SAM R34/R35 and WLR089U0

SAM R34/R35 and WLR089U0 Radio Utility Commands Reference Manual

Introduction

The SAM R34/R35 is a family of ultra-low power microcontrollers combined with a UHF transceiver communication interface that supports LoRa® and FSK modulations. The SAM R34/R35 devices and the WLR089U0 module can be tested for Radio Frequency (RF) performance using a simple UART interface to a host. This UART command interface handles the radio configuration and control through an optimized text command/response interface to the host system. This document is intended to describe in detail the various commands available for RF testing on SAM R34/R35 devices including the WLR089U0 module.

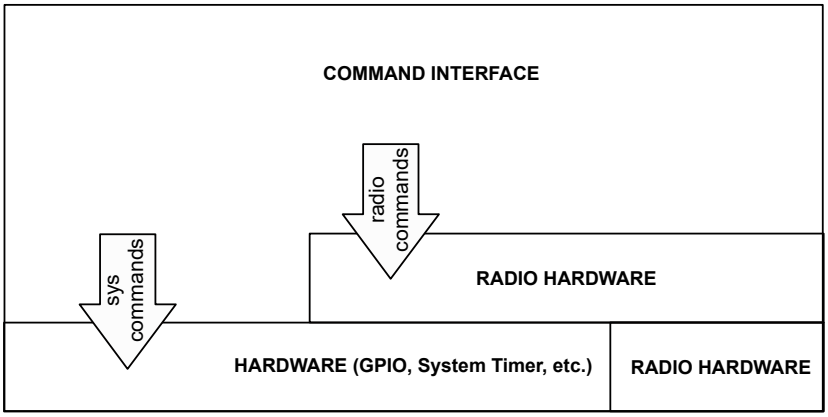
To enable the SAM R34/R35 or WLR089U0 devices for RF testing, the devices need to be programmed with the device specific SAM R34/R35 or WLR089U0 Radio Utility Firmware Project. The Radio utility commands can be exercised on the ATSAMR34-XPRO evaluation kit (DM320111) for quick RF evaluation. Atmel Studio 7 is used for programming the Radio Utility hex file through a micro-B USB cable to the debug USB port. The target device is programmed and debugged by the on-board Embedded Debugger and no external programmer or debugger tool is needed when using the ATSAMR34-XPRO for evaluation. Refer to the Atmel Studio 7 User Guide for information regarding how to compile and program the kit.

The following figure illustrates the command interface view of the SAM R34/R35. The supportable command types are:

- Radio configuration and control, using the radio group of commands
- Other SiP functions, using the system group of commands

This command reference manual is also applicable for the WLR089U0 module, which is based on the ATSAMR34J18B IC. When using the WLR089U0 module, use the WLR089 Xplained Pro (EV23M25A) as the development kit. The user must program the WLR089U0 module with the WLR089 Radio Utility firmware project.

Figure 1. Command Interface of SAM R34/R35 Architecture



Features

- Transmission of Packet using LoRa and FSK Modulation

- Reception of Packet using LoRa and FSK Modulation
- CW Mode Transmission
- Variety of Commands for RF Configuration

Note: Refer to the SAM R34/R35 Low Power LoRa® Sub-GHz SiP Datasheet (DS70005356) for details on the hardware specifications of the SAM R34/R35 devices.

Table of Contents

1. Quick References

1.1 Design Documentation and Relevant Links

- ATSAMR34-XPRO User Guide (DS50002803)
- SAM R34/R35 Low Power LoRa® Sub-GHz SiP Data Sheet (DS70005356)
- WLR089 Xplained Pro User's Guide (DS50003040A)
- WLR089U0 Low Power LoRa® Sub-GHz Module Data Sheet (DS70005435A)
- ATSAMR34-XPRO / WLR089-XPRO – Radio Utility commands can be exercised on the ATSAMR34-XPRO to ATSAMR34-XPRO/WLR089-XPRO development kit for quick RF evaluation
- Microchip RF tools for LoRa - RF tool can be used for detailed testing and RF certification
- Atmel Studio – Atmel Studio presents Free Atmel IDE for development of C/C++ and assembler code for microcontrollers and relevant documentation
- EDBG User Guide – User guide containing more information about the on-board Embedded Debugger
- Data Visualizer – Data Visualizer is a program used for processing and visualizing data. Data Visualizer can receive data from various sources such as the Embedded Debugger Data Gateway Interface found on Xplained Pro boards and COM ports

2. UART Interface

All of the SAM R34/R35 devices including the WLR089U0 module settings and commands are transmitted over UART using the ASCII interface. All commands need to be terminated with <CR><LF> and any generated replies also terminated by the same sequence. The default settings for the UART interface are 115200 bps, 8 bits, no parity, 1 Stop bit, no flow control.

By default, the pins PA04 and PA05 of SAM R34/R35 and WLR089U0 are configured to UART_TX (Output-Communication UART Transmit) and UART_RX (Input-Communication UART Receive).

This document illustrates various RF testing commands using the ATSAMR34 XPRO and WLR089 XPRO evaluation kit. The kits have an on-board embedded debugger (EDBG). The EDBG features a CDC class USB interface that implements a Virtual COM Port. To enable easy communication between a PC and the SiP, PA04 pin and PA05 pins are connected to the SAM R34/R35 devices including the WLR089U0 module. The configuration options such as baud rate, parity and stop bits must be specified in the terminal application, which propagates the configuration to the EDBG Virtual COM port on connection. When using the ATSAMR34 XPRO and WLR089 XPRO, a terminal application such as Tera Term can be used to communicate from a PC to the SiP.

Communication to the SAM R34/R35 and WLR089U0 through another SERCOM can be enabled by changing the hardware configuration of the Radio Utility Firmware Project.

3. Command Reference

The SAM R34/R35 devices, including the WLR089U0 module, support a variety of commands for configuration. This section describes the commands in detail and provides examples.

3.1 Command Syntax

The user sends the commands followed by optional parameters to issue commands to the SAM R34/R35 SiP or WLR089U0. Commands (keywords) are case-sensitive, and spaces must not be used in parameters. Hexadecimal input data can be uppercase or lowercase. String text data is case-insensitive. The use of shorthand for parameters is NOT supported.

Depending on the command, the parameter may expect values in either decimal or hexadecimal form; refer to the command description for the expected form. For example, when configuring the frequency, the command expects a decimal value in Hertz such as 923300000 (923.3 MHz). To enter a number in hex form, use the value directly. For example, the hex value `0xFF` would be entered as FF.

3.2 Command Organization

The following table shows the general command categories.

Table 3-1. Command Types

Command Type	Keyword	Description
System	`<sys>`	Provides system level behavior actions, gathers status information on the firmware and hardware version.
Transceiver Commands	`<radio>`	Provides radio specific configurations, directly accessing and updating the transceiver setup.

Notes: Upon successful reception of commands, the SiP responds with one of the following:

- `ok`
- `invalid_param`
- Requested Information
- Descriptive Error Message

3.3 System Commands

System commands begin with the system keyword `<sys>` and include the categories.

Table 3-2. System Commands

Parameter	Description
`sleep`	Puts the system in sleep for a finite number of milliseconds.
`reset`	Resets and restarts the SAM R34/R35 devices including the WLR089U0 module.
`factoryRESET`	Resets internal configurations to factory default values and restarts SAM R34/R35 devices including the WLR089U0 module.
`get ver`	Provides information related to hardware platform, firmware version and so on.

3.3.1 `sys reset`

This command resets and restarts the SAM R34/R35 devices including the WLR089U0 module; stored internal configurations will be loaded automatically upon reboot.

Response:

- ```
  Last reset cause: System Reset Request
  LoRaWAN Stack UP
  SAMR34 Xpro MLS_SDK_X_Y_E_Z MMM DD YYYY HH:MM:SS
  ```
- `Invalid_param`: If the entered command is invalid

Example:

```
sys reset

Last reset cause: System Reset Request
LoRaWAN Stack UP
SAMR34 Xpro MLS_SDK_1_0_P_4 Jul 14 2020 20:52:31
```

3.3.2 sys sleep <sleepmode> <sleepduration>

`<sleepmode>` string represents the type of Sleep mode. The following are the supported Sleep modes:

- Standby
- Backup

`<sleepduration>` string represents the duration of Sleep. The minimum sleep time for any of the sleep modes is 1000 ms.

3.3.2.1 sys sleep standby <sleepduration>

Response:

- `sleep_ok <x> ms`
- `invalid_param`: If the entered Sleep mode or command is invalid

The system will be in Sleep mode for the maximum time (36 hours, 26 minutes). For example, if any software timer is running/transaction is ongoing, the sleep time will be lower than the specified maximum sleep time. Pressing the User button (SW0) on ATSAMR34 XPRO/WLR089 XPRO will wake up the system.

Example:

```
sys sleep standby 1000  // Puts the system to standby sleep for a duration of 1000ms
sleep_ok 990 ms
```

3.3.2.2 sys sleep backup <sleepduration>

Response:

- ```
  Last reset cause: LoRaWAN Stack UP
  SAMR34 Xpro MLS_SDK_1_0_P_4 Jul 14 2020 20:52:31
  ```
- `invalid_param`: If the entered Sleep mode or command is invalid

The system will be in Backup sleep mode. Only resetting the device will wake up the device from sleep.

Example:

```
sys sleep backup 1000

Last reset cause: System Reset Request
LoRaWAN Stack UP
SAMR34 Xpro MLS_SDK_1_0_P_4 Jul 14 2020 20:52:31
```

3.3.3 sys factoryRESET

This command resets SAM R34/R35 devices, including the WLR089U0 module SiP internal configurations, to factory default values and restarts them. The user will lose all RF settings.

Response:

```
sys factoryRESET

Last reset cause: System Reset Request
LoRaWAN Stack UP
SAMR34 Xpro MLS_SDK_X_Y_E_Z MMM DD YYYY HH:MM:SS
```

Example:

```
sys factoryRESET

Last reset cause: System Reset Request
LoRaWAN Stack UP
SAMR34 Xpro MLS_SDK_1_0_P_4 Jul 14 2020 20:52:31
```

3.3.4 sys get ver

This command returns the information related to the hardware platform, firmware version, release date and timestamp on firmware creation.

Response:

```
SAMR34 Xpro MLS_SDK_X_Y_E_Z MMM DD YYYY HH:MM:SS
```

where,

- MLS_SDK_X_Y_E_Z is the firmware version
- MMM is the month
- DD is the day
- YYYY is the year
- HH:MM:SS is the hour, minutes, seconds

Format – [HW] [FW] [Date] [Time]). [Date] and [Time] refer to the release of the firmware.

Example:

```
sys get ver

SAMR34 Xpro MLS_SDK_1_0_P_4 Jul 14 2020 20:52:31
```

3.4 Radio Commands

Table 3-3. Radio Commands

Parameter	Description
rx	Configures the radio to receive simple radio packets according to prior configuration settings.
tx	Configures a simple radio packet transmission according to the prior configuration settings.
cw	Puts the SiP into a Continuous Wave (CW) transmission for system tuning or certification use.
set	Allows modification to the radio setting directly. This command allows the user to change the method of radio operation within the SiP type band limits.
get	Grants the ability to read out the present radio configuration settings.

Table 3-4. Radio Parameters Availability for Different Operations

Command	radio get	radio set	Availability for LoRa® Modulation	Availability for FSK Modulation
bt	√	√	—	√
mod	√	√	√	√
freq	√	√	√	√
pwr	√	√	√	√
sf	√	√	√	—

.....continued				
Command	**radio get**	**radio set**	**Availability for LoRa® Modulation**	**Availability for FSK Modulation**
afcbw	√	√	—	√
rxbw	√	√	—	√
bitrate	√	√	—	√
fdev	√	√	—	√
prlen	√	√	—	√
crc	√	√	√	√
iqi	√	√	√	—
cr	√	√	√	—
wdt	√	√	√	√
sync	√	√	√	√
bw	√	√	√	—
snr	√	—	√	—
pktrssi	√	—	√	—
lbt	√	√	√	√

3.4.1 radio rx <rxWindowSize>

<rxWindowSize>: Decimal number representing the number of symbols (for LoRa modulation) or time-out in milliseconds (for FSK modulation) that the receiver is opened, from 0 to 65535. Set <rxWindowSize> to '0' in order to enable the Continuous Reception mode. Continuous Reception mode is exited once a valid packet is received or if an rxstop command is issued or the Watchdog Timer expires.

Response: This command may reply with two responses:

1. The first response is received immediately after entering the command.
2. If the command is valid (ok reply is received), a second response is received after the reception of a packet or after the time-out occurred.

Response after entering the command:

- ok – If the parameter is valid and the transceiver is configured in Receive mode.
- invalid_param – If the parameter is not valid.
- busy – If the transceiver is currently busy.

Response after the receive process:

- radio_rx <data> – If the reception was successful, the <data>: hexadecimal value that is received.
- radio_err – If the reception is not successful, a reception time-out occurred.
  - Example: radio rx 0 // Puts the radio into continuous Receive mode.

Note: Ensure the radio Watchdog Timer time-out is higher than the Receive window size.

3.4.2 radio rxstop

This command enables the transceiver to go to Sleep mode from Receive mode.
Note: The radio rxstop command can be used to exit the Continuous Reception mode.

Response after entering the command:

- ok – If the transceiver has successfully come out from Receive mode.
- invalid_request – If the transceiver is not in Receive mode.

Example:

```
radio rxstop // Stops the Receive and enables the transceiver to Sleep
```

3.4.3 radio tx <data> <count>

<data>: Hexadecimal value representing the data to be transmitted, from 0 to 255 bytes for LoRa modulation and from 0 to 64 bytes for FSK modulation.

<count>: Decimal value representing the count of the data to transmitted multiple times from 0 to 65535 bytes for LoRa modulation and for FSK modulation.

Response: This command may reply with the following responses.

1. The first response is received immediately after entering the command.
2. If the command is valid (ok reply received), a second reply radio_tx_ok is received as per the <count> value denoting the number of effective transmissions. If the count value is '0', a second reply is received one time after the effective transmission; transmission happens one time.
3. This response gives a summary of the transmission. The responses are:
   - Total packet (Total packet transmissions initiated)
   - Sent (Total packets transmitted successfully)
   - Channel busy (Total packet transmission failures)

Response after entering the command:

- ok – If the parameter is valid and the transceiver is configured in Transmit mode.
- invalid_param – If the parameter is not valid.
- busy – If the transceiver is currently busy.
- radio_tx_ok – If the transmission was successful and the transmission will be repeated until it reaches the count value.
- radio_err – If the transmission was unsuccessful (interrupted by radio Watchdog Timer time-out). This command transmits the <data> passed number of times as per the value given in the count.

Example:

```
radio tx 55aa55aa55aa 5 // Transmit a packet 5 times
ok
radio_tx_ok
radio_tx_ok
radio_tx_ok
radio_tx_ok
radio_tx_ok
Total packet: 5,Sent: 5,Channel busy: 0
```

Notes:

1. In order to meet ETSI regulations in the given frequency bands, the radio has to use either Listen Before Talk (LBT) + Adaptive Frequency Agility (AFA) or duty cycle limitations. By issuing the radio tx <data> command, the module does not perform the LBT before transmission, thus the user has to make sure that duty cycle limits are not violated.
2. When transmitting FSK packets, the payload and the 2-byte CRC is whitened by being XORed with a pseudo-random sequence generated by an LFSR with the polynomial X9 + X5 + 1. This process is automatically reverted on reception so that it is transparent to the user.

Response after the effective transmission:

- radio_tx_ok – If the transmission was successful.
- radio_err – If the transmission was unsuccessful. This command transmits the <data> passed.

3.4.4 radio cw <state>

<state>: String representing the state of the Continuous Wave (CW) mode, either on or off.

Response:

- `ok` if the state is `on`.
  - `SAMR34 X.Y.Z MMM DD YYYY HH:MM:SS` where, `X.Y.Z` is the firmware version, MMM is the month, DD is the day, YYYY is the year, HH:MM:SS is the hour, minutes, seconds (format: [HW] [FW] [Date] [Time]). [Date] and [Time] refer to the firmware release, if the state is `off`.
- `invalid_param` if the state is not valid.

This command will enable or disable the CW mode on the SiP. CW mode allows the user to put the transceiver into Transmission mode to observe the generated signal. By altering the radio settings, the user can observe the changes in transmissions levels. For example, `radio cw on`.

3.4.5 Radio Set Commands

Table 3-5. Radio Set Commands

Parameter	Description
`bt`	Set the data shaping for the Frequency Shift Keying (FSK) modulation type.
`mod`	Set the SiP Modulation mode.
`freq`	Set the current operation frequency for the radio.
`pwr`	Set the output power level used by the radio during transmission.
`sf`	Set the requested Spreading Factor (SF) to be used during transmission.
`afcbw`	Set the value used by the automatic frequency correction bandwidth.
`rxbw`	Set the operational receive bandwidth.
`bitrate`	Set the FSK bit rate.
`fdev`	Set the frequency deviation allowed by the end device.
`prlen`	Set the preamble length used during transmissions.
`crc`	Set if a CRC header is to be used.
`iqi`	Set if IQ inversion is used.
`cr`	Set the coding rate used by the radio.
`wdt`	Set the time-out limit for the radio Watchdog Timer.
`sync`	Set the sync word used.
`bw`	Set the value used for the radio bandwidth.
`pa`	Set the PABOOST to use maximum power for radio operation.
`reg`	Set to write the given value to a chosen radio register.
`lbt`	Set the listen Before Talk parameters.

3.4.5.1 `radio set bt <gfBT>`

`<gfBT>`: String representing the Gaussian baseband data shaping, enabling GFSK modulation. Parameter values can be: `none, 1.0, 0.5, 0.3`.

Response:

- `ok` if the data shaping is valid.
- `invalid_param` if the data shaping is not valid.

This command modifies the data shaping applied to FSK transmissions. Entering any `<gfBT>` other than none will result in a Gaussian Filter BT being applied to transmissions in FSK mode. For example: `radio set bt none` // Data shaping in FSK mode is disabled or null.

3.4.5.2 radio set mod <mode>

<mode>: String representing the modulation method, either `lora` or `fsk`.

Response:

- `ok` if the modulation is valid
- `invalid_param` if the modulation is not valid

This command changes the modulation method being used by the SiP. Altering the mode of operation does not affect previously set parameters, variables or registers. FSK mode also allows GFSK transmissions when data shaping is enabled. For example: `radio set mod lora`.

3.4.5.3 radio set freq <frequency>

<frequency>: Decimal representing the frequency, from 137000000 to 175000000 or from 410000000 to 525000000 or from 862000000 to 1020000000, in Hz.

Response:

- `ok` if the frequency is valid.
- `invalid_param` if the frequency is not valid.

This command changes the communication frequency of the radio transceiver. For example: `radio set freq 868000000`.

3.4.5.4 radio set pwr <pwrout>

<pwrOut>: Signed decimal number representing the transceiver output power, from 2 to 20 or -4 to 15 depending on paboost “On” or “Off” state, respectively.

Response:

- `ok` if the output power is valid.
- `invalid_param` if the output power is not valid.

This command changes the transceiver output power. It is possible to set the output power above the regulatory limits. This power setting allows some compensation on the cable or transmission line loss. For more details on output power, refer to the *SAM R34/R35 Low Power LoRa® Sub-GHz SiP Datasheet (DS70005356)* and *WLR089U0 Low Power LoRa® Sub-GHz Module Datasheet (DS70005435A)*.

Example:

```
radio set pwr 14
```

Note: If PABOOST is turned on by using the command `radio set pa on`, the expected power output is mentioned below.

Table 3-6. Output Power when `radio set pa on`

TX Power Setting	Output Power (dBm)
2	3.0
3	4.0
4	5.0
5	6.0
6	7.0
7	8.0
8	9.0
9	10.0
10	11.0
11	12.0
12	13.0
14	14.7
15	15.5
16	16.3
17	17.0
20	18.5

Note: If PABOOST is turned off by using the command `radio set pa off`, the expected power output is mentioned below.

Table 3-7. Output Power when `radio set pa off`

TX Power Setting	Output Power (dBm)
-4	-4.9
-3	-4.0
-2	-2.9
-1	-1.9
0	-1.7
1	-0.6
2	0.4
3	1.4
4	2.5
5	3.6
6	4.7
7	5.8
8	6.9
9	8.1
10	9.3
11	10.4
12	11.6
13	12.5
14	13.5
15	14.1

3.4.5.5 `radio set sf <spreadingFactor>`

`<spreadingFactor>`: String representing the spreading factor. The parameter values can be:

`sf7, sf8, sf9, sf10, sf11` or `sf12`.

Response:

- `ok` if the spreading factor is valid.
- `invalid_param` if the spreading factor is not valid.

This command sets the spreading factor used during transmission. For example, `radio set sf sf7`.

3.4.5.6 `radio set afcbw <autoFreqBand>`

`<autoFreqBand>`: Float representing the automatic frequency correction in kHz. The parameter values can be: 250, 125, 62.5, 31.3, 15.6, 7.8, 3.9, 200, 100, 50, 25, 12.5, 6.3, 3.1, 166.7, 83.3, 41.7, 20.8, 10.4, 5.2, 2.6.

Response:

- `ok` if the automatic frequency correction is valid
- `invalid_param` if the automatic frequency correction is not valid

This command modifies the automatic frequency correction bandwidth for receiving/transmitting. For example, `radio set afcbw 125`.

3.4.5.7 radio set rxbw <rxbandwidth>

<rxBandwidth>: Float representing the signal bandwidth in kHz. Parameter values can be: 250, 125, 62.5, 31.3, 15.6, 7.8, 3.9, 200, 100, 50, 25, 12.5, 6.3, 3.1, 166.7, 83.3, 41.7, 20.8, 10.4, 5.2, 2.6.

Response:

- ok if the signal bandwidth is valid
- invalid_param if signal bandwidth is not valid

This command sets the signal bandwidth when receiving. For example, radio set rxbw 250 // Signal bandwidth for receiving is 250 kHz.

3.4.5.8 radio set bitrate <fskBitRate>

<fskBitRate>: Decimal number representing the FSK bit rate value, from 1 to 300000.

Response:

- ok if the bit rate value is valid
- invalid_param if the bit rate value is not valid

This command sets the FSK bit rate value. For example, radio set bitrate 5000 // FSK bit rate is set to 5 kbps.

3.4.5.9 radio set fdev <freqdev>

<freqDev>: Decimal number representing the frequency deviation, from 0 to 200000.

Response:

- ok if the frequency deviation is valid
- invalid_param if the frequency deviation is not valid

This command sets the frequency deviation during the operation. For example, radio set fdev 5000 // Frequency deviation is 5 kHz.

3.4.5.10 radio set prlen <preamble>

<preamble>: Decimal number representing the preamble length, from 0 to 65535.

Response:

- ok if the preamble length is valid
- invalid_param if the preamble length is not valid

This command sets the preamble length for transmit/receive. For example, radio set prlen 8 // Preamble length is 8.

3.4.5.11 radio set crc <crcHeader>

<crcHeader>: String representing the state of the CRC header, either on or off.

Response:

- ok if the state is valid
- invalid_param if the state is not valid

This command enables or disables the CRC header for communications. For example, radio set crc on // Enables the CRC header.

3.4.5.12 radio set iqi <iqInvert>

<iqInvert>: String representing the state of the invert IQ, either on or off.

Response:

- ok if the state is valid
- invalid_param if the state is not valid

This command enables or disables the Invert IQ for communications. For example, radio set iqi on // Invert IQ is enabled.

3.4.5.13 radio set cr <codingRate>

<codingRate>: String representing the coding rate. Parameter values can be: 4/5, 4/6, 4/7, 4/8.

Response:

- `ok` if the coding rate is valid
- `invalid_param` if the coding rate is not valid

This command modifies the coding rate currently being used by the radio. For example, `radio set cr 4/7` // The coding rate is set to 4/7.

3.4.5.14 radio set wdt <watchDog>

`<watchDog>`: Decimal number representing the time-out length for the Watchdog Timer, from 0 to 4294967295. Set to '`0`' to disable this functionality.

Response:

- `ok` if the Watchdog time-out is valid
- `invalid_param` if the Watchdog time-out is not valid

This command updates the time-out length in milliseconds applied, to the radio Watchdog Timer. If this functionality is enabled, then the Watchdog Timer is started for every transceiver reception or transmission. The Watchdog Timer is stopped when the operation in progress in finished. For example, `radio set wdt 2000` // The Watchdog Timer is configured for 2000 ms.

Note: Ensure the value configured for the Watchdog Timer matches the radio configurations. For example, set the `<watchDog>` value to '`0`' to disable this functionality during the radio continuous reception.

3.4.5.15 radio set sync <syncWord>

`<syncWord>`: Hexadecimal value representing the Sync word used during communication. For LoRa modulation, one byte is used, for FSK, up to eight bytes can be entered.

Response:

- `ok` if the sync word is valid
- `invalid_param` if the sync word is not valid

This command configures the sync word used during communication. For example, `radio set sync 12` // LoRa modulation in use.

3.4.5.16 radio set bw <bandWidth>

`<bandWidth>`: Decimal representing the operating radio bandwidth in kHz. Parameter values can be: 125, 250, 500.

Response:

- `ok` if the bandwidth is valid
- `invalid_param` if the bandwidth is not valid

This command sets the operating radio bandwidth for LoRa operation. For example, `radio set bw 250` // The operating bandwidth is 250 kHz.

3.4.5.17 radio set pa <paboost>

`<paboost>`: String represents the state of the PABOOST, either on or off.

Response:

- `ok` if the state is valid
- `invalid_param` if the state is not valid

This command enables the PABOOST to use maximum power for radio operation. For example, `radio set pa on` // Enables the PABOOST.

3.4.5.18 radio set reg <regAddr> <regValue>

`<regAddr>`: Hexadecimal value representing the address of the radio register.

`<regValue>`: Hexadecimal value representing the value to be written to `regAddr`.

Response:

- `ok` if the parameters are valid
- `invalid_param` if the `regaddr` & `regvalue` is not valid

This command writes the given value to a chosen radio register. For example: `radio set reg 02 05` // Sets the value `0x05` to a radio register `0x02`.

3.4.5.19 radio set lbt <ScanPeriod> <Threshold> <NumOfSamples> <TransmitOn>

- `<ScanPeriod>`: Decimal number representing the scan duration of a single channel
- `<Threshold>`: Signed decimal number representing the threshold above which channel is assumed to be occupied
- `<NumOfSamples>`: Decimal number representing the number of RSSI read samples for a single channel
- `<TransmitOn>`: Bool value for radio to decide if the transmit request is LBT based

Response:

- `ok` if the parameter is valid.
- `invalid_param` if the parameter is not valid.

Example:

```
radio set lbt 5 -90 10 1
ok
```

Note: The `radio set lbt` command is supported only for KR920 and JP920 channel plans.

3.4.6 Radio Get Commands

Table 3-8. Radio Get Commands

Parameter	Description
bt	Get the data shaping for the Frequency Shift Keying (FSK) modulation type.
mod	Get the SiP Modulation mode.
freq	Get the current operation frequency for the radio.
pwr	Get the output power level used by the radio during transmission.
sf	Get the requested Spreading Factor (SF) to be used during transmission.
afcbw	Get the value used by the automatic frequency correction bandwidth.
rxbw	Get the operational receive bandwidth.
bitrate	Get the FSK bit rate.
fdev	Get the frequency deviation allowed by the end device.
prlen	Get the preamble length used during transmissions.
crc	Get if a CRC header is to be used.
iqi	Get if an IQ inversion is used.
cr	Get the coding rate used by the radio.
wdt	Get the time-out limit for the Watchdog Timer.
bw	Get the value used for the radio bandwidth.
snr	Get the signal noise ratio (SNR) of the last received packet.
sync	Returns the current synchronization word for the radio.
pa	Get the status of PABOOST.
reg	Get the data address from the particular radio register.
regdump	Get the set of register value from mentioned starting and ending register address.

.....continued	
Parameter	**Description**
pktrssi	Get the RSSI value of last received packet while in LoRa modulation.
lbt	Gets the listen Before Talk parameters.

3.4.6.1 radio get bt

Response: String representing the configuration for data shaping. Parameter values can be: `none, 1.0, 0.5, 0.3`. This command reads back the current configuration for data shaping applied to FSK transmissions. For example, `radio get bt` // Reads the current data shaping FSK configuration. The default value is 0.5.

3.4.6.2 radio get mod

Response: String representing the current mode of operation of the SiP, either `lora` or `fsk`. This command reads back the current mode of operation of the SiP. For example, `radio get mod` // Reads if SiP is modulating in LoRa or FSK. The default mode is LoRa.

3.4.6.3 radio get freq

Response: Decimal representing the frequency, from 137000000 to 175000000 or from 410000000 to 525000000 or from 862000000 to 1020000000, in Hz. This command reads back the current operation frequency of the module. For example, `radio get freq` // Reads back the current frequency the transceiver communicates on. The default frequency is 868100000.

3.4.6.4 radio get pwr

Response: Signed decimal representing the current power level, from 2 to 20. This command reads back the current power level settings used in operation. For example, `radio get pwr` // Reads back the current transmit output power. The default power level is 1.

3.4.6.5 radio get sf

Response: String representing the current spreading factor. This command reads back the current spreading factor being used by the transceiver. Parameter values can be: `sf7, sf8, sf9, sf10, sf11, sf12`. For example, `radio get sf` // Reads back the current spreading factor settings. The default factor is `sf7`.

3.4.6.6 radio get afcbw

Response: Float representing the automatic frequency correction band in kHz. Parameter values can be: 250, 125, 62.5, 31.3, 15.6, 7.8, 3.9, 200, 100, 50, 25, 12.5, 6.3, 3.1, 166.7, 83.3, 41.7, 20.8, 10.4, 5.2, 2.6. This command reads back the status of the Automatic Frequency Correction Bandwidth. For example, `radio get afcbw` // Reads back the current automatic frequency correction bandwidth. The default band is 41.7.

3.4.6.7 radio get rxbw

Response: Float representing the signal bandwidth in kHz. Parameter values can be: 250, 125, 62.5, 31.3, 15.6, 7.8, 3.9, 200, 100, 50, 25, 12.5, 6.3, 3.1, 166.7, 83.3, 41.7, 20.8, 10.4, 5.2, 2.6. This command reads back the signal bandwidth used for receiving. For example, `radio get rxbw` // Reads back the receive signal bandwidth. The default bandwidth is 25.

3.4.6.8 radio get bitrate

Response: Signed decimal representing the configured bit rate, from 1 to 300000. This command reads back the configured bit rate for FSK communications. For example, `radio get bitrate` // Reads back the current FSK bit rate setting. The default bit rate is 50000.

3.4.6.9 radio get fdev

Response: Signed decimal representing the frequency deviation setting, from 0 to 200000. This command reads the frequency deviation setting on the transceiver. For example, `radio get fdev` // Reads back the current configured frequency deviation setting. The default value is 25000.

3.4.6.10 radio get prlen

Response: Signed decimal representing the preamble length, from 0 to 65535. This command reads the current preamble length used for communication. For example, `radio get prlen` // Reads back the preamble length used by the transceiver. The default length is 8.

3.4.6.11 radio get crc

Response: String representing the status of the CRC header, either `on` or `off`. This command reads back the status of the CRC header, to determine if it is to be included during operation. For example, `radio get crc` // Reads back if the CRC header is enabled for use. The default status is `on`.

3.4.6.12 radio get iqi

Response: String representing the status of the Invert IQ functionality, either `on` or `off`. This command reads back the status of the Invert IQ functionality. For example, `radio get iqi` // Reads back the status of the Invert IQ functionality. The default status is `off`.

3.4.6.13 radio get cr

Response: String representing the current value settings used for the coding rate. Parameter values can be: `4/5, 4/6, 4/7, 4/8`. This command reads back the current value settings used for the coding rate during communication. For example, `radio get cr` // Reads back the current coding rate transceiver settings. The default value is `4/5`.

3.4.6.14 radio get wdt

Response: Decimal number representing the length used for the Watchdog time-out, from 0 to 4294967295. This command reads back in milliseconds, the length used for the Watchdog time-out. For example, `radio get wdt` // Reads back the current time-out value applied to the Watchdog Timer. The default value is 15000.

3.4.6.15 radio get bw

Response: Decimal representing the current operating radio bandwidth in kHz. Parameter values can be: 125, 250 or 500. This command reads back the current operating radio bandwidth used by the transceiver. For example, `radio get bw` // Reads back the current operational bandwidth applied to transmissions. The default bandwidth is 125.

3.4.6.16 radio get snr

Response: Signed decimal number representing the signal to noise ratio (SNR), from -128 to 127. This command reads back the Signal Noise Radio (SNR) for the last received packet. For example, `radio get snr` // Reads back the measured SNR for the previous packet reception. The default value is -128.

3.4.6.17 radio get sync

Response: Up to 8-byte hexadecimal number representing the synchronization word. This command reads back the current synchronization word for the radio, depending on the modulation method set by the radio set mod <mode> command. For example, `radio get sync` // Reads back the current synchronization word. The default value is 34.

3.4.6.18 radio get pa

Response: String representing the status of the PABOOST, either `on` or `off`. This command reads back the status of the PABOOST to determine if it is to be included during operation. For example, `radio get pa` // Reads back if the PABOOST is enabled for use. The default status is `off`.

3.4.6.19 radio get reg <regAddr>

Response: Hexadecimal value representing the address of the radio register. This command returns the data from the particular radio register. For example, `radio get reg 10` // Reads back the content of the radio register, which is at address 10.

3.4.6.20 radio get regdump <regAddrStart> <regAddrEnd>

`<regAddrStart>`: Hexadecimal value representing the Starting address of the radio register to be read.

`<regAddrEnd>`: Hexadecimal value representing the End address of the radio register to be read.

Response: Sequence of hexadecimal values read from the radio registers from the start value to the end. This command returns the set of register values from the starting address contained in regAddrStart to the ending address contained in regAddrEnd. For example,

```
radio get regdump 00 05 // Returns the register content from address 0x00 to 0x05
Register Address: 0x0 Value: 0x0
Register Address: 0x1 Value: 0x88
Register Address: 0x2 Value: 0x1a
Register Address: 0x3 Value: 0xb
Register Address: 0x4 Value: 0x0
Register Address: 0x5 Value: 0x52
```

Note: The register address must be given as hexadecimal value.

3.4.6.21 radio get pktrssi

Response: Signed decimal number representing the rssi value of the last received packet using LoRa modulation. This command reads back the RSSI value of the last received packet while in LoRa modulation. For example, `radio get pktrssi` // Reads back rssi value of the last received packet. The default value is 0 – If transceiver is in FSK modulation, this value will be 0.

3.4.6.22 radio get lbt

Response:

```
radio get lbt

<ScanPeriod> <Threshold> <NumOfSamples> <TransmitOn>
```

- `<ScanPeriod>`: Decimal number representing the scan duration of a single channel. The default value is 0.
- `<Threshold>`: Signed decimal number representing the threshold above which channel is assumed to be occupied. The default value is 0.
- `<NumOfSamples>`: Decimal number representing the number of RSSI read samples for a single channel. The default value is 0.
- `<TransmitOn>`: Bool value for the radio to decide if the transmit request is LBT based. The default value is 0.

Example:

```
radio get lbt

5 -90 5 1
```

4. Document Revision History

Revision	Date	Section	Description
B	09/2020	Introduction	Updated the section
		3.3 System Commands	• Added `sys get ver` and `sys factoryRESET` commands. • Updated the following commands: – `sys reset` – `sys sleep` – `sys get ver`
		3.4 Radio Commands	Updated *Radio Parameters Availability for Different Operations* table
		3.4.4 radio cw <state>	Remove Note
		3.4.5 Radio Set Commands	• Updated the transceiver output power of `radio set pwr` command • Updated the default current spreading factor of `radio get sf` command • Added `radio set lbt` command
		3.4.6 Radio Get Commands	• Updated the default power level of `radio get pwr` command • Updated the default current spreading factor of `radio get sf` command • Added `radio get lbt` command
		Document	Added references about WLR089U0 throughout the document
A	10/2018	Document	Initial Revision

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods being used in attempts to breach the code protection features of the Microchip devices. We believe that these methods require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Attempts to breach these code protection features, most likely, cannot be accomplished without violating Microchip's intellectual property rights.
- Microchip is willing to work with any customer who is concerned about the integrity of its code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable." Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Trademarks

ISBN: 978-1-5224-6712-0

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: www.microchip.com/support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4485-5910 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-72400 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820