# Command List

**SEONGJI**

**Semptember 10, 2019**

## List

## Command

### MODEL and Firmware Ver.

| MODEL | Firmware |
|---|---|
| SFM20R | EVBSFM20R_V300 |
| SRM200 | EVBSRM200_V300 |
| | |
| | |

## SIGFOX command

| Command | Name | Description |
|---|---|---|
| AT | Dummy Command | Just returns 'OK' and does nothing else. Can be used to check communication. |
| AT$SB=bit[,bit] | Send Bit | Send a bit status (0 or 1). Optional bit flag indicates if AX−SFEU should receive a downlink frame. |
| AT$SF=frame[,bit] | Send Frame | Send payload data, 1 to 12 bytes. Optional bit flag indicates if AX−SFEU should receive a downlink frame. |
| AT$SO | Manually send out of band message | Send the out−of−band message. |
| AT$TR? | Get the transmit repeat | Returns the number of transmit repeats. Default: 2 |
| AT$TR=? | Get transmit range | Returns the allowed range of transmit repeats. |
| AT$TR=uint | Get transmit repeat | Sets the transmit repeat. |
| ATSuint? | Get Register | Query a specific configuration register's value. See chapter "Registers" for a list of registers. |
| ATSuint=uint | Set Register | Change a configuration register. |
| ATSuint=? | Get Register Range | Returns the allowed range of transmit repeats. |
| AT$IF=uint | Set TX Frequency | Set the output carrier macro channel for Sigfox frames. |
| AT$IF? | Get TX Frequency | Get the currently chosen TX frequency. |
| AT$DR=uint | Set RX Frequency | Set the reception carrier macro channel for Sigfox frames. |
| AT$DR? | Get RX Frequency | Get the currently chosen RX frequency. |
| AT$CW=uint,bit[,uint_opt] | Continuous Wave | To run emission tests for Sigfox certification it is necessary to send a continuous wave, i.e. just the base frequency without any modulation. Parameters:<br>**Name** / **Range** / **Description**<br>Frequency / 800000000−999999999, 0 / Continuous wave frequency in Hz. Use 868130000 for Sigfox or 0 to keep previous frequency.<br>Mode / 0, 1 / Enable or disable carrier wave.<br>Power / 0−14 / dBm of signal \| Default: 14 |
| AT$CB=uint_opt,bit | Test Mode:<br>TX constant byte | For emission testing it is useful to send a specific bit pattern. The first parameter specifies the byte to send. Use '−1' for a (pseudo−)random pattern. Parameters:<br>**Name** / **Range** / **Decsription**<br>Pattern / 0−255, −1 / Byte to send. Use '−1' for a (pseudo−)random pattern.<br>Mode / 0, 1 / Enable or disable pattern test mode. |
| AT$T? | Get Temperature | Measure internal temperature and return it in 1/10$^{th}$ of a degree Celsius. |
| AT$V? | Get Voltages | Return current voltage and voltage measured during the last transmission in mV. |

| Command | Name | Description |
|---|---|---|
| AT$I=uint | Information | Display various product information:<br>0: Software Name & Version<br>Example Response: AX−SFEU 1.0.6−ETSI<br>1: Contact Details<br>Example Response: support@axsem.com<br>2: Silicon revision lower byte<br>Example Response: 8F<br>3: Silicon revision upper byte<br>Example Response: 00<br>4: Major Firmware Version<br>Example Response: 1<br>5: Minor Firmware Version<br>Example Response: 0<br>7: Firmware Variant (Frequency Band etc. (EU/US))<br>Example Response: ETSI<br>8: Firmware VCS Version<br>Example Response: v1.0.2−36<br>9: SIGFOX Library Version<br>Example Response: DL0−1.4<br>10: Device ID<br>Example Response: 00012345<br>11: PAC<br>Example Response: 0123456789ABCDEF |
| AT$P=uint | Set Power Mode | To conserve power, the AX−SFEU can be put to sleep manually. Depending on power mode, you will be responsible for waking up the AX−SFEU again!<br>0: software reset (settings will be reset to values in flash)<br>1: sleep (send a break to wake up)<br>2: deep sleep (toggle GPIO9 or RESET_N pin to wake up; the AX−SFEU is not running and all settings will be reset!) |
| AT$WR | Save Config | Write all settings to flash (RX/TX frequencies, registers) so they survive reset/deep sleep or loss of power.<br>Use AT$P=0 to reset the AX−SFEU and load settings from flash. |
| AT$TM=mode,config | Activates the Sigfox Testmode | Available test modes:<br>0. TX BPSK<br>Send only BPSK with Synchro Bit + Synchro frame + PN sequence: No hopping centered on the TX_frequency. Config bits 0 to 6 define the number of repetitions. Bit 7 of config defines if a delay is applied of not in the loop<br>1. TX Protocol:<br>Tx mode with full protocol with Sigfox key: Send Sigfox protocol frames with initiate downlink flag = True. Config defines the number of repetitions.<br>2. RX Protocol:<br>This mode tests the complete downlink protocol in Downlink only. Config defines the number of repetitions.<br>3. RX GFSK:<br>RX mode with known pattern with SB + SF + Pattern on RX_frequency (internal comparison with received frame ⇔ known pattern = AA AA B2 27 1F 20 41 84 32 68 C5 BA AE 79 E7 F6 DD 9B. Config defines the number of repetitions. Config defines the number of repetitions.<br>4. RX Sensitivity:<br>Does uplink + downlink frame with Sigfox key and specific timings. This test is specific to SIGFOX's test equipments & softwares.<br>5. TX Synthesis:<br>Does one uplink frame on each Sigfox channel to measure frequency synthesis step |
| AT$SE | Starts AT$TM−3,255 indefinitely | Convenience command for sensitivity tests |
| | | |

## WIFI command

| Command | Name | Description |
|---|---|---|
| AT+CWLAP | Lists available APs | <ssid> string, SSID of AP<br><rssi> signal strength<br><mac> string MAC address |
| | | |

## Sigfox Monarch Command

| Command | Example | Description |
|---|---|---|
| help | help<CR><LF> | display command list |
| node_get_version | node_get_version<CR><LF> | |
| node_open_with_zone | node_open_with_zone 1<CR><LF> | u<br>open the library in the selected RC Zone. 1=RC1, 2=RC2, 3=RC3c, 4=RC4, 5=RC5, 6=RC6<br>Argment 1 : zone (1~6) |
| node_close | node_close<CR><LF> | close the library |
| node_send_frame | node_send_frame "12345678" 2 0<CR><LF><br>node_send_frame {0102030405060708} 2 0<CR><LF> | buu<br>send a frame<br>Argment 1 : data (ASCII or Hex)<br>Argment 2 : tx repeat<br>Argment 3 : rx flag |
| node_execute_monarch_scan | node_execute_monarch_scan 0x3F 5 2<CR><LF> | uuu<br>Execute Monarch scan<br>Argment 1 : rc_capability (mask)<br>bit 5 4 3 2 1 0<br>rc 6 5 4 3 2 1<br>Argment 2 : time<br>Argment 3 :<br>unit(0=ms,1=s,2=min, 3=hour) |
| node_set_std_config | node_set_std_config 1 0 0 1<CR><LF> | wwwv<br>Argment 1 : channel mask (1~32)<br>Argment 2 : channel mask (33~64)<br>Argment 3 : channel mask (65~86)<br>Argment 4 : timer enable (0,1) |
| get_id | get_id<CR><LF> | Get id |
| get_pac | get_pac<CR><LF> | Get pac. |
| _set_rcz | _set_rcz 1<CR><LF> | u<br>Set rc zone<br>Argment 1 : zone (1~6) |
| switch_public_key | switch_public_key 1<CR><LF> | u<br>Argment 1 : flag (0,1) |
| switch_test_credentials | switch_test_credentials 1<CR><LF> | u<br>Set test credentials<br>Argment 1 : flag (0,1) |
| node_test_mode | node_test_mode 0 1<CR><LF> | uu<br>enter a specified test mode<br>Argment 1 : zone (1~6)<br>Argment 2 : test mode<br>0:TX_BPSK<br>1:TX_PROTOCOL<br>2:RX_PROTOCOL<br>3:RX_GFSK<br>4:RX_SENSI<br>5:TX_SYNTH<br>6:TX_FREQ_DISTRIBUTION<br>11:TX_BIT<br>12:PUBLIC_KEY<br>13:NVM |

| | | |
|---|---|---|
| node_monarch_test_mode | node_monarch_test_mode 0 7 63<CR><LF> | uuu<br>Argment 1 : zone (1~6)<br>Argment 2 : test mode<br>7:RX_MONARCH_PATTERN_LISTENING_SWEEP<br>8:RX_MONARCH_PATTERN_LISTENING_WINDOW<br>9:RX_MONARCH_BEACON<br>10:RX_MONARCH_SENSI<br>Argment 3 : rc mask<br>bit 5 4 3 2 1 0<br>rc 6 5 4 3 2 1 |
| start_continuous_transmission | start_continuous_transmission 8681300000 0 0<CR><LF> | wuu<br>Argment 1 : freq<br>Argment 2 : mode(0: CW, 1: modulation)<br>Argment 3 : rc zone |
| stop_continuous_transmission | stop_continuous_transmission<CR><LF> | |
| switch_pa | switch_pa 1<CR><LF> | u<br>Turn Power Amplifier On/Off<br>Argment 1 : flag (0:off, 1:on) |
| | | |