Data sheet

Wireless UART firmware version 4.02

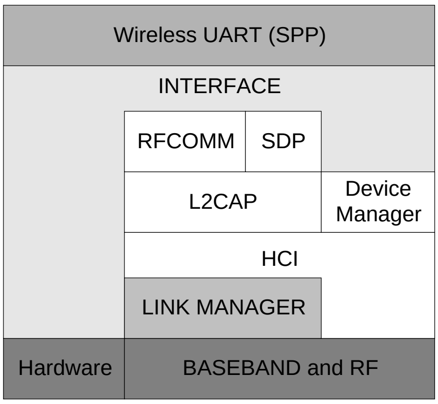
Rev: 22 December 2008

Table of contents

1 General Information

Free2move's Wireless UART (WU) firmware is intended to replace the cable(s) connecting portable and/or fixed electronic devices. Key features are robustness, high configurability, high security, low complexity and low power.

The WU firmware is compliant with the Bluetooth Serial Port Profile (SPP) for setting up emulated serial cable connections between connected devices. There is no additional need for drivers or an external host with Bluetooth software when using the WU firmware.



Wireless UART architecture

The WU application runs on top of an embedded Bluetooth v2.0 + EDR compliant stack, including protocols up to the RFCOMM layer. Point-to-point connections are supported. This means that a unit running WU can be either a master of one slave unit or participate in a piconet as a slave. Up to 7 slaves can be active in a piconet.

The WU firmware offers one asynchronous data channel and one synchronous voice channel, both channels capable of full duplex transmissions.

When a successful Bluetooth connection is established the data channel and the voice channel can be used simultaneously or separately. All information sent/received at the data/voice interface of the WU unit is exchanged transparently via Bluetooth with the connected remote device.

The WU unit is set to operate in a default mode that allows the user to communicate via the asynchronous data channel over Bluetooth, as soon as a successful connection has been established. This can be achieved without sending any configuration commands to the WU firmware. However, as long as there is no Bluetooth connection established, it is possible to configure the WU firmware via commands (described in this document) sent on the data interface.

1.1 Supported hardware

The WU v4.xx firmware is currently supported on the following Free2move products:

- F2M03GX – General purpose, extended range module with U.FL-connector (Class 1)
- F2M03GXA – General purpose, extended range module with antenna (Class 1)
- F2M03GLA – General purpose, low power module with antenna (Class 1)
- F2M03MLA – Multimedia, Low power module with Antenna. (Class1)
- F2M03ALA – Audio, Low power module with Antenna. (Class2)
- F2M01SXA – Serial Port Plug Gen 2 (Class1)

1.2 General I/O

General I/O interfaces are used for different purposes between the WU firmware and the Host:

- Asynchronous data interface – configuration of the WU firmware or exchange transparent digital information between the connected Bluetooth devices.
- Synchronous voice interface – exchange transparent voice information between the connected Bluetooth devices.
- Bluetooth connectivity PIO interface – indication and disconnection of the established Bluetooth connection.
- Emulate serial handshaking PIO lines interface – DTE or DCE serial handshake emulation between the connected Bluetooth devices.

1.2.1 Asynchronous data interface

The WU firmware uses a UART interface to the Host for communication. When no Bluetooth connection is established the interface can be used by the Host to configure the WU firmware. When a Bluetooth connection is established the interface is used to exchange transparent digital information between the Host and the connected remote Bluetooth device.

UART interface:

UART I/O	Signal Direction	F2M01 - Active (RS-232)	F2M03 - Active (TTL)	Description
TX	Output	Low	High	UART transmit data
RX	Input	Low	High	UART receive data
RTS	Output	High	Low	UART request to send
CTS	Input	High	Low	UART clear to send

1.2.2 Synchronous voice interface

The voice information exchanged between the connected Bluetooth devices is routed directly via the hardware of the WU unit (F2M03 only). Depending on which F2M03 product (table below) being used the information is routed over different hardware interfaces. It could either be routed over the Pulse Code Modulation (PCM) interface, for F2M03 products that require an external CODEC, or directly over the microphone and speaker interface, for F2M03 products with internal CODEC.

Audio interface:

Free2move Product	Hardware Interface
F2M03GX	PCM
F2M03GXA	PCM
F2M03GLA	PCM
F2M03MLA	Internal Audio Codec
F2M03ALA	Internal Audio Codec
F2M01SXA	None

Free2move products using the PCM interface (see product datasheet for detailed information) are configured as Master of the PCM interface with the following settings:

PCM hardware interface:

PCM I/O	Signal Direction	Value	Description
PCM_IN	Input	13-bit linear 2s complement coding	Synchronous data input
PCM_OUT	Output	13-bit linear 2s complement coding	Synchronous data output
PCM_CLOCK	Output	256kHz	Synchronous data clock
PCM_SYNC	Output	8kHz long frame sync	Synchronous data sync

Recommended external CODEC is Winbond W681360R 13-bit linear CODEC.

1.2.3 Bluetooth connectivity PIO

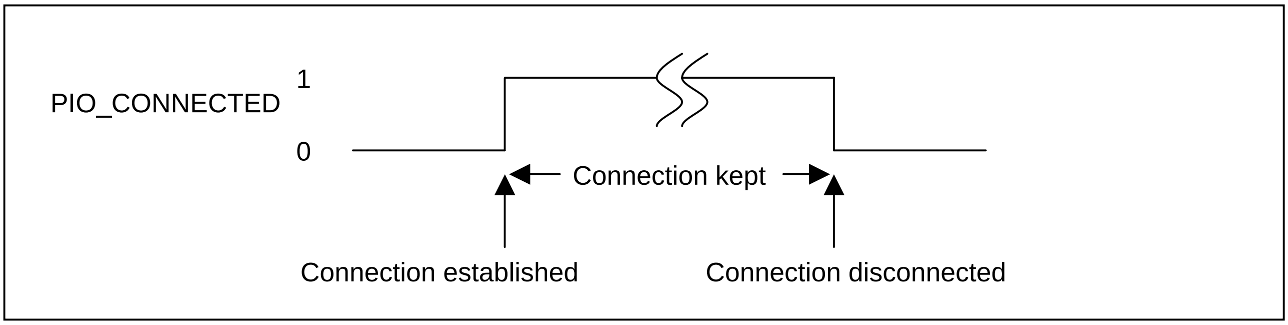
PIO's are used to control/monitor the Bluetooth connectivity of the WU firmware (F2M03 only).

Bluetooth connectivity PIO:

PIO	Signal Direction	Active (TTL)
PIO_DISCONNECT	Input	High
PIO_CONNECTED	Output	High

PIO	PIO F2M03GLA/GX/GXA	PIO F2M03ALA	PIO F2M03MLA
PIO_DISCONNECT	2	2	0
PIO_CONNECTED	3	3	1

PIO_CONNECTED is an output used to inform the Host about the current Bluetooth status, if there is a Bluetooth connection established or not (figure below). PIO_CONNECTED will go high as soon as a successful connection has been established and will stay high as long as the connection is kept. When the connection is disconnected PIO_CONNECTED will go low.



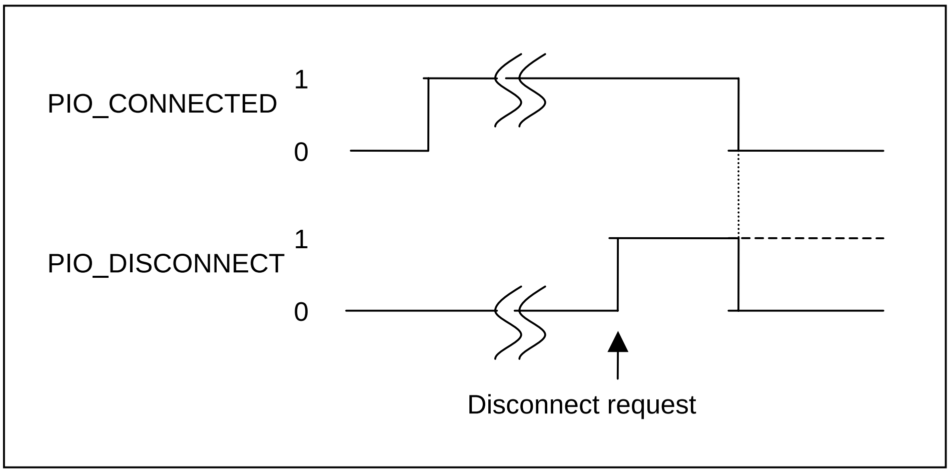
PIO_CONNECTED Bluetooth status

PIO_CONNECTED summary:

PIO_CONNECTED (TTL)	Description
Low	No Bluetooth connection established.
High	Bluetooth connection established with a remote device.

PIO_DISCONNECT is an input used by the Host to control the Bluetooth connectivity of the WU unit.

If there is a Bluetooth connection established the Host can disconnect the Bluetooth connection by setting PIO_DISCONNECT high (figure below). The WU firmware detects the change in level and issues a disconnect request. When the connection has been successfully closed down PIO_CONNECTED will be set low. The Host can now set PIO_DISCONNECT low again to allow a new Bluetooth connection to be established or keep PIO_DISCONNECT high to prevent a new connection establishment.



PIO_DISCONNECT disconnect request

Note: The time from PIO_DISCONNECT goes high until PIO_CONNECTED goes low is unpredictable. It depends on different factors such as CPU load of the connected devices and the Bluetooth link quality.

If there is no Bluetooth connection established the Host can prevent the establishment of any Bluetooth connection by setting PIO_DISCONNECT high. As long as PIO_DISCONNECT is held high no Bluetooth connection will be established and the RF parts in the hardware will be in idle mode.

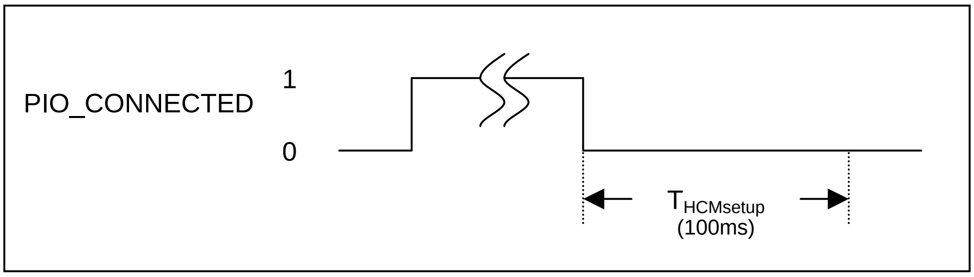
As soon as PIO_DISCONNECT goes low the RF activity will be switched back to its normal operating mode.

PIO_DISCONNECT summary:

PIO_DISCONNECT (TTL)	PIO_CONNECTED (TTL)	PIO_DISCONNECT Description
Low	Low	Held low will enable the RF activity.
High	Low	Held high will prevent the establishment of any Bluetooth connection. RF activity will enter idle mode.
Low	High	No effect
High	High	Held high will disconnect the current Bluetooth connection.

When attempting to enter *"Host Controlled Mode"* (HCM) it is preferable to set PIO_DISCONNECT high to prevent a Bluetooth connection from being established during the HCM attempt. Once entered HCM the WU unit will be discoverable for other Bluetooth devices but will not accept any connection requests as described in the *"Host Controlled Mode"* section.

Note: It is not possible to enter HCM immediately after the connection has been closed down (figure below). There is a setup time, $T_{HCMsetup}$, of at least 100ms, depending on the *"Host controlled mode configuration"*, before the WU unit is ready to accept HCM request command.



Setup time before HCM can be entered

1.2.4 Emulate serial handshaking PIO lines

PIO's are used to emulate serial handshaking lines between the connected Bluetooth devices. When a successful Bluetooth connection is established; DCD, DSR, RI and DTR signals are transferred between the connected units. Whenever a local PIO pin's level change the remote device will receive a notification. Emulation can either be DTE or DCE.

The F2M01SXA Serial Port Plug can act as a DCE device with DTR and DSR signals enabled in the D-Sub connector.

More detailed information can be found in the *"Emulate serial handshaking lines"* section under the *"Advanced Configuration Commands"* chapter.

Emulate serial handshaking PIO:

Emulated Signal	**Signal Direction Emulate DTE**	**Signal Direction Emulate DCE**	**Active (TTL)**
RI	Input	Output	High
DTR	Output	Input	High
DCD	Input	Output	High
DSR	Input	Output	High

Emulated Signal	**PIO F2M03G**	**PIO F2M03ALA**	**PIO F2M03MLA**	**D-Sub PIN F2M01SXA**
RI	4	4	8	N/A
DTR	5	5	9	5
DCD	6	6	6	N/A
DSR	7	7	7	7

1.3 Default settings

The WU units are shipped with the following default settings:

Parameter	Default Value
Operating mode	Endpoint (Bluetooth Slave)
Connect accept rule	Accepts all units
Security mode	Security off
Encryption mode	Encryption off
PIN code	“0000” in ASCII format
Local Bluetooth name	Free2move WU
Local SDP-service name	SerialPort
Baud rate	38400
Data bits	8
Parity	None
Stop bits	1
Hardware flow control	On
Override serial baud rate	Disabled
Throughput/latency mode	Throughput
Link supervision timeout	20sec
Inquiry scan interval	1.28sec
Inquiry scan window	11.25ms
Page scan interval	1.28sec
Page scan window	11.25ms
Low power mode	Disabled
Class of device	Uncategorized
Default transmit power	+4dBm
Maximum transmit power (Class1)	+20dBm
Maximum transmit power (Class2)	+4dBm
Connection established notification mode	Disabled
Connection closed notification mode	Disabled
Emulate serial handshaking lines	Disabled
Override serial port baud rate	Disabled
Quality of service setup	Disabled
Setup synchronous connection	Disabled
Remote RFCOMM server channel	Disabled

The default settings allow the user to communicate via Bluetooth, without sending any configuration commands, as soon as a successful connection has been established. Information sent and received on the serial interface of the WU unit at 38400 bps is transmitted transparently between the two connected devices.

These default settings are valid as long as no configuration has been made by the user.

1.4 Command interface

As long as there is no Bluetooth connection established, it is possible to configure the WU firmware via commands (described in this document) sent on the serial interface.

See *“Wireless UART message sequence chart examples”* section for detailed examples.

The following serial settings are used for configuration mode and are not configurable:

Parameter	Default Value
Baud rate	38400
Data bits	8
Parity	None
Stop bits	1
Hardware flow control	On

Note: The baud rate used for configuration is not related to the baud rate specified in the *“Serial port configuration”* command. As soon as a successful connection has been established the baud rate specified in the *“Serial port configuration”* will be used.

The commands are transferred as data packets. The different types of data packets that can be transferred over the serial interface are described in terms of a request, response or an indication.

- A request is always sent from the Host to the WU firmware. A request is always followed by a response.
- A response is always a response upon a request and cannot be achieved without sending a request first. A response is sent from the WU firmware to the Host.
- An indication is always sent from the WU firmware to the Host. An indication can be sent from the WU firmware at any time, indicating some changes that will affect the Host.

Note: When sending a request, the response must be received before sending a new request.

Note: If a parameter in the command consist of more than one byte, MSB must/will always be sent first, followed by the remaining bytes and LSB last.

All response/indication commands include a STATUS byte as the first command parameter (this description is not included for each command):

STATUS: Size: 1 byte

Value	Parameter Description
0x01	Command success
0xFC	RF not ready
0xFD	Command invalid parameter
0xFE	Command failed
0xFF	Command error

RF not ready (0xFC) will be the STATUS parameter value as long as the Bluetooth radio of the WU unit is busy and not able to perform the requested radio command i.e. inquiry or pairing. Entering *“Host Controlled Mode”* and *“Operating mode”* is Connecting unit, no radio activities can be made until the page timeout is finished. The default page timeout time is 5.12sec.

Common for all packets is that they start with one byte describing the command type; the next byte indicates the total length of the following command parameters (including the STATUS byte for response and indication commands).

Note: The WU firmware will not start processing the requested command until all the expected bytes have been received.

1.5 Performance

1.5.1 Throughput

The WU firmware is a complete on-chip application; limited resources restrict the maximum throughput. The table below shows the maximum achieved throughput when streaming 1MB of data between two connected WU v3.00 devices at close range.

Direction	Baud Rate	Maximum Throughput (kbit/s (throughput mode))	Maximum Throughput (kbit/s) (latency mode)
Master to Slave	57600	~57.6	~57.6
Slave to Master	57600	~57.6	~57.6
Full duplex	57600	~57.6	~50.5
Master to Slave	115200	~115.1	~93.9
Slave to Master	115200	~115.1	~79.6
Full duplex	115200	~114.5	~42.0
Master to Slave	230400	~223.1	~158.0
Slave to Master	230400	~221.4	~117.7
Full duplex	230400	~172.7	~86.2
Master to Slave	460800	~228.6	~206.7
Slave to Master	460800	~222.7	~154.1
Full duplex	460800	~173.3	~109.8
Master to Slave	921600	~240.1	~235.7
Slave to Master	921600	~235.4	~186.0
Full duplex	921600	~174.7	~150.5

The figures above was achieved during performance tests with the following serial interface:

Parameter	Default Value
Baud rate	57600 – 921600
Data bits	8
Parity	None
Stop bits	1
Hardware flow control	On

1.5.2 Latency

The latency depends on the amount of data that is transferred but it is around 15-30 ms one way.
The Wireless UART firmware can be configured to optimize transmissions for latency. The commands is described in section *“Throughput/latency mode”.*

2 Host Controlled Mode

Note: Before any commands can be sent to the WU firmware, it must be set in *Host Controlled Mode* (HCM). The WU firmware can only enter HCM when there is no Bluetooth connection established. Commands are accepted at the serial interface using the settings described in *"Command interface"* section above.

When no Bluetooth connection has been established and the WU firmware has not switched to HCM; all information sent on the serial interface to the WU unit will be parsed. If not correct *"Switch to Host controlled mode request (0x01)"* command is received the data will be discarded and no action is taken.

Once entered HCM there are several commands that can be issued:

- Configuration commands
- Information commands
- Software / Hardware reboot commands
- Inquiry commands (search for Bluetooth devices in the neighborhood)
- Pairing and security commands (device security – authentication and encryption)
- Control commands
- Advanced configuration commands
- Synchronous connection commands (voice connection establishment and CODEC gain control)

To exit HCM and apply the new settings the *"Run"* command in the *"Software / Hardware Reboot Commands"* section must be sent. All settings are stored in persistent memory.

See *"Wireless UART message sequence chart examples"* section for detailed examples.

2.1 Enter Host controlled mode

Before sending a *"Switch to Host controlled mode request (0x01)"* command the Host must ensure that the WU unit is ready to accept HCM request command. If the WU unit is not ready to accept a HCM request no HCM response will be received.

There is important timing that the Host must take into account to ensure that the WU unit is ready to accept HCM request command.

It is not possible to enter HCM immediately after the RESET of the WU unit is released. There is a setup time of 800ms before the WU unit is ready to accept HCM request command.

It is not possible to enter HCM immediately after the Bluetooth connection has been closed down. There is a setup time of at least 100ms, depending on which mode the WU unit is configured to, before the WU unit is ready to accept HCM request command. See the *"Host controlled mode configuration"* section for more information.

Important timing summary:

State	Setup time
Release of RESET	800 milliseconds
Bluetooth disconnected	100 milliseconds

To be able to enter HCM, the correct *"Switch to Host controlled mode request (0x01)"* command with the correct CODE must be sent. When the WU firmware has successfully changed operating mode to *Host Controlled Mode* a *"Switch to Host controlled mode response (0x01)"* command will be sent to the Host.

Parameter description:

CODE: Size: 4 bytes

Value	Parameter Description
0xFF, 0x00, 0x55, 0xAA	CODE to switch to HCM

Note: Once entered HCM the WU unit will be discoverable for other Bluetooth devices but will not accept any connection requests (inquiry scan will be enabled and page scan will be disabled by default).

These scan parameters are used in HCM:

Parameter	Value	Parameter Description
INQUIRY_SCAN_INTERVAL	0x0800	Inquiry scan interval time = 1.28sec
INQUIRY_SCAN_WINDOW	0x0012	Inquiry scan window time = 11.25msec
PAGE_SCAN_INTERVAL	0x0000	Page scan disabled
PAGE_SCAN_WINDOW	0x0000	Page scan disabled

2.1.1 Switch to Host controlled mode request (0x01)

This command requests that the WU firmware switches to HCM.

Command	Length	Command Parameter 1
0x01	0x04	CODE

2.1.2 Switch to Host controlled mode response (0x01)

This response command will be returned when the WU firmware has successfully changed operating mode to *Host Controlled Mode.*

Command	Length	Command Parameter 1
0x01	0x01	STATUS

2.2 Host controlled mode configuration

When the Bluetooth connection has been closed down the Host controlled mode configuration determines when the WU firmware is allowed to prepare for parsing received serial information and start looking for the *"Switch to Host controlled mode request (0x01)"* command.

There are two different modes:

- Immediately when the Bluetooth connection is closed
- After the Bluetooth connection has been closed and all buffered UART data has been sent out to the Host

If the WU firmware should prepare for HCM immediately, all buffered UART data will be discarded and the important timing described in the *"Bluetooth connectivity PIO"* and the *"Enter Host controlled mode"* section applies.

If the WU firmware should prepare for HCM after all buffered UART data has been sent out to the Host the same important timing also applies here. However the important timing is delayed while waiting for the last byte to be sent out to the Host before the WU firmware is able to continue with the program execution.
Note: This mode can cause a potential "dead-lock" situation when hardware flow control is enabled. If the Host does not allow the WU firmware to send out any data it will wait here forever until it has completed the task.

Parameter description:

MODE: Size: 1 byte

Value	Parameter Description
0x00	Immediately when the Bluetooth connection is closed
0x01	After the Bluetooth connection has been closed and all buffered UART data has been sent out to the Host
Default (F2M01) = 0x01	After the Bluetooth connection has been closed and all buffered UART data has been sent out to the Host
Default (F2M03) = 0x00	Immediately when the Bluetooth connection is closed

2.2.1 Read allowed to switch to Host controlled mode configuration request (0x02)

A request to read the allowed to switch to Host controlled mode configuration.

Command	Length	Command Parameter 1
0x02	0x00	-

2.2.2 Read allowed to switch to Host controlled mode configuration response (0x02)

If STATUS is success, the allowed to switch to Host controlled mode configuration is returned in the MODE parameter otherwise MODE will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x02	0x02	STATUS	MODE

2.2.3 Set allowed to switch to Host controlled mode configuration request (0x03)

A request to change the allowed to switch to Host controlled mode configuration.

Command	Length	Command Parameter 1
0x03	0x01	MODE

2.2.4 Set allowed to switch to Host controlled mode configuration response (0x03)

If STATUS is success, the new allowed to switch to Host controlled mode configuration has been stored in persistent memory.

Command	Length	Command Parameter 1
0x03	0x01	STATUS

2.3 ASCII controlled mode configuration

2.3.1 Switch to ASCII controlled mode request (0x04)

This command requests that the WU firmware switches to ASCII mode. For more information regarding ASCII- mode please see sction *ASCII controlled mode*.

Command	Length	Command Parameter 1
0x04	0x00	-

2.3.2 Switch to ASCII controlled mode response (0x04)

This response command will be returned when the WU firmware has successfully changed operating mode to ASCII Controlled Mode.

Command	Length	Command Parameter 1
0x04	0x01	STATUS

2.3.3 Read allowed to switch to ASCII controlled mode configuration request (0x05)

A request to read the allowed to switch to ASCII controlled mode configuration.

Command	Length	Command Parameter 1
0x05	0x00	-

2.3.4 Read allowed to switch to ASCII controlled mode configuration response (0x05)

If STATUS is success, the allowed to switch to ASCII controlled mode configuration is returned in the MODE parameter otherwise MODE will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x05	0x02	STATUS	MODE

2.3.5 Set allowed to switch to ASCII controlled mode configuration request (0x06)

A request to change the allowed to switch to Host controlled mode configuration.

Command	Length	Command Parameter 1
0x06	0x01	MODE

2.3.6 Set allowed to switch to ASCII controlled mode configuration response (0x06)

If STATUS is success, the new allowed to switch to ASCII controlled mode configuration has been stored in persistent memory.

Command	Length	Command Parameter 1
0x06	0x01	STATUS

3 Configuration Commands in HCM

Note: Before any configuration commands can be sent to the WU firmware, it must be set in *"Host Controlled Mode"*.

The basic settings of the WU firmware can be read and modified by using these configuration commands.

3.1 Profile mode

The Wireless UART firmware supports either the Serial Port Profile(SPP) or a limited support of Dial-Up Networking(DUN) Profile.

The DUN Profile is based on SPP but with the correct Service Discovery Parameters set for the DUN profile. The AT control interface is up to the Host Processor to implement. For more information regarding the DUN profile please read the specification available at http://www.bluetooth.com/.

Parameter description:

PROFILE_MODE: Size: 1 byte

Value	Parameter Description
0x00	SPP Mode (Default)
0x01	DUN Mode

3.1.1 Read profile mode request (0x10)

A request to read the normal operating mode.

Command	Length	Command Parameter 1
0x10	0x00	-

3.1.2 Read profile mode response (0x10)

If STATUS is success, the normal profile mode is returned in the PROFILE_MODE parameter.

Command	Length	Command Parameter 1	Command Parameter 2
0x10	0x02	STATUS	PROFILE_MODE

3.1.3 Set profile mode request (0x11)

A request to change the profile mode.

Command	Length	Command Parameter 1
0x11	0x01	PROFILE_MODE

Set profile mode response (0x11)

If STATUS is success, the new profile mode has been stored in persistent memory.

Command	Length	Command Parameter 1
0x11	0x01	STATUS

3.2 Operating mode

There are two normal operating modes:

- Connecting mode – Bluetooth master
- Endpoint mode – Bluetooth slave

In Connecting mode the WU firmware will try to establish connections to other Bluetooth devices in the neighborhood. This is controlled by the *“Connect rule”*.

In Endpoint mode the WU firmware will accept connections according to the *“Connect accept rule”*.

Parameter description:

OPERATING_MODE: Size: 1 byte

Value	Parameter Description
0x01	Connecting Mode
0x02	Endpoint Mode

3.2.1 Read operating mode request (0x12)

A request to read the normal operating mode.

Command	Length	Command Parameter 1
0x12	0x00	-

3.2.2 Read operating mode response (0x12)

If STATUS is success, the normal operating mode is returned in the OPERATING_MODE parameter otherwise OPERATING_MODE will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x12	0x02	STATUS	OPERATING_MODE

3.2.3 Set operating mode request (0x13)

A request to change the normal operating mode.

Command	Length	Command Parameter 1
0x13	0x01	OPERATING_MODE

3.2.4 Set operating mode response (0x13)

If STATUS is success, the new operating mode has been stored in persistent memory.

Command	Length	Command Parameter 1
0x13	0x01	STATUS

3.3 Serial port configuration

The configurable serial port settings are used as soon as a successful Bluetooth connection has been established. Information is sent and received with the specified UART settings.

Standard baud rates from 2400-3686400bps can be used. If a non-standard baud rate must be used, the BAUD_RATE parameter can be overridden with the *“Override serial port configuration baud rate”* commands in the *“Advanced Configuration Commands”* section.

For the F2M03 modules hardware flow control is always enabled and it is not configurable.

Even, odd or no parity can be chosen also 1 or 2 stop bits can be used. 8 data bits are always used for the data and are not configurable.

Note: The serial port settings of the WU unit must be configured to the same as the one used by the communicating equipment it is connected to.

Parameter description:

BAUD_RATE: Size: 1 byte

Value	Parameter Description (bps)
0x01	2400
0x02	4800
0x03	9600
0x04	19200
0x05	38400
0x06	57600
0x07	115200
0x08	230400
0x09	460800
0x0A	921600
0x0B	1382400
0x0C	1843200
0x0D	2764800
0x0E	3686400

FLOW_CONTROL (not configurable for F2M03): Size: 1 byte

Value	Parameter Description
0x00	Hardware flow contriol OFF
0x01	Hardware flow control ON

PARITY: Size: 1 byte

Value	Parameter Description
0x00	No parity
0x01	Odd parity
0x02	Even parity

DATA_BITS: Size: 1 byte

Value	Parameter Description
0x08	Number of data bits to use (not configurable)

STOP_BITS: Size: 1 byte

Value	Parameter Description
0x01	1 stop bit
0x02	2 stop bits

3.3.1 Read serial port configuration request (0x14)

A request to read the serial port configuration.

Command	Length	Command Parameter 1
0x14	0x00	-

3.3.2 Read serial port configuration response (0x14)

If STATUS is success, the serial port configuration is returned otherwise the BAUD_RATE, FLOW_CONTROL, PARITY, DATA_BITS and STOP_BITS parameters will be set to 0 (zero).

Cmd	Len	Cmd Par 1	Cmd Par 2	Cmd Par 3	Cmd Par 4	Cmd Par 5	Cmd Par 6
0x14	0x06	STATUS	BAUD_RATE	FLOW_CONTROL	PARITY	DATA_BITS	STOP_BITS

3.3.3 Set serial port configuration request (0x15)

A request to change the serial port configuration.

Command	Length	Command parameter 1	Command parameter 2	Command parameter 3	Command parameter 4	Command parameter 5
0x15	0x05	BAUD_RATE	FLOW_CONTROL	PARITY	DATA_BITS	STOP_BITS

3.3.4 Set serial port configuration response (0x15)

If STATUS is success, the new serial port settings have been stored in persistent memory.

Command	Length	Command Parameter 1
0x15	0x01	STATUS

3.4 Connect rule

The connect rule parameter are only used when the WU firmware is configured to operate as a Connecting unit and authentication is disabled. See the *“Operating mode”* section for more information.

The connect rule is the Bluetooth address of the remote device the connecting unit shall connect to.

Parameter description:

BLUETOOTH_ADDRESS: Size: 6 bytes

Value	Parameter Description
0x000000000000-0xFFFFFFFFFFFF	Remote Bluetooth address

3.4.1 Read connect rule request (0x16)

A request to read the Bluetooth address of the remote device stored in connect rule.

Command	Length	Command Parameter 1
0x16	0x00	-

3.4.2 Read connect rule response (0x16)

If STATUS is success, the address of the remote Bluetooth device (if any) is returned in the BLUETOOTH_ADDRESS parameter otherwise the value will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x16	0x07	STATUS	BLUETOOTH_ADDRESS

3.4.3 Set connect rule request (0x17)

A request to set the connect rule to the address of a remote Bluetooth device.

Command	Length	Command Parameter 1
0x17	0x06	BLUETOOTH ADDRESS

3.4.4 Set connect rule response (0x17)

If STATUS is success, the new connect rule has been stored in persistent memory.

Command	Length	Command Parameter 1
0x17	0x01	STATUS

3.5 Connect accept rule

The connect accept rule parameter are only used when the WU firmware is configured to operate as an Endpoint unit and authentication is disabled. See the *"Operating mode"* section for more information.

The connect accept rule is the Bluetooth address of a remote device. The WU unit will only accept connections from this device. If the Bluetooth address in the connect accept rule is set to 0 (zero) any remote device is allowed to connect.

BLUETOOTH_ADDRESS: Size: 6 bytes

Value	Parameter Description
0x000000000000	Accept all devices
0x000000000001-0xFFFFFFFFFFFF	Remote Bluetooth address

3.5.1 Read connect accept rule request (0x18)

A request to read the address of the accepted remote Bluetooth device.

Command	Length	Command Parameter 1
0x18	0x00	-

3.5.2 Read connect accept rule response (0x18)

If STATUS is success, the address of the accepted remote Bluetooth device (if any) is returned in the BLUETOOTH_ADDRESS parameter. If BLUETOOTH_ADDRESS is 0 (zero) all devices are accepted.

Command	Length	Command Parameter 1	Command Parameter 2
0x18	0x07	STATUS	BLUETOOTH_ADDRESS

3.5.3 Set connect accept rule request (0x19)

A request to set the connect accept rule to the address of a remote Bluetooth device.
If BLUETOOTH_ADDRESS is set to 0 (zero) all devices will be accepted.

Command	Length	Command Parameter 1
0x19	0x06	BLUETOOTH_ADDRESS

3.5.4 Set connect accept rule response (0x19)

If STATUS is success, the new connect accept rule has been stored in persistent memory.

Command	Length	Command Parameter 1
0x19	0x01	STATUS

3.6 Local Bluetooth name

The local Bluetooth name is a user-friendly name that the device presents itself with.

A Bluetooth device may send a request to get the user-friendly name of another Bluetooth device, e.g. when searching for other Bluetooth devices in the neighborhood (doing an inquiry). The user-friendly name provides the user with the ability to distinguish one Bluetooth device from another besides the unique Bluetooth address.

The NAME parameter is an UTF-8 encoded string with up to 32 bytes in length. The NAME parameter must be null-terminated '\0' (0x00) if the UTF-8 encoded string is less than 32 bytes. If the NAME parameter is 32 bytes, the last byte will be used for null-termination.

Parameter description:

NAME: Size: 1-32 bytes

Value	Parameter Description
0x00-0xFF	Local Bluetooth name in UTF-8 format, terminated with '\0' (0x00)

3.6.1 Read local Bluetooth name request (0x1C)

A request to read the local Bluetooth name.

Command	Length	Command Parameter 1
0x1C	0x00	-

3.6.2 Read local Bluetooth name response (0x1C)

If STATUS is success, the local Bluetooth name is returned in the NAME parameter. The NAME is terminated with '\0' (0x00) and the string terminator is included in the length parameter.

If STATUS is not success, LENGTH will be 0x02 and NAME will be (one byte) string terminated '\0' (0x00).

Command	Length	Command Parameter 1	Command Parameter 2
0x1C	0x02-0x21	STATUS	NAME

3.6.3 Set local Bluetooth name request (0x1D)

A request to change the local Bluetooth name. The NAME must end with '\0' (0x00) and the length parameter is including the string terminator.

Command	Length	Command Parameter 1
0x1D	0x01-0x20	NAME

3.6.4 Set local Bluetooth name response (0x1D)

If STATUS is success, the new local Bluetooth name has been stored in persistent memory.

Command	Length	Command Parameter 1
0x1D	0x01	STATUS

3.7 Local SDP-service name

The local Service Discovery Protocol (SDP) - service name is a user-friendly name of the service that the device offers. The default SDP-service name is “SerialPort” hence the WU firmware offers the Serial Port Profile (SPP) - service.

The Service Discovery Protocol allows Bluetooth devices to discover what services are available, or to find a Bluetooth device that supports a specific service.

The NAME parameter is an UTF-8 encoded string with up to 18 bytes in length. The NAME parameter must be null-terminated ‘\0’ (0x00).

Note: This feature isn’t available when DUN profile mode is activted.

Parameter description:

NAME: Size: 1-18 bytes

Value	Parameter Description
0x00-0xFF	Local SDP-service name in UTF-8 format, terminated with ‘\0’ (0x00)

3.7.1 Read local SDP-service name request (0x1E)

A request to read the local SDP-service name.

Command	Length	Command Parameter 1
0x1E	0x00	-

3.7.2 Read local SDP-service name response (0x1E)

If STATUS is success, the local SDP-service name is returned in the NAME parameter. The NAME is terminated with ‘\0’ (0x00) and the string terminator is included in the length parameter.

If STATUS is not success, LENGTH will be 0x02 and NAME will be (one byte) string terminated ‘\0’ (0x00).

Command	Length	Command Parameter 1	Command Parameter 2
0x1E	0x02-0x13	STATUS	NAME

3.7.3 Set local SDP-service name request (0x1F)

A request to change the local SDP-service name. The NAME must end with ‘\0’ (0x00) and the length parameter is including the string terminator.

Command	Length	Command Parameter 1
0x1F	0x01-0x12	NAME

3.7.4 Set local SDP-service name response (0x1F)

If STATUS is success, the new local SDP-service name has been stored in persistent memory.

Command	Length	Command Parameter 1
0x1F	0x01	STATUS

3.8 LED indication mode

The following LED indication mode commands are only supported on the F2M01SXA product. All other attempts to modify these parameters will fail.

Parameter description:

POWER_LED: Size: 1 byte

Value	Parameter Description
0x00	Disabled – will be turned off 3 seconds after power on
0x01	Enabled – always on
0x02	Enabled – Led flashes differently depending on active state

BT_CONNECTED_LED: Size: 1 byte

Value	Parameter Description
0x00	Disabled – will be turned off 3 seconds after power on
0x01	Enabled

RX_TX_LED: Size: 1 byte

Value	Parameter Description
0x00	Disabled
0x01	Enabled

3.8.1 Read LED indication mode request (0x20)

A request to read the LED indication mode.

Command	Length	Command Parameter 1
0x20	0x00	-

3.8.2 Read LED indication mode response (0x20)

If STATUS is success, the LED indication mode paramters are returned. If STATUS is not success, POWER_LED, BT_CONNECTED_LED and RX_TX_LED will be set to zero.

Command	Length	Paramter 1	Parameter 2	Parameter 3	Parameter 2
0x20	0x04	STATUS	POWER_LED	BT_CONNECTED_LED	RX_TX_LED

3.8.3 Set LED indication mode request (0x21)

A request to change the LED indication mode parameters.

Command	Length	Parameter 1	Parameter 2	Parameter 2
0x21	0x03	POWER_LED	BT_CONNECTED_LED	RX_TX_LED

3.8.4 Set LED indication mode response (0x21)

If STATUS is success, the new LED indication mode parameters have been stored in persistent memory.

Command	Length	Command Parameter 1
0x21	0x01	STATUS

3.9 Connection established notification mode

The connection established notification mode allows the user to configure the WU firmware into a verbose mode.

When enabled the user will receive a notification, every time a successful Bluetooth connection is established, via the UART interface before any other UART data received from the remote device is transmitted to the Host.

Note: The connection established notification mode could cause a potential “dead-lock” situation when hardware flow control is enabled. If the Host does not allow the WU firmware to send out any data it will wait here forever until it has completed the task.

This mode can be used as a substitute for Host’s that do not have the possibility to monitor the Bluetooth connectivity PIO_CONNECTED to find out when a successful Bluetooth connection is established. The drawback is that the Host must parse all received data looking for the notification.

There are three types of notification modes available that determines the type of information that will be presented to the Host:

- Notification bytes – a user-defined number of bytes
- Remote Bluetooth address – the Bluetooth address of the connected remote device in hex notation
- Notification bytes + remote Bluetooth address – the user-defined bytes plus the Bluetooth address of the remote device. For example “Connected”<Bluetooth address in hex notation>

Parameter description:

MODE: Size: 1 byte

Value	Parameter Description
0x00	Disabled
0x01	Enabled – Notification
0x02	Enabled– Remote Bluetooth address
0x03	Enabled – Notification + remote Bluetooth address
Default = 0x00	Disabled

NOTIFICATION: Size: 1-20 bytes

Value	Parameter Description
0x00	Disabled
0x00-0xFF	Notification information that should be sent out to the Host immediately when a successful Bluetooth connection is established
Default = 0x00	Disabled

To disable connection established notification MODE and NOTIFICATION is set to 0 (zero).
To use MODE 0x02, remote Bluetooth address, the NOTIFICATION parameter must be set to 0 (zero).

3.9.1 Read connection established notification mode request (0x22)

A request to read the connection established notification mode.

Command	Length	Command Parameter 1
0x22	0x00	-

3.9.2 Read connection established notification mode response (0x22)

If STATUS is success, the connection established mode parameters are returned otherwise MODE and NOTIFICATION will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2	Command Parameter 3
0x22	0x03-0x16	STATUS	MODE	NOTIFICATION

3.9.3 Set connection established notification mode request (0x23)

A request to change the connection established notification mode parameters.

Command	Length	Command Parameter 1	Command Parameter 2
0x23	0x02-0x15	MODE	NOTIFICATION

3.9.4 Set connection established notification mode response (0x23)

If STATUS is success, the new connection established notification mode parameters have been stored in persistent memory.

Command	Length	Command Parameter 1
0x23	0x01	STATUS

3.10 Connection closed notification mode

The connection closed notification mode allows the user to configure the WU firmware into a verbose mode.

When enabled the user will receive a notification, every time the Bluetooth connection is disconnected, via the UART interface after all buffered UART data received from the remote device has been transmitted to the Host.

Note: The connection closed notification mode could cause a potential "dead-lock" situation when hardware flow control is enabled. If the Host does not allow the WU firmware to send out any data it will wait here forever until it has completed the task.

This mode can be used as a substitute for Host's that do not have the possibility to monitor the Bluetooth connectivity BT_CONNECTED to find out when the Bluetooth connection is disconnected. The drawback is that the Host must parse all received data looking for the notification.

There are three types of notification modes available that determines the type of information that will be presented to the Host:

- Notification bytes – a user-defined number of bytes
- Remote Bluetooth address – the Bluetooth address of the connected remote device in hex notation
- Notification bytes + remote Bluetooth address – the user-defined bytes plus the Bluetooth address of the remote device. For example "Connected"<Bluetooth address in hex notation>

Parameter description:

MODE: Size: 1 byte

Value	Parameter Description
0x00	Disabled
0x01	Enabled – Notification
0x02	Enabled– Remote Bluetooth address
0x03	Enabled – Notification + remote Bluetooth address
Default = 0x00	Disabled

NOTIFICATION: Size: 1-20 bytes

Value	Parameter Description
0x00	Disabled
0x00-0xFF	Notification information that should be sent out to the Host immediately when a successful Bluetooth connection is established
Default = 0x00	Disabled

To disable connection established notification MODE and NOTIFICATION is set to 0 (zero).
To use MODE 0x02, remote Bluetooth address, the NOTIFICATION parameter must be set to 0 (zero).

3.10.1 Read connection closed notification mode request (0x24)

A request to read the connection closed notification mode.

Command	Length	Command Parameter 1
0x24	0x00	-

3.10.2 Read connection closed notification mode response (0x24)

If STATUS is success, the connection closed mode parameters are returned otherwise MODE and NOTIFICATION will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2	Command Parameter 3
0x24	0x03-0x16	STATUS	MODE	NOTIFICATION

3.10.3 Set connection closed notification mode request (0x25)

A request to change the connection closed notification mode parameters.

Command	Length	Command Parameter 1	Command Parameter 2
0x25	0x02-0x15	MODE	NOTIFICATION

3.10.4 Set connection closed notification mode response (0x25)

If STATUS is success, the new connection closed notification mode parameters have been stored in persistent memory.

Command	Length	Command Parameter 1
0x25	0x01	STATUS

3.11 Remote RFCOMM server channel mode

The remote RFCOMM server channel mode is only used when the WU firmware is configured to operate as a Connecting unit. See the *"Operating mode"* section for more information.

The following commands allows the customer to control the behavior of the SDP-service search performed on a remote device.

Before the Connecting unit attempts to establish a Bluetooth Serial Port Profile connection with the configured remote device, it performs an SDP-service search. The SDP-service search is used to make sure that the Serial Port Profile service is provided by the remote device and to gather necessary information about this service, such as the RFCOMM server channel number, which will be used in the connection establishment procedure. If the remote device does not provide the Serial Port Profile service, no connection will be established.

It is possible to enable and disable the remote RFCOMM server channel mode. The remote RFCOMM server channel number can be modified.

The MODE configuration parameter defines if remote RFCOMM server channel mode is enabled or disabled.

The CHANNEL configuration parameter defines the remote RFCOMM server channel number that must match the RFCOMM server channel number retrieved in the SDP-service search.

When remote RFCOMM server channel mode is enabled. The Connecting unit will perform an SDP-service search on all of the provided Serial Port Profile services (if any) by the remote device, looking for a RFCOMM server channel number that will match the configured CHANNEL number. If the channel numbers match, the SDP-service search will stop and an attempt to create a Bluetooth Serial Port Profile connection with the specified remote RFCOMM server channel number will be made. If no channel numbers match, no connection will be established.

When remote RFCOMM server channel mode is disabled. The Connecting unit will perform an SDP-service search on the remote device. It will select the first provided service response (if any) and attempt to create a Bluetooth Serial Port Profile connection to the retrieved RFCOMM server channel. If no service is provided, no connection will be established.

Parameter description:

MODE: Size: 1 byte

Value	Parameter Description
0x00	Remote RFCOMM server channel mode disabled
0x01	Remote RFCOMM server channel mode enabled
Default = 0x00	Remote RFCOMM server channel mode disabled

CHANNEL: Size: 1 byte

Value	Parameter Description
0x00	Remote RFCOMM server channel mode disabled
N = 0x00–0xFF	Range for N: 0x00 – 0xFF Remote RFCOMM server channel
Default N = 0x00	Remote RFCOMM server channel mode disabled

To disable the remote RFCOMM server channel mode MODE and CHANNEL is set to 0 (zero).

3.11.1 Read remote RFCOMM server channel mode request (0x26)

Request to read the remote RFCOMM server channel mode parameters.

Command	Length	Command Parameter 1
0x26	0x00	-

3.11.2 Read remote RFCOMM server channel mode response (0x26)

If STATUS is success, the remote RFCOMM server channel mode parameters are returned otherwise MODE and CHANNEL will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2	Command Parameter 3
0x26	0x03	STATUS	MODE	CHANNEL

3.11.3 Set remote RFCOMM server channel mode request (0x27)

A request to set the remote RFCOMM server channel mode parameters.

Command	Length	Command Parameter 1	Command Parameter 2
0x27	0x02	MODE	CHANNEL

3.11.4 Set remote RFCOMM server channel mode response (0x27)

If STATUS is success, the remote RFCOMM server channel mode parameters have been stored in persistent memory.

Command	Length	Command Parameter 1
0x27	0x01	STATUS

4 Information Commands

Note: Before any information commands can be sent to the WU firmware, it must be set in *“Host Controlled Mode”*.

Commands in this section are read only. It is not possible to change the values of them in any way.

Product information, article number and local Bluetooth address can be read.

4.1 Product information

The production information is specified according to Free2move standard and is useful for support issues.

Parameter description:

PRODUCT_INFORMATION: Size: 12 bytes

Value	Parameter Description
0x00-0xFF	Product information

4.1.1 Read product information request (0x40)

Description:
Request to read the product information.

Command	Length	Command Parameter
0x40	0x00	-

4.1.2 Read product information response (0x40)

If STATUS is success, the product information is returned according to Free2move standard otherwise PRODUCT_INFORMATION will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x40	0x0D	STATUS	PRODUCT_INFORMATION

4.2 Article number

The article number is specified according to Free2move standard and is useful for support issues and sales requests.

Parameter description:

ARTICLE_NUMBER: Size: 20 bytes

Value	Parameter Description
0x00-0xFF	Product information

4.2.1 Read article number request (0x41)

Request to read the article number.

Command	Length	Command Parameter 1
0x41	0x00	-

4.2.2 Read article number response (0x41)

If STATUS is success, the article number is returned according to Free2move standard otherwise ARTICLE_NUMBER will be one byte long and set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x41	0x02-0x15	STATUS	ARTICLE_NUMBER

4.3 Local Bluetooth address

Each Bluetooth unit has a unique 48-bit IEEE MAC address know as Bluetooth Device Address (BD_ADDR). The local Bluetooth address of the WU unit is assigned in production.

Parameter description:

BD_ADDR: Size: 6 bytes

Value	Parameter Description
0x000000000000-0xFFFFFFFFFFFF	Bluetooth address

4.3.1 Read local Bluetooth address request (0x42)

Request to read the local Bluetooth address of the WU unit.

Command	Length	Command Parameter
0x42	0x00	-

4.3.2 Read local Bluetooth address response (0x42)

If STATUS is success, the local Bluetooth address is returned otherwise BD_ADDR will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x42	0x07	STATUS	BD_ADDR

5 Software / Hardware Reboot Commands

Note: Before any software / hardware reboot commands can be sent to the WU firmware, it must be set in *“Host Controlled Mode”*.

5.1 Run

This option forces the WU unit to restart and the application to run in normal operation, i.e. Endpoint or Connecting mode. *“Host Controlled Mode”* will be exited. New settings will be applied and used.

5.1.1 Run request (0x50)

This request forces the WU unit to restart and the application to run in normal operation. *“Host Controlled Mode”* will be exited.

Command	Length	Command Parameter 1
0x50	0x00	-

5.1.2 Run response (0x50)

If STATUS is success this response will be returned before the WU unit restarts. The new settings will be applied on startup.

Command	Length	Command Parameter 1
0x50	0x01	STATUS

6 Inquiry Commands

Note: Before any inquiry commands can be sent to the WU firmware, it must be set in *"Host Controlled Mode"*.

Inquiry is a procedure used to discover other Bluetooth devices in the neighborhood.

To be able to discover a Bluetooth device on inquiry the remote device must be discoverable e.g. have inquiry scan enabled.

The WU firmware offers the possibility to control the discoverability, when operating mode is Endpoint, via the inquiry scan parameters. This is done with the *"Inquiry scan"* commands in the *"Advanced Configuration Commands"* section.

6.1 Scan for devices

To discover Bluetooth devices in the neighborhood, the search is started with a *"Scan for devices request (0x60)"*. The WU firmware responds to the Host's request via the *"Scan for devices response (0x60)"* command, if STATUS is SUCCESS the inquiry procedure has started and the WU firmware is searching for devices. For each discovered device a *"Scan for devices response indication (0x61)"* will be sent to the Host with detailed information about the discovered device. The search continues until the Host receives a *"Scan for devices response indication (0x61)"* with the length parameter set to 1 (one) and STATUS parameter set to SUCCESS indicating that the search has terminated.

There are several parameters that must be specified before the search begins:

- The number seconds a search shall be active
- A filter (if any) used when searching for devices of as certain class
- A filter (if any) used when searching for devices of a certain inquiry access code
- The possibility to include the remote Bluetooth name of devices found
- The possibility to include the received signal strength indication (RSSI) of devices found
- The possibility to search for a specific number of devices

The TIMEOUT parameter controls the maximum amount of time for the inquiry before it is terminated. The timeout is specified as defined in the HCI section of the Bluetooth specification. The time the inquiry is performed for is in fact timeout * 1.28 seconds. The allowed values of timeout are in the range 0x01 to 0x30. This corresponds to an inquiry timeout range of 1.28 to 61.44 seconds.

The class of device filter, COD, can be used as a filter when searching for devices of a certain class. Set to 0 (zero) to be notified of all devices within range regardless of their class of device. If the class of device is set only devices with those bits set in their class of device will be reported. This filter can be made as specific or as general as necessary. For example, to discover all devices with major device class set to audio/video set the class of device field to 0x400. This will result in devices with, for example, class of device set to 0x200404 and 0x200408 to be reported. If, however, we want to limit our device search even further and only discover devices with their class of device set to headset, for example, the class of device field should be set to 0x200404. Please note that the filter reports devices that have a minimum of the specified bits set, however it is possible that a device also has other bits set in its class of device field. In that case that device will be reported back as well.
The class of device field is 24bits, which means that the MSB of the COD filter must always be set to 0x00, since the parameter is 32bits (4 bytes).

The lower address part LAP from which the inquiry access code should be derived. The General / Unlimited Inquiry Access Code (GIAC) is specified in this document as 0x9E8B33 and this is the code most applications will use when performing an inquiry.
The LAP field is 24bits, which means that the MSB of the LAP filter must always be set to 0x00, since the parameter is 32bits (4 bytes).

See the “Bluetooth Assigned Numbers” document in the Bluetooth specification for more detailed information about the class of device field and the inquiry access code.
https://www.bluetooth.org/foundry/assignnumb/document/baseband

If the remote name request RNR parameter is enabled the user-friendly Bluetooth name of the found remote device will be included in the response.

The MAX_RESP parameter controls the maximum number of responses. Once this many devices have been discovered the inquiry process will terminate and the Host will be notified. Maximum number of responses set to 0 (zero) indicates unlimited number of devices. In this case the inquiry will terminate when the timeout expires. It is not recommended that this field is set to zero or a value higher than ten as this may result in firmware stability issues.

The RSSI parameter controls whether the Received Signal Strength Indication (RSSI) value of the inquiry response message from the discovered device shall be measured or not.
If RSSI is enabled several responses from the same remote device may be received during the search.
If RSSI is disabled only one response from each found device will be returned during the search.

Scan for devices request parameter description:

INQ_LAP: Size: 4 bytes

Value	Parameter Description
0x009E8B00	Limited dedicated inquiry access code
0x009E8B33	General/unlimited inquiry access code

MAX_RESP: Size: 1 byte

Value	Parameter Description
0x00	Unlimited number of devices
0x01-0xFF	Maximum number of devices

TIMEOUT: Size: 1 byte

Value	Parameter Description
0x01-0x30	Amount of time before inquiry is terminated

COD: Size: 4 bytes

Value	Parameter Description
0x00000000	No class of device filter, all devices will be returned
0x00000001-0x00FFFFFF	Class of device filter, devices of a certain class will be returned

RSSI: Size: 1 byte

Value	Parameter Description
0x00	Standard inquiry mode (no RSSI)
0x01	Inquiry with RSSI

RNR: Size: 1 byte

Value	Parameter Description
0x00	Disable remote name request
0x01	Enable remote name request

The response of a Bluetooth device discovered in the neighborhood is returned with a number of informative parameters.

The Bluetooth address, BD_ADDR, of the discovered device.

If requested the user-friendly Bluetooth name, NAME, is included in UTF-8 format. If the remote device did not respond to the remote name request the NAME parameter returned will be one byte long and set to 0 (zero). The NAME is terminated with '\0' (0x00) and the string terminator is included in the length parameter. If RNR is not enabled the NAME parameter will be one byte long and set to 0 (zero).

RSSI of the discovered device. Set to 0x7FFF if value not available.
Note: The RSSI value is returned in signed integer16 format.

The COD parameter indicates the class of device of the discovered device. See Bluetooth Assigned Numbers for more information about the class of device field.
https://www.bluetooth.org/foundry/assignnumb/document/baseband

Note: The response parameters listed below are considered to be of an advanced informative nature. For in depth understanding please refer to the Bluetooth specification. For the normal user these parameters are of no greater importance and can safely be ignored.

The CLK_OFFSET parameter indicates the clock offset of the discovered device.

The PS_REP parameter indicates the page scan repetition mode used by the discovered device.

The PS_MODE parameter indicates the default page scan mode that is used by the discovered device. Currently one mandatory page scan mode and three optional page scan modes are defined.

When the length parameter equals 1 (one) in the response and STATUS is SUCCESS, this indicates that the search is over and no more devices will be found.

Scan for devices response parameter description:

LENGTH: Size: 1byte

Value	Parameter Description
0x01	Search for devices has terminated. STATUS will be SUCCESS
0x12-0x31	Length of response

BD_ADDR: Size: 6 bytes

Value	Parameter Description
0x000000000000-0xFFFFFFFFFFFF	Bluetooth address of the discovered device

COD: Size: 4 bytes

Value	Parameter Description
0x00000001-0x00FFFFFF	Class of device of the discovered device

CLK_OFFSET: Size: 2 bytes

Value	Parameter Description
0x0000-0xFFFF	Clock offset of the discovered device

PS_REP: Size: 1 byte

Value	Parameter Description
0x00	R0 – page scan interval <=1.28 seconds and = page scan window
0x01	R1 – page scan interval <=1.28 seconds
0x02	R2 – page scan interval <=2.56 seconds
0x03	Unknown page scan mode

PS_MODE: Size: 1 byte

Value	Parameter Description
0x00	Mandatory page scan mode
0x01	Optional page scan mode I
0x02	Optional page scan mode II
0x03	Optional page scan mode III
0x04	Unknown page scan mode

RSSI: Size: 2 bytes

Value	Parameter Description
0x8000-0x7FFF	RSSI value of the discovered device. Set to 0x7FFF if value not available **Note**: Signed integer16

NAME: Size: 1-32 bytes

Value	Parameter Description
0x00-0xFF	Name of remote device in UTF-8 format. String terminated with '\0' (0x00)

6.1.1 Scan for devices request (0x60)

Request to start a search for remote Bluetooth devices in the neighborhood. The search will be performed for either the specified amount of time, TIMEOUT, or until the specified maximum number of responses, MAX_RESP, is reached before it terminates. Specific filters, INQ_LAP and/or COD, can be set to limit the number of responses received when searching for devices within range. The user-friendly Bluetooth name and the signal strength of the found remote device can be included in the response by enabling the RNR and the RSSI parameters.

Cmd	Len	Cmd Par 1	Cmd Par 2	Cmd Par 3	Cmd Par 4	Cmd Par 5	Cmd Par 6
0x60	0x0C	INQ_LAP	MAX_RESP	TIMEOUT	COD	RSSI	RNR

6.1.2 Scan for devices response (0x60)

If STATUS is success the search for remote Bluetooth devices in the neighborhood has started.

Command	Length	Command Parameter 1
0x60	0x01	STATUS

6.1.3 Scan for devices response indication (0x61)

This indication is sent to the Host whenever a remote Bluetooth device is found.

Cmd	Len	Cmd Par 1	Cmd Par 2	Cmd Par 3	Cmd Par 4	Cmd Par 5	Cmd Par 6	Cmd Par 7	Cmd Par 8
0x61	0x12-0x31	STATUS	BD_ADDR	COD	CLK_OFFSET	PS_REP	PS_MODE	RSSI	NAME

When the search has terminated the LENGTH parameter will be set to 1 (one) and STATUS parameter will be set to SUCCESS. The rest of the device specific parameters are omitted.

Command	Length	Command Parameter 1
0x61	0x01	STATUS

7 Pairing And Security Commands

Note: Before any pairing and security commands can be sent to the WU firmware, it must be set in *"Host Controlled Mode"*.

The *"Security mode"* configuration determines the level of security required by the WU firmware when establishing Bluetooth connections.

If link level enforced security or service level enforced security is selected the devices must be paired before a successful connection is established.

The WU firmware uses a pairing mechanism called "pairing on the fly" that allows the pairing procedure to be executed "on the fly" when establishing the Bluetooth connection. The PIN code used during the "pairing on the fly" is the PIN code specified with the *"PIN code"* command.

For a successful pairing to take place both devices must exchange the same PIN code, otherwise the pairing attempt will fail. If the pairing attempt was successful, a unique link key is generated and saved in persistent memory by the WU firmware. This link key will be used for secure verification of the relationship between the paired devices.

The WU firmware allows one remote device to be paired at each time. When a successful pairing has been made, the remote device's Bluetooth address is saved as the *"Paired device"*.

7.1 Security mode

The security mode configuration determines the level of security required by the WU firmware when establishing Bluetooth connections.

There are four levels of security:

- Security mode 0 and 1 allows connections to be established without security.
- Security mode 2 requires security on service level, i.e. when connecting to the Bluetooth Serial Port Profile (SPP) service.
- Security mode 3 requires security on link level, i.e. when attempting to establish a Bluetooth baseband connection.

The SECURITY_MODE parameter controls where in the channel establishment procedures initiation of security takes place, depending on the configured security mode.

The ENCRYPTION_MODE parameter controls if the traffic shall be encrypted or not. Encryption is only available for security mode 2 and 3.

When *"Operating mode"* is Endpoint mode – the WU firmware will accept secure connections according to the *"Connect accept rule"*.

It can either be a 1:1 or 1:N relationship between the local device and the remote device(s) depending on the configured *"Connect accept rule"*.

1:1 relationship means that only the remote device specified is accepted and allowed to pair and establish a secure connection. Any other device that attempts to pair and establish a secure connection will be rejected even if they know the correct PIN code. To allow a new device to pair and connect *"Host Controlled Mode"* must be entered and a new *"Connect accept rule"* must be specified.

1:N relationship means that all devices are allowed to be paired and connected to the local device, however only one device at the time. Any remote device that knows the correct PIN code is allowed to pair and establish a secure connection. The last paired device is always stored as the *"Paired device"* in persistent memory by the WU firmware.

When *"Operating mode"* is Connecting mode – the WU firmware attempts to establish a secure connection to the remote Bluetooth device specified in the *"Connect rule"*.

In Connecting mode the relationship is always 1:1.

Parameter description:

SECURITY_MODE: Size: 1 byte

Value	Parameter Description
0x00	Security mode 0 – security off
0x01	Security mode 1 – non-secure
0x02	Security mode 2 – service level enforced security
0x03	Security mode 3 – link level enforced security
Default = 0x00	Security mode 0 – security off

ENCRYPTION_MODE: Size: 1 byte

Value	Parameter Description
0x00	Encryption off
0x01	Encrypt point to point traffic
0x02	Encrypt point to point and broadcast traffic
Default = 0x00	Encryption off

To disable the security mode SECURITY_MODE and ENCRYPTOIN_MODE is set to 0 (zero).

7.1.1 Read security mode request (0x62)

Request to read the current security mode configuration.

Command	Length	Command Parameter 1
0x62	0x00	-

7.1.2 Read security mode response (0x62)

The current security mode configuration is returned if STATUS is success. Otherwise the SECURITY_MODE and ENCRYPTION_MODE parameters will be 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2	Command Parameter 3
0x62	0x03	STATUS	SECURITY_MODE	ENCRYPTION_MODE

7.1.3 Set security mode request (0x63)

Request to change the current security mode configuration.

Command	Length	Command Parameter 1	Command Parameter 2
0x63	0x02	SECURITY_MODE	ENCRYPTION_MODE

7.1.4 Set security mode response (0x63)

Returns the status of the set security mode request. If STATUS is success the new security configuration has been stored in persistent memory.

Command	Length	Command Parameter 1
0x63	0x01	STATUS

7.2 PIN code

The Personal Identification Number, PIN code, is a user-friendly number that can be used to authenticate connections to a device before pairing has taken place.

The PIN code will be used when the WU firmware is configured for secure connection e.g. security mode 2 or security mode 3 via the *"Set security mode request (0x63)"* command described in the *"Security mode"* section.

The PIN code specified in the WU firmware and the remote device's PIN code must be equal for a successful pairing to take place in order to establish a secure connection. The last paired device is always stored as the *"Paired device"* in the WU firmware.

Parameter description:

PIN_CODE: Size: 1-16 byte

Value	Parameter Description
0x00-0xFF	Pin code used for secure connections

The default PIN code is set to "0000" in ASCII notation (0x30 0x30 0x30 0x30 in hex notation). The default value is valid as long as no configuration has been made by the user.

7.2.1 Read PIN code request (0x64)

A request to read the PIN code.

Command	Length	Command Parameter 1
0x64	0x00	-

7.2.2 Read PIN code response (0x64)

Returns the PIN code.

If STATUS is not success, LENGTH will be 0x02 and PIN_CODE will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x64	0x02- 0x11	STATUS	PIN_CODE

7.2.3 Set PIN code request (0x65)

Request to set a new PIN code that should be used for secure connections.

Command	Length	Command Parameter 1
0x65	0x01- 0x10	PIN_CODE

7.2.4 Set PIN code response (0x65)

Returns the status of the set PIN code request. If success the new PIN code has been stored in persistent memory.

Command	Length	Command Parameter 1
0x65	0x01	STATUS

7.3 Paired device

When a successful pairing has been achieved the Bluetooth address of the remote device is stored in the paired device field.

Parameter description:

BLUETOOTH_ADDRESS: Size: 6 bytes

Value	Parameter Description
0x000000000000-0xFFFFFFFFFFFF	Bluetooth address of paired device

7.3.1 Read Paired device request (0x66)

Request to read the Bluetooth address (if any) of the paired remote device.

Command	Length	Command Parameter 1
0x66	0x00	-

7.3.2 Read Paired device response (0x66)

If STATUS is success the Bluetooth address of the paired device is returned otherwise BLUETOOTH_ADDRESS will be 0 (zero).

If no paired device BLUETOOTH_ADDRESS will be 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0x66	0x07	STATUS	BLUETOOTH_ADDRESS

7.3.3 Set Paired device request (0x67)

Request to erase the Bluetooth address of the last stored device and the link keys of the 8 last connected devices.

Command	Length	Command Parameter 1
0x67	0x00	-

7.3.4 Set Paired device response (0x67)

If STATUS is success the Bluetooth address of the paired device is erased.

Command	Length	Command Parameter 1
0x67	0x01	STATUS

8 Control Commands

Note: Before any control commands can be sent to the WU firmware, it must be set in *"Host Controlled Mode"*.

8.1 Cancel the current action

By using this command the user is able to cancel the current action of the WU unit. Currently there is only the inquiry action to cancel.

A cancel request can be used to terminate the ongoing action in the WU unit. The outcome of the cancel request is determined by the action being cancelled.

For example to cancel an ongoing inquiry a *"Cancel request (0x80)"* command must be sent. A *"Cancel response (0x80)"* command will be received indicating the outcome. If STATUS is success the WU firmware attempts to cancel the inquiry procedure. When the inquiry procedure has been successfully cancelled a *"Scan for devices response indication (0x61)"* with length parameter set to 1 (one) and STATUS parameter set to success will be received, indicating that the search has terminated.

8.1.1 Cancel request (0x80)

A cancel request can be used to terminate the ongoing inquiry. The outcome of the cancel request is dependent of the action being cancelled.

Command	Length	Command Parameter
0x80	0x00	-

8.1.2 Cancel response (0x80)

This is the response of a cancel request. If STATUS is success then an attempt to cancel the current action is being performed.

Command	Length	Command Parameter
0x80	0x01	STATUS

9 Advanced Configuration Commands

Note: Before any advanced configuration commands can be sent to the WU firmware, it must be set in *"Host Controlled Mode"*.

WARNING!!
The following advanced configuration commands should be used with great care. Do not change anything unless you are absolutely sure of what you are doing. Changing these parameters may affect the performance of the WU firmware.
You will need to have in depth Bluetooth knowledge to use some of these commands.

9.1 Throughput/latency mode

By using these commands the WU firmware can be optimized for either throughput or latency. The throughput/latency mode configuration allows the user to fine-tune the low-level settings associated with the UART.

Throughput mode attempts to maximize the throughput at the cost of moderate latency. If low latency is more important than maximized throughput, latency mode may be selected.

When optimized for latency – the UART driver of the WU firmware is more aggressive at looking for subsequent data bytes at the cost of increased CPU load, hence the trade-off against throughput.

Parameter description:

MODE: Size: 1 byte

Value	Parameter Description
0x01	Throughput mode
0x02	Latency mode
Default = 0x01	Throughput mode

9.1.1 Read throughput/latency mode request (0xC0)

Request to read the throughput/latency mode.

Command	Length	Command Parameter 1
0xC0	0x00	-

9.1.2 Read throughput/latency mode response (0xC0)

If STATUS is success, the throughput/latency mode is returned otherwise MODE will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xC0	0x02	STATUS	MODE

9.1.3 Set throughput/latency mode request (0xC1)

Request to set the throughput/latency mode.

Command	Length	Command Parameter 1
0xC1	0x01	MODE

9.1.4 Set throughput/latency mode response (0xC1)

If STATUS is success, the throughput/latency mode has been stored in persistent memory.

Command	Length	Command Parameter 1
0xC1	0x01	STATUS

9.2 Link supervision timeout

A Bluetooth connection may break down due to various reasons such as a device moving out of range or a power failure condition. Since this may happen without any prior warning, it is important to monitor the link on both the Connecting and the Endpoint side. To be able to supervise link loss, both the Connecting and the Endpoint use link supervision timers.

If at any time in connected state, the timer reaches the supervision TIMEOUT value, the connection is reset. The timeout period, TIMEOUT, is negotiated when a successful Bluetooth connection is established. Its value must be chosen so that the supervision timeout is longer than the low power save mode periods specified in the *"Low power mode"* configuration. Link supervision of a parked Endpoint will be done by unparking and re-parking the Endpoint.

The link supervision timeout has a span of 0.625ms – 40.9s. The table below shows possible values. See Bluetooth specification for more information.

Parameter description:

TIMEOUT: Size: 2 bytes

Value	Parameter Description
0x0000	No link supervision timeout
N = 0x0001- 0xFFFF	Link supervision timeout time = N*0.625 msec Range for N: 0x0001 – 0xFFFF Time range: 0.625msec – 40.9sec
Default N = 0x7D00	Link supervision timeout time = 20sec

To disable link supervision, TIMEOUT is set to 0 (zero). This makes it unnecessary for the Connecting unit to unpark and then park the connected Endpoint every ~40 seconds. By using the no link supervision timeout setting, the park mode is not limited. It is however not recommended to disable the link supervision timeout nor to use a very short link supervision timeout, as this may cause unstable behavior of the Bluetooth connection.

9.2.1 Read link supervision timeout request (0xC2)

Request to read the link supervision timeout.

Command	Length	Command Parameter 1
0xC2	0x00	-

9.2.2 Read link supervision timeout response (0xC2)

If STATUS is success, the link supervision timeout is returned otherwise TIMEOUT will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xC2	0x03	STATUS	TIMEOUT

9.2.3 Set link supervision timeout request (0xC3)

Request to set the link supervision timeout.

Command	Length	Command Parameter 1
0xC3	0x02	TIMEOUT

9.2.4 Set link supervision timeout response (0xC3)

If STATUS is success, the link supervision has been stored in persistent memory.

Command	Length	Command Parameter 1
0xC3	0x01	STATUS

9.3 Inquiry scan

The inquiry scan parameters are only applied when the WU firmware is configured to operate as an Endpoint and as long as there is no Bluetooth connection established.

It is possible to enable and disable inquiry scan. The scan interval and scan window for inquiry scan can be modified.

The INQUIRY_SCAN_INTERVAL configuration parameter defines the amount of time between consecutive inquiry scans. This is defined as the time interval from when the device started its last inquiry scan until it begins the next inquiry scan.

The INQUIRY_SCAN_WINDOW configuration parameter defines the amount of time for the duration of the inquiry scan. The INQUIRY_SCAN_WINDOW can only be less than or equal to the INQUIRY_SCAN_INTERVAL.

- By choosing a long interval time and a short window time the average current consumption can be decreased at the cost of slow response time on inquiry.
- By choosing a short interval time and a long window time faster response time on inquiry can be obtained at the cost of increased average current consumption.

When inquiry scan is enabled, the Endpoint unit is discoverable for other Bluetooth units on inquiry.

The inquiry scan parameters have a span of 11.25ms – 2.56s. The tables below show possible values. See Bluetooth specification for more information.

Parameter description:

INQUIRY_SCAN_INTERVAL: Size: 2 bytes

Value	Parameter Description
0x0000	Inquiry scan disabled
N = 0x0012 – 0x1000	Inquiry scan interval time = N * 0.625msec (INQUIRY_SCAN_INTERVAL >= INQUIRY_SCAN_WINDOW) Range for N: 0x0012 – 0x1000 Time range: 11.25 – 2560msec
Default N = 0x0800	Inquiry scan interval time = 1.28sec

INQUIRY_SCAN_WINDOW: Size: 2 bytes

Value	Parameter Description
0x0000	Inquiry scan disabled
N = 0x0012 – 0x1000	Inquiry scan window time = N * 0.625msec (INQUIRY_SCAN_INTERVAL >= INQUIRY_SCAN_WINDOW) Range for N: 0x0012 – 0x1000 Time range: 11.25 – 2560msec
Default N = 0x0012	Inquiry scan window time = 11.25msec

To disable inquiry scan INQUIRY_SCAN_INTERVAL and INQUIRY_SCAN_WINDOW is set to 0 (zero).

9.3.1 Read inquiry scan parameters request (0xC4)

Request to read the inquiry scan parameters.

Command	Length	Command Parameter 1
0xC4	0x00	-

9.3.2 Read inquiry scan parameters response (0xC4)

If STATUS is success, the inquiry scan parameters are returned otherwise INQUIRY_SCAN_INTERVAL and INQUIRY_SCAN_WINDOW will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2	Command Parameter 3
0xC4	0x05	STATUS	INQUIRY_SCAN_INTERVAL	INQUIRY_SCAN_WINDOW

9.3.3 Set inquiry scan parameters request (0xC5)

Request to set the inquiry scan parameters.

Command	Length	Command Parameter 1	Command Parameter 2
0xC5	0x04	INQUIRY_SCAN_INTERVAL	INQUIRY_SCAN_WINDOW

9.3.4 Set inquiry scan parameters response (0xC5)

If STATUS is success, the inquiry scan parameters have been stored in persistent memory.

Command	Length	Command Parameter 1
0xC5	0x01	STATUS

9.4 Page scan

The page scan parameters are only applied when the WU firmware is configured to operate as an Endpoint and as long as there is no Bluetooth connection established.

It is possible to enable and disable page scan. The scan interval and scan window for page scan can be modified.

The PAGE_SCAN_INTERVAL configuration parameter defines the amount of time between consecutive page scans. This is defined as the time interval from when the device started its last page scan until it begins the next page scan.

The PAGE_SCAN_WINDOW configuration parameter defines the amount of time for the duration of the page scan. The PAGE_SCAN_WINDOW can only be less than or equal to the PAGE_SCAN_INTERVAL.

- By choosing a long interval time and a short window time the average current consumption can be decreased at the cost of slow response time on page.
- By choosing a short interval time and a long window time faster response time on page can be obtained at the cost of increased average current consumption.

When page scan is enabled, the Endpoint unit is connectable for other Bluetooth units on page.

The page scan parameters have a span of 11.25ms – 2.56s. The tables below show possible values. See Bluetooth specification for more information.

Parameter description:

PAGE_SCAN_INTERVAL: Size: 2 bytes

Value	Parameter Description
0x0000	Page scan disabled
N = 0x0012 – 0x1000	Page scan interval time = N * 0.625msec (PAGE_SCAN_INTERVAL >= PAGE_SCAN_WINDOW) Range for N: 0x0012 – 0x1000 Time range: 11.25 – 2560msec
Default N = 0x0800	Page scan interval time = 1.28sec

PAGE_SCAN_WINDOW: Size: 2 bytes

Value	Parameter Description
0x0000	Page scan disabled
N = 0x0012 – 0x1000	Page scan window time = N * 0.625msec (PAGE_SCAN_INTERVAL >= PAGE_SCAN_WINDOW) Range for N: 0x0012 – 0x1000 Time range: 11.25 – 2560msec
Default N = 0x0012	Page scan window time = 11.25msec

To disable page scan PAGE_SCAN_INTERVAL and PAGE_SCAN_WINDOW is set to 0 (zero).

9.4.1 Read page scan parameters request (0xC6)

Request to read the page scan parameters.

Command	Length	Command Parameter 1
0xC6	0x00	-

9.4.2 Read page scan parameters response (0xC6)

If STATUS is success, the page scan parameters are returned otherwise the PAGE_SCAN_INTERVAL and PAGE_SCAN_WINDOW will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2	Command Parameter 3
0xC6	0x05	STATUS	PAGE_SCAN_INTERVAL	PAGE_SCAN_WINDOW

9.4.3 Set page scan parameters request (0xC7)

Request to set the page scan parameters.

Command	Length	Command Parameter 1	Command Parameter 2
0xC7	0x04	PAGE_SCAN_INTERVAL	PAGE_SCAN_WINDOW

9.4.4 Set page scan parameters response (0xC7)

If STATUS is success, the page scan parameters have been stored in persistent memory.

Command	Length	Command Parameter 1
0xC7	0x01	STATUS

9.5 Class of device

The class of device parameter is used to indicate the capabilities of the local device to other devices, i.e. indicating what type of device it is. For example, when searching for other devices with a Bluetooth mobile phone, each device discovered would have a symbol associated with it to give the user an easy overview of what type of device it is. A headset would appear with a symbol that looks like an earphone and a laptop or desktop workstation would appear as a computer.

When searching for other Bluetooth devices in the neighborhood with the WU firmware, the *“Scan for devices”* command in the *“Inquiry Commands”* section, it is possible to use a specific class of device filter for searching after devices of a certain class.

The class of device parameter consists of the following fields:

- Major service class (bits 23-13)
- Major device class (bits 12-8)
- Minor device class (bits 7-2)
- Format type (bits 1-0)

The class of device is 24bits, this means that the MSB of the CLASS_OF_DEVICE must always be set to 0x00, since the parameter is 32bits (4 bytes).

See Bluetooth Assigned Numbers for more information about the class of device.
https://www.bluetooth.org/foundry/assignnumb/document/baseband

The default class of device value of the WU firmware is set to uncategorized because nothing seems to fit cable replacement very well.

Parameter description:

CLASS_OF_DEVICE: Size: 4 bytes

Value	Parameter Description
0x00000000 – 0x00FFFFFF	Class of device
Default = 0x00001F00	Miscellaneous - uncategorized

9.5.1 Read class of device request (0xCC)

Request to read the class of device.

Command	Length	Command Parameter 1
0xCC	0x00	-

9.5.2 Read class of device response (0xCC)

If STATUS is success, the class of device parameter is returned otherwise CLASS_OF_DEVICE will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xCC	0x05	STATUS	CLASS_OF_DEVICE

9.5.3 Set class of device request (0xCD)

Request to set the class of device.

Command	Length	Command Parameter 1
0xCD	0x04	CLASS_OF_DEVICE

9.5.4 Set class of device response (0xCD)

If STATUS is success, the class of device has been stored in persistent memory.

Command	Length	Command Parameter 1
0xCD	0x01	STATUS

9.6 Default transmit power

The default transmit power, measured in dBm, is the power used by the Bluetooth radio for page, inquiry and their scan responses. This is also the initial power used for new connections.

The default transmit power has a valid range of –90dBm to +20dBm, however depending on the radio power class used and its radio power table, this range may not be fully used. It is up to the user to set the desired value for this parameter. The table below shows an example with suggested value.

Parameter description:

DEFAULT_TX_POWER: Size: 2 bytes

Value	Parameter Description
N = 0xFFA6 – 0x0014	Range for N: 0xFFA6 – 0x0014 Default transmit power: -90dBm to +20dBm **Note**: Signed integer16
Default N = 0x0004	Default transmit power = +4dBm

The recommended default value for DEFAULT_TX_POWER is +4dBm.

- Setting a lower value than default will decrease the range for the unit. It may not be able to discover/connect to devices far away.
- Setting a higher value than default will increase the range for the unit. It might be able to discover/connect to devices further away, however it may also interfere with devices at close range, due to sending with too much output power. The consequences might be that the device will have problems to discover/connect to devices at close range.

The requested value in the DEFAULT_TX_POWER parameter will be rounded down to the next available value when set, so the value set may be less than the requested. The WU firmware uses the highest value in the radio power table that is less than or equal to the requested default transmit power. The actual default transmit power used will be returned in the response command. The table below shows the radio power table used in the WU firmware for different radio power classes.

Note: If DEFAULT_TX_POWER > MAXIMUM_TX_POWER, the value for maximum transmit power will be returned.

Radio power table:

F2M03GLA (dBm)	F2M03GX/GXA, F2M01SXA (dBm)	F2M03MLA (dBm)	F2M03ALA (dBm)
+8	+20	+6	+4
+4	+16	+3	0
0	+12	0	-4
-4	+8	-4	-8
-8	+4	-8	-12
-12	0	-12	-16
-16	-4	-16	-20
-20	-8	-20	-24

9.6.1 Read default transmit power request (0xD2)

Request to read the default transmit power.

Command	Length	Command Parameter 1
0xD2	0x00	-

9.6.2 Read default transmit power response (0xD2)

If STATUS is success, the default transmit power is returned otherwise DEFAULT_TX_POWER will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xD2	0x03	STATUS	DEFAULT_TX_POWER

9.6.3 Set default transmit power request (0xD3)

Request to set the default transmit power.

Command	Length	Command Parameter 1
0xD3	0x02	DEFAULT_TX_POWER

9.6.4 Set default transmit power response (0xD3)

If STATUS is success, the default transmit power has been stored in persistent memory. The actual default transmit power that will be used is returned in the DEFAULT_TX_POWER parameter.

If STATUS is not success, DEFAULT_TX_POWER will be 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xD3	0x03	STATUS	DEFAULT_TX_POWER

9.7 Maximum transmit power

The maximum transmit power, measured in dBm, ensures that the local Bluetooth radio never transmits with higher power than this value.

Maximum transmit power control can be used to adapt the WU unit to follow the rules and regulations regarding output power for certain countries but also be used for reducing the range of the unit.

Once a successful connection has been established, it is possible to control the remote device’s transmit power (if the remote device supports this feature). The remote device can request an increase or decrease

of transmit power depending on how “strong” the signal is. The maximum transmit power is only referenced when increasing the local transmit power. The WU firmware supports power control.
The maximum transmit power has a valid range of –90dBm to +20dBm, however depending on the radio power class used and its radio power table, this range may not be fully used. It is up to the user to set the desired value for this parameter. The table below shows an example with suggested value.

Parameter description:

MAXIMUM_TX_POWER: Size: 2 bytes

Value	Parameter Description
N = 0xFFA6 – 0x0014	Range for N: 0xFFA6 – 0x0014 Maximum transmit power: -90dBm to +20dBm **Note**: Signed integer16
Default Class1 N = 0x0014	Maximum transmit power = +20dBm
Default Class2 N = 0x0004	Maximum transmit power = +4dBm

The recommended default value for MAXIMUM_TX_POWER is +20 dBm for radio class 1 and +4dBm for radio class 2

- Setting a lower value than default will decrease the range for the unit.
- Setting a higher value than default is not possible, due to the transmit power limits of the different radio classes.

The requested value set in the MAXIMUM_TX_POWER parameter will be rounded down to the next available value when set, so the value set may be less than the requested. The WU firmware uses the highest value in the radio power table that is less than or equal to the requested maximum transmit power. The actual maximum transmit power used will be returned in the response command. The table below shows the radio power table used in the WU firmware for different radio power classes.

Radio power table:

F2M03GLA (dBm)	F2M03GX/GXA, F2M01SXA (dBm)	F2M03MLA (dBm)	F2M03ALA (dBm)
+8	+20	+6	+4
+4	+16	+3	0
0	+12	0	-4
-4	+8	-4	-8
-8	+4	-8	-12
-12	0	-12	-16
-16	-4	-16	-20
-20	-8	-20	-24

9.7.1 Read maximum transmit power request (0xD4)

Request to read the maximum transmit power.

Command	Length	Command Parameter 1
0xD4	0x00	-

9.7.2 Read maximum transmit power response (0xD4)

If STATUS is success, the maximum transmit power is returned otherwise the MAXIMUM_TX_POWER will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xD4	0x03	STATUS	MAXIMUM_TX_POWER

9.7.3 Set maximum transmit power request (0xD5)

Request to set the maximum transmit power.

Command	Length	Command Parameter 1
0xD5	0x02	MAXIMUM_TX_POWER

9.7.4 Set maximum transmit power response (0xD5)

If STATUS is success, the maximum transmit power has been stored in persistent memory. The actual maximum transmit power that will be used is returned in the MAXIMUM_TX_POWER parameter.

If STATUS is not success, MAXIMUM_TX_POWER will be 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xD5	0x03	STATUS	MAXIMUM_TX_POWER

9.8 Emulate serial handshaking lines

The F2M03 module with WU firmware can emulate serial handshaking lines using PIO pins. When a successful Bluetooth connection is established; DCD, DSR, RI and DTR signals are transferred between the connected units. Whenever a local PIO pin's level change the remote device will receive a notification.

The F2M01SXA can emulate DTR and DSR as a DCE device. RI and DCD is not connected in the D-Sub.

The RTS and CTS lines are always used for local hardware flow control and are not transferred.

The WU firmware can emulate a DTE or a DCE. Serial handshaking emulation can be enabled or disabled.

The following PIO's are used when emulating serial handshaking lines:

Emulated Signal	Signal Direction Emulate DTE	Signal Direction Emulate DCE	Active (TTL)
RI	Input	Output	High
DTR	Output	Input	High
DCD	Input	Output	High
DSR	Input	Output	High

Emulated Signal	PIO F2M03G	PIO F2M03MLA	PIO F2M03ALA	PIN F2M01SXA
RI	4	8	4	Not connected
DTR	5	9	5	4
DCD	6	6	6	Not connected
DSR	7	7	7	6

Parameter description:

MODE: Size: 1 byte

Value	Parameter Description
0x00	Serial emulation disabled
0x01	Emulate DTE
0x02	Emulate DCE
Default = 0x00	Serial emulation disabled

9.8.1 Read emulate serial handshaking mode request (0xD6)

Request to read the emulate serial handshaking mode.

Command	Length	Command Parameter 1
0xD6	0x00	-

9.8.2 Read emulate serial handshaking mode response (0xD6)

If STATUS is success, the emulate serial handshaking mode is returned otherwise MODE will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xD6	0x02	STATUS	MODE

9.8.3 Set emulate serial handshaking mode request (0xD7)

Request to set the emulate serial handshaking mode.

Command	Length	Command Parameter 1
0xD7	0x01	MODE

9.8.4 Set emulate serial handshaking mode response (0xD7)

If STATUS is success, the emulate serial handshaking mode has been stored in persistent memory.

Command	Length	Command Parameter 1
0xD7	0x01	STATUS

9.9 Override serial port configuration baud rate

The specified baud rate in the *"Serial port configuration"* command can be overridden with a user defined baud rate.

It is possible to set baud rates from 244 baud up to the maximum rated speed for the UART hardware 4.0 Mbaud.

Parameter description:

OVERRIDE: Size: 1 byte

Value	Parameter Description
0x00	Override serial baud rate disabled
0x01	Override serial baud rate enabled
Default = 0x00	Override serial baud rate disabled

BAUD_RATE: Size: 2 bytes

Value	Parameter Description
0x0000	Override serial baud rate disabled
N = 0x0001 – 0x4000	Serial baud rate Baud Rate = N / 0.004096
Default = 0x0000	Override serial baud rate disabled

To disable the override serial port configuration baud rate OVERRIDE and BAUD_RATE is set to 0 (zero).

9.9.1 Read override serial port configuration baud rate request (0xD8)

Request to read the override serial port configuration baud rate.

Command	Length	Command Parameter 1
0xD8	0x00	-

9.9.2 Read override serial port configuration baud rate response (0xD8)

If STATUS is success, the override serial port configuration baud rate is returned otherwise MODE and BAUD_RATE will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2	Command Parameter 3
0xD8	0x04	STATUS	OVERRIDE	BAUD_RATE

9.9.3 Set override serial port configuration baud rate request (0xD9)

Request to set the override serial port configuration baud rate.

Command	Length	Command Parameter 1	Command Parameter 2
0xD9	0x03	OVERRIDE	BAUD_RATE

9.9.4 Set override serial port configuration baud rate response (0xD9)

If STATUS is success, the override serial port configuration baud rate has been stored in persistent memory.

Command	Length	Command Parameter 1
0xD9	0x01	STATUS

9.10 Quality of service setup

The Link Manager of the WU firmware provides Quality of Service (QoS) capabilities. “Best effort” and “Guaranteed” service types are supported.

A poll interval, which is defined as the maximum time between subsequent transmissions from the master to a particular slave on the ACL link, is used to support bandwidth allocation and latency control. The poll interval is guaranteed in the active mode except when there are collisions with page, page scan, inquiry and inquiry scan. The poll interval is also known as Tpoll.

Connections are given a “Best effort” service type with a Tpoll value of 40 slots (25msec), by default, when created by the master running WU firmware.

If a different service type and/or Tpoll value for the established connection is desired, a QoS_Setup command must be sent to request other parameters to be used instead. The *“Set quality of service setup request (0xDB)”* command is used to specify the parameters that will be used in the QoS_Setup command. The QoS_Setup command (if enabled) will be sent when a successful connection is established and can be sent by the master as well as the slave. The requested parameters can be rejected, e.g., if QoS_Setup is sent on a slave the change may be rejected by the master. The Host can only set the requested parameters for the link, it does not know the actual parameters negotiated.

Being able to adjust the link’s Tpoll value allows reduction of the maximum latency for starting data transfers from slave to master at the cost of extra polling (and thus power consumption) on the master.

The SERVICE_TYPE and LATENCY parameters of the QoS_Setup command can be configured, the other parameters are not configurable and are set to its default values.

The command's LATENCY parameter is translated from microseconds to piconet slots (rounded down) and used in an attempt to set the connection's Tpoll value. The Tpoll value is also subjected to some limits.

The command's SERVICE_TYPE parameter indicates the level of service required.

If "Best effort" is selected, the service type does not require any guarantees. The fields of the QoS_Setup should be treated as hints by the remote device. The remote device may choose to ignore the fields, try to satisfy the hint or respond with the settings it will try to meet.

If "Guaranteed" is selected, the remote device will "guarantee" the latency. This allows the master to boost the connection's priority when using sniff mode, allowing for short, tight sniff timing.

Parameter description:

SERVICE_TYPE: Size: 1 byte

Value	Parameter Description
0x00	Quality of service setup disabled
0x01	Best effort
0x02	Guaranteed
Default = 0x00	Quality of service setup disabled

LATENCY: Size: 4 bytes

Value	Parameter Description
0x00000000	Quality of service setup disabled
0x00000001 – 0x000F4240	Latency in microseconds Time Range: 1usec – 1sec
Default = 0x00000000	Quality of service setup disabled

To disable the use of quality of service setup SERVICE_TYPE and LATENCY must be set to 0 (zero).

9.10.1 Read quality of service setup request (0xDA)

Request to read the quality of service setup.

Command	Length	Command Parameter 1
0xDA	0x00	-

9.10.2 Read quality of service setup response (0xDA)

If STATUS is success, the quality of service setup is returned otherwise SERVICE_TYPE and LATENCY will be set to 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2	Command Parameter 3
0xDA	0x06	STATUS	SERVICE_TYPE	LATENCY

9.10.3 Set quality of service setup request (0xDB)

Request to set the quality of service setup.

Command	Length	Command Parameter 1	Command Parameter 2
0xDB	0x05	SERVICE_TYPE	LATENCY

9.10.4 Set quality of service setup response (0xDB)

If STATUS is success, the quality of service setup has been stored in persistent memory.

Command	Length	Command Parameter 1
0xDB	0x01	STATUS

9.11 Extended DCE control signals

The WU firmware can be configured to enable extended control signals for DCE devices.

The following extended DCE control signals mode commands are only supported on the F2M01SXA. It is a DCE device with DTR and DSR pin available in the D-Sub connector.

All other attempts to modify these parameters will fail.

When the extended control signals are configured for Emulate Serial Handshake the configuration parameters available under *"Emulate serial handshaking lines"* is used.

When the extended control signals are configured for Bluetooth control the PIO_DISCONNECT and PIO_CONNECTED functionality is enabled on the corresponding pin of the D-Sub.

DCD_MODE (N/A in F2M01SXA): Size: 1 byte

Value	Parameter Description
0x00	Emulate serial handshaking line
0x01	Bluetooth connection controlled line
0x02	Output logic '0' (+12V RS232)
0x03	Output logic '1' (-12V RS232)
Default = 0x00	Emulate serial handshaking line

DSR_MODE: Size: 1 byte

Value	Parameter Description
0x00	Emulate serial handshaking line
0x01	Bluetooth connection controlled line (PIO_CONNECTED)
0x02	Output logic '0' (+12V RS232)
0x03	Output logic '1' (-12V RS232)
Default = 0x00	Emulate serial handshaking line

DTR_MODE: Size: 1 byte

Value	Parameter Description
0x00	Emulate serial handshaking line
0x01	Bluetooth connection controlled line (PIO_DISCONNECT)
0x02	Input don't care
Default = 0x00	Emulate serial handshaking line

9.11.1 Read extended DCE control signals mode request (0xDC)

Request to read the extended DCE control signals mode.

Command	Length	Command Parameter 1
0xDC	0x00	-

9.11.2 Read extended DCE control signals mode response (0xDC)

If STATUS is success, the extended DCE control signals is returned otherwise DCD_MODE, DSR_MODE and DTR_MODE will be set to 0 (zero).

Command	Length	Parameter 1	Parameter 2	Parameter 3	Parameter 4
0xDC	0x04	STATUS	DCD_MODE	DSR_MODE	DTR_MODE

9.11.3 Set extended DCE control signals mode request (0xDD)

Request to set the extended DCE control signals.

Command	Length	Parameter 1	Parameter 2	Paramter 3
0xDD	0x03	DCD_MODE	DSR_MODE	DTR_MODE

9.11.4 Set extended DCE control signals mode p response (0xDD)

If STATUS is success, the extended DCE control signals has been stored in persistent memory.

Command	Length	Command Parameter 1
0xDD	0x01	STATUS

9.12 Low power mode

The WU firmware implements a low power state machine that can be enabled to save power on low data rate links.

The low power state machine takes the form of a table of successive power mode(s) that it moves through over periods of increasing inactivity. If at anytime there is activity on the link, control will be reset to the first power state. Up to four different power states can be specified in the table.

The ENTRIES parameter defines the number of power state(s) present in the table.

Parameter description:

ENTRIES: Size: 1 byte

Value	Parameter Description
0x00-0x04	Number of power state(s)
Default = 0x00	No power states

Each power state in the table must be defined using the following parameter notation:

POWER_STATE: Size: 11 bytes

Par 1	Par 2	Par 3	Par 4	Par 5	Par 6
MODE	MIN_INTERVAL	MAX_INTERVAL	ATTEMPT	TIMEOUT	DURATION

The MODE parameter defines the Bluetooth power mode.

The MIN_INTERVAL, MAX_INTERVAL, ATTEMPT and TIMEOUT parameters define the parameters of the selected Bluetooth power mode.

The DURATION is the length of time, in seconds, that the low power state machine will remain in this state. When this period expires, control moves to the next state.
Note: The last power state in the table must have an infinite duration (zero).

Power state parameter description:

MODE: Size: 1 byte

Value	Parameter Description
0x00	Active mode
0x01	Sniff mode
0xFF	Passive mode Passive mode is a “don’t care” setting where the local device will not attempt to alter the power mode

MIN_INTERVAL: Size: 2 bytes

Value	Parameter Description
N = 0x0001 – 0xFFFF	Minimum acceptable number of Baseband slots between each sniff period. (MAX_INTERVAL >= MIN_INTERVAL) Length = N * 0.625msec (1 Baseband slot) Range for N: 0x0001 – 0xFFFF Time Range: 0.625msec – 40.9sec

MAX_INTERVAL: Size: 2 bytes

Value	Parameter Description
N = 0x0001 – 0xFFFF	Maximum acceptable number of Baseband slots between each sniff period. (MAX_INTERVAL >= MIN_INTERVAL) Length = N * 0.625msec (1 Baseband slot) Range for N: 0x0001 – 0xFFFF Time Range: 0.625msec – 40.9sec

ATTEMPT: Size: 2 bytes

Value	Parameter Description
N = 0x0001 – 0x7FFF	Number of Baseband receive slots for sniff attempt. Length = (2 * N – 1)* 0.625msec Range for N: 0x0001 – 0x7FFF Time Range: 0.625msec – 40.9sec

TIMEOUT: Size: 2 bytes

Value	Parameter Description
N = 0x0000 – 0x7FFF	Number of Baseband receive slots for sniff timeout. Length = (2 * N –1) * 0.625msec if N > 0, Length = 0msec if N = 0 Range for N: 0x0000 – 0x7FFF Time Range: 0msec – 40.9sec

DURATION: Size: 2 bytes

Value	Parameter Description
0x0000-0xFFFF	Time in seconds to spend in this mode

The low power state machine supports active mode and sniff mode. Park mode is currently not supported.

In active mode no power saving is applied. MIN_INTERVAL, MAX_INTERVAL, ATTEMPT and TIMEOUT must be set to 0 (zero).

In sniff mode the traffic is reduced to periodic sniff slots, where the device wakes up periodically to listen for and transmit data (if any).

For sniff mode to be applied both the WU unit and the remote device must support it. If one of them does not support sniff mode they will not be allowed to go into low power mode.

Both master and slave can request that a connection shall enter sniff mode. The sniff mode parameters are negotiated so that the devices at each end of the connection share the same parameters for sniff mode.

When one side requests sniff mode, the other side has three choices: it can accept the sniff mode request with the proposed parameters, it can reject the sniff mode request, or it can return the request with different timing parameters. The sniff mode request messages can be sent back and forth until one side agrees to the proposed sniff mode parameters.

The sniff MAX_INTERVAL and sniff MIN_INTERVAL parameters are used to specify the requested acceptable maximum and minimum periods in the sniff interval. The sniff MIN_INTERVAL parameter cannot be greater than the sniff MAX_INTERVAL parameter. The sniff interval defines the amount of time between each consecutive sniff period. The Host can only set the requested minimum and maximum values for the sniff interval, it does not know the actual sniff interval negotiated.
Note: The sniff MAX_INTERVAL must be less than the *“Link supervision timeout”* configuration parameter.

In sniff mode the duty cycle of the slave’s listen activity can be reduced. When a connection is established, it has to listen in every slot to the master’s traffic. With sniff mode the time slots where the master can start transmission to a specific slave can be reduced. The master can only start transmission in specified time slots to the slave. These so called sniff slots are spaced regularly with a sniff interval. Two parameters control the listening activity in the slave: the sniff ATTEMPT and the sniff TIMEOUT. The slave starts listening at the sniff slots for N sniff ATTEMPT consecutive receive slots unless a packet with matching active member address is received. After every reception of a packet with matching active member address, the slave continues listening at the subsequent N sniff TIMEOUT or remaining of the receive slots, whichever is greater.

Note:

- For N sniff TIMEOUT > 0, the slave continues listening as long as it receives packets with matching active member address.
- That N sniff ATTEMPT = 1 and N sniff TIMEOUT = 0 cause the slave to listen only at the first sniff slot, irrespective of packets received from the master.
- That N sniff ATTEMPT = 0 is not allowed.

If the Host sends data to the WU unit when the connection is in sniff mode, the WU firmware will keep the data in its buffers until either the data can be transmitted or when a disconnection occurs.

There is a trade-off between the intervals selected and the current consumption. Longer intervals will reduce the current consumption. However, they will also increase the latency in communicating with the remote device. Users may perceive this as a lack of responsiveness.

There are no optimal default intervals that are suitable in every design. It is therefore up to the user to determine the intervals that is considered the most appropriate. See Bluetooth specification for more information.

9.12.1 Power state example 1

No power saving mode.

This is equivalent to sending a *“Set low power table request (0xDF)”* with LENGTH set to 1 (one) and ENTRIES set to 0 (zero):

MODE	MIN_INTERVAL	MAX_INTERVAL	ATTEMPT	TIMEOUT	DURATION
ACTIVE	0	0	0	0	0

9.12.2 Power state example 2

Stay active for 5 seconds and then attempt to enter sniff mode with the specified parameters.

MODE	MIN_INTERVAL	MAX_INTERVAL	ATTEMPT	TIMEOUT	DURATION
ACTIVE	0	0	0	0	5
SNIFF	32	200	1	8	0

9.12.3 Power state example 3

Enter sniff mode immediately using a short sniff interval. After 1 second attempt to enter sniff mode using a slightly longer sniff interval (for greater power save). After 30 seconds in this mode attempt to enter sniff mode using an even longer interval (for even greater power save due to long idleness of the link). And so forth.

MODE	MIN_INTERVAL	MAX_INTERVAL	ATTEMPT	TIMEOUT	DURATION
SNIFF	20	52	1	1	1
SNIFF	54	162	1	16	30
SNIFF	164	402	1	16	600
SNIFF	404	802	1	16	0

9.12.4 Read low power table request (0xDE)

A request to read the low power table configuration.

Command	Length	Command Parameter 1
0xDE	0x00	-

9.12.5 Read low power table response (0xDE)

If STATUS is success, the low power table is returned otherwise LENGTH will be 2 (two) and ENTRIES will be set to 0 (zero) meaning that no low power entries are present. For detailed information about the PSn entries please refer to the POWER_STATE definition above.

Cmd	Len	Cmd Par 1	Cmd Par 2	Cmd Par 3	Cmd Par 4	Cmd Par 5	Cmd Par 6
0xDE	0x02, 0x0D, 0x18, 0x23, 0x2E	STATUS	ENTRIES	PS1 (if present)	PS2 (if present)	PS3 (if present)	PS4 (if present)

9.12.6 Set low power table request (0xDF)

A request to change the low power table configuration. For detailed information about the PSn entries please refer to the POWER_STATE definition above.

Cmd	Len	Cmd Par 1	Cmd Par 2	Cmd Par 3	Cmd Par 4	Cmd Par 5
0xDF	0x01, 0x0C, 0x17, 0x22, 0x2D	ENTRIES	PS1 (if present)	PS2 (if present)	PS3 (if present)	PS4 (if present)

9.12.7 Set low power table response (0xDF)

If STATUS is success, the new low power table has been stored in persistent memory.

Command	Length	Command Parameter 1
0xDF	0x01	STATUS

10 Synchronous Connection Commands

Note: Before any synchronous connection commands can be sent to the WU firmware, it must be set in *"Host Controlled Mode"*.

The following synchronous connection commands are only supported on the F2M03 product family. All other attempts to modify these parameters will fail.

The F2M03 module with WU firmware can establish a Synchronous Connection Oriented (SCO) link or an Extended Synchronous Connection Oriented (eSCO) link used primarily for voice traffic to a remote WU device. Once the SCO/eSCO connection is successfully established, the WU unit applies its local input gain and output gain parameters to control the audio level of the CODEC. See *"Setup synchronous connection command"* and *"CODEC Gain Control"* below for more information about establishing a SCO/eSCO connection and controlling the audio level of the CODEC.

The data sent to and received from a SCO/eSCO connection is routed directly via the hardware in the F2M03 module. Depending on which F2M03 module (table below) being used for the SCO/eSCO connection, the SCO/eSCO data is routed over different hardware interfaces. SCO/eSCO data will either be routed over the Pulse Code Modulation (PCM) interface, for F2M03 products that require an external CODEC, or directly over the microphone and speaker interface, for F2M03 products with internal CODEC.

Audio interface:

Free2move Product	Hardware Interface
F2M03GX	PCM
F2M03GXA	PCM
F2M03GLA	PCM
F2M03MLA	Analogue
F2M03ALA	Analogue
F2M01SXA	None

Free2move products using the PCM interface (see product datasheet for detailed information) are configured as Master of the PCM interface with the following settings:

PCM hardware interface:

PCM I/O	Signal Direction	Value	Description
PCM_IN	Input	13-bit linear 2s complement coding	Synchronous data input
PCM_OUT	Output	13-bit linear 2s complement coding	Synchronous data output
PCM_CLOCK	Output	256kHz	Synchronous data clock
PCM_SYNC	Output	8kHz long frame sync	Synchronous data sync

Recommended external CODEC is Winbond W681360R 13-bit linear CODEC.

10.1 Setup synchronous connection command

Note: An attempt to setup a synchronous connection will be performed once a successful connection is established.

The setup synchronous connection command allows the WU firmware to establish a Synchronous Connection Oriented (SCO) link or an Extended Synchronous Connection Oriented (eSCO) link to a remote WU device.

Note: Attempting to establish a SCO/eSCO connection to a remote "non-Free2move" device over the Bluetooth Serial Port Profile may or may not work. It is not required by the Bluetooth specification to support SCO/eSCO connection establishment when using the Bluetooth Serial Port Profile. It depends if the remote vendor's device allow SCO/eSCO connections to be established when running SPP.

Establishing SCO/eSCO link between WU units is a special feature of the WU firmware that allows users to not only replace a serial cable, for UART data transmission, but also to replace an audio-cable wireless. The ACL link and the SCO/eSCO link can be used simultaneously or separately, both are capable of full duplex transmissions.

The SCO link is a 64kbit/s symmetric link and typically supports time-bounded voice traffic. SCO packets are transmitted over reserved intervals. Once the SCO connection is established, both master and slave units may send SCO packets without being polled. SCO data in the SCO packets are never retransmitted.

The eSCO link is a symmetric or asymmetric link. ESCO links offer a number of extensions over the standard SCO links, in that they support a more flexible combination of packet types and selectable data contents in the packets and selectable slot periods, allowing a range of synchronous bit rates to be supported. ESCO links also offer limited retransmission of packets (unlike SCO links where there is no retransmission). If these retransmissions are required they take place in the slot that follow the reserved slots, otherwise the slots may be used for other traffic.

It is possible to enable and disable the setup synchronous connection command. The packet type used for the SCO/eSCO connection can be modified.

The MODE configuration parameter defines if setup synchronous connection command is enabled or disabled.

The TXB parameter defines the transmission bandwidth in octets per seconds. For SCO connections this must be set to 8000 (0x1F40) because the SCO link is 64kbit/s. When set to “don’t care” the resulting transmit bandwidth is set by the connecting device. If the connecting device has set the transmit bandwidth to “don’t care” the firmware automatically selects an appropriate bandwidth.

The RXB parameter defines the receive bandwidth in octets per seconds. For SCO connections this must be set to 8000 (0x1F40) because the SCO link is 64kbit/s. When set to “don’t care” the resulting receive bandwidth is set by the connecting device. If the connecting device has set the receive bandwidth to “don’t care” the firmware automatically selects an appropriate bandwidth.

The MAX_LAT parameter defines the maximum latency in milliseconds. For SCO connections this must be set to “don’t care” (0xFFFF).

The VOICE_SETT parameter defines the air coding format that should be used.

The RET_EFF parameter defines the retransmission effort. For SCO connections this must be set to “no retransmissions” (0x00) or “don’t care” (0xFF) because SCO packets are never retransmitted.

The PKT_TYPE configuration parameter defines the packet type that will be used for the SCO/eSCO connection.

For more detailed information about SCO and eSCO and its configuration parameters please refer to the Bluetooth specification.

When setup synchronous connection command is enabled and *“Operating mode”* is Connecting mode – once a successful connection is established the connecting unit initiates an attempt to establish a synchronous connection using the specified packet type and parameters.

When setup synchronous connection command is enabled and *“Operating mode”* is Endpoint mode – once a successful connection is established the endpoint unit accepts an attempt to setup a synchronous connection using the specified packet type and parameters.

If the SCO/eSCO connection is successfully established, the WU unit applies its local input gain and output gain parameters to control the audio level of the CODEC. See *“CODEC Gain Control”* below for more information about controlling the audio level of the CODEC.

Note: When a SCO/eSCO connection is established the performance regarding throughput (streaming raw UART data) of the WU firmware may decrease. The amount of decrease in throughput depends upon the packet type used for the SCO/eSCO connection.

Parameter description:

MODE: Size: 1 byte

Value	Parameter Description
0x00	Setup synchronous connection command disabled
0x01	Setup synchronous connection command enabled
Default = 0x00	Setup synchronous connection command disabled

TXB: Size: 4 bytes

Value	Parameter Description
0x00000000	Disabled, when MODE is 0x00
0x00000000-0xFFFFFFFE	Transmit bandwidth in octets per second
0xFFFFFFFF	Don’t care
Default = 0x00000000	Disabled

RXB: Size: 4 bytes

Value	Parameter Description
0x00000000	Disabled, when MODE is 0x00
0x00000000-0xFFFFFFFE	Receive bandwidth in octets per second
0xFFFFFFFF	Don’t care
Default = 0x00000000	Disabled

MAX_LAT: Size: 2 bytes

Value	Parameter Description
0x0000	Disabled, when MODE is 0x00
0x0000-0x0003	Reserved
0x0004-0xFFFE	Maximum latency in milliseconds
0xFFFF	Don’t care
Default = 0x0000	Disabled

VOICE_SETT: Size: 2 bytes

Value	Parameter Description
0x0000	Disabled, when MODE is 0x00
0x0000	CVSD
0x0001	µ-law
0x0002	A-law
0x0003	Transparent data
Default = 0x0000	Disabled

RET_EFF: Size: 1 byte

Value	Parameter Description
0x00	Disabled, when MODE is 0x00
0x00	No retransmissions
0x01	At least one retransmission, optimize for power consumption
0x02	At least one retransmission, optimize for link quality
0x03-0xFE	Reserved
0xFF	Don’t care
Default = 0x00	Disabled

PKT_TYPE: Size: 2 bytes

Value	Parameter Description
0x0000	Disabled, when MODE is 0x00
0x0001	HV1 (SCO)
0x0002	HV2 (SCO)
0x0004	HV3 (SCO)
0x0007	All SCO (HV3|HV2|HV1)
0x0008	EV3 (eSCO)
0x0010	EV4 (eSCO)
0x0020	EV5 (eSCO)
0x0038	All eSCO (EV5|EV4|EV3)
0x0040	2-EV3 (EDR eSCO)
0x0080	3-EV3 (EDR eSCO)
0x0100	2-EV5 (EDR eSCO)
0x0200	3-EV5 (EDR eSCO)
0x03C0	All EDR eSCO
0x03FF	All packet types
Default = 0x0000	Disabled

To disable the setup synchronous connection command MODE and all other parameters is set to 0 (zero).

10.1.1 Read setup synchronous connection command request (0xF0)

Request to read the setup synchronous connection command parameters.

Command	Length	Command Parameter 1
0xF0	0x00	-

10.1.2 Read setup synchronous connection command response (0xF0)

If STATUS is success, the setup synchronous connection command parameters are returned otherwise MODE and all other parameters will be set to 0 (zero).

Cmd	Len	Cmd Par 1	Cmd Par 2	Cmd Par 3	Cmd Par 4	Cmd Par 5	Cmd Par 6	Cmd Par 7	Cmd Par 8
0xF0	0x11	STATUS	MODE	TXB	RXB	MAX_LAT	VOICE_SETT	RET_EFF	PKT_TYPE

10.1.3 Set setup synchronous connection command request (0xF1)

A request to set the setup synchronous connection command parameters.

Cmd	Len	Cmd Par 1	Cmd Par 2	Cmd Par 3	Cmd Par 4	Cmd Par 5	Cmd Par 6	Cmd Par 7
0xF1	0x10	MODE	TXB	RXB	MAX_LAT	VOICE_SETT	RET_EFF	PKT_TYPE

10.1.4 Set setup synchronous connection command response (0xF1)

If STATUS is success, the setup synchronous connection command parameters have been stored in persistent memory.

Command	Length	Command Parameter 1
0xF1	0x01	STATUS

10.2 CODEC Gain Control

When *"Setup synchronous connection command"* is enabled, an attempt to setup a SCO/eSCO connection is performed once a successful connection is established. If the SCO/eSCO connection is successfully established, the WU unit applies its local input gain and output gain parameters to control the audio level of the CODEC. Input gain is used to control the gain of the internal input CODEC on those Free2move products that includes it (see table below). Output gain is used to control the gain of either the internal output CODEC on those Free2move products that includes it (see table below) or the external CODEC.

CODEC gain control:

Free2move Product	Internal Input CODEC	Input Gain Control	Internal Output CODEC	Output Gain Control
F2M03GX	No	No	No	Yes
F2M03GXA	No	No	No	Yes
F2M03GLA	No	No	No	Yes
F2M03MLA	Yes	Yes	Yes	Yes
F2M03ALA	Yes	Yes	Yes	Yes
F2M01SXA	No	No	No	No

Legal values for input GAIN are from 0 (zero) up to, and including, the value RANGE returned in *"Read CODEC input gain range response (0xF5)"*. If 0 (zero) is returned, CODEC control is not possible.

Legal values for output GAIN are from 0 (zero) up to, and including, the value RANGE returned in *"Read CODEC output gain range response (0xF2)"*. If 0 (zero) is returned, CODEC control is not possible.

A GAIN value of 0 (zero) means minimum gain.

Parameter description:

RANGE: Size: 2 bytes

Value	Parameter Description
0x0000	CODEC input/output gain control not possible
0x0001-0xFFFF	Supported range of values for input/output gain

GAIN: Size: 2 bytes

Value	Parameter Description
0x0000-RANGE	Desired input/output gain
Default (F2M03G) = 0x0000	Input gain not possible
Default (F2M03G) = 0x0004	Output gain

10.2.1 Read CODEC output gain range request (0xF2)

Request to read the supported range of values for the output gain of the CODEC.

Command	Length	Command Parameter 1
0xF2	0x00	-

10.2.2 Read CODEC output gain range response (0xF2)

Returns the range of values for the output gain of the CODEC. If STATUS is success then the values from 0 (zero) up to RANGE are permitted in *"Set CODEC output gain request (0xF4)"*. If 0 (zero) RANGE is returned, output gain control is not possible. If STATUS is not success RANGE will be 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xF2	0x03	STATUS	RANGE

10.2.3 Read CODEC output gain request (0xF3)

Request to read the CODEC output gain value used when a successful SCO/eSCO connection has been established.

Command	Length	Command Parameter 1
0xF3	0x00	-

10.2.4 Read CODEC output gain response (0xF3)

If STATUS is success, the CODEC output gain value is returned otherwise RANGE will be 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xF3	0x03	STATUS	RANGE

10.2.5 Set CODEC output gain request (0xF4)

Request to set the CODEC output gain value used when a successful SCO/eSCO connection has been established. Legal values for the CODEC output gain are from 0 (zero) up to, and including, the value in RANGE returned by *"Read CODEC output gain range response (0xF2)"*.

Command	Length	Command Parameter 1
0xF4	0x02	GAIN

10.2.6 Set CODEC output gain response (0xF4)

If STATUS is success, the CODEC output gain has been stored in persistent memory.

Command	Length	Command Parameter 1
0xF4	0x01	STATUS

10.2.7 Read CODEC input gain range request (0xF5)

Request to read the supported range of values for the input gain of the CODEC.

Command	Length	Command Parameter 1
0xF5	0x00	-

10.2.8 Read CODEC input gain range response (0xF5)

Returns the range of values for the input gain of the CODEC. If STATUS is success then the values from 0 (zero) up to RANGE are permitted in *"Set CODEC input gain request (0xF7)"*. If 0 (zero) RANGE is returned, input gain control is not possible. If STATUS is not success RANGE will be 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xF5	0x03	STATUS	RANGE

10.2.9 Read CODEC input gain request (0xF6)

Request to read the CODEC input gain value used when a successful SCO/eSCO connection has been established.

Command	Length	Command Parameter 1
0xF6	0x00	-

10.2.10 Read CODEC input gain response (0xF6)

If STATUS is success, the CODEC input gain value is returned otherwise RANGE will be 0 (zero).

Command	Length	Command Parameter 1	Command Parameter 2
0xF6	0x03	STATUS	RANGE

10.2.11 Set CODEC input gain request (0xF7)

Request to set the CODEC input gain value used when a successful SCO/eSCO connection has been established. Legal values for the CODEC input gain are from 0 (zero) up to, and including, the value in RANGE returned by *"Read CODEC input gain range response (0xF5)"*.

Command	Length	Command Parameter 1
0xF7	0x02	GAIN

10.2.12 Set CODEC input gain response (0xF7)

If STATUS is success, the CODEC input gain has been stored in persistent memory.

Command	Length	Command Parameter 1
0xF7	0x01	STATUS

11 Wireless UART message sequence chart examples

This chapter is intended to give the Wireless UART user an overview of the message exchange sequence between the Host and the WU unit.

For each example it is recommended that the user refer to the associated section(s) of this document for detailed information and greater understanding about the messages exchanged.

11.1 Enter Host controlled mode

Before any configuration commands are accepted by the Wireless UART (WU) firmware, it must be set in *"Host Controlled Mode"* (HCM) first.

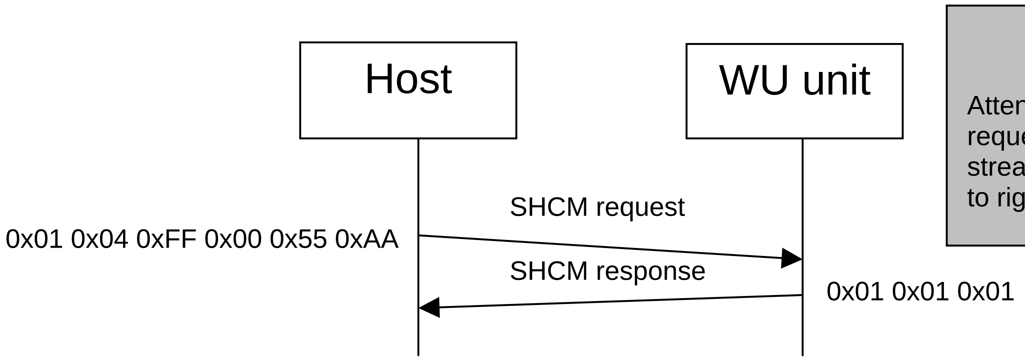
The following example shows the message exchange sequence for entering HCM.

It is important to remember that it is only possible to configure the WU firmware via WU protocol commands as long as there is no Bluetooth connection established.

The following serial settings are used for HCM configuration mode:

Parameter	Default Value
Baud rate	38400
Data bits	8
Parity	None
Stop bits	1
Hardware flow control	On

To enter HCM a *"Switch to Host controlled mode request (0x01)"* (SHCM request) command must be sent from the Host to the WU unit, which answers with a *"Switch to Host controlled mode response (0x01)"* (SHCM response) when the WU unit has successfully changed operating mode to HCM.



!
Attention: Bytes in the request/response data streams are sent from left to right.

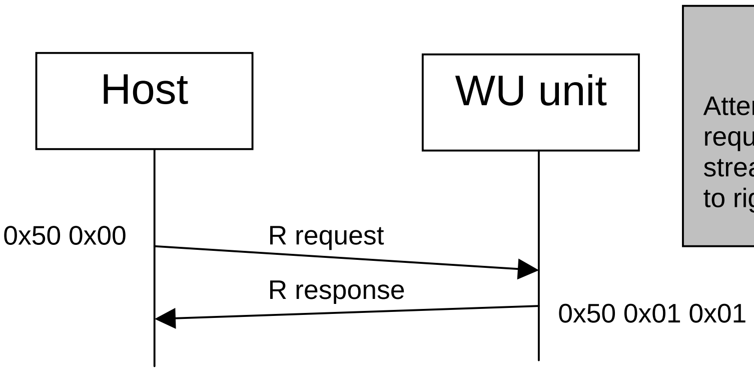
Message sequence chart – enter Host controlled mode.

11.2 Exit Host controlled mode

To exit HCM and apply the new settings the *“Run”* command in the *“Software / Hardware Reboot Commands”* section must be sent. All settings are stored in persistent memory.

The following example shows the message exchange sequence to exit HCM. In this example it is assumed that the WU unit is already in HCM.

The *“Run request (0x50)”* (R request) command forces the WU unit to restart and run in normal execution mode, i.e., Endpoint mode or Connecting mode. The result is that the HCM is exited and the new settings will be applied and *“Run response (0x50)”* (R response) is sent by the WU unit.



!

Attention: Bytes in the request/response data streams are sent from left to right.

Message sequence chart – exit Host controlled mode.

11.3 Operating mode

The following example shows the message exchange sequence to configure the operating mode of the WU unit.

There are two normal operating modes:

- Connecting mode – Bluetooth master
- Endpoint mode – Bluetooth slave

The *“Set operating mode request (0x13)”* (SOM request) command, described in the *“Operating mode”* section, either sets the WU unit in Connecting mode or Endpoint mode.

In this example the WU unit is configured for Endpoint mode:

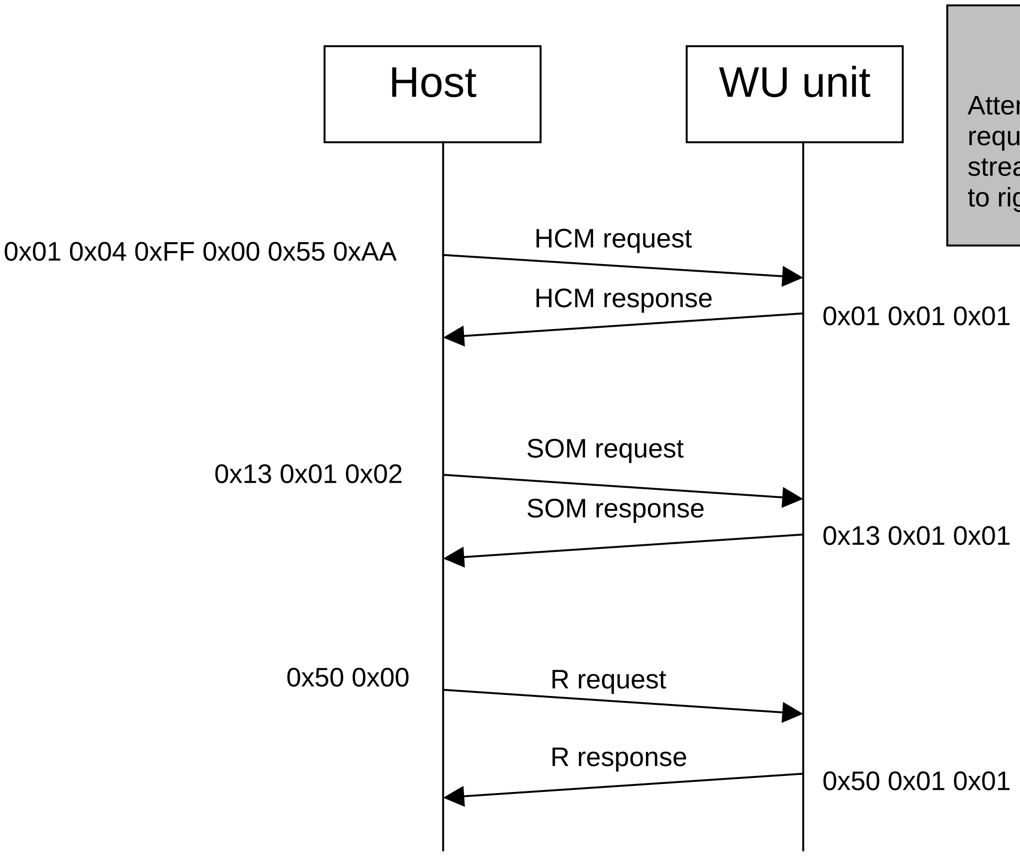
OPERATING_MODE: Size: 1 byte

Value	Parameter Description
0x02	Endpoint Mode

The WU is first set in HCM, the upper request/response exchange in the figure below. Then the actual set operating mode request/response exchange is performed, the middle request/response exchange. In order to restart and enable the new setting, the run request/response exchange is performed, the lower request/response exchange.

!

Attention: Bytes in the request/response data streams are sent from left to right.



Message sequence chart – configuring the operation mode.

11.4 Serial port configuration

The following example shows the message exchange sequence for changing the UART settings of the WU unit.

Note: The serial settings used for configuration are not related to the serial settings specified in the *"Serial port configuration"* command. As soon as a successful Bluetooth connection has been established the serial settings specified in the *"Serial port configuration"* will be used.

The *"Set serial port configuration request (0x15)"* (SSPC request) command, described in the *"Serial port configuration"* section, configures the WU unit's UART settings.

In this example the UART settings are configured to:

BAUD_RATE: Size: 1 byte

Value	Parameter Description (bps)
0x03	9600

FLOW_CONTROL: Size: 1 byte

Value	Parameter Description
0x01	Hardware flow control ON

PARITY: Size: 1 byte

Value	Parameter Description
0x00	No parity

DATA_BITS: Size: 1 byte

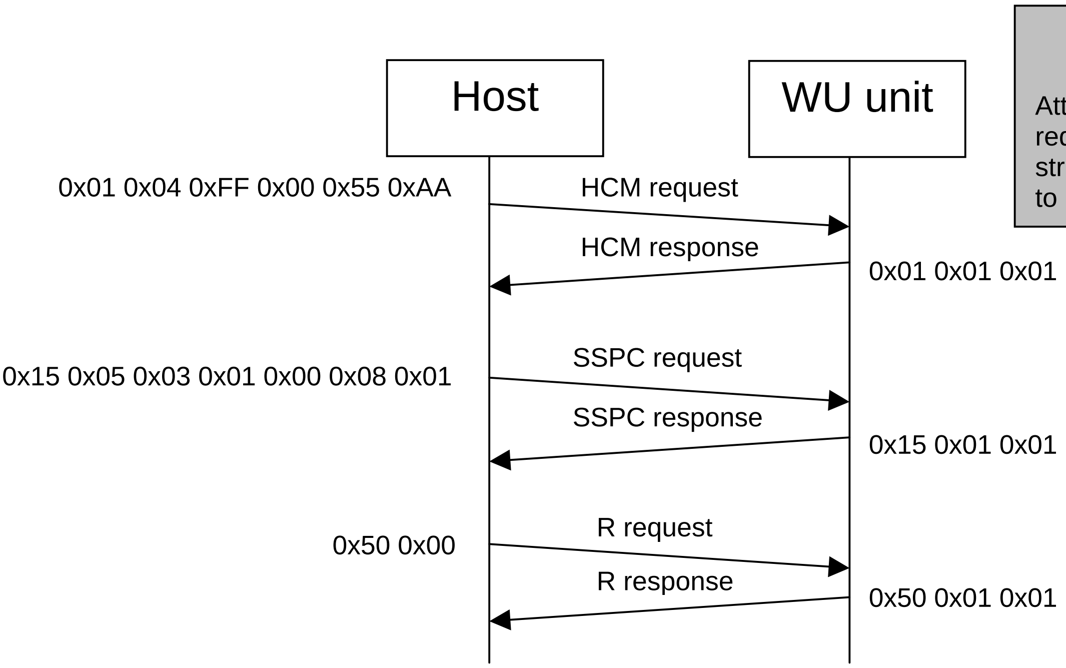
Value	Parameter Description
0x08	Number of data bits to use (Not configurable at the moment)

STOP_BITS: Size: 1 byte

Value	Parameter Description
0x01	1 stop bit

!

Attention: Bytes in the request/response data streams are sent from left to right.



Message sequence chart – serial port configuration.

12 ASCII controlled mode

New for version 4.02 of Wireless UART is a ASCII controlled mode which will enable ASCII commands for controlling the firmware during the whole connection procedure.

The ASCII mode uses the same settings used under Host Controlled Mode. For example if the data baud rate is configured for 115200 bps it will use this baud rate also when switching to ASCII-mode. Chapter "*ASCII controlled mode configuration"* describes how to enter ASCII mode.

All commands sent and received when in ASCII mode are ended by a carriage return ('\r').If a command isn't recognized by the firmware, it will answer with "EVENT UNKNOWN COMMAND\r\n".

All commands and arguments are interpreted as ASCII data. When an argument is specified as (text) this only state that the input or output argument is alphabetical ASCII characters. When an argument is specified as (number) it states that the input or output argument is interpreted as an ASCII number.

All command responses and events can be terminated with a "comma"(,) followed by a comment. For example the response to the "DO DISCONNECT\r" command can be "DO DISCONNECT FAIL, Not connected\r\n"

12.1 DO Commands

The DO commands are typically the commands that change

All DO commands sent to the WU will be responded with the status information. If the command is successful the WU will respond with an OK and if the command is unsuccessful the module will respond with FAIL.

12.1.1 CONNECT

Command	Sub command	Argument	Action
DO CONNECT	ADDR	[(text "XXXXXXXXXXXX")]	Connects to the specified address
	PAIRED		Connects to the last paired device
	STORED		Connects to the address stored in Connect rule

Response:

Command	Status
DO CONNECT	OK/FAIL

12.1.2 DISCONNECT

Command	Action
DO DISCONNECT	Disconnect from remote device

Response:

Command	Status
DO DISCONNECT	OK/FAIL

12.1.3 DISCOVER ON

Command	Action
DO DISCOVER ON	Start discovering nearby devices.

Response:

Command	Status
DO DISCOVER ON	OK/FAIL

12.1.4 DISCOVER OFF

Command	Action
DO DISCOVER OFF	Stops active discovery process.

Response:

Command	Status
DO DISCOVER OFF	OK/FAIL

12.1.5 RESET

Soft reset the firmware:

Command	Action
DO RESET	Soft reset the module

Response:

Command	Status
DO RESET	OK

12.1.6 EXIT

Exit ASCII mode:

Command	Action
DO EXIT	Exit ASCII mode and enter HCM.

Response:

Command	Status
DO EXIT	OK

12.1.7 SEND

The SEND command uses the Serial Port Profile for sending argument X to the remote device.

Command	Sub command	Argument	Action
DO SEND	STRING	[(text X)]	Sends string X of data to the connected SPP device
DO SEND	BYTE	[(number X 0-255)]	Sends character represented as unsigned 8 bit integer value

Response:

Command	Status
DO SEND	OK/FAIL

12.2 GET commands

The following commands are used to retrieve information from the WU firmware or the connected device.

The GET commands will either return status OK or FAIL. If status is OK the requested argument is returned within brackets after the requested command. All commands can return a trailing comment after a comma sign.

12.2.1 ADDR

Read the local Bluetooth address:

Command
GET ADDR

Response:

Command	Status	Argument, Bluetooth address
GET ADDR	OK/FAIL	[(text ”XX:XX:XX:XX:XX:XX”)]

12.2.2 REMOTE ADDR

Read remote Bluetooth address from the current connection:

Command
GET REMOTE ADDR

Response:

Command	Status	Argument
GET REMOTE ADDR	OK/FAIL	[(text ”XX:XX:XX:XX:XX:XX”)]

12.2.3 REMOTE NAME

Read remote Bluetooth name from the current connection or address:

Command	Sub command	Argument, Bluetooth address
GET REMOTE NAME		
GET REMOTE NAME	ADDR	[(text ”XXXXXXXXXXXX”)]

Response:

Command	Status	Argument
GET REMOTE NAME	OK/FAIL	[(text)]

12.2.4 FIRMWARE

Read firmware name.

Command
GET FIRMWARE

Response:

Command	Status	Argument, Firmware name
GET FIRMWARE	OK	[(text)]

12.2.5 SWVERSION

Read software version of firmware:

Command
GET SWVERSION

Response:

Command	Status	Argument, Software version
GET SWVERSION	OK	[(text)]

12.2.6 HWVERSION

Read hardware version of the Bluetooth module:

Command
GET HWVERSION

Response:

Command	Status	Argument, Hardware version
GET HWVERSION	OK/FAIL	[(text)]

12.2.7 ARTICLE

Read article number of product:

Command
GET ARTICLE

Response:

Command	Status	Argument, Article number
GET ARTICLE	OK/FAIL	[(text)]

12.2.8 FCC

Read FCC id of product:

Command
GET FCC

Response:

Command	Status	Argument, FCC ID
GET FCC	OK	[(text)]

12.2.9 QDID

Read Bluetooth Qualified Design Identification (QDID) of product:

Command
GET QDID

Response:

Command	Status	Argument, Bluetooth QD ID
GET QDID	OK	[(text)]

12.2.10 HARDWARE

Read name of Bluetooth product:

Command
GET HARDWARE

Response:

Command	Status	Argument, Product ID	Comment, Product name
GET HARDWARE	OK	[(number)]	(text)

Product ID	Product name
8	F2M03GX
9	F2M03GXA
10	F2M03GLA
11	F2M03MLA
12	F2M03ALA
13	F2M03MX
14	F2M01SXA

12.2.11 STATE

Read the current state of the Bluetooth Profile:

Command
GET STATE

Response:

Command	Status	Argument, State ID	Comment, State name
GET STATE	OK	[(number 0 - 7)]	(text)

State ID	State name
0	Initialising
1	Ready
2	Discovering
4	Connecting
5	Connected
6	Disconnecting
7	Disconnected

12.2.12 RSSI

Read the Receive Signal Strength Indicator(RSSI) value of the current Bluetooth connection. The value can be from –128 to 127. The RSSI value is only present if a connection is established.

Command
GET RSSI

Response:

Command	Status	Argument, RSSI value
GET RSSI	OK/FAIL	[(number -128 – 127)]

12.2.13 LINKQUALITY

Read the Link Quality value from the current Bluetooth connection. The link quality is a value of the current Bit Error Rate (BER). The LINKQUALITY value is only present if a connection is established.

Command
GET LINKQUALITY

Response:

Command	Status	Argument, Link quality value
GET LINKQUALITY	OK/FAIL	[(number 0-255)]

As a rule of thumbs, a link with a BER between 0 and 0.1% is workable. Trying to run a link with a BER above 1% gives poor results. The table below shows the different value representation of the BER reading.

Between 255 and 215 each digit represent 0.0025% BER. Thus:

Value	BER
255	0.0
254	0.0025%
253	0.0050%
252	0.0075%
etc., until…	
215	0.1%

Between 214 and 90 each digit represent 0.08% BER. Thus:

Value	BER
215	0.1%
214	0.18%
213	0.26%
etc., until…	
90	2.1%

Between 89 and 0 each digit represent 0.64% BER. Thus:

Value	BER
90	2.1%
89	2.74%
88	3.38%
etc., until…	
0	59.7%

12.3 EVENT Indications

12.3.1 ENTER

The ENTER event is received when entering ASCII mode.

Event
EVENT ENTER

12.3.2 STATE

The STATE event is received on every state change. For more information see the *“GET STATE”* command.

Event	Argument, State ID	Comment, State name
EVENT STATE	[(number 0 - 7)]	(text)

12.3.3 RECEIVE

The RECEIVE event is received when the WU firmware receives a data packet from the remote device.

Event	Argument 1, Length	Argument 2, Data
EVENT RECEIVE	[(number) ,	(text)]

12.3.4 DISCOVER ADDR

Event received upon successful discovery of a remote device.

Event	Argument, Address
EVENT DISCOVER ADDR	[(text ”XX:XX:XX:XX:XX:XX”)]

12.3.5 DISCOVER FINISHED

Event received upon discovery timeout.

Event
EVENT DISCOVER FINISHED

13 Revision History

13.1 v4.02 changes relative to v.4.01 release

1) Wireless UART v.4.02 is currently supported on the following Free2move products:

- F2M03GX – General purpose, extended range module with U.FL-connector (Class 1)
- F2M03GXA – General purpose, extended range module with antenna (Class 1)
- F2M03GLA – General purpose, low power module with antenna (Class 1)
- F2M03MLA – Multimedia, Low power module with Antenna. (Class1)
- F2M03ALA – Audio, Low power module with Antenna. (Class2)
- F2M01SXA – Serial Port Plug Gen 2 (Class1)

2) Added new Host controlled mode configuration commands:

- Switch to ASCII-mode
- Automatically switch to ASCII-mode

3) Added new configuration commands:

- Profile mode
- LED indication mode

4) Added new pairing and security command:

- Set paired unit

5) Added new advanced configuration command:

- Extended DCE control signals

13.2 v4.01 changes relative to v4.00 release

None

13.3 v4.00 changes relative to v3.04 release

1) The Wireless UART application now runs on top of an embedded Bluetooth v2.0 + EDR compliant stack.

2) Wireless UART v4.xx is currently supported on the following Free2move products:

- F2M03GX – General purpose, extended range module with U.FL-connector (Class 1)
- F2M03GXA – General purpose, extended range module with antenna (Class 1)
- F2M03GLA – General purpose, low power module with antenna (Class 1)

3) Added new Host controlled mode configuration command:

- Allowed to switch to Host controlled mode configuration

4) Added new configuration commands:

- Connection established notification mode
- Connection closed notification mode
- Remote RFCOMM server channel mode

5) Added new inquiry command:

- Scan for devices

6) Added new pairing and security commands:

- Security mode
- PIN code
- Paired device

7) Added new advanced configuration command:

- Low power mode

8) Added new synchronous connection command:

- Setup synchronous connection command

9) Default “Local Bluetooth name” changed to “Free2move WU”.

The following Wireless UART v3.xx commands are no longer available:

- Inquiry commands:
  - Scan for devices
- Pairing commands:
  - Paired device
  - Pair as slave
  - Pair as master
  - Pairing complete
  - Pairing and security
  - Local pin code
- Advanced configuration commands:
  - Sniff mode
  - Park mode
- SCO connection commands:
  - Add SCO connection

13.4 v3.04 changes relative to v3.02 release

None.

13.5 v3.02 changes relative to v3.01 release

1) Added support for the F2M03AC2 hardware:

- Wireless UART firmware for F2M03AC2 is available

2) Added new advanced configuration commands:

- Quality of service setup

3) Changed default inquiry scan parameters used in Host controlled mode. New parameters:

- Inquiry scan interval – 0x0800 (1.28sec)
- Inquiry scan window – 0x0012 (11.25msec)

4) Added new SCO connection commands (F2M03 only):

- Add SCO connection
- CODEC gain control (input/output gain)

13.6 v3.01 changes relative to v3.00 release

None.

13.7 v3.00 changes relative to v2.00 release

1) Added new advanced configuration commands:

- Throughput/latency mode
- Link supervision timeout
- Inquiry scan parameters
- Page scan parameters
- Sniff mode
- Park mode
- Class of device
- Default transmit power
- Maximum transmit power
- Emulate serial handshaking lines DTE or DCE (F2M03 only)
- Override serial port configuration baud rate

2) Added new information command:

- Article number

3) Added support for device firmware upgrade (DFU):

- New digitally signed Wireless UART firmware's can be uploaded via the UART interface.

4) Added new STATUS command parameter:

- RF not ready (0xFC)

5) Extended the functionality for PIO[2] (F2M03 only):

Bluetooth connection established:

- o Held HIGH will disconnect the current Bluetooth connection (as before).

No Bluetooth connection:

- o Held HIGH will prevent the establishment of any Bluetooth connection. RF activity will enter idle mode.
- o Held LOW will enable the RF activity.

6) Emulate serial handshaking lines:

- PIO[4], PIO[5], PIO[6] and PIO[7] are used.

7) *"Pairing and security"* commands:

- Renamed the parameter ALLOW_REPAIRING to MODE. Functionality is the same.

8) *"Pair as slave"* command:

- Renamed the parameter LOCAL_PIN_CODE to PASSKEY. Functionality is the same.

9) *"Pair as master"* command:

- Renamed the parameter REMOTE_PIN_CODE to PASSKEY. Functionality is the same.

Contact information

For support questions please contact your local dealer
For other purposes use: info@free2move.net
Website: www.free2move.net

Local dealer/distributor

The information given herein includes text, drawings, illustrations and schematics that are believed to be reliable. However, Free2move makes no warranties as to its accuracy or completeness and disclaims any liability in connection with its use. Free2move will in no case be liable for any incidental, indirect or consequential damages arising out of sale, resale, use or misuse of the product. Users of Free2move products should make their own evaluation to determine the suitability of each such product for the specific application.