



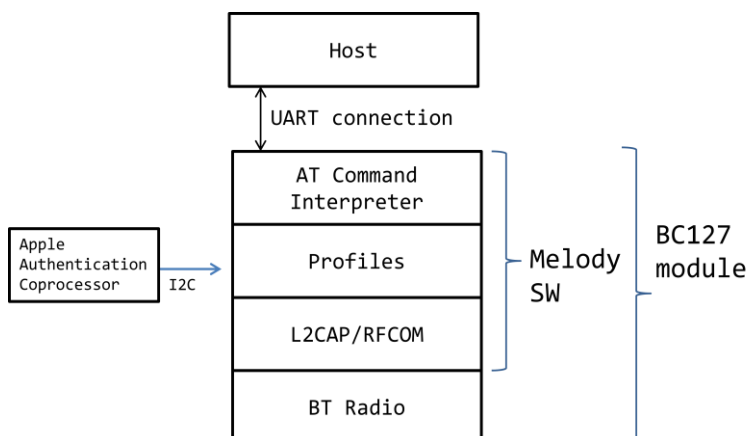
Key Specifications

- Best-in-Class Configurable Bluetooth 4.0 + BLE and Wireless Audio
- Embedded Software and Stack that provides an abstraction layer of the Bluetooth Link
- Controlled via UART, USB or GPIO interfaces
- Profiles: HFP and AG 1.6, PBAP, MAP, A2DP1.3 Sink and Source, AVRCP1.5, DIS, PXP, BC Smart, BC Proximity, iBeacon, HID, Melody command over BLE
- Multipoint Support
- True Wireless Stereo (TWS) support
- Supports data transfer over BLE, SPP, iAP1, and iAP2 protocols
- Supports aptX, AAC, SBC, and MP3 codecs
- Supports digital audio input and output I2S, PCM, SPDIF at sample rates 39kHz - 50kHz
- Supports analogue audio input and output from module ADC and DAC
- Future proof - uses the latest hardware and supports firmware upgrade (DFU)



Description

Melody is an embedded firmware running on the BC127 module. It offers both Classic Bluetooth and Bluetooth Low Energy functionality. It provides all functions needed for designing a wireless Audio device that can connect seamlessly to Smartphone applications and deliver the best audio quality and user experience



Applications

- Wireless Speakers, Docks and Headsets
- SmartPhone Controlled Audio Systems
- Automotive Infotainment Systems
- Medical Devices
- High Quality Audio Streaming
- Gaming Accessories and MP3 Players



Table of Contents

Key Specifications	1
Description	1
Applications.....	1
Introduction	4
Setting Up - Equipment.....	4
Mandatory iAP Configuration Before Use.....	6
Melody Default Behaviour and Operating Modes.....	7
Bluetooth Commands	8
PIO Control	17
PIO Functionality with PIO Control Disabled	18
Audio Enable PIO	18
LED Indications.....	19
Configuration Commands	20
Event Notifications	28
Data Mode	31
cVc and HFP 1.6 Wide Band Speech (WBS)	31
Pairing.....	31
Power management.....	32
Melody Source	32
Melody Multipoint	33
Melody TWS	33
DFU (Device Firmware Upgrade)	33
General Notes.....	34
New features	35
Appendix A: Tone parameters and flags	39
Appendix B: Use case example:.....	42
1. Discovering and pairing with devices	42
2. Tones	43
3. Music (A2DP & AVRCP)	44
4. HFP & PBAP	45
5. Data transfer (SPP).....	46

Manual v5.7 RC8 MFI Rev A

6. Data transfer (IAP)	47
7. Data transfer (BLE)	48
8. HFP-AG	49
9. Music Source (A2DP & AVRCP)	50
10. BLE Central Role	51
11. HID host	52
12. HID device	53
13. Music Manager Example configurations	55
14. Routing example	56



Introduction

Melody is an embedded firmware solution running on the BC127 module. It includes an application, the Bluetooth profiles and the Bluetooth protocol stack both for Classic Bluetooth and Bluetooth Low Energy. It therefore allows using the BC127 module to develop a Bluetooth device without detailed knowledge of the Bluetooth standard. Melody provides a simple high level UART-based command interface. So the Bluetooth module can be controlled from a host processor.

Setting Up - Equipment

To start you need to have:

- A Bluetooth Development board such as BC127-DEVKIT. Please contact sales@bluecreation.com for more information.
- A computer with a serial port running a serial terminal, such as PuTTY or HyperTerminal for Windows or an equivalent program, to communicate over the COM interface.

Connect your PC to your Bluetooth module using the serial port. By default, Melody uses the following UART settings:

- Baud rate : 9600bps
- Data bits : 8
- Stop bits : 1
- Parity bit : No parity
- HW Flow Control : Disabled

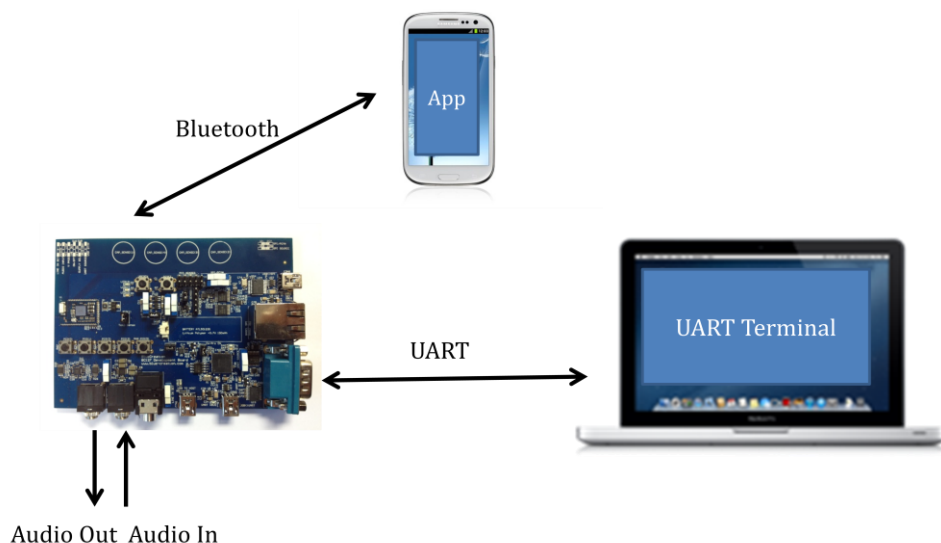


Figure 1: Example configuration



Manual v5.7 RC8 MFI Rev A

Once you have configured your serial terminal and opened the COM port, power up the development kit. You should see a prompt appear on the screen of the terminal. If you see a prompt and a READY, the module is ready to operate.

You are now ready to control the module!

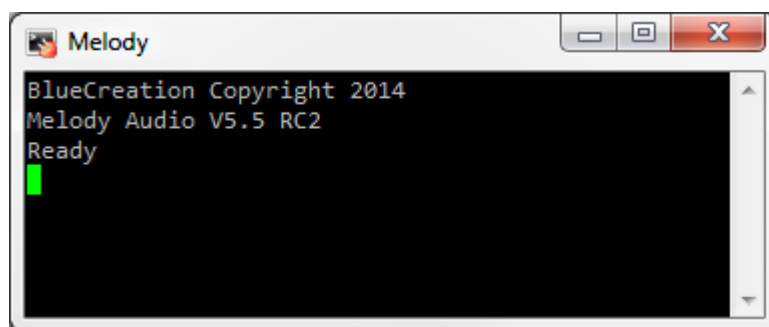


Figure 2: Melody initial load

If you do not see the prompt and READY appear, please check:

- 1) That the module is powered ON and receiving power
- 2) The UART settings on your serial terminal are the same as above
- 3) The module is preloaded with Melody firmware



Mandatory iAP Configuration Before Use

The module will not operate until some IAP configuration parameters are set. The device ID includes the PID (Product ID, 16bits) and VID (Vendor ID, 16bits). These numbers are assigned for a Vendor and Product.

Please note that if the user does not set the Device_ID and IAP configuration parameters, the module will not operate and the Bluetooth radio will not turn ON.

Configuration is done over UART *once* on initial start-up. The following commands are used:

```
SET DEVICE_ID=XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
IAP ACCESSORY_NAME=Chars(48)
IAP MANUFACTURER_NAME=Chars(48)
IAP MODEL_NAME=Chars(48)
IAP SERIAL_NO=Chars(16)
IAP SEED_ID=Chars(10)
IAP ACCESSORY_CAPS=X X X X X X X X
IAP PROTOCOL_STRING=Chars(48)
IAP HARDWARE_VER=D D D
IAP FIRMWARE_VER=D D D

WRITE

RESET
```

X is a hexadecimal value representing an 4 bit integer (*X* is between 0 and F)

Chars(x) is a string of characters with a maximum length of *x* characters

D is a decimal number between 0 and 9

The table below shows the Device ID different attributes.

Attribute	Type	Format
Source	2 bytes	XXXX
VendorID	2 bytes	XXXX
ProductID	2 bytes	XXXX
BCD Version	2 bytes	XXXX
Software Version	8 bytes	XXXX XXXX XXXX XXXX

Table 1: Device ID attributes

For more information on these values, please refer to the MFi Accessory Firmware Specification R46 and MFi Accessory Interface Specification R4. After that sequence of commands, the module will then store the PID and VID in its memory and is then ready to operate. This only needs to be done the first time.



Melody Default Behaviour and Operating Modes

When Melody boots with factory default settings, it will be discoverable. Any Bluetooth enabled phone or other Bluetooth device will be able to see it, if it scans for (discovers) Bluetooth devices. Melody will appear as 'BlueCreation-XXXXXX'¹

When Melody boots with factory default settings, it will be in Command Mode. In this mode, Melody parses the commands coming from the UART and executes them. This allows control over the Bluetooth. Once connected, Melody will transparently transfer any audio stream to the DAC (by default), or I2S or PCM (when configured). If a SPP connection is established, the user can enter Data Mode. This is a mode in which Melody simply passes over data received to the UART acting as a simple data pipe. The Data Mode is explained later in the document in more details.

To switch between Melody operating modes: Command Mode and Data Mode, the following commands are used:

Command	Switch from:	Switch to:
\$\$\$\$	Data Mode	Command Mode
ENTER_DATA	Command Mode	Data Mode

Table 2: Operating mode commands

¹ XXXXXX will be six hexadecimal digits that correspond to the last six hexadecimal digits of the device Bluetooth address. If you are using the BC127-DEVKIT, the LED will be flashing alternatively to indicate It is Discoverable



Bluetooth Commands

When in Command mode, the module accepts commands from the host via the UART interface. The generic syntax for commands is:

COMMAND (parameter_1) (parameter_2) ... (parameter_n)\r

with a space between each parameter and a carriage return ('\r' or ASCII 0x0D) at the end of each command.

Commands need to be completely entered within the specified command timeout (400ms by default). Melody will return an 'OK' after each command to indicate that the command was executed. An 'ERROR' will be returned if the command has not been executed or if the parameters are wrong.

The different commands to control the Bluetooth link are listed in alphabetical order below. Mandatory parameters are listed in "()" optional parameters are listed in "[]".

Please note that all optional **[link_ID]** parameters are mandatory for multipoint and TWS operation. For all commands that list **[link_ID]** as parameter, please supply the link id of the link you would like that command to be applied to.

UART Command	Description
\$\$\$\$	Exit Data mode
ADVERTISING (mode)	ON starts BLE advertising, OFF stops BLE advertising, for BLE_ROLE=1 ON starts iBeacon, OFF stops iBeacon, for BLE_ROLE=3
ANSWER [link_ID]	Accepts an incoming call on the HFP link.
BATTERY_STATUS	Returns the battery status of the battery attached to the module by detecting the voltage level
CALL [link_ID] (number)	Dials the number on the HFP link In AG no parameter is passed, and command initiates a bidirectional HFP audio connection between devices.
CLOSE (link_ID)	Terminates the Bluetooth Profile connection. Linked is a number that defines the connection ID. all closes all links
CONFIG	Shows all configuration registers
CVC_CFG (type) [(key) (length)]	Read or write the CVC configuration stored on the module. type can be WB or NB . If only type parameter is supplied all 4 config keys for this type will be displayed. If key and length parameters are present, the key indicated will be written with a value of size length. key can be 0 - 3 length is in 16 bit words and can be 0 - 64. 0 will delete the key. The CVC_CFG command will reply with OK/ERROR. If length was supplied after OK a raw list of upto 24 space separated words followed by carriage return has to be supplied.



Manual v5.7 RC8 MFI Rev A

	Each carriage return will be replied to with an OK/ERROR if the operation is successfully. If more than the length words are supplied the overall operation will fail.																		
DFU	Puts the device in DFU mode ²																		
DISCOVERABLE (mode)	OFF makes the device non discoverable ON puts the module in discoverable mode ³																		
DESC (size)	Sets the USB descriptor to use when acting as a hid device. Size is the length of binary data following (similar to Send when using raw data). By default an USB descriptor for a simple keyboard is loaded. When using this command any previously loaded USB descriptor is deleted. Any USB descriptor loaded with this command is not persistent across a reset.																		
DEV_INFO (param)=value	This command is used to set the data to be displayed by the BLE Device Information service. The param types and default values are listed below: ACCESSORY_NAME=XXXXXXXXXX MANUFACTURER_NAME=XXXXXXXXXX MODEL_NAME=XXXXXXXXXX SERIAL_NO=XXXXXXXXXX HARDWARE_VER=0 0 0 FIRMWARE_VER=0 0 0																		
DUT	Enters DUT mode to use testing tools with the serial port and default UART configuration for production																		
END [link_ID]	Ends a call on the HFP/AG link																		
ENTER_DATA	Enters Data mode																		
HELP	Returns available list of commands																		
GET (config_name)	Reads the value of a configuration parameter.																		
IAP <Parameter>=value	Sets IAP parameters. These are the parameters that identify the accessory for iOS devices and applications. Maximum sizes of configurations in bytes/chars: <table> <tr><td>ACCESSORY_NAME</td><td>48 (chars)</td></tr> <tr><td>MANUFACTURER_NAME</td><td>48 (chars)</td></tr> <tr><td>MODEL_NAME</td><td>48 (chars)</td></tr> <tr><td>SERIAL_NO</td><td>16 (chars)</td></tr> <tr><td>SEED_ID</td><td>10 (chars)</td></tr> <tr><td>PROTOCOL_STRING</td><td>48 (chars)</td></tr> <tr><td>ACCESSORY_CAPS</td><td>8 Hex values space separated</td></tr> <tr><td>HARDWARE_VER</td><td>3 digits space separated</td></tr> <tr><td>FIRMWARE_VER</td><td>3 digits space separated</td></tr> </table> <p>Please refer to MFi Accessory Firmware Specification R46 and MFi Accessory Interface Specification R4.</p>	ACCESSORY_NAME	48 (chars)	MANUFACTURER_NAME	48 (chars)	MODEL_NAME	48 (chars)	SERIAL_NO	16 (chars)	SEED_ID	10 (chars)	PROTOCOL_STRING	48 (chars)	ACCESSORY_CAPS	8 Hex values space separated	HARDWARE_VER	3 digits space separated	FIRMWARE_VER	3 digits space separated
ACCESSORY_NAME	48 (chars)																		
MANUFACTURER_NAME	48 (chars)																		
MODEL_NAME	48 (chars)																		
SERIAL_NO	16 (chars)																		
SEED_ID	10 (chars)																		
PROTOCOL_STRING	48 (chars)																		
ACCESSORY_CAPS	8 Hex values space separated																		
HARDWARE_VER	3 digits space separated																		
FIRMWARE_VER	3 digits space separated																		
IAP_APP_REQ <protocol>	This command will send the iOS device to open an application that has																		

²DFU allows downloading a new firmware upgrades onto the Bluetooth module via the UART interface and allows users to upgrade melody to new releases.

³DISCOVERABLE ON will disconnect all active connections.



Manual v5.7 RC8 MFI Rev A

	the protocol specified
INQUIRY (timeout) [TWS]	Searches Bluetooth devices in the area for maximum period of time (equal to $1.28s * \text{timeout}$). Timeout takes a value between 1 and 48. Supplying the optional TWS parameter will only show TWS devices ⁴ .
LICENSE <flag>	This command is used to retrieve the license keys for APTX and cVc if <flag> is omitted. If <flag> is set to APTX or CVC, then it will retrieve the corresponding license key.
LICENSE <flag>=<value>	This command is used to set the license key to either APTX or cVc. <flag> can be either CVC or APTX. <value> is 5 words consisting of the license key (e.g. 1111 2222 3333 4444 5555)
LIST	Lists paired devices in the format LIST (BT_addr) (Space separated list of supported profiles)
MM_CFG (key) [(length)]	Read or write the Music Manager configuration stored on the module. MM_CFG key will read the PSKey at that location. MM_CFG key length will be write the PSKey. key is DSP PSKey number. It can be between 24 and 38. length is the number of 16bit words to be written for that PSKey. length can be 0 - 64. 0 will delete the key. The MM_CFG command will reply with OK/ERROR. After a valid MM_CFG key length command, BC127 will reply OK and expect a raw list of 16bit values followed by carriage (\r). If the list is greater than 24 x 16 bits values, an additional \r has to be supplied after 24 x 16bits values are sent. Each carriage return will be replied to with an OK/ERROR if the operation is successfully. If more than the length words are supplied the overall operation will fail.
MUSIC [link_ID] (instruction)	Controls the music stream state and sends AVRCP instructions (if AVRCP available)(Instruction=PLAY, PAUSE, STOP, FORWARD ⁵ , BACKWARD ⁴) Please use the A2Dp link ids for multipoint.
NAME (BT_addr)	Returns the friendly name of device with the provided Bluetooth address
OPEN (BT_addr) (profile)	Establishes a connection with a given Bluetooth address (formatted as 12 Hexadecimal digits with no separators. e.g.: 3859F9CCB893) and a certain profile. profile can be: SPP, A2DP, AVRCP, MAP ⁶ , HFP, PBAP ⁷ , HID. If profile is left blank, SPP will be assumed.
PASSKEY (type) (key)	This function is used to respond to a SSP user passkey confirm (type 1) or request (type 2), as indicated by a PAIR_PASSKEY notification. For confirmation, use key=1 to accept and key=0 to reject. For passkey request, please supply 6 digit passkey as requested.
PIO (PIOx) (state)	Sets the PIO state to the required level. PIOx is in [1,12]; level is 0 for low, and 1 for high
POWER (mode)	OFF disconnects all active connections and puts the device in limbo mode, where it is not connectable, or discoverable. As a consequence,

⁴ To find TWS devices, the DEVICE_ID configuration needs to be the non-zero and the same on both sides.

⁵ Only available when acting as Sink to A2DP connection i.e. CLASSIC_ROLE=0

⁶ In Melody, MAP requires an active HFP connection. MAP is optional and not part of the default release

⁷ AVRCP requires an active A2DP connection; PBAB requires an active HFP connection.



Manual v5.7 RC8 MFI Rev A

	the UART commands that cannot be executed are rejected. ON returns the device to a connectable state.
PULL_PBOOK (pbook)	<p>If no parameter is specified downloads the phone book of the connected phone.</p> <p>(pbook) can be:</p> <p>phonebook – download phone book (same as no parameter)</p> <p>outgoing – download outgoing calls list</p> <p>incoming – download incoming calls list</p> <p>missed – download missed calls list</p> <p>combined – download the combined calls list</p> <p>The response format is:</p> <p>PBAP_PB NAME: Name</p> <p>PBAP_PB TEL: number</p> <p>PBAP_PB OK when download completes</p>
PULL_ABORT	<p>Aborts an active phonebook download.</p> <p>By default, the PBAP profile will not be closed if an active download is in process.</p> <p>Aborting the operation is required before closing the pbap profile.</p>
READ_DESC (BT_addr)	Reads USB descriptor from SDP of the remote HID device with address BT_addr.
REJECT	Rejects an incoming call on the HFP link
RESET	Resets the device
RESTORE	Unpairs all paired devices and Restores configuration to factory settings; Writing settings needs to be followed by WRITE and RESET for the changes to take effect.
ROLE ([link_ID] or [profile] <role>)	<p>Change the classic connection role of the local device. Notification will be sent if the new role is different from the current role. Multipoint only.</p> <p>link_ID is the link id to change the role on. Note that the id is required per device and not per profile.</p> <p><role> can be:</p> <p>MASTER – Change the role of the local device to be the master of the link</p> <p>SLAVE – Change the role of the local device to be the slave of the link.</p>
ROUTE [audio_source]	<p>Select which audio routing to apply. Only available audio sources will be accepted.</p> <p>If no parameter is supplied, the current routing will be displayed. Please refer to audio_source below for enumeration of meanings.</p> <p>audio_source can be:</p> <p>0 – unroute current routing; This will be overridden by next automatic route event (i.e. HFP call or A2DP media stream)</p> <p>2 – A2DP link 2 audio; This will be overridden by next automatic route event (i.e. HFP call or A2DP media stream)</p> <p>3 – A2DP link 3 audio; This will be overridden by next automatic route event (i.e. HFP call or A2DP media stream)</p> <p>8 – HFP link 8 audio; This will be overridden by next automatic route</p>



Manual v5.7 RC8 MFI Rev A

	<p>event (i.e. HFP call or A2DP media stream)</p> <p>9 – HFP link 9 audio; This will be overridden by next automatic route event (i.e. HFP call or A2DP media stream)</p> <p>40 – Analogue loopback; This routing will not be overridden by next automatic route event.</p> <p>41 – Analogue<->Digital bidirectional pass-through; This routing will not be overridden by next automatic route event.</p> <p>Notes:</p> <ol style="list-style-type: none"> 1) If the codec rate set is not 44.1kHz or 48kHz, when analogue loopback is used, the sample rate used will be 44.1kHz 2) Analogue<->Digital pass-through can only operate at 44.1kHz and 48kHz. Please ensure that your digital interface is configured to one of these before using this routing; 3) Analogue<->Digital pass-through will not be available if Melody is not configured to use a digital interface 4) In order for a routing to be accepted there has to be an audio stream associated with that audio_source: for A2DP that means an active media stream, for HFP an ongoing call with audio; Analogue loopback and Analogue<->Digital pass-through will always have available audio streams 5) Attempting to route an already routed stream will return ERROR
RSSI	Returns the receiver signal Strength of the link. -70dBm is a good link, -80dBm is a poor link
SCAN (timeout)	Searches BLE-enabled Melody devices in the area for maximum period of time (given in seconds). Use the value of 0 for an infinite timeout. Scans are stopped when a connection is attempted or when they time out.
SEND ([link_ID] or [profile]) ([characters] or [size])	<p>Sends characters or data through a selected profile (HID, IAP, SPP or BLE). If no profile is specified, it will send to the first connected profile in the order (BLE, SPP).</p> <p>If Multipoint or Raw_Data mode the first supplied parameter must be a link id; For single point this must be a profile.</p> <p>If in Raw data mode the second parameter must be the length of the data to be sent. The command will return status based on the availability of space to send the data and profile connection status. Raw data of 'length' bytes must be sent over UART immediately afterwards. Melody will return into normal command mode after that length has been received. A notification will be sent by melody indicating the status of the data send. For BLE length is limited to 128bytes.</p> <p>When Raw Data is not enabled the second parameter is a string that must not exceed 128 characters. The string may contain all characters except: 0x00, 0x08, 0x0D as those are special characters used in command mode.</p>



Manual v5.7 RC8 MFI Rev A

	When sending HID data, Raw data mode must be enabled.
SET (config)=value	Sets a new value to a configuration parameter.
STATUS	<p>Lists the device status for Classic and BLE: STATUS (Classic status) (BLE status) Where (Classic status) can be: OFF, CONNECTABLE DISCOVERABLE, CONNECTABLE, CONNECTED and (BLE status) can be: OFF, IDLE, ADVERTISING, CONNECTED.</p> <p>And Link status for established links in the format: LINK (link id) CONNECTED (BT_Addr) (Audio_Codec/AVRCP_State) (if applicable)</p>
TOGGLE_VR [link_ID]	Start/Stop Voice call command on the phone
TONE⁸ (flag) (value) (flag) (value) ...	<p>Plays a tone based on the input. A tone must have at least 1 note. A note must have a length parameter.</p> <p>Please refer to Appendix B for full information regarding, flags, accepted values and descriptions.</p> <p>Flags :</p> <p>Tempo TE [0 - 4095] Timbre TI [0 - 7] Volume V [0 - 255] Decay D [00 - FF] (enter value in hex9) Note N [A - F + octave 0 - 9] eg:AF7, A7, AS7 Length L [1,2,4...64] or [3,6,12...96] triplets</p>
TRANSFER_CALL [link_ID]	<p>This command will transfer the audio between Melody and the phone in a call. If the audio is in Melody, then after this command, the audio will be on the phone.</p> <p>Note that some phones disconnect the Bluetooth link when the audio is transferred back to the phone</p>
UNPAIR	Clears the list of previously paired devices
VERSION	Returns information on the firmware version number
VOLUME¹⁰ [(flag) [=value]]	<p>Display volume information for single point scenario.</p> <p>Supplying the command without a parameter will display all volume</p>

⁸ Currently not supported over digital audio output.

⁹ Parsed as fixed point of the following format in binary bbbb.bbbb or hexadecimal X.X

¹⁰ Single Point version of VOLUME command



Manual v5.7 RC8 MFI Rev A

	<p>data. Please be advised that this data is not accurate if the respective connection is not active and routed at the moment.</p> <p>If arguments are supplied, the following values are available:</p> <p>flag can be: (available in all scenarios)</p> <p>UP – increases the volume for current active routing</p> <p>DOWN – decreases the volume for current active routing</p> <p>HFP – returns the speaker volume of the connected HFP link</p> <p>MIC_MUTE – returns the MIC state of the mic</p> <p>A2DP returns the speaker volume of the connected A2DP link</p> <p>WIRED_A – display volume for Analogue loopback routing</p> <p>WIRED_D – display volume for Analogue<->Digital pass-through routing</p> <p>WIRED_MUTE – display mute status for current active WIRED routing</p> <p>An additional argument [=value] can be supplied to set the respective (flag)'s value as follows:</p> <p>HFP – value must be in [0 ; 15] – desired volume for link</p> <p>MIC_MUTE^{11 12} – value can be ON/OFF and enables/disables microphone mute</p> <p>A2DP – value must be in [0 ; 15] – desired volume for link</p> <p>WIRED_A – value must be in [0 ; 15] – desired volume for link</p> <p>WIRED_D – value must be in [0 ; 15] – desired volume for link</p> <p>WIRED_MUTE¹² – value can be ON/OFF and enables/disables ADC input mute</p> <p>Additional Notes on Volume command behaviour:</p> <ul style="list-style-type: none"> • Melody stores specifics and different volume levels for each profile. • When in Analogue Loopback (WIRED_A) or Analogue<->Digital (WIRED_D) mode, the VOLUME UP/DOWN command will only affect the DAC gains. • WIRED_MUTE will only affect the ADC gain; In addition changing volume for these routing will not save the respective volume levels. On subsequent re-routing the values specified in the WIRED config will be used. • In CLASSIC_ROLE=0 X (sink) volume changes the local volume or if the remote side supports it, indicates a volume change and the sides synchronise volumes using AVRCP Absolute volume.
--	--

¹¹ Please note that this will not be applied if cVc plugin is used, as it has full control of the audio input signal path.

¹² MUTE=ON sets ADC gain to 0; depending on input signal level, this may still result in some audio passing through. MIC_MUTE=OFF reverts to previous Input gain level



Manual v5.7 RC8 MFI Rev A

	<p>This does not affect the stream volume.</p> <ul style="list-style-type: none"> In CLASSIC_ROLE=1 X (source) volume changes are sent over the air as Absolute volume commands. No local/stream volume change is applied. In order to change the stream volume or control input levels, please use the INPUT_GAIN configuration.
VOLUME¹³ (link_ID) [(value) or (flag) [= (value)]]	<p>Display volume information for multi-point scenario.</p> <p>The link ID supplied must be one for either of the A2DP links (2,3) or HFP links (8,9), or the WIRED_A (40) or WIRED_D (41) routing link_id placeholders.</p> <p>Supplying only link_ID will display the volume information only for active links (connected and having audio streams associated) or in the case of WIRED links, when routed. Supplying a link_ID to an inactive link will return ERROR.</p> <p>Optional arguments can be supplied to set the values for active links. To set the volume for a specific link: Provide a link_ID and (value) which can be in [0 ; 15] Provide a link_ID and (flag) which can be UP/DOWN</p> <p>To set the MUTE parameters for a HFP and WIRED link the respective link_ID and the following (flag) can be used: MIC_MUTE¹⁴ ¹⁵ – value can be ON/OFF and enables/disables ADC input mute WIRED_MUTE¹⁵ – value can be ON/OFF and enables/disables ADC input mute</p> <p>Additional Notes on Volume command behaviour:</p> <ul style="list-style-type: none"> Melody stores specifics and different volume levels for each profile. When in Analogue Loopback (WIRED_A) or Analogue<->Digital (WIRED_D) mode, the VOLUME UP/DOWN command will only affect the DAC gains. WIRED_MUTE will only affect the ADC gain; In addition changing volume for these routing will not save the respective volume levels. On subsequent re-routing the values specified in the WIRED config will be used. In CLASSIC_ROLE=0 X (sink) volume changes the local volume

¹³ Multi Point version of **VOLUME** command

¹⁴ Please note that this will not be applied if cVc plugin is used, as it has full control of the audio input signal path.

¹⁵ **MUTE=ON** sets ADC gain to 0; depending on input signal level, this may still result in some audio passing through. **MIC_MUTE=OFF** reverts to previous Input gain level



Manual v5.7 RC8 MFI Rev A

	<p>or if the remote side supports it, indicates a volume change and the sides synchronise volumes using AVRCP Absolute volume. This does not affect the stream volume.</p> <ul style="list-style-type: none">• In CLASSIC_ROLE=1 X (source) volume changes are sent over the air as Absolute volume commands. No local/stream volume change is applied. In order to change the stream volume or control input levels, please use the INPUT_GAIN configuration.
WRITE	Store configurations.

Table 3: Command Mode Bluetooth Commands



PIO Control

By default, the module can also be controlled by PIOs. This allows the module to work autonomously without the need of a host processor as in a wireless speaker for example. The table below lists the UART commands that also have a PIO equivalent. This PIO functionality can be turned off (See configuration parameters).

Please note that when in Multipoint, PIO controls will take effect on the primary (lower number) link id;

GPIO ¹⁶	Melody Classic Role	UART Equivalent	Description
VREGEN single	0	DISCOVERABLE ON	If Sink, Melody will become discoverable. This disconnects all connections.
VREGEN long	All	SET CLASSIC_ROLE=1 SET AUTOCONN=1 RESET	Enable Source mode and auto connection and resets Melody. On reset and failing auto-connection, Melody will start inquiry and pairing to first device with RSSI below the threshold set in the RSSI_THRESH configuration. ¹⁷
PIO_0single	All	VOLUME UP	Increases volume for HFP and A2DP
PIO_1single	All	VOLUME DOWN	Decreases volume for HFP and A2DP
PIO_2 single	All	MUSIC PLAY/MUSIC PAUSE/ANSWER/END	Starts playing music or stop playing music for A2DP Source and Sink. If there is an incoming HFP call it will answer the call. During an ongoing call it will end the call.
PIO_2 long	0	CALL	If Melody is Sink, it will initiate a CALL (last number redial).
	1	CALL	If Source, it will initiate a call (CALL).
PIO_4single	All	MUSIC BACKWARD	Sends AVRCP instruction BACKWARD
PIO_5 single	All	MUSIC FORWARD	Sends AVRCP instruction FORWARD (Next track)

Table 4: Command Mode General PIO Bluetooth Commands

Note: 'single' is defined as any press and release that happens within 999ms. 'long' is defined as holding the PIO high for 1000ms. A repeated 'long' press will be detected if the PIO is held high for multiples of 1000ms.

¹⁶ Refer to BC127 Datasheet for location of PIO on the module pinout.

¹⁷ Please note that this may fail if there is another discoverable device below this threshold. Please press VREG again and repeat procedure.



PIO Functionality with PIO Control Disabled

If PIO control is disabled (SET GPIOCONTROL=OFF), PIO(0) and PIO(5) are used as the table below.

Please note that when in Multipoint, PIO controls will take effect on the primary (lower number) link id;

PIO	Type	Description
PIO_0	Output	High if Melody is connected Low if Melody is not connected
PIO_5	Input	High to enter data mode Low to enter command mode

Table 5: PIO control and indication

Audio Enable PIO

On non-MFI builds, PIO 3 will go high for an active SCO or A2DP stream. This can be used for example as an audio enable to an amplifier.

This feature is not configurable.



LED Indications

By default, the module has LEDs control enabled. This allows the module toggle the LEDs:

Please note that Classic and Smart indication occur on different LEDs and will be displayed concurrently based on state.

Classic State	LED pattern (repeating)
Discoverable	Fast Flash Toggle LED 0 and LED1
Connected Sink	LED 1 flashes once approx 2.5s
Connected Source	LED 0 flashes once approx 2.5s
Connectable Sink	LED 1 blinks slowly
Connectable Source	LED 0 blinks slowly
OFF or DFU	All LEDs are OFF

Table 6: LED Patterns Classic

Smart State	LED pattern (repeating)
Idle/Off	LED 2 off
Advertising	LED 2 blinks slowly
Connected	LED 2 flashes once approx 2.5s

Table 7: LED Patterns Smart

Other states	LED pattern (once)
Powering On	All LEDs On for 1s
Powering Off	All LEDs quick blink 3 times
Off or DFU	All LEDs off

Table 8: Other indications



Configuration Commands

When Melody is in Command mode, the user can configure general parameters for the module. These parameters are stored in the RAM memory. If required, the parameters can be stored to Flash memory. When the module reboots, it will boot with the parameters that are saved to Flash memory.

There are 4 main commands to configure parameters. The commands to modify configuration parameters are described below. Mandatory parameters are listed in "()" optional parameters are listed in "[]".

Command	Description
GET (config_name)	Reads the value of a configuration parameter.
RESTORE	Restore configuration parameters to default factory settings.
SET (config_name)=param_1 param_2 ...param_n	Sets a new value to a configuration parameter.
WRITE	Write configuration changes to Flash memory. This command writes all the current configuration parameters to non-volatile memory. These are then read at boot-up time.

Table 9: General Configuration Commands

Note: If the module boots up with PIO_0 high, the factory default configurations are reset. This allows reverting to a known and working configuration state if severe problems are encountered.

The different configuration parameters are described in alphabetical order in the table below. Once modified, many configuration parameters will not take effect before a reboot. Therefore they need to be stored to Flash before rebooting.

Configuration Parameter	Description	Default	Requires Reboot
AUDIO=(value)	AUDIO allows to route the A2DP audio stream: 0 – Analogue 1 – PCM Master ¹⁸ 2 – PCM Slave 3 – I2S Master ¹⁸ 4 – I2S Slave 5 – SPDIF ¹⁸ Note: When using options 1 to 4, HFP calls are mono (only left channel).	0	Yes

¹⁸ Please note that when Melody is acting as master of digital audio connections, clocks will only be supplied when there is an active routing at the moment.



Manual v5.7 RC8 MFI Rev A

AUTOCONN=(value)	<p>0 – Disabled</p> <p>1 – Auto-connects on Reset (the device will try to reconnect to previously paired devices with all possible profiles)</p> <p>2 – Auto-connect to specific address on Reset. (Melody will try and connect with the Bluetooth address specified by the REMOTE_ADDR parameter)</p> <p><i>Note: Melody will store up to 8 paired devices in its persistent memory. So AUTOCONN=1 means it will try to connect to each one and attempt a number of times (MAX_REC) before giving up. When the pairing memory is full, Melody will erase the oldest paired device. So the 8 stored paired devices are the 8 most recently paired devices</i></p>	0	Yes
BATT_THRESH=(chrgLvl) (crit) (low) (lvl0) (lvl1) (lvl2) (lvl3)	<p>Set the battery voltage level thresholds. These are used internally by the battery charger to determine behaviour, and also reflect the responses to the BATTERY_STATUS command.</p> <p>chrgLvl – charge voltage limit threshold – charging will stop once battery reaches this voltage (in mV)</p> <p>crit – critical battery level – device will shut off until charged above this level (in units of 20mV)</p> <p>low – low battery level (in units of 20mV)</p> <p>lvl0 – Level 0 (in units of 20mV)</p> <p>lvl1 – Level 1 (in units of 20mV)</p> <p>lvl2 – Level 2 (in units of 20mV)</p> <p>lvl3 – Level 3 (in units of 20mV)</p>	4250 145 150 155 165 175	Yes
BAUD=(value)	UART Baud rate (9600,19200,38400,57600,115200, 230400, 460800, 921200)	9600	No
BLE_ROLE=(value)	<p>Chooses the BLE role of the device:</p> <p>0 – BLE disabled</p> <p>1 – Peripheral</p> <p>2 – Central</p> <p>3 – iBeacon</p> <p>Before setting BLE_ROLE to 1,2 or 3, HID_ROLE must be set to 0.</p>	1	Yes
BPS=(value)	<p>This is only for I2S and PCM Codecs. It defines the number of bits per sample (16 supported for PCM, 16 and 24 for I2C. 24 bits per sample will be LSB zero padded 16bit values).</p> <p><i>Note: To configure I2C input resolution, use the I2C config.</i></p>	16	Yes
CLASSIC_ROLE=(role) [mode]	Sets the device role for the classic Bluetooth profiles HFP and A2DP; profiles supported in source are: HFP,	0	Yes



Manual v5.7 RC8 MFI Rev A

	A2DP, AVRCP, and SPP Role can be 0 – Sink 1 – Source Mode can be: 0 – single point 1 – multipoint 2 – TWS		
CMD_TO= (value)	Guard time for detecting a \$\$\$\$ command to switch from data mode to command mode. There should be 20ms x CMD_TO between the last character and the \$\$\$\$. And there should be 20ms x CMD_TO after the \$\$\$\$ for the command to be recognised.	20	No
COD= (value)	Configures Class of Device	240404	Yes
CODEC= (Codec) (Fs) (Mode)) [hfpBias] [a2dpBias]	CODEC is a decimal representation of 4 bit value specifying which codecs are to be enabled: Eg: 0 – No extra codecs (SBC only) (bitmap 000b) 1 – aptX enabled (bitmap0001b) ¹⁹ 2 – MP3 enabled (bitmap0010b) 4 – AAC enabled5 (bitmap0100b) 8 – aptX Low Latency (bitmap 1000b) 15 – All extra codecs (bitmap1111b) Fs - (Sample Rate in Hz) 39000 to 50000, Note: For digital audio this parameter needs to be set to something in range. For analogue output 0 indicates that the output sample rate will be the same as the sample rate over the air. Mode: 0 is Mono, 1 is Stereo hfpBias: enable/disable mic bias when in HFP mode a2dpBias: enable/disable mic bias when in A2DP mode	0 0 1 1 0	Yes
CONN_TO= (value)	Time Melody stays connectable (in seconds) after a connection has been established in Multipoint. Values accepted: 0 – 65534. 0 – never stop being connectable;	0	No
DEEP_SLEEP= (value)	Enable and Disable Deep Sleep	OFF	No
DEVICE_ID= (word_1) ... (word_8)	The Device ID sets the configuration parameters for the Device ID Profile. This profile sets the following as an SDP record so that remote devices can get extra	0000 0000 0000 0000 0000 0000	Yes

¹⁹ Please contact BlueCreation if you require a licence for aptX or aptX-LL



Manual v5.7 RC8 MFI Rev A

	<p>information about the product:</p> <p>Vendor Id source (2 bytes) Vendor Id(2 bytes) Product Id(2 bytes) BCD version (2 bytes) Software version (8 bytes)</p> <p>For more information about these settings, refer to Device ID profile specification on the 23lueetooth.org adopted specification page</p>	0000 0000	
DISCOVERABLE=(value) (timeout)	<p>Mode defines discoverable scheme after boot:</p> <p>0 – non discoverable 1 – discoverable after reconnection failed, if auto-connect is enabled 2 – discoverable immediately overrides auto-connect</p> <p>Timeout defines time to discoverable after boot (timeout=0 means infinite, number is value in milliseconds to be discoverable before going into deep sleep).</p>	1 0	Yes
ENABLE_A2DP=(value)	Enables and disables A2DP profile	ON	Yes
ENABLE_ANDROID_BLE=(value)	Enables and disables Android BLE support.	OFF	Yes
ENABLE_AVRCP=(value)	Enables and disables AVRCP profile (depends on A2DP profile)	ON	Yes
ENABLE_BATT_IND=(value)	Enables and disabled the iOS battery indication ²⁰	ON	No
ENABLE_HFP=(value)	Enables and disables HFP profile and AG in Source mode	ON	Yes
ENABLE_HFP_CVC=(value)	Enables and disables cVc Note: CVC is not supported by AG	OFF	Yes
ENABLE_HFP_NREC=(value)	Sets the HFP NREC flag, which tells the AG not to use its internal Noise Reduction/Echo Canceller algorithm. We suggest you set this to ON if using CVC or an external noise reduction platform.	OFF	Yes
ENABLE_HFP_WBS=(value)	Enables and disables Wide band speech support Note: WBS is not supported by AG	ON	Yes
ENABLE_IAP=(value)	Enables and disables IAP. Before enabling IAP, HID_ROLE and BLE_ROLE must both be set to 0.	ON	Yes
ENABLE_IAP_V2=(value)	Enables and disables IAP v2.	ON	Yes

²⁰ Due to iOS behaviour, this needs to be changed before initial Pairing to iOS device. Changing after that will not remove indicator on iOS.



Manual v5.7 RC8 MFI Rev A

	Before enabling IAP v2, HID_ROLE and BLE_ROLE must both be set to 0.		
ENABLE_MAP=(value)	Enables and disables MAP profile	ON	Yes
ENABLE_PBAP=(value)	Enables and disables PBAP profile (depends on HFP profile)	ON	Yes
ENABLE_RAW_DATA=(val)	Enables and disables raw data mode.	OFF	Yes
ENABLE_SPP=(value)	Enables and disables SPP profile	ON	Yes
ENABLE_SPP_SNIFF=(state) (min_interval) (max_interval) (attempt) (timeout) [duration]	Enables SPP only connections to go into sniff mode which allows Melody to use low power modes. These values take effect if SPP is the only connected profile state: OFF – active ON – passive (if all parameters are 0) – sniff with passive (with all parameters supplied) Parameters: min_interval: the minimum interval of the sniff period max_interval: the maximum interval of the sniff period attempt: Determines for how many slots the slave shall listen when the slave is not treating this as a scatternet link. Timeout: Determines for how many additional slots the slave shall listen when the slave is not treating this as a scatternet link. Duration: the time to stay in this sniff state.	OFF 0 0 0 0	No ²¹
FLOW_CTRL=(value)	Enables and disables UART Flow Control	OFF	Yes
FORCE_ANALOG_MIC=(value)	Forces the use of the analogue input for MIC when a digital interface is selected in AUDIO (I2S, PCM, SPDIF)	OFF	Yes
GPIOCONTROL=(value)	Enables and disables PIO control	ON	Yes
HID_ROLE=(value)	Value is 0 if hid if disabled, 1 if host and 2 if hid device. Before setting HID_ROLE to 1 or 2, BLE_ROLE must be set to 0.	0	Yes
I2S=(value)	Configures the I2S parameters. I2Sconfig is a 10 bit value represented by the 10 LSB of a 4 digit hexadecimal number: Bit [0]: Justification format: 0 – left justified; 1 – right justified. Bit [1]: Left Justify delay: 0 – MSB of SD data occurs in the first SCLK period following WS transition; 1 – is MSB of SD data occurs in the second SCLK period. Bit [2]: Channel polarity: 0 – SD data is left channel when WS is high; 1 – SD data is right channel. Bit [3]: Audio attenuation enable (On SD_IN): 0 – 17-bit SD data is rounded down to 16bits; 1– the audio attenuation defined in bits [7:4] is applied over 24 bits	0002	Yes

²¹ Changing the sniff parameters require the connection to restart.



Manual v5.7 RC8 MFI Rev A

	<p>with saturated rounding. Requires Bit [10] to be 0.</p> <p>Bits [7:4]: Audio attenuation configuration in 6dB steps.</p> <p>Bits [9:8]: Justify resolution of data on SD_IN: 00 – 16-bit; 01 – 20-bit; 10 – 24-bit; 11 – reserved.</p> <p>Bit [10]: 16 bit crop enable: 0 – 17-bit SD_IN data is rounded down to 16bits; 1 – only the most significant 16bits of data are received.</p> <p>Note that for Master mode the Master clock is generated at $F_s * 2 * bps * 2$ Hz. The word clock is generated at F_s Hz, where F_s is taken from the <code>CODEC</code> config and bps is supplied by the <code>BPS</code> config. Note that for slave mode operation at 16bps the Master clock supplied must be at least $F_s * 2 * bps * 4$.</p>		
IBEACON_MAJOR= (value)	Sets the iBeacon Major data field.	8765	Yes
IBEACON_MINOR= (value)	Sets the iBeacon Minor data field.	4321	Yes
IBEACON_POWER= (value)	Sets the iBeacon power field (in dBm)	-35	Yes
IBEACON_UUID= (values)	Sets the 128bit (16 octet) iBeacon UUID in Little Endian format.	00 00 11 01 00 00 10 00 80 00 00 80 5F 9B 34 FB	Yes
INPUT_GAIN= (value)	Sets the analogue input gain. Values supported are 0 - 15	15	No
LOCAL_ADDR= (value²²)	Contains the local Bluetooth address of the device.	0	No
MAX_REC= (value)	Maximum times to try to reconnect to a device in PDL	2	Yes
MM (enable) [(user_eq_en) (bank) (bass_boost_en) (3d_enhance_en) (compander_en) (dither_en) (speaker_eq_en)]	<p>Configure the Music Manager audio enhancements for A2DP music streams. All configuration parameters following MM are 0 or 1 except bank which is between 0 and 6.</p> <p>enable – mandatory, enable or disable MM</p> <p>All subsequent parameters are optional but must be provided in order. i.e. if you want to modify (bass boost) you have to supply (user_eq_en) and (bank).</p> <p>user_eq_en – enable/disable the user equaliser block bank – set the equaliser bank to use when user equaliser is enabled. Available banks are 0 to 6. bass_boost_en – enable/disable the bass boost block</p>	0 0 0 0 0 0 0	No

²² Please note that value is a read only.



Manual v5.7 RC8 MFI Rev A

	3d_enhance_en – enable/disable the 3D enhancement block compander_en – enable/disable the compander (compressor/expander) block dither_en – enable/disable the post processing/dither block speaker_eq_en – enable/disable the speaker equaliser block Please note that these configurations only enable/disable music manager blocks. Those need to be configured use UFE, as specified by CSR.		
MUSIC_META_DATA= (value)	Enable (ON) and disable(OFF) AVRCP 1.3 metadata notifications sent by Melody.	ON	No
NAME= (value)	Friendly Name; Last 6 symbols of the name XXXXXX are the last 6 symbols of the Bluetooth address	BlueCreation-XXXXXX	Yes
NAME_SHORT= (value)	Short Name, used for Melody Smart Advertisements, maximum of 7 characters, where XXXXX are the last 5 hex digits of the Bluetooth address	BCXXXXX	Yes
PARITY= (value)	UART Parity (1=none, 2=even, 3=odd)	1	No
PIN= (value)	Pin Code	0000	
REMOTE_ADDR= (value)	Address of device to auto-connect to on reset. Works only if devices are paired. Will attempt MAX_REC times and then go discoverable	0	Yes
RSSI_THRESH= (value)	Set threshold for Source RSSI pair/connection	-75	Yes
SPP_TRANSPARENT= (value)	Enable SPP Data Mode immediately on SPP connection. For multipoint this is only active on the primary link.	OFF	Yes
SSP_CAPS= (value)	Set SSP capabilities for this device. Accepted values are: 0 – display only 1 – display yes/no 2 – keyboard only 3 – no display, no keyboard (i.e. JustWorks) 4 – display and keyboard 5 – reject SSP request	3	No
TWS_MODE= (value)	Set the TWS mode. Mode can be: 0 – Auto (Master is Left Ch, Slave is Right Ch) 1 – Left Ch 2 – Right Ch 3 – Stereo	0	No
USB_HOST= (value)	Used to be able to send commands over USB (ON) instead of UART (OFF)	OFF	Yes
UUID_DATA= (value)	Change the default value for Melody Smart Data Characteristic UUID	89F7 D93C	Yes
UUID_SPP= (value)	Change the UUID for the SPP profile	00 00 11 01 00 00 10 00	Yes



Manual v5.7 RC8 MFI Rev A

		80 00 00 80 5F 9B 34 FB	
UUID_SRV=(value)	Change the default value for Melody Smart primary service UUID (BLE)	68E3 28F0	Yes
VREG_ROLE=(value)	Changes behaviour of VREG button when held (3s press) 0 – Nothing 1 – Power On/Off 2 – Switch role between Source/Sink	0	No
WIRED=(out_gain) [(in_gain) [(bias) [(preamp)]]]	Set the desired configuration for the Analogue interface when using WIRED routing. out_gain – must be in 0 – 15 in_gain – must be in 0 – 15 bias – must be 0 or 1 to disable/enable MIC bias on input lines preamp – must be 0 or 1 to disable/enable Mic Pre-Amp (+20dB) on input lines Only the first parameter is mandatory; all subsequent parameters are optional but must be provided in order. i.e. if you want to modify (bias) you have to supply (in_gain). Please note that these configurations will be loaded at the start of every wired routing and will not be applied while a routing is active.	8 8 0 0	No

Table 10: Configuration Parameters and values



Event Notifications

Melody uses 'notifications', or text prompts, to notify the host of events in the Bluetooth link (for example, completed commands or incoming connections), to provide information, or to require action. Notifications are generated only when Melody is in command or remote configuration mode.

The syntax used is **NOTIFICATION [link_ID] (Parameter)**.

Event	Description
ABS_VOL=[link_ID] (value)	The current absolute volume set as an integer in the range 0 – 127 (representing 0 – 100% as per spec).
AVRCP_MEDIA [link_ID] (property: value)	ARTIST: string TITLE: string ALBUM: string NUMBER: integer TOTAL_NUMBER: integer PLAYING TIME (MS): integer
AVRCP_PLAY [link_ID]	When an AVRCP play event is received from the remote device
AVRCP_STOP [link_ID]	When an AVRCP stop event is received from the remote device
AVRCP_PAUSE [link_ID]	When an AVRCP pause event is received from the remote device
AVRCP_FORWARD [link_ID]	When an AVRCP forward event is received from the remote device
AVRCP_BACKWARD [link_ID]	When an AVRCP backward event is received from the remote device
BC_SMART PROX (value)	BC Smart Proximity Service indicates that this is the closest (1) or not (0) device. If an indication of 0 is received the remote side may disconnect.
CALL (phone number) [link_ID]	In HFP mode, indicates that there is an incoming call from the supplied phone number. In AG mode, indicates that the handsfree wants to establish a call to this number.
CALL_ACTIVE[link_ID]	Indicates call is in progress.
CALL_END[link_ID]	Indicates call has been ended.
CALL_INCOMING [link_ID]	Indicates an incoming call.
CLOSE_ERROR (profile) [link_ID]	A connection could not be closed for the profile indicated.
CVC_CFG (TYPE) (KEY) [value]	TYPE – WB or NB for wideband and narrowband CVC keys. KEY is the key id 0 – 3; Value – is the key value in ASCII representation of 16bit hexadecimal words. Value will not be present if the key is empty.
ERROR	Error when a command has not been executed or the parameters are not correct
HANGUP [link_ID]	When HFP voice call is disconnected
INQUIRY (BDADDR) (COD) (RSI)	Returned after an INQUIRY command is used if there are discoverable devices nearby. Returns Bluetooth address, class of device and RSSI
LINK_LOSS [link_ID] (profile)	The link specified by link id and profile has suffered link loss
OPEN_ERROR (profile) [link_ID]	A connection has failed or there was a link loss for the profile indicated



Manual v5.7 RC8 MFI Rev A

OPEN_OK (profile) [link_ID]	A connection has been successfully established for the profile indicated
PAIR_ERROR (Bluetooth address)	When the module fails to pair with the remote device with the address specified
PAIR_OK (Bluetooth address)	When the module successfully pairs with the remote device with the address specified
PAIR_PASSKEY (BDADDRESS) (TYPE) [PASSKEY]	A SSP passkey request is being made. Depending on type these may require user action. TYPE: 0 – Passkey request. Enter passkey using the PASSKEY command 1 – Confirm passkey. Display passkey and have user confirm passkey using the PASSKEY command 2 – Display passkey. Display passkey for user. Note: the PASSKEY parameter will not be supplied for type 0.
PAIR_PENDING	When the module is starting to pair with a remote device
PXP LLA (level)	Proximity Profile, Link Loss Service alert level set to: 0 – No Alert; 1- Mild Alert, 2 – High Alert
PXM IMM (level)	Proximity Profile, Immediate Alert Service alert level set to: 0 – No Alert; 1- Mild Alert, 2 – High Alert
READY	Melody is ready to take commands
RECV IAP ([characters] or [size+data])	Data received from the IAP link. When in raw data mode, second parameter is raw data bytes in the form: {length_MSB, length_LSB, data_byte 0,...,data_byte(length-1)} There will be no trailing \n\r after a raw data receive packet.
RECV SPP [link_ID] (characters)	Data received from the SPP link
RECV BLE [link_ID] (characters)	Data received from the BLE link
RECV HID [size] (report data)	Data received from the HID link.
RING [link_ID]	Indicates that a call on HFP has sent a ring indicator (in-band ringing should be heard).
ROLE (BT ADDRESS) <role>	When the remove device changes the classic role, an indication event will be sent to the host with the remote device request a role change.
ROLE_OK <role>	Successful change of the classic role when the ROLE command was sent by the host.
ROLE_NOT_ALLOWED <role>	Failure to change the classic role. This is reached when the remote device refuses the role change requested by the host.
SCAN (BT ADDRESS) (<BT SHORT NAME> (ADVERTISING FLAGS) (RSSI)	Returned after calling SCAN and contains information about advertising devices. If the advertising data does not contain a short name, UNKNOWN will be displayed. The Advertising flags indicate device capabilities: 0x0A indicated this is a General Discoverable Dual Mode device (such as Melody), 0x02 indicates this is a General Discoverable Single mode device (such as Melody Smart).
SCO_OPEN	A SCO connection has been opened and audio is routed.



Manual v5.7 RC8 MFI Rev A

SCO_CLOSE	A SCO connection has been closed and audio is no longer routed.
SEND_ERROR	iAP 1 packet sent over the air but not acknowledged. Last packet sent was lost. Please resend of necessary. SPP packet could not be sent.
SEND_ERROR ACK (packet_id)	iAP 2 packet with packet_id failed. You must acknowledge this error message with the IAP_RESET command, and resend all packets starting with packet_id, as they have been lost.
SEND_ERROR IAP_APPEND	An error occurred while Melody was receiving raw data. This resets the data send session and you will need to stop sending raw data and resend. Most probably due to connection or data session loss.
SEND_ERROR IAP_SEND	All raw data has been successfully received, but Melody could not send data over the air. Most probably due to connection or data session loss.
SEND_PENDING (packet_id)	iAP: when all raw data has been received and packet is being sent over the air. This is the packet_id being used for the iAP protocol. For iAP 2, it is to be saved and used if and SEND_ERROR ACK message is received.
SIRI_STS=(status)	0 - Not available on this platform 1 - Available and enabled 2 - Available but not enabled
SEND_OK	Packet has been sent successfully

Table 11: Melody Notifications



Data Mode

When the module is connected, it will parse all received data, to identify commands. However, for SPP connections and in order to speed up data rate, the host can put the module into Data Mode. In Data Mode, the module becomes a transparent link and forwards any data received from the UART to the SPP connection. Melody no longer parses the incoming UART stream. To return to Command Mode, one has to stop sending data for a period of time equal to or greater than the set command timeout (CMD_TO) and send '\$\$\$\$' via the UART line, within the command timeout.

cVc and HFP 1.6 Wide Band Speech (WBS)

Melody 4.0 adds support for Clear Voice Capture (cVc) which is an algorithm provided by Cambridge Silicon Radio (CSR) for echo and noise cancellation and HFP 1.6 which adds Wide Band Speech (WBS) to supported codecs. cVc enabled is required for WBS to work. The commands for enabling cVc and WBS are in the parameters table. The use of cVc requires a license key. Please contact info@blue-creation.com or your distributor for more information about cVc.

Pairing

When discoverable, Melody will accept any pairing request from devices in range. By default Melody will use secure simple pairing (Bluetooth 2.1 and above). If the devices trying to pair are older (Bluetooth 2.0 and below), Melody will accept the older pairing procedure.

For Bluetooth 2.1 devices and newer Melody will accept any pairing request using the 'Just Works' Bluetooth 2.1 method. This means that the module will accept any connection from Smartphones and other devices. The user of the device will not be required to enter a PIN code.

For Bluetooth 2.0 devices and older Melody will require a PIN code to accept a connection. The PIN code is set to 0000 by default, but can be reconfigured. This means that the phone user will be required to enter the pin code in order to pair successfully.



Power management

Melody is configured to reduce the power consumption in both connected and not connected state.

Not Connected

By default, if Melody is not connected, it will be connectable and stays discoverable for a certain amount of time (configurable). If connection is lost, Melody will attempt to reconnect the number of times specified by the MAX_REC parameter. After that, and if Deep Sleep is enabled, Melody will go into Deep sleep mode (a very low power consumption mode). In this mode Melody will not be discoverable or able to accept any connection. This allows reducing current consumption to a minimum when the device is not connected and not operational. In order for Melody to get out of this mode, a random character has to be sent through the UART. After a 20ms wake-up time the module will be operational again. If GPIO commands are enabled, any GPIO change will also get Melody out Deep Sleep.

Note: Waking Melody up will not disable Deep Sleep and the module will go back into Deep Sleep after the command is executed. It is recommended if Deep Sleep is enabled, to always send a carriage return before a command. This will not return anything if Melody wakes up from Deep Sleep and will return ERROR if Melody is already awake. After sending the carriage return, wait for 20ms (the chip wake-up time) and then send the command. The chip will stay awake for 1s after the last byte is received on the UART.

Connected:

When Melody is connected, it is set-up to reduce power consumption while maintaining the connection. In this state, Melody will automatically establish a link policy to allow it to reduce power consumption while still being able to receive link updates.

Note: If Deep Sleep is enabled, Melody will go into deep sleep after 50ms of no activity.

Melody Source

Melody 5 adds support to source HFP-AG and A2DP source. This mode can be activated by setting the CLASSIC_ROLE=1 configuration to ON, writing, and resetting the device. Once Melody is reset, it will disable all profiles that are not supported in source mode.

In source mode, Melody can pair and connect with another Melody in sink mode or to regular headsets. It can then stream music over A2DP to the headset, send AVRCP commands for control, and start a voice call with the headset.



Melody Multipoint

Melody 5.2 supports Multipoint allowing two (2) connections for the HFP/AG, A2DP, AVRCP, and SPP profiles, along with a single connection over Bluetooth Smart.

To enable Multipoint Melody needs to be configured to be in `CLASSIC_ROLE=X` 1. X - any accepted value.

This forces all commands and notifications to use the `[link_id]` parameter, to allow for control of the two connection instances per profile. GPIO control will only work on the primary (lower number) connection, if established.

Melody TWS

Melody 5.2 includes True Wireless Stereo (TWS) support. TWS allows two Melody boards to connect to each other and share audio that is being streamed to one of them. The Melody boards can be configured to be Left, Right or Stereo channels. This allows the use case for true wireless stereo speakers over Bluetooth.

To enable TWS Melody needs to be configured to be in `CLASSIC_ROLE=X` 2 - any accepted value. If Melody is in `CLASSIC_ROLE=1` X, `CLASSIC_ROLE` will be reverted to `CLASSIC_ROLE=0` 2.

Please note that in TWS mode, volume controls are only available on the A2DP link to the streaming device.

DFU (Device Firmware Upgrade)

Melody versions above 4 RC2 support firmware upgrade. This can be done over UART from your PC using the Melody Device Firmware Upgrade Tool available from BlueCreation. Please <http://bluecreation.com/firmwareupdates.php> for more details and downloads.

You can use DFU to upgrade to a newer version of Melody and add additional features.



General Notes

- BlueCreation's products are not authorised for use in life-support or safety-critical applications. Use in such applications is done at the sole discretion of the customer. BlueCreation will not warrant the use of its devices in such applications.
- While every care has been taken to ensure the accuracy of the contents of this document, BlueCreation cannot accept responsibility for any errors. BlueCreation reserves the right to make modifications, corrections and any other changes to its products at any time. Customers should obtain the latest information before placing orders.
- BlueCreation, other products, services and names used in this document may have been trademarked by their respective owners. The publication of this information does not imply that any license is granted under any patent or other rights owned by BlueCreation.
- Refer to <http://www.bluecreation.com/> for more information.
- BlueCreation® is a trading name for Cambridge Executive Limited.



New features

- Melody 5.7 RC8 MFI
 - When GPIO disabled and iAP is the only active profile, PIO 0 will correctly indicate connected state
- Melody 5.7 RC8
 - Calling wired route modes twice in a row caused a panic. This has now been resolved, a routing will not be allowed if it is already active.
 - PCM interface configuration fixes: PCM interface operates at 16BPS, stereo, bit clock equal to $2 \times 16 \times F_s$, long sync is used with a sync length of 16 bit clocks.
 - Issuing AVRCP PLAY from remote device in CLASSIC_ROLE=1 1 now works as expected
 - Microphone bias is now correctly applied for WIRED A routing
 - BLE_AUTO_ADV was sometimes causing a panic due to a race condition; This has been fixed
- Melody 5.7 RC7
 - ROUTE command re-worked with routing syntax and logic changed and additional routing options introduced.
 - VOLUME command appended to work with WIRED_A and WORED_D routing; Manual entry re-structured to clarify syntax and behaviour.
- Melody 5.7 RC6
 - Fixed some issues with multiple A2DP connections when in source mode
 - Changed default I2S configuration parameters to be I2S compliant
 - Some fixes with tone generation/playback support
- Melody 5.7 RC5
 - Added Music Manager Support for Non-TWS mode. This is supported through the MM_CFG command and MM config
- Melody 5.7 RC4
 - Added the ability to send commands via USB
- Melody 5.7 RC3
 - Fix an issue where INQUIRY command was not finishing after the correct timeout.



Manual v5.7 RC8 MFI Rev A

- Melody 5.7 RC2
 - Revert A2DP DSP and plugin to an older commit since they contained some issues.
- Melody 5.7 RC1
 - Add BLE command mode to be able to send commands and receive replies from Melody. Currently the only command that returns other than "OK" and "ERROR" is the "GET" command
- Melody 5.6 RC5
 - Bug fixing and code clean-up
 - Fix a bug where power off was not working correctly.
- Melody 5.6 RC 3
 - When in HID host mode, the HID protocol will try to automatically reconnect in accordance with the AUTOCONN configuration parameter
- Melody 5.6 RC 2
 - In source mode, the CODEC config's sample rate controls the sample rate over the air. Valid values are 44100 Hz and 48000 Hz. If the value is different than 44100, the default of 48000 Hz is used.
 - Added new indicators for the SCO connection's state: SCO_OPEN and SCO_CLOSE
 - Added new indicators for the call state: CALL_INCOMING
 - Fixed existing indicators CALL_ACTIVE and CALL_END to reflect the actual call state
 - Added a new parameter to the ROUTE command: ROUTE_DIGITAL_{ON, OFF} which allows audio routing between the analogue inputs/outputs and a chosen digital interface
- Melody 5.5 RC 11
 - ROLE command to switch roles on classic Bluetooth
 - ROLE_OK and ROLE_NOT_ALLOWED events to notify the host of the result of the ROLE command
 - ROLE <BD ADDR> to notify the host when the remote device changed the role.
- Melody 5.5 RC 10
 - On non-MFI builds, PIO 3 will go high when and active audio connection is made (SCO or A2DP stream).



Manual v5.7 RC8 MFI Rev A

- Melody 5.5 RC 9
 - Add error handling to ensure BLE and HID are not enabled concurrently.
 - Add "PENDING" when "READ DESC" command is sent and "OK" when reply is received.
- Melody 5.5 RC 8
 - HID device support added.
- Melody 5.5 RC 7
 - HID host support added.
- Melody 5.5 RC 2
 - iBeacon support added
 - Proximity Service support added
 - BC Proximity Service support added
- Melody 5.5 RC 1
 - Device Information Service support added
- Melody 5.2 RC 1
 - Multipoint support
 - TWS support
 - Raw Data send for all profiles
 - Classic and Smart states displayed separately
 - LED indications updated to reflect new state structure
 - Configurable Secure Simple Pairing (SSP) support (out of band pairing not supported)
 - Microphone Bias control for both A2DP and HFP links
 - Access to CVC configuration keys
 - Timeout before entering Sniff mode for SPP is now configurable.
- Melody 5.0 RC11
 - Melody now goes into connectable mode (Classic) if all classic connections are closed, but a BLE connection remains open.
- Melody 5.0 RC10
 - Melody parser accepts all octet values. Melody SEND functions will accept all input other than backspace (0x08) and carriage return (0x0D). Those are currently reserved for deleting the last character in the input buffer and issuing a command.
 - UART now additionally supports 230400, 460800, 921200 bps.
 - Battery charging and threshold parameter configuration added (BATT_THRESH)
 - iOS battery indication enable/disable configuration added (ENABLE_BATT_IND)
 - Android 4.3 BLE support configuration added (ENABLE_ANDROID_SUPPORT)
 - SPP Sniff parameters are now also applied to iAP connections (on MFI builds)
 - Source pairing procedure configuration added ability to choose RSSI level (RSSI_THRESH)
- Melody 5.0 RC9
 - Added configuration to retrieve local Bluetooth address (LOCAL_ADDRESS)



Manual v5.7 RC8 MFI Rev A

- Melody 5.0 RC8
 - Added SPP Sniff parameter configuration (SPP_SNIFF)

Appendix A: Tone parameters and flags

Tones are defined as a sequence of notes described with pitch (N or TN) described through length (L), tempo (TE), loudness (V), timbre (TI) and decay (D). All of these parameters can be individually set for each note, changed at any place in the tone string or omitted (all except length) to use default values. Below you can find tables describing the different parameters and the flags used to set them and the values accepted.

Parameter	Flag	Accepted value and meaning
Tempo	TE	0 – 4095 Given in quarter notes (crotches) per minute. If no tempo is specified, default is 120.
Timbre	TI	0 – 7 Defines timber of following notes. Please see Table 9 for all available timbres. Default is 0 (Sine).
Volume	V	0 – 255 Sets the volume for the tone. Default is 255 (max).
Decay	D	00 – FF Takes a hexadecimal values that is interpreted as a fixed point decimal number according to this format 0000.0000. Eg 15 = 1.5. As each tone is played, its volume decreases with a variable rate. Low values for this parameter cause notes to decay very quickly, whereas high values cause the notes to continue with an almost constant volume. A value of 005 (meaning 0.5) will cause each note to reach zero halfway through its duration giving a staccato feel. A value of 20 (meaning 2.0) will cause each note to reach half its initial volume when the next note starts. The default value is 20. This allows notes of the same length to be tied together with TN.
Note	N	This describes a note pitch in the format: Note{F/S} [octave] Note is any note in A – G or R. R is used for a rest or pause. It has to be described for consistency with an octave eg R0.



Manual v5.7 RC8 MFI Rev A

		<p>The optional F or S following that indicates if the note is Flat or Sharp. Note, not all notes have flat and sharp variants.</p> <p>The mandatory octave parameter gives the note octave. It takes values between 0 – 9.</p> <p>A note is describes as G4 or AS7 for example.</p> <p>Each note must have a length parameter associated with it and following immediately after:</p> <p>G4 L 1</p>
Tied Note	TN	<p>Ties note to next one. This means that the pitch of the note appears to change (as opposed to a new note starting), and the volume continues to decay from the previous note. An appropriate decay must be set for this to work.</p> <p>Tied Notes are described in the same manner as Notes.</p>
Length	L	<p>Ringtone duration.</p> <p>Please look at Table 10 for details.</p>

Table 12: TONE parameter detailed description

Timbre Value	Timbre Description
0	Sine wave
1	Square wave
2	Sawtooth wave
3	Triangle wave
4	Triangle wave (asymmetric)
5	Clipped sine wave
6	Simulates a plucked instrument

Table 13: Timbre value and description

Length Values	UK Notation	American Notation
1	Semibreve	Whole note



Manual v5.7 RC8 MFI Rev A

2	Minim	Half note
4	Crotchet	Quarter note
8	Quaver	Eighth note
16	Semiquaver	Sixteenth note
32	Demisemiquaver	Thirty-second note
64	Hemidemisemiquaver	Sixty-fourth note
3	Minim triplet	Half note triplet
6	Crotchet triplet	Quarter note triplet
12	Quaver triplet	Eighth note triplet
24	Semiquaver triplet	Sixteenth note triplet
48	Demisemiquaver triplet	Thirty-second note triplet
96	Hemidemisemiquaver triplet	Sixty-fourth note triplet

Table 14: Note length values and description



Appendix B: Use case example:

1. Discovering and pairing with devices

Discovering a phone from Melody and initiating pairing:

Make sure BT is on and the phone is discoverable if you want to be able to see it in Melody. Look for discoverable devices.

```
>INQUIRY 15
```

Melody will return a list of discoverable device Bluetooth addresses and device classes.

```
>INQUIRY {BT ADDRESS} {DEVICE CLASS} {RSSI}
```

If you do not know your phone's BT address, but you know its name you can have Melody resolve device names.

```
>NAME {BTADDRESS}
```

Melody will return:

```
>NAME {BT ADDRESS} {DEVICE NAME}
```

Keep doing that until you find the correct device. Take note of its BD address. Type in:

```
>OPEN {BD ADDRESS} {PROFILE}
```

This will automatically pair Melody and the device and open the respective profile connection. Have in mind that pairing is sometimes linked to specific profiles; therefore it is best if you open all profiles that you want to use with the given device while the two devices are still in pairing stage.

Melody will respond with a status `OPEN_OK`/`OPEN_ERROR` for each opened profiles.

Discovering Melody from a phone and initiating pairing:

Use status command to make sure Melody is discoverable

```
>STATUS
```

If not, type:

```
>DISCOVERABLE ON
```

Now look for devices from your mobile phone. Select and initiate pairing with Melody. This should also automatically connect the phone to Melody for all available profiles.

Melody will display `OPEN_OK` /`OPEN_ERRPR` confirmations for every profile connection opened.



Manual v5.7 RC8 MFI Rev A

2. Tones

The Tone command is used to play back tones on the Melody/BC127 audio output. Tones are mixed with any ongoing audio playback.

These are some example tones that demonstrate some of the Tone command capabilities:

```
>TONE TE 400 V 64 TI 0 N C5 L 8 N R0 L 32 N E5 L 8 N R0 L 32 N G5 L 8 N R0 L 32 N  
B5 L 4 N R0 L 1 N C6 L 2 TN C6 L 8
```

A sample tone is played back.

```
>TONE V 64 N C6 L 4 V 128 N C6 L 4 V 255 N C6 L 4 V 128 N C6 L 4 V 64 N C6 L 4
```

A variable volume single note is played back.

```
>TONE V 128 TI 0 N C4 L 8 N D4 L 8 N E4 L 8 N F4 L 8 N G4 L 8 N A4 L 8 N B4 L 8 N  
C5 L 8
```

A musical scale starting from C4 and omitting any flat and sharp tones is played.



Manual v5.7 RC8 MFI Rev A

3. Music (A2DP & AVRCP)

This use case assumes devices are paired.

If A2DP and AVRCP are not connected open those profiles now. This can be done either by connecting to Melody from the phone or by connecting from Melody:

```
>OPEN {BT ADDRESS} A2DP  
>OPEN {BT ADDRESS} AVRCP (if supported)
```

Initiate playback on phone.

Now use Melody to control playback, if AVRCP is open and supported:

```
>MUSIC PAUSE pauses playback  
>MUSIC PLAY resumes playback  
>MUSIC FORWARD plays next song  
>MUSIC BACKWARD starts current song from beginning or if repeated in quick succession plays previous song  
>VOLUME UP increases Melody DAC volume  
>VOLUME DOWN decreases Melody DAC volume  
>MUSIC STOP stops playback
```

Note: On phones that support AVRCP 1.3 and newer song information is displayed



Manual v5.7 RC8 MFI Rev A

4. HFP & PBAP

This use case assumes devices are paired.

If HFP and PBAP (if supported) are not connected open those profiles now. This can be done either by connecting to Melody from the phone or by connecting from Melody:

```
>OPEN {BT ADDRESS} HFP  
>OPEN {BT ADDRESS} PBAP
```

Making a call:

```
>CALL {NUMBER}
```

To accept or reject an incoming call

```
>ANSWER or >REJECT
```

To end any call:

```
>END
```

To pull the phonebook from the phone

```
>PULL_PBOOK
```

Note: Pulling the phonebook will just display phonebook information. Melody will not store and cannot utilize this data in any way.



Manual v5.7 RC8 MFI Rev A

5. Data transfer (SPP)

This use case assumes devices are paired.

Note: a special application on the phone is needed in order to use SPP and connection to SPP needs to be initiated from that application.

To send data to the phones type:

```
>SEND {CHARACTERS}
```

Whenever data is received from the phone it will be displayed as follows:

```
>RECV SPP {character}
```

Data mode can be used to increase data throughput.

To enter data mode, type:

```
>ENTER_DATA
```

Everything sent over UART will then immediately be forwarded to SPP and vice versa.

To exit DATA mode, type:

```
>$$$$
```

In ENABLE_RAW_DATA mode:

```
>SEND 0 {size}
```

```
<OK
```

```
>{raw data of length 'size' bytes}
```

```
<OK
```



Manual v5.7 RC8 MFI Rev A

6. Data transfer (IAP)

This use case assumes devices are paired.

Note: You will need a corresponding application on the iOS device to handle the data connection.

To send data to the phones type:

```
>SEND {CHARACTERS}
```

Whenever data is received from the phone it will be displayed as follows:

```
>RECV IAP {character}
```



Manual v5.7 RC8 MFI Rev A

7. Data transfer (BLE)

Note: For BLE: a special application on the phone is needed to scan for, discover, connect and interact with the Melody service to transfer data.

To send data to the phones type:

```
>SEND {CHARACTERS}
```

Whenever data is received from the phone it will be displayed as follows:

```
>RECV BLE {character}
```

In ENABLE_RAW_DATA mode:

```
>SEND 12 {size}
```

```
<OK
```

```
>{raw data of length 'size' bytes}
```

```
<OK
```




Manual v5.7 RC8 MFI Rev A

8. HFP-AG

Set Melody to source mode and reset to start the new mode.

```
>SET CLASSIC_ROLE=1
```

```
>WRITE
```

```
>RESET
```

Open a connection with a headset using the HFP-AG profile

```
>OPEN {BT ADDRESS} AG
```

Make a voice call and open audio

```
>CALL
```

To end any call:

```
>END
```



Manual v5.7 RC8 MFI Rev A

9. Music Source (A2DP & AVRCP)

This use case assumes Melody is in source mode and devices are paired. Melody has a music player connected to its stereo input jack.

If A2DP and AVRCP are not connected open those profiles now. This can be done either by connecting to Melody from the phone or by connecting from Melody:

```
>OPEN {BT ADDRESS} A2DP  
>OPEN {BT ADDRESS} AVRCP (if supported)
```

Use Melody to control playback:

```
>MUSIC PLAY starts playback  
>VOLUME UP increases Melody ADC volume  
>VOLUME DOWN decreases Melody ADC volume  
>MUSIC STOP stops playback
```

Issuing AVRCP commands on the headset will result in an event, and in case of AVRCP_PLAY and AVRCP_STOP, an immediate effect will occur alongside the event.



Manual v5.7 RC8 MFI Rev A

10. BLE Central Role

This use case shows how Melody can be used in the BLE Central role.

Enable BLE Central role:

```
>SET BLE_ROLE=2
```

```
>WRITE
```

```
>RESET
```

On the partner device(s), issue the ADVERTISING ON command.

Scan for BLE-enabled Melody devices:

```
>SCAN 10
```

Melody will return a list of discovered devices:

```
>SCAN {BT ADDRESS} <{BT SHORT NAME}>{ADVERTISING FLAGS} {RSSI}
```

```
> ...
```

```
>OK
```

Open a BT connection to a selected device:

```
>PEN {BT ADDRESS} BLE
```

```
>OPEN_OK BLE
```

Melody connects to the partner's Melody service and is ready for data exchange.

```
>SEND hello
```

```
>OK
```

On message reception, a notification is issued:

```
>RECV BLE hello
```



Manual v5.7 RC8 MFI Rev A

11. HID host

This use case assumes devices are paired.

First ensure hid is enabled and data can be sent:

```
> SET ENABLE_RAW_DATA=ON
> SET BLE_ROLE=0
> SET HID_ROLE=1
>WRITE
>RESET
```

To open a hid connection to a hid device:

```
> OPEN {BT ADDRESS} HID
```

To read the USB descriptor from the remote HID device:

```
> READ DESC {BT ADDRESS}
```

When HID reports are received over hid, an event will be received:

```
>RCV HID {size} {data}
```

Size is the number of octets of binary data following. Melody simply outputs the received data without processing. It is the responsibility of the host to handle it.

To send data:

```
>SEND 16 {size}
> {binary data}
```

16 is the connection id for the first hid connection. size is the length of the following binary data. Similarly to received data, any data sent is not parsed or processed by Melody, only sent to the remote device.

The connection status can be viewed using status:

```
> STATUS
```

The connection can be closed using close:

```
> CLOSE 16
```



Manual v5.7 RC8 MFI Rev A

12. HID device

This use case assumes devices are paired.

First ensure hid is enabled and data can be sent:

```
> SET COD=000540
> SET ENABLE_RAW_DATA=ON
> SET BLE_ROLE=0
> SET HID_ROLE=2
>WRITE
>RESET
```

This will make Melody, on pairing with the remote device, enumerate as a keyboard.

By default, the usb descriptor is set to behave as a simple keyboard.

To set the usb descriptor:

```
>DESC {size}
> {binary data}
```

Size is the number of octets of binary data following.

When HID reports are received over hid, an event will be received:

```
> RECV HID {size} {data}
```

Size is the number of octets of binary data following. Melody simply outputs the received data without processing. It is the responsibility of the host to handle it.

To send data:

```
>SEND 16 {size}
> {binary data}
```

16 is the connection id for the first hid connection. size is the length of the following binary data. Similarly to receive data, any data sent is not parsed or processed by Melody, only sent to the remote device.

The connection status can be viewed using status:

```
> STATUS
```

The connection can be closed using close:

```
> CLOSE 16
```



Manual v5.7 RC8 MFI Rev A

Examples:

Keyboard:

Melody act by default as a simple Keyboard, here are some examples of HID report that you can send:

- Pressing 'a':

```
> SEND 16 8
```

```
> 000004000000000023
```

- Release key:

```
> SEND 16 8
```

```
> 0000000000000000
```

Mouse:

If you want to use Melody as a Mouse, you have to change the COD (000580) and set a different USB descriptor. Here is an example:

```
> DESC 71
```

```
>05010902A1010901A10005091901290815002501950875018102950081030501093009311601F826  
FF07750C9502810609381581257F750895018106050C0A380295018106C0C0
```

Here are some HID reports that you can send:

- Move the mouse vertically:

```
> SEND 16 6
```

```
> 000030000000
```

- Left-Button press:

```
> SEND 16 6
```

```
> 010000000000
```

- Button release:

```
> SEND 16 6
```

```
> 000000000000
```

²³ Please note that the binary data are hexadecimal values in those examples



Manual v5.7 RC8 MFI Rev A

13. Music Manager Example configurations

Melody ships with generic example configurations for the Music Manager.

The pre-set equalizer banks give the following settings:

- 0 - Flat response
- 1 - Bass boost
- 2 - Treble boost
- 3 - Rock
- 4 - Jazz
- 5 - Bass boost (identical to 1)
- 6 - Treble boost (identical to 2)

The other enhancement blocks are pre-set to the following:

Bass boost - +6dB for frequencies under 400Hz

3d Enhancement - 300 sample delay path modelling

Compander - 2:1 Compress from -20dB, Hard Limit from -6dB

Dithering - Noise shaped dithering

Speaker Eq - flat response

Enable Music Manager and set equaliser to Rock

```
> SET MM=1 1 3
```

Enable Music Manager and set equaliser to Rock and add 3d Enhancement

```
> SET MM=1 1 3 0 1
```

Enable Music Manager and set equaliser to Rock and add Bass boost and 3d Enhancement

```
> SET MM=1 1 3 1 1
```

All Music Manager configurations can be modified using CSR UFE from ADK 2.0 following the procedures outlined in CSR documentation.



Manual v5.7 RC8 MFI Rev A

14. Routing example

Melody allows users to select which audio stream to send to the configured audio output.

This example assumes Melody is configured for multi-point mode (CLASSIC ROLE=0 1), analogue audio output is used (AUDIO=0) and that two phones have been paired and connected and streaming has been started on both phones.

Melody will be routing the first started stream by default. You can check current routing:

```
> ROUTE
< ROUTE=2
```

This indicates that the phone that connected first started the stream first and is routed to the audio output

To switch to the other phone:

```
> ROUTE 3
< OK
```

You should immediately hear the new audio stream on the output

To switch audio routing to Analogue Loopback mode (WIRED_A):

```
> Route 40
< OK
```

This will output whatever signal is present on the analogue audio input to the analogue output. By default, the analogue input is configured as a line input.

To change wired mode settings, please use the WIRED command.

If you wish to for stop any audio from being routed:

```
> ROUTE 0
< OK
```

This will disconnect any current routing from the output.

Please note that since Melody does auto routing, any subsequent incoming Bluetooth audio event (new stream or new call) will be routed according to the output.

Also, please note that WIRED routing modes are overrides and will stop auto routing events from routing.