F
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - PN_START - default:0
    0x1 = Load PN-9 engine with seed value at the start of this field.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_FIELD_1_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	PN_START	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.48. PKT_FIELD_1_CRC_CONFIG

- Summary: Field 1 CRC configuration bits.
- Purpose:
  - Field 1 CRC configuration bits.
- Property: 0x1210
- Default: 0x00
- Fields:
  - CRC_START - default:0
  - 0x1 = Load CRC engine with seed value at the start of this field using CRC_SEED.
  - SEND_CRC - default:0
  - 0x1 = Transmit CRC at the end of this field
  - CHECK_CRC - default:0
  - 0x1 = Check CRC at the end of this field. If PH_FIELD_SPLIT is set, this bit is ignored.
  - CRC_ENABLE - default:0

    0x1 = Enable CRC over this field.
- Register View

PKT_FIELD_1_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CRC_START	0	SEND_CRC	0	CHECK_CRC	0	CRC_ENABLE	
0	0	0	0	0	0	0	

3.4.49. PKT_FIELD_2_LENGTH_12_8

- Summary: Byte 1 of field length
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x1211
- Default: 0x00
- Fields:
  - FIELD_2_LENGTH_12_8[4:0] - default:0x00

    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_FIELD_2_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			FIELD_2_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.50. PKT_FIELD_2_LENGTH_7_0

- Summary: Byte 0 of field length
- Purpose:
  - See byte 1 for details.
- Property: 0x1212
- Default: 0x00
- Fields:
  - FIELD_2_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0-0xff
- Register View

PKT_FIELD_2_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FIELD_2_LENGTH_7_0[7:0]							
0x00							

3.4.51. PKT_FIELD_2_CONFIG

- Summary: Field 2 configuration bits.
- Purpose:
  - Field 2 configuration bits common to TX and RX.
- Property: 0x1213
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - RESERVED - default:0 Reserved.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_FIELD_2_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	RESERVED	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.52. PKT_FIELD_2_CRC_CONFIG

- Summary: Field 2 CRC configuration bits.
- Purpose:
  - Field 2 CRC configuration bits.
- Property: 0x1214
- Default: 0x00
- Fields:
  - RESERVED[1:0] - default:0x0 Reserved.
  - SEND_CRC - default:0

    0x1 = Transmit CRC at the end of this field
  - CHECK_CRC - default:0

    0x1 = Check CRC at the end of this field. If PH_FIELD_SPLIT is set, this bit is ignored.
  - CRC_ENABLE - default:0

    0x1 = Enable CRC over this field.
- Register View

PKT_FIELD_2_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED[1:0]		SEND_CRC	0	CHECK_CRC	0	CRC_ENABLE	
0x0		0	0	0	0	0	

3.4.53. PKT_FIELD_3_LENGTH_12_8

- Summary: Byte 1 of field length
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x1215
- Default: 0x00
- Fields:
  - FIELD_3_LENGTH_12_8[4:0] - default:0x00

    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_FIELD_3_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			FIELD_3_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.54. PKT_FIELD_3_LENGTH_7_0

- Summary: Byte 0 of field length
- Purpose:
  - See byte 1 for details.
- Property: 0x1216
- Default: 0x00
- Fields:
  - FIELD_3_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0–0xff
- Register View

PKT_FIELD_3_LENGTH_7_0							
7	6	5	4	3	2	1	0
FIELD_3_LENGTH_7_0[7:0]							
0x00							

3.4.55. PKT_FIELD_3_CONFIG

- Summary: Field 3 configuration bits.
- Purpose:
  - Field 3 configuration bits common to TX and RX.
- Property: 0x1217
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - RESERVED - default:0 Reserved.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_FIELD_3_CONFIG							
7	6	5	4	3	2	1	0
0x0			4FSK	0	RESERVED	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.56. PKT_FIELD_3_CRC_CONFIG

- Summary: Field 3 CRC configuration bits.
- Purpose:
  - Field 3 CRC configuration bits.
- Property: 0x1218
- Default: 0x00
- Fields:
  - RESERVED[1:0] - default:0x0 Reserved.
  - SEND_CRC - default:0
  - 0x1 = Transmit CRC at the end of this field
  - CHECK_CRC - default:0
  - 0x1 = Check CRC at the end of this field. If PH_FIELD_SPLIT is set, this bit is ignored.
  - CRC_ENABLE - default:0

    0x1 = Enable CRC over this field.
- Register View

PKT_FIELD_3_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED[1:0]		SEND_CRC	0	CHECK_CRC	0	CRC_ENABLE	
0x0		0	0	0	0	0	

3.4.57. PKT_FIELD_4_LENGTH_12_8

- Summary: Byte 1 of field length
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x1219
- Default: 0x00
- Fields:
  - FIELD_4_LENGTH_12_8[4:0] - default:0x00

    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_FIELD_4_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			FIELD_4_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.58. PKT_FIELD_4_LENGTH_7_0

- Summary: Byte 0 of field length
- Purpose:
  - See byte 1 for details.
- Property: 0x121A
- Default: 0x00
- Fields:
  - FIELD_4_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0–0xff
- Register View

PKT_FIELD_4_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FIELD_4_LENGTH_7_0[7:0]							
0x00							

3.4.59. PKT_FIELD_4_CONFIG

- Summary: Field 4 configuration bits.
- Purpose:
  - Field 4 configuration bits common to TX and RX.
- Property: 0x121B
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - RESERVED - default:0 Reserved.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_FIELD_4_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	RESERVED	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.60. PKT_FIELD_4_CRC_CONFIG

- Summary: Field 4 CRC configuration bits.
- Purpose:
  - Field 4 CRC configuration bits.
- Property: 0x121C
- Default: 0x00
- Fields:
  - RESERVED[1:0] - default:0x0 Reserved.
  - SEND_CRC - default:0
    0x1 = Transmit CRC at the end of this field
  - CHECK_CRC - default:0
    0x1 = Check CRC at the end of this field. If PH_FIELD_SPLIT is set, this bit is ignored.
  - CRC_ENABLE - default:0
    0x1 = Enable CRC over this field.
- Register View

PKT_FIELD_4_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED[1:0]		SEND_CRC	0	CHECK_CRC	0	CRC_ENABLE	
0x0		0	0	0	0	0	

3.4.61. PKT_FIELD_5_LENGTH_12_8

- Summary: Byte 1 of field length
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x121D
- Default: 0x00
- Fields:
  - FIELD_5_LENGTH_12_8[4:0] - default:0x00
    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_FIELD_5_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			FIELD_5_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.62. PKT_FIELD_5_LENGTH_7_0

- Summary: Byte 0 of field length
- Purpose:
  - See byte 1 for details.
- Property: 0x121E
- Default: 0x00
- Fields:
  - FIELD_5_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0–0xff
- Register View

PKT_FIELD_5_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FIELD_5_LENGTH_7_0[7:0]							
0x00							

3.4.63. PKT_FIELD_5_CONFIG

- Summary: Field 5 configuration bits.
- Purpose:
  - Field 5 configuration bits common to TX and RX.
- Property: 0x121F
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - RESERVED - default:0 Reserved.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_FIELD_5_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	RESERVED	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.64. PKT_FIELD_5_CRC_CONFIG

- Summary: Field 5 CRC configuration bits.
- Purpose:
  - Field 5 CRC configuration bits.
- Property: 0x1220
- Default: 0x00
- Fields:
  - RESERVED[1:0] - default:0x0 Reserved.
  - SEND_CRC - default:0

    0x1 = Transmit CRC at the end of this field
  - CHECK_CRC - default:0

    0x1 = Check CRC at the end of this field. If PH_FIELD_SPLIT is set, this bit is ignored.
  - CRC_ENABLE - default:0

    0x1 = Enable CRC over this field.
- Register View

PKT_FIELD_5_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED[1:0]		SEND_CRC	0	CHECK_CRC	0	CRC_ENABLE	
0x0		0	0	0	0	0	

3.4.65. PKT_RX_FIELD_1_LENGTH_12_8

- Summary: Byte 1 of field length for RX
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x1221
- Default: 0x00
- Fields:
  - RX_FIELD_1_LENGTH_12_8[4:0] - default:0x00

    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_RX_FIELD_1_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			RX_FIELD_1_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.66. PKT_RX_FIELD_1_LENGTH_7_0

- Summary: Byte 0 of field length for RX
- Purpose:
  - See byte 1 for details.
- Property: 0x1222
- Default: 0x00
- Fields:
  - RX_FIELD_1_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0–0xff
- Register View

PKT_RX_FIELD_1_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RX_FIELD_1_LENGTH_7_0[7:0]							
0x00							

3.4.67. PKT_RX_FIELD_1_CONFIG

- Summary: Field 1 configuration bits for RX.
- Purpose:
  - Field 1 configuration bits for RX.
- Property: 0x1223
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - PN_START - default:0
    0x1 = Load PN-9 engine with seed value at the start of this field.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_RX_FIELD_1_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	PN_START	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.68. PKT_RX_FIELD_1_CRC_CONFIG

- Summary: Field 1 CRC configuration bits for RX.
- Purpose:
  - Field 1 CRC configuration bits.
- Property: 0x1224
- Default: 0x00
- Fields:
  - CRC_START - default:0

    0x1 = Load CRC engine with seed value at the start of this field using CRC_SEED.
  - CHECK_CRC - default:0

    0x1 = Check CRC at the end of this field
  - CRC_ENABLE - default:0

    0x1 = Enable CRC over this field.
- Register View

PKT_RX_FIELD_1_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CRC_START	0x0			CHECK_CRC	0	CRC_ENABLE	
0	0x0			0	0	0	

3.4.69. PKT_RX_FIELD_2_LENGTH_12_8

- Summary: Byte 1 of field length for RX
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x1225
- Default: 0x00
- Fields:
  - RX_FIELD_2_LENGTH_12_8[4:0] - default:0x00

    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_RX_FIELD_2_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			RX_FIELD_2_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.70. PKT_RX_FIELD_2_LENGTH_7_0

- Summary: Byte 0 of field length for RX
- Purpose:
  - See byte 1 for details.
- Property: 0x1226
- Default: 0x00
- Fields:
  - RX_FIELD_2_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0–0xff
- Register View

PKT_RX_FIELD_2_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RX_FIELD_2_LENGTH_7_0[7:0]							
0x00							

3.4.71. PKT_RX_FIELD_2_CONFIG

- Summary: Field 2 configuration bits for RX.
- Purpose:
  - Field 2 configuration bits for RX.
- Property: 0x1227
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - RESERVED - default:0 Reserved.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_RX_FIELD_2_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	RESERVED	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.72. PKT_RX_FIELD_2_CRC_CONFIG

- Summary: Field 2 CRC configuration bits for RX.
- Purpose:
  - Field 2 CRC configuration bits.
- Property: 0x1228
- Default: 0x00
- Fields:
  - RESERVED[1:0] - default:0x0 Reserved.
  - CHECK_CRC - default:0
    0x1 = Check CRC at the end of this field
  - CRC_ENABLE - default:0
    0x1 = Enable CRC over this field.
- Register View

PKT_RX_FIELD_2_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED[1:0]		0x0		CHECK_CRC	0	CRC_ENABLE	
0x0		0x0		0	0	0	

3.4.73. PKT_RX_FIELD_3_LENGTH_12_8

- Summary: Byte 1 of field length for RX
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x1229
- Default: 0x00
- Fields:
  - RX_FIELD_3_LENGTH_12_8[4:0] - default:0x00
    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_RX_FIELD_3_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			RX_FIELD_3_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.74. PKT_RX_FIELD_3_LENGTH_7_0

- Summary: Byte 0 of field length for RX
- Purpose:
  - See byte 1 for details.
- Property: 0x122A
- Default: 0x00
- Fields:
  - RX_FIELD_3_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0–0xff
- Register View

PKT_RX_FIELD_3_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RX_FIELD_3_LENGTH_7_0[7:0]							
0x00							

3.4.75. PKT_RX_FIELD_3_CONFIG

- Summary: Field 3 configuration bits for RX.
- Purpose:
  - Field 3 configuration bits for RX.
- Property: 0x122B
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - RESERVED - default:0 Reserved.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_RX_FIELD_3_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	RESERVED	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.76. PKT_RX_FIELD_3_CRC_CONFIG

- Summary: Field 3 CRC configuration bits for RX.
- Purpose:
  - Field 3 CRC configuration bits.
- Property: 0x122C
- Default: 0x00
- Fields:
  - RESERVED[1:0] - default:0x0 Reserved.
  - CHECK_CRC - default:0
    0x1 = Check CRC at the end of this field
  - CRC_ENABLE - default:0
    0x1 = Enable CRC over this field.
- Register View

PKT_RX_FIELD_3_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED[1:0]		0x0		CHECK_CRC	0	CRC_ENABLE	
0x0		0x0		0	0	0	

3.4.77. PKT_RX_FIELD_4_LENGTH_12_8

- Summary: Byte 1 of field length for RX
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x122D
- Default: 0x00
- Fields:
  - RX_FIELD_4_LENGTH_12_8[4:0] - default:0x00
    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_RX_FIELD_4_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			RX_FIELD_4_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.78. PKT_RX_FIELD_4_LENGTH_7_0

- Summary: Byte 0 of field length for RX
- Purpose:
  - See byte 1 for details.
- Property: 0x122E
- Default: 0x00
- Fields:
  - RX_FIELD_4_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0-0xff
- Register View

PKT_RX_FIELD_4_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RX_FIELD_4_LENGTH_7_0[7:0]							
0x00							

3.4.79. PKT_RX_FIELD_4_CONFIG

- Summary: Field 4 configuration bits for RX.
- Purpose:
  - Field 4 configuration bits for RX.
- Property: 0x122F
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - RESERVED - default:0 Reserved.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_RX_FIELD_4_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	RESERVED	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.80. PKT_RX_FIELD_4_CRC_CONFIG

- Summary: Field 4 CRC configuration bits for RX.
- Purpose:
  - Field 4 CRC configuration bits.
- Property: 0x1230
- Default: 0x00
- Fields:
  - RESERVED[1:0] - default:0x0 Reserved.
  - CHECK_CRC - default:0
    0x1 = Check CRC at the end of this field
  - CRC_ENABLE - default:0
    0x1 = Enable CRC over this field.
- Register View

PKT_RX_FIELD_4_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED[1:0]		0x0		CHECK_CRC	0	CRC_ENABLE	
0x0		0x0		0	0	0	

3.4.81. PKT_RX_FIELD_5_LENGTH_12_8

- Summary: Byte 1 of field length for RX
- Purpose:
  - This property specifices the length of this field in bytes.
  - A value of zero in this property means that the field is not used.
  - If the field is programmed as a variable length field, this property sets the maximum length of the field.
  - Used along with byte 0 property.
- Property: 0x1231
- Default: 0x00
- Fields:
  - RX_FIELD_5_LENGTH_12_8[4:0] - default:0x00
    0x0 = Bit 8 to 12 of the field length.
- Register View

PKT_RX_FIELD_5_LENGTH_12_8							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			RX_FIELD_5_LENGTH_12_8[4:0]				
0x0			0x00				

3.4.82. PKT_RX_FIELD_5_LENGTH_7_0

- Summary: Byte 0 of field length for RX
- Purpose:
  - See byte 1 for details.
- Property: 0x1232
- Default: 0x00
- Fields:
  - RX_FIELD_5_LENGTH_7_0[7:0] - default:0x00 See byte 1 for details.
    Range: 0–0xff
- Register View

PKT_RX_FIELD_5_LENGTH_7_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RX_FIELD_5_LENGTH_7_0[7:0]							
0x00							

3.4.83. PKT_RX_FIELD_5_CONFIG

- Summary: Field 5 configuration bits for RX.
- Purpose:
  - Field 5 configuration bits for RX.
- Property: 0x1233
- Default: 0x00
- Fields:
  - 4FSK - default:0
    0x1 = Enable 4fsk on this field.
  - RESERVED - default:0 Reserved.
  - WHITEN - default:0
    0x1 = Enable whitening on this field.
  - MANCH - default:0
    0x1 = Enable manchester encoding on this field.
- Register View

PKT_RX_FIELD_5_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			4FSK	0	RESERVED	WHITEN	MANCH
0x0			0	0	0	0	0

3.4.84. PKT_RX_FIELD_5_CRC_CONFIG

- Summary: Field 5 CRC configuration bits for RX.
- Purpose:
  - Field 5 CRC configuration bits.
- Property: 0x1234
- Default: 0x00
- Fields:
  - RESERVED[1:0] - default:0x0 Reserved.
  - CHECK_CRC - default:0
  - 0x1 = Check CRC at the end of this field
  - CRC_ENABLE - default:0

    0x1 = Enable CRC over this field.
- Register View

PKT_RX_FIELD_5_CRC_CONFIG							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED[1:0]		0x0		CHECK_CRC	0	CRC_ENABLE	
0x0		0x0		0	0	0	

3.4.85. MODEM_MOD_TYPE

- Summary: Modulation Type
- Purpose:
  - This property selects between OOK, FSK, 4FSK and GFSK modulation, modulation source, and tx direct mode control.
  - The modulator must be configured for one mode through the entire packet. If portions of the packet alternate between FSK and 4FSK modes, the modem should be programmed to 4FSK mode.
- Property: 0x2000
- Default: 0x02
- Fields:
  - TX_DIRECT_MODE_TYPE - default:0

    0 = Direct mode operates in synchronous mode, applies to TX only.

    1 = Direct mode operates in asynchronous mode, applies to TX only. GFSK is not supported.
  - TX_DIRECT_MODE_GPIO[1:0] - default:0x0. Selects which GPIO will be used as the TX data source if MOD_SOURCE = 1. The gpio selected here must be configured as a CMOS input using the GPIO_PIN_CFG command.

    0 = TX direct mode uses gpio0 as data source, applies to TX only.

    1 = TX direct mode uses gpio1 as data source, applies to TX only.

    2 = TX direct mode uses gpio2 as data source, applies to TX only.

    3 = TX direct mode uses gpio3 as data source, applies to TX only.
  - MOD_SOURCE[1:0] - default:0x0

    0 = Modulation source is packet handler fifo

    1 = Modulation source is direct mode pin

    2 = Modulation source is pseudo-random generator
  - MOD_TYPE[2:0] - default:0x2

    0 = CW

    1 = OOK

    2 = 2FSK

    3 = 2GFSK

    4 = 4FSK

    5 = 4GFSK
- Register View

MODEM_MOD_TYPE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
TX_DIRECT_MODE_TYPE	TX_DIRECT_MODE_GPIO[1:0]		MOD_SOURCE[1:0]		MOD_TYPE[2:0]		
0	0x0		0x0		0x2		

3.4.86. MODEM_MAP_CONTROL

- Summary: Controls bit mapping.
- Purpose:
  - Modem Mapping Control.
- Property: 0x2001
- Default: 0x80
- Fields:
  - enmanch - default:1

    0 = Disable Manchester coding.

    1 = Enable Manchester coding.
  - eninv_rxbit - default:0

    0 = Do not invert RX data bits.

    1 = Invert RX data bits.
  - eninv_txbit - default:0

    0 = Do not invert TX data bits.

    1 = Invert TX data bits.
  - eninv_fd - default:0 If set, frequency deviation's priority from negative to positive.
- Register View

MODEM_MAP_CONTROL							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
ENMANCH	ENINV_RXBIT	ENINV_TXBIT	ENINV_FD				
1	0	0	0				

3.4.87. MODEM_DATA_RATE_2

- Summary: Byte 2 of TX data rate in bps (bits per second).
- Purpose:
  - Data rate, unsigned 24-bit
- Property: 0x2003
- Default: 0x0F
- Fields:
  - dr_23_16[7:0] - default:0x0F
    Range: 0–255
- Register View

MODEM_DATA_RATE_2							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
DR_23_16[7:0]							
0x0F							

3.4.88. MODEM_DATA_RATE_1

- Summary: Byte 1 of TX data rate in bps (bits per second).
- Purpose:
  - Data rate, unsigned 24-bit
- Property: 0x2004
- Default: 0x42
- Fields:
  - dr_15_8[7:0] - default:0x42
    Range: 0–255
- Register View

MODEM_DATA_RATE_1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
DR_15_8[7:0]							
0X42							

3.4.89. MODEM_DATA_RATE_0

- Summary: Byte 0 of TX data rate in bps (bits per second).
- Purpose:
  - Data rate, unsigned 24-bit
- Property: 0x2005
- Default: 0x40
- Fields:
  - dr_7_0[7:0] - default:0x40
    Range: 0–255
- Register View

MODEM_DATA_RATE_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
dr_7_0[7:0]							
0x40							

3.4.90. MODEM_FREQ_DEV_2

- Summary: Byte 2 of TX frequency deviation (a 17-bit unsigned number). This only programs the MSB of TX frequency deviation.
- Purpose:
  - Frequency deviation, unsigned 17-bit.
- Property: 0x200A
- Default: 0x00
- Fields:
  - freqdev_16 - default:0
- Register View

MODEM_FREQ_DEV_2							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x00							FREQDEV_16
0x00							0

3.4.91. MODEM_FREQ_DEV_1

- Summary: Byte 1 of frequency deviation.
- Purpose:
  - Frequency deviation, unsigned 17-bit.
- Property: 0x200B
- Default: 0x06
- Fields:
  - freqdev_15_8[7:0] - default:0x06
    Range: 0–255
- Register View

MODEM_FREQ_DEV_1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
freqdev_15_8[7:0]							
0x06							

3.4.92. MODEM_FREQ_DEV_0

- Summary: Byte 0 of frequency deviation.
- Purpose:
  - Frequency deviation, unsigned 17-bit.
- Property: 0x200C
- Default: 0xD3
- Fields:
  - freqdev_7_0[7:0] - default:0xD3
    Range: 0–255
- Register View

MODEM_FREQ_DEV_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
freqdev_7_0[7:0]							
0xD3							

3.4.93. MODEM_ANT_DIV_CONTROL

- Summary: Specifies antenna diversity controls. Antenna diversity mode is valid for standard packet only.
- Purpose:
  - Specifies pm detection threshold and GPIO config in antenna diversity mode.
- Property: 0x2049
- Default: 0x80
- Fields:
  - ant2pm_thd[3:0] - default:0x8 The second phase preamble detection threshold in ANT-DIV mode. Default is set to 8 bits threshold.
  - matap - default:0 Number of taps for moving average filter during Antenna Diversity RSSI evaluation. Allows for reduced noise variation on measured RSSI value but with slower update rate.

    0 = Filter tap length is 8*Tb prior to first PREAMBLE_VALID, and 4*Tb thereafter.

    1 = Filter tap length is 8*Tb.
  - antdiv[2:0] - default:0x0 The GPIO must be configured for antenna diversity for the algorithm to work properly.

    0 = RX/TX state: GPIO-Ant1=1, GPIO Ant2=0: Non-RX/TX State GPIO Ant1=0, GPIO Ant2 = 0.

    1 = RX/TX state: GPIO-Ant1=0, GPIO Ant2=1: Non-RX/TX State GPIO Ant1=0, GPIO Ant2 = 0.

    2 = RX/TX state: GPIO-Ant1=1, GPIO Ant2=0: Non-RX/TX State GPIO Ant1=1, GPIO Ant2 = 1.

    3 = RX/TX state: GPIO-Ant1=0, GPIO Ant2=1: Non-RX/TX State GPIO Ant1=1, GPIO Ant2 = 1.

    4 = RX/TX state: GPIO=Antenna diversity algorithm: Non-RX/TX State GPIO Ant1=0, GPIO Ant2 = 0.

    5 = RX/TX state: GPIO=Antenna diversity algorithm: Non-RX/TX State GPIO Ant1=1, GPIO Ant2 = 1.

    6 = RX/TX state: GPIO=Antenna diversity algorithm in beacon mode: Non-RX/TX State GPIO Ant1=0, GPIO Ant2 = 0.

    7 = RX/TX state: GPIO=Antenna diversity algorithm in beacon mode: Non-RX/TX State GPIO Ant1=1, GPIO Ant2 = 1.
- Register View

MODEM_ANT_DIV_CONTROL							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
ANT2PM_THD[3:0]				MATAP	ANTDIV[2:0]		
0X8				0	0X0		

3.4.94. MODEM_RSSI_THRESH

- Summary: RSSI threshold control
- Purpose:
  - Selects threshold for clear channel assessment. If RSSI value is above this threshold, the CCA GPIO will be high and the RSSI interrupt will be generated.
- Property: 0x204A
- Default: 0xFF
- Fields:
  - RSSI_THRESH[7:0] - default:0x08 Selects threshold for clear channel assessment. If RSSI value is above this threshold, the CCA GPIO will be high and the RSSI interrupt will be generated.

    Range: 0–255
- Register View

MODEM_RSSI_THRESH							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RSSI_THRESH[7:0]							
0xFF							

3.4.95. MODEM_RSSI_JUMP_THRESH

- Summary: RSSI jumping detection threshold.
- Purpose:
  - RSSI jumping detection threshold, step in 1dB.
- Property: 0x204B
- Default: 0x0C
- Fields:
  - rssijmpthd[6:0] - default:0x0C RSSI jumping detection threshold.
- Register View

MODEM_RSSI_JUMP_THRESH							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	rssijmpthd[6:0]						
0	0x0C						

3.4.96. MODEM_RSSI_CONTROL

- Summary: RSSI control
- Purpose:
  - Selects where in the packet to latch the RSSI value in the RSSI Latch fast response register. The latched value can also be read using GET_MODEM_STATUS command.
- Property: 0x204C
- Default: 0x01
- Fields:
  - AVERAGE - default:0
    0 = RSSI updated every bit.
    1 = RSSI averaged over 4 bits
  - LATCH[1:0] - default:0x1
    0 = Latch disabled, will always read 0
    1 = Latches at preamble detect
    2 = Latches at sync detect
    3 = Latches RSSI 4Tb (7Tb if averaging is enabled) after RX is enabled.
- Register View

MODEM_RSSI_CONTROL							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0			AVERAGE	0x0		LATCH[1:0]	
0x0			0	0x0		0x1	

3.4.97. MODEM_RSSI_CONTROL2

- Summary: RSSI control
- Purpose:
  - Enable RSSI jumping detection. Used to detect an RSSI jump as configured by MODEM_RSSI_CONTROL while receiving a packet. Can be useful to detect interferring or secondary incoming packet.
- Property: 0x204D
- Default: 0x00
- Fields:
  - rssijmp_dwn - default:0 If set, enable RSSI jumping-down detection.
  - rssijmp_up - default:0 If set, enable RSSI jumping-up detection.
  - enrssijmp - default:0
    Enable RSSI jumping detection.
    Once RSSI difference between 2Tb or 4Tb is above the RSSI jumping threshold, and interrupt will be generated.
  - jmpdlylen - default:0
    0 = RSSI jumping detection is running with 2Tb
    1 = RSSI jumping detection is running with 4Tb
  - enjmprx - default:0 If set, RSSI jumping detection will force RX machine to reset.
- Register View

MODEM_RSSI_CONTROL2							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0X0		RSSIJMP_DWN	RSSIJMP_UP	ENRSSIJMP	JMPDLYLEN	ENJMPRX	
0x0		0	0	0	0	0	

3.4.98. MODEM_RSSI_COMP

- Summary: RSSI reading offset.
- Purpose:
  - Offsets RSSI curve in 1dB steps. 32 is no offset, lower will adjust RSSI down, and higher will adjust RSSI up.
- Property: 0x204E
- Default: 0x32
- Fields:
- rssi_comp[6:0] - default:0x32 RSSI reading offset.

  Range: 0–127
- Register View

MODEM_RSSI_COMP							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	RSSI_COMP[6:0]						
0	0x32						

3.4.99. PA_MODE

- Summary: PA operating mode and groups.
- Purpose:
  - Specify PA mode
- Property: 0x2200
- Default: 0x10
- Fields:
  - PA_GROUP[3:0] - default: 0x02

    1 = 1 group on

    2 = 2 groups on
  - PA_MODE[1:0] - default:0x0 PA mode.

    0 = Switch, for Square Wave or Class E

    1 = Switch Current
- Register View

PA_MODE							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x0		PA_GROUP[3:0]				PA_MODE[1:0]	
0x0		0x4				0x0	

3.4.100. PA_PWR_LVL

- Summary: PA Power Level Configuration
- Purpose:
  - Adjusts the TX power level in fine resolution.
- Property: 0x2201
- Default: 0x7F
- Fields:
  - DDAC[6:0] - default:0x7F
    Range: 0-127
- Register View

PA_PWR_LVL							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	DDAC[6:0]						
0	0x7F						

3.4.101. PA_BIAS_CLKDUTY

- Summary: PA Bias and TX clock duty cycle configuration
- Property: 0x2202
- Default: 0x00
- Fields:
  - CLK_DUTY[1:0] - default:0x0 Select 25% or 50% duty cycle clocks for transmitter to improve transmit efficiency. 25% is more efficient but has higher harmonics compared to 50%.
    0 = TXP: 50%, TXN: 50%
    1 = TXP: 25%, TXN: 25%
  - OB[5:0] - default:0x00 Factor to multiply the PA output current as a way to control the output power steps. This setting only applies to switched current mode.
- Range: 0–63
- Register View

PA_BIAS_CLKDUTY							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CLK_DUTY[1:0]		OB[5:0]					
0x0		0x00					

3.4.102. PA_TC

- Summary: PA Ramping Time Control Register
- Purpose:
  - Configuration control for PA power ramping in order to minimize switching spectrum noise. In (G)FSK mode, the values of TC and FSK_MOD_DLY should be programmed together so data modulation only occurs after the PA power ramping has been completed.
- Property: 0x2203
- Default: 0x5D
- Fields:
  - FSK_MOD_DLY[2:0] - default:0x02
    In (G)FSK mode this is the delay from PA enable to the start of modulation. This is necessary to give the PA time to ramp. 0 = Modulation will begin 2 µs after PA is enabled. 1 = Modulation will begin 6 µs after PA is enabled. 2 = Modulation will begin 10 µs after PA is enabled. 3 = Modulation will begin 14 µs after PA is enabled. 4 = Modulation will begin 18 µs after PA is enabled. 5 = Modulation will begin 22 µs after PA is enabled. 6 = Modulation will begin 26 µs after PA is enabled.
    7 = Modulation will begin 30 µs after PA is enabled.
    TC[4:0] - default:0x1D Ramping time (up and down) for the output device. The ramp rate is determined by the formula: Ramp time = 20 s/(32-TC) NOTE: If using (G)FSK modulation this value must be programmed to a value less than 29.
- Register View

PA_TC							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FSK_MOD_DLY[2:0]			TC[4:0]				
0x02			0x1D				

3.4.103. MATCH_VALUE_1

- Summary: Match 1 value.
- Purpose:
  - See “AN626: Packet Handler Operation for Si446x RFICs” for details on using the packet handler features.
- Property: 0x3000
- Default: 0x00
- Fields:
  - VALUE_1[7:0] - default:0x00
    Range: 0–0xFF
- Register View

MATCH_VALUE_1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
VALUE_1[7:0]							
0x00							

3.4.104. MATCH_MASK_1

- Summary: Match 1 mask.
- Purpose:
  - See “AN626: Packet Handler Operation for Si446x RFICs” for details on using the packet handler features.
- Property: 0x3001
- Default: 0x00
- Fields:
  - MASK_1[7:0] - default:0x00
    Range: 0–0xFF
- Register View

MATCH_MASK_1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
MASK_1[7:0]							
0x00							

3.4.105. MATCH_CTRL_1

- Summary: Packet match enable and match 1 configuration.
- Purpose:
  - Enable packet match processing and pattern 1 matches or not.
- Property: 0x3002
- Default: 0x00
- Fields:
  - POLARITY - default:0
    0x00 = True if packet matches.
    0x01 = True if packet doesn't match.
  - MATCH_EN - default:0 Note: This bit is quite different from other pattern match controlling.
    1 = Enable packet match.
  - OFFSET[4:0] - default:0x00 Pattern match 1 offset in byte after sync word
    Range: 0–0x1F
- Register View

MATCH_CTRL_1							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
POLARITY	MATCH_EN	0	OFFSET[4:0]				
0	0	0	0x00				

3.4.106. MATCH_VALUE_2

- Summary: Match 2 value.
- Purpose:
  -
- Property: 0x3003
- Default: 0x00
- Fields:
  - VALUE_2[7:0] - default:0x00
    Range: 0–0xFF
- Register View

MATCH_VALUE_2							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
VALUE_2[7:0]							
0x00							

3.4.107. MATCH_MASK_2

- Summary: Match 2 mask.
- Purpose:
  -
- Property: 0x3004
- Default: 0x00
- Fields:
  - MASK_2[7:0] - default:0x00
    Range: 0–0xFF
- Register View

MATCH_MASK_2							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
MASK_2[7:0]							
0x00							

3.4.108. MATCH_CTRL_2

- Summary: Match 2 configuration.
- Purpose:
  - Enable pattern 2 matches or not.
- Property: 0x3005
- Default: 0x00
- Fields:
  - POLARITY - default:0
    0x00 = True if packet matches.
    0x01 = True if packet doesn't match.
  - LOGIC - default:0
    0x00 = AND with previous MATCH field.
    0x01 = OR with previous MATCH field.
  - OFFSET[4:0] - default:0x00 Match 2 offset in byte after sync word
    Range: 0–0x1F
- Register View

MATCH_CTRL_2							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
POLARITY	LOGIC	0	OFFSET[4:0]				
0	0	0	0x00				

3.4.109. MATCH_VALUE_3

- Summary: Match 3 value.
- Purpose:
  -
- Property: 0x3006
- Default: 0x00
- Fields:
  - VALUE_3[7:0] - default:0x00
    Range: 0–0xFF
- Register View

MATCH_VALUE_3							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
VALUE_3[7:0]							
0x00							

3.4.110. MATCH_MASK_3

- Summary: Match 3 mask.
- Purpose:
  -
- Property: 0x3007
- Default: 0x00
- Fields:
  - MASK_3[7:0] - default:0x00
    Range: 0–0xFF
- Register View

MATCH_MASK_3							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
MASK_3[7:0]							
0x00							

3.4.111. MATCH_CTRL_3

- Summary: Match 3 configuration.
- Purpose:
  - Enable pattern 3 matches or not.
- Property: 0x3008
- Default: 0x00
- Fields:
  - POLARITY - default:0
    0x00 = True if packet matches.
    0x01 = True if packet doesn't match.
  - LOGIC - default:0
    0x00 = AND with previous MATCH field.
    0x01 = OR with previous MATCH field.
  - OFFSET[4:0] - default:0x00 Match 3 offset in byte after sync word
    Range: 0–0x1F
- Register View

MATCH_CTRL_3							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
POLARITY	LOGIC	0	OFFSET[4:0]				
0	0	0	0x00				

3.4.112. MATCH_VALUE_4

- Summary: Match 4 value.
- Purpose:
  -
- Property: 0x3009
- Default: 0x00
- Fields:
  - VALUE_4[7:0] - default:0x00
    Range: 0–0xFF
- Register View

MATCH_VALUE_4							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
VALUE_4[7:0]							
0x00							

3.4.113. MATCH_MASK_4

- Summary: Match 4 mask.
- Purpose:
  -
- Property: 0x300A
- Default: 0x00
- Fields:
  - MASK_4[7:0] - default:0x00
    Range: 0–0xFF
- Register View

MATCH_MASK_4							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
MASK_4[7:0]							
0x00							

3.4.114. MATCH_CTRL_4

- Summary: Match 4 configuration.
- Purpose:
  - Enable pattern 4 matches or not.
- Property: 0x300B
- Default: 0x00
- Fields:
  - POLARITY - default:0
    0x00 = True if packet matches.
    0x01 = True if packet doesn't match.
  - LOGIC - default:0
    0x00 = AND with previous MATCH field.
    0x01 = OR with previous MATCH field.
  - OFFSET[4:0] - default:0x00 Match 4 offset in byte after sync word
    Range: 0–0x1F
- Register View

MATCH_CTRL_4							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
POLARITY	LOGIC	0	OFFSET[4:0]				
0	0	0	0x00				

3.4.115. FREQ_CONTROL_VCOCNT_RX_ADJ

- Summary: VCO target count adjustment for RX
- Purpose:
  - VCO target count adjustment for RX, signed
- Property: 0x4007
- Default: 0xFF
- Fields:
  - vcocnt_rx_adj[7:0] - default:0xFF
    Range: –128 to 127
- Register View

FREQ_CONTROL_VCOCNT_RX_ADJ							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
VCOCNT_RX_ADJ[7:0]							
0XFF							

3.4.116. RX_HOP_CONTROL

- Summary: RX hop control.
- Purpose:
  - Sets RSSI timeout value and select RX hop condition.
- Property: 0x5000
- Default: 0x04
- Fields:
  - HOP_EN[2:0] - default:0x0 RX hop condition.

    0 = Hop disabled

    1 = Hop if preamble timeout occurs. If no preamble detected after RX preamble timeout, then hop. Otherwise a preamble is detected, stay on channel.

    2 = Hop if either RSSI timeout occurs or preamble timeout occurs. Either timeout condition forces hop, whichever occurs first. Otherwise stay on channel.

    3 = Hop if preamble timeout or invalid sync word.

    4 = Hop on RSSI timeout, preamble timeout or invalid sync word.
  - RSSI_TIMEOUT[3:0] - default:0x4 Sets the RSSI time out expressed in nibbles.
- Register View

<table>
<tr><th colspan="8">RX_HOP_CONTROL</th></tr>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>0</td><td colspan="3">HOP_EN[2:0]</td><td colspan="4">RSSI_TIMEOUT[3:0]</td></tr>
<tr><td>0</td><td colspan="3">0x0</td><td colspan="4">0x4</td></tr>
</table>

3.4.117. RX_HOP_TABLE_SIZE

- Summary: Number of entries in the RX hop table.
- Purpose:
  - Number of entries in the RX hop table.
- Property: 0x5001
- Default: 0x01
- Fields:
  - RX_HOP_TABLE_SIZE[6:0] - default:0x01

    Range: 1–64
- Register View

<table>
<tr><th colspan="8">RX_HOP_TABLE_SIZE</th></tr>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>0</td><td colspan="7">RX_HOP_TABLE_SIZE[6:0]</td></tr>
<tr><td>0</td><td colspan="7">0x01</td></tr>
</table>

3.4.118. RX_HOP_TABLE_ENTRY_0

- Summary: No.1 entry in RX hopping table.
- Purpose:
  - No.1 entry in RX hopping table. Skip this entry if 0xFF.
- Property: 0x5002
- Default: 0
- Fields:
  - CHANNEL_NUM[7:0] - default:0x00
    Range: 0–255
    255 = Hopping entry is invalid.
- Register View

RX_HOP_TABLE_ENTRY_0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CHANNEL_NUM[7:0]							
0x00							

3.4.119. RX_HOP_TABLE_ENTRY_xx

- Summary: No.x entry in RX hopping table.
- Purpose:
- No.2 entry in RX hopping table. Skip this entry if 0xFF.
- Property: 0x50xx
- Default: 1
- Fields:
  - CHANNEL_NUM[7:0] - default:0x01
    Range: 0–255
    255 = Hopping entry is invalid.
- Register View

RX_HOP_TABLE_ENTRY_xx							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CHANNEL_NUM[7:0]							
0x01							

CONTACT INFORMATION

Silicon Laboratories Inc.

400 West Cesar Chavez
Austin, TX 78701
Tel: 1+(512) 416-8500
Fax: 1+(512) 416-9669
Toll Free: 1+(877) 444-3032

Please visit the Silicon Labs Technical Support web page:
https://www.silabs.com/support/pages/contacttechnicalsupport.aspx
and register to submit a technical support request.

The information in this document is believed to be accurate in all respects at the time of publication but is subject to change without notice. Silicon Laboratories assumes no responsibility for errors and omissions, and disclaims responsibility for any consequences resulting from the use of information included herein. Additionally, Silicon Laboratories assumes no responsibility for the functioning of undescribed features or parameters. Silicon Laboratories reserves the right to make changes without further notice. Silicon Laboratories makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Silicon Laboratories assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. Silicon Laboratories products are not designed, intended, or authorized for use in applications intended to support or sustain life, or for any other application in which the failure of the Silicon Laboratories product could create a situation where personal injury or death may occur. Should Buyer purchase or use Silicon Laboratories products for any such unintended or unauthorized application, Buyer shall indemnify and hold Silicon Laboratories harmless against all claims and damages.